3-D Seismic Imaging

Biondo L. Biondi - Stanford University

© September 17, 2004

Contents

0.1 2-D vs. 3-D: more accurate imaging 3

0.2 2-D vs. 3-D: extra information 10

0.3 2-D vs. 3-D: new challenges 11

1 3-D data geometries **13**

1.1 Data coordinates 14

1.2 Marine-data geometries 16

1.2.1 Streamer geometries 16

1.2.2 Parallel-swath geometries (OBC and OBS) 18

1.3 Land-data geometries 19

1.3.1 Cross-swath geometries 19

1.3.2 Button-patch geometries 21

1.4 Narrow azimuth vs. wide azimuth 21

1.5 Sorting and binning 24

2 Full prestack migration by Kirchhoff's methods **27**

2.1 Constant-velocity migration 27

2.2 Migration in complex media 32

2.2.1 Time vs. depth imaging 35

2.3 Computational cost of prestack migration 38

2.3.1 Limited-aperture prestack migration 38

3 Approximations of full prestack migration **43**

3.1 Normal moveout 44

3.1.1 Stacking velocity from dipping reflectors 47
3.2 Dip moveout (DMO) 50
3.2.1 Computational cost of poststack imaging 51
3.2.2 Poststack imaging of SEG-EAGE salt data set 52
3.3 Azimuth moveout 55
3.3.1 Geometric derivation of the AMO saddle 57
3.3.2 Application to coherent partial stacking 59
3.3.3 Transformation to common-azimuth data 63
3.3.4 Prestack imaging after partial stacking of SEG-EAGE salt data set . . 63
3.4 Two-pass 3-D prestack migration 65
3.4.1 Geometric interpretation of two-pass migration 72
3.4.2 Two-pass migration of SEG-EAGE salt data set 73

4 Principles of wavefield-continuation migration 77
4.1 Reverse-time migration 80
4.2 Downward-continuation migration 85
4.2.1 Shot-gather migration by downward continuation 86
4.2.2 Source-receiver migration 87
4.2.3 Equivalence of source-receiver migration and shot-profile migration . 89

5 Downward-continuation methods 95
5.1 Frequency-wavenumber (ω-k) domain methods 95
5.2 Mixed frequency-wavenumber/space (ω-k/ω-x) methods 96
5.2.1 Split-step migration 96
5.2.2 Higher-order mixed ω-k/ω-x methods 99
5.3 Frequency-space (ω-x) methods 108
5.3.1 Splitting methods for 3-D downward continuation 108
5.3.2 McClellan downward continuation 113

6 Common Image Gathers 123
6.1 Common Image Gathers by Kirchhoff migration 124

6.1.1 Offset-Domain Common Image Gathers 124
6.1.2 Angle-Domain Common Image Gathers 125
6.1.3 Artifacts in Common Image Gathers by Kirchhoff migration 128
6.2 Angle-Domain Common Image Gathers by wavefield-continuation 133
6.2.1 ADCIG before imaging – ADCIG$\left(p_{x_h}, p_{y_h}\right)$ 135
6.2.2 ADCIG after imaging – ADCIG(γ, ϕ) 141
6.2.3 Examples of 3-D ADCIG(γ, ϕ) 145

7 Efficient wavefield-continuation methods for prestack migration 151
7.1 Migration of synthesized common-source gathers 153
7.1.1 Phase-encoding migration 154
7.1.2 Plane-wave migration 156
7.2 Source-receiver migration in midpoint-offset coordinates 161
7.2.1 Offset plane-wave downward-continuation and migration 162
7.2.2 Common-azimuth downward-continuation and migration 163
7.2.3 Narrow-azimuth downward-continuation and migration 172

8 Imaging and aliasing 181
8.1 Aliasing fundamentals 182
8.1.1 The anti-aliasing dilemma 184
8.2 Aliasing in Kirchhoff migration 186
8.2.1 Aliasing in image space 186
8.2.2 Operator aliasing 191
8.3 Aliasing in wavefield-continuation migration 200
8.3.1 Aliasing of wavefield-continuation imaging operators 204
8.3.2 Aliasing in image space 212

9 Imaging and partial subsurface illumination 215
9.1 Equalization of imaging operators 217
9.1.1 Least-squares inverses 220
9.1.2 Diagonal approximations of least-squares inverses 222

9.2 Filling illumination gaps by model regularization . . . 229
9.2.1 Regularization by imposing continuity over offsets and azimuths . . . 233
9.2.2 Regularization by AMO . . . 234
9.2.3 Imaging of a 3-D land data set . . . 235
9.3 Regularized inversion of prestack migration . . . 242

10 Principles of velocity analysis **253**
10.1 Flat reflectors in $V_{rms}(\tau)$ media . . . 255
10.2 Dipping reflectors in $V_{rms}(\tau)$ media . . . 256
10.3 Dipping reflectors in smooth $V_{rms}(\tau, x, y)$ media . . . 259
10.4 Traveltime reflection tomography . . . 263
10.4.1 Formalization of traveltime tomography as an inverse problem . . . 265
10.4.2 Evaluation of the linearized reflection-tomography operator . . . 266
10.4.3 Tomographic inversion of stacking velocities . . . 270
10.4.4 Multi-azimuth velocity estimation . . . 274

11 Migration Velocity Analysis **281**
11.1 Time-migration velocity analysis . . . 283
11.2 Extracting velocity information from prestack images . . . 285
11.2.1 Residual Moveout (RMO) Analysis . . . 285
11.2.2 Residual prestack migration . . . 298
11.3 Vertical interval-velocity updates from measured average-velocity errors . . . 301
11.4 Tomographic migration velocity analysis . . . 307
11.4.1 Objective function of tomographic migration velocity analysis . . . 307
11.4.2 Evaluation of the linearized tomographic-MVA operator . . . 309
11.4.3 Example of 3-D tomographic MVA . . . 310
11.4.4 Improving the convergence of tomographic MVA . . . 311

12 Migration Velocity Analysis by wavefield methods **327**
12.1 Objective function of wavefield migration velocity analysis . . . 329
12.2 Linearization of wave propagation with respect to the velocity function . . . 334

12.2.1 Application of $\mathbf{M}_w$ and $\mathbf{M}'_w$ by downward continuation 336
12.3 Convergence of wavefield migration velocity analysis 342
12.4 A robust WEMVA method 345
12.5 Examples of subsalt wave-equation MVA 351
12.5.1 Synthetic example 352
12.5.2 Gulf of Mexico data example 356

1 Seplib3d: a software package for processing 3-D data 365
1.1 Data Format 366
1.1.1 Structure of a Seplib3d data set 366
1.1.2 Data and Headers Coordinate Systems 367
1.1.3 Mapping between the header records and the data records 367
1.1.4 Gridding information 368
1.1.5 Standard Header Keys 369
1.2 Utilities 369
1.3 ACCESSOR ROUTINES 370
1.3.1 Accessors to coordinate system parameters 370
1.3.2 Accessors to header formats 370
1.3.3 Accessors to header values 371
1.3.4 Headers navigation routines 371
1.3.5 Accessors to file links 371
1.4 3-D prestack example 371
1.4.1 Exploring the data 372
1.4.2 Analyzing binning parameters 373
1.4.3 Efficient processing 374

2 The SEG-EAGE salt data set - (*to be written*) 381

Index 383

Preface

In the past decade, 3-D reflection seismology has almost entirely replaced 2-D seismology in the seismic industry. Recording 3-D surveys has become the norm, instead of the exception. The application of 2-D seismology is mostly limited to reconnaissance surveys, or to where recording 3-D data is still prohibitively expensive, such as in rough mountains and wild forests. However, academic research and teaching have struggled to keep up with the 3-D revolution. As a consequence of this tardiness, there are no books available that introduce the theory of seismic imaging from the 3-D perspective. This book is aimed at filling the gap.

Seismic processing of 3-D data is inherently different from 2-D processing. The differences begin with data acquisition: 3-D data geometries are considerably more irregular than 2-D geometries. Furthermore, 3-D acquisition geometries are never "complete" because sources and receivers are never laid out in dense areal arrays covering the surface above the target. These fundamental differences, together with the increased dimensionality of the problem, strongly influence the methods applied to process, visualize, and interpret the final images. Most 3-D imaging methods and algorithms cannot be derived from their 2-D equivalent "just by adding a couple of dimensions to the 2-D equations". This book introduces seismic imaging from the 3-D perspective, starting from a 3-D Earth model. However, because the reader is likely to be familiar and comfortable already with 2-D processing methods, whenever useful I discuss the connections between 3-D algorithms and the corresponding 2-D algorithms.

The book attempts to cover all the important aspects of 3-D imaging. It links the migration methods with data acquisition and velocity estimation, because in practice they are inextricably intertwined. Data geometries strongly influence the choice of 3-D imaging methods. At the beginning of the book, I present the most common acquisition geometries, and continue to discuss the relationships between imaging methods and acquisition geometries throughout the text. The imaging algorithms are introduced assuming regular and adequate sampling. However, two chapters (Chapter 8 and Chapter 9) explicitly discuss the problems and solutions related to irregular and inadequate spatial sampling of the data.

Velocity estimation is an integral component of the imaging process. On one hand, to create a good image we need to provide a good velocity function to the migration process. On the other hand, in complex areas velocity is estimated by iterative migration and velocity updating. Migration methods are presented first, because they provide the basic understanding necessary to discuss the velocity updating process.

Seismic imaging algorithms can be divided in two broad categories: integral methods (e.g.

Kirchhoff methods) and wavefield-continuation methods. Integral methods can be described by simple geometrical objects such as rays and summation surfaces. They are thus more easily understood by intuition than the wavefield-continuation methods. My introduction of the basic principles of 3-D imaging exploits the didactical advantages of integral methods. However, wavefield-continuation methods can yield more accurate images of complex subsurface structures. The book introduces wavefield-continuation imaging methods by leveraging the intuitive understanding gained during the study of integral methods. Wavefield-continuation methods are the subject of my on-going research and that of my graduate students. Therefore, the wavefield-continuation methods described are more advanced, though less well established, than the corresponding integral methods.

Seismic imaging technology is data driven, and the book contains many examples of applications. The examples both illustrate the rationale of the methods, and expose their strengths and weaknesses. The data examples are drawn both from real data sets and from a realistic synthetic data set: the SEG-EAGE salt data set. This data set is distributed freely and is widely used in the geophysical community. For the reader's convenience a subset of this data set (known as C3 Narrow-Azimuth) is contained in the DVD included with the book. Appendix 2 briefly describes this data set.

The software needed to produce all the examples is also freely distributed (either in DVD and over the Internet). A reader with the necessary computer equipment (a powerful Unix workstation) and the patience to wait for weeks-long runs, should be able to reproduce the images obtained from the the SEG-EAGE salt data set. Appendix 1 describes the foundations of Seplib3D, the main software package that is needed to generate most of the results shown in the text.

The book starts from the introduction of the basic concepts and methods in 3-D seismic imaging. To follow the first part of the book, the reader is only expected to have an elementary understanding of 2-D seismic methods. The book can thus be used for teaching a first-level graduate class as well as a short course for professionals. The second part of the book covers more complex topics and recent research advances. This material can be used in an advanced graduate class in seismic imaging.. To facilitate the teaching of the material in this book, the attached DVD includes a document in PDF format that has been specifically formatted to be electronically projected during a lecture. Many figures in this electronic document can be animated, thus providing a more cogent illustration of the concepts described in the text. Thanks to Bill Schneider fro correction of RMO function

I would like to thank the following companies and organizations (in alphabetical order) for providing the data sets used in this book: Bill Symes and the TRIP project at Rice University for the synthetic data set used in Chapter 4, and 6, BP (Arco) for the button-patch data geometry used in Chapter 1, BP and ExxonMobil for the deep water Gulf of Mexico data set used in Chapter 9, ChevronTexaco for the Gulf of Mexico data set used in the introduction, ChevronTexaco for the OBC data geometry used in Chapter 1, ConocoPhilips and ExxonMobil for the North Sea data set used in the Introduction and Chapters 3, 9, and 1, Ecopetrol for the South America data set used in Chapter 9, Husky Oil for the cross-swath data geometry used in Chapter 1, IFP for the Marmousi data set used in Chapter 8, SEG and EAGE for the Salt

data set used in all the chapters of the book, SMART JV and its partners for the Sigsbee data set used in Chapter 9, Total for the North Sea data set used in Chapters 4, 9 and 11, Unocal for the Gulf of Mexico OBC data set used in the Introduction and Chapters 5, 8.

Introduction

The driving force behind the fast switch from 2-D seismology to 3-D seismology is the dramatic increase in information and accuracy obtained by 3-D seismic images, compared with the images obtained by 2-D seismic. This increase is substantial not only for structures that are obviously three-dimensional, such as the Gulf of Mexico salt dome shown in Figures 1 and 2, but also for areas where the geology has a predominant dip-direction, such as the North Sea data shown in Figures 6, 7, 8, and 9. Furthermore, at the reservoir scale the Earth is seldom two-dimensional. Only 3-D imaging can provide the detailed knowledge of small-scale reservoir features, such as small faulting and thin channeling. This knowledge is often crucial to an effective exploration and production strategy. Figure 10 shows an example of a complex channel system that would have been be undetected without 3-D seismic.

The economical value provided by the quality and quantity of available information is compounded with a substantial reduction of the costs related to 3-D seismic acquisition and processing. This technological trend is continuing unabated and it is likely to continue for the foreseeable future, as 3-D seismic becomes more widespread and achieves even larger economy of scale.

In this introduction, we will explore the differences between images obtained from 2-D seismology and 3-D seismology and gain an understanding of the causes of these differences and of the new challenges presented by 3-D seismic imaging.

0.1 2-D vs. 3-D: more accurate imaging

The improvements in imaging the subsurface assured by 3-D seismology over 2-D seismology are well illustrated by the images shown in Figures 1 and 2. The figures compare the results of applying 3-D and 2-D post-stack depth migration to a zero-offset cube (Chapter 2). The sections on the right show the results of full 3-D migrations of the data; the sections on the left show the results of performing independent 2-D migrations on each in-line section (horizontal direction in Figure 1-2–D). If we had available only 2-D data collected along any of the in-lines, we would have obtained the distorted images of the subsurface corresponding to the in-line sections extracted from the 2-D migrated cube (e.g., Figure 2-2–D). The depth slices shown in Figure 1 show how the shape and extent of the salt dome would have been grossly mis-interpreted from 2-D images. If the only image available were the 2-D in-line section shown in Figure 2-2–D, the interpreter would have positioned a salt dome below the subsurface

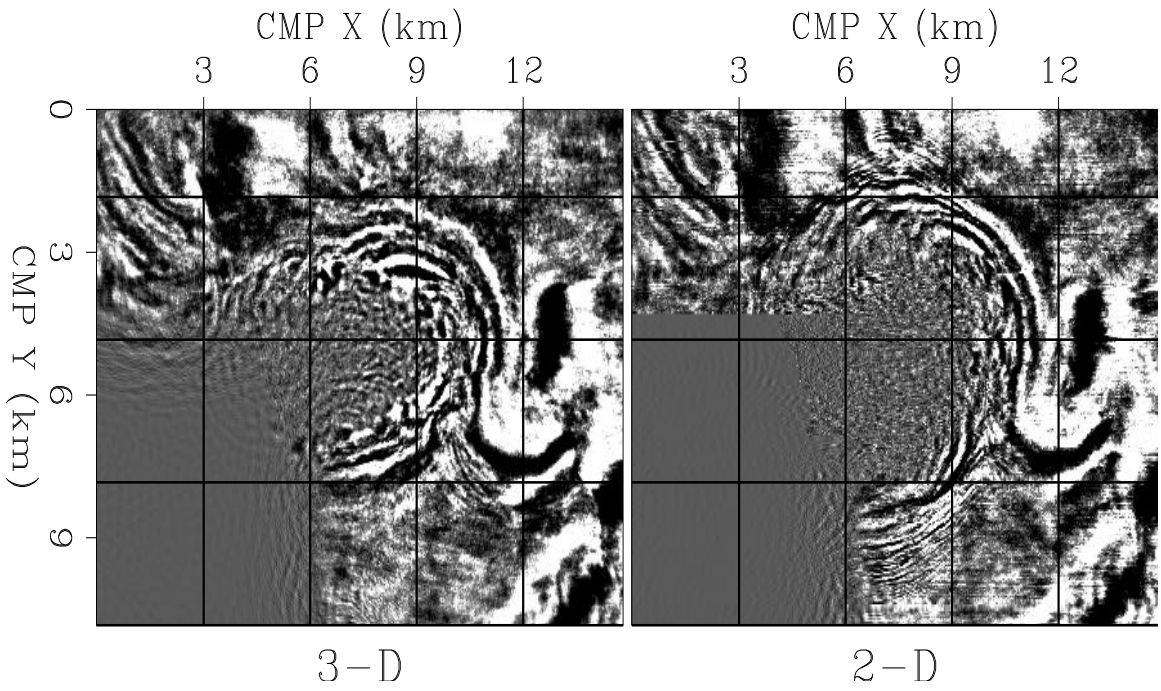


Figure 1: Depth slices of a Gulf of Mexico salt dome: 3-D (left) vs. 2-D (right). The slice is cut at Depth = 3.63 km. intro-z11050uno [NR]

at a location where only sediments are present in reality (Figure 2-3–D).

When we acquire a 2-D seismic survey, we implicitly assume that the Earth is a cylinder, with its axis in the direction orthogonal to the survey. If this underlying assumption is fulfilled, the image obtained by 2-D seismic is an accurate representation of a vertical section of the subsurface. However, when this assumption is not fulfilled, 2-D seismic produces a distorted image of the subsurface. The schematic in Figure 3 shows the main reasons why 2-D seismic produces distorted images when imaging 3-D structures. The point diffractor R_{3D} is out-of-the-plane with respect to the 2-D acquisition direction. It creates reflections that are incorrectly back-propagated in the Earth along the vertical plane passing through the acquisition direction, and that are imaged at the wrong location R_{2D}. To image the diffractor at its correct location R_{3D}, we need to apply 3-D imaging for back-propagating the recorded reflection along an oblique plane. However, even after applying 3-D imaging methods to a single 2-D line, we would be left with the ambiguity of where to place the diffractor along the semi-circular curve perpendicular to the acquisition direction, marked as *Cross-line migration* in the figure.

To resolve this ambiguity, we need to estimate the cross-line component of the propagation direction. This estimate is possible only by measuring the reflection **step-out** along the cross-line direction, that is, by recording 3-D data. Figure 4 and Figure 5 schematically illustrate this concept. Seismic imaging algorithms rely on the difference in the arrival times of the same event recorded by nearby traces (time step-out) to estimate the event propagation direction. The nearby traces may correspond to different receiver locations (G in the figures), or different source locations (S in the figures). This estimate is usually implicit in the

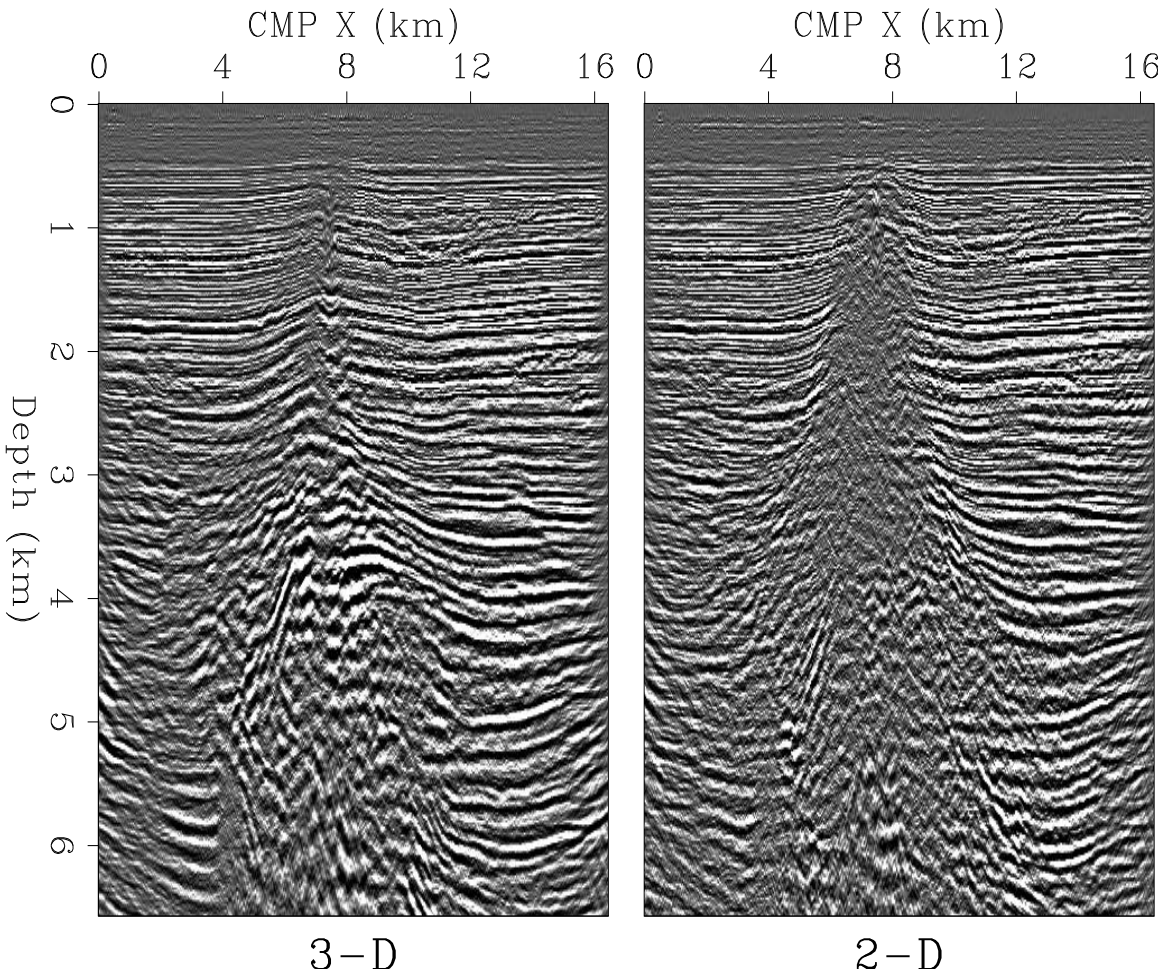


Figure 2: In-line sections of a Gulf of Mexico salt dome: 3-D (left) vs. 2-D (right). The section is cut at CMP Y = 3.25 km. intro-y9900uno [NR]

imaging algorithms, but nevertheless it is crucial to the imaging process. If data are recorded along only one direction (Figure 4), the only information available is about the angle that the propagation direction makes with the recording direction. No information is available on the cross-line angle. Conversely, if data are recorded along both horizontal directions (Figure 5), the propagation direction can be fully resolved. Notice that, at least in principle, for the imaging problem to be fully determined the wavefield needs to be properly sampled at both the source and receiver locations. Unfortunately, this is rarely the case with practical acquisition geometries.

When 2-D imaging is performed, the error in positioning the diffractor has both a cross-line direction component Δy, and a depth component Δz. The reflector R_{2D} needs to be moved along a semicircle in the cross-line direction to be placed to its correct position R_{3D}. In the following we will see that, at least when the velocity is constant, this semicircular trajectory corresponds to the semi-circle of a zero-offset cross-line migration. The analytical expression of this semicircle is independent of velocity and it depends only on the apparent depth ρ of

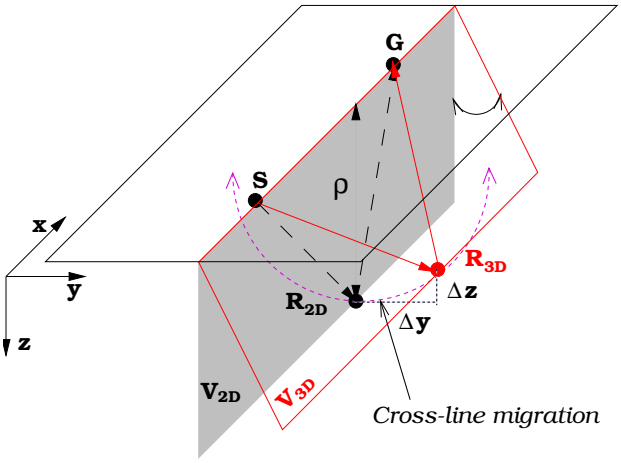


Figure 3: 2-D vs. 3-D imaging of an out-of-plane diffractor. intro-outofplane [NR]

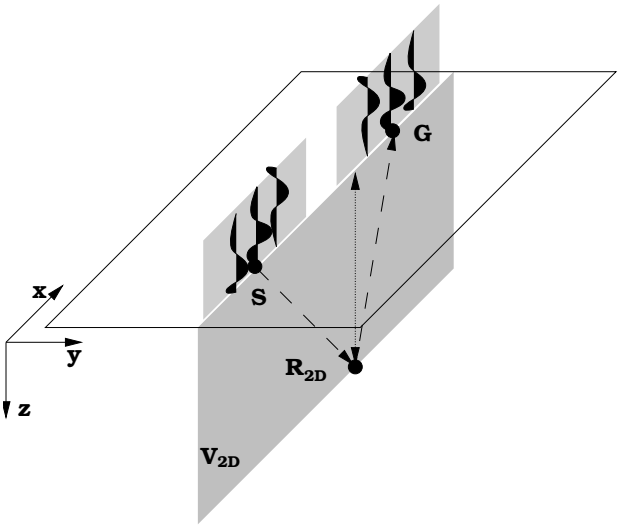


Figure 4: 2-D acquisition and imaging. Only the in-plane propagation direction can be determined from 2-D data. intro-2D-aq [NR]

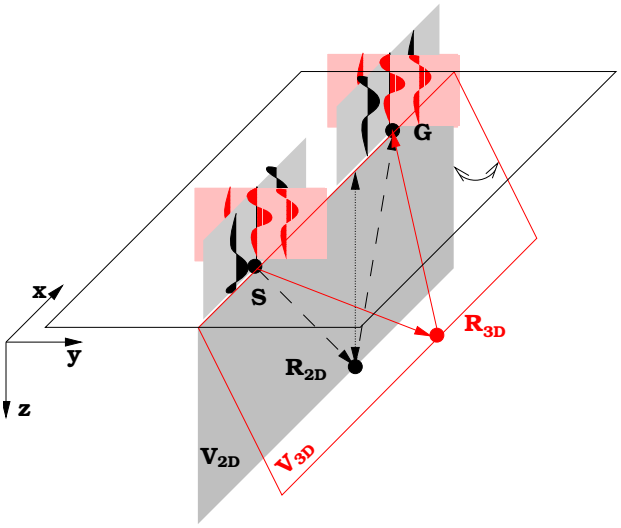


Figure 5: 3-D acquisition and imaging. The data contained in the red traces allow the unambiguous determination of the propagation direction and positioning of the diffractor. intro-3D-2D-aq [NR]

the reflector; that is,

$$\Delta y^2 + (\rho - \Delta z)^2 = \rho^2. \quad (1)$$

In this simple case, therefore, the imaging of 3-D data can be performed as two sequential steps: imaging along 2-D lines followed by cross-line migration. However, when the propagation velocity is not constant, this simple decomposition of 3-D imaging is not correct, and full 3-D migration is required. In the general case, the in-line 2-D migration cannot focus all the data properly because the velocity function V_{2D} along the vertical plane is different than the true velocity function V_{3D} along the oblique plane; when the velocity variations are large, the propagation paths do not even lie on planes, but along more complex surfaces.

The next several figures show concrete examples of the different types of error introduced by 2-D imaging when it is applied to geological structures that present 3-D features. The figures compare the sections of prestack depth migrated cubes. The sections on the left show the results of full 3-D migrations of the data; the sections on the right show the results of performing independent 2-D migrations on parallel seismic lines. Figure 6 shows in a cross-line section the mispositioning of reflectors caused by 2-D imaging. The arrows point to two dipping reflectors that are grossly mispositioned by 2-D imaging. From the 2-D image the width of the channel between the two reflectors would be underestimated by about 300 meters. Figure 7 shows the need for a cross-line migration to properly focus the data. The diffractions created by a rough salt-sediment interface, indicated by the arrows in the figure, are correctly collapsed only in the 3-D image.

The errors in the structural interpretation caused by 2-D seismic are further illustrated in Figure 8. This figure shows depth slices cut through the same 2-D-imaged and 3-D-imaged cubes, as in the previous figures. The broad reflection indicated by the A arrow has been mispositioned by several hundred meters, but even more evident is the misfocusing and mispositioning of the reflector indicated by arrow B. The results of 2-D imaging show a bow-tie shape instead of the correct sinusoidal shape.

The lack of proper cross-line migration is not the only cause for inaccurate structural imaging by 2-D seismic in a 3-D world. When propagation velocity significantly varies in the cross-line direction, the velocity function (V_{2D}) that is used to back-propagate the reflections along the vertical plane, is in general different from the correct velocity function (V_{3D}) that should be used for propagating the reflections along the correct oblique plane. The use of a wrong velocity function causes both a mispositioning and a defocusing of the imaged reflectors. By comparing two in-line sections from the image cubes previously presented, Figure 9 shows an example of the misfocusing introduced by 2-D seismology. The reflections from the salt flanks (Arrow A) and the reflections for the bottom of the salt (Arrow B) are significantly better focused in the 3-D image than in the 2-D image. Notice also the better focusing of the reservoir interval (Arrow C).

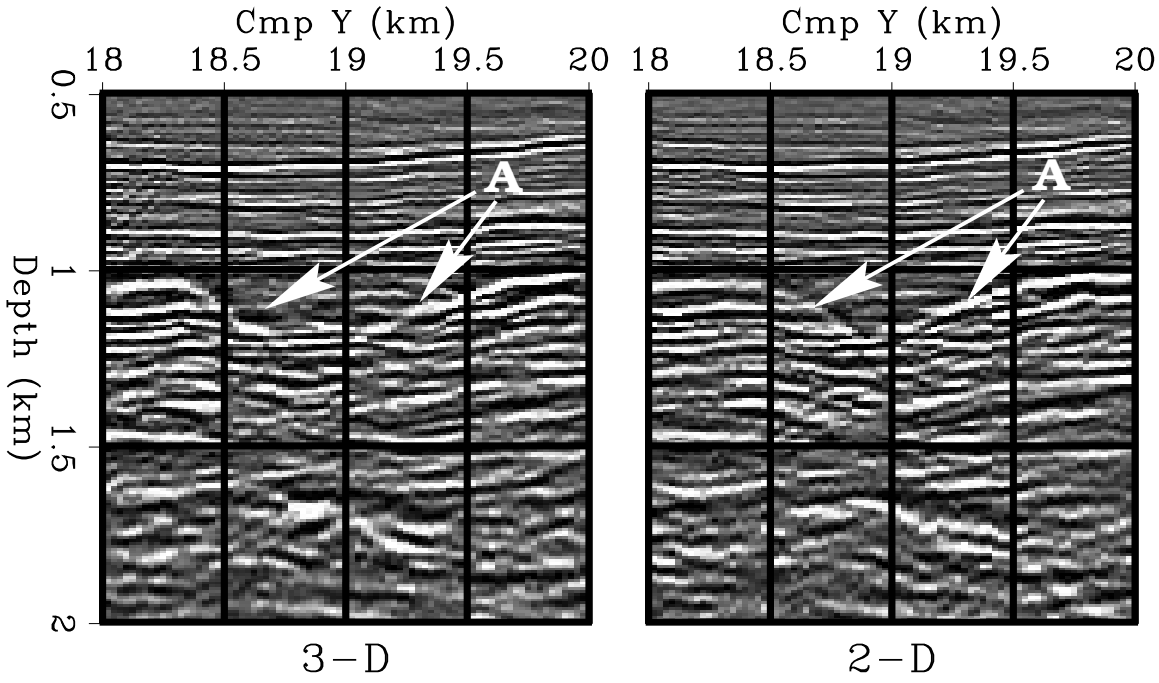


Figure 6: Cross-line sections showing that 2-D imaging causes lateral mispositioning of dipping reflectors. intro-channel [NR]

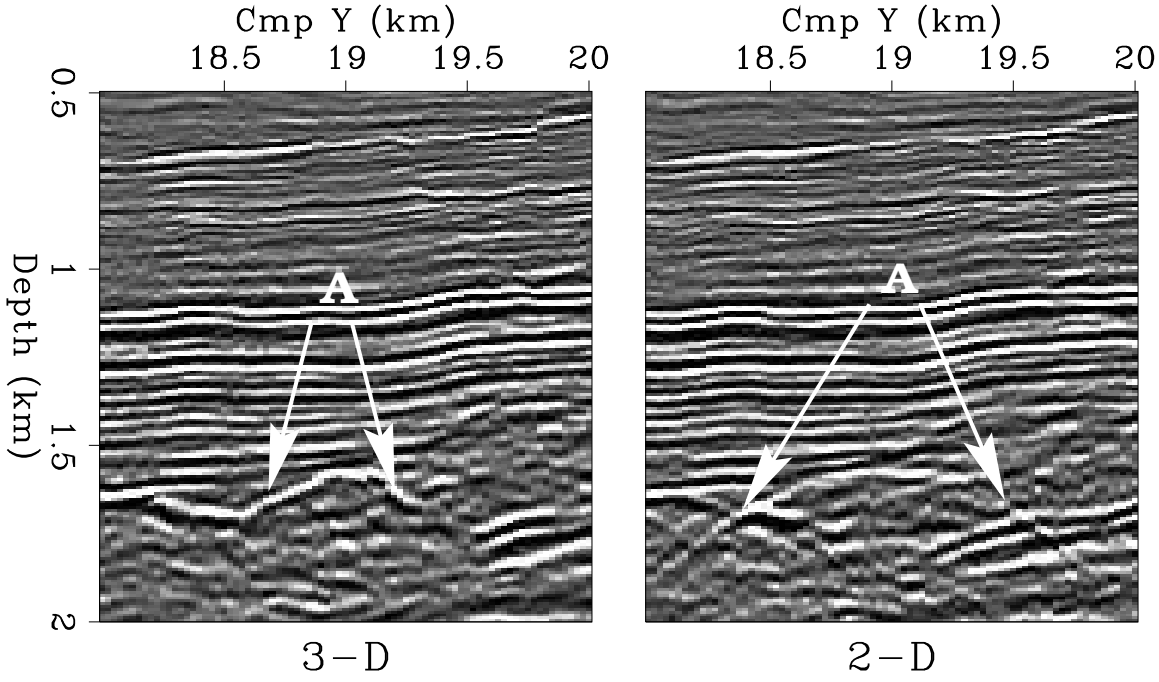


Figure 7: Cross-line sections showing that 2-D imaging does not focus diffractions properly. intro-diffr [NR]

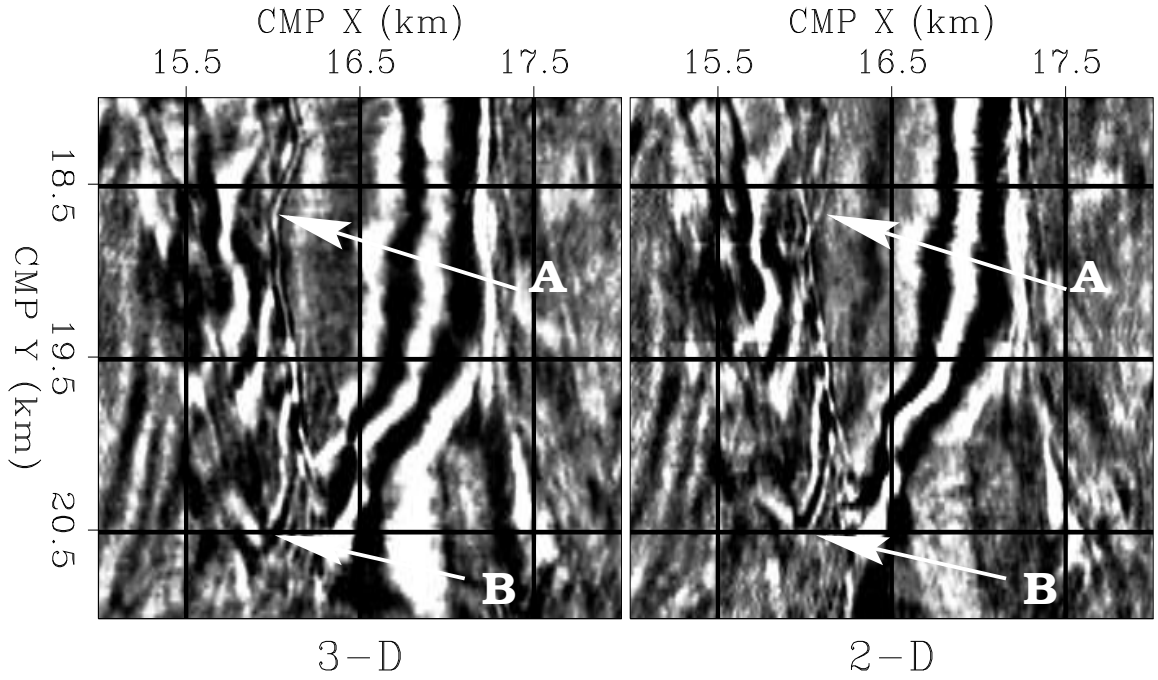


Figure 8: Depth slices showing that 2-D imaging causes misfocusing and lateral mispositioning. intro-depthsl [NR]

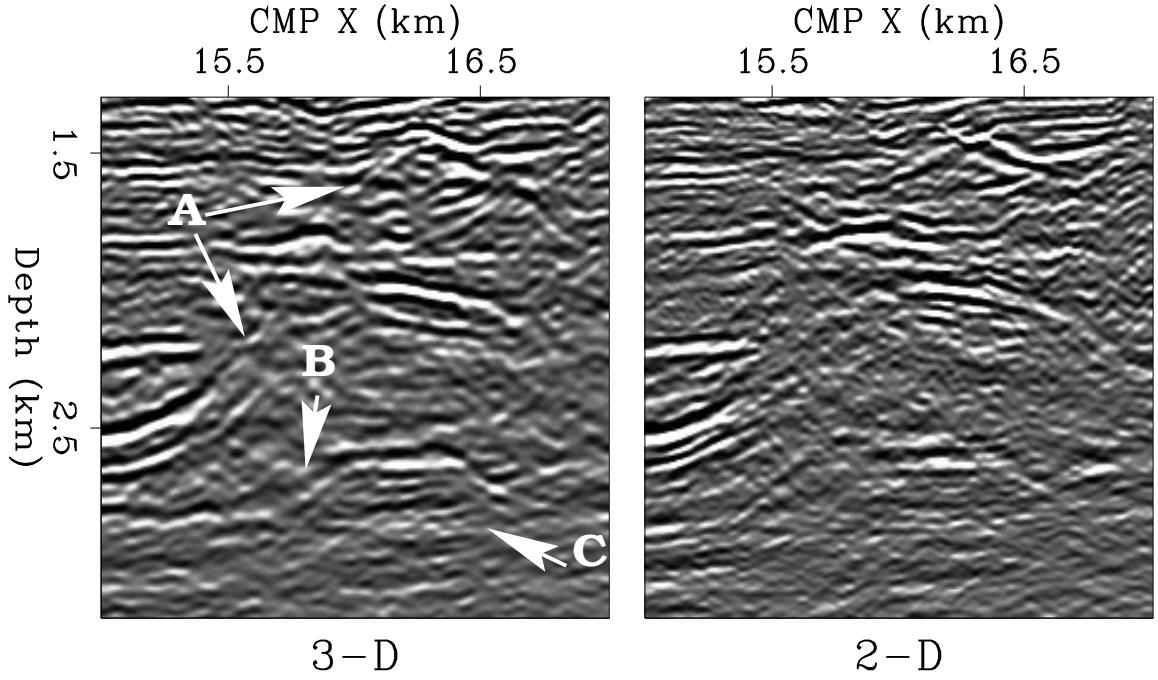


Figure 9: In-line sections showing that 2-D images are misfocused because of lateral velocity variations. intro-salt [NR]

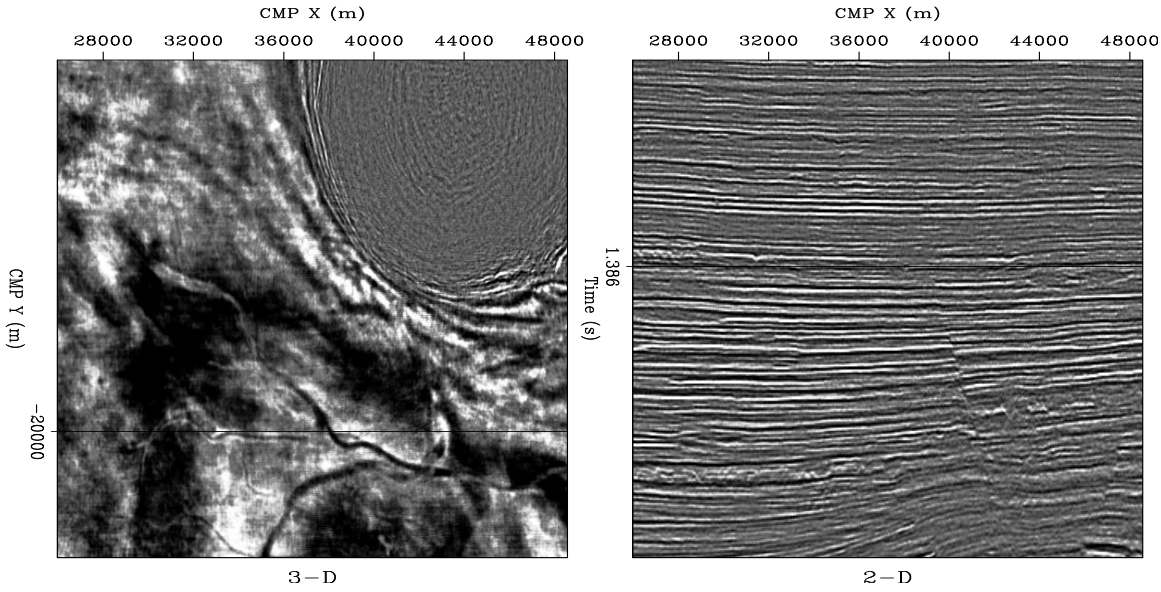


Figure 10: Time slice (t=1.386 s) and in-line section (CMP X=-20000 m) of a 3-D image from data recorded in the Gulf of Mexico. Notice the complex channel system that is visible in the time slice on the left. It would be impossible to reconstruct it from a set of vertical slices like the section on the right. intro-addinfo-overn [CR]

0.2 2-D vs. 3-D: extra information

In the previous section, we have seen examples of errors introduced by the 2-D imaging of 3-D structures. 3-D seismic not only can increase the accuracy of structural images of the subsurface, but also provides a wealth of stratigraphic information that is not present in 2-D seismic.

Figure 10 shows an example of geological information that would be impossible to recover from a set of 2-D lines. The time slice on the left displays a complex channel system embedded in sediments in the proximity of a salt dome in the Gulf of Mexico. These channels are thin features, barely above the threshold of seismic resolution. If we had available only the in-line section shown on the right of Figure 10, it would be difficult to even detect the presence of the channels. It would be impossible to reconstruct the complex 3-D structure of the channel system.

Even when the geological structure is 2-D or 1-D, 3-D seismic can provide additional information on the rock parameters. This additional information is contained in the data recorded along different source-receiver azimuthal directions. Typically, land data is acquired with a fairly wide range of azimuthal directions, whereas marine data has a narrower range. However, even for marine data, the trend is towards acquiring surveys with a wider range of azimuths for the source-receiver pairs, both because of the widespread use of multi-streamer acquisition configurations and because of the increasing use of Ocean Bottom Cables (OBC) surveys. Recording data with a wide source-receiver azimuth provides additional information

on the subsurface properties. For example, measuring propagation velocity and reflectivity along different azimuthal directions can be useful in estimating anisotropic properties caused by vertical fracturing in reservoir rocks. The imaging methods described in this book assume an isotropic Earth; therefore, they cannot be directly applied to imaging an anisotropic Earth. However, the basic concepts of 3-D imaging are still essential when extracting anisotropy information from 3-D data. Tsvankin (2001) provides a good introduction to the issues related to anisotropic imaging.

The availability of high-resolution 3-D images has also paved the way for time-lapse seismic (Jack, 1997), which is often referred to as 4-D seismic. Time-lapse seismic is used to monitor the evolution of a reservoir over time and contributes to the development of optimal strategies for managing reservoirs. In this book I do not focus on the application of 3-D imaging to 4-D seismic; nevertheless the success of time-lapse studies is dependent on the quality of 3-D imaging. Often the changes over time in the reservoir seismic properties are small and close to the seismic resolution; only high-fidelity imaging can properly uncover them.

0.3 2-D vs. 3-D: new challenges

Many of the basic concepts that have been developed for 2-D seismic imaging are still valid for 3-D imaging. However, 3-D imaging presents several new problems that are caused by the 3-D data's higher dimensionality. As we will see in more detail in Chapter 1, 3-D seismic prestack data is defined in a 5-D space, compared with the 3-D space of 2-D prestack data. The five dimensions are the recording time (t), the two components of the source position ($x_s, \hat{\mathbf{s}}_z$), and the two components of the receiver position (x_g, y_g). An obvious problem related to the high dimensionality of 3-D seismic is the amount of data that is recorded in the field and that needs to be processed. Modern surveys often produce several terabytes of data. Such large quantities of data are difficult to handle, visualize, and process. An important aspect of processing a 3-D data set is to devise the right strategy for reducing the time and resources necessary to obtain an image of the subsurface from the original large data set, without compromising the accuracy of the results.

The two most common methods for reducing processing cost are aimed at reducing the dimensionality of the computational domain. These methods are useful in 2-D, but they are essential in 3-D, because of the higher dimensionality of the data space. The first method, commonly called **stacking**, reduces the size of the data by averaging the recorded traces along appropriate ranges of the four spatial axes. The most common example of stacking is offset stacking, where recorded traces are averaged over offset after the application of Normal Moveout (**NMO**) (Chapter 3). Offset stacking yields a great reduction in computational cost because the cost of many imaging operators is directly proportional to the size of the input data. Common azimuth migration (Chapter 7) is another example of an imaging algorithm that achieves high computational efficiency by exploiting a reduction in data dimensionality (from 5-D to 4-D). In this case, a common-azimuth data set is generated from marine data that are averaged over azimuths after applying Azimuth Moveout (**AMO**) (Chapter 3). Common-azimuth data is then imaged by using a wavefield-continuation migration method.

The second class of method used to reduce the computational complexity is based on **full separation**, or **splitting** the computations along different data axes. Two-pass migrations are typical examples of applications of full-separation methods. In general, by decoupling the computations along an axis x of length n_x and an axis y of length n_y, we reduce the computational cost from being of order $O(n_x \times n_y)$ to being of order $O(n_x + n_y)$. The cost advantages of both stacking and separation are achieved at the expense of generality; the derived imaging methods are accurate only for particular classes of data sets. Because the selection of the appropriate processing strategy is a crucial decision, we will discuss the available choices as a function of their trade-off between accuracy and speed.

The other major challenge of 3-D seismic is that 3-D prestack seismic data is sparsely and irregularly sampled along the spatial coordinates $(x_s, \hat{\mathbf{s}}_z, x_g, y_g)$. Because of practical and economical considerations, actual 3-D surveys recorded in the field (Chapter 1) never record seismic traces at every possible location for the source and the receiver, and therefore the data never fill the 5-D space potentially available. On the contrary, much effort is spent to optimize survey design so that the desired accuracy in the image is achieved while the number of recorded traces is minimized. The result is that the seismic wavefield is almost never well sampled along both the source axes and the receiver axes. The condition for unambiguous imaging that we introduced in Section 0.1 when discussing Figure 5 is thus seldom fulfilled. To generate high-fidelity images we must safeguard our imaging algorithm from generating aliasing artifacts (Chapter 8). Crucial to this goal is exploiting the redundant information contained in the recorded data by making assumptions on both the reflectivity model and the complexity of the wave propagation.

Irregularity in spatial sampling may cause severe variation in the spatial illumination of the subsurface. Depending on the degree of variability in data density and the complexity of the imaging algorithm, irregular sampling may cause distortions in the image amplitudes as well as the creation of coherent artifacts in the image. Chapter 9 introduces the basic concepts for compensating for variable data density by normalizing imaging operators. It also introduces a method that exploits data redundancy to reduce both aliasing and irregular sampling artifacts.

REFERENCES

Jack, I. G., 1997, Time-Lapse Seismic in Reservoir Management: Society of Exploration Geophysicists.

Tsvankin, I., 2001, Seismic signatures and analysis of reflection data in anisotropic media: Elsevier Science.

Chapter 1

3-D data geometries

Because of practical and economical considerations, 3-D surveys are never acquired with full regular sampling of the spatial axes. The problem of designing 3-D surveys presents many more degrees of freedom than the design of 2-D ones and it has no standard or unique solution. The design of 3-D acquisition geometries is the result of many trade-offs between data quality, logistics, and cost. Further, nominal designs are often modified to accommodate the operational obstacles encountered in the field.

Acquisition design and processing are becoming more and more connected because the characteristics of the acquisition geometry strongly influence the data processing. Understanding the principles of acquisition design is necessary to the understanding of many data-processing issues. The main goal of conventional acquisition design is to obtain an adequately and regularly sampled stacked cube that can be accurately imaged by post-stack migration. Other important design parameters are the minimum and maximum offsets. The minimum offset must be small enough to guarantee adequate coverage of shallow targets. Maximum offset must be sufficiently large to allow accurate velocity estimates that are necessary for both stacking and post-stack imaging. However, in common acquisition geometries the sampling of the offset axes may be inadequate when the data require more sophisticated prestack processing than simple stacking. Even the application of standard dip moveout (DMO) (Chapter 3) may be problematic with some commonly used acquisition geometries. In these cases, the requirement that the midpoint axes are adequately sampled is not sufficient, because the offset and azimuth sampling play an important role as well.

Another important emerging issue in modern 3-D survey design is subsurface illumination, as we image targets under increasingly complex overburden, and as we demand more control on the amplitudes of the images. Complex wave propagation associated with large velocity contrasts (e.g. salt or basalt bodies) may cause the data to illuminate the target only partially, even if the wavefield is sufficiently sampled at the surface. Because target illumination is strongly related to the data offset and azimuth, a careful survey design can dramatically improve the image quality. Survey-modeling tools that analyze the target illumination by ray tracing through an *a priori* model of the subsurface geology help to guide the design process towards successful acquisition geometries. However, the usefulness of these tools is limited

by how accurately the subsurface geology and velocity model are known.

Although there are no preset solutions to the problem of acquisition design, there are a few acquisition schemes that are commonly used as templates to be adapted to individual surveys. In this chapter, we will discuss common marine and land acquisition geometries and their characteristics. More details and deeper analysis are presented in the following two excellent books: Designing Seismic Surveys in Two and Three Dimensions (Stone, 1994) presents a comprehensive introduction to seismic-data acquisition design. Seismic Wavefield Sampling (Vermeer, 1990) discusses many issues related to wavefield sampling and seismic processing, though the analysis is limited to 2-D data.

1.1 Data coordinates

In any survey, each seismic trace is characterized by the corresponding positions of its source and receiver. The coordinates of source and receiver are defined relative to an orthogonal coordinate system with x and y as the horizontal axes and z as the vertical axis. In this book we always assume flat recording surface with the origin of the coordinates lying at the surface, unless topography of the recording surface is explicitly mentioned. Therefore, the source position is a two-dimensional vector $\mathbf{s} = (x_s, \hat{\mathbf{s}}_z)$, and the receiver position is a two-dimensional vector $\mathbf{g} = \left(x_g, y_g\right)$. If the receivers are preferentially aligned along one direction, as they are for marine surveys and most land surveys, we will assume that the x axis is aligned with this preferential direction. In this case we will call the x axis the **in-line** axis and the y axis the **cross-line** axis. Given these definitions, the geometry of a 3-D prestack data set is defined in a 4-D continuum $\left(x_s, \hat{\mathbf{s}}_z, x_g, y_g\right)$, and the data is defined in a 5-D continuum $\left(t, x_s, \hat{\mathbf{s}}_z, x_g, y_g\right)$, where t is the recording time. Each trace represents a discrete sample in the 4-D geometry space. Usually the geometry space is irregularly sampled along all the axes, and often it is inadequately sampled along some or all axes, as well.

The data coordinates expressed in terms of source-receiver coordinates are often called the **field coordinates**, as opposed to the **midpoint-offset coordinates**. The midpoint-offset coordinates are defined by the following transformation of the field coordinates:

$$\begin{aligned} \mathbf{m} &= \frac{\mathbf{g}+\mathbf{s}}{2} \\ \mathbf{h} &= \frac{\mathbf{g}-\mathbf{s}}{2}, \end{aligned} \tag{1.1}$$

where the vector $\mathbf{m} = (x_m, y_m)$ is the midpoint, and the vector $\vec{\mathbf{h}} = (x_h, y_h)$ is the half-offset. Often the half-offset vector is more conveniently defined in terms of the **absolute half-offset** h and the **source-receiver azimuth** θ_h as $\vec{\mathbf{h}} = h(\cos\theta_h, \sin\theta_h)$. Because the source-receiver azimuth is uniquely defined for each trace, it is often referred to as just the **trace azimuth** or **data azimuth**. This angle must be distinguished from the **midpoint azimuth** ϕ_m that is related to the geological dips in the image, as we will see in following chapters. Figure 1.1 shows the geometric relationships between the field coordinate vectors (**g** and **s**) and the offset-midpoint coordinate vectors (**m** and **h**).

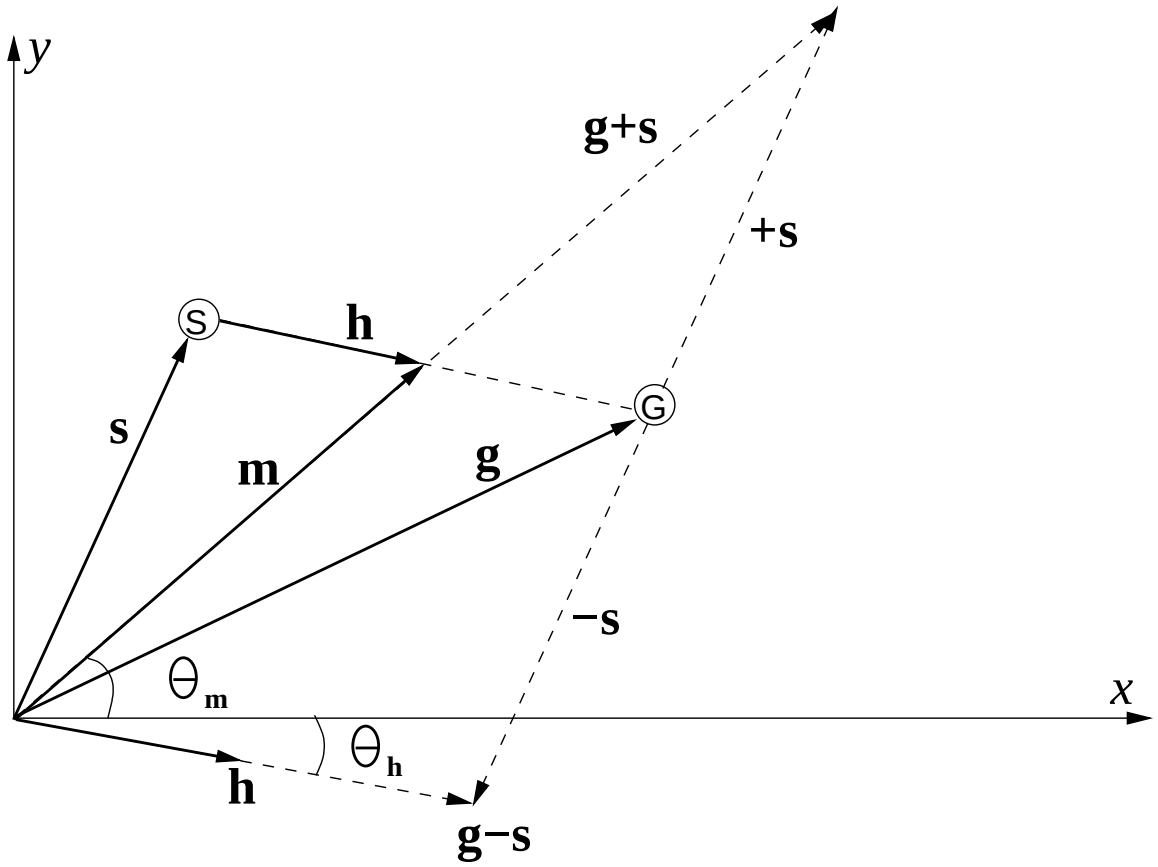


Figure 1.1: Geometric relationships between the field coordinate vectors (**g** and **s**) and the offset-midpoint coordinate vectors (**m** and **h**). geom-geom-3d [NR]

Acquisition geometries can vary greatly according to the physical environment (e.g., land or marine), the subsurface structural complexity, and the overall goals of the survey. In the following section, we present and analyze several templates of acquisition geometries that are commonly used in survey design. All the geometry templates share the common characteristic that the midpoint axes are fairly well, and often even regularly, sampled. Good sampling of the midpoint axes is necessary to accomplish the paramount goal of obtaining accurate poststack-migration results. In contrast, the sampling of the offset axes may vary according to the choice of an acquisition template. In our analysis, we will focus on the sampling of the source and receiver axes and of the offset axes for each geometry template.

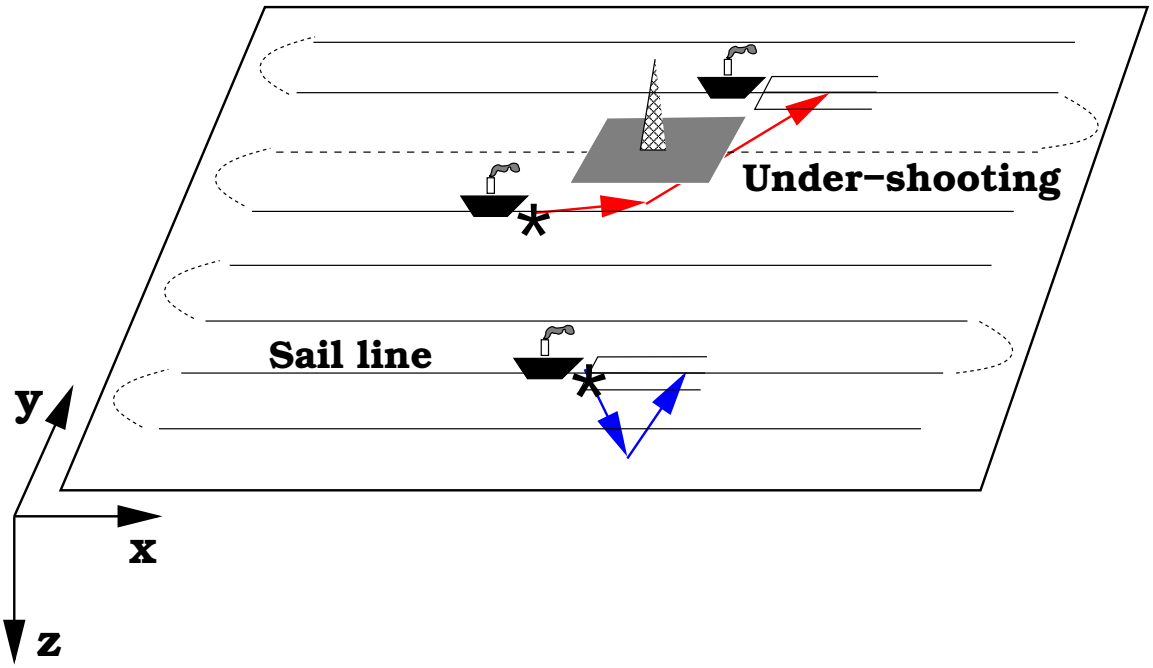


Figure 1.2: Cartoon representing typical marine streamer acquisitions. geom-sail-lines [NR]

1.2 Marine-data geometries

1.2.1 Streamer geometries

Modern marine data is acquired by towing several streamers behind a boat. Each streamer incorporates hundreds of hydrophones located at regular intervals. Streamers can be as long as six to eight kilometers. One or more sources of seismic energy are positioned close to the stern of the boat. Sources are usually arrays of air-guns, but experiments with marine vibroseis are yielding promising results. As the cartoon in Figure 1.2 shows, the boat passes over the target on multiple passes along (ideally) straight lines (called **sail lines**) that are equally spaced and parallel to each other. The sail-line direction is considered the in-line direction of the surveys. Currents and other factors often cause the sail-lines not to be perfectly straight; even more commonly, currents cause the streamers not to follow straight trajectories behind the boat. The lateral displacement of the streamers is often called **cable feathering**.

Another difficulty encountered in areas already developed for hydrocarbon production is that platforms, or other permanent obstacles, can be on the desired path of the boat. A technique called **under-shooting** is often employed to avoid gaps in the coverage of the survey just below the obstacles, which are often just above targets of interest. As shown in Figure 1.2, under-shooting requires two boats, one carrying the sources and the other towing the hydrophone streamers. The data recorded by under-shooting often have a wider azimuth range and larger minimum offset than regularly recorded data; both the lack of near-offset data and the wide azimuth may cause problems during processing.

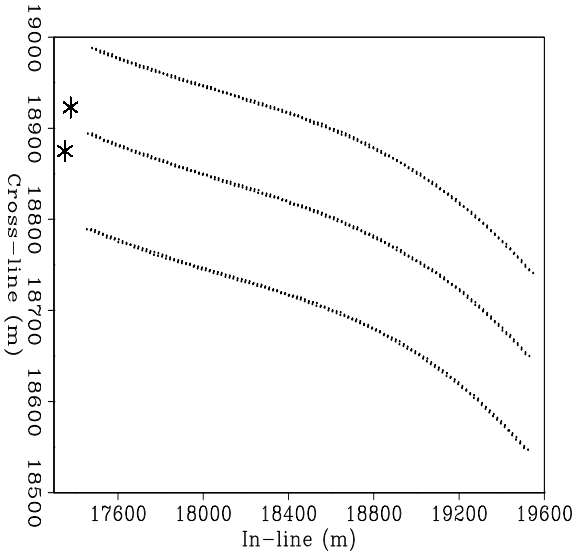


Figure 1.3: Source and receiver locations for two consecutive shot gathers of a three-streamers double-source acquisition in the North Sea. Receivers are represented by dots and sources by asterisks. geom-RS-nsea [CR]

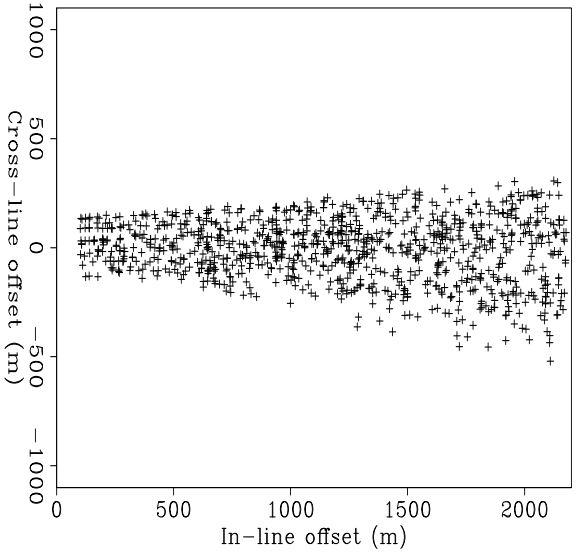


Figure 1.4: Offsets of 1,000 traces randomly selected from the same streamer data set as the shot gathers represented in Figure 1.3. geom-Off-nsea [CR]

Figure 1.3 shows the source and receiver locations for two consecutive **shot gathers** (a shot gather is the collection of all the traces recorded with a particular source location) recorded in the North Sea. The shooting configuration had three streamers and two sources. The sources were activated in alternating fashion. The dots in the figure represent the receiver locations along the streamers, and the two asterisks represent the source locations, at either side of the central streamer. The streamers are separated 100 m in the cross-line direction and are not along straight lines because of strong cable feathering.

Figure 1.4 shows the offsets of 1,000 traces, randomly selected from the data set. We notice that the in-line component of the offset x_h covers a much wider range (200 m – 2,200 m) than does the cross-line component y_h (-400 m – 400 m). The cross-line range is wider at far offset, because cable feathering is more pronounced at the far end of the streamers. The cross-line offset range does not drop to zero at near offset because of the use of multiple-streamers; as is measurable from Figure 1.3, the cross-line distance between each source and the outer streamer on the opposite side is 120 m. Because of this narrow azimuth range, the in-line component of the offset is densely sampled and prestack imaging is robust.

Notwithstanding the advantages of narrow-azimuth acquisition, the trend in streamer surveys is to increase the number of streamers, and consequently increase the azimuth range of the data. The increase in the number of streamers is motivated by cost considerations: the more streamers are towed, the wider the swath of the subsurface a boat can cover with a single pass, thus reducing the total acquisition time.

1.2.2 Parallel-swath geometries (OBC and OBS)

As seismic data are increasingly being used to optimize development and production strategies, more and more surveys are acquired in highly developed areas where towing kilometer-long streamers would be difficult or even hazardous. Therefore, an increasing number of surveys are acquired with an array of receiver cables placed at the bottom of the ocean. The sources are at the surface and are carried on boats as in standard streamer surveys. This kind of acquisition technology is called **OBC**, for Ocean Bottom Cable.

A particular kind of OBC survey deploys three-component geophones to record shear waves in addition to pressure waves. This kind of survey is often called **OBS**, for Ocean Bottom Seismometer. Since the sources are at the ocean surface, and since shear waves do not propagate in liquids, the shear waves must be created by mode conversion either at the water bottom or at the reflectors. Shear waves carry information on rock and fluid properties that is complementary to the information carried by pressure waves. Furthermore, the propagation of shear waves is less affected by slow and attenuating areas in the subsurface (e.g. gas clouds). Underneath these “obstacles”, shear-waves data may provide better structural images than pressure-waves data.

Both OBC and OBS geometries resemble land geometries more than classical marine ones. A popular acquisition template is the **parallel-swath** shooting that has an advantage over the cross-swath shooting (analyzed in Section 1.3.1) in that it allows the deployment of long receiver cables and it requires less turns of the shooting boat.

Figure 1.5 shows the source and receiver positions for two consecutive shot gathers of an OBS survey recorded in the North Sea. As in previous figures, receivers are represented by dots and sources by asterisks. The receivers are laid along parallel lines; the nominal separation between receiver lines is 400 m. The sources move in the direction parallel to the receiver lines, hence the name parallel swath. Notice that, in contrast to streamer surveys (Figure 1.3), the receiver locations are exactly the same for the two shot gathers, since the receiver cables are stationary at the ocean bottom.

Figure 1.6. shows the geometry of a common **receiver gather** (a receiver gather is the collection of all the traces recorded at the same receiver location for all the shots) for the same OBS data set. The receiver is represented by a large dot in the middle of the figure; the sources are represented by asterisks. The source spacing is fairly small in both directions (about 50 meters), and thus the wavefield is relatively well sampled in a receiver gather. Prestack imaging in the receiver-gather domain is therefore an option.

Figure 1.7 shows the offsets of 1,000 traces, randomly selected from the data set. As with the streamer geometry discussed previously, the offsets of a cross-swath geometry have a preferential direction aligned along the receiver lines. Therefore, in this case also, we can consider the receiver-line direction as the in-line direction. However, compared with those of streamer geometry, the cross-line offsets (maximum of 1.4 km) are not negligible, creating potential problems for some prestack processing algorithms (e.g. DMO – Chapter 3).

Notice that for both OBC and OBS data the elevation of the sources (ocean surface) is different from the elevation of the receivers (ocean bottom). This difference may create some practical problems and may require the adaptation of algorithms that have been developed assuming that sources and receivers are at the same elevation. However, it rarely creates new conceptual challenges.

1.3 Land-data geometries

Land-data geometries vary more widely than marine ones because receiver locations are not constrained to be attached to a towed streamer. Therefore, there are several possible templates that can be adopted when a land survey is designed. We will analyze two templates that illustrate some of the issues encountered in land geometry design, and will discuss their respective advantages and disadvantages. One important template that we will not discuss despite it has some theoretical advantages, is the **cross spread** (Vermeer, 1998); it is seldom applied because of its cost.

1.3.1 Cross-swath geometries

The first template we will examine is commonly called **cross-swath** shooting. An example of this geometry is given in Figure 1.8, which shows the source and receiver positions for two consecutive shot gathers recorded in the Canadian Rockies. As in previous figures, receivers are represented by dots and sources by asterisks. The receivers are laid along parallel lines;

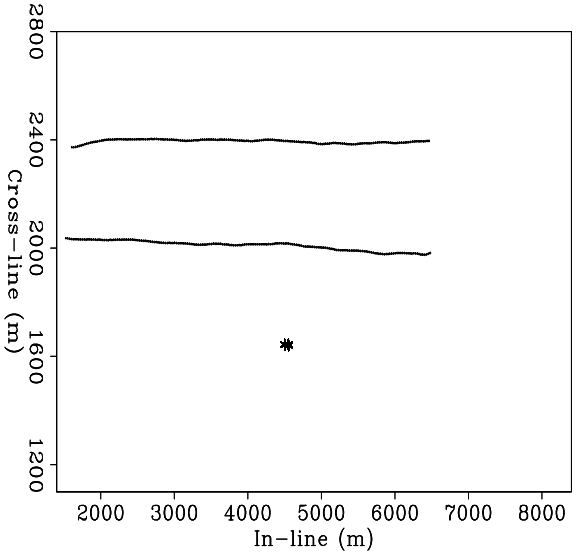


Figure 1.5: Source and receiver locations for two consecutive shot gathers of an OBS data set from the North Sea. Receivers are represented by dots and sources by asterisks. geom-Shot_obs_1 [CR]

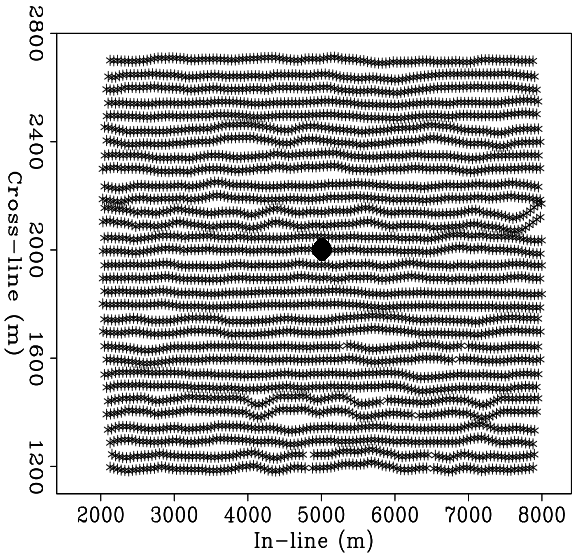


Figure 1.6: Source and receiver locations for one receiver gather of an OBS data set. The receiver is represented by a large dot and the sources are represented by asterisks. geom-Rec_obs_1 [CR]

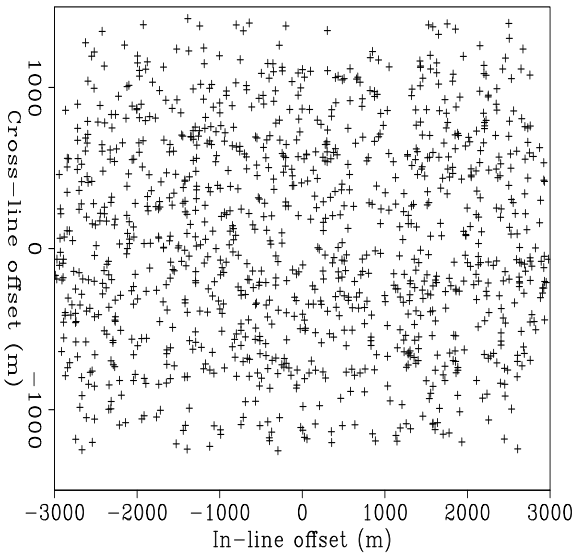


Figure 1.7: Offsets of 1,000 traces randomly selected from an OBS data set. geom-Off_obs [CR]

the separation between the receiver lines is 250 m. The receiver lines are crooked because of obstacles in the terrain. The sources are moved in a direction approximately orthogonal to the receiver lines; hence the name cross swath. When the source moves outside the swath covered by the receiver array, the farthest receiver line is rolled over to the side of the new shots. This acquisition configuration assures that the receiver arrays of individual shot gathers are well sampled along the receiver-line direction, but they are aliased in the orthogonal direction. Conversely, common receiver gathers are well sampled along the cross-line direction, but poorly sampled along the in-line directions. Only the midpoint axes are well sampled along both directions.

Figure 1.9 shows the offsets of 1,000 traces, randomly selected from the data set. As with the marine geometries discussed previously, the offsets of a cross-swath geometry have a preferential direction aligned along the receiver lines. Therefore, in this case also, we can consider the receiver-line direction to be the in-line direction, with the sources moving along the cross-line direction. However, compared with those of streamer geometries, the trace azimuths are not so clustered around the in-line azimuth and the offset plane is densely sampled only at near offsets.

1.3.2 Button-patch geometries

Button-patch acquisition has been developed with the aim of sampling the data azimuths over the whole range between 0° and 360°. Wide-azimuth data are necessary to gather information on how seismic properties, such as velocity and reflectivity, vary with propagation direction. This detailed information can be useful when the subsurface is anisotropic.

The acquisition is called “button patch” because the receivers are deployed in a checkered pattern (buttons), as shown in Figure 1.10. The ensemble of several buttons makes up a patch that is rolled along to cover contiguous areas. The receiver locations (dots) and source location (asterisk) from a shot gather recorded in Wyoming with the button-patch technique are shown in Figure 1.11. The irregularity in the location of the buttons visible on the bottom-right corner of the receiver array is caused by obstacles on the surface. As expected, the offsets of button-patch acquisition are well distributed on the entire offset plane, as shown in Figure 1.12. The figure shows the offsets of 1,000 traces, randomly selected from the data set.

1.4 Narrow azimuth vs. wide azimuth

The main distinction between the four acquisition geometries that we have discussed is in their sampling of the offset plane. To gain an intuition of the different sampling of the offset plane achieved by different geometries, we can plot again Figure 1.4, 1.7, and 1.9 together and on the same scale as Figure 1.12. The top-left panel in Figure 1.13 shows Figure 1.4 rescaled, the top-right panel shows Figure 1.7 rescaled, the bottom-left panel shows Figure 1.9 rescaled, and finally the bottom-right panel shows Figure 1.12. For the streamer geometry, only the positive half of the in-line offset component is plotted, because **reciprocal traces**

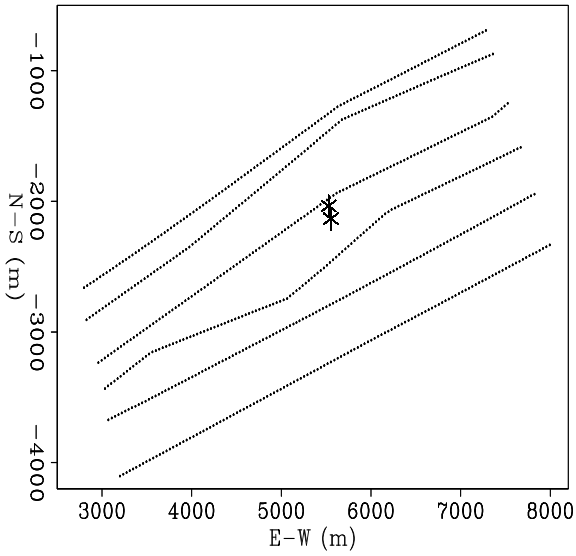


Figure 1.8: Source and receiver locations for two consecutive shot gathers of a cross-swath land acquisition geometry from the Canadian Rockies. Receivers are represented by dots and sources by asterisks. geom-RS-husky [CR]

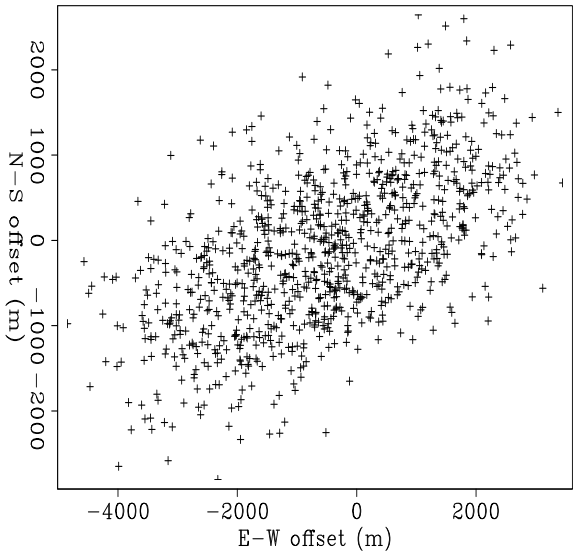


Figure 1.9: Offsets of 1,000 traces randomly selected from a cross-swath data set. geom-Off-husky [CR]

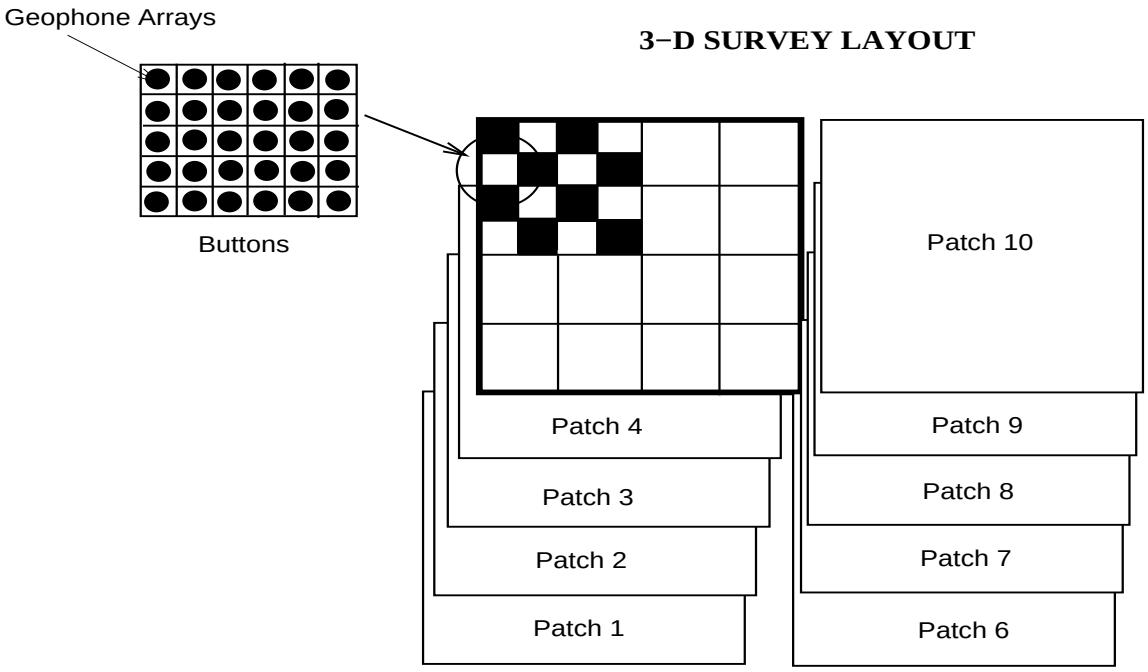


Figure 1.10: Button-patch acquisition geometry. Provided by J.M. Darnal; Arco Exploration and Production Technology. geom-layout [NR]

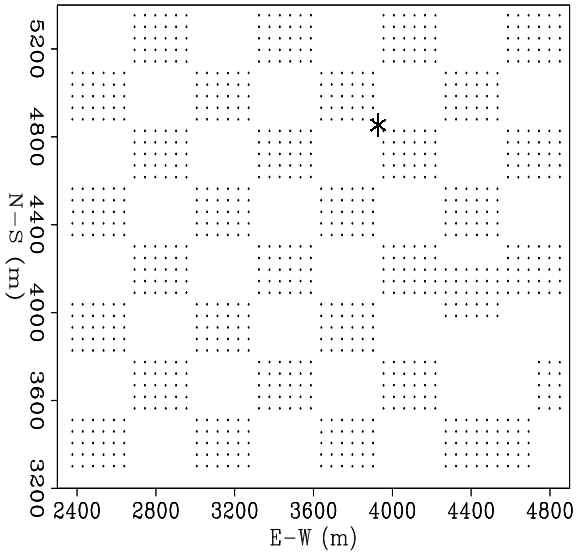


Figure 1.11: Source and receiver locations for one shot gather of a button-patch land acquisition geometry from Wyoming. Receivers are represented by dots and the source is represented by an asterisk. geom-RS-arco [CR]

(a trace pair with the respective source and receiver locations exchanged) provide equivalent sampling for prestack imaging operators. This comparison is only qualitative, because other factors such as propagation velocity, target depth, and signal frequency should be considered when the sampling criteria of seismic reflections are determined. However, the difference in offset coverage and trace density among the four geometries is quite dramatic. The azimuthal coverage increases as we move from streamer geometry, to parallel swath, to cross swath, and finally to the button-patch geometry.

Wide-azimuth geometries have the obvious advantage that data with different azimuths can provide useful information on the anisotropy of the subsurface, as well on the heterogeneity of the velocity field. On the other hand, if we consider the number of recorded traces as fixed, the drawback of wide-azimuth geometries is that they sample the offset plane less densely, and thus they are more subject to aliasing problems of the offset axes. This is usually not a problem if no imaging operator is applied in the prestack domain, and only trace-to-trace transformations, such as NMO or stacking (Chapter 3), are applied to prestack data. However, if prestack imaging is necessary because of structural complexity (DMO – Chapter 3). and/or strong velocity variations (full prestack migration – Chapter 2), the sparse sampling of the offset axes may cause artifacts in the imaging results (Canning and Gardner, 1996). This trade-off between the width of the azimuth coverage and data density can be avoided by either recording more traces, or by improving the methods used to prestack image wide-azimuth data.

Considerations of subsurface illumination may further complicate the trade-off between narrow and wide azimuth data. Since, under complex overburden, target illumination varies with the data azimuth, a wide-azimuth survey has "illumination holes" more uniformly distributed in the subsurface. Therefore, it has a better chance of providing at least partial information on all the geological structures than a narrow-azimuth survey has. For marine exploration, wide-azimuth OBC surveys can be used to address this problem. A practical and economical solution is to acquire two conventional narrow-azimuth data sets with the sailing directions orthogonal to each other.

1.5 Sorting and binning

To facilitate the subsequent data processing, often the order of the traces is changed according to the traces' geometry. **Sorting** and **binning** are two common operations that reorder the traces.

Sorting is a reordering of the traces according to specific trace coordinates, called **sorting indexes**. Typical sorting operations are the reordering of the data in shot gathers or receiver gathers. These sorting operations are conceptually simple because all the traces sharing the same source (receiver) location have the exact same source (receiver) coordinates. On the other hand, common-midpoint sorting is complicated by the fact that, because of irregularities in the data acquisition, traces do not share exactly the same midpoint coordinates. Therefore, common-midpoint sorting requires a prior common-midpoint binning.

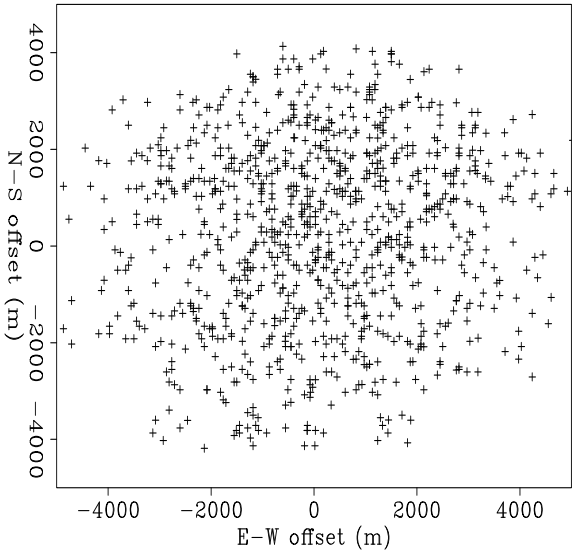


Figure 1.12: Offsets of 1,000 traces randomly selected from a button-patch data set. geom-Off-arco [CR,M]

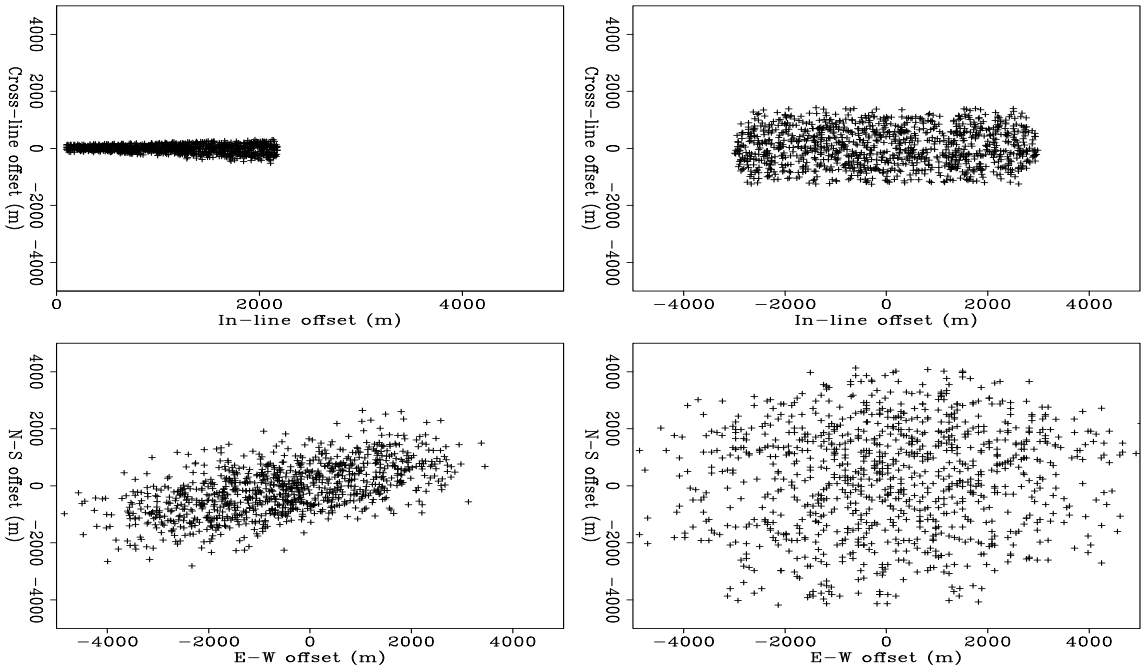


Figure 1.13: Rescaled plots of the offset planes shown in Figures 1.4 (top-left), 1.7 (top-right), 1.9 (bottom-left), and 1.12 (bottom-right). Notice the substantial differences in offset coverage and trace density among the four geometries. geom-All-Off-sc [CR,M]

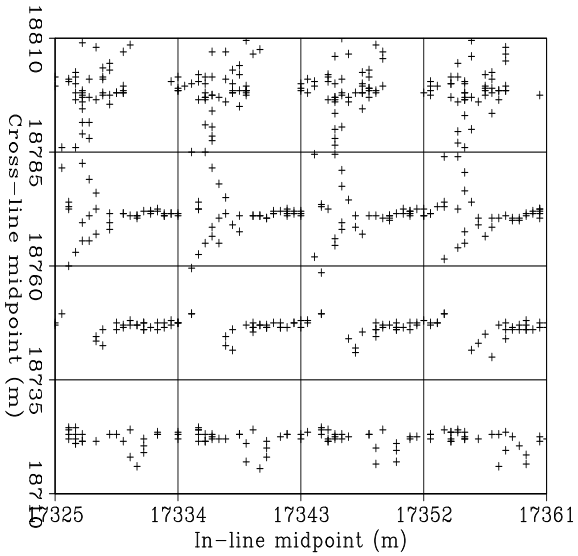


Figure 1.14: Midpoint plane for the streamer-data set. The grid superimposed onto the plot represents the binning grid. geom-Cmp-nsea [CR]

Common-midpoint binning is not only necessary to sort the data, but it can be also useful to perform basic operations such as stacking. The data are binned by superimposing a regular grid onto the midpoint plane; then all the traces that fall within each grid cell, or bin, are assigned effective midpoint-coordinates equal to the central point of the cell. The nominal **bin size**, that is, the length of each side of the cells, is determined by the acquisition parameters and can vary between the in-line direction and the cross-line direction. The nominal **fold** is equal to the number of traces that would fall in each grid if there were no irregularities in the geometry.

Figure 1.14 shows a small part of the midpoint plane of the streamer geometry discussed previously (Section 1.2.1). The grid superimposed onto the traces represents the binning grid. The nominal bin sizes are 9.375 m in the in-line direction and 25 m in the cross-line direction. Because of geometry irregularities, mostly caused by cable feathering, the effective fold varies from 24 to 30 for this subset of the data; the nominal fold of the acquisition was 28.

REFERENCES

Canning, A., and Gardner, G. H. F., 1996, Another look at the question of azimuth: The Leading Edge, **15**, no. 07, 821–823.

Stone, D. G., 1994, Designing Seismic Surveys in Two and Three Dimensions: Soc. Expl. Geophys., Tulsa.

Vermeer, G. J. O., 1990, Seismic Wavefield Sampling: Soc. Expl. Geophys., Tulsa.

Vermeer, G. J. O., 1998, 3-D symmetric sampling: Geophysics, **63**, no. 05, 1629–1647.

Chapter 2

Full prestack migration by Kirchhoff's methods

The ultimate goal of recording seismic data is to recover an image of the geological structure in the subsurface. Imaging is the most computationally demanding and data-intensive component of seismic processing. Therefore, considerable effort has been spent in devising effective imaging strategies that yield accurate images but are computationally affordable. The most general and most expensive type of migrations are those that operate directly on the whole prestack data set; they are thus called full prestack migrations. All the other imaging methods are designed to approximate full prestack migration; that is, they aim to achieve the same image accuracy but at a fraction of the computational cost. In this chapter, we analyze the main characteristics of 3-D prestack migration and its computational costs. An understanding of these features is important to determine when full prestack migration is required, and to evaluate its several available approximations.

Subsurface imaging is the result of two processing steps that are tightly connected: velocity estimation and migration. Migration focuses the data and produces a subsurface image, but adequate information on propagation velocity is needed for migration to focus the data properly and to produce the correct image at depth. This chapter will discuss the strong dependence of prestack migration on the accuracy of the velocity function. In particular, we will discuss the principles of time migration and depth migration, and their differences with respect to the reliability of the estimates of the velocity function that each of them require. We leave the discussion of how to estimate the velocity function to a later chapter (Chapter 11).

2.1 Constant-velocity migration

To image the reflectors in the subsurface, migration removes from the recorded data the effects of the wave propagation from the surface to the reflectors and back to the surface. Migration is theoretically founded in wave-propagation theory. Different types of migration use different ways of solving the wave equation. Because of the irregular spatial sampling of 3-D prestack

data, the method of choice for 3-D prestack migration is often Kirchhoff migration.

Kirchhoff migration is based on Green's function theory and on an integral solution of the wave equation. Although the theoretical derivation of Kirchhoff migration is rather involved, the final result is extraordinarily simple (Schneider, 1978). The general form is given by the following integral expression:

$$I(\xi) = \int_{\Omega_\xi} W(\xi, \mathbf{m}, \mathbf{h})\, D[t = t_D(\xi, \mathbf{m}, \mathbf{h}), \mathbf{m}, \mathbf{h}]\; d\mathbf{m}\, d\mathbf{h}. \tag{2.1}$$

The image $I(\xi)$, which is defined in a three-dimensional space $\xi = (z_\xi, x_\xi, y_\xi)$, is equal to the integral of the data values $D(t, \mathbf{m}, \mathbf{h})$ evaluated at the time $t_D(\xi, \mathbf{m}, \mathbf{h})$ and weighted by an appropriate factor $W(\xi, \mathbf{m}, \mathbf{h})$. The amplitude factor should also include a one-dimensional convolution by a derivative filter. The order of the derivative depends on the dimensionality of the integration. A first derivative filter should be applied for 3-D migration.

The domain of integration in equation (2.1) does not span the whole input space, but it is limited to a region Ω_ξ in the midpoint **m** plane. It that is centered around the image location ξ. This region Ω_ξ is often called the **migration aperture**; it plays an important role in determining the dip limitation and the cost of the migration procedure.

The time shift $t_D(\xi, \mathbf{m}, \mathbf{h})$ is given by the total time delay accumulated as the reflections propagate from the source position **s** to the image point $I(\xi)$, and come back to the surface to be recorded at the receiver position **g**. For constant velocity, these time delays can be evaluated analytically as,

$$t_D = t_s + t_g = \frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{s}\right|^2}}{V} + \frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{g}\right|^2}}{V}, \tag{2.2}$$

or, in midpoint-offset coordinates, as

$$t_D = \frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{m} + \mathbf{h}\right|^2}}{V} + \frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{m} - \mathbf{h}\right|^2}}{V}, \tag{2.3}$$

where $\overline{\mathbf{xy}}_\xi = \left(x_\xi, y_\xi\right)$ represents the horizontal projection of the image-coordinates vector.

In practice, 3-D prestack data are recorded on a discrete set of surface points and we approximate the integral in equation (2.1) with a finite sum. We rewrite equation (2.1) as

$$I(\xi) \approx \sum_{i \in \Omega_\xi} W_i(\xi, \mathbf{m}_i, \mathbf{h}_i)\, D[t = t_D(\xi, \mathbf{m}_i, \mathbf{h}_i), \mathbf{m}_i, \mathbf{h}_i]. \tag{2.4}$$

The subscript i in the equation above reminds us that we record a finite number of data traces and that trace locations are defined on a discrete, though irregular, spatial grid.

An accurate definition of the weights in equation (2.4) is seldom important for producing a structural image, but it is crucial when reflection amplitudes are used to determine the relative strengths of the reflectors and to estimate petrophysical parameters in the subsurface. There is a whole body of research dedicated to the definition of the appropriate integration weights, so we are not going to analyze this issue in detail here. The general principle for deriving the

weights is to pose migration as an inverse problem. Cohen and Bleistein (1979) did much of the pioneering work and derived the solution for simple background media (constant and horizontally layered). Beylkin (1985) defined the problem as the inversion of a Radon transform, independent of the particular shape of the summation surfaces, and thus enabled the generalization of Cohen's and Bleistein's results to complex media. Schleicher et al. (1993) reach results equivalent to the previous authors, but their derivation is strictly based on ray theory and on the results of dynamic ray tracing (**U**Cervený and P**U**sen**U**cík, 1983). Bleistein and Gray (2001) provide a clear and intuitive introduction to the subject.

The integration weights are obviously dependent not only on the **kinematics** of wave-propagation (the shape of the summation surfaces), but also on the **dynamics**, that is, on the defocusing and focusing of the seismic wave as it propagates through the subsurface. In simple velocity media, this effect, which is often called **geometrical spreading**, can be well approximated by a factor proportional to the inverse of traveltime. However, in more complex media it needs to be computed numerically together with the traveltime functions.

The summation expressed in equation (2.4) can be numerically implemented in two alternative ways that are mathematically equivalent, but that may have rather different numerical properties. The first class of algorithms includes **gathering** methods, which loop over each image point and gather the contributions of all the input traces that are within the migration aperture. In contrast, the algorithms belonging to the second class, often referred as **spraying** methods, loop over each input trace and spray the data over all the image traces within the migration aperture.

The time delays computed by equation (2.2) define a family of **summation surfaces** for the gathering methods and a family of **spreading surfaces** for the spraying methods. In the next two subsections we will analyze and visualize these multi-dimensional surfaces.

Summation surfaces

The summation surfaces belong to a family of hyperboloids defined by equation (2.2). In addition to being the basis of Kirchhoff migration, the summation surfaces have the important physical interpretation of being diffraction surfaces for point scatterers located in the subsurface. The diffraction from an isolated point scatterer in the Earth's subsurface would create events with the same shape as the summation surfaces.

Summation surfaces are defined in the five-dimensional data space. However, visualizing 5-D surfaces is still beyond our capabilities, and thus we are limited to showing 3-D sections that represent interesting and meaningful special cases. The simplest of these special cases is the summation surface for zero-offset data. In this case, the double square-root of equation (2.2) simplifies into a single square-root, and the summation surface becomes the hyperboloid of rotation given by the following equation:

$$t_D = 2\,\frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{m}\right|^2}}{V}. \tag{2.5}$$

Figure 2.1 shows the zero-offset hyperboloid in a three-dimensional perspective. The vertical axis (identified by Z in the axis triplet visible at the lower left corner of the figure) is the time axis, whereas the horizontal axes are the midpoint axes. The contour lines identify circles with equal time delays. The cross at the apex of the hyperboloid identifies the image point corresponding to the summation surface. The time of the cross is equivalent to the two-way traveltime $\tau_\xi = 2z_\xi / V$.

A constant-offset summation surface is another meaningful 3-D section of the whole 5-D summation surface. Figure 2.2 shows a constant-offset hyperboloid in a three-dimensional perspective. For the sake of simplicity, but without loss of generality, the offset vector is assumed to be aligned along the in-line direction which is identified as X in the figure. The constant-offset hyperboloid is flat at the top, and is not circularly symmetric as was the zero-offset one; it is squeezed on the top along the vertical direction and squeezed horizontally along the cross-line direction, which is identified as Y in the figure. The contour lines identify points with equal time delays; the zero-offset circles have become constant-offset ellipses. As before, the cross identifies the image point corresponding to the summation surface, but in this case the image point is detached and above the summation surface. Varying the data azimuth while keeping the absolute offset constant would be equivalent to rotating the hyperboloid shown in Figure 2.2. In contrast, varying the absolute offset while keeping the azimuth constant would result in different amounts of both vertical and horizontal squeezing.

To gain a better intuition for the whole 5-D surface, Figure 2.3 shows simultaneously both the zero-offset hyperboloid and the constant-offset hyperboloid corresponding to the same image point. For large time delays, the two surfaces approach the same asymptote and, at the limit, become tangent. The in-line edges of the constant-offset surfaces approach the zero-offset surface faster than do the cross-line edges.

Spreading surfaces

Spreading surfaces are dual to the summation surfaces and they belong to a family of ellipsoids defined in the three-dimensional image space. The analytical representation of these ellipsoids can be derived with a little algebraic manipulation from equation (2.2). The canonical form for this family of ellipsoids is

$$\frac{4\left(x_\xi - x_m\right)^2}{t_D^2 V^2} + \frac{4\left(y_\xi - y_m\right)^2}{t_D^2 V^2 - 4x_h^2} + \frac{4z_\xi^2}{t_D^2 V^2 - 4x_h^2} = 1, \tag{2.6}$$

where x_m and y_m are the midpoints and x_h is the offset of the input data trace. It can be immediately verified that a zero-offset trace is sprayed along a spherical surface defined by the following equation:

$$\frac{4\left(x_\xi - x_m\right)^2}{t_D^2 V^2} + \frac{4\left(y_\xi - y_m\right)^2}{t_D^2 V^2} + \frac{4z_\xi^2}{t_D^2 V^2} = 1. \tag{2.7}$$

Figure 2.4 shows a three-dimensional rendering of the the zero-offset hemisphere. The spraying surfaces are defined in the three-dimensional image space; therefore, in Figure 2.4 and the

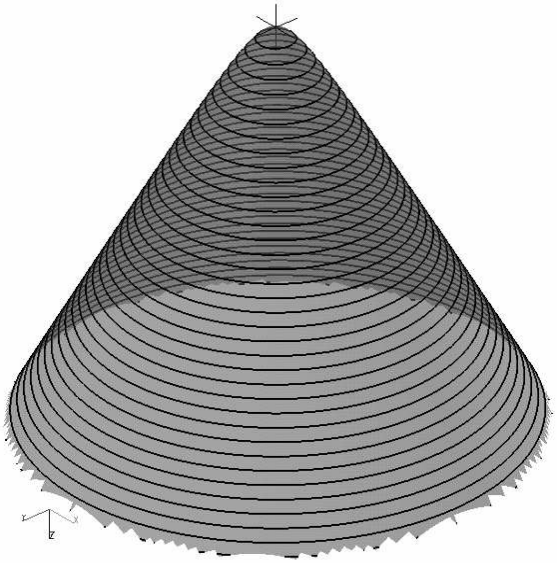


Figure 2.1: Zero-offset summation surface defined in the data space. The vertical axis is the time axis; the horizontal axes are the midpoint axes. The contour lines identify circles with equal time delays. The cross at the apex of the hyperboloid identifies the image point corresponding to the summation surface. The time of the cross is equivalent to the two-way traveltime $\tau_\xi = 2z_\xi / V$. cheops-cheops-zo [NR]

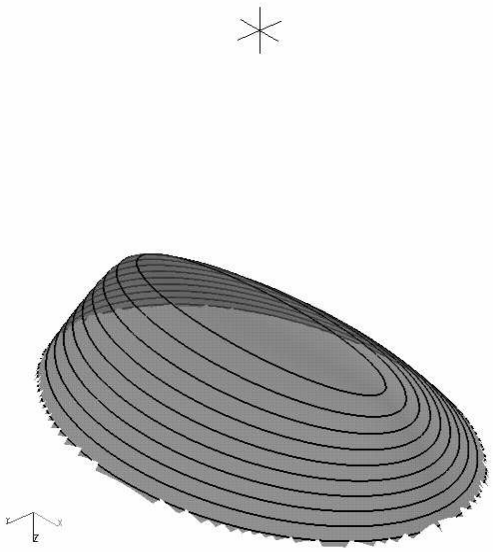


Figure 2.2: Constant-offset summation surface defined in the data space. The vertical axis is the time axis; the horizontal axes are the midpoint axes. The contour lines identify ellipses with equal time delays. The cross identifies the image point corresponding to the summation surface. The time of the cross is equivalent to the two-way traveltime $\tau_\xi = 2z_\xi / V$. cheops-cheops-3km [NR]

two following figures, the vertical axis is depth. The contour lines identify circles with equal depth. The cross barely visible at the bottom of the hemisphere represents an impulse in the input trace. The equivalent depth of the impulse is given by $z_\xi = V\tau_\xi/2$.

A constant-offset trace is sprayed along the ellipsoid that is shown in Figure 2.5. Compared to the zero-offset hemisphere, the constant-offset ellipsoid is squeezed both vertically and along the cross-line direction and thus the contour lines become elliptical. The bottom of the ellipsoid is above the equivalent input impulse, identified by the cross in the figure. Figure 2.6 shows simultaneously both the zero-offset hemisphere and the constant-offset ellipsoid. Notice that the two surfaces become tangent along the in-line edges at the $z_\xi = 0$ plane.

2.2 Migration in complex media

When the velocity is not constant, the time-delay function between the subsurface image point and the source/receiver point at the surface cannot be computed using the simple relationship in equation (2.2); consequently the summation surfaces have more complex shapes than the one represented in Figures 2.1–2.3.

In the more general case, the time-delays used for migration are computed numerically. There are several methods for computing time delays through a complex velocity function. All of them solve the **Eikonal equation**, a high-frequency approximation of the wave equation (Bleistein, 1984). The large majority of the numerical methods are either **ray-tracing** methods (**U**Cervený and P**U**sen**U**cík, 1983; Sava and Fomel, 2001), which solve the Eikonal equation along its characteristics, or finite-difference methods (Vidale, 1990; Popovici and Sethian, 2002).

Figure 2.7, illustrates how the downgoing and the upgoing rays are traced through an interval velocity function $v(z,x,y)$. The resulting time delays can be expressed analytically in the following expression:

$$t_D = t_s\left[\xi, \mathbf{s}, v(z,x,y)\right] + t_g\left[\xi, \mathbf{g}, v(z,x,y)\right], \tag{2.8}$$

where t_s is the time delay from the source location $\mathbf{s}$ to the image point ξ, and t_g is the time delay from the same image point ξ to the receiver location $\mathbf{g}$.

When reflection amplitudes are of interest, the geometrical spreading (a_s, a_g) along the rays connecting the image points and the source/receiver points can also to be computed numerically by solving the associated **transport equation**. The combination of the traveltimes (t_s, t_g) and the amplitudes (a_s, a_g) defines the **asymptotic Green functions** for the wave-propagation phenomenon in the given complex media. According to Green's theorem, these functions can be used to compute the wavefield $P(\omega,\xi)$ generated by a source $P_\mathbf{s}(\omega)$, as

$$P(\omega,\xi) \approx P_\mathbf{s}(\omega)\, a_s \exp(i\omega t_s). \tag{2.9}$$

By assuming that both t_s and a_s are the solution of asymptotic approximations of the wave equation (respectively the Eikonal and the transport equation) and that they are independent

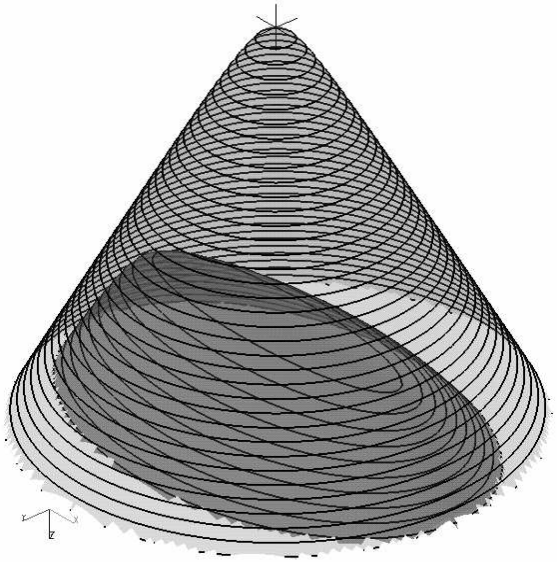


Figure 2.3: Zero-offset and constant-offset summation surfaces defined in the data space. The vertical axis is the time axis; the horizontal axes are the midpoint axes. The cross identifies the image point corresponding to both summation surfaces. The time of the cross is equivalent to the two-way traveltime $\tau_\xi = 2z_\xi/V$. cheops-cheops-zo-3km-side [NR]

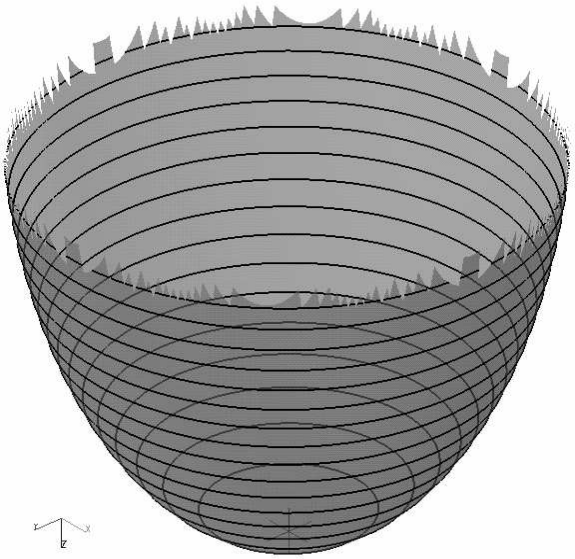


Figure 2.4: Zero-offset spreading surface defined in the three-dimensional image space. The vertical axis is the depth axis. The contour lines identify circles at equal depth. The cross at the bottom of the hemisphere identifies the input impulse that is sprayed along the surface. The equivalent depth of the impulse is given by $z_\xi = V\tau_\xi/2$. cheops-ellips-zo [NR]

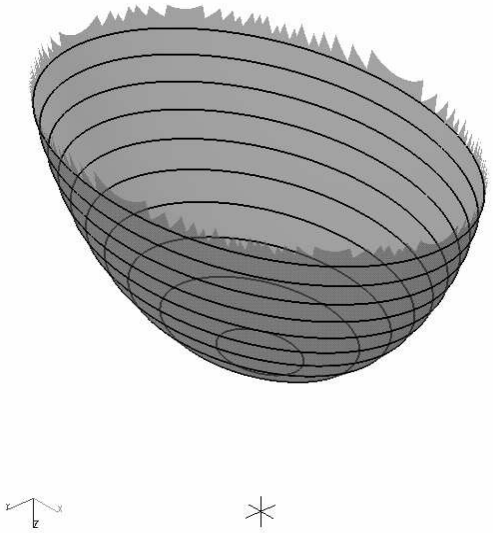


Figure 2.5: Constant-offset spreading surface defined in the three-dimensional image space. The vertical axis is the depth axis. The contour lines identify ellipses at equal depth. The cross at the bottom of the figure identifies the input impulse that is sprayed along the surface. The equivalent depth of the impulse is given by $z_\xi = V\tau_\xi/2$. cheops-ellips-1.5km [NR]

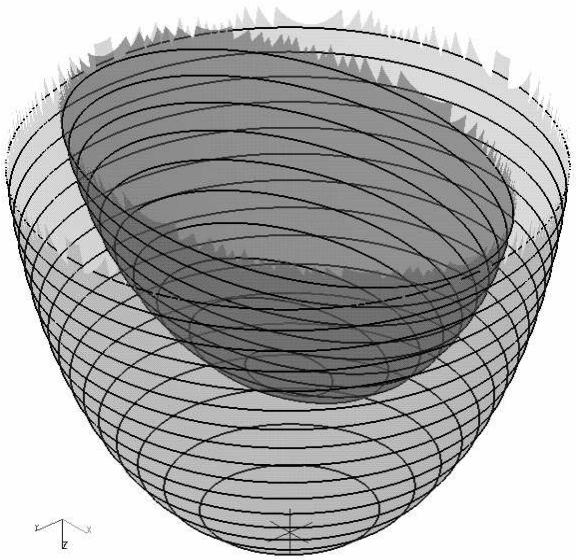


Figure 2.6: Zero-offset and constant-offset spreading surfaces defined in the three-dimensional image space. The vertical axis is the depth axis. The cross at the bottom of the figure identifies the input impulse that is sprayed along the surfaces. The equivalent depth of the impulse is given by $z_\xi = V\tau_\xi/2$. cheops-ellips-zo-1.5km-side [NR]

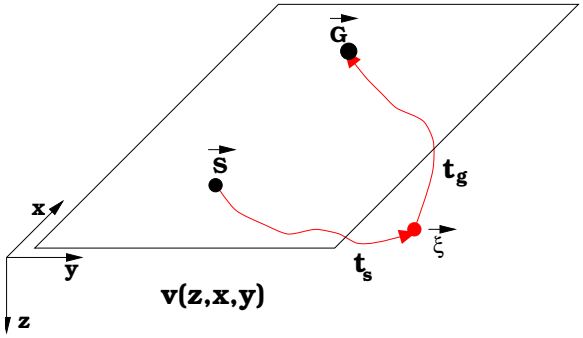


Figure 2.7: Graphical illustration of the raytracing necessary to compute the time delays used for prestack depth migration. cheops-rays [NR]

from the temporal frequency ω, we introduce an asymptotic approximation in the imaging process. This approximation is at the basis of Kirchhoff migration, and constitutes a fundamental theoretical limitation of the method.

When the velocity function is really complex, and the wavefronts triplicate, the Green functions become multivalued, and consequently the summation surfaces become multi-branched. In theory, the computation of the migration integral (summation) is possible for arbitrary complexity of the summation surfaces. In practice, when the multipathing is severe, both the computation of the Green functions and the numerical evaluation of the migration integral become expensive and inaccurate. This is a practical limitation of Kirchhoff migration.

2.2.1 Time vs. depth imaging

It is not always necessary to compute the time-delay functions t_s and t_g numerically. When the velocity is slowly changing in the horizontal direction, the time-delay functions can often be approximated by equation (2.2), but with the constant velocity V replaced by the **root-mean-square velocity** function V_{rms}. The V_{rms} function is related to the interval velocity function by the Dix formula [equation (10.2)]. In these situations, the image is often computed in the time domain, and the migration is called **time migration**, in contrast with **depth migration**.

Their names notwithstanding, the main difference between time and depth migration is not related to the definition of the vertical axis of the image (time or depth). Depth migration uses time-delay functions that are computed numerically by equation (2.8). Time migration uses time-delay functions that are computed analytically as follows:

$$t_D = \sqrt{\frac{\tau_\xi^2}{4} + \frac{\left|\overline{\mathbf{xy}}_\xi - \mathbf{s}\right|^2}{V_{rms}\left(\tau_\xi, x_\xi, y_\xi\right)^2}} + \sqrt{\frac{\tau_\xi^2}{4} + \frac{\left|\overline{\mathbf{xy}}_\xi - \mathbf{g}\right|^2}{V_{rms}\left(\tau_\xi, x_\xi, y_\xi\right)^2}}. \tag{2.10}$$

Notice that to compute the time delays for time migration we need an estimate of average velocities at each image location. In contrast, for depth migration we need an estimate of interval velocity at every point in the subsurface.

As we will see in Chapter 11, the quality of the data focusing can be measured from the image obtained by prestack migration. For time migration, the average velocities can be directly estimated by measuring the focusing quality on a point-to-point basis at each image location, because the average velocity (V_{rms}) enters directly in the expressions of the corresponding summation surfaces. In contrast, there is not a point-to-point direct correspondence between time delays and interval velocity. Therefore, estimating interval velocity from measures of focusing quality in the image is a complex inverse problem, also known as **migration tomography**.

Another crucial distinction between time migration and depth migration, is that time migration only focuses the data, and leaves the task of time-to-depth mapping to a subsequent independent step, that is often called **map migration**. When the velocity is constant, the depth coordinate z_ξ of the image and the corresponding two-way traveltime τ_ξ are uniquely

connected by the simple relationship $z_\xi = V\tau_\xi/2$. When the velocity is not constant, the relationship between two-way traveltime and depth is more complex. However, if the velocity is only a function of depth, the relationship is still univocal and we can use a simple one-dimensional stretching to transform an image defined in time into an image defined in depth, and vice versa. If the velocity varies also laterally, the relationship becomes equivocal. In many cases, time can still be mapped into depth, but the use of **image rays** is required (Hubral, 1977). Image rays are convenient conceptual constructs, but they are not real physical entities.

This splitting of the focusing and mapping steps is important in practice, because in many cases the velocity parameters necessary to focus the data properly (average velocities) are easier to estimate than the velocity parameters necessary for an accurate time-to-depth conversion (interval velocities). In contrast, depth migration relies on interval velocity estimates for both focusing and mapping. When interval velocity estimates are inaccurate, depth migration not only mispositions the reflectors in depth but also misfocuses the data.

The robustness of time migration makes it an essential imaging tool. However, it must be applied within the limitations imposed by its underlying assumptions. When velocity variations are sufficiently strong to create non-hyperbolic diffraction surfaces, time migration cannot focus the data properly, and depth migration becomes necessary. The following example illustrates this issue.

Limitations of time imaging

To illustrate the differences between time and depth migration, we will look at an example from the SEG-EAGE salt data set (Aminzadeh et al., 1996). In particular, we will compare the migration results, and we will analyze the Green functions corresponding to two representative image-point locations (above and below a salt body) for an in-line section. Figure 2.8 shows the velocity model for the section of interest.

Figure 2.9 shows a vertical in-line section of the 3-D prestack migrated cube using time migration. The "true" V_{rms} function was used, as computed from the "true" interval velocity function using the Dix formula. Figure 2.10 shows a vertical in-line section of the 3-D prestack migrated cube using depth migration. The "true" interval velocity function was used for depth migration. The two sections are similar above the salt body, but below the salt body they differ substantially, beyond the obvious laterally-variant stretching of the vertical axis. In particular, the bottom of the salt and the horizontal basement reflector (z=3.6 km) are much better imaged using depth migration. However, even depth migration does not image perfectly all the subsalt reflectors, because some of the reflectors are insufficiently illuminated by the marine-like acquisition geometry of the data. The differences between Figure 2.9 and Figure 2.10 can be easily explained by analyzing how well the time-delay functions computed using equation (2.10) approximate the time delays of the "true" Green functions computed using a wavefield modeling program.

Figure 2.11 shows two snapshots of the wavefield taken at times t=0 seconds and t=0.6 seconds, when the source is located above the salt. Notice the circular shape of the upgoing wavefield. Figure 2.12 shows the wavefield recorded at the surface, as a function of recording

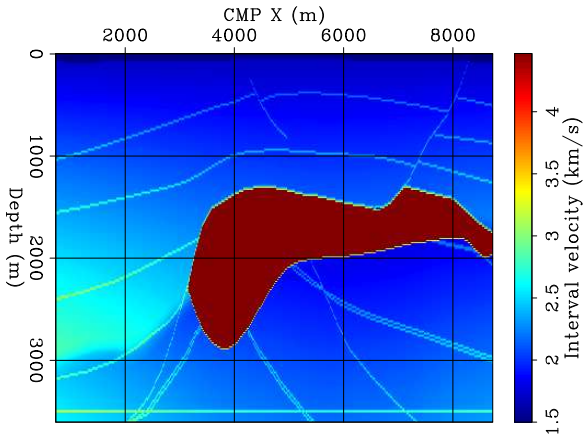


Figure 2.8: In-line section of the interval velocity model for the SEG-EAGE salt data set. cheops-InLine-vel-col [NR]

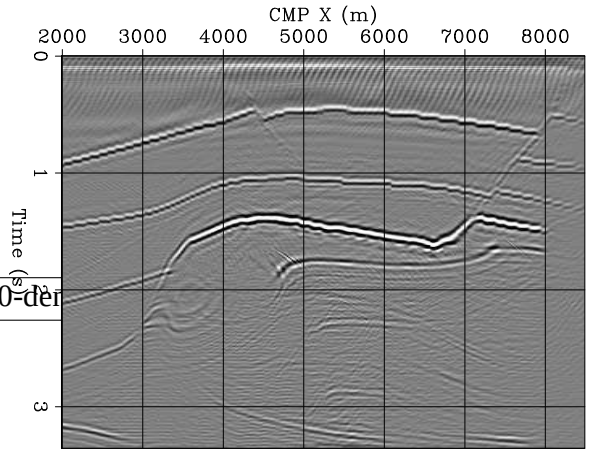


Figure 2.9: 3-D time migration of the in-line section shown in Figure 2.8, computed using the "true" V_{rms} function. cheops-Cheops-Salt-inline-onepass-y5790-der [NR,M]

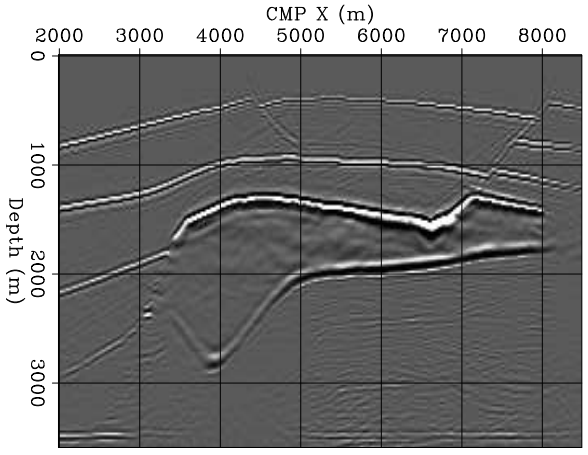


Figure 2.10: 3-D depth migration of the in-line section shown in Figure 2.8, computed using the "true" interval velocity function. cheops-InLine-depth-5790-overn [NR,M]

location and recording time. It represents the full-wavefield Green function in time-space domain corresponding to a reflector point located above the salt. A hyperbolic trajectory, which represents the approximate time-delay function employed by time migration, is superimposed onto the wavefield. In this case, the hyperbolic trajectory overlaps the "true" Green function almost perfectly.

Figure 2.13 and Figure 2.14 show the corresponding results when the image point is located below the salt. Notice that the shape of the upgoing wavefield shown in Figure 2.13 is far from being circular. In Figure 2.14 the analytical hyperbolic trajectory (yellow on blue line) substantially differs form the "true" Green function computed using wavefield modeling. In contrast, the time delay function computed by numerically solving the Eikonal equation (yellow on red line) overlaps the 'true" Green function almost perfectly.

2.3 Computational cost of prestack migration

The computational cost of 3-D prestack migration can be large. There is a clear incentive to reduce the computational cost, especially considering that in complex media several migrations are often necessary to converge to an acceptable velocity model. Therefore, cheaper alternative methods are often applied to image the data, at least at the beginning of an imaging project. However, when choosing between full prestack migration and one of the several cheaper alternatives, a seismologist is often faced with a trade-off between cost and accuracy. For both sides of this trade-off to be understood, it is useful to analyze the computational cost of the different alternatives.

The most expensive computational component of Kirchhoff migration is usually the summation expressed in equation (2.4). Assuming infinite aperture, this summation is an all-to-all operation, and thus its computational cost can be expressed as follows:

$$\text{PreKir} \propto \kappa_{\text{PreKir}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi\right) \times (\text{Nx}_\text{m} \times \text{Ny}_\text{m} \times \text{Nf}), \tag{2.11}$$

where $(\text{Nz}_\xi, \text{Nx}_\xi, \text{Ny}_\xi)$ are the dimensions of the image space, and $(\text{Nx}_\text{m}, \text{Ny}_\text{m}, \text{Nf})$ are the dimensions of the data space, with Nf equal to the nominal fold of the survey. The constant κ_{PreKir} in front of the cost function depends on the migration type. For depth migration it can, in first approximation, absorb the cost of computing and interpolating time delays. Typically κ_{PreKir} ranges between 5 and 30 flop (floating point operations). A simple substitution of typical values for a small marine surveys, such as $\text{Nz}_\xi = 2^2 \times 10^2, \text{Nx}_\xi = 2 \times 10^2, \text{Ny}_\xi = 2 \times 10^2$, and $\text{Nx}_\text{m} = 2^3 \times 10^2, \text{Ny}_\text{m} = 2^3 \times 10^2, \text{Nf} = 10^2$, and $\kappa_{\text{PreKir}} = 2 \times 10$, leads to the huge number of PreKir $\approx 2^{11} \times 10^{13} \approx 2 \times 10^{16}$ flop. Even if a powerful computer, computing at an effective rate of 10^{10}flop/s, could be dedicated to the task, it would result in a *long* 2×10^6 s ≈ 23 days of computing time.

2.3.1 Limited-aperture prestack migration

Fortunately the migration aperture is not infinite, but is limited by considerations of both aliasing and dip range in the image. Typically the aperture has a cone-like shape, such as the

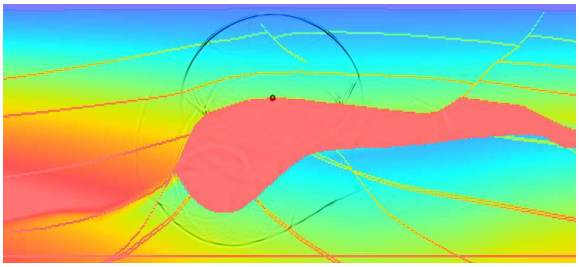


Figure 2.11: Snapshots of the wavefield when the source is located above the salt (t=0 s and t=0.6 s). cheops-Wave-Vel-SEG-SALT-2-t0-t2 [NR]

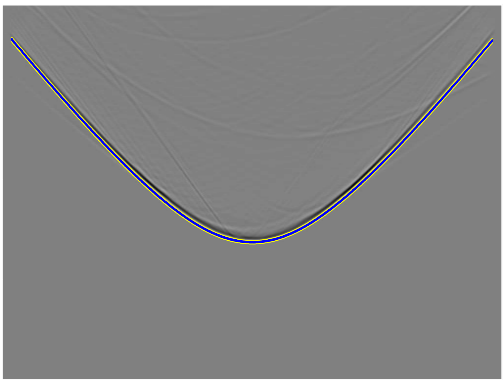


Figure 2.12: Wavefield recorded at the surface, corresponding to the wave modeling shown in Figure 2.11. The hyperbolic trajectory superimposed onto the wavefield represents the approximate time-delay function employed by time migration. cheops-Model-Hyperb-SEG-SALT-2 [NR]

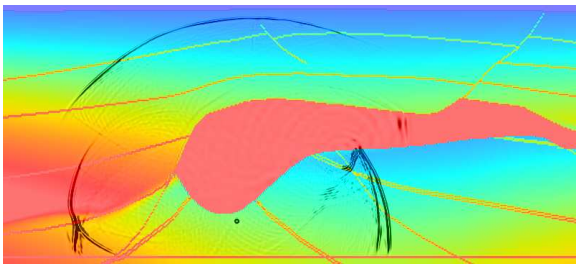


Figure 2.13: Snapshots of the wavefield when the source is located below the salt (t=0 s and t=1 s). cheops-Wave-Vel-SEG-SALT-1-t0-t2 [NR]

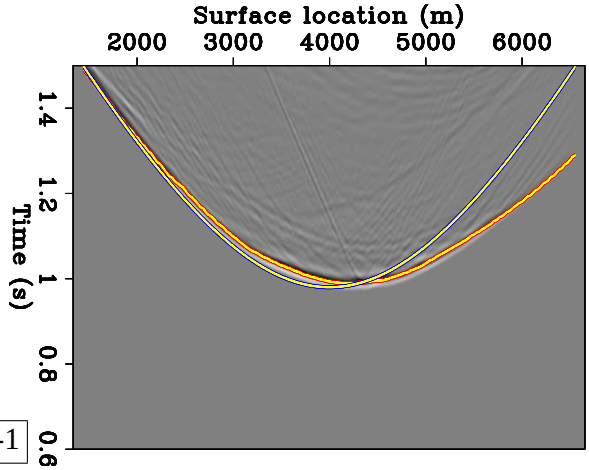


Figure 2.14: Wavefield recorded at the surface, corresponding to the wave modeling shown in Figure 2.13. The hyperbolic trajectories superimposed onto the wavefield represent the approximate time-delay function employed by time migration (yellow on blue line) and the time delay function computed by numerically solving the Eikonal equation (yellow on red line). cheops-Model-Time-Hyperb-SEG-SALT-1 [NR,M]

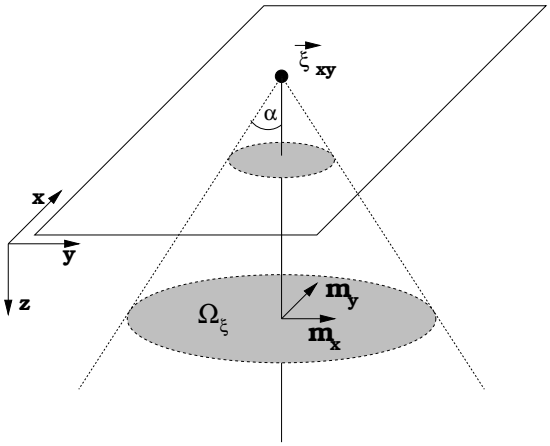


Figure 2.15: Illustration of the cone-shaped migration aperture for Kirchhoff 3-D prestack migration. cheops-aperture [NR]

one illustrated in Figure 2.15. The input midpoints that contribute to a single image-point are limited to a circular area centered around the surface location $\overline{\mathbf{xy}}_\xi$ of the image point. The diameter of this circle is directly proportional to $(z_\xi \times \tan\alpha)$, where α is the aperture of the cone as shown in the figure. Therefore, the cost function expressed in equation (2.11) becomes the more favorable

$$\text{PreKir} \propto \kappa_{\text{PreKir}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi\right) \times \left[\left(\text{Nz}_\xi \times \tan\alpha\right)^2 \times \pi/3 \times \text{Nf}\right], \tag{2.12}$$

that grows in proportion to only the size of the area covered by the image, not in proportion to the area covered by the survey. For our example, limiting the aperture to an angle $\alpha = 45°$ would reduce the total computational time by about a factor of 4, to a more manageable 6 days. However, even when the migration aperture is limited, the cost of 3-D prestack Kirchhoff migration grows with the cube of the target depth, making the full-volume imaging of deep target extremely expensive.

In reality, the migration aperture is seldom the same in the in-line direction as in the cross-line direction. Often the subsurface structure has a predominant dip direction, and the a priori knowledge of this structural directivity can be exploited to reduce the cost of imaging by migrating the data with a smaller aperture in the cross-line direction (α_y) than in the in-line direction (α_x). However, as the following examples illustrate, this practice may lead to incomplete imaging of all reflectors.

The trade-offs between cost and accuracy when setting the migration aperture are shown by the following 3-D migration examples. They show images for the same line as the images presented in Figures 2.9 –2.10; notice however, that these are near-offset (0–.5km) migrations, while the migrations shown in Figures 2.9 –2.10 are full-offset ones (0–2.3 km). The velocity model (Figure 2.8) shows two normal faults above the salt body. Figure 2.16 shows the result of time-migrating the data with a narrow aperture in both the in-line and cross-line direction $(\alpha_x = 30° \text{ and } \alpha_y = 5°)$. Only the flat reflectors in the sediments and the flat parts of the top and bottom of the salt are imaged. When the data are migrated with a wide aperture in the in-line direction $(\alpha_x = 70°)$, and a slightly wider aperture in the cross-line direction

$(\alpha_y = 30^\circ)$, the image of the top and flanks of the salt considerably improves. But still the two fault-planes above the salt are missing (Figure 2.17). These two fault planes have a strong dip-component in the cross-line direction, thus they require full cross-line aperture as well. Only when $\alpha_y = 60^\circ$ the fault planes are nicely imaged (Figure 2.18).

REFERENCES

Aminzadeh, F., Burkhard, N., Long, J., Kunz, T., and Duclos, P., 1996, Three dimensional SEG/EAGE models - an update: The Leading Edge, **2**, 131–134.

Beylkin, G., 1985, Imaging of discontinuities in the inverse scattering problem by the inversion of causal Radon transform: Journal of Mathematical Physics, **26**, 99–108.

Bleistein, N., and Gray, S. H., 2001, From the Hagedoorn imaging technique to Kirchhoff migration and inversion: Geophys. Prosp., **49**, no. 6, 629–643.

Bleistein, N., 1984, Mathematical methods for wave phenomena: Academic Press.

Cohen, J. K., and Bleistein, N., 1979, Velocity inversion procedure for acoustic waves: Geophysics, **44**, no. 6, 1077–1087.

Hubral, P., 1977, Time migration - some ray theoretical aspects: Geophys. Prosp., **25**, no. 4, 738–745.

Popovici, A. M., and Sethian, J. 3-D imaging using higher order fast marching traveltimes:, 2002.

Sava, P., and Fomel, S., 2001, 3-D traveltime computation using Huygens wavefront tracing: Geophysics, **66**, no. 3, 883–889.

Schleicher, J., Tygel, M., and Hubral, P., 1993, 3-D true-amplitude finite-offset migration: Geophysics, **58**, no. 8, 1112–1126.

Schneider, W. A., 1978, Integral formulation for migration in two-dimensions and three-dimensions: Geophysics, **43**, no. 1, 49–76.

UCervený, V., and P**U**sen**U**cík, I., 1983, Gaussian beam and paraxial ray approximation in three-dimansional inhomogeneous media: Bull. Seis. Soc. Am., **70**, 47–77.

Vidale, J. E., 1990, Finite-difference calculation of traveltimes in three dimensions: Geophysics, **55**, no. 05, 521–526.

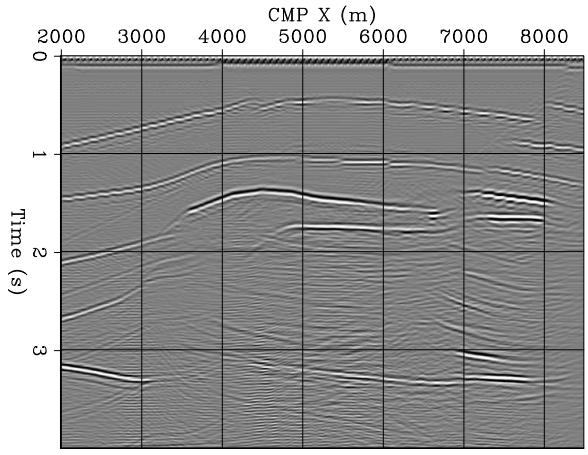


Figure 2.16: Migrated in-line section of the SEG-EAGE salt data set with narrow migration apertures in both the in-line and cross-line direction $(\alpha_x = 30^\circ \text{ and } \alpha_y = 5^\circ)$. cheops-InLine-smx-smy-no [NR,M]

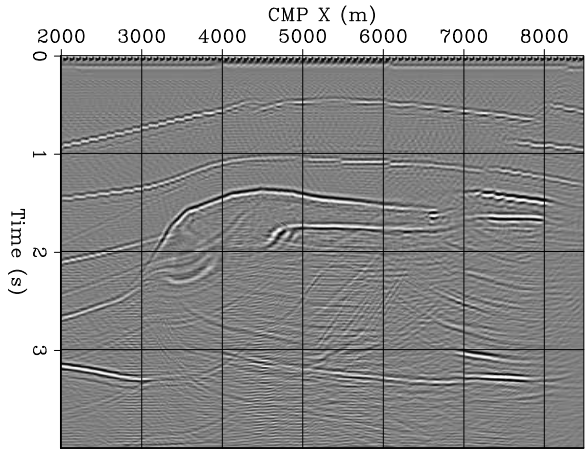


Figure 2.17: Migrated in-line section of the SEG-EAGE salt data set with a wide migration aperture in the in-line direction and a narrow aperture in the cross-line direction $(\alpha_x = 70^\circ \text{ and } \alpha_y = 30^\circ)$. cheops-InLine-wix-smy-no [NR,M]

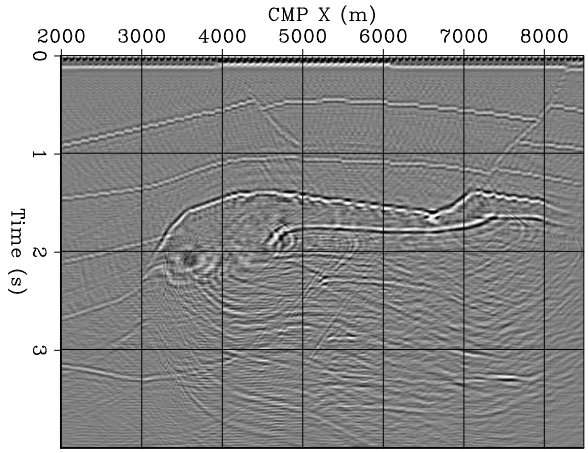


Figure 2.18: Migrated in-line section of the SEG-EAGE salt data set with wide migration apertures in both the in-line and cross-line direction $(\alpha_x = 70^\circ \text{ and } \alpha_y = 60^\circ)$. cheops-InLine-wix-vwiy-no [NR,M]

Chapter 3

Approximations of full prestack migration

Ideally, every seismic data set should be imaged with 3-D prestack migration. In practice, however, 3-D prestack migration is applied to a small, though growing, number of surveys. The computational cost of prestack migration is the main rationale for its limited use. Often, less expensive methods can be applied that yield satisfactory results in less time and with less resources. But computational complexity is not the only consideration. Prestack migration is very sensitive to the choice of the velocity function and to irregular sampling of the data. This chapter introduces approximate methods for imaging prestack data that are less expensive, and often more robust, than full prestack migration.

The most computationally effective method is based on the Normal MoveOut (NMO) operator, which approximately transforms prestack data to equivalent zero-offset data. One of the most basic seismic-processing operators, NMO consists of a simple stretching of the seismic traces. In the presence of dipping reflectors, NMO is not sufficiently accurate, and a more complex operator, Dip MoveOut (**DMO**), becomes necessary to transform prestack data into equivalent zero-offset data. The DMO operator moves energy across the midpoint axes, partially focusing (migrating) the data. Therefore, it belongs to a family of **partial prestack migration** operators.

Another partial prestack migration operator that can be applied to reduce the computational cost of prestack migration is the Azimuth Moveout (AMO) operator. AMO is more general than DMO, because it transforms prestack data into equivalent data with arbitrary offset and azimuth; in contrast, DMO is only capable of transforming non-zero-offset data into zero-offset data. Because of its flexibility, AMO can be thus applied to several 3-D prestack processing problems. This chapter discusses two applications of AMO: 1) partial stacking of prestack data and 2) transformation of marine data to common azimuth. Partial stacking is useful to reduce the size of prestack data and to regularize irregular acquisition geometries. The transformation of marine data to common-azimuth data enables the application of efficient depth-imaging methods, such as the methods described in Chapter 7.

Finally, the last part of the chapter presents the concept of **two-pass migration**. Two-pass migrations achieve computational efficiency by decoupling the migration process along the two horizontal coordinate axes; they are typical examples of full-separation methods. Full-

separation methods are exact only in ideal situations (e.g. constant velocity), but they may achieve satisfactory accuracy in several practically relevant cases.

3.1 Normal moveout

In seismic processing, NMO is important, both as an approximate imaging operator and as a tool to estimate propagation velocity. NMO is a trace-to-trace transformation that remaps the time axis of data traces according to the propagation velocity and the trace offset. For a trace with absolute offset h, the mapping is analytically expressed as follows:

$$t_N = \sqrt{t_D^2 - \frac{4h^2}{V_N^2}}, \tag{3.1}$$

where t_N is the new time coordinate after the application of NMO, and V_N is the NMO or **stacking velocity**. Stacking velocity can be a function of the NMO time and of the midpoint coordinates and has the general functional form of $V_N(t_N, x_m, y_m)$.

To first order, NMO transforms data collected at a finite offset to equivalent zero-offset data. The transformation to zero-offset is only kinematic and approximate; even in constant velocity, the NMO time t_N is exactly equivalent to zero-offset time only when reflectors have no dip component along the source-receiver azimuth of the trace.

NMO is often used in combination with stacking (**NMO+Stack**) and zero-offset migration to approximate full prestack migration. This approximate procedure, which is often called **poststack imaging**, was for a long time the only way to image seismic data. Even today, when more accurate imaging procedures are available to the explorationist, NMO+Stack is still used for the preliminary analysis of the vast majority of seismic data sets.

Poststack imaging is based on the assumption that the **stacked cube** obtained by stacking NMO-transformed traces is a good approximation of the true zero-offset cube. When this assumption is true, the subsurface image can be obtained by applying either zero-offset time or zero-offset depth migration. The large cost advantage of this procedure compared with full prestack migration derives from the data reduction achieved by NMO+Stack. The stacked cube has only three dimensions (time and two midpoints), compared with the five dimensions of prestack data. More importantly, the number of data traces is reduced by a factor equivalent to the average data fold. Because the cost of performing NMO+Stack is negligible compared with performing the subsequent zero-offset migration, the whole chain of processes is cheaper than full prestack migration by a factor proportional to the average data fold.

In addition to its lower computational cost, poststack imaging is attractive because it is a robust procedure with respect to both uncertainties in the velocity estimates and irregularities in the data sampling. The NMO+Stack sequence can be seen as an approximate partial prestack time migration that focuses data along the offset axes. The only velocity information that NMO+Stack needs to focus the data along offsets is the average velocity function $V_N(t_N, x_m, y_m)$, which can be directly measured from the data. The same stacking velocity

function can be also used, possibly with some corrections, to focus the data along the midpoint axis by zero-offset time migration. Alternatively, if the interval velocity can be estimated with some confidence, the stacked cube can be depth-imaged with zero-offset depth migration. Because NMO is a trace-to-trace transformation, with no spreading of energy across midpoints, NMO+Stack is immune to operator aliasing across midpoints. Also, as we discuss in Chapter 8, the offset axis can be severely undersampled without compromising the quality of the stack. Irregular spatial coverage may cause uneven amplitudes in the stacked cube, but in Chapter 9 we discussed simple techniques to balance amplitudes by compensating for irregular coverage.

However attractive and robust, NMO+Stack cannot image all data sets with the required accuracy. When the velocity function is so complex as to cause non-hyperbolic moveouts across offsets, the simple hyperbolic NMO equation is obviously inappropriate to correct for the time delays of prestack traces. We will examine the issues related to non-hyperbolic moveouts in a later chapter. The most common failure of the NMO+Stack procedure is in the presence of dipping reflectors. To analyze the approximations that are implicitly made when the data are imaged by NMO+Stack followed by zero-offset migration, we evaluate analytically the summation (and spreading) surface of the prestack imaging operator that is equivalent to the application of NMO+Stack followed by zero-offset migration. We then compare the surface of this equivalent operator with the full prestack-migration summation surface, as derived in Chapter 2. The derivation starts by substituting the time after NMO t_N for t_D in the zero-offset migration equation (2.5); and yields the following summation surface in NMO time:

$$t_N = 2\,\frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{m}\right|^2}}{V}. \tag{3.2}$$

The equivalent summation surface in the data space is immediately derived by eliminating t_N using the NMO equation (3.1) and by setting $V_N = V$:

$$t_D = 2\,\frac{\sqrt{z_\xi^2 + \left|\overline{\mathbf{xy}}_\xi - \mathbf{m}\right|^2 + h^2}}{V}. \tag{3.3}$$

Equation (3.3) defines an hyperboloid of rotation that is obviously different from the surface of full prestack migration as defined by equation (2.3). Figure 3.1 shows both these summation surfaces. The flattened outer surface is the full prestack-migration surface, and the inner, circularly symmetric surface is the one defined by equation (3.3). The two surfaces are tangent at the top and all along the curve oriented in the cross-line direction y and passing through the apex. The interpretation of this observation is that flat reflections, and in general reflections that have no dip component along the in-line direction, are perfectly migrated by the NMO+Stack and zero-offset migration procedure. All the other reflections in the data are migrated inaccurately when full prestack migration is approximated by its less costly alternative. The degree of inaccuracy varies and is greatest for steeply dipping shallow reflections.

To analyze the approximations introduced by NMO+Stack followed by zero-offset migration, we can also compare the spreading surface of full prestack migration with the spreading

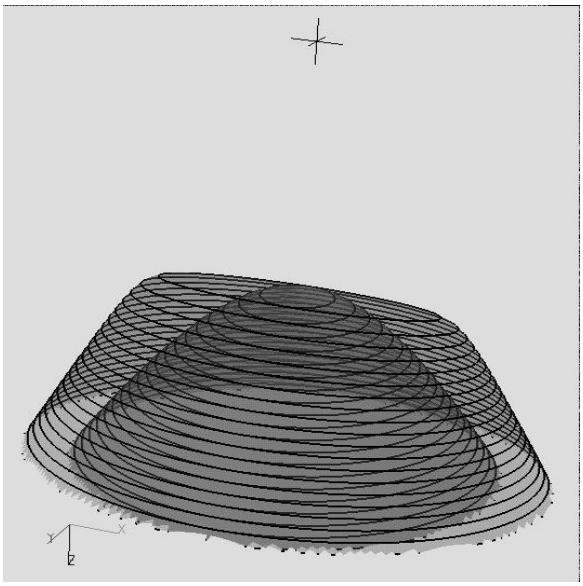


Figure 3.1: Summation surfaces for full constant-offset prestack migration and NMO+Stack followed by zero-offset migration. partial-cheops [NR]

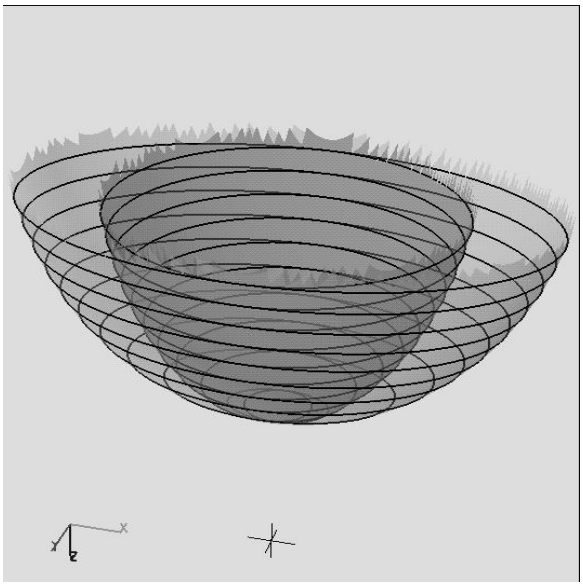


Figure 3.2: Spreading surfaces for full constant-offset prestack migration and NMO+Stack followed by zero-offset migration. partial-ellips [NR]

surface of the poststack imaging procedure. The zero-offset migration hemisphere can be written in terms of NMO time as follows:

$$\frac{4\left(x_\xi - x_m\right)^2}{t_N^2 V^2} + \frac{4\left(y_\xi - y_m\right)^2}{t_N^2 V^2} + \frac{4z_\xi^2}{t_N^2 V^2} = 1. \tag{3.4}$$

Substituting the NMO equation (3.1) into equation (3.4), we obtain the equivalent spreading surface:

$$\frac{4\left(x_\xi - x_m\right)^2}{t_D^2 V^2 - 4h^2} + \frac{4\left(y_\xi - y_m\right)^2}{t_D^2 V^2 - 4h^2} + \frac{4z_\xi^2}{t_D^2 V^2 - 4h^2} = 1. \tag{3.5}$$

This surface is spherical and obviously different from the ellipsoid of full prestack migration presented in equation (2.6). Figure 3.2 shows both the full prestack-migration ellipsoid (outer surface) and its spherical approximation (inner surface). As for the summation surfaces, the exact operator and the approximate operator are tangent along the curve passing through the bottom of the surfaces and oriented in the cross-line direction. Therefore, only the reflectors that are not dipping with respect to the in-line direction are perfectly imaged by the approximate procedure. The larger the dip component along the in-line direction, the larger the error is.

3.1.1 Stacking velocity from dipping reflectors

The errors introduced by the NMO+Stack procedure can also be analyzed, and partially corrected, by making the stacking velocity a function of the reflectors' dip. Consistent with the results of the previous section, the necessary correction to the NMO velocity is greatest when the trace azimuth is aligned with the structural dip direction, and is null when the trace azimuth is aligned with the structural strike direction.

The analytical derivation of the stacking velocity as a function of the reflector dip and data azimuth was first presented by F.K. Levin (1971); we will follow closely his classical derivation. Figure 3.3 shows the geometry of a reflection from a dipping plane. We assume that the source is at the origin of the axis, and that the receiver is on the in-line axis x at a distance $2x_h$ from the source. The derivation is based on the fact that the length of the whole raypath, from the source to the reflection point and back to the receiver, is equal to the path from the source to the **image receiver** $\tilde{g}$. The image receiver is located below the reflector, and belongs to the orthogonal to the reflector passing through the actual receiver location.

If we express the dipping plane as follows:

$$z\cos\alpha + x\cos\theta + y\cos\beta = \bar{d}, \tag{3.6}$$

the equation for the orthogonal line passing through the image receiver is:

$$\frac{z_{\tilde{g}}}{2\cos\alpha} = \frac{x_{\tilde{g}} - 2x_h}{2\cos\theta} = \frac{y_{\tilde{g}}}{2\cos\beta}. \tag{3.7}$$

The coordinates of the image receiver are immediately derived by requiring that the orthogonal line passes through the actual receiver location $(0, 2x_h, 0)$:

$$\begin{aligned} z_{\tilde{g}} &= \left(\bar{d} - 2x_h \cos\theta\right) 2\cos\alpha \\ x_{\tilde{g}} &= \left(\bar{d} - 2x_h \cos\theta\right) 2\cos\theta + 2x_h \\ y_{\tilde{g}} &= \left(\bar{d} - 2x_h \cos\theta\right) 2\cos\beta. \end{aligned} \tag{3.8}$$

The square of the total length of the reflection path is:

$$\begin{aligned} d_{s-g}^2 = d_{s-\tilde{g}}^2 &= \\ z_{\tilde{g}}^2 + x_{\tilde{g}}^2 + y_{\tilde{g}}^2 &= \\ 4\bar{d}^2 + 4x_h^2 - 8\bar{d}x_h\cos\theta &= \\ 4d_{\text{zo}}^2 + 4x_h^2\sin^2\theta, & \end{aligned} \tag{3.9}$$

where the length of the zero-offset path d_{zo} is given by

$$d_{\text{zo}} = \bar{d} - x_h\cos\theta. \tag{3.10}$$

The previous expressions show that NMO velocity is a function of the angle θ between the trace-azimuth direction and the orthogonal to the dipping reflector. Figure 3.4 shows how θ relates to the angle η between the trace azimuth and the dip direction of the reflector, and to the angle α between the vertical axes and the orthogonal to the reflector. These three angles are analytically related by

$$\cos\theta = \sin\alpha\cos\eta. \tag{3.11}$$

The final expression for the NMO velocity is thus

$$V_N = \frac{V}{\sqrt{1 - \sin^2\alpha\cos^2\eta}}. \tag{3.12}$$

Consistent with the results obtained in Section 3.1, the stacking velocity is equal to the medium velocity when either the reflector is flat ($\alpha = 0$) or the dip direction is perpendicular to the trace azimuth ($\eta = \pm 90°$). In these cases, the poststack imaging procedure migrates the reflections exactly. In contrast, when the dip direction is parallel to the trace azimuth ($\eta = 0$) and the reflector is vertical ($\alpha = \pm 90°$), the NMO velocity tends to infinity. In these cases no NMO should be applied to transform the data to zero offset. This result explains why the summation and spreading surfaces for zero-offset and constant-offset migration shown in Figure 2.3 and Figure 2.6 become tangent for vertical in-line dips.

Prestack data without conflicting dips can be correctly transformed to zero-offset, by applying NMO with the stacking velocity given by equation (3.12). However, when there are conflicting dips in the data, a simple trace-to-trace transformation like NMO is insufficient, because it would require using different stacking velocity functions at the same time on the same data. In these cases, a partial prestack migration operator is necessary to transform to zero-offset all the dips in the data. The next section introduces the dip moveout (DMO) operator, which can be used to accomplish this task.

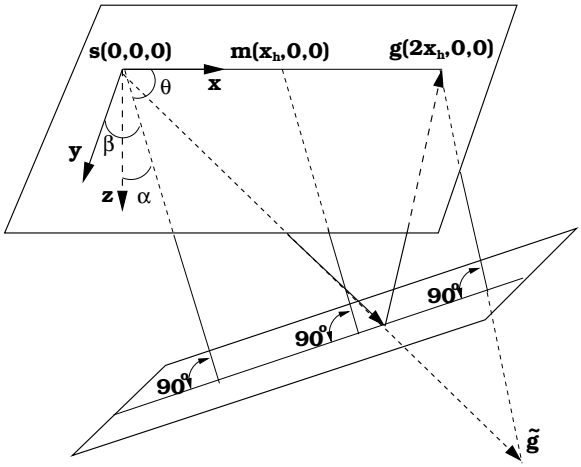


Figure 3.3: Geometry of a reflection from a dipping plane. The image receiver $\tilde{g}$ belongs to the perpendicular to the reflector passing through the actual receiver position. partial-dip-plane [NR]

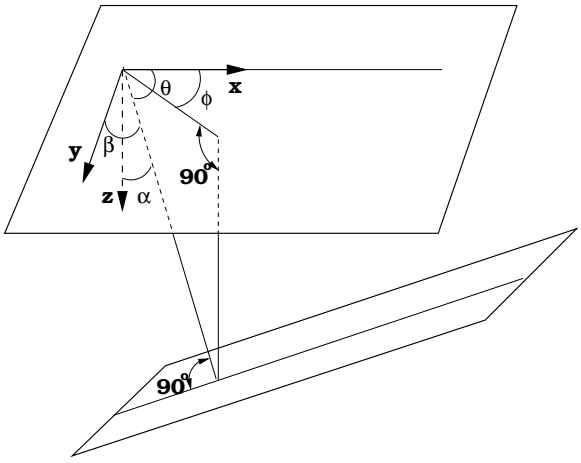


Figure 3.4: Geometrical relationship among θ, α and η. partial-dip-plane-angles [NR]

3.2 Dip moveout (DMO)

To overcome the limitations of the simple NMO operator, DMO introduces a dip-dependent correction in the transformation of prestack data into zero-offset data. We will refer to this improved transformation that includes DMO as **NMO+DMO+Stack**, to distinguish it from the less accurate NMO+Stack.

There are several DMO formulations; the original DMO (Deregowski and Rocca, 1981) was designed to be applied after NMO. More recently, Forel and Gardner (1988) presented an alternative formulation that can be applied before NMO. In either case, if the propagation velocity is assumed constant, the DMO operator is independent from the actual value of the velocity. DMO velocity-independence is attractive because it means that NMO, which is considerably less expensive than DMO, is the only velocity-dependent component of the transformation to zero offset. The advantage of Forel's and Gardner's formulation is that the inexpensive and velocity-dependent NMO operator is applied after the relatively expensive and velocity-independent DMO operator. Therefore, NMO velocity can be updated without reapplying DMO. On the other hand, the application of a residual NMO after using Deregowski's and Rocca's DMO introduces only small errors, if the residual moveout is sufficiently small.

DMO is not a simple trace-to-trace transformation as NMO is, but it performs a *partial prestack migration* of the data by spreading (or summing) energy across the midpoint axes. To derive the constant-velocity DMO spreading surface, we start from the ellipsoid that represents the full prestack-migration spreading surface. In the derivation that follows, which is a generalization to 3-D of the 2-D derivation presented by Hale (1991), DMO is assumed to be applied after NMO.

The full prestack-migration ellipsoid, when the midpoint of the data impulse is located at the origin and the offset vector is oriented along the in-line axis x, can be expressed parametrically as follows:

$$\begin{aligned} z_\xi &= \frac{t_N V}{2} \sin\alpha \cos\beta \\ x_\xi &= \frac{t_D V}{2} \cos\alpha \\ y_\xi &= \frac{t_N V}{2} \sin\alpha \sin\beta, \end{aligned} \tag{3.13}$$

where the angles α and β are shown in Figure 3.5. The zero-offset ray reflected from any point on the ellipsoid belongs to the line orthogonal to the ellipsoid at the reflection point. The expression for this orthogonal line is:

$$\frac{\left(z_{\text{zo}} - z_\xi\right) t_N V}{\sin\alpha \cos\beta} = \frac{\left(x_{\text{zo}} - x_\xi\right) t_D V}{\cos\alpha} = \frac{\left(y_{\text{zo}} - y_\xi\right) t_N V}{\sin\alpha \sin\beta}, \tag{3.14}$$

where z_{zo}, x_{zo}, and y_{zo} are the running coordinates along the zero-offset ray. Setting z_{zo} equal to zero in the previous equation, and using the NMO equation (3.1) and the relationships in equation (3.13), we can determine the horizontal coordinates of the emergence point of the

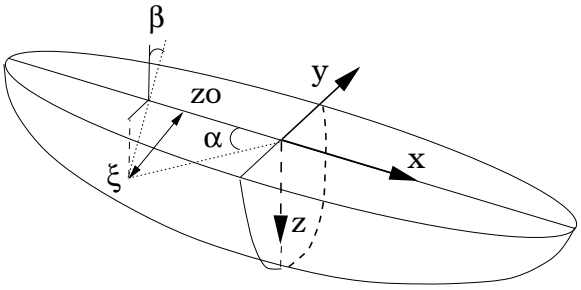


Figure 3.5: Prestack migration ellipsoid and zero-offset ray. partial-dmo-ellipse [NR]

zero-offset ray at the surface, as follows:

$$\begin{aligned} z_{\text{zo}} &= 0 \\ x_{\text{zo}} &= \frac{2h^2 \cos\alpha}{t_D V} \\ y_{\text{zo}} &= 0. \end{aligned} \tag{3.15}$$

We can substitute these values, together with the expressions for the reflection-point coordinates in equation (3.13), into the following expression for the traveltime along the zero-offset ray:

$$\frac{t_{\text{zo}}^2 V^2}{4} = z_\xi^2 + \left(x_\xi - x_{\text{zo}}\right)^2 + y_\xi^2. \tag{3.16}$$

This substitution leads, after a few algebraic manipulations, to the following expression for the DMO operator:

$$\frac{t_{\text{zo}}^2}{t_N^2} + \frac{x_{\text{zo}}^2}{h^2} = 1. \tag{3.17}$$

Equation (3.17) is a functional relationship among the zero-offset time t_{zo}, the NMO time t_N, the absolute offset h, and the zero-offset in-line midpoint x_{zo}. For a given offset h and NMO time t_N, equation (3.17) represents an ellipse defined in the zero-offset time-midpoint space. This ellipse is the spreading surface, or more appropriately the spreading curve, of the DMO operator. The DMO ellipse is coplanar with the offset vector; and the time of the bottom of the ellipse is equal to the NMO time as expected, because flat reflectors do not require a DMO correction. The ellipse passes through the source and receiver locations at the surface. The shape of the ellipse is velocity-independent; however the effective aperture of the operator is velocity-dependent, because the zero-offset midpoint x_{zo} is bound by equation (3.15) to satisfy the following inequality:

$$|x_{\text{zo}}| \leq \frac{2h^2}{t_D V} \leq h. \tag{3.18}$$

3.2.1 Computational cost of poststack imaging

Saving in computational cost is one of the main rationales for using poststack imaging; that is, applying zero-offset migration to data obtained by either NMO+Stack or NMO+DMO+Stack.

In this section we compare the costs of these procedures with the cost of full prestack migration.

The cost of zero-offset migration is equal to the cost of full prestack migration, as defined in equation (2.12), divided by the data average fold; that is,

$$\text{ZoffKir} \propto \kappa_{\text{ZoffKir}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi\right) \times \left[\left(\text{Nz}_\xi \times \tan\alpha\right)^2 \times \pi/3\right]. \tag{3.19}$$

The cost of NMO+Stack is equal to

$$\text{NMO} \propto \kappa_{\text{NMO}} \times (\text{Nt} \times \text{Nx}_\text{m} \times \text{Ny}_\text{m}) \times (\text{Nf}), \tag{3.20}$$

where Nt is the number of time samples of the stacked cube. This cost is negligible compared with the cost of zero-offset migration.

The cost of DMO is proportional to the number of input samples multiplied by the average offset length, Nh/2; that is:

$$\text{DMO} \propto \kappa_{\text{DMO}} \times (\text{Nt} \times \text{Nx}_\text{m} \times \text{Ny}_\text{m}) \times (\text{Nf} \times \text{Nh}/2). \tag{3.21}$$

This is not negligible, but it is still considerably less than the cost of full prestack-migration operators. The reduced computational cost derives from the 2-D nature of the DMO operator as well as from its limited aperture.

However, considering only the number of arithmetic operations underestimates the real cost of stacking and DMO. When only a few arithmetic operations are performed per unit of data, and when either the data or the results need to be accessed in a semi-random fashion, the data-access cost may actually outstrip the computational cost. 3-D DMO is one of these cases.

In many practical situations, the application of 3-D DMO is further complicated by issues related to irregular and incomplete sampling (e.g. spatial aliasing) of the data. Methods exist to prevent the final image from being adversely affected by artifacts caused by poor spatial sampling, as we will discuss in Chapter 8 and Chapter 9. However, these methods increase the computational complexity by making the proportionality factor κ_{DMO} in equation (3.21) substantially larger.

3.2.2 Poststack imaging of SEG-EAGE salt data set

To illustrate the capabilities and limitations of poststack imaging, we analyze the results obtained by processing a subset of the SEG-EAGE salt data set. We focus on the area around the crest of the salt body. Figure 3.6. shows an in-line section of the interval velocity function in this area. The reflections from the steeply dipping flanks present an insurmountable challenge for the simple NMO+Stack procedure. Furthermore, the presence of a rugose salt top on the right side of the salt crest presents problems also for the NMO+DMO+Stack procedure. Some of the dipping events are recorded at later times than the reflections from the top of the salt, as is evident from the stacked sections shown in either Figure 3.8 or Figure 3.9. Therefore, when determining the appropriate NMO velocity for correcting the events before DMO, there

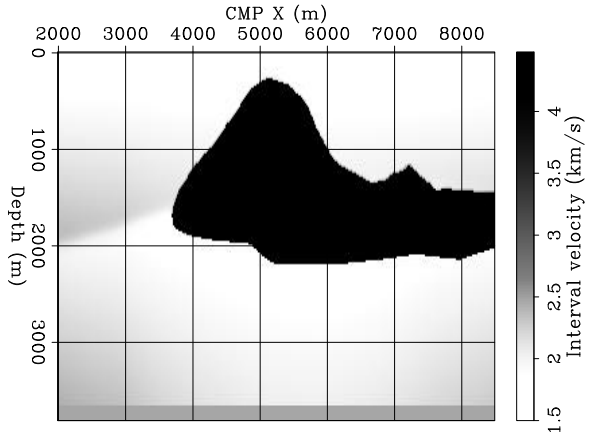


Figure 3.6: In-line section (CMP Y=8 km) of the interval velocity function. partial-Vint_inline-y6800-9200-overn [CR,M]

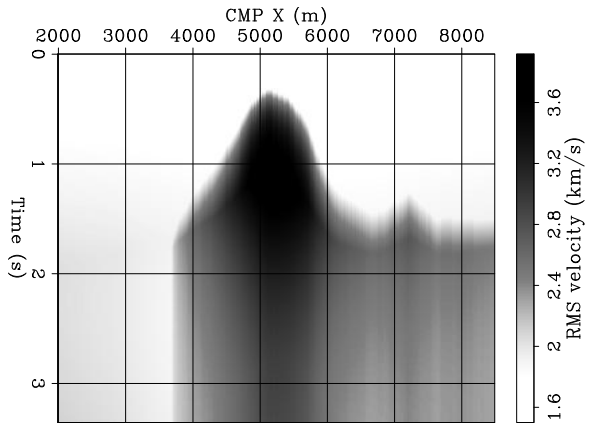


Figure 3.7: In-line section (CMP Y=8 km) of the RMS velocity function. partial-Vrms_inline-y6800-9200-overn [CR,M]

is a potential conflict between the NMO velocity required for the flattish events and the NMO velocity required for the dipping events. The dipping reflections recorded on the right of the crest might be undercorrected if the true RMS velocity, which is displayed in Figure 3.7, is used. In practice, if the aim is to image only the reflectors above the salt body and the top of the salt itself, this problem could be sidestepped by ignoring the salt when setting the NMO velocity. In our example we used the true RMS velocity (Figure 3.7) to illustrate the problems that the NMO+DMO+Stack procedure encounters in complex situations.

Figure 3.8 shows the zero-offset cube obtained by NMO+Stack. The reflections from the steeply dipping salt flanks that are oriented along the in-line direction (CMP X) have been strongly attenuated, whereas the reflections oriented along the strike direction (CMP Y) have been properly stacked. This result is consistent with the theoretical analysis of both the azimuthal dependency of stacking velocity from dipping reflectors (Section 3.1.1), and the errors introduced by NMO+Stack (Section 3.1).

Figure 3.9 shows the zero-offset cube obtained by NMO+DMO+Stack. The reflections from the shallow, steeply dipping salt flanks (above 1.5 seconds) have been properly stacked regardless of their azimuthal orientation. However, the reflections from the deeper portion of the right flank are not properly stacked because of the NMO-velocity conflict discussed above.

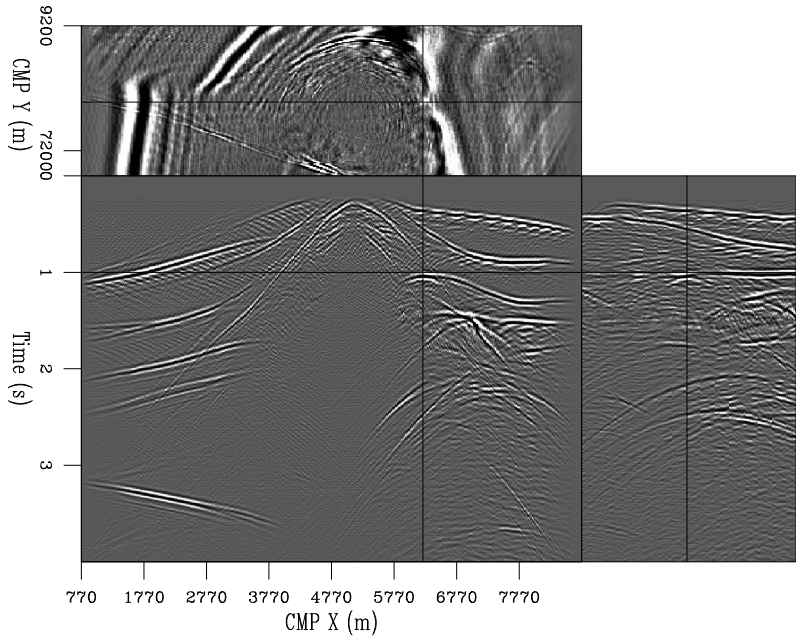


Figure 3.8: Zero-offset cube obtained by NMO+Stack. The reflections from the steeply dipping salt flanks that are oriented along the in-line direction (CMP X) have been strongly attenuated, whereas the reflections oriented along the strike direction (CMP Y) have been properly stacked. partial-Salt-nmo-stack-y6800-9200-overn [CR,M]

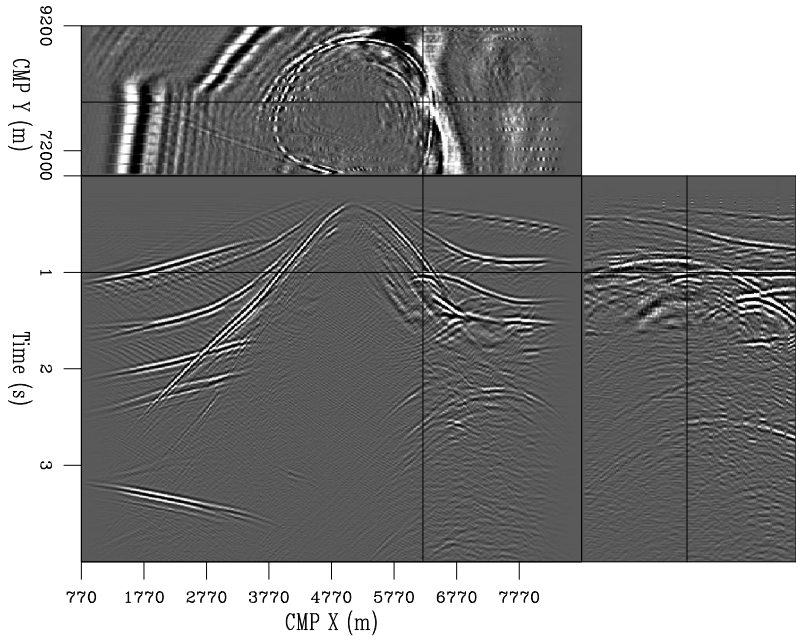


Figure 3.9: Zero-offset cube obtained by NMO+DMO+Stack. The reflections from the shallow parts of the steeply dipping salt flanks have been properly stacked regardless of their azimuthal orientation. However the deeper reflections on the right of the salt crest (below 1.5 seconds) are affected by the NMO-velocity conflict discussed in the text. partial-Salt-dmo-y6800-9200-overn [CR,M]

The limitations of the stacking procedure in the presence of complex geology have direct consequences on the final migrated images. Figure 3.10 is an in-line section (CMP Y=8 kilometers) obtained by full prestack time migration using the true RMS velocity function. It is the best image that we can obtain by time imaging, and thus we use it as a reference.

Figure 3.11 shows the result of migrating by zero-offset time migration the data cube obtained by NMO+Stack (Figure 3.8). As expected, the steeply dipping salt flanks are poorly imaged. The image of the steeply dipping salt flank on the left can be substantially improved by the application of DMO before stacking. Figure 3.9 shows the result of migrating by zero-offset time migration the data cube obtained by NMO+DMO+Stack (Figure 3.9). However, while the left flank is better imaged than in Figure 3.11, the lower part of the right flank and the small peak at CMP X=7 kilometers are more poorly imaged by the migration after the application of DMO than by the migration without it.

In the next section, we will introduce another prestack partial-migration operator (AMO), which allows us to overcome DMO limitations by enabling the coherent stacking of only a subrange of offsets. Because the final imaging requires prestack migration of the partial stacks, the use of AMO instead of DMO increases the computational cost, but it allows substantial savings compared with the computational requirements of full prestack time migration, which is needed to create the image shown in Figure 3.10.

3.3 Azimuth moveout

Prestack partial-migration operators are useful tools to transform prestack data, thereby reducing the computational cost and increasing the accuracy of seismic imaging. The DMO operator introduced in Section 3.2 transforms prestack data to equivalent zero-offset data and thus improves the quality of the stack. This section introduces a more general prestack partial migration called azimuth moveout (AMO). AMO is more general than DMO, in the sense that it transforms prestack data into equivalent data with arbitrary offset and azimuth; in contrast, DMO can only transform non-zero-offset data into zero-offset data.

Azimuth moveout is a prestack partial migration that transforms 3-D prestack data with a given offset and azimuth to equivalent data with different offset and azimuth. The AMO operator is derived by collapsing into a single step the cascade of an imaging operator and its corresponding forward-modeling operator (Biondi et al., 1998). In principle, any 3-D prestack imaging operator can be used for defining AMO. AMO has been derived both as a cascade of DMO and “inverse” DMO (Ronen, 1987; Canning and Gardner, 1992), and as the cascade of full 3-D prestack constant-velocity migration and its inverse. AMO is applied after NMO, thus accounting for velocity heterogeneities, at least to first order.

AMO is not a single-trace to single-trace transformation, but rather a partial-migration operator that moves events across midpoints according to their dip. Its impulse response is a saddle in the time-midpoint domain. Figure 3.13 shows an example of the impulse response of the AMO operator.

To define the analytical expression of the AMO saddle, we analyze the schematic of the

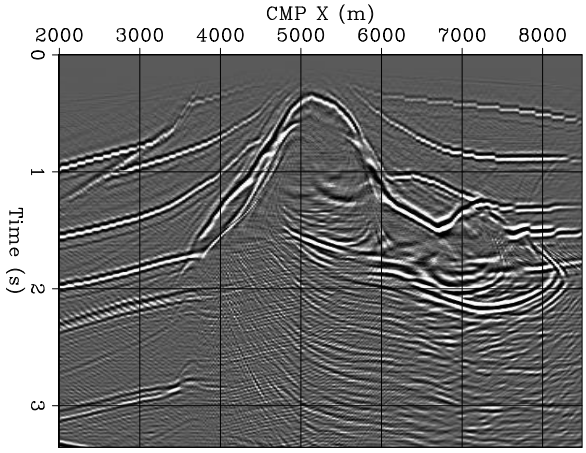


Figure 3.10: In-line section (CMP Y=8 km) obtained by full prestack time migration partial-Salt-migpre-y6800-9200-overn [CR,M]

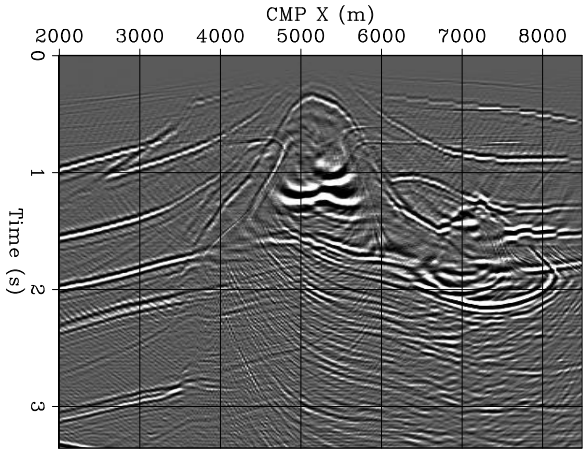


Figure 3.11: In-line section (CMP Y=8 km) obtained by zero-offset time migration of the data cube obtained by NMO+Stack (Figure 3.8). The steeply dipping salt flanks are poorly imaged. partial-Salt-migzo-y6800-9200-overn [CR,M]

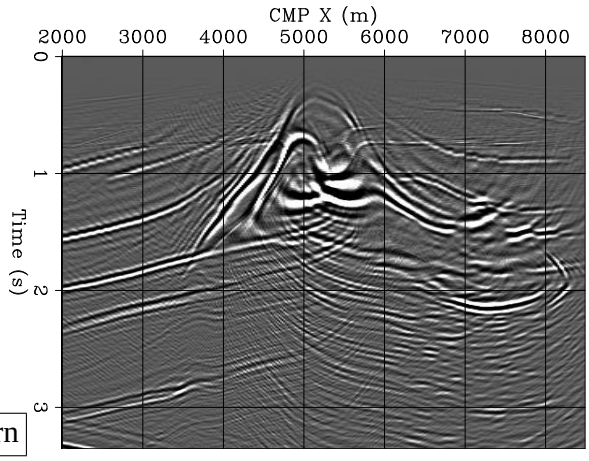


Figure 3.12: In-line section (CMP Y=8 km) obtained by zero-offset time migration of the data cube obtained by NMO+DMO+Stack (Figure 3.9). The steeply dipping salt flank on the left is better imaged than in Figure 3.11. However the salt flanks on the right and the small peak at CMP X=7 km are not well imaged because of the fast variations in RMS velocity associated with the salt body. partial-Salt-migzo-dmo-y6800-9200-overn [CR,M]

AMO transformation in the midpoint plane as displayed in Figure 3.14. An input trace with offset vector $\mathbf{h}_1$ and midpoint at the origin is transformed into equivalent data with offset vector $\mathbf{h}_2$ and with its midpoint shifted by the vector $\Delta\mathbf{m} = (\Delta m_x, \Delta m_y) = \Delta m(\cos\Delta\phi, \sin\Delta\phi)$. The next section (Section 3.3.1) presents the geometrical derivation of the analytical expression for the kinematics of the AMO transformation. The input time t_1 (after NMO) is related to the output time t_2 (before inverse NMO) as follows:

$$t_2 = t_1 \frac{h_2}{h_1} \sqrt{\frac{h_1^2 \sin^2(\theta_1 - \theta_2) - \Delta m^2 \sin^2(\theta_2 - \Delta\phi)}{h_2^2 \sin^2(\theta_1 - \theta_2) - \Delta m^2 \sin^2(\theta_1 - \Delta\phi)}}. \tag{3.22}$$

Equation (3.22) defines a skewed saddle on the $(\Delta m_x, \Delta m_y)$ plane; its shape and spatial extent are controlled by the input time t_1, by the absolute offsets h_1 and h_2, and by the **azimuth rotation** $\Delta\theta = \theta_1 - \theta_2$. Consistent with intuition, the spatial extent of the operator is largest when $\Delta\theta = 90°$, and it vanishes when both the azimuth rotation $\Delta\theta$ and the **offset continuation** $\Delta h = h_1 - h_2$ tend to zero. Furthermore, it can be easily verified that for the zero-dip components of the data, $t_2 = t_1$; that is, the kinematics of zero-dip data after NMO do not depend on azimuth and offset.

The expression for the kinematics is velocity-independent, but the lateral aperture of the operator depends on velocity. An upper bound on the spatial extent of the AMO operator is defined by the region where the expression in equation (3.22) is valid. This region is delimited by the parallelogram with main diagonal $(\mathbf{h}_1 + \mathbf{h}_2)$ and minor diagonal $(\mathbf{h}_1 - \mathbf{h}_2)$, as shown in Figure 3.15. The effective AMO aperture is often much narrower than the parallelogram and is, for given $\mathbf{h}_1$ and $\mathbf{h}_2$, a function of the minimum velocity V_{min} and of the input traveltime (Biondi et al., 1998).

Figure 3.16 shows the effective AMO impulse response when the velocity-dependent aperture limitation, corresponding to a realistic minimum velocity of 2 kilometers, is applied to the impulse response shown in Figure 3.13. The surface shown in Figure 3.16 is significantly narrower than the whole impulse response shown in Figure 3.13. This velocity-dependent aperture limitation is important for efficient use of AMO, and it contributes to the cost-effectiveness of AMO compared to full 3-D prestack migration.

Notwithstanding the constant-velocity assumption underlying its derivation, AMO can effectively be applied to data recorded in areas where the velocity function is complex. The first-order effects of velocity variations are removed by NMO, which is applied before AMO. AMO can successfully transform data to nearby offsets and azimuth when velocity variations are too strong for more global transformations, such as DMO, to transform data correctly. Because AMO is correct to the first order, its results are accurate if the amounts of azimuth rotation $\Delta\theta$ and offset continuation Δh are sufficiently small.

3.3.1 Geometric derivation of the AMO saddle

The AMO operator is defined as the cascade of a 3-D prestack-imaging operator and the corresponding 3-D prestack forward-modeling operator. The expression for the AMO saddle of

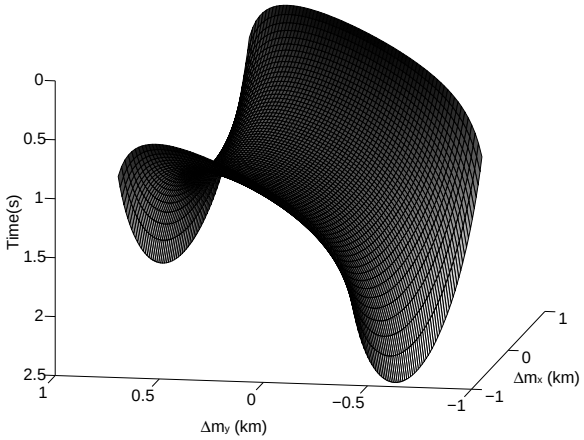


Figure 3.13: The full AMO impulse response ($V_{min} \simeq 0$) when $t_1 = 1$ s, $h_1 = 2$ km, $h_2 = 1.8$ km, $\theta_1 = 0°$, and $\theta_2 = 30°$. partial-amo-max-new [NR]

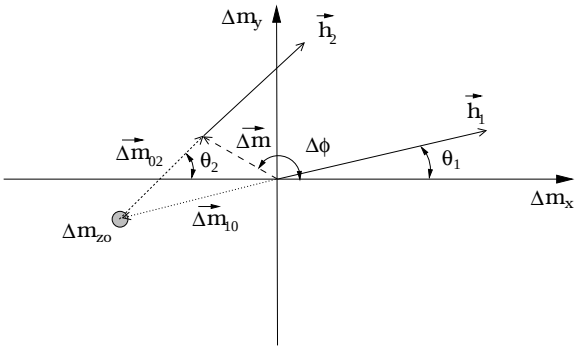


Figure 3.14: Schematic of AMO transformation. An input trace with offset vector $\mathbf{h}_1$ and midpoint at the origin is transformed into equivalent data with offset vector $\mathbf{h}_2$ and with midpoint shifted by the difference vector $\Delta\mathbf{m}$. partial-dmo-cascade-new [NR]

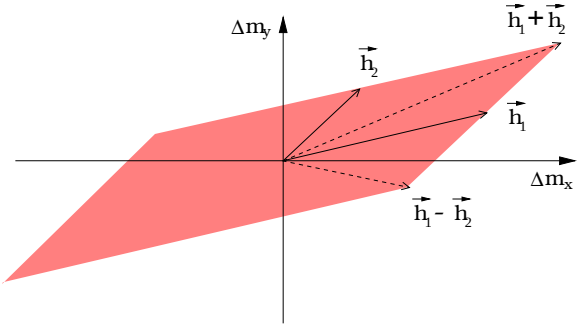


Figure 3.15: Maximum aperture of the AMO operator as a function of the input offset vector $\mathbf{h}_1$ and the output offset vector $\mathbf{h}_2$. partial-max-apert [NR]

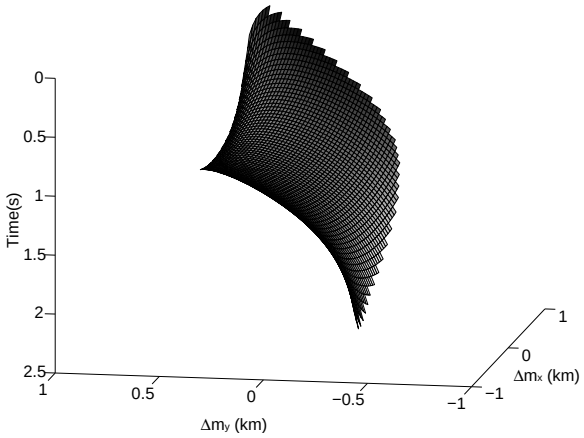


Figure 3.16: The effective AMO impulse response when $V_{min} = 2$ km/s, $t_1 = 1$ s, $h_1 = 2$ km, $h_2 = 1.8$ km, $\theta_1 = 0°$, and $\theta_2 = 30°$. This surface is a subset of the saddle shown in Figure 3.13. partial-amo-eff-new [NR]

equation (3.22) can be derived both as a cascade of DMO and "inverse" DMO and as the cascade of full prestack constant-velocity migration and its inverse (Biondi et al., 1998). Purely kinematic methods as well as wave-equation methods can be used to derive the kinematics and the amplitudes of the AMO operator.

The simplest derivation of AMO is a geometric construction based on its definition as the cascade of DMO and inverse DMO. The schematic shown in Figure 3.14 contains the important elements of this geometric construction, as it was first introduced by Fomel (1995). A trace with input offset vector $\mathbf{h}_1$ and midpoint at the origin is transformed into equivalent data with output offset vector $\mathbf{h}_2$ and midpoint $\Delta\mathbf{m}$ by an intermediate transformation to zero offset. Because DMO is a 2-D operator, the intermediate zero-offset midpoint location Δm_{zo} is uniquely defined as the intersection of the line containing $\mathbf{h}_1$ and the line containing $\mathbf{h}_2$. The vector $\Delta\mathbf{m}$ can be thus decomposed as the sum of the two vectors: the forward DMO midpoint shift $\Delta\mathbf{m_{10}}$ and the inverse DMO midpoint shift $\Delta\mathbf{m_{02}}$. The time shifts corresponding to these transformations are given by the standard DMO transformation [equation (3.17)]:

$$t_0 = t_1 \frac{\sqrt{h_1^2 - \Delta m_{10}^2}}{h_1}, \tag{3.23}$$

and the inverse DMO transformation:

$$t_2 = t_0 \frac{h_2}{\sqrt{h_2^2 - \Delta m_{02}^2}}. \tag{3.24}$$

The AMO transformation is then given by a simple substitution of the intermediate zero-offset time t_0 from equation (3.23) into equation (3.24); that is:

$$t_2 = t_1 \frac{h_2}{h_1} \sqrt{\frac{h_1^2 - \Delta m_{10}^2}{h_2^2 - \Delta m_{02}^2}}. \tag{3.25}$$

The expression for the AMO saddle of equation (3.22) is immediately derived from equation (3.25), by substituting Δm_{10} and Δm_{02} with the following expressions, which can be derived by application of the rule of sines to the triangle $(\Delta\mathbf{m}, \Delta\mathbf{m_{10}}, \Delta\mathbf{m_{02}})$ in Figure 3.14:

$$\left| \frac{\Delta m}{\sin(\theta_1 - \theta_2)} \right| = \left| \frac{\Delta m_{10}}{\sin(\theta_2 - \Delta\phi)} \right| = \left| \frac{\Delta m_{02}}{\sin(\theta_1 - \Delta\phi)} \right|. \tag{3.26}$$

3.3.2 Application to coherent partial stacking

AMO has several applications for 3-D prestack processing. In this section we discuss the use of AMO in conjunction with partial stacking to reduce the computational cost of 3-D prestack depth imaging. As with our previous definitions, we will refer to the sequence of NMO followed by AMO and then by partial stacking as **NMO+AMO+Partial Stack**. By reducing the amount of data to be migrated, partial stacking reduces the cost of 3-D prestack imaging, because the cost of migration is approximately proportional to the amount of data to be migrated [equation (2.12)]. However, for partial stacking to enhance reflections and suppress

noise, reflections must be coherent across the traces to be stacked. Normal moveout increases the coherency of reflections over offset by a first-order correction of the traveltime. However, a simple trace-to-trace transformation such as NMO is insufficient when the reflections have conflicting dips or diffractions occur. By correctly moving the dipping energy across midpoints, AMO insures the preservation of all the dips in the data during partial stacking.

The following examples show the application of AMO to the partial stacking of a data set recorded in the North Sea. To be properly imaged, the data require 3-D prestack depth migration, because of a shallow chalk layer with variable thickness and a deeper salt layer with a central upswelling (Hanson and Witney, 1995). The data-acquisition configuration was double-source and triple streamer. The nominal common-midpoint spacing was 9.375 meters in the in-line direction, and 25 meters in the cross-line direction. The cable length was 2.2 kilometers with maximum feathering of approximately 17 degrees. Figure 3.17 shows the offset-azimuth distribution of a small subset of the data traces. As is typical for such acquisition geometry, the offset-azimuth distribution is characterized by six distinct trends, most distinguishable at small offsets, that correspond to each possible source-streamer pair.

Two distinct partial-stacking methods have been applied to six different subsets of the data: NMO followed by partial stacking (NMO+Partial Stack); and NMO followed by AMO and partial stacking (NMO+AMO+Partial Stack). The subsets were determined according to the absolute value of the offset. Starting from zero, each offset range was .4 kilometers wide. The boundaries between the offset ranges are shown as vertical bars in Figure 3.17. For each offset range, the output data were a regularly sampled cube, with nominal offset equal to the midpoint of the range; that is: .2, .6, 1, 1.4, 1.8 and 2.2 kilometers. The data-reduction rate achieved by partial stacking is approximately 7. In general, the longer the offset and the larger the azimuth rotation, the more significant is the effect of AMO on the data. For the geometry of this data set, the most significant effects are visible for the longer offset ranges, starting from the .8–1.2 kilometers range.

Figure 3.18 compares two time slices obtained with the two flows described above, for the .8–1.2 kilometers offset range. Figure 3.18a shows the slice obtained by simple NMO+Partial Stack, while Figure 3.18b shows the results of NMO+AMO+Partial Stack. The difference plot (Figure 3.18c) clearly shows that most trends of high-frequency diffractions were not correctly preserved by the conventional process. The most evident differences tend to occur for reflections that are perpendicular to the in-line direction. This observation is consistent with the fact that the conventional NMO-stacking process is most inaccurate for reflections with dips that are aligned with the nominal azimuth, and thus they appear orthogonal to the in-line direction in time slices.

Figure 3.19 shows windows of an in-line section for the 1.2–1.6 kilometers offset range. This in-line section is located at 20.940 kilometers and is centered around reflections from the left side of the salt swell. The dipping salt flank reflection is better preserved by the application of AMO. Further, as for the shallower section, some mildly dipping reflections appear to be “cleaner” after AMO. A possible explanation for this phenomenon is that the incoherent stacking of the diffractions, and of other steeply dipping reflections, contributes to the general level of background noise in the data obtained by simple NMO-stacking.

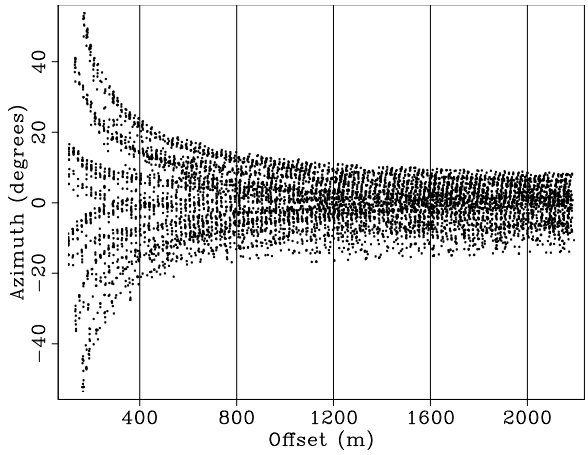
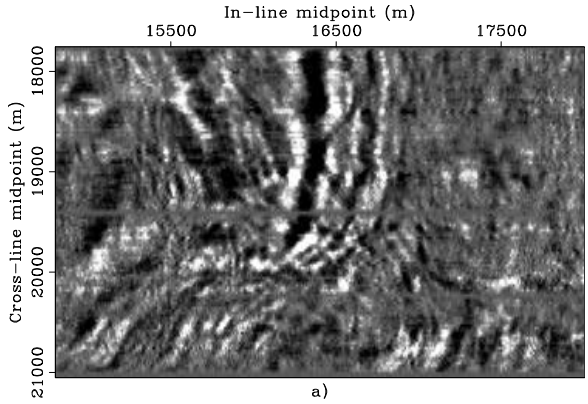


Figure 3.17: Offset-azimuth distribution of the North Sea data set. The vertical bars show the boundaries among the offset ranges that were used for partial stacking. partial-OffAzsm [CR]



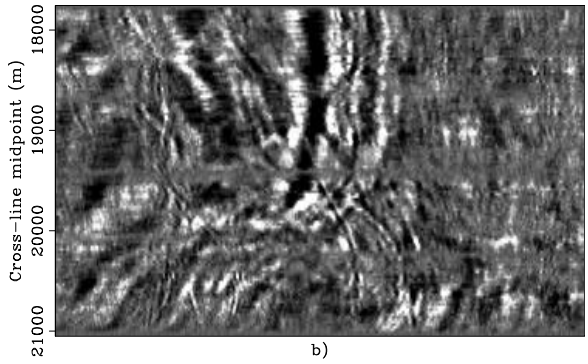
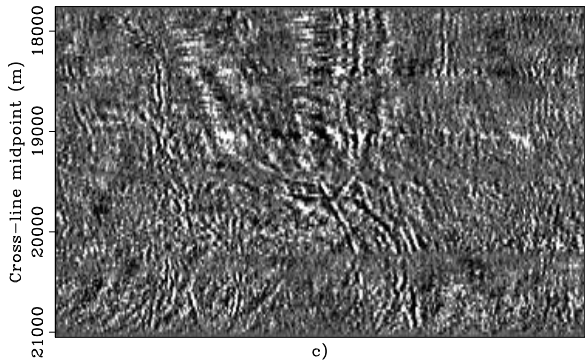


Figure 3.18: Time slices (1.068 s) for the .8–1.2 kilometers offset range, obtained by a) NMO-stacking; b) NMO-AMO-stacking; c) subtracting a) from b). partial-Compts [CR]



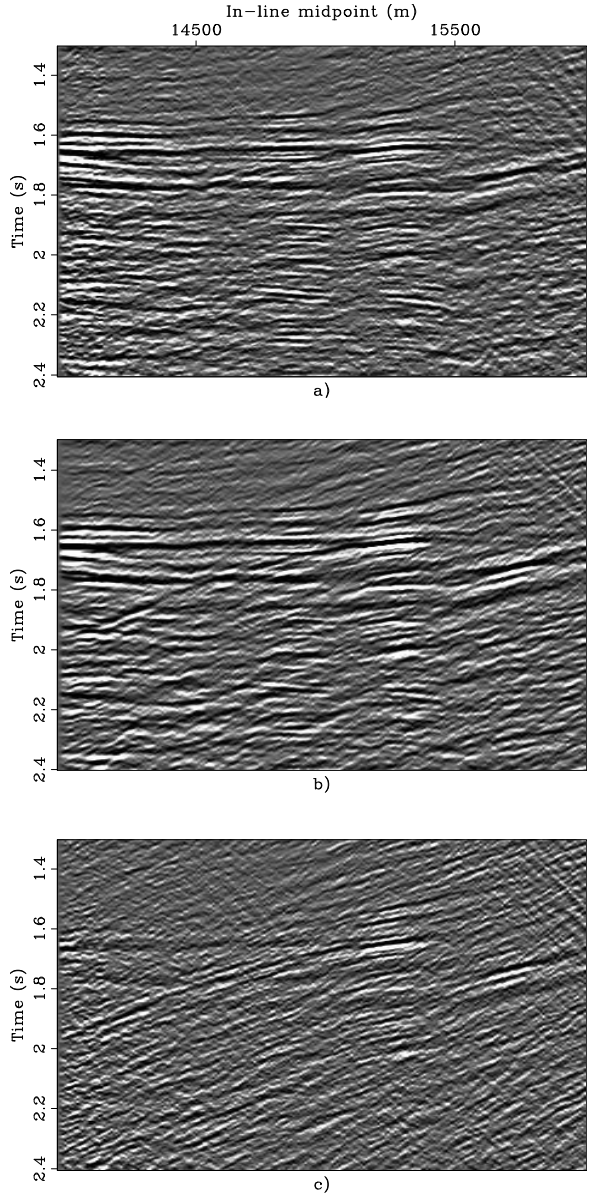


Figure 3.19: In-line sections (CMP Y=20.940 km) for the 1.2–1.6 kilometers offset range, obtained by a) NMO-stacking; b) NMO-AMO-stacking; c) subtracting a) from b). partial-Compll [CR]

Computational cost of prestack imaging by AMO

The computational cost of prestack imaging by AMO mostly resides in the cost of the prestack migrations needed to image the partially stacked data sets. When these migrations are performed using an integral method (Kirchhoff migration), the cost of each migration is similar to the cost of a zero-offset migration, as defined in equation (3.19). Therefore, the overall cost is proportional to the number of offset ranges that are used to perform the partial stacking of the whole data set.

The computational cost of the application of AMO is more complex to characterize, because the operator's effective aperture depends on the relative azimuth rotation and offset continuation between the input and output geometries (Figure 3.15), as well on the minimum

velocity V_{min} and the input traveltime (Figure 3.16). To make the matters even more complex, the anti-aliasing of the AMO operator may add substantial cost to its application (Biondi et al., 1998). A simplified estimate of the cost of AMO is similar to the estimate of cost of DMO expressed in equation (3.21); that is:

$$\text{AMO} \propto \kappa_{\text{AMO}} \times (\text{Nt} \times \text{Nx}_{\text{m}} \times \text{Ny}_{\text{m}}) \times (\text{Nf} \times \text{Nh}/2)\,. \tag{3.27}$$

3.3.3 Transformation to common-azimuth data

Another useful application of AMO to 3-D prestack imaging is the transformation of marine data to equivalent common-azimuth data. Common-azimuth data can be effectively depth imaged with efficient algorithms, such the common-azimuth migration described in Chapter 7. To transform marine data to common-azimuth, AMO needs to perform a simple azimuth rotation, whereas both an azimuth rotation and an offset continuation are required when reducing the data size by partial stacking, as described in Section 3.3.2. Therefore, the effects of including AMO in the common-azimuth transformation are less significant than the effects of including AMO in the partial-stacking process described in Section 3.3.2. The width of the azimuth range in the data geometry is an important factor in determining whether the inclusion of AMO is required for accurate imaging. In general, the more streamers used for the marine acquisition, the more AMO has an impact on the accuracy of common-azimuth imaging. As for partial stacking, the standard alternative to AMO for the transformation of prestack data to common-azimuth data is the application of residual NMO followed by midpoint interpolation.

Figure 3.20 and Figure 3.21 show the effects of including AMO in the common-azimuth imaging procedure. The figures show sections extracted from image cubes obtained by the common-azimuth depth migration described in Chapter 7. The sections on the left were obtained with AMO in the processing sequence; the ones on the right by transforming the data to common-azimuth using residual NMO followed by midpoint interpolation. Figure 3.20 compares depth slices. The fault-plane reflections indicated by the arrows are better imaged in the results obtained with AMO. These fault-plane reflections correspond to the diffractions shown in Figure 3.18, which were better preserved when AMO was used to partial-stack the data. Figure 3.21 compares in-line sections. The image obtained with AMO better preserves the salt-flank reflection (Arrow A) and the reflection from a dolomite rafting inside the salt layer (Arrow B). In general, the inclusion of AMO improves the accuracy of common-azimuth depth imaging even for data sets with a relatively small azimuth range.

3.3.4 Prestack imaging after partial stacking of SEG-EAGE salt data set

To evaluate the improvements that prestack imaging after partial stacking can achieve over poststack imaging, we analyze the results obtained by processing the same area of the SEG-EAGE salt data set that we analyzed in Section 3.2.2. As before, we focus our attention on the reflections from the salt flanks and the small peak at CMP X=7 kilometers. However, instead

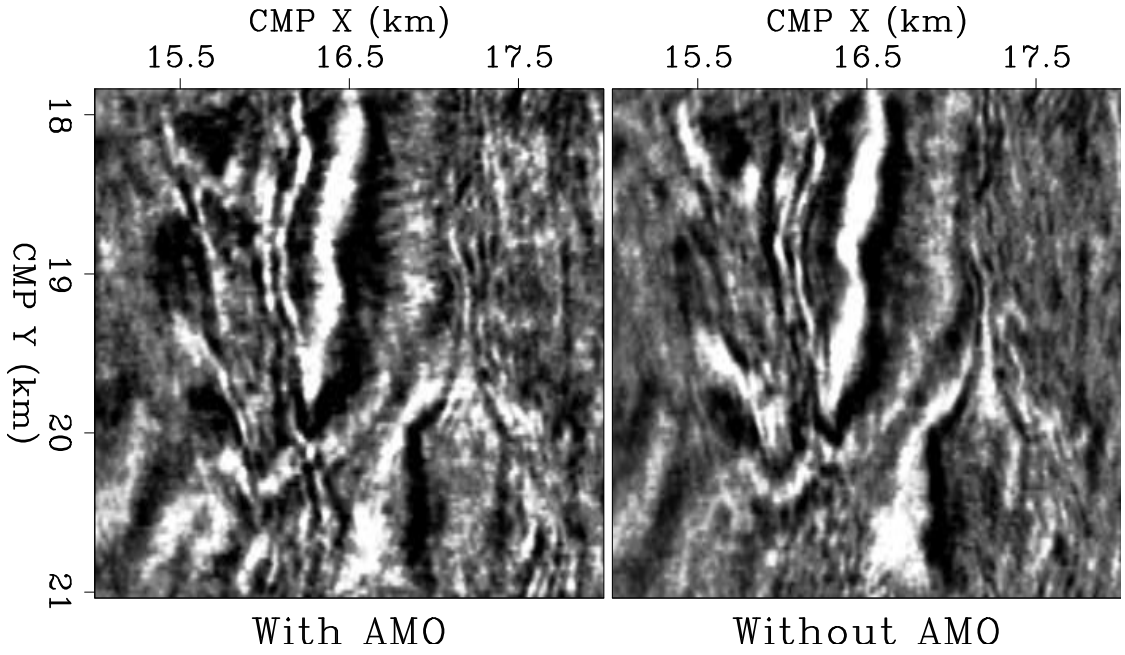


Figure 3.20: Depth slices extracted from two cubes imaged by common-azimuth migration, including AMO (left) and not including AMO (right). partial-dslice-1288-arr [CR]

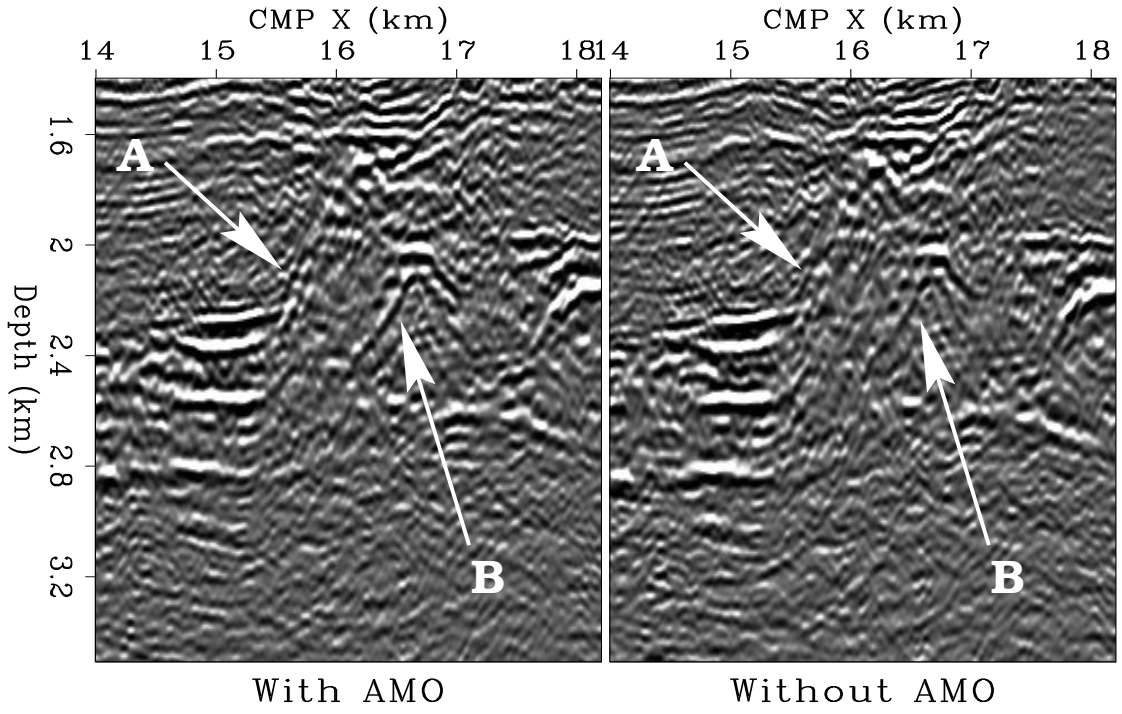


Figure 3.21: In-line sections extracted from two cubes imaged by common-azimuth migration, including AMO (left) and not including AMO (right). partial-xslice-18950-arr [CR]

of imaging the whole offset range in the data (0–2.6 kilometers), we use for imaging only the data traces with absolute offsets within the 0.89–1.53 kilometers range.

Figure 3.22 shows the data cube obtained by NMO+Partial Stack of the 0.89–1.53 kilometers offset range. The reflections from the steeply dipping salt flanks that are oriented along the in-line direction (CMP X) have been attenuated, whereas the reflections oriented along the strike direction (CMP Y) have been properly stacked. Notice that the dipping events are strongly attenuated in a one-second-long time window that starts from the first recorded event (after muting). Consequently, the deeper reflections are less attenuated after NMO+Partial Stack than after NMO+Stack (Figure 3.8).

Figure 3.23 shows the data cube obtained by NMO+AMO+Partial Stack. The reflections from the steeply dipping salt flanks have been properly stacked regardless of their azimuthal orientation. The deeper reflections on the right of the salt crest (below 1.5 seconds), which were poorly stacked after NMO+DMO+Stack (Figure 3.9), are now properly stacked.

The final migrated images in Figure 3.22 and 3.23 show that applying NMO+AMO+Partial Stack instead of NMO+Partial Stack better preserves the dipping events. Figure 3.24 shows an in-line section (CMP Y=8 kilometers) obtained by prestack time migration of the 0.89–1.53 kilometers offset range using the true RMS velocity function. It is the best image that we can obtain by time imaging this particular offset range, and thus we use it as a reference.

Figure 3.25 shows the in-line section obtained by prestack time migration of the data cube obtained by NMO+Partial Stack of the 0.89–1.53 kilometers offset range. The steeply dipping salt flanks are poorly imaged, with the events above one second practically missing from the image. In contrast, these events are as well-imaged by NMO+AMO+Partial Stack (Figure 3.26) as by the prestack migration shown in Figure 3.24. The small peak at CMP X=7 kilometers is not perfectly imaged in Figure 3.26, but it is substantially better imaged than after NMO+DMO+Stack (Figure 3.12).

3.4 Two-pass 3-D prestack migration

The previous sections presented imaging methods that reduce computational cost by decreasing the number of offsets to be migrated: NMO+DMO+Stack reduces the number of offsets to one (zero offset), whereas NMO+AMO+Partial Stack reduces the number of offsets to an arbitrary number, usually lower than the number of offsets in the full prestack dataset. Instead of (or in addition to) reducing the number of offsets, separating the computation between different spatial axes can also reduce the cost of 3-D imaging. This separation reduces the dimensionality, and thus the cost, of imaging operators. There are full-separation methods, which completely decouple the computation along different spatial axes, and there are splitting methods for which the decoupling is only partial, usually at the level of a depth step in downward-continuation algorithms. Splitting methods are justified only on the grounds of computational savings, whereas full-separation methods may also increase the robustness of the imaging procedure, because partially migrated results are more interpretable than the original data.

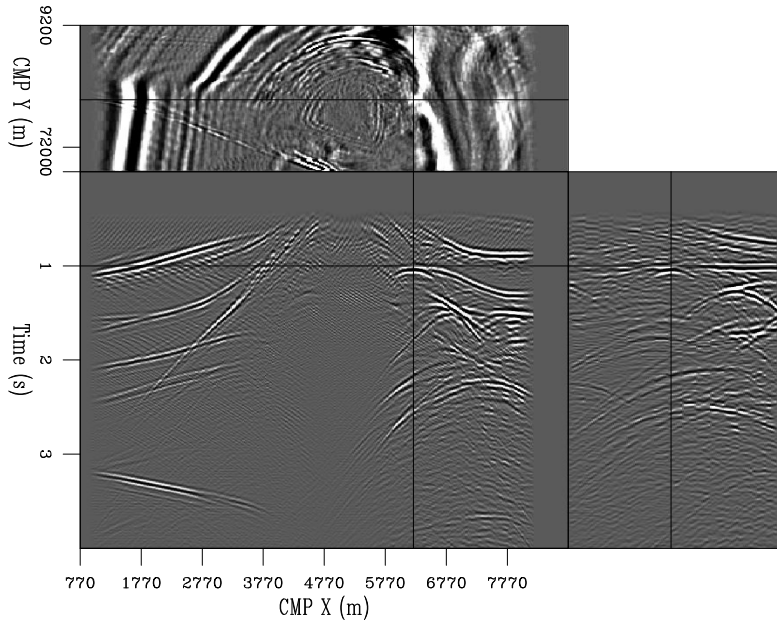


Figure 3.22: Data cube obtained by NMO+Partial Stack of the 0.89–1.53 kilometers offset range. The reflections from the salt flanks that are oriented along the in-line direction (CMP X) have been attenuated. Notice that the dipping events are strongly attenuated in a one-second-long time window that starts from the first recorded event (after muting). partial-Salt-bin-y6800-9200-aoff890-1530-overn [CR,M]

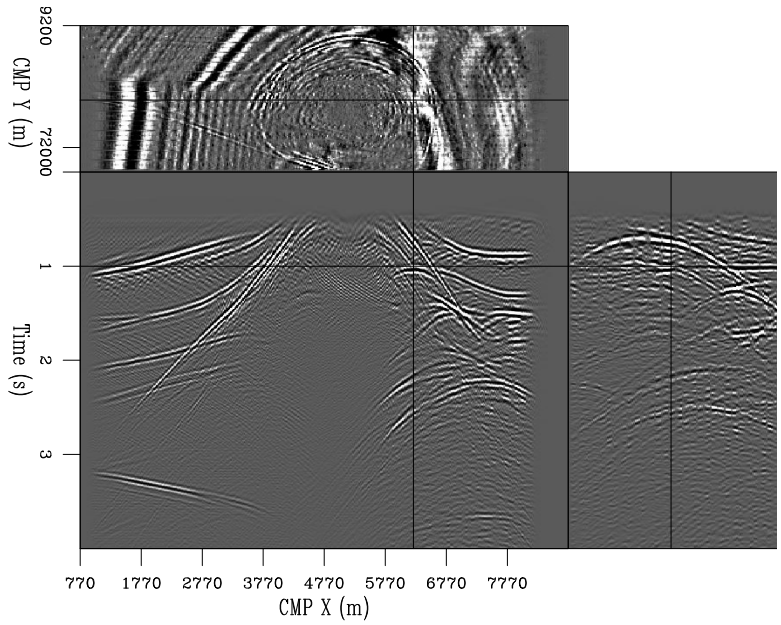


Figure 3.23: Data cube obtained by NMO+AMO+Partial Stack of the 0.89–1.53 kilometers offset range. The reflections from the salt flanks have been properly stacked regardless of their azimuthal orientation. The deeper reflections on the right of the salt crest (below 1.5 seconds), which were poorly stacked after NMO+DMO+Stack (Figure 3.9), are now properly stacked. partial-Salt-amo-y6800-9200-aoff890-1530-overn [CR,M]

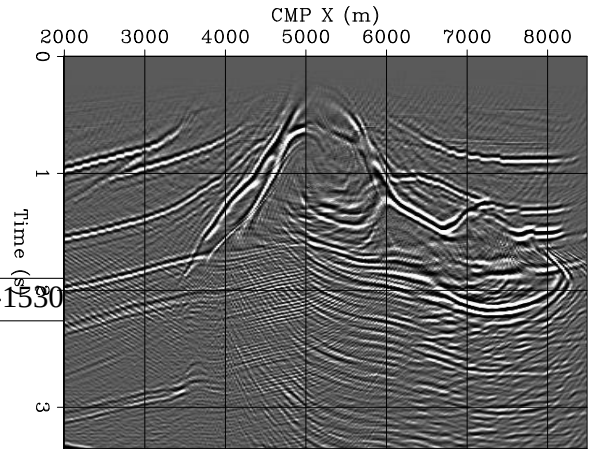


Figure 3.24: In-line section (CMP X=8 km) obtained by prestack time migration of the 0.89–1.53 kilometers offset range. partial-Salt-migpre-y6800-9200-aoff890-1530 [CR,M]

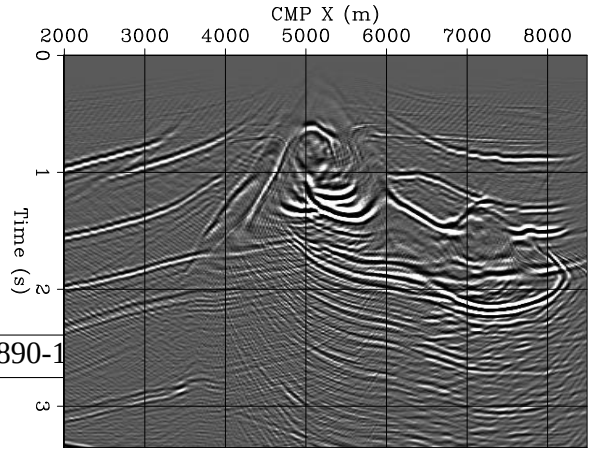


Figure 3.25: In-line section (CMP X=8 km) obtained by prestack time migration of the data cube obtained by NMO+Partial Stack of the 0.89–1.53 kilometers offset range (Figure 3.22). The steeply dipping salt flanks are poorly imaged. partial-Salt-migpre-bin-y6800-9200-aoff890-1 [CR,M]

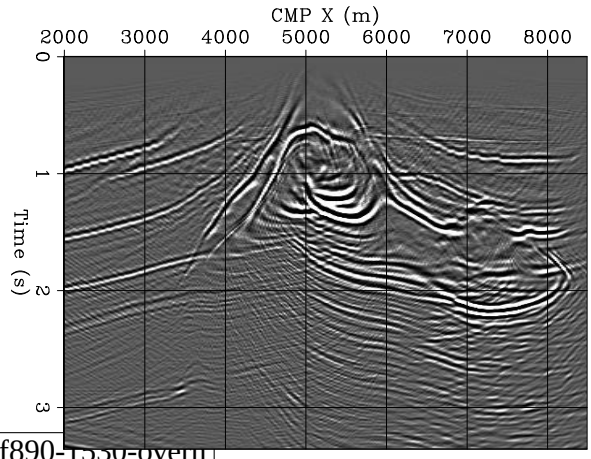


Figure 3.26: In-line section (CMP X=8 km) obtained by prestack time migration of the data cube obtained by NMO+AMO+Partial Stack of the 0.89–1.53 kilometers offset range (Figure 3.23). The steeply dipping salt flanks on both sides of the crest are better imaged than in Figure 3.25. The small peak at CMP X=7 km is not perfectly imaged, but it is substantially better imaged than after the application of DMO (Figure 3.12). partial-Salt-migpre-amo-y6800-9200-aoff890-1530-overl [CR,M]

Two-pass migrations are typical examples of full-separation methods that produce *simplified* intermediary results. The first pass reduces the data to an ensemble of equivalent parallel 2-D lines that can then be migrated and interpreted separately. There are two possible alternative strategies for two-pass 3-D prestack migration. The simplest, and the one we discuss in this section, is to perform 2-D prestack migration along the in-line axis followed by poststack migration along the cross-line axis. Canning and Gardner (1993) and Devaux et al. (1996) proposed to invert the order of the in-line and cross-line migrations; that is, to perform first a "modified" prestack cross-line migration, followed by standard in-line migration. The advantage of this alternative is that the first cross-line migration is fairly insensitive to migration velocity, and thus the problem of 3-D velocity analysis is reduced to an ensemble of independent 2-D velocity-analysis problems for parallel lines.

Constant-velocity prestack migration can be exactly separated into a cascade of prestack migration along the in-line axis followed by poststack migration along the cross-line axis. The analytical proof of full separation is immediate for both summation surfaces and spreading surfaces. If, for the sake of simplicity, the data azimuth is assumed to be aligned with the in-line axis x, it is straightforward to verify that the cascade of prestack migration in the in-line direction [equation (3.28)] with zero-offset migration in the cross-line direction [equation (3.29)] yields the expression of full prestack migration [equation (3.30)]:

$$t_D = \sqrt{\frac{z_{\bar{x}}^2}{V^2} + \frac{\left(x_\xi - x_m + x_h\right)^2}{V^2}} + \sqrt{\frac{z_{\bar{x}}^2}{V^2} + \frac{\left(x_\xi - x_m - x_h\right)^2}{V^2}}, \tag{3.28}$$

$$z_{\bar{x}} = \sqrt{z_\xi^2 + \left(y_\xi - y_m\right)^2}, \tag{3.29}$$

$$\begin{aligned} t_D &= \sqrt{\frac{z_\xi^2}{V^2} + \frac{\left(x_\xi - x_m + x_h\right)^2 + \left(y_\xi - y_m\right)^2}{V^2}} \\ &+ \sqrt{\frac{z_\xi^2}{V^2} + \frac{\left(x_\xi - x_m - x_h\right)^2 + \left(y_\xi - y_m\right)^2}{V^2}}. \end{aligned} \tag{3.30}$$

Figure 3.27 illustrates this procedure for the special case of zero-offset migration. The data are first summed along the parallel hyperbolas aligned along the in-line direction (thinner lines in the figure). The results of each of these partial migrations are set at the apexes of the hyperbolas. The cross-line migration hyperbola, which is drawn with a thicker line in the figure, passes through the apexes of all the in-line hyperbolas. Summation along this cross-line hyperbola focuses the data at the correct image point, which is marked by a cross in the figure.

Figure 3.28 illustrates the prestack procedure, which is similar to, but slightly less obvious than, the zero-offset one. In this case the results of the summations along the parallel flat-top hyperbolas (thinner lines in the figure) are set at focal points above the summation curves themselves. These focal points belong to the same cross-line hyperbola as in the zero-offset case, and their energy is focused at the correct image location by use of cross-line migration.

The derivation of the dual result - that is, the proof that prestack migration by spreading also fully separates - is even simpler than the previous one. To obtain the ellipsoid of full

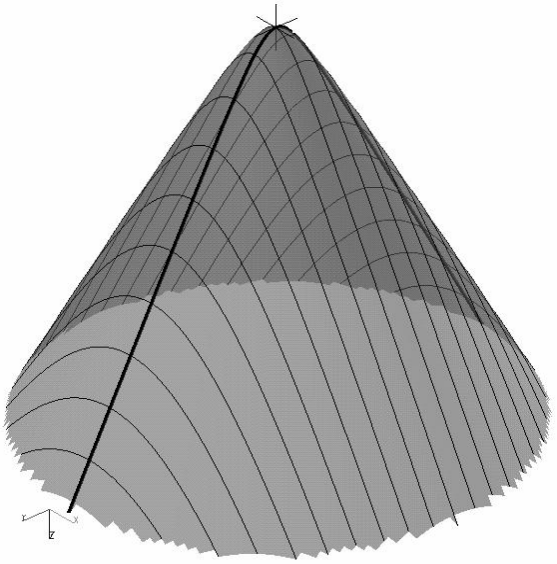


Figure 3.27: Full separation of the summation surface for zero-offset migration. partial-cheops-zo-twopass [NR]

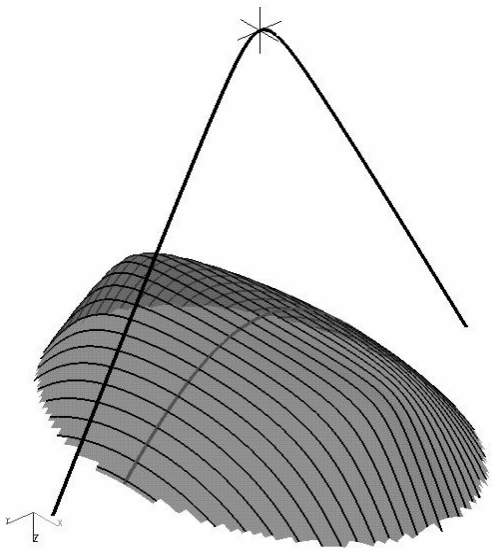


Figure 3.28: Full separation of the summation surface for constant-offset migration. partial-cheops-3km-twopass [NR]

prestack migration [equation (3.33)], we first spread energy along the in-line ellipse [equation (3.31)], and then spread energy along the cross-line circles [equation (3.32)]:

$$\frac{4\left(x_\xi - x_m\right)^2}{t_D^2 V^2} + \frac{4z_{\frac{x}{2}}^2}{t_D^2 V^2 - 4h^2} = 1, \tag{3.31}$$

$$\frac{\left(y_\xi - y_m\right)^2}{z_{\frac{x}{2}}^2} + \frac{z_\xi^2}{z_{\frac{x}{2}}^2} = 1, \tag{3.32}$$

$$\frac{4\left(x_\xi - x_m\right)^2}{t_D^2 V^2} + \frac{4\left(y_\xi - y_m\right)^2}{t_D^2 V^2 - 4h^2} + \frac{4z_\xi^2}{t_D^2 V^2 - 4h^2} = 1. \tag{3.33}$$

Figure 3.29 illustrates the two-pass spreading of an impulse over the zero-offset hemisphere. First the impulse (a cross in the figure) is spread along the in-line circle (thick line in the figure). Second, each point on the circle is spread along parallel circles (thin lines in the figure) oriented in the cross-line direction.

The prestack case is similar to the poststack one. Figure 3.30 illustrates the two-pass spreading of an impulse over the prestack ellipsoid. First the impulse (a cross in the figure) is spread along the in-line ellipse (thick line in the figure). Second, each point on the ellipse is spread along parallel circles (thin lines in the figure) oriented in the cross-line direction.

Applying the cascade of equation (3.31) with equation (3.32), or equivalently the cascade of equation (3.28) with equation (3.29), is less computationally intensive than applying the full migration operators [respectively equation (3.33) and equation (3.30)], because a cascade of two summations over 2-D curves replaces one summation over a 3-D surface. Furthermore, the cross-line migration is only a zero-offset migration. The cost function of two-pass prestack migration with infinite aperture can be written as follows:

$$\text{PreKirSep} \propto \kappa_{\text{PreKirSep}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi\right) \times \left(\text{Nx}_\text{m} \times \text{Nf} + \text{Ny}_\text{m}\right). \tag{3.34}$$

Consequently, the cost function for two-pass migration with infinite aperture is reduced by a factor of $\left(\text{Nx}_\text{m} \times \text{Ny}_\text{m} \times \text{Nf}\right) / \left(\text{Nx}_\text{m} \times \text{Nf} + \text{Ny}_\text{m}\right) \approx \text{Ny}_\text{m}/2$, compared with the cost function for full-prestack migration as defined in equation (2.11).

Similarly, the cost function of two-pass prestack migration with limited aperture can be written as follows:

$$\text{PreKirSep} \propto \kappa_{\text{PreKirSep}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi\right) \times \left(\text{Nz}_\xi \times \tan\alpha\right) \times \left(\text{Nf} + 1\right), \tag{3.35}$$

which is smaller by a factor of $\left(\text{Nz}_\xi \times \tan\alpha \times \pi/3 \times \text{Nf}\right) / \left(\text{Nf} + 1\right) \approx \text{Nz}_\xi \times \tan\alpha$, compared with the cost function for full-prestack migration as defined in equation (2.12).

Two-pass migrations are computationally convenient, not only because of the low flop-count shown in the previous equation, but also because they allow access to the data a single line at a time (in-line or cross-line). When the size of the whole data set is large compared with the computational resources available, this favorable data-access pattern can be as strong an incentive as the low flop-count for the use of two-pass migrations.

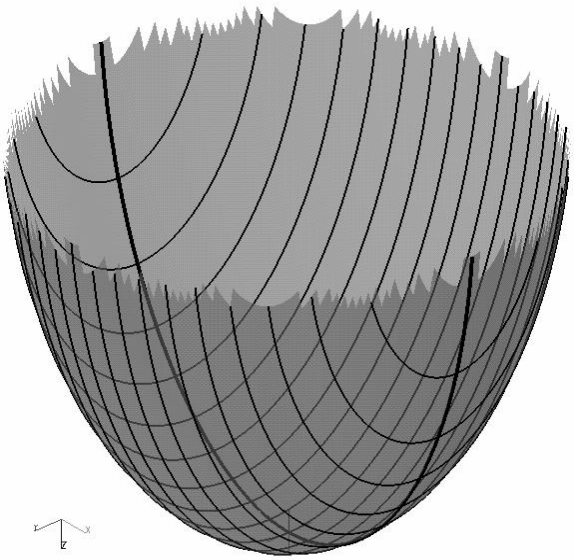


Figure 3.29: Full separation of the spreading surface for zero-offset migration. partial-ellips-zo-twopass [NR]

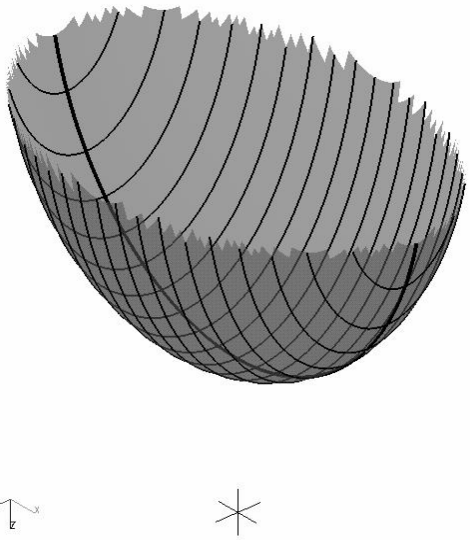


Figure 3.30: Full separation of the spreading surface for constant-offset migration. partial-ellips-1.5km-twopass [NR]

The considerable reduction in cost achieved by full separation justifies its popularity for zero-offset migration at the beginning of the development of 3-D seismic (Gibson et al., 1983). However, two-pass migrations are inaccurate when the velocity is not constant, even if it varies only with depth. The errors are larger at steeper dips and for strong velocity gradients (Dickinson, 1988). Further, the generalization of two-pass methods to true depth-migration problems is arduous. The next section presents a geometric interpretation of two-pass migration, which provides an intuitive insight into its accuracy limitations. Nowadays, accurate and reasonably inexpensive alternatives to two-pass zero-offset migrations are available. They are mostly based on downward-continuation methods, which will be introduced in Chapter 4

However, more recently, two-pass migration has been revived with 3-D prestack migration applications in mind. Beyond being computationally inexpensive, two-pass prestack migrations promise to simplify the estimation of 3-D velocity models for depth imaging, because they reduce the 3-D problem to a series of independent 2-D problems (Devaux et al., 1996).

3.4.1 Geometric interpretation of two-pass migration

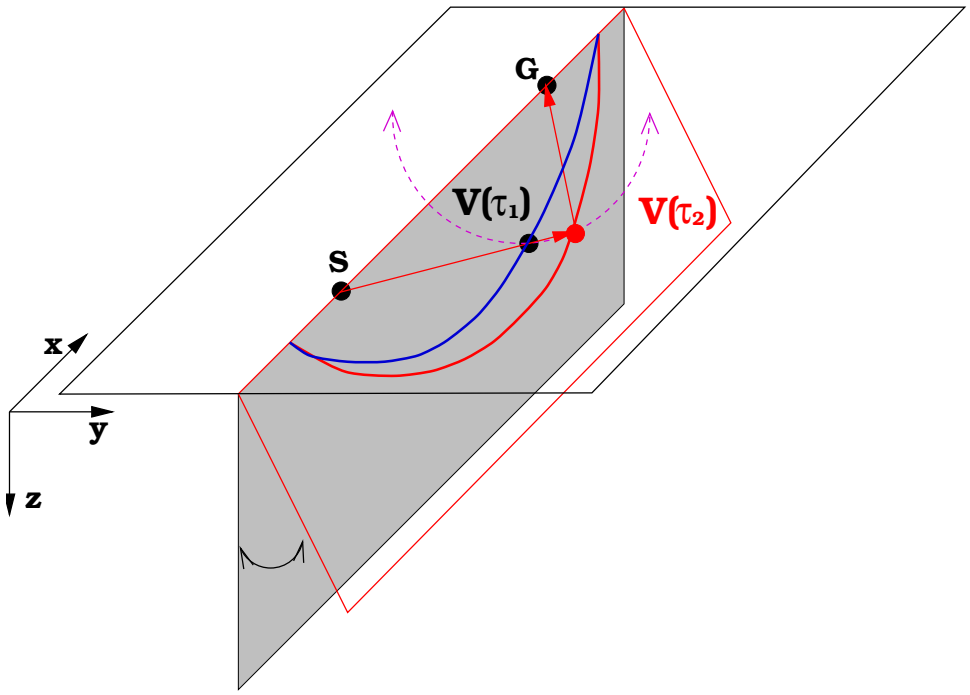


Figure 3.31: Geometric interpretation of two-pass migration. partial-two-pass [NR]

Two-pass prestack migration in a constant velocity medium has a simple geometric interpretation, which is illustrated by Figure 3.31. Because the velocity is constant and no ray-bending occurs, a reflection recorded at the surface with a source location **s** and receiver location **g** must propagate along one of the planes passing though the shot and receiver locations. In the general case this **propagation plane** is oblique. Migration should spread reflections along ellipses belonging to their respective propagation planes. The spreading can

be achieved in two steps: first by spreading along ellipses lying on the vertical plane (in-line prestack migration), and then by rotating the vertical ellipse until it is coplanar with the the actual propagation plane (cross-line zero-offset migration). When the velocity is constant, the two-step procedure is perfectly equivalent to a single-step procedure. The shape of the prestack ellipse is independent of the dip of the actual propagation plane. The arbitrary choice of the vertical plane for the first step is thus inconsequential. But when the velocity is not constant, the shape of the prestack ellipse depends on the dip of the actual propagation plane. Consequently, any one choice cannot be accurate for all the reflections. Even in the simplest case of velocity varying only as a function of depth, the V_{rms} of a reflection propagating along the vertical plane can be significantly different than the V_{rms} of a reflection propagating along a steep oblique plane. For example, if the velocity monotonically increases with depth, the RMS velocity $V(\tau_1)$ measured along the vertical plane is always larger than the RMS velocity $V(\tau_2)$ measured along the oblique plane shown in the figure. The steeper the propagation plane, the larger the error is.

This ambiguity in the shape of the first in-line migration ellipse makes it difficult to generalize full-separation methods to depth migration. However, the basic idea that operators can be separated along the in-line and the cross-line directions can be gainfully applied to depth-migration methods based on downward-continuation. For these methods, splitting is applied at the level of each downward-continuation step. Chapter 5 discusses splitting methods for downward-continuation migration.

3.4.2 Two-pass migration of SEG-EAGE salt data set

The analysis of two-pass migration presented in this chapter, and the geometric interpretation of its inaccuracies, are illustrated by the following example obtained by time migrating the SEG-EAGE salt data set.

Figure 3.32 shows an in-line section of full 3-D prestack time migration of the first 1.8 km offsets of the salt data set. All the reflectors above the salt, including the dipping faults oriented at 45 degrees to the acquisition direction, are well imaged.

Figure 3.33 shows the result of 2-D prestack time migration along the in-line direction, for the same cross-line location as the section shown in Figure 3.32. The salt flank on the left of the section is well imaged because it is oriented along the in-line direction. However, the dipping faults oriented at 45 degrees to the acquisition direction are grossly mispositioned.

Figure 3.34 shows the result of 2-D zero-offset time migration along the cross-line direction of the cube obtained by 2-D prestack time migration along the in-line direction. In other words, Figure 3.34 shows the results of 3-D two-pass prestack migration of the data set, according to the two-pass migration theory developed in this section. The dipping faults oriented at 45 degrees to the acquisition direction are well imaged and positioned very close to the correct location. However, careful comparison of the section shown in Figure 3.34 (approximate) with the section shown in Figure 3.32 (exact) shows that the dipping fault is slightly overmigrated and thus mispositioned. The overmigration is caused by the effects of the increase in V_{rms} with depth, as predicted by the geometric interpretation above.

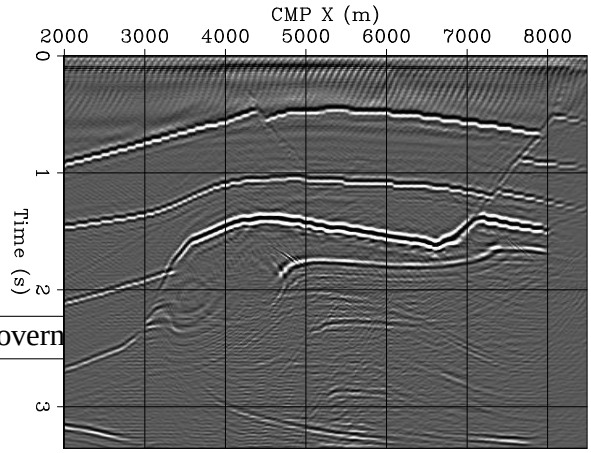


Figure 3.32: Full 3-D prestack time migration of the salt data set. All the reflectors above the salt, including the dipping faults oriented at 45 degrees to the acquisition direction, are well imaged. partial-Salt-inline-onepass-y5790-dense-overn [CR,M]

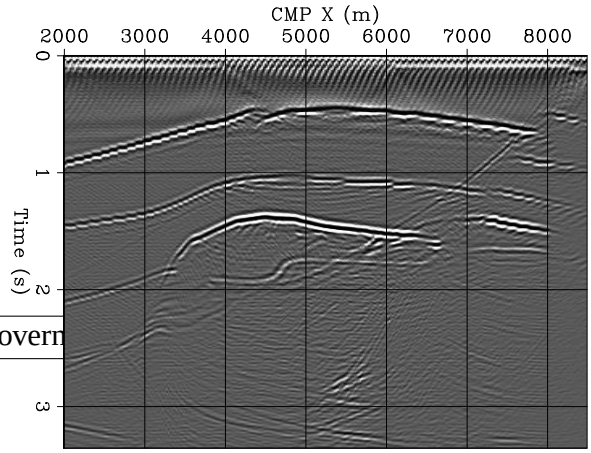


Figure 3.33: In-line 2-D prestack time migration, for the same cross-line location as the section shown in Figure 3.32. The dipping faults oriented at 45 degrees to the acquisition direction are grossly mispositioned. partial-Salt-inline-twopass-y5790-dense-overn [CR,M]

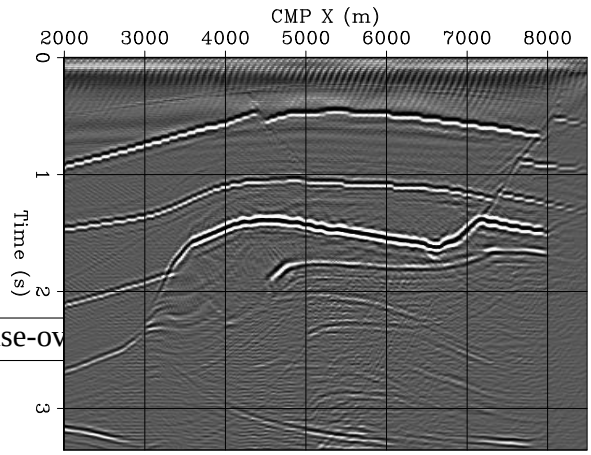


Figure 3.34: Two-pass 3-D prestack time migration, for the same cross-line location as the sections shown in Figures 3.32 and 3.33. Notice the slight overmigration of the dipping fault. partial-Salt-crossline-twopass-y5790-dense-ov [CR,M]

Figure 3.35 shows the time slice cut at 1.3 s through the image obtained by full 3-D prestack migration, corresponding to the migration result shown in Figure 3.32. Figure 3.36 shows the same time slice as in Figure 3.35, but corresponding to the 2-D in-line migration shown in Figure 3.33. Figure 3.37 shows the same time slice as in Figures 3.35 and 3.36, but corresponding to the two-pass 3-D migration results. Notice the gross mispositioning of the dipping fault in the 2-D in-line migration results and the slight overmigration in the two-pass 3-D migration results.

REFERENCES

Biondi, B., Fomel, S., and Chemingui, N., 1998, Azimuth moveout for 3-D prestack imaging: Geophysics, **63**, no. 2, 574–588.

Canning, A., and Gardner, G. H. F., 1992, Feathering correction for 3-D marine data: 62nd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 955–957.

Canning, A., and Gardner, G. H. F., 1993, Two-pass 3D pre-stack depth migration: 63rd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 892–894.

Deregowski, S. M., and Rocca, F., 1981, Geometrical optics and wave theory of constant offset sections in layered media: Geophys. Prosp., **29**, no. 3, 374–406.

Devaux, V., Gardner, G. H. F., and Rampersad, T., 1996, 3-D prestack depth migration by Kirchhoff operator splitting: Soc. Expl. Geophys., 66th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 455–458.

Dickinson, J. A., 1988, Evaluation of two-pass three-dimensional migration: Geophysics, **53**, no. 1, 32–49.

Fomel, S., and Biondi, B., 1995, The time and space formulation of azimuth moveout: 65th Ann. Internat. Meeting, Soc. Expl. Geophys., Expanded Abstracts, 1449–1452.

Forel, D., and Gardner, G. H. F., 1988, A three-dimensional perspective on two-dimensional dip moveout: Geophysics, **53**, no. 5, 604–610.

Gibson, B., Larner, K., and Levin, S., 1983, Efficient three-dimensional migration in two steps: Geophys. Prosp., **31**, no. 1, 1–33.

Hale, D., 1991, Course notes: Dip moveout processing: Soc. Expl. Geophys., Tulsa.

Hanson, D. W., and Witney, S. A., 1995, 3-D prestack depth migration – velocity model building and case history: 1995 Spring Symposium of the Geophys. Soc. of Tulsa, Soc. Expl. Geophys., Seismic Depth Estimation, 27–52.

Levin, F. K., 1971, Apparent Velocity from Dipping Interface Reflections: Geophysics, **36**, no. 3, 510–516.

Ronen, J., 1987, Wave equation trace interpolation: Geophysics, **52**, 973–984.

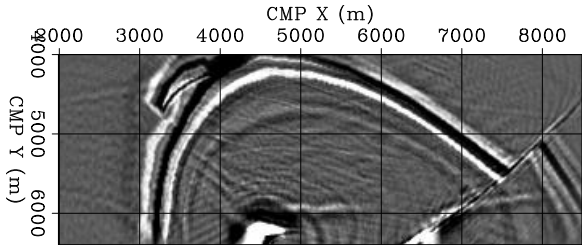


Figure 3.35: Time slice cut at 1.3 s through the image obtained by full 3-D prestack migration. This time slice corresponds to the full 3-D prestack migration shown in Figure 3.32. partial-Salt-onepass-t1.3-dense-overn [CR,M]

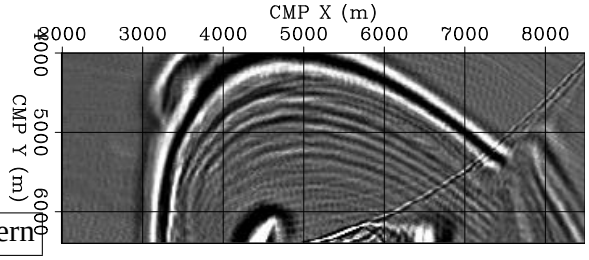


Figure 3.36: Time slice cut at 1.3 s through the image obtained by 2-D in-line migrations. This time slice corresponds to the section shown in Figure 3.33. Notice the gross mispositioning of the dipping fault. partial-Salt-inline-twopass-t1.3-dense-overn [CR,M]

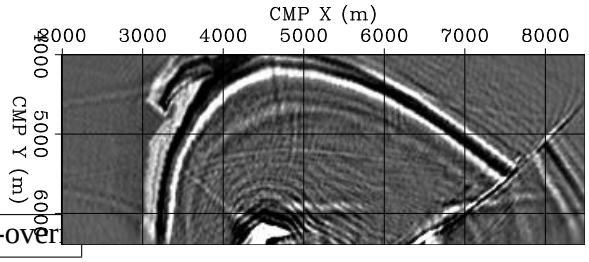


Figure 3.37: Time slice cut at 1.3 s through the image obtained by two-pass 3-D migration. This time slice corresponds to the section shown in Figure 3.34. Notice the slight overmigration of the dipping fault. partial-Salt-crossline-twopass-t1.3-dense-over [CR,M]

Chapter 4

Principles of wavefield-continuation migration

Migration methods based on the numerical propagation of the recorded wavefield can overcome some of the limitations of Kirchhoff migration methods. However, their computational cost, in particular for prestack imaging, can be substantially higher than the computational cost of Kirchhoff methods. They are currently used widely to image 3-D poststack data. Their use for imaging 3-D prestack data is becoming more popular, but it is still an active research area.

Wavefield-continuation methods can yield better images than Kirchhoff methods for depth-migration problems. They provide an accurate solution of the wave-equation over the whole range of seismic frequencies, whereas Kirchhoff methods are based on a high-frequency approximation of the wave equation. Furthermore, wavefield-continuation methods naturally handle multipathing of the reflected energy induced by complex velocity functions. In contrast, when multipathing occurs, Kirchhoff methods require the summation of the data over complex multivalued surfaces. This process can be cumbersome and prone to errors.

The following example illustrates a case in which the severe multipathing of the wavefield makes wavefield-continuation methods advisable. Figure 4.1 shows a cross-line section through the SEG-EAGE salt velocity model. Figure 4.2 compares the images obtained by 3-D prestack migration with a wavefield-continuation method (top) and a Kirchhoff method (bottom) using the SEG-EAGE C3-NA data set. The location of this section coincides with the section shown in Figure 4.1. The improvements achieved by the wavefield-continuation migration in the sub-salt image are evident. In the top image there are several reflectors that are not visible in the bottom image. Furthermore, several imaging artifacts that degrade the bottom image are not present in the top image. The wavefield-continuation image was computed by applying a computational scheme called common-azimuth migration (Biondi and Palacharla, 1996), which has a cost comparable to using Kirchhoff migration for full volume images. Common-azimuth migration is now routinely applied in production imaging projects.

The superior image quality achieved by wavefield-continuation migration can be understood by analyzing the results of wavefield modeling in the vertical section corresponding to the images shown in Figure 4.2. Figure 4.3 shows two snapshots of the wavefield taken at

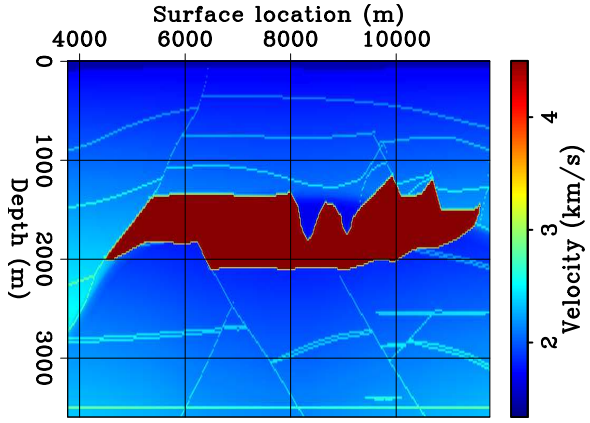


Figure 4.1: Cross-line section of the interval velocity model for the SEG-EAGE salt data set. wemig-Vel-SEG-SALT-WAVE-col [NR]

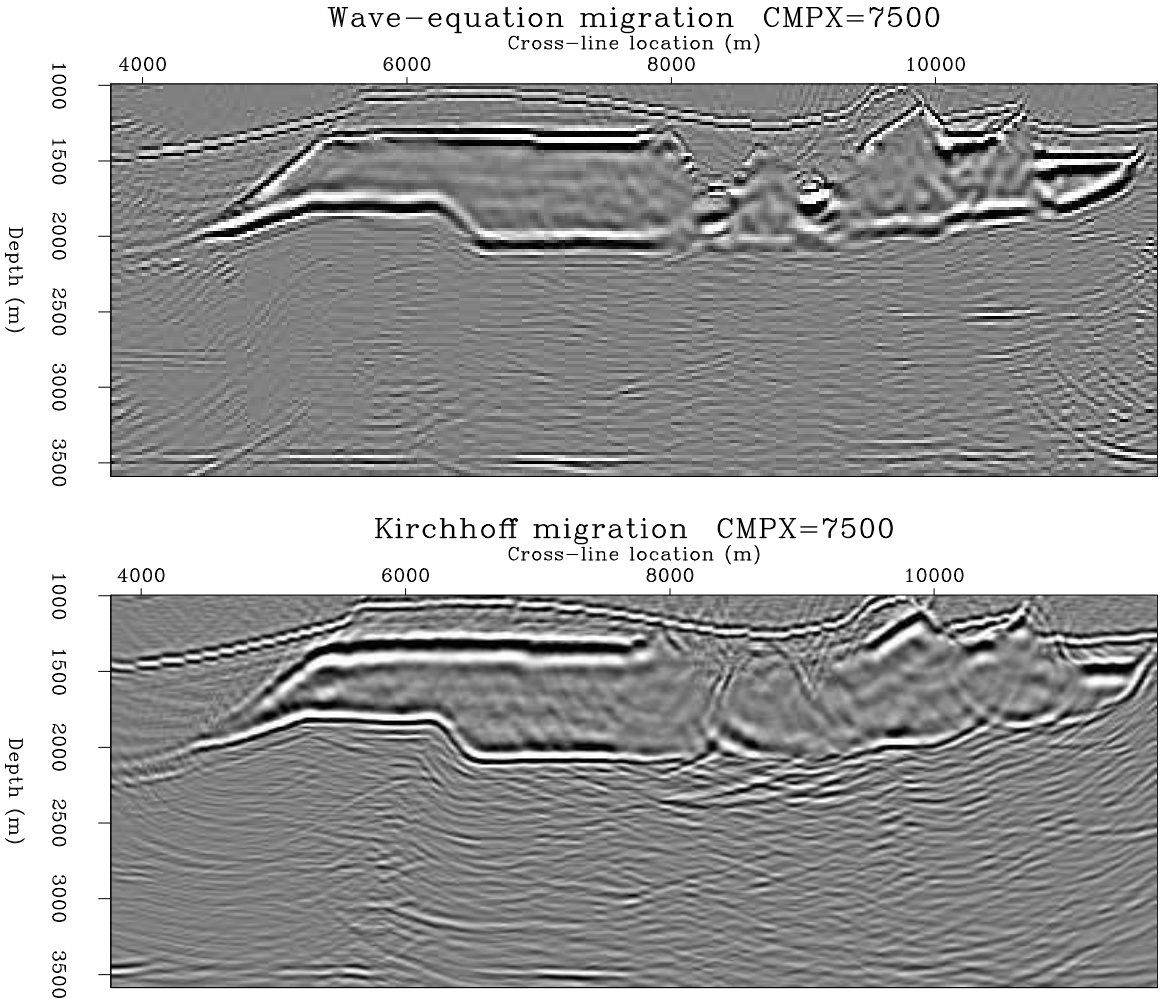


Figure 4.2: Sections of 3-D prestack migration results of a synthetic data set: wavefield-continuation migration (top), Kirchhoff migration (bottom). The location of this section coincides with the section shown in Figure 4.1. wemig-Both-salt-under-x7500-ok [NR]

times t=0 and t=1 second, when the source is located below the salt body. Figure 4.4 shows the wavefield recorded at the surface. Notice the complex multipathing of the wavefield and the multi-branching of the Green function. The line superimposed on the wavefield represents the time-delay function computed using a finite-difference solution of the Eikonal equation. In this case, the Eikonal solution is a poor approximation of the "true" Green function.

Full-volume imaging of zero-offset data is usually more efficient using wavefield-continuation methods rather than Kirchhoff methods. However, wavefield-continuation methods work best with regular and complete computational grids. Because prestack recording geometries are far from being complete and regular (Chapter 1), the gaps in the recorded data need to be filled with zero traces. Depending on the acquisition geometry, the number of zero traces that are needed may be enormous. The resulting increase in the size of the computational domain may make wavefield-continuation methods very expensive. Another important computational advantage of Kirchhoff methods is their relative efficiency in imaging only a subset of the whole image cube (**target-oriented migration**). For target-oriented migration, the computational cost of Kirchhoff migration is reduced in proportion to the reduction in image size. In contrast, the computational cost of target-oriented migration by wavefield-continuation methods is seldom substantially lower than the cost of full-volume migration.

A "conceptual" advantage of Kirchhoff migration is that is easier to visualize and understand. This characteristic is the main reason for introducing the concept of migration with Kirchhoff methods (Chapter 2). Ray tracing (which is at the basis of Kirchhoff methods) provides an intuitive representation of wave propagation. Furthermore, Kirchhoff-migration operators are multi-dimensional convolutional operators. The graphical analysis of the summation kernels of these convolutional operators provides a direct insight into their main properties and into the relation between the recorded data and the migrated image. In contrast, for wavefield-continuation methods the relation between image and data is indirect, because these methods are based on two consecutive but distinct steps: 1) numerical propagation of the recorded wavefield (and possibly of the source function), and 2) formation of the image by applying an imaging condition to the propagated wavefields.

Many methods have been proposed for imaging seismic data by wavefield continuation. All of them have the two main components of wavefield propagation and image formation, but they differ in how these two steps are carried out. The methods can be classified according to 1) the dimension along which the wavefields are continued (e.g depth or time), 2) the spatial dimension of the computational domain (source-receiver or shot gathers), and 3) the numerical method used to propagate the wavefields. Numerical analysis offers us several methods for solving the wave equation, each with its own domain of applicability and limitations. As usual, the main trade-off is between computational cost and accuracy. In Chapter 5, I discuss in detail the numerical methods for downward continuing wavefields.

The basic principles of wavefield-continuation migration are independent of the specific migration method. Indeed, as we will see at the end of this chapter, it can be shown that when appropriate conditions are fulfilled, wavefield-continuation methods that appear to be different actually produce the same image cube. We will first analyze the basic principles of wavefield-continuation migration in the case where the wavefields are propagated in the time domain.

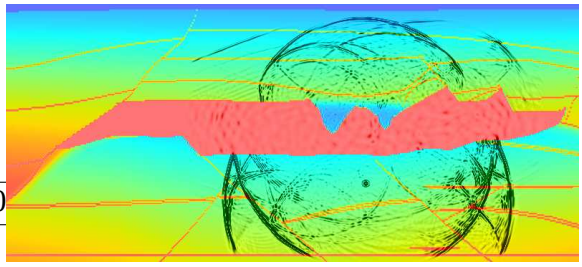


Figure 4.3: Snapshots of the wavefield at t=0 s and t=1 s, when the source is located below a salt body with a rugose top. wemig-Wave-Vel-SEG-SALT-WAVE-2-t0 [NR]

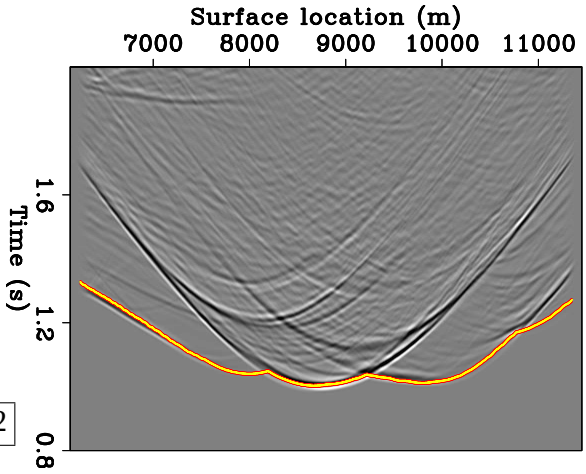


Figure 4.4: Wavefield recorded at the surface, corresponding to the wave modeling shown in Figure 4.3. The trajectory superimposed onto the wavefield represents the time-delay function computed by numerically solving the Eikonal equation. The Eikonal solution is a poor representation of the full-wavefield Green function. wemig-Model-Time-SEG-SALT-WAVE-2 [NR,M]

The family of migration methods based on time-domain propagation is commonly known as **reverse-time migration**.

4.1 Reverse-time migration

Reverse-time migration (Baysal et al., 1983; Whitmore, 1983) is probably the most intuitive wavefield-continuation migration method, though it is seldom applied in practice because it is also the most computationally expensive. The most straightforward way to introduce reverse-time migration is by defining the migration of the data collected from one single experiment; that is, a shot profile. This kind of migration is thus called **shot-gather migration** (Jacobs, 1982; Etgen, 1986). In shot-gather migration each profile is migrated independently. The image cubes obtained for all shots are then summed.

Two wavefields are independently propagated in shot-profile migration: 1) the receiver wavefield is propagated starting from the recorded data, and 2) the source wavefield is propagated starting from an assumed source wavelet. Both the source wavefield and the recorded wavefield are continued along the time axis; the source wavefield is propagated forward in time, whereas the recorded wavefield is propagated backward in time (this characteristic justifies the name “reverse-time” for this migration method). The image is formed by cross-correlating the two wavefields and evaluating the correlation at zero time (Claerbout, 1971).

If $P^s(t,z,x,y;\mathbf{s}_i)$ and $P^g(t,z,x,y;\mathbf{s}_i)$ are respectively the source wavefield and the recorded

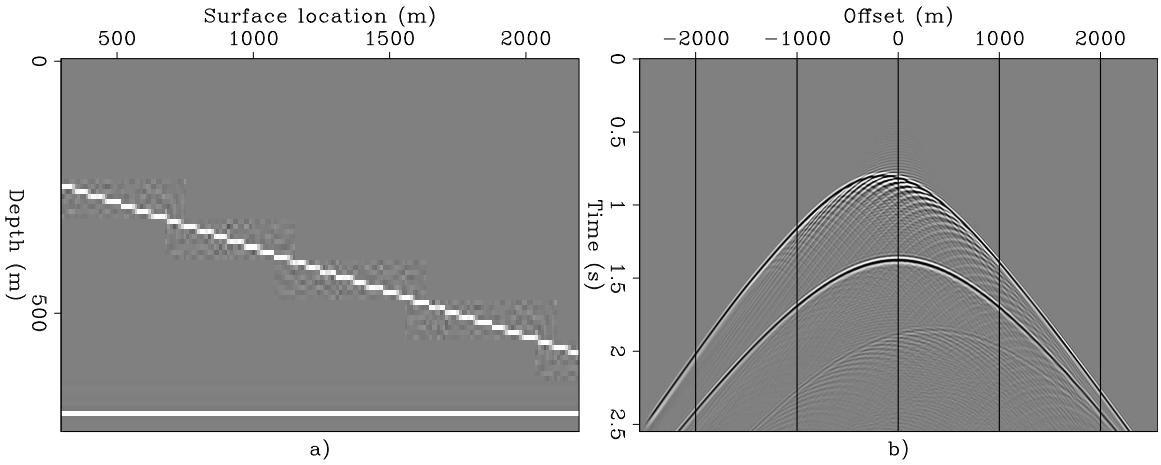


Figure 4.5: Velocity model (a) and shot profile (b) used for illustrating the reverse-time migration process. wemig-Vel-Shot-overn [NR]

wavefield for the i-th source location $\mathbf{s}_i$, as functions of time t and subsurface location (z,x,y), the image is formed by applying the following **imaging condition**:

$$I\left(z_\xi, x_\xi, y_\xi\right) = \sum_i \sum_t P^s\left(t, z=z_\xi, x=x_\xi, y=y_\xi; \mathbf{s}_i\right) P^g\left(t, z=z_\xi, x=x_\xi, y=y_\xi; \mathbf{s}_i\right). \tag{4.1}$$

This expression distinguishes between the image-space coordinates $\left(z_\xi, x_\xi, y_\xi\right)$ and the physical-space coordinates (z,x,y), though in practice the two spaces often coincide. In the following, for the sake of simplifying the notation, I distinguish between the two coordinate systems only when it is necessary to avoid ambiguities.

The application of the imaging condition (4.1) produces an image that has the correct kinematics (structural image), but not necessarily has the correct amplitudes. In this respect, it is similar to the Kirchhoff migration formula in equation (2.4), when the summation weights are ignored.

The following simple example illustrates the process of reverse-time migration, and provides a sense of how the imaging condition in equation (4.1) works. Figure 4.5a shows the velocity model used to generate the shot profile shown in Figure 4.5b. The direct arrival was removed from the shot profile to avoid artifacts during the migration. As expected, the reflection from the dipping event is asymmetric around zero-offset, and the definition of the dipping reflector as a staircase creates several diffractions along that reflection.

Figures 4.6–4.8 show three snapshots of the reverse-time migration process. The snapshots are shown in reverse-time order, to be consistent with the order with which the actual computations are performed on the data. The panels in the middle (panels b) show the receiver wavefield $[P^g(z,x)]$ as it is propagated backward using the recorded shot profile (Figure 4.5b) as boundary condition on the top of the model. The panels on the left (panels a) show the source wavefield $[P^s(z,x)]$ as it propagates through the model, and the panels on the right (panels c) show the image as it progresses from being empty when the source wavefield does

not interfere with the receiver wavefield (Figure 4.6c) to being fully formed after the two wavefields have "passed" each other (Figure 4.8c).

The migration velocity function used to generate these figures is the background velocity field (constant in this case); that is, it does not include the reflectors (which are represented by the white lines in Figure 4.5a). Including the reflectors – or any velocity discontinuity – in the velocity model would generate internal reflections during the propagation of both wavefields and thus cause artifacts in the image. This problem is illustrated in Figures 4.9–4.11 which were generated using the velocity field shown in Figure 4.5a, including the reflectors. When the source wavefields (panels a) and the receiver wavefields (panels b) shown in Figures 4.9–4.11 are compared with the corresponding wavefields in Figures 4.6–4.8, it is evident that adding the interfaces to the velocity model created internal reflections. The presence of these internal reflections when the two wavefields are cross-correlated creates the strong artifacts in the image that are evident in the right panels (panels c) of Figures 4.9–4.11.

A simple way to eliminate the internal reflections would be to smooth the velocity model. However this smoothing may adversely affect the kinematics of the propagating wavefields when the actual velocity contrast is strong (e.g. in salt bodies). Baysal et al. (1984) proposed a method to reduce the internal reflections by matching the acoustic impedance at the velocity discontinuities. Their methods is effective for zero-offset migration because it eliminates normal reflections, but it is not effective for prestack migration. Even when the computational cost is no longer an obstacle to the wide-spread adoption of reverse-time migration, the problem related to internal reflections will need to be solved. Cunha (1992) and Karrenbach (1991) present some ideas on how to address this problem in principle.

The imaging condition in equation (4.1) creates a "stacked" image cube. In many ways it is equivalent to the result of Kirchhoff migration when the images for all the data offsets are stacked together. However, for both velocity estimation and amplitude-versus-angle (AVA) analysis it is often useful to create a "prestack" image cube. The imaging condition in equation (4.1) can be generalized by introducing the **subsurface half offsets** x_{ξ_h} and y_{ξ_h}. The image is now obtained by correlating the source wavefield and the receiver wavefield shifted horizontally with respect to each other as follows:

$$I\left(z_\xi, x_\xi, y_\xi, x_{\xi_h}, y_{\xi_h}\right) = \sum_i \sum_t P^g\left(t, z_\xi, x_\xi + x_{\xi_h}, y_\xi + y_{\xi_h}; \mathbf{s}_i\right) P^s\left(t, z_\xi, x_\xi - x_{\xi_h}, y_\xi - y_{\xi_h}; \mathbf{s}_i\right). \tag{4.2}$$

Obviously, the imaging condition in equation (4.1) is a special case of the imaging condition in equation (4.2), where x_{ξ_h} and y_{ξ_h} are zero.

This generalization of the imaging condition is important and allows to extract both velocity and AVA information from the migrated cube. The physical interpretation of the subsurface offsets is not immediate, but it becomes clear when the subsurface offsets are equated to the data offset of the recorded data datumed at depth, as it is explained in Section 4.2 and demonstrated in Section 4.2.3.

The prestack image cube defined as a function of the subsurface offsets is seldom useful in itself. Chapter 6 presents methods to transform the prestack-image obtained by the imaging condition in equation (4.2) into another prestack-image cube function of the aperture angle

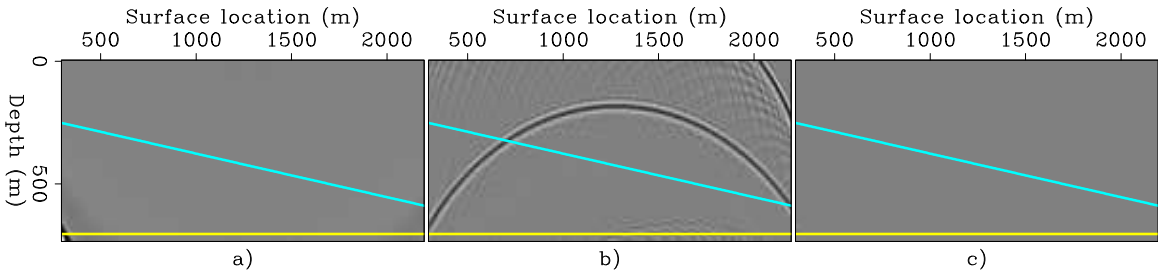


Figure 4.6: Reverse-time migration with constant (background) velocity function. Snapshots at t=1.20 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). Neither reflector has been imaged yet. wemig-Wave-diff-all-refl-1.20-overn [NR,M]

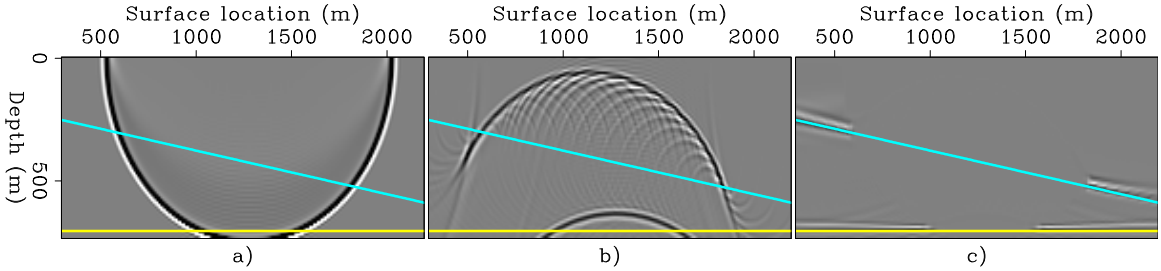


Figure 4.7: Reverse-time migration with constant (background) velocity function. Snapshots at t=.75 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). The bottom reflector is almost fully imaged, and the shallow reflector is only partially imaged. wemig-Wave-diff-all-refl-.75-overn [NR,M]

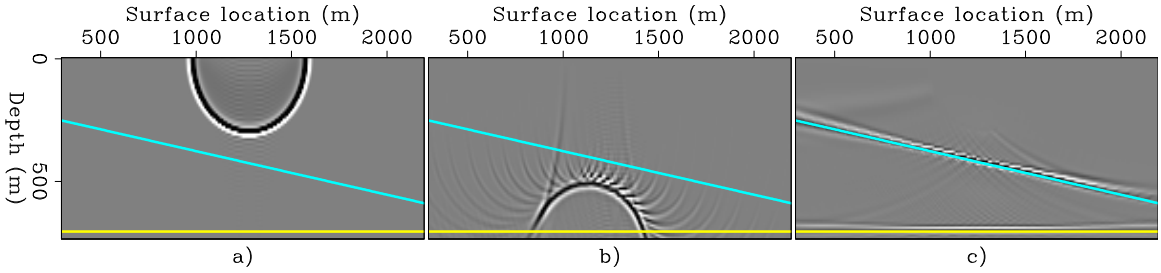


Figure 4.8: Reverse-time migration with constant (background) velocity function. Snapshots at t=.30 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). Both reflectors are fully imaged. wemig-Wave-diff-all-refl-.30-overn [NR]

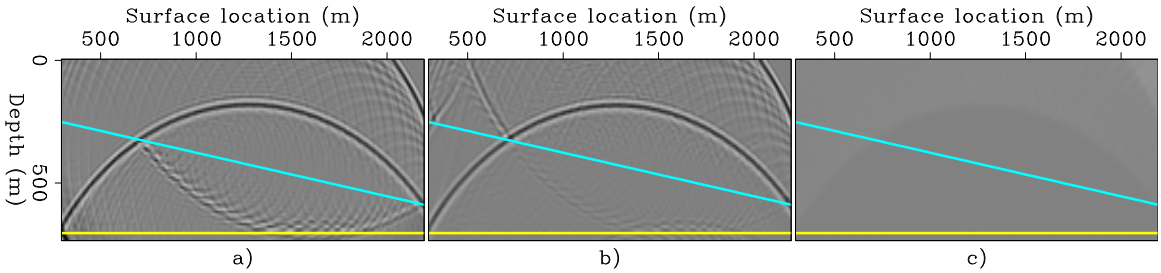


Figure 4.9: Reverse-time migration with true velocity function (including reflectors). Snapshots at t=1.20 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). In the source wavefield (panel a), notice the reflection from the flat interface. The cross-correlation between this reflection and the backward propagated receiver wavefield (panel b) creates a faint shadow in the image (panel c). wemig-Wave-nodir-all-refl-1.20-overn [NR,M]

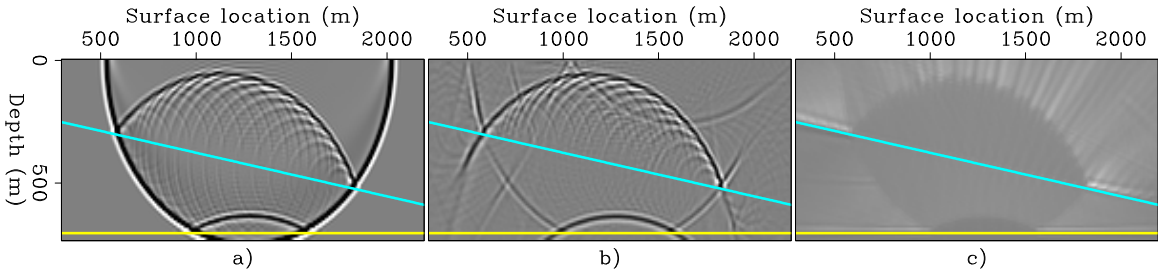


Figure 4.10: Reverse-time migration with true velocity function (including reflectors). Snapshots at t=.75 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). The reflection from the dipping interface in the source wavefield (panel a) creates additional artifacts in the image (panel c). wemig-Wave-nodir-all-refl-.75-overn [NR]

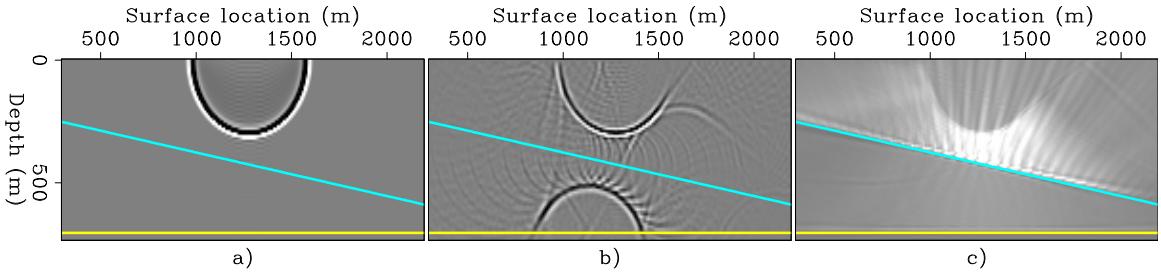


Figure 4.11: Reverse-time migration with true velocity function (including reflectors). Snapshots at t=.30 seconds of source wavefield (a), receiver wavefield (b), and image progression (c). In the receiver wavefield (panel b), the constructive interference of the reflections from the two interfaces creates a circular wavefront. This wavefront constructively correlates with the direct source wavefield (panel a) and causes additional artifacts in the image (panel c). wemig-Wave-nodir-all-refl-.30-overn [NR]

and azimuth at the reflection point, and thus making it valuable for both velocity estimation and AVA analysis purposes.

4.2 Downward-continuation migration

The principles of downward-continuation migration are similar to those of reverse-time migration. The main difference is that the wavefields are propagated along the depth axis instead of along the time axis.

Propagating the wavefields along the depth axis has two meaningful advantages: 1) the computations can be carried out in the temporal-frequency domain with substantial reduction of both computational and memory requirements, and 2) discontinuities in the velocity model almost never cause image artifacts discussed in the previous section (Section 4.1). Furthermore, downward-continuation leads to the definition of another useful family of migration methods, **source-receiver migration**. Source-receiver migration is based on the concept of **survey sinking** and thus is dependent on the downward-continuation of the recorded wavefield. Source-receiver migrations have different characteristics than shot-profile migrations, and have several advantages for marine data.

Downward-continuation methods are based on a recursive solution of the one-way wave equation, a detailed derivation of which is beyond the scope of this book. The reader interested in more background material is encouraged to read Claerbout's *Imaging the Earth's Interior* (1985). The basic continuation step used to compute the wavefield at depth $z + \Delta z$ from the wavefield at depth z can be expressed in the frequency-wavenumber domain as follows:

$$P_{z+\Delta z}\left(\omega, k_x, k_y\right) = P_z\left(\omega, k_x, k_y\right) e^{ik_z \Delta z}; \tag{4.3}$$

where ω is the temporal frequency, k_x and k_y are the horizontal wavenumbers, and k_z is the vertical wavenumber.

The vertical wavenumber can be expressed by the following **dispersion relation**, which is often called the Single Square Root (SSR) equation:

$$k_z = \mathrm{SSR}\left(\omega, k_x, k_y\right) = -\sqrt{\frac{\omega^2}{v\left(z,x,y\right)^2} - \left(k_x^2 + k_y^2\right)}, \tag{4.4}$$

where $v(z,x,y)$ is the propagation velocity. The minus sign in front of the square root means that we want to downward continue upward-propagating waves, in accordance with the sign convention defined by Claerbout (1985). In equation (4.4) there is an apparent conflict between the use of the horizontal wavenumbers k_x and k_y, and the expression of the propagation velocity v as a function of the horizontal coordinates x and y. Reconciling this conflict is actually the main challenge of the numerical solution of the downward-continuation equation.

Chapter 5 presents a detailed overview of several methods that have been proposed to solve equation (4.3). For the purposes of this chapter, the downward-continuation operator $e^{ik_z \Delta z}$ can be considered as a non-stationary convolutional operator, with the convolution performed

along at least one of the the horizontal axes. When necessary, I will specify the axes along which the convolution is performed by adding a subscript to the notation introduced in equation (4.3). For example, I use the notation $e^{ik_{z_s}\Delta z}$ to denote that the convolution is performed along the shot axes. Furthermore, I will denote with $e^{ik_z z}$ the convolutional operator that propagates a wavefield from the surface all the way down to depth z. This operator is given by the cascade of all the $e^{ik_z\Delta z}$ for all the depth steps Δz necessary to go from the surface to z.

4.2.1 Shot-gather migration by downward continuation

Shot-gather migration by downward continuation is performed in a fashion similar to shot-gather migration by time-domain propagation. The receiver wavefield at depth P_z^g is computed by downward continuing the recorded data $P_{z=0}^g$ as follows:

$$P_z^g = P_{z=0}^g e^{ik_z z}. \tag{4.5}$$

The source wavefield at depth P_z^s is computed by downward continuing the assumed source wavelet $P_{z=0}^s$:

$$P_z^s = P_{z=0}^s e^{-ik_z z}. \tag{4.6}$$

The negative sign in front of the exponential in equation (4.6) is necessary because the source wavefield propagates downward, as opposed to propagating upward as in the receiver wavefield.

The image can be formed by cross-correlating the two wavefields along the time axis (multiplication by the complex conjugate in the frequency domain) and evaluating the correlation at zero time (summation over frequencies):

$$\begin{aligned} &I\left(z_\xi, x_\xi, y_\xi, x_{\xi_h}, y_{\xi_h}\right) \\ &\quad = \sum_i \sum_\omega P_z^g\left(\omega, x_\xi + x_{\xi_h}, y_\xi + y_{\xi_h}; \mathbf{s}_i\right) \overline{P_z^s\left(\omega, x_\xi - x_{\xi_h}, y_\xi - y_{\xi_h}; \mathbf{s}_i\right)}. \end{aligned} \tag{4.7}$$

4.2.2 Source-receiver migration

Source-receiver migration is based on the concept of survey sinking. After each depth propagation step, the propagated wavefield is equivalent to the data that would have been recorded if all sources and receivers were placed at the new depth level. This task is accomplished by downward continuing all the source and receiver gathers at each depth step. (Schultz and Sherwood, 1980). Therefore, the basic downward continuation is performed by applying the Double Square Root (DSR) equation,

$$k_z = \mathrm{DSR}\left(\omega, \mathbf{k_s}, \mathbf{k_g}\right) = k_{z_\mathbf{s}} + k_{z_\mathbf{g}} = -\sqrt{\frac{\omega^2}{v^2(\mathbf{s}, z)} - \mathbf{k_s}^2} - \sqrt{\frac{\omega^2}{v^2(\mathbf{g}, z)} - \mathbf{k_g}^2}, \tag{4.8}$$

where $\mathbf{k_s}$ and $\mathbf{k_g}$ are the wavenumbers along the source and receiver axes respectively, and where $v(\mathbf{s}, z)$ and $v(\mathbf{g}, z)$ are the propagation velocities at the source and receiver locations.

The first square root in the DSR downward-continues the source wavefield, whereas the second square root in the DSR downward-continues the receiver wavefield.

It is often convenient to express the double square root equation as a function of the midpoint and offset coordinates, in place of the source and receiver coordinates. In this case equation (4.8) becomes

$$\begin{aligned} k_z &= \mathrm{DSR}(\omega, \mathbf{k_m}, \mathbf{k_h}) \\ &= -\sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}(\mathbf{k_m}-\mathbf{k_h})\cdot(\mathbf{k_m}-\mathbf{k_h})} - \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}(\mathbf{k_m}+\mathbf{k_h})\cdot(\mathbf{k_m}+\mathbf{k_h})}, \end{aligned} \tag{4.9}$$

where $\mathbf{k_m}$ and $\mathbf{k_h}$ are the midpoint wavenumber and the offset wavenumber, respectively.

Notice that the structure of the DSR equation is similar to that of equation (2.3) for the summation surfaces of prestack migration. However, the velocities in the DSR are interval velocities, not the average velocities of equation (2.3).

Applying the DSR operator sinks the whole survey $P_z(\omega, x_m, y_m, x_h, y_h)$ together at increasing depth levels; that is:

$$P_{z+\Delta z}(\omega, x_m, y_m, x_h, y_h) = P_z(\omega, x_m, y_m, x_h, y_h)\, e^{ik_{z_\mathbf{s}}\Delta z} e^{ik_{z_\mathbf{g}}\Delta z}. \tag{4.10}$$

The image is extracted from the wavefield by applying an imaging condition at each depth level (Claerbout, 1985); that is, by evaluating the wavefield at zero time (i.e. summation over frequencies) and zero offset as follows:

$$I\left(z_\xi, x_\xi, y_\xi\right) = \sum_\omega P_z\left(\omega, x_m = x_\xi, y_m = y_\xi, x_h = 0, y_h = 0\right). \tag{4.11}$$

The intuitive explanation for applying the imaging condition in equation (4.11) is that a reflector located at the same depth level as the source and receivers would generate a reflection recorded at zero offset and zero time. Another intuitive justification of this imaging condition is that the wavefield recorded at the surface was generated by the reflectors exploding at zero time (Claerbout, 1985).

As in the case of shot-profile migration, the imaging condition for source-receiver migration can be generalized to image also the subsurface offsets as follows:

$$I\left(z_\xi, x_\xi, y_\xi, x_{\xi_h}, y_{\xi_h}\right) = \sum_\omega P_z\left(\omega, x_m = x_\xi, y_m = y_\xi, x_h = x_{\xi_h}, y_h = y_{\xi_h}\right). \tag{4.12}$$

With source-receiver migration the subsurface offsets (x_{ξ_h} and y_{ξ_h}) coincide with the data offsets of the downward-continued survey. This interpretation of the subsurface offsets in the image as the data offsets of a survey datumed using survey sinking is important; it allows the direct physical interpretation of the prestack images obtained using the imaging condition in equation (4.12). Furthermore, because of the equivalence between the prestack images

obtained by shot-profile migration and by source-receiver migration (demonstrated in Section 4.2.3), it leads to the physical interpretation of the prestack images obtained using the imaging conditions in equation (4.2) and equation (4.7).

When using source-receiver migration, the image is immediately available as a function of the subsurface offsets. In contrast, obtaining the prestack-image cube by shot-profile migration requires the additional wavefield correlations defined in equation (4.7). The capability of generating the prestack image cube without computing additional correlations between wavefields is a cost advantage of source-receiver migration over shot-profile migration. In 3-D, this cost saving can be substantial, because the offset space is a plane.

Zero-offset migration

The equation for zero-offset downward continuation can be derived from the prestack equation (4.9) by setting the offset wavenumbers k_{x_h} and k_{y_h} to zero, and $\mathbf{m} = \mathbf{s} = \mathbf{g}$ (i.e. by setting the offset to zero). With these substitutions, the DSR equation simplifies into an SSR equation as follows:

$$k_z = \mathrm{SSR}(\omega, \mathbf{k}) = -\sqrt{\frac{4\omega^2}{v^2(\mathbf{m}, z)} - \mathbf{k_m} \cdot \mathbf{k_m}}. \tag{4.13}$$

This equation is similar to the SSR equation (4.4), with the important difference that now the migration velocity is one half of the true interval velocity.

When zero-offset data are migrated, the imaging condition of equation (4.11) then obviously simplifies to

$$I\left(z_\xi, x_\xi, y_\xi\right) = \sum_\omega P_z\left(\omega, x_m = x_\xi, y_m = y_\xi\right). \tag{4.14}$$

The reader should be aware that setting the offset wavenumbers to zero in equation (4.9) is not equivalent to evaluating the wavefield at zero offset! The justification for this apparently incorrect substitution is based on a stationary-phase approximation of the DSR, which is similar to the stationary-phase approximation used to derive common-azimuth migration in Chapter 7. Setting the offset wavenumbers to zero is equivalent to assuming that for all zero-offset events the source raypath coincides with the receiver raypath. Therefore, strictly speaking, migration of zero-offset data by use of the SSR operator is a common-angle migration at normal-incidence instead of zero-offset migration (see Section 6.1.2 for an introduction to common-angle migration). While it is true that all normal-incidence reflections are recorded at zero offset, not necessarily all the events recorded at zero-offset are normal-incidence reflections. There are interesting situations when either lateral velocity anomalies and/or multiple reflections cause this assumption to break down (Claerbout, 1985).

Figures 4.12 and 4.13 show an example of reflections that break the assumption that the up-going path coincides with the down-going path for all zero-offset events. Figure 4.12 shows the raypath corresponding to an event reflected by both walls of a deep valley etched on the top of a salt body. This particular kind of multiple reflections are often called **prismatic reflections** (Broto and Lailly, 2001). Prismatic reflections potentially carry useful information on

the geometry of the salt body and on the velocity in the sediments. However, this information is seldom used because of the difficulties in imaging prismatic reflections, though they are commonly recorded when the top-of-salt surface is rugged. Figure 4.13 show the zero-offset section (a) and common-midpoint gather (b) extracted from a synthetic survey modeled assuming the velocity function shown in Figure 4.12. Because of reciprocity, the common midpoint gather (Figure 4.13b) is symmetric around zero offset. However, the prismatic events cross the zero-offset line with a non-zero time dip; i.e. with $\mathbf{k_h} \neq 0$. Therefore, the prismatic reflections would not be properly imaged if equation (4.13) were used to image the zero-offset section (Figure 4.13a).

4.2.3 Equivalence of source-receiver migration and shot-profile migration

At first glance, shot-profile migration and source-receiver migration seem to be substantially different algorithms. The basic principles used by the two schemes are different. Shot-profile migration is performed by *independently* propagating the source wavefield and the receiver wavefield. The image is obtained by cross-correlating the two wavefields. Source-receiver migration is based on the concept of survey sinking, by which we recursively synthesize equivalent data sets at increasing depth. At each depth step, imaging is performed by extracting the wavefield at zero time.

In this section I demonstrate that the two migration methods produce exactly the same image cube, following the demonstration presented in Biondi (2003); this demonstration is equivalent to the one presented by Wapenaar and Berkhout (1987). This result implies that both the quality of the migrated image and the accuracy of the velocity updating information (Chapter 6) are independent of the choice between source-receiver migration and shot-profile migration.

Notwithstanding their theoretical equivalence, the efficiency of 3-D shot-profile migration can be substantially different from the efficiency of 3-D source-receiver migration. The relative advantages of each method strongly depend on the acquisition geometry. In general, source-receiver migration is more attractive for marine data that have limited azimuthal range (Biondi and Palacharla, 1996), while shot-profile migration is better suited for land or Ocean Bottom Cable (OBC) acquisition geometries. Jeannot (1988) discussed some other practical aspects and the relative advantages of source-receiver and shot-profile migration for 2-D data. Most of his observations are also valid for 3-D data.

I demonstrate the identity of the two methods when shot-profile migration satisfies three specific requirements: 1) the source function is an impulse at zero time and a delta function in space, 2) the imaging condition is the cross-correlation of the source wavefield by the receiver wavefield, and 3) the source and receiver wavefields are propagated by downward continuation (i.e. not by reverse-time propagation). Another obvious assumption is that the same numerical algorithm is employed to downward continue the wavefields for both migration methods.

The equivalence of the two migration methods is based on the linearity of migration and on the commutative properties of the downward-continuation operator. For simplicity, I demonstrate the equivalence of the two migration methods by showing that the images obtained by

migrating the traces recorded in a single shot profile are the same. The linearity of both migrations with respect to the input wavefield makes the extension to the full data set obvious.

To create the whole prestack wavefield from the traces recorded in a single shot profile, we add the live traces to a cube of zero traces. The resulting data cube is equal to the outer product of two functions: the first represents the recorded data $P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})$ and is independent of the source-coordinate $\mathbf{s}$. The second function is a delta function that is centered at $\bar{\mathbf{s}}$ and it is independent of the receiver-coordinate $\mathbf{g}$. The recorded data for the whole survey can thus be expressed as

$$P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})\,\delta(\mathbf{s}-\bar{\mathbf{s}}). \tag{4.15}$$

Because of the commutative properties of the downward-continuation operator demonstrated in Biondi (2003), the survey downward continued at depth z using the DSR operator is expressed as

$$P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})\,\delta(\mathbf{s}-\bar{\mathbf{s}})\,e^{ik_{z_\mathbf{s}}z}e^{ik_{z_\mathbf{g}}z}. \tag{4.16}$$

The recorded data $P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})$ is independent of the source location, and thus it can be pulled out of the convolution along the source axes. Equation (4.16) then becomes

$$\left[P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})\,e^{ik_{z_\mathbf{g}}z}\right]\left[\delta(\mathbf{s}-\bar{\mathbf{s}})\,e^{ik_{z_\mathbf{s}}z}\right]=\left[P^g_{z=0}(\omega,\mathbf{g};\bar{\mathbf{s}})\,e^{ik_{z_\mathbf{g}}z}\right]\left[\overline{\delta(\mathbf{s}-\bar{\mathbf{s}})\,e^{-ik_{z_\mathbf{s}}z}}\right]. \tag{4.17}$$

On the right-hand side of equation (4.17), the first square bracket is the receiver wavefield downward continued at depth z, and the second square bracket is the complex conjugate of the source wavefield downward continued at depth z. These two terms are thus the same as those that enter in the imaging condition for shot-profile migration, as expressed in equation (4.7). Therefore, the prestack-image cube obtained by source-receiver migration is exactly the same as the prestack-image cube obtained by shot-profile migration. Notice the importance of the negative sign in front of the exponential in equation (4.6) for deriving the result in equation (4.17).

Illustration of equivalence on synthetic data set

To illustrate the theoretical result reached in the previous section, I migrated one shot record from a synthetic data set that was modeled over a medium with strong lateral velocity variations. These velocity variations cause coherent artifacts ("ghost reflectors") in the image produced by migrating a single shot. The test supports the theoretical result that in the source-receiver migration the downward continuations along the shot axis and along the receiver axis do indeed commute in the presence of strong lateral variations. It also confirms that the migration artifacts produced by source-receiver migration and shot-profile migration are the same. Figure 4.14 shows the velocity function assumed to model this synthetic data set. The reflector has a simple flat geometry, but a strong velocity anomaly above it creates severe multipathing that challenges different migration schemes. Figure 4.15 shows the shot profile used for the test; the source location is at -0.5 kilometers. Notice the complex multipathing in the recorded wavefield.

Figure 4.16 shows the zero subsurface offset images (equivalent to the stacked images produced by Kirchhoff migration) produced by the two migration methods. The panel on the left shows the image produced by shot-profile migration, and the panel on the right shows the image produced by source-receiver migration. The two images are almost identical, except for small differences in artifacts. The source-receiver migration operated on the data sorted in midpoint-offset coordinates, and thus the boundary conditions are different for the two algorithms. This difference is expected to cause small differences in migration artifacts between the two methods. Not only is the flat reflector the same in the two images, but so are the strong ghost reflectors visible between the surface locations of 0 and 1 kilometers, which are caused by the triplication of the wavepath. The maximum amplitude of the differences between the two images is about 4% of the maximum amplitude of the images.

Figure 4.17 compares two slices of the migrated cubes at constant surface location $x_\xi = 0.1$ kilometers (called common-image gathers as described in Chapter 6). Again the images are identical for both the "true" reflector and the ghost reflectors.

REFERENCES

Baysal, E., Kosloff, D. D., and Sherwood, J. W. C., 1983, Reverse time migration: Geophysics, **48**, no. 11, 1514–1524.

Baysal, E., Kosloff, D. D., and Sherwood, J. W. C., 1984, A two-way nonreflecting wave equation: Geophysics, **49**, no. 02, 132–141.

Biondi, B., and Palacharla, G., 1996, 3-D prestack migration of common-azimuth data: Geophysics, **61**, 1822–1832.

Biondi, B., 2003, Equivalence of source-receiver migration and shot-profile migration: Geophysics, **68**, 1340–1347.

Broto, K., and Lailly, P., 2001, Towards the tomographic inversion of prismatic reflections: 71st Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 726–729.

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

Claerbout, J. F., 1985, Imaging the Earth's Interior: Blackwell Scientific Publications.

Cunha, C. A., 1992, Elastic modeling and migration in earth models: Ph.D. thesis, Stanford University.

Etgen, J., 1986, Prestack reverse time migration of shot profiles: SEP–**50**, 151–170, `http://sep.stanford.edu/research/reports`.

Jacobs, B., 1982, The prestack migration of profiles: Ph.D. thesis, Stanford University.

Jeannot, J. P., 1988, Full prestack versus shot record migration: Practical aspects: 58th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, Session:S12.2.

Karrenbach, M., 1991, Prestack reverse-time migration in anisotropic media: SEP–**70**, 132–144, http://sep.stanford.edu/research/reports.

Schultz, P. S., and Sherwood, J. W. C., 1980, Depth migration before stack: Geophysics, **45**, 376–393.

Wapenaar, C. P. A., and Berkhout, A. J., 1987, Full prestack versus shot record migration: 69th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, Session:S15.7.

Whitmore, N. D., 1983, Iterative depth migration by backward time propagation: 53rd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, Session:S10.1.

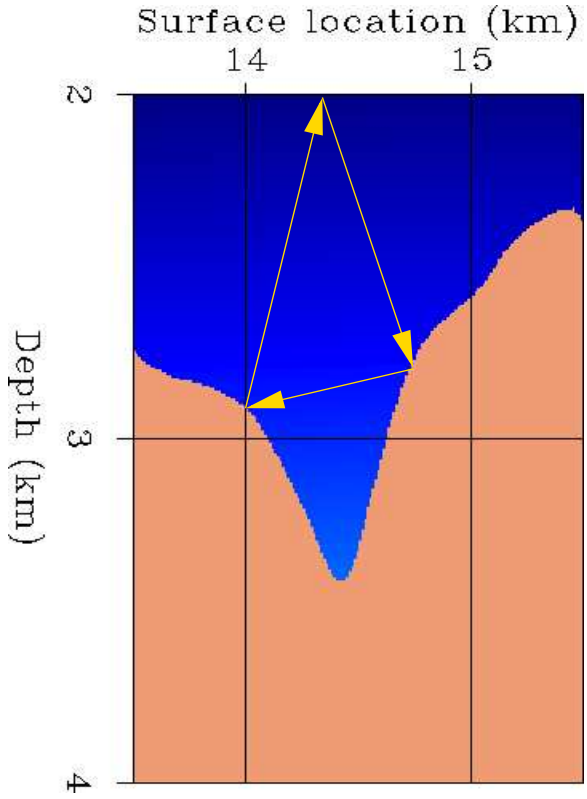


Figure 4.12: Sketch of a raypath corresponding to a prismatic reflection recorded at zero offset at the surface. These kind of reflections are a particular kind of multiple reflections, and they are recorded fairly frequently over salt bodies with rugged top. wemig-vel-prism-stretch-ann [NR]

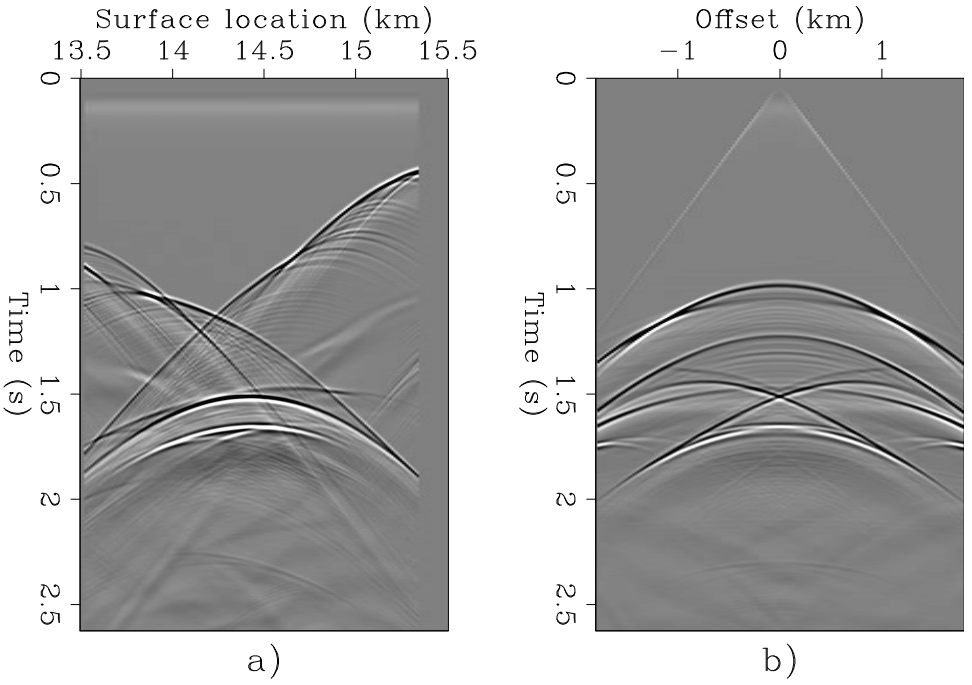


Figure 4.13: Zero-offset section (a) and common-midpoint gather (b) extracted from a synthetic survey modeled assuming the velocity function shown in Figure 4.12. The common-midpoint gather (panel b) is symmetric around zero offset because of reciprocity, but the prismatic reflections cross the zero-offset line with a non-zero time dip. wemig-zoff-cmp-prism-overy [NR]

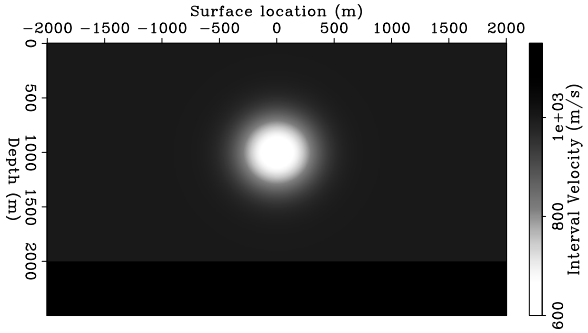


Figure 4.14: Velocity function assumed to model the synthetic data set used for the tests. wemig-Vel-trip [NR]

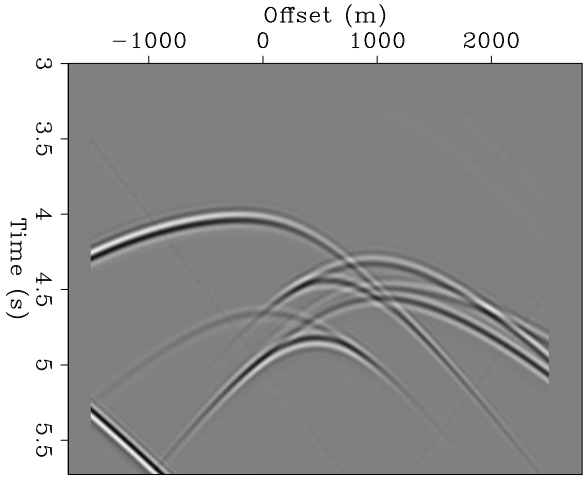


Figure 4.15: Shot profile used for the tests. The source location is at -0.5 kilometers. wemig-Shot-trip [NR]

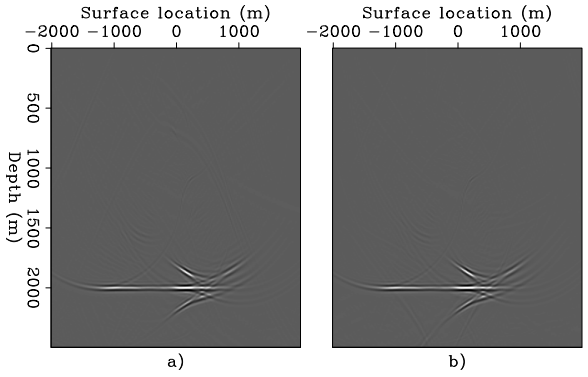


Figure 4.16: Zero subsurface offset sections of the migrated cubes obtained using: a) shot-profile migration, b) source-receiver migration. wemig-Mig-trip-both [NR]

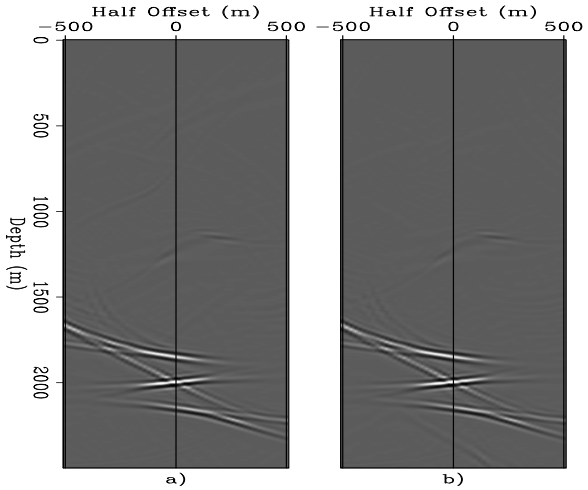


Figure 4.17: Offset-domain common-image gathers obtained by slicing the migrated cubes at the surface location of 0.1 kilometers: a) shot-profile image, b) source-receiver image. wemig-Cig-trip-100-both [NR]

Chapter 5

Downward-continuation methods

The numerical solution of the one-way wave equation is the cornerstone of all downward-continuation migration methods. Over the years a wide variety of solutions have been proposed based on approximations of the SSR operator introduced in Chapter 4. There is no absolute optimum among these methods, since the particular problem at hand determines the ideal balance of accuracy, computational cost, flexibility, and robustness in the selection of a downward-continuation method. This chapter provides a fairly complete overview of the numerical methods that have been developed for downward-continuing wavefields.

Downward-continuation methods can be classified according to the computational domain in which the SSR operator is applied. The most efficient, but also the least flexible, are the migrations performing the computations in the **frequency-wavenumber domain (ω-k)**. Those purely in the **frequency-space domain (ω-x)** are at the other extreme of the spectrum; that is, they are the most flexible but tend to be also the most computationally intensive. Between these extremes, there are the **mixed-domain (ω-k/ω-x)** methods. We start our discussion with the ω-k methods, and then quickly move on to the more useful ω-k/ω-x methods.

5.1 Frequency-wavenumber (ω-k) domain methods

When the propagation velocity is assumed to be a function only of depth, wavefields can be downward-continued efficiently in the frequency-wavenumber (ω-k) domain by simple multiplication with the SSR downward-continuation operator introduced in Section 4.2:

$$P_{z+\Delta z}(\omega,\mathbf{k}) = P_z(\omega,\mathbf{k})\, e^{i\Delta z\sqrt{\frac{\omega^2}{v(z)^2}-|\mathbf{k}|^2}}, \tag{5.1}$$

where ω is the temporal frequency, $\mathbf{k}$ is the vector of the horizontal wavenumbers, and $v(z)$ is the depth-dependent velocity.

The downward-continuation operator expressed in equation (5.1) describes a dip-dependent ($\mathbf{k}/\omega$) phase shift of the wavefield, and thus it is often called **phase-shift operator** (Gazdag,

1978). When the medium velocity is strictly a function of depth, the application of equation (5.1) downward-continues the data without any approximation of the kinematics of waves propagating along one direction with respect to the vertical axis. (Depending on the sign of the exponential the waves can propagate either upward or downward.) However, the effects of velocity variations along the depth direction on geometrical spreading are not correctly taken into account by the phase-shift operator, and consequently the amplitudes of the downward-continued wavefield are not correct.

The remainder of this chapter presents a wide-range of methods to solve the SSR equation when lateral velocity variations are not negligible, and thus the accuracy of the phase-shift operator is unacceptable. To adapt the phase-shift operator to lateral velocity variations it is necessary to perform at least part of the computation in the space domain. Several methods perform part of the computation in the wavenumber domain and part in the space domain; these methods can be categorized as mixed ω-k/ω-x-domain methods.

5.2 Mixed frequency-wavenumber/space (ω-k/ω-x) methods

Gazdag and Sguazzero (1984) proposed the first method to generalize the phase-shift method to lateral velocity variations. Because their method is based on phase shift and interpolation, they called it **Phase Shift Plus Interpolation (PSPI) migration**. Their method can be summarized as follows: 1) propagate the wavefield at each depth step by phase shift using several (two or more) constant velocities (**reference velocities**), 2) transform the wavefields obtained by these constant-velocity continuations into the space domain, and 3) interpolate between the wavefields according to the differences between the medium velocity and the reference velocities. PSPI is still a popular method because of its conceptual simplicity and its flexibility. Up to a limit, the accuracy of PSPI downward-continuation can be increased at will by simply increasing the number of reference velocities.

I will not discuss PSPI migration in any further detail, but in the next section I present **Split-step migration** that is closely related to PSPI.

5.2.1 Split-step migration

Split-step migration (Stoffa et al., 1990) retains PSPI's characteristics of simplicity and flexibility but has the conceptual advantage that it is easily related to the more accurate methods that are introduced in Section 5.2.2. In light of the theory introduced in Section 5.2.2, the application of a split-step correction can be interpreted as the application of of a first-order (with respect to velocity perturbations) correction operator to the phase-shift operator, whereas the more accurate methods presented in Section 5.2.2 apply a higher-order correction operator.

Split-step downward continuation is based on the following approximation of the SSR equa-

tion:

$$k_z = \mathrm{SSR}(\omega, \mathbf{k}) \approx \underbrace{\left(\sqrt{\frac{\omega^2}{v_{\mathrm{ref}}^2} - |\mathbf{k}|^2}\right)}_{\omega-\mathrm{k\ domain}} + \underbrace{\left(\frac{\omega}{v(z,x,y)} - \frac{\omega}{v_{\mathrm{ref}}}\right)}_{\omega-\mathrm{x\ domain}}, \tag{5.2}$$

where v_{ref} is called the reference velocity. The reference velocity is usually set equal to the average velocity in the current extrapolation layer. As indicated by the underlining braces, the operator implementing the first term in expression (5.2) is applied in the ω-k domain, whereas the second term represents a spatially varying time-shift that can be easily applied in the ω-x domain.

Figure 5.1 shows the flow-chart of the inner kernel of the split-step algorithm. At every depth step, the wavefield is first transformed into the wavenumber domain by a forward FFT. The transformed wavefield is then downward-propagated with the SSR operator, assuming the velocity to be constant and equal to the reference velocity. The subsequent, spatially varying time-shift, corrects, at least to first order, for the discrepancies between the reference velocity and the actual medium velocity.

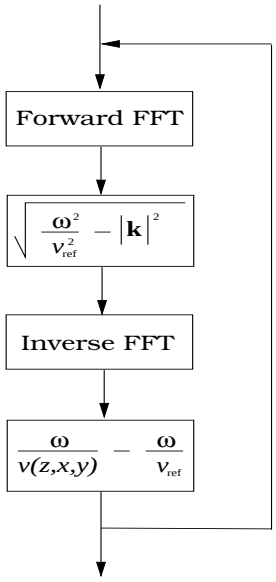


Figure 5.1: Flow-chart of the inner kernel of the split-step algorithm. At every depth step, the wavefield is first transformed into the wavenumber domain by a forward FFT. The transformed wavefield is then downward propagated with the SSR operator, assuming the constant velocity v_{ref}. The following spatially varying time-shift corrects, at least to first order, for the discrepancies between the reference velocity v_{ref} and the actual medium velocity $v(\mathbf{m}, z)$. down-split-cons [NR]

Performing the downward propagation partially in the wavenumber domain and partially in the space domain requires the computation of at least a forward and an inverse Fourier transform at each depth step. Because the cost of an FFT of length N grows as $N \log_2 N$, the cost function of split-step migration is equal to

$$\mathrm{ZoffDwn}_{(\omega-K/\omega-X)} \propto \kappa_{\mathrm{ZoffDwn}_{(\omega-K/\omega-X)}} \times \left(\mathrm{Nz}_\xi \times \mathrm{Nx}_\xi \times \mathrm{Ny}_\xi \times \log_2 \mathrm{Nx}_\xi \times \log_2 \mathrm{Ny}_\xi\right) \times (\mathrm{Nt}). \tag{5.3}$$

This cost function is quite attractive; typically Nx_ξ and Ny_ξ are of the order of the thousands, and thus $\log_2 \mathrm{Nx}_\xi$ and $\log_2 \mathrm{Ny}_\xi$ are approximately equal to 10.

The split-step approximation in equation (5.2) is exact when either the velocity is constant and equal to the reference velocity, or the midpoint wavenumber is equal to zero. The steeper

the reflections and the stronger the lateral variations in velocity, the greater is the error introduced by the split-step operator. Section 5.2.2 shows examples of impulse responses that illustrate this dependency of the errors from the reflectors dips and the velocity contrast.

The accuracy of split-step migration can be improved by using more than one reference velocity. In this modified scheme, often called **Extended Split-Step migration** (Kessinger, 1992), as many reference wavefields are generated as reference velocities are used. A single wavefield is then estimated at each depth step by an interpolation in the space domain of the reference wavefields corresponding to all reference velocities. At each spatial location, the interpolation weights are computed according to the difference between the actual medium velocity and the respective reference velocity.

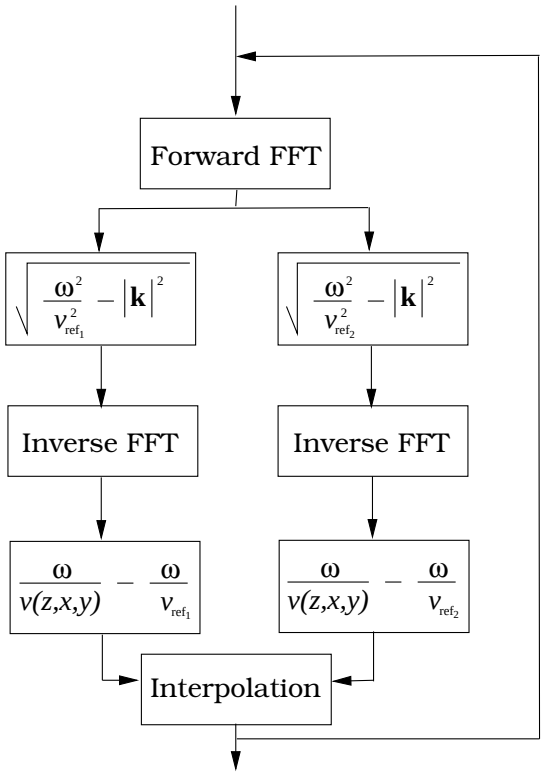


Figure 5.2: Flow-chart of the extended split-step method, which uses multiple reference velocities. The wavefield is downward continued separately for each reference velocity $v_{\text{ref}_1}, v_{\text{ref}_2}, ...$, using the split-step approximation. At the end of each depth step, the resulting wavefields are combined by interpolation. Only one forward FFT is required at each depth step, but as many inverse FFTs are required as reference velocities are used. down-split-var [NR]

Figure 5.2 shows the flow-chart of the extended split-step method, which uses multiple reference velocities. Only one forward FFT is required at each depth step, but as many inverse FFTs are required as references velocities are used. Therefore the cost function of the simple split-step method expressed in equation (5.3) must be multiplied by the number of reference velocities. If the velocity variations are large, several reference velocities are needed to achieve accurate results. Consequently, when the velocity variations are large, the cost of the mixed-domain method may be too high compared to the pure space-domain methods presented in Section 5.3.

Figures 5.3 and 5.4 show the results of using split-step to migrate a zero-offset data set recorded over a salt dome in the Gulf of Mexico. This example illustrates the need for multiple reference velocities when split-step migration is used with a rapidly changing medium

velocity. The sections on the left are extracted from an image cube that was migrated with one reference velocity, whereas the sections on the right were obtained with two reference velocities. There are visible differences between the cross-line sections shown in Figure 5.3. The use of just one reference velocity causes the artifacts on the salt flank, where the migration velocity rapidly changes because of the salt-sediment transition. The mispositioned events are also visible in the depth slice shown on the left in Figure 5.4. These spurious events overlap with the sediments' reflections and distort the image of the salt flanks, making the interpretation of the salt boundaries more difficult.

The real data shown in the previous figure require two reference velocities to be migrated satisfactorily, because the sediment velocity has small lateral variations and the salt dome has a simple shape. In contrast, the proper zero-offset migration of the SEG-EAGE salt data set requires many more reference velocities. Figures 5.5 and 5.6 compare the in-line section migrated with three reference velocities and with six reference velocities. The top of the salt and the salt flanks are better focused when six reference velocities are used. Figures 5.7 and 5.8 compare the cross-line section migrated with three reference velocities and with six reference velocities. The bottom of the salt and the flanks of the canyons in the middle part of the salt body are better focused when six reference velocities are used. However, to obtain the better images, the computational cost was doubled. In the next section we discuss a family of methods that can be used to improve the accuracy of mixed-domain downward continuation more cost effectively than by increasing the number of reference velocities in an extended split-step downward continuation.

5.2.2 Higher-order mixed ω-**k**/ω-**x** methods

The previous examples show that when the lateral velocity variations are strong and continuous – that is the velocity function cannot be simply represented by a few discrete values – the extended split-step method would require several reference velocities to produce good images. In those cases, it is advantageous to use a more accurate approximation to the SSR operator than the split-step approximation shown in equation (5.2). These considerations have led to the developments of several methods that approximate the SSR by adding additional terms, of higher order in the spatial derivatives (**k**), to the split-step approximation. All these methods reduce to the split-step method for vertically propagating waves – that is when **k** vanishes – but propagate dipping events with higher accuracy. The most accurate and efficient method of this group is the **Fourier Finite-Difference (FFD) migration** introduced by Ristow and Rühl (1994). The extended local Born-Fourier migration (Huang et al., 1999) and pseudo-screen propagator methods (Xie and Wu, 1999) are less accurate, but of more straightforward derivation, and thus I will present them first.

Pseudo-screen method

The basic idea of all these methods is fairly simple, and is based on a Taylor expansion of the SSR equation in terms of perturbations of the velocity (or slowness) function, as suggested by

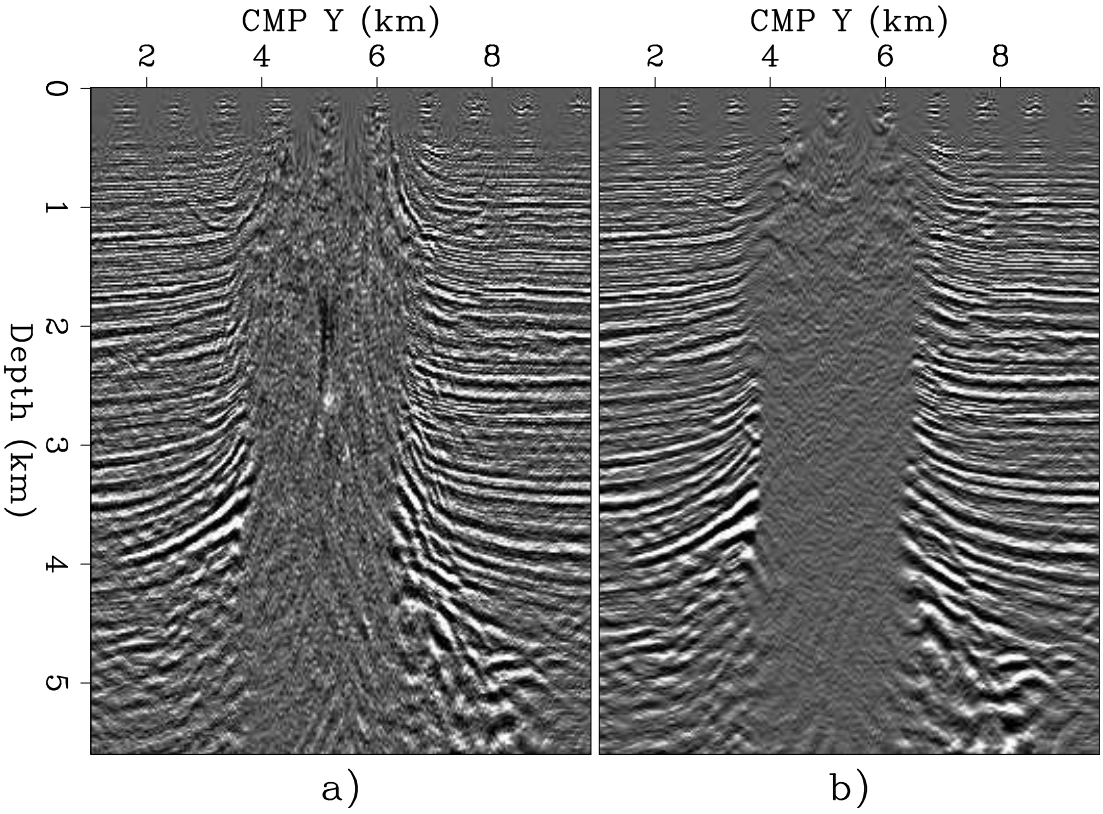


Figure 5.3: Cross-line sections obtained by split-step migration with only one reference velocity (a) and with two reference velocities (b). down-xslice-split-overn [CR,M]

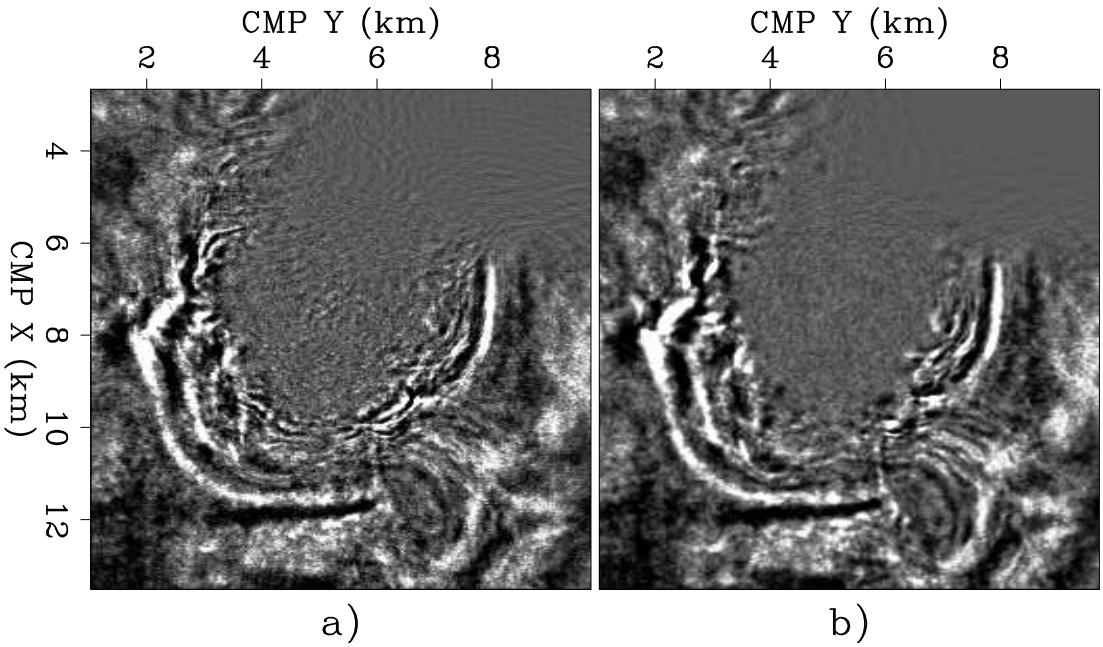


Figure 5.4: Depth slices obtained by split-step migration with only one reference velocity (a) and with two reference velocities (b). down-dslice-split-overn [CR,M]

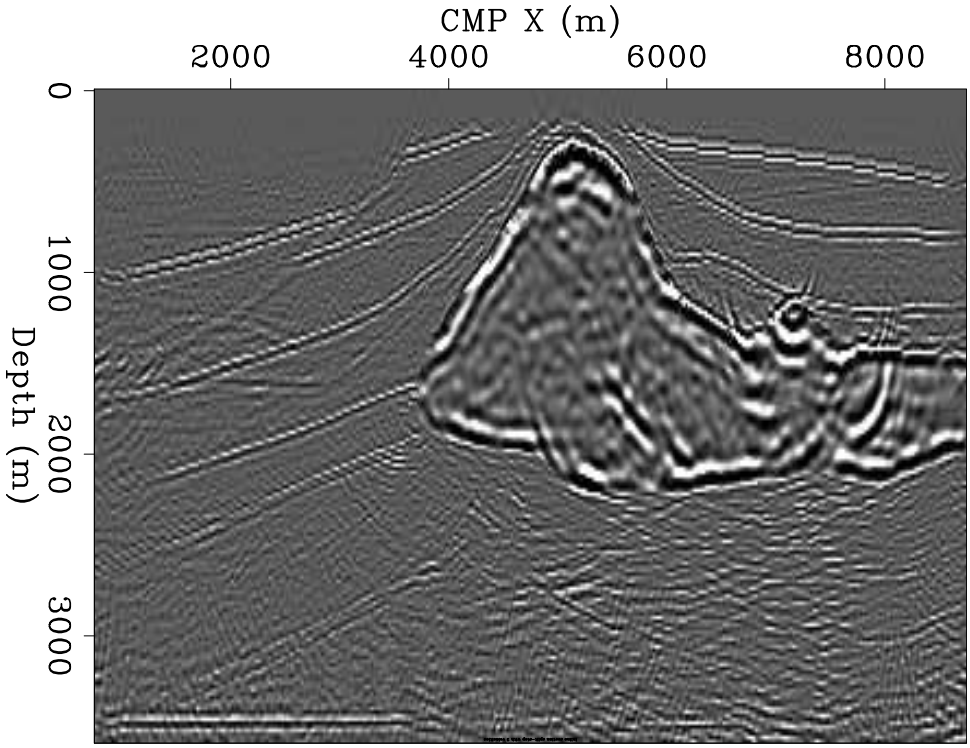


Figure 5.5: In-line section (CMP Y = 8,010 m) of the zero-offset SEG-EAGE salt data set migrated with three reference velocities. down-Salt-3v-x8010 [CR,M]

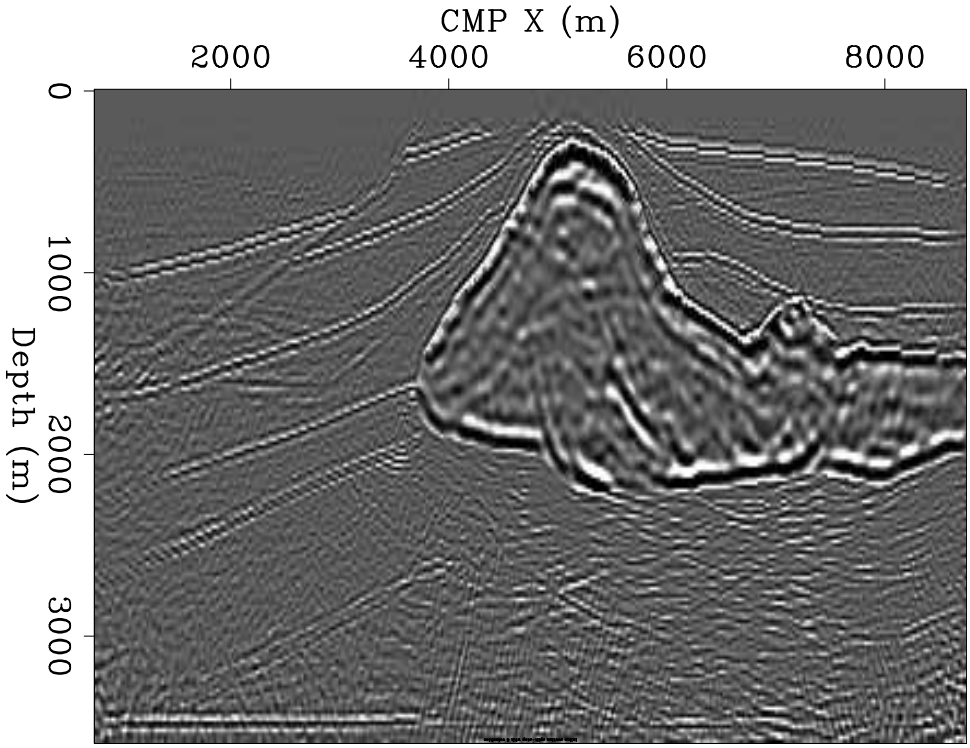


Figure 5.6: In-line section (CMP Y = 8,010 m) of the zero-offset SEG-EAGE salt data set migrated with six reference velocities. down-Salt-6v-x8010 [CR,M]

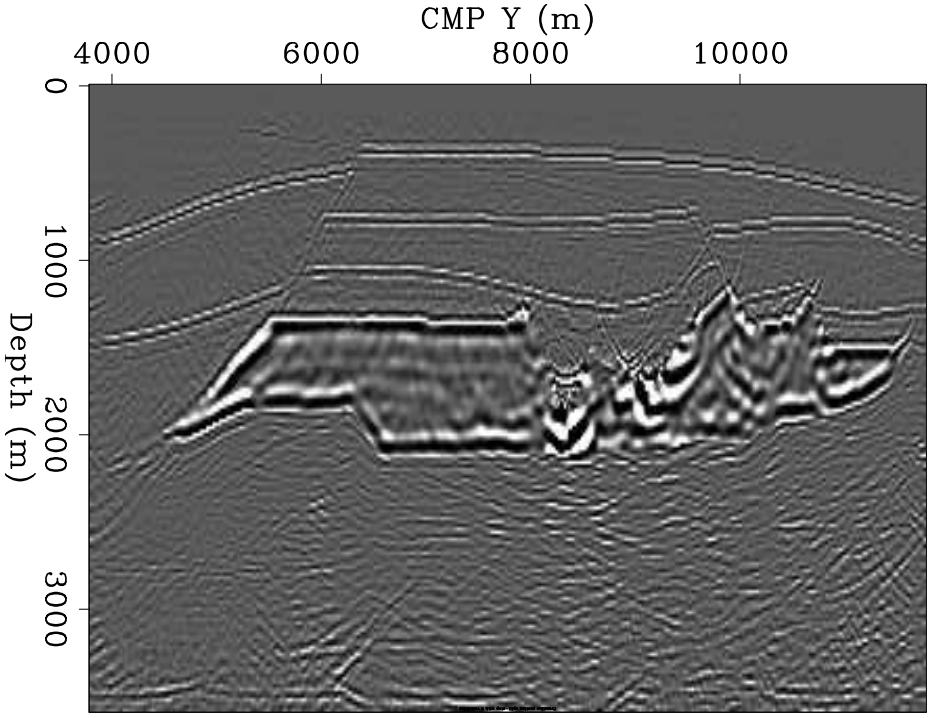


Figure 5.7: Cross-line section (CMP X = 7,440) of the zero-offset SEG-EAGE salt data set migrated with three reference velocities. down-Salt-3v-y7440 [CR,M]

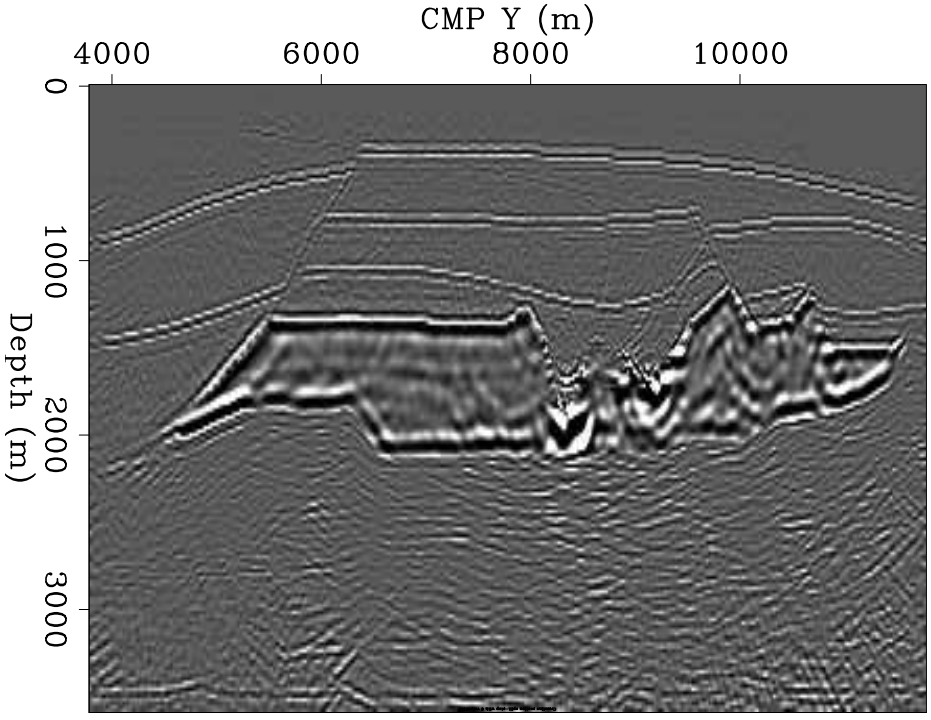


Figure 5.8: Cross-line section (CMP X = 7,440) of the zero-offset SEG-EAGE salt data set migrated with six reference velocities. down-Salt-6v-y7440 [CR,M]

the following equation:

$$P_{z+\Delta z}(\omega,\mathbf{k}) = P_z(\omega,\mathbf{k})e^{ik_z\Delta z} \approx P_z(\omega,\mathbf{k})e^{ik_z^{\text{ref}}\Delta z + i\left.\frac{dk_z}{ds}\right|_{s_{\text{ref}}}\Delta s\Delta z}. \tag{5.4}$$

The first derivative of the vertical wavenumber k_z with respect to slowness is given by

$$\left.\frac{dk_z}{ds}\right|_{s_{\text{ref}}} = \frac{\omega}{\sqrt{1-\frac{|\mathbf{k}|^2}{\omega^2 s_{\text{ref}}^2}}}. \tag{5.5}$$

The methods differ in how this correction term is applied. A robust and computationally efficient way of applying the correction term is by an implicit finite-difference scheme. To apply implicit finite difference we first need to approximate the square root by a rational function. This goal can be accomplished by a continued fraction expansion of the square root. This method was first introduced in geophysics to approximate directly the SSR operator (Claerbout, 1985). A good compromise between accuracy and computational cost is provided by the so called 45-degree approximation. According to this expansion we can write the square root in the denominator of the derivative in equation (5.5) as

$$\sqrt{1-X^2} \approx 1 - \frac{X^2}{2-\frac{X^2}{2}} = \frac{4-3X^2}{4-X^2}, \tag{5.6}$$

where $X = |\mathbf{k}|/\omega s_{\text{ref}}$. Inverting this expression, we can recast the derivative as the sum of a constant (1) and a rational function of X^2; that is,

$$\frac{1}{\sqrt{1-X^2}} \approx 1 + \frac{2X^2}{4-3X^2}, \tag{5.7}$$

and equation (5.5) can be approximated as follows:

$$\left.\frac{dk_z}{ds}\right|_{s_{\text{ref}}} \approx \omega\left[1 + \frac{2v_{\text{ref}}^2X^2}{4-3v_{\text{ref}}^2X^2}\right], \tag{5.8}$$

where obviously $v_{\text{ref}} = 1/s_{\text{ref}}$.

The vertical wavenumber k_z is now approximated by the sum of three terms:

$$\begin{aligned} k_z &\approx k_z^{\text{ref}} + \omega\Delta s + \omega\Delta s\frac{2\left(k_x^2+k_y^2\right)}{4\omega^2 s_{\text{ref}}^2 - 3\left(k_x^2+k_y^2\right)} \\ &= k_z^{\text{ref}} + k_z^{\text{split-step}} + k_z^{\text{f-d}}, \end{aligned} \tag{5.9}$$

where the first term (k_z^{ref}) represents the downward continuation with a reference velocity and is applied in the ω-k domain, the second term ($k_z^{\text{split-step}}$) is the split-step correction and is applied as a simple phase-shift in the ω-x domain, and the third term ($k_z^{\text{f-d}}$) is applied in the ω-x domain as an implicit finite-difference operator.

The direct application of $k_z^{\text{f-d}}$ by an implicit finite-difference method requires the numerical solution of several banded systems of simultaneous equations. The cost of solving a

banded system is proportional to the number of unknowns multiplied by the width of the diagonal band in the matrix. In this case the cost is high because the width of the diagonal band is proportional to the number of grid points along one of the two horizontal directions. Splitting, as presented in Section 5.3.1, can be used to reduce the computational cost. After splitting, the system of simultaneous equations becomes tridiagonal. The computational cost of solving a tridiagonal system is substantially smaller than solving the full banded system because the width of the diagonal band is only three.

To avoid splitting but maintain computational efficiency, new methods based on the helix transform could be used to approximate $k_z^{\mathrm{f-d}}$ (Rickett et al., 1998). However, because splitting is applied to a residual operator, the anisotropy introduced by splitting $k_z^{\mathrm{f-d}}$ is less of a concern than when splitting is applied directly to the SSR (Section 5.3.1).

Splitting is equivalent to approximating $k_z^{\mathrm{f-d}}$ as the sum of two terms:

$$k_z^{\mathrm{f-d}} \approx k_{z_x}^{\mathrm{f-d}} + k_{z_y}^{\mathrm{f-d}} = \omega \Delta s \left(\frac{2k_x^2}{4\omega^2 s_{\mathrm{ref}}^2 - 3k_x^2} + \frac{2k_y^2}{4\omega^2 s_{\mathrm{ref}}^2 - 3k_y^2} \right). \tag{5.10}$$

After splitting, each term ($k_{z_x}^{\mathrm{f-d}}$ and $k_{z_y}^{\mathrm{f-d}}$) can be applied to the wavefield by using a Crank-Nicolson scheme that is equivalent to approximating the exponential operator as follows:

$$\begin{aligned} P_{z'} = P_z e^{i k_{z_x}^{\mathrm{f-d}} \Delta z} &\approx P_z \frac{1 + i\omega\Delta s \frac{2k_x^2}{4\omega^2 s_{\mathrm{ref}}^2 - 3k_x^2} \frac{\Delta z}{2}}{1 - i\omega\Delta s \frac{2k_x^2}{4\omega^2 s_{\mathrm{ref}}^2 - 3k_x^2} \frac{\Delta z}{2}} \\ &= P_z \frac{4\omega^2 s_{\mathrm{ref}}^2 - 3k_x^2 + i\omega\Delta s k_x^2 \Delta z}{4\omega^2 s_{\mathrm{ref}}^2 - 3k_x^2 - i\omega\Delta s k_x^2 \Delta z}. \end{aligned} \tag{5.11}$$

The spatial second-derivative operator, represented by k_x^2, can be approximated by a centered first-order finite-difference approximation of the Laplacian:

$$k_x^2 = -\nabla^2 P \approx \frac{-P^{x-\Delta_x} + 2P^x - P^{x+\Delta_x}}{\Delta_x^2}. \tag{5.12}$$

Substituting equation (5.12) into equation (5.11) leads to the tridiagonal system of equations determined by the following relations:

$$\begin{aligned} &P_{z'}^{x-\Delta_x}(3 + i\omega\Delta s\Delta z) + P_{z'}^{x}\left(4\omega^2 s_{\mathrm{ref}}^2 \Delta_x^2 - 6 - i2\omega\Delta s\Delta z\right) + P_{z'}^{x+\Delta_x}(3 + i\omega\Delta s\Delta z) = \\ &P_{z}^{x-\Delta_x}(3 - i\omega\Delta s\Delta z) + P_{z}^{x}\left(4\omega^2 s_{\mathrm{ref}}^2 \Delta_x^2 - 6 + i2\omega\Delta s\Delta z\right) + P_{z}^{x+\Delta_x}(3 - i\omega\Delta s\Delta z). \end{aligned} \tag{5.13}$$

Implicit finite-difference methods have the important theoretical and practical advantage of being unconditionally stable when the velocity functions (the reference velocity and the medium velocity) are smooth. However, numerical instability may arise when either of the velocity functions has sharp discontinuities (Ristow and Rühl, 1994). Stability is a necessary condition for a migration method to be practically useful. Biondi (2002) presents a reformulation of the finite-difference algorithm outlined in equation (5.13) that guarantees stability

even in presence of sharp discontinuities in the velocity functions. His method is an adaptation of the bulletproofing theory developed by Godfrey et al. (1979) and Brown (1979) for the 45-degree equation.

The improvements in accuracy achieved by applying the higher-order correction are evident when comparing the migration results of the SEG-EAGE salt data set obtained using a simple split-step correction and a higher-order correction. Figure 5.9 shows the cross-line section taken from the migrated cubes obtained by a pseudo-screen method using three reference velocities. It should be compared with the same section obtained using a split-step method with three reference velocities (Figure 5.7) and six reference velocities (Figure 5.8). The flanks of the canyons in the middle part of the salt body and the bottom of the salt are better focused when the higher-order correction is applied. The quality of the image in Figure 5.9 is close to the image obtained using six reference velocities and displayed in Figure 5.8.

Fourier Finite Difference (FFD)

The Fourier Finite Difference (FFD) correction achieves better accuracy than the pseudo-screen correction because it is based on a direct expansion of the difference between the square root evaluated at the medium velocity v and the square root evaluated at the reference velocity v_{ref}, instead of being based on the expansion of the square root around the reference velocity. The downward-continued wavefield is approximated as

$$P_{z+\Delta z}(\omega,\mathbf{k}) = P_z(\omega,\mathbf{k})\, e^{ik_z \Delta z} = P_z\left(\omega,k_x,k_y\right) e^{ik_z^{\text{ref}}\Delta z + i\frac{\Delta k_z}{\Delta s}\Delta s \Delta z}, \tag{5.14}$$

where the Taylor series of the correction term is now

$$\frac{\Delta k_z}{\Delta s} \approx \omega\left[1+\frac{v_{\text{ref}} v X^2}{2}+\frac{v_{\text{ref}} v\left(v_{\text{ref}}^2+v^2+v_{\text{ref}} v\right)X^4}{8}+...\right], \tag{5.15}$$

and the continued fraction approximation of the correction term is

$$\frac{\Delta k_z}{\Delta s} \approx \omega\left[1+\frac{2 v_{\text{ref}} v X^2}{4-\left(v_{\text{ref}}^2+v^2+v_{\text{ref}} v\right)X^2}\right]. \tag{5.16}$$

Notice that equation (5.16) reduces to equation (5.8) if $v = v_{\text{ref}}$. Therefore, at the limit when the difference between the reference velocity and the medium velocity is small, the two correction terms are equivalent, but they differ for larger corrections.

The gain in accuracy achieved by the FFD correction over the simple split-step correction is illustrated in Figure 5.10. It compares the phase curves obtained after the split-step correction [equation (5.2)] and the FFD correction [equation (5.16)] were applied. The medium velocity v is equal to 2 km/s, and two reference velocities are assumed: one 10% lower than the medium velocity (1.8 km/s), the other one 10% higher than the medium velocity (2.2 km/s).

Figure 5.11 shows the impulse responses associated with the phase curves shown in Figure 5.10. The maximum frequency in the data is 42 Hz, and the spatial sampling is 10 m in

both directions. Figure 5.11c shows the exact impulse response for the medium velocity equal to 2 km/s. Figure 5.11a shows the impulse response with reference velocity equal to 2.2 km/s and split-step correction. Figure 5.11b shows the impulse response with reference velocity equal to 2.2 km/s and FFD correction. Figure 5.11d shows the impulse response with reference velocity equal to 1.8 km/s and FFD correction. Figure 5.11e shows the impulse response with reference velocity equal to 1.8 km/s and split-step correction.

In Figure 5.11, starting from the panel on the top and moving downward, the impulse responses get narrower. The shape of the impulse responses obtained using the FFD correction [panels b) and d)] is closer to the correct shape [panel c)] than the shape of the impulse responses obtained using the split-step correction [panels a) and e)]. However, for steep propagation angles the impulse responses obtained using the FFD correction show clear signs of frequency-dispersion. These numerical artifacts are caused by the discretization errors of the horizontal Laplacian operator represented by X^2. To generate these figures I used the first-order three-point approximation of the Laplacian described in equation (5.12). The phase curves shown in Figure 5.10 neglect this approximation, and thus they represent the effective phase shift for zero-frequency data. Frequency-dispersion artifacts could be reduced if the accuracy of the discrete Laplacian operator were improved, for example by use of the well-known "1/6 trick" (Claerbout, 1985). Another way to reduce frequency dispersion would be to employ more accurate, but also more computationally expensive, approximations of the Laplacian, such as a second-order five-point approximation.

To give an idea of the extent of the frequency dispersion, Figure 5.12 shows the impulse response with reference velocity equal to 1.8 km/s and FFD correction, but without frequency dispersion. This impulse response was computed by applying the FFD correction in the frequency-wavenumber domain, taking advantage of the fact that both the reference and the medium velocity are constant. This option is not available in realistic situations when the velocities are laterally varying. The comparison of Figure 5.12 with Figure 5.11d shows that frequency dispersion causes the high frequencies to be undercorrected with respect to the low frequencies, and with respect to the ideal (no dispersion) situation.

Some frequency-dispersion artifacts are unavoidable whenever a finite-difference correction (either pseudo-screen or FFD) is applied. However, they can be substantially mitigated by taking advantage of the errors being in the opposite directions for opposite signs of the velocity correction (e.g Figure 5.11b and Figure 5.11d). Biondi (2002) presents a method (**Fourier Finite-Difference Plus Interpolation - FFDPI**) to exploit these opposite directions of the frequency-dispersion errors, so that the related artifacts are reduced without any additional computational complexity.

Figure 5.13 presents a quantitative illustration of the frequency-dispersion errors and of the advantages of interpolating between two wavefields obtained with FFD corrections of opposite signs. It compares the relative phase errors measured as a function of the propagation angle, for split-step, FFD, FFDPI, and extended split-step. As in Figure 5.10, the medium velocity v is equal to 2 km/s, and two reference velocities are assumed: one 10% lower than the medium velocity (1.8 km/s), and one 10% higher than the medium velocity (2.2 km/s). Two temporal frequencies of the wavefield were assumed: 0 Hz and 100 Hz. The frequency

of 100 Hz corresponds to the Nyquist wavenumber for the waves propagating at 90 degrees with velocity of 2 km/s and spatial sampling of 10 m. Therefore, the 100-Hz error curves shown in Figure 5.13 correspond to the worst possible case for both the FFD and the FFDPI methods. As expected, the FFD corrections have smaller errors than the split-step correction for all angles and both frequencies (0 and 100 Hz). At zero frequency the FFD corrections have lower error than the extended split-step method, though they are computationally less expensive. However, because of frequency dispersion, at 100 Hz the simple FFD correction has a worse behavior than the extended split-step method. Interpolation between the two FFD corrections substantially reduces the zero-frequency errors (to less than 1% up to 65 degrees), and brings the 100 Hz errors below the extended split-step errors.

Interpolation between wavefields reduces also the errors caused by splitting. Section 5.3.1 analyzes in detail the operator anisotropy introduced by splitting. Figure 5.14 shows an example of the phase errors related to this operator anisotropy. It compares relative phase errors as a function of the azimuth measured for a propagation angle of 61 degrees. As in the previous figures, the medium velocity v is equal to 2 km/s, and two reference velocities are assumed: one 10% lower than the medium velocity (1.8 km/s), and one 10% higher than the medium velocity (2.2 km/s). The plots show the phase errors at two frequencies (0 Hz and 100 Hz) for the FFDPI algorithm, the FFD correction starting from the lower reference velocity, and the FFD correction starting from the higher reference velocity. Notice that for both the simple FFD correction cases the azimuthal anisotropy decreases as the frequency increases, though the average phase error increases as well. But the crucial feature of the phase-error function for the FFD correction, is that the azimuthal variations are in opposite directions when the differences between the reference velocities and the medium velocity have opposite signs. Consequently, the phase error of the interpolation method is contained within the $\pm$ 1% band and is much lower than the error of either of the simple FFD corrections. At higher frequencies (100 Hz), the impulse response of FFDPI is almost perfectly isotropic.

Figures 5.15 and 5.16 summarize the analysis of the phase errors related to both frequency dispersion and operator anisotropy. Figure 5.15 shows the depth slice of three impulse responses superimposed onto each other. The outermost circular event corresponds to the FFD correction starting from a reference velocity of 2.2 km/s. The middle event corresponds to the exact impulse response with the medium velocity of 2 km/s. The innermost event corresponds to the FFD corrections starting from a reference velocity of 1.8 km/s. The depth of the slices corresponds to a propagation angle of 64.2 degrees, which is close to the maximum propagation angle (65.4 degrees) for the high reference velocity (2.2 km/s). As predicted by the curves shown in Figure 5.14, the azimuthal anisotropy is frequency-dependent, and the frequency dispersion is smaller for azimuths oriented at 45 degrees with respect to the coordinate axes.

The comparison of Figure 5.16 with Figure 5.15 demonstrates the reduction in migration anisotropy achieved by employing FFDPI in conjunction with splitting. Figure 5.16 is the merge of two impulse responses along the in-line direction, cut at the same depth as the slices shown in Figure 5.15. For negative values of the in-line coordinate, the plot shows the depth slice for the exact impulse response. For positive values of the in-line coordinate, the plot shows the depth slice for the impulse response obtained by FFDPI. It is evident that the re-

sult of the interpolation scheme is much less affected by azimuthal anisotropy and frequency dispersion than the results of the two simple FFD corrections showed in Figure 5.15.

5.3 Frequency-space (ω-x) methods

Pure space-domain (ω-x) methods are preferable to mixed-domain methods when the velocity variations are so strong that too many reference velocities are required for accurate mixed-domain migrations. The ω-x domain methods are based on the approximation of the SSR operator with a finite-length convolutional filter in the space domain. Because ω-x methods are applied as convolutions in the space domain, the coefficients of the depth-extrapolation filter can vary as a function of the midpoint axes. Therefore, ω-x methods can easily accommodate strong velocity variations without any additional computational cost and complexity. However, because the midpoint-wavenumber vector **k** is two-dimensional, a direct implementation of equation (4.4) would lead to a two-dimensional convolutional filter. Two-dimensional filters are often cumbersome to design, and they are expensive to apply, because their computational cost grows with the square of the filter length. Therefore, alternative methods for implementing equation (4.4) in the space domain are necessary to make ω-x migrations competitive with ω-k/ω-x migrations.

Section 5.3.1 presents a simple splitting method to approximate the two-dimensional convolution with a cascade of two one-dimensional convolutions. Splitting is simple and efficient, but it introduces errors that are unacceptable in many practical cases. Section 5.3.2 introduces methods that have been recently introduced to avoid splitting and reduce the large computational cost incurred by full 2-D convolutional filters.

5.3.1 Splitting methods for 3-D downward continuation

The challenge with ω-x methods is that a straightforward representation of equation (4.4) as a 2-D convolutional filter along the midpoint axes results in a fairly expensive algorithm. The cost function for such an algorithm is the following:

$$\mathrm{ZoffDwn}_{(\omega-X)} \propto \kappa_{\mathrm{ZoffDwn}_{(\omega-X)}} \times \left(\mathrm{Nz}_{\xi} \times \mathrm{Nx}_{\xi} \times \mathrm{Ny}_{\xi}\right) \times \left(\mathrm{Nt} \times \mathrm{Nop}^2\right). \tag{5.17}$$

The dependency of the cost function on the square of the filter length Nop makes the method unattractive, hence the desire to split the 2-D convolution into the cascade of two 1-D convolutions. With splitting, the cost function is reduced to a more attractive functional:

$$\mathrm{ZoffDwnSpl}_{(\omega-X)} \propto \kappa_{\mathrm{ZoffDwnSpl}_{(\omega-X)}} \times \left(\mathrm{Nz}_{\xi} \times \mathrm{Nx}_{\xi} \times \mathrm{Ny}_{\xi}\right) \times (\mathrm{Nt} \times 2 \times \mathrm{Nop}). \tag{5.18}$$

Unfortunately, the 2-D convolution expressed in equation (4.4) does not fully separate into the cascade of two 1-D convolutions along the midpoint axes, although equation (4.4) is very similar to the expression for the summation surfaces of zero-offset Kirchhoff migration in equation (2.5). When operators are represented as summation surfaces, cascading an in-line

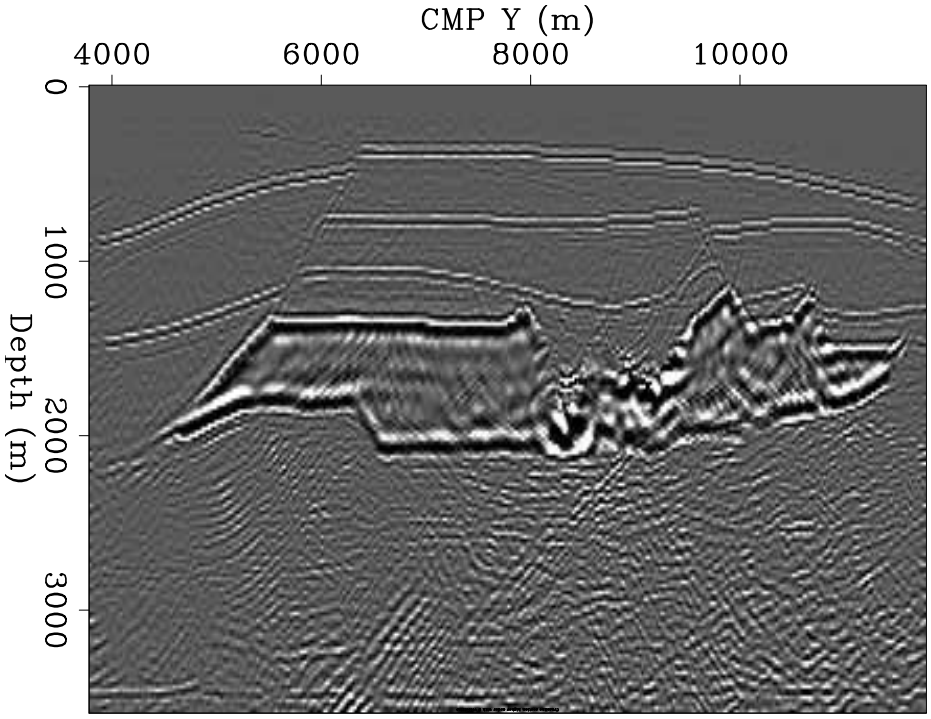


Figure 5.9: Cross-line section (CMP X = 7,440) of the zero-offset SEG-EAGE salt data set migrated with three reference velocities and higher-order correction. down-Salt-3v-fd-y7440 [CR,M]

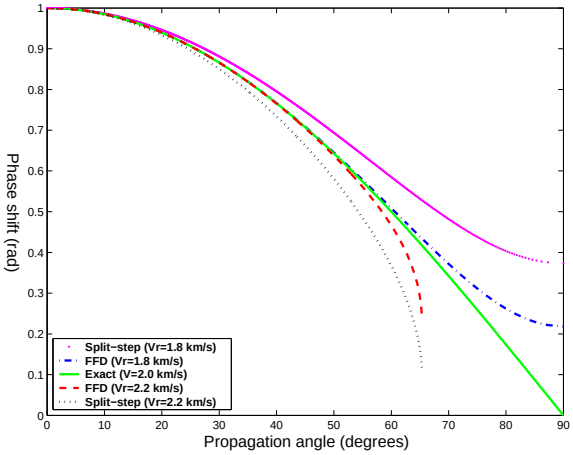


Figure 5.10: Phase curves that compare the accuracy of the Fourier Finite-Difference (FFD) correction with the simple split-step correction. down-ffd_3dsi [CR]

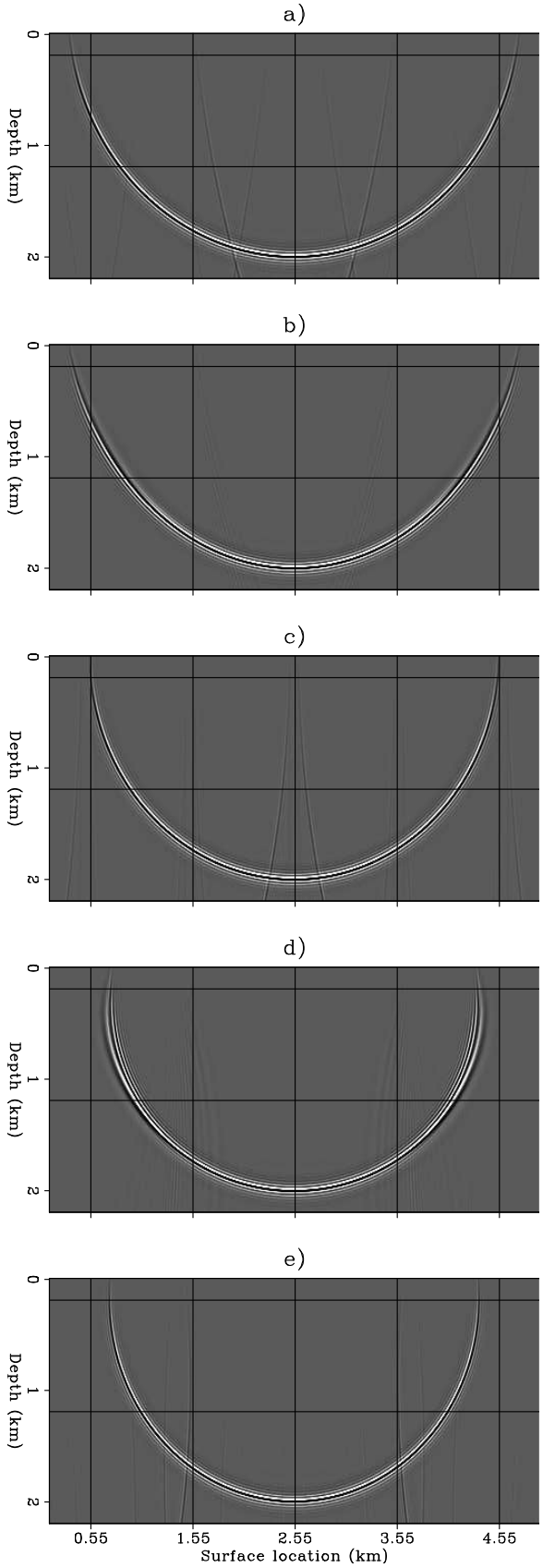


Figure 5.11: Impulse response for the medium velocity equal to 2 km/s and obtained with: a) reference velocity equal to 2.2 km/s and split-step correction, b) reference velocity equal to 2.2 km/s and FFD correction, c) reference velocity equal to 2 km/s (i.e. exact impulse response), d) reference velocity equal to 1.8 km/s and FFD correction, e) reference velocity equal to 1.8 km/s and split-step correction. Starting from the panel on the top and moving downward, the impulse responses get narrower. down-Line-impulse-all-overn [CR,M]

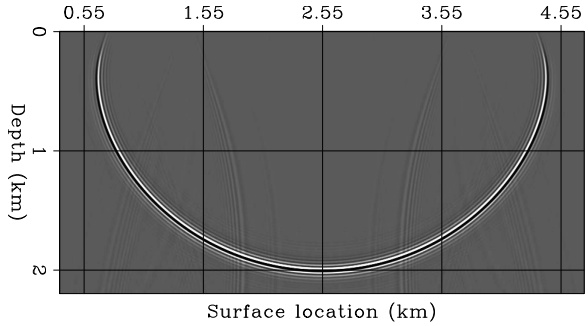


Figure 5.12: Impulse response for the medium velocity equal to 2 km/s and obtained with reference velocity equal to 1.8 km/s and FFD correction applied in the wavenumber domain and thus without frequency dispersion. Compare with Figure 5.11d. down-Line-impulse-pos-hd-overn [CR,M]

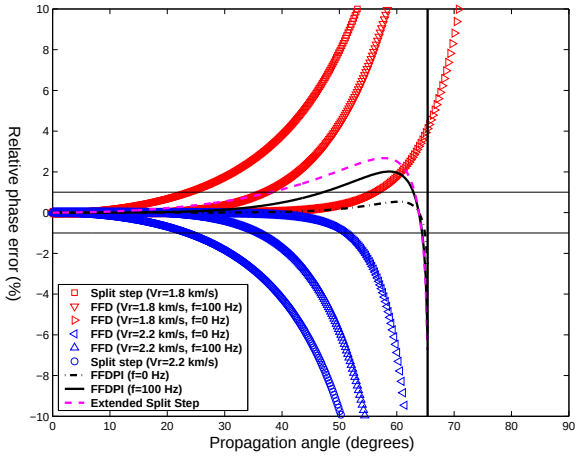


Figure 5.13: Relative phase-error curves assuming v=2 km/s and starting from two reference velocities (v_{ref}=1.8 km/s and v_{ref}=2.2 km/s), for split step, FFD, FFDPI and extended split-step. Two temporal frequencies of the wavefield were assumed: 0 Hz and 100 Hz. The vertical solid line indicates the maximum propagation angle (65.4 degrees) when v_{ref}=2.2 km/s and v=2 km/s. The horizontal solid lines indicate the 1% phase error level. down-errfreq_3dsi [CR]

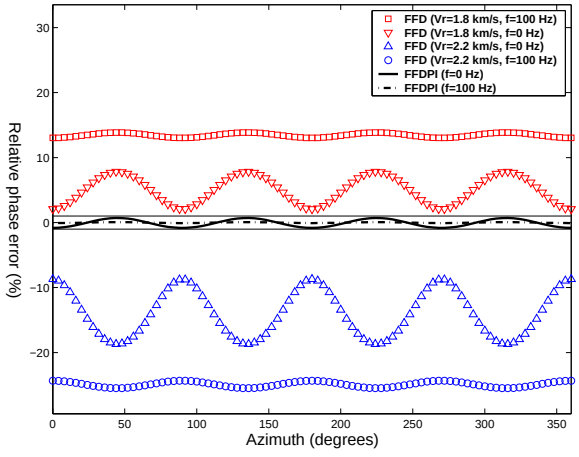


Figure 5.14: Relative phase-error curves for FFD and FFDPI, as a function of the azimuth. The medium velocity was assumed to be v=2 km/s and the two reference velocities were v_{ref}=1.8 km/s and v_{ref}=2.2 km/s. Two temporal frequencies of the wavefield were assumed: 0 Hz and 100 Hz. The horizontal solid lines indicate the $\pm$ 1% phase-error level. down-azimerr_3dsi [CR]

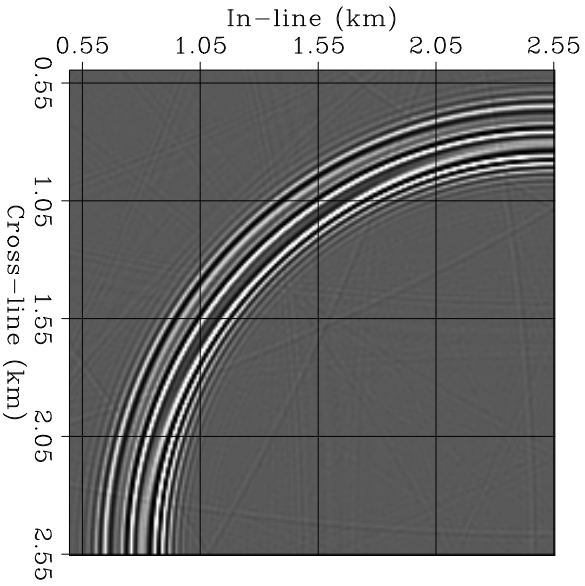


Figure 5.15: Depth slices through impulse responses: 1) innermost event corresponds to the FFD corrections starting from a reference velocity of 1.8 km/s, 2) middle event corresponds to the exact impulse response with the medium velocity of 2 km/s, 3) outermost event corresponds to the FFD corrections starting from a reference velocity of 2.2 km/s. down-Aniso-rist [CR]

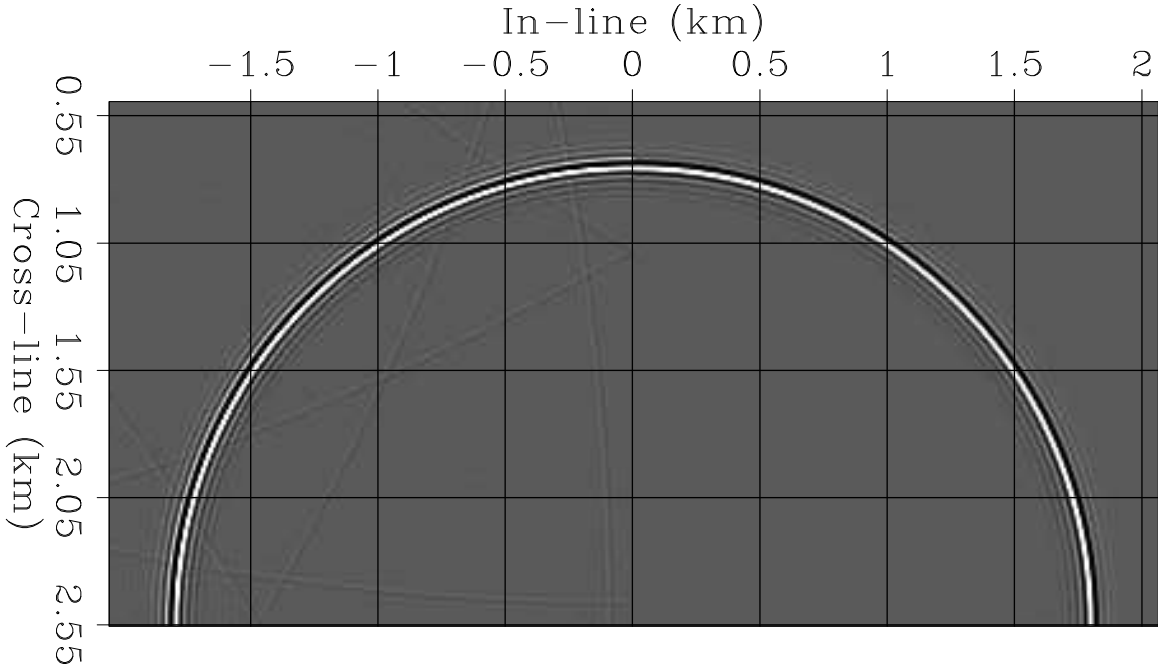


Figure 5.16: Depth slices through impulse responses: 1) left half corresponds to the exact impulse response with the medium velocity of 2 km/s, 2) right half corresponds to the FFDPI results. down-Iso-rist [CR]

operator with a cross-line operator is expressed as a substitution of variables; while, cascading convolutional filters in the wavenumber domain is represented as a multiplication of the corresponding transfer functions. To split the downward-continuation operator, we need to rewrite the SSR dispersion relation as a sum of terms that are functions of either k_{x_m} or k_{y_m}. An exact decomposition of the square root in this fashion is impossible, and thus the SSR equation can be only approximated as a cascade of 1-D convolutional filters. The SSR equation is usually approximated as follows:

$$\mathrm{SSR}(\omega,\mathbf{k}) \approx \underbrace{\sqrt{\frac{4\omega^2}{v(z,x,y)^2}-k_{x_m}^2}}_{\text{Convolution in } x} + \underbrace{\sqrt{\frac{4\omega^2}{v(z,x,y)^2}-k_{y_m}^2}}_{\text{Convolution in } y} - \frac{2\omega}{v(z,x,y)}. \tag{5.19}$$

Splitting is exact for dips that are aligned along either of the midpoint axes, for which either k_{x_m} or k_{y_m} are equal to zero. The error introduced by the approximation in equation (5.19) is maximum for reflections oriented at 45° with respect to the midpoint axes and steeply dipping with respect to the surface.

Figure 5.17 and Figure 5.18 compare depth slices of the migration impulse responses generated with the exact SSR equation (left) and its splitting approximation (right). Figure 5.17 corresponds to a a reflector dip of 30°. It shows that splitting achieves a good approximation of the desired circular response when the reflector dip is moderate. In contrast, Figure 5.18 corresponds to a reflector dip of 60°; the results obtained by splitting show an unacceptable anisotropy of the impulse response.

Figure 5.19 and Figure 5.20 show slices extracted from migrated cubes of a real data set collected in the Gulf of Mexico over a salt dome. Figure 5.19 compares two diagonal slices obtained by splitting (left) and by a full downward-continuation operator (right). The two results show several significant differences both in the quality of the reflectors' focusing and in their positioning. The dipping reflectors close to the salt-sediment boundaries are the ones that are most negatively affected by splitting. Figure 5.20 compares two depth slices through the migrated cubes obtained by splitting (left) and by a full downward-continuation operator (right). The most evident differences between the two sections is the mispositioning caused by splitting of the broad reflector at CMP X=12 km.

The anisotropy caused by splitting can be compensated for with correction filters. Li (1991) first proposed a method based on a correction filter. However, other methods have been recently developed that approximate the SSR equation more accurately than do splitting methods; an example is the McClellan method described in Section 5.3 (Hale, 1991). The computational complexity of these new methods grows only linearly with the filter length, as does the cost of splitting methods, but the new methods have more isotropic impulse responses.

5.3.2 McClellan downward continuation

This section presents a method to approximate the two-dimensional convolution that is more expensive than simple splitting, but for which the computational cost still grows linearly with

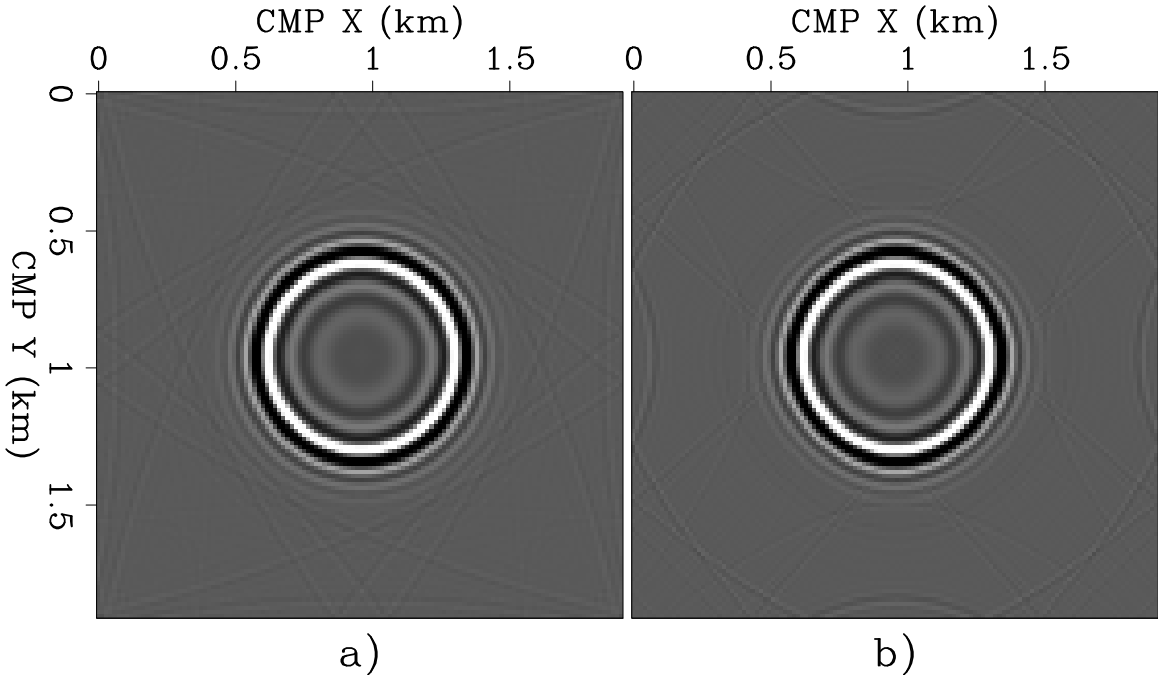


Figure 5.17: Depth slices extracted from zero-offset migration impulse responses corresponding to a reflector dip of 30°, obtained with the exact operator (left) and splitting (right). down-dslice-30-overn [CR,M]

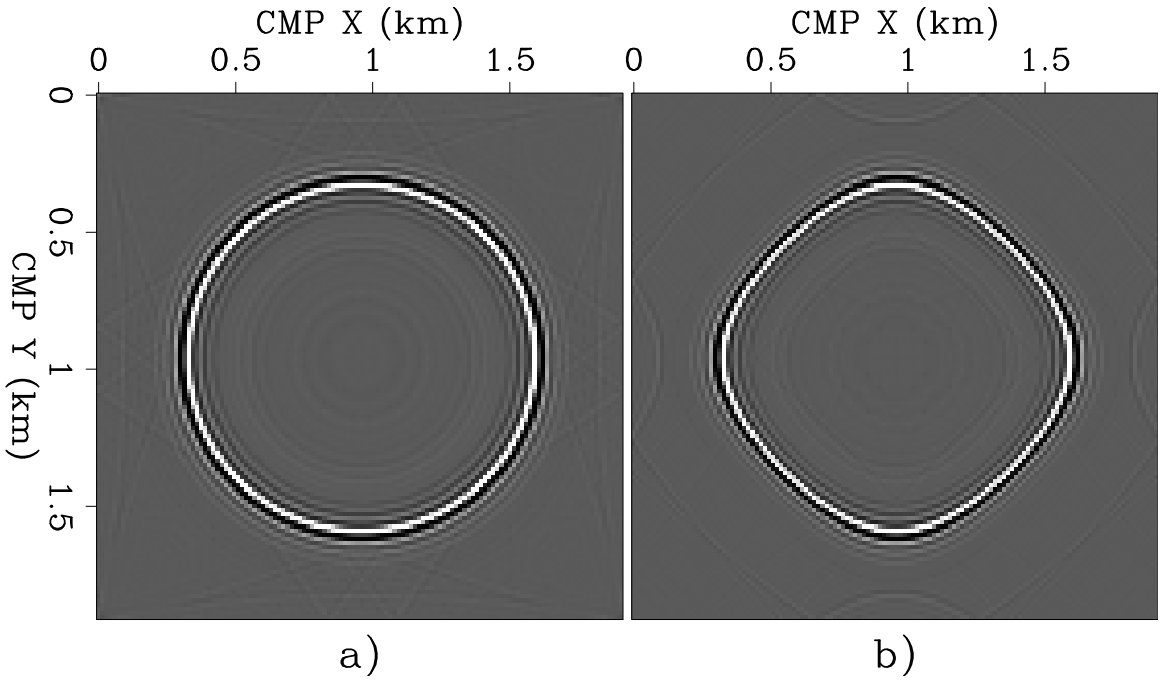


Figure 5.18: Depth slices extracted from zero-offset migration impulse responses corresponding to a reflector dip of 60°, obtained with the exact operator (left) and splitting (right). down-dslice-60-overn [CR,M]

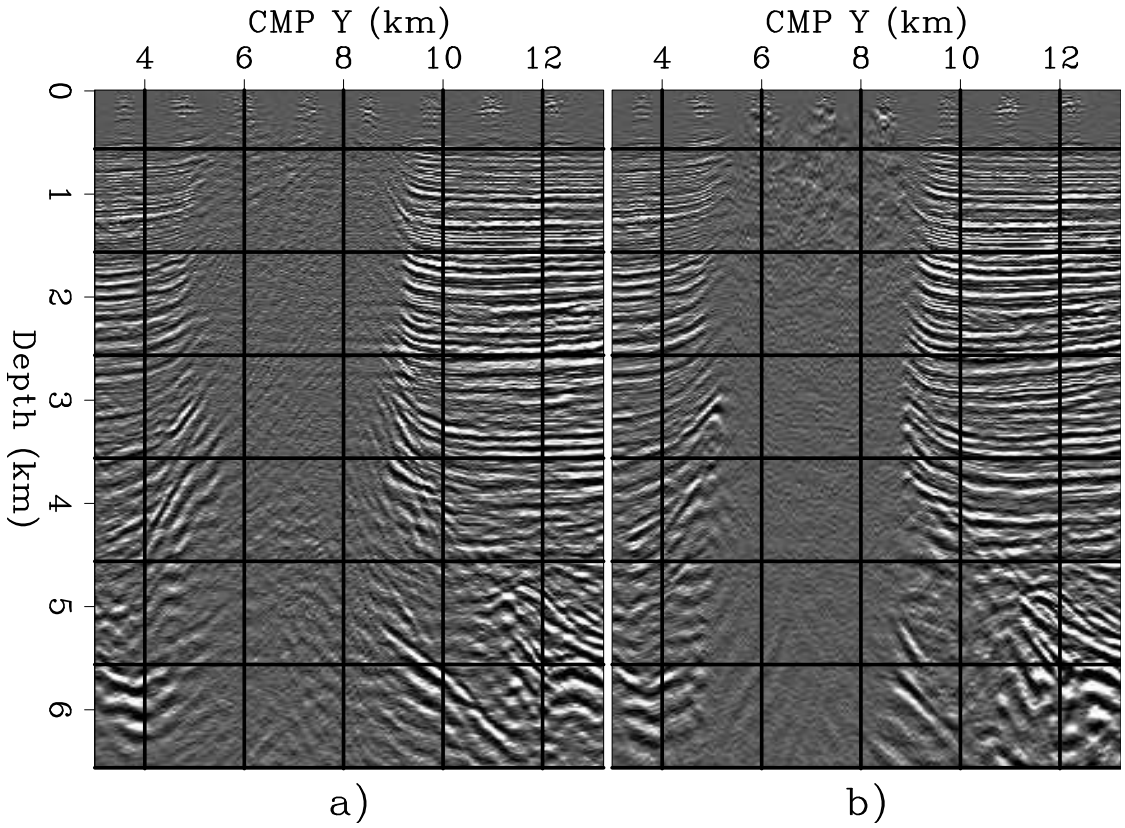


Figure 5.19: Diagonal slices extracted from zero-offset migrations. The slice on the left was obtained with splitting, while the slice on the right was obtained by use of a full downward-continuation operator. Notice the differences in focusing and positioning of the sediments' terminations around the salt dome. down-diag-salt-overn [CR,M]

the filter length, as it does for splitting. To reduce the computational cost and to simplify the design of the continuation operator, the method exploits the circular symmetry of the downward-continuation operator. The symmetry of the continuation operator in the space domain directly derives from circular symmetry of the SSR as a function of the midpoint wavenumbers. SSR is circularly symmetric in the wavenumber domain because it is a one-dimensional function of the midpoint-wavenumber vector magnitude $|\mathbf{k}|$. Therefore, it can be approximated by the following finite-length summation:

$$\exp[\mathrm{SSR}(\omega,\mathbf{k})] \approx \sum_{n=-\mathrm{Nop}+1}^{\mathrm{Nop}-1} h_n(\omega/v)\exp[i\,(n\,|\mathbf{k}|)]\,. \tag{5.20}$$

The circular symmetry of the operator guarantees that the coefficients $h_n(\omega/v)$ are even; that is $h_n(\omega/v) = h_{-n}(\omega/v)$. Therefore, the summation in (5.20) can be rewritten as the following summation of only cosine terms:

$$\exp[\mathrm{SSR}(\omega,\mathbf{k})] \approx h_0(\omega/v) + \sum_{n=1}^{\mathrm{Nop}-1} h_n(\omega/v)\cos(n\,|\mathbf{k}|)\,. \tag{5.21}$$

Notice that the previous expression would be exactly the same if we were considering a 2-D downward continuation filter, where k_{x_m} is substituted for $|\mathbf{k}|$. Therefore the **depth-extrapolation filter** coefficients h_n are function only of the ratio ω/v and can be designed with any of the methods proposed for designing depth-extrapolation filters in 2-D. Nautiyal et al. (1993) present an intuitive method for 2-D filter design.

From a practical viewpoint, it is convenient that the coefficients of the depth-extrapolation filter are a function of the ratio ω/v and not of both variables independently. Tables containing the filter coefficients are thus one-dimensional; they are easy to precompute and can be stored in fast memory without difficulties.

Although equation (5.21) leads to a simpler filter design, its direct application would still require the convolution with the circularly symmetric 2-D filters represented in the wavenumber domain by $\cos(n|\mathbf{k}|)$. These filters are expensive to apply because their length grows linearly with n. Fortunately, the McClellan transforms (McClellan and Chan, 1977) can be used to reduce drastically the cost of applying equation (5.21), as first presented by Hale (1991). The McClellan transforms are based on the following Chebyshev recursion formula for the cosine function:

$$\cos(n\,|\mathbf{k}|) = 2\cos[(n-1)\,|\mathbf{k}|]\cos|\mathbf{k}| - \cos[(n-2)\,|\mathbf{k}|]. \tag{5.22}$$

Using this recursion formula, $\cos(n|\mathbf{k}|)$ can be expressed, for any integer n, in terms of an nth-order polynomial of $\cos|\mathbf{k}|$. Using the recursion of equation (5.22), equation (5.21) can be written in terms of powers of $\cos|\mathbf{k}|$ as follows:

$$\begin{aligned} \exp[\mathrm{SSR}(\omega,\mathbf{k})] &\approx \sum_{n=0}^{\mathrm{Nop}-1} h_n(\omega/v)\cos(n\,|\mathbf{k}|) \\ &= \sum_{n=0}^{\mathrm{Nop}-1} b_n(\omega/v)\cos^n|\mathbf{k}|, \end{aligned} \tag{5.23}$$

where the coefficients b_n can be obtained from the coefficients h_n. Therefore, by using the McClellan transforms, we can substitute recursive convolutions with the compact $\cos|\mathbf{k}|$ filter for the convolution with the long $\cos(n|\mathbf{k}|)$ filters.

An efficient implementation of the McClellan transforms fully exploits the recursive nature of the Chebyshev formula, and requires only $\mathrm{Nop}-1$ applications of the $\cos|\mathbf{k}|$ filter. Figure 5.21 shows a graphical representation of an efficient implementation of the McClellan transforms, where the filter G is a space-domain approximation of $\cos|\mathbf{k}|$.

The simplest, though least accurate, approximation for $\cos|\mathbf{k}|$ is expressed in the wavenumber domain as follows:

$$\cos|\mathbf{k}| \approx G_{9\mathrm{pts}}\left(k_{x_m},k_{y_m}\right) = -1+\frac{1}{2}\left(1+\cos k_{x_m}\right)\left(1+\cos k_{y_m}\right). \tag{5.24}$$

In the space domain, the filter represented by equation (5.24) is the following 3×3 convolutional filter:

$$G_{9\mathrm{pts}} = \begin{array}{|c|c|c|} \hline \texttt{1/8} & \texttt{1/4} & \texttt{1/8} \\ \hline 1/4 & \text{-}1/2 & 1/4 \\ \hline 1/8 & 1/4 & 1/8 \\ \hline \end{array}\ . \tag{5.25}$$

Hale (1991) also suggested a more accurate approximation of the $\cos|\mathbf{k}|$ that is only marginally more expensive. This improved cosine filter is represented in the wavenumber domain as

$$\cos|\mathbf{k}| \approx G_{17\text{pts}}\left(k_{x_m}, k_{y_m}\right) = G_{9\text{pts}} - \frac{c}{2}\left[1-\cos\left(2k_{x_m}\right)\right]\left[1-\cos\left(2k_{y_m}\right)\right], \tag{5.26}$$

where c is chosen by exactly matching $G_{17\text{pts}}$ to $\cos|\mathbf{k}|$ at a particular value of $|\mathbf{k}|$ along the diagonal $k_{x_m} = k_{y_m}$. When $G_{17\text{pts}}$ is matched at $|\mathbf{k}| = \pi/3$, c is equal to 0.0255. The space representation of the $G_{17\text{pts}}$ has 17 coefficients different from zero:

$$G_{17\text{pts}} = \begin{array}{|c|c|c|c|c|}\hline \texttt{-c/8} & \texttt{0} & \texttt{c/4} & \texttt{0} & \texttt{-c/8} \\ \hline 0 & 1/8 & 1/4 & 1/8 & 0 \\ \hline c/4 & 1/4 & -(1+c)/2 & 1/4 & c/4 \\ \hline 0 & 1/8 & 1/4 & 1/8 & 0 \\ \hline -c/8 & 0 & c/4 & 0 & -c/8 \\ \hline \end{array} . \tag{5.27}$$

Both the filters in 5.25 and 5.27 have a quadrantal symmetry that can be further exploited to reduce the number of complex multiplications necessary to convolve G with the wavefield, and thus reducing the computational cost of the method.

Both approximations of the G filter cause errors in the wavefield propagation. These errors mostly cause a slight anisotropy of the effective propagation operator, with obviously the $G_{9\text{pts}}$ filter being more anisotropic than the $G_{17\text{pts}}$ filter. Biondi and Palacharla (1994) proposed an effective method to reduce this operator anisotropy by alternating the usage of $G_{9\text{pts}}$ with the usage of a filter similar to $G_{9\text{pts}}$, but rotated by 45 degrees with respect to the horizontal axes. The additional accuracy is gained at the expenses of a negligible increase in computational cost.

The cost function of McClellan migrations resembles the cost function of splitting [equation (5.18)] because it grows only linearly with the number Nop of coefficients in the depth-extrapolation filter; that is,

$$\text{ZoffDwn}_{(\omega-X)} \propto \kappa_{\text{ZoffDwn}_{(\omega-X)}} \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi \times \text{Nop}\right) \times (\text{Nt}). \tag{5.28}$$

However, the leading constant $\kappa_{\text{ZoffDwn}_{(\omega-X)}}$ is considerably higher for McClellan migration than for splitting, because of the convolutions with the filter G.

The main disadvantage of the application of the McClellan transform to downward continuation is the difficulty of generalizing the method to the practically important case where the samplings along the two horizontal directions are different($\Delta x \neq \Delta y$). Soubaras (1994) presented a method that, like Hale's method, takes advantage of the circular symmetry of the 3-D downward-propagation operator, but is more flexible with respect to spatial sampling.

Figure 5.22 shows two cross-line sections extracted from an image cube obtained by McClellan migration. The section on the left was obtained with less accurate, cheaper filters than those used for the section on the right. For the less expensive solution, Nop was equal to 20 and $G = G_{9\text{pts}}$, whereas Nop was equal to 40 and $G = G_{17\text{pts}}$ for the more expensive solution. The image on the left was obtained in a third of the computer time required to compute the

image on the right. No significant differences are noticeable between the two sections, with the exception that the more expensive migration preserved slightly better the steep reflections.

Figure 5.23 shows two depth slices extracted from the same migrated cubes shown in Figure 5.22. The image obtained with the more expensive filters (right) is slightly sharper than the other one. Again, the main differences are visible in the vicinity of the salt-sediment boundary.

REFERENCES

Biondi, B. L., and Palacharla, G., 1994, 3-D depth migration by rotated McClellan filter: Geophysical Prospecting, **43**, 1005–1020.

Biondi, B., 2002, Stable wide-angle Fourier finite-difference downward extrapolation of 3-D wavefields: Geophysics, **67**, no. 3, 872–882.

Brown, D., 1979, Muir's rules for matrices: Another look at stability: SEP–**20**, 125–142.

Claerbout, J. F., 1985, Imaging the Earth's Interior: Blackwell Scientific Publications.

Gazdag, J., and Sguazzero, P., 1984, Migration of seismic data by phase-shift plus interpolation: Geophysics, **49**, no. 2, 124–131.

Gazdag, J., 1978, Wave equation migration with the phase-shift method: Geophysics, **43**, no. 10, 1342–1351.

Godfrey, R. J., Muir, F., and Claerbout, J. F., 1979, Stable extrapolation: SEP–**16**, 83–87.

Hale, D., 1991, 3-D depth migration via McClellan transformations: Geophysics, **56**, 1778–1785.

Huang, L. Y., Fehler, M. C., and Wu, R. S., 1999, Extended local Born Fourier migration method: Geophysics, **64**, 1524–1534.

Kessinger, W., 1992, Extended split-step Fourier migration: 62nd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 917–920.

Li, Z., 1991, Compensating finite-difference errors in 3-D migration and modeling: Geophysics, **56**, no. 10, 1650–1660.

McClellan, J., and Chan, D., 1977, A 2-D FIR filter structure derived from the Chebyshev recursion: IEEE Trans. Circuits Syst., **CAS-24**, 372–384.

Nautiyal, A., Gray, S. H., Whitmore, N. D., and Garing, J. D., 1993, Stability versus accuracy for an explicit wavefield extrapolation operator: Geophysics, **58**, 277–283.

Rickett, J., Claerbout, J., and Fomel, S., 1998, Implicit 3-D depth migration by wavefield extrapolation with helical boundary conditions: 68th Ann. Internat. Meeting, Soc. Expl. Geophys., 1124–1127.

Ristow, D., and Rühl, T., 1994, Fourier finite-difference migration: Geophysics, **59**, no. 12, 1882–1893.

Soubaras, R., 1994, Signal-preserving random noise attenuation by the f-x projection:, *in* 64th Ann. Internat. Mtg Soc. of Expl. Geophys., 1576–1579.

Stoffa, P. L., Fokkema, J., de Luna Freire, R. M., and Kessinger, W. P., 1990, Split-step Fourier migration: Geophysics, **55**, no. 4, 410–421.

Xie, X. B., and Wu, R. S., 1999, Improving the wide angle accuracy of the screen propagator for elastic wave propagation:, *in* 69th Ann. Internat. Mtg Soc. of Expl. Geophys., 1863–1866.

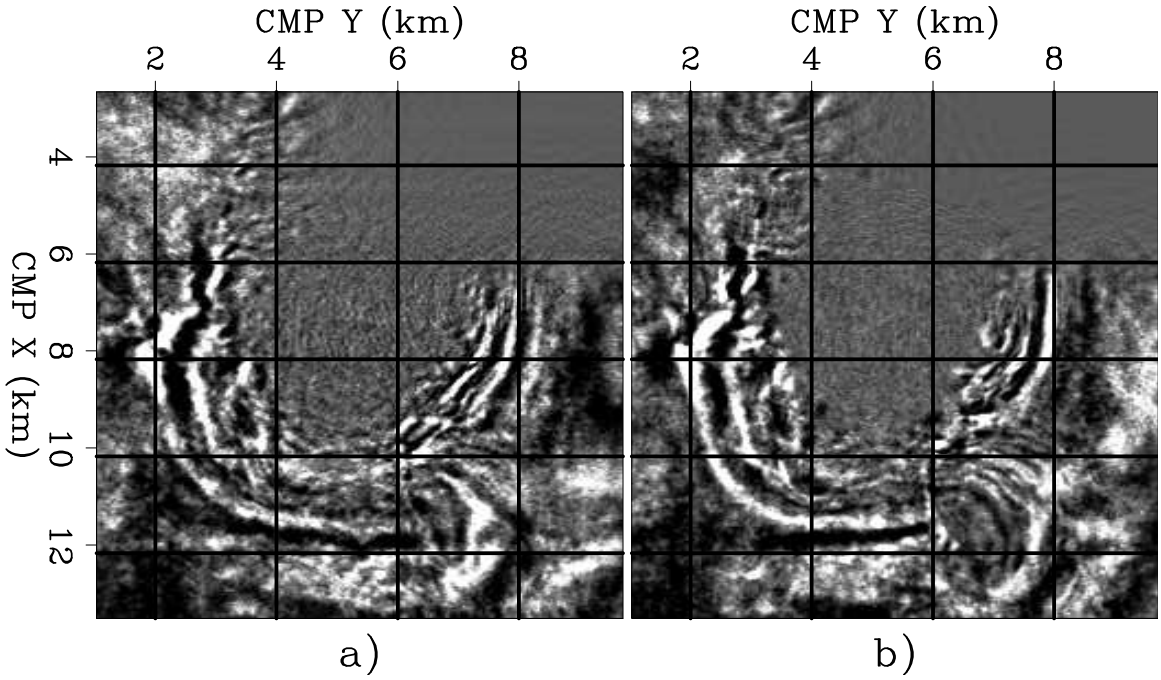


Figure 5.20: Depth slices extracted from zero-offset migrations. The slice on the left was obtained with splitting, while the slice on the right was obtained by use of a full downward-continuation operator. Notice the different positioning of the reflector at CMP X=12 km, and the different phases of many reflectors. down-dslice-salt-overn [CR,M]

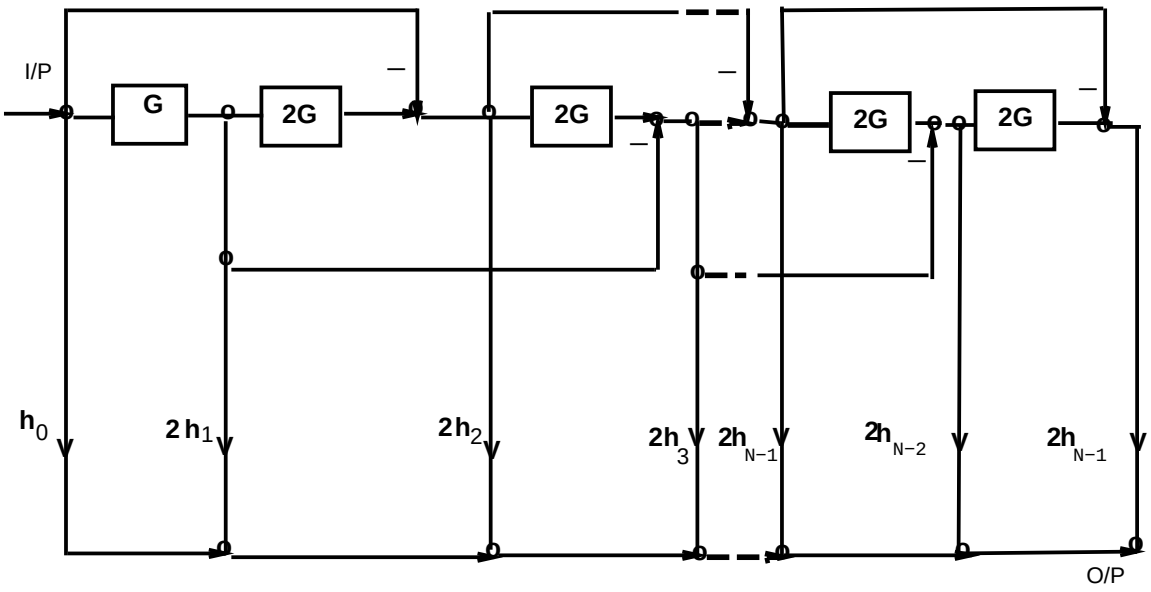


Figure 5.21: Efficient implementation of the recursive algorithm that implements the McClellan transforms. down-cheby [NR]

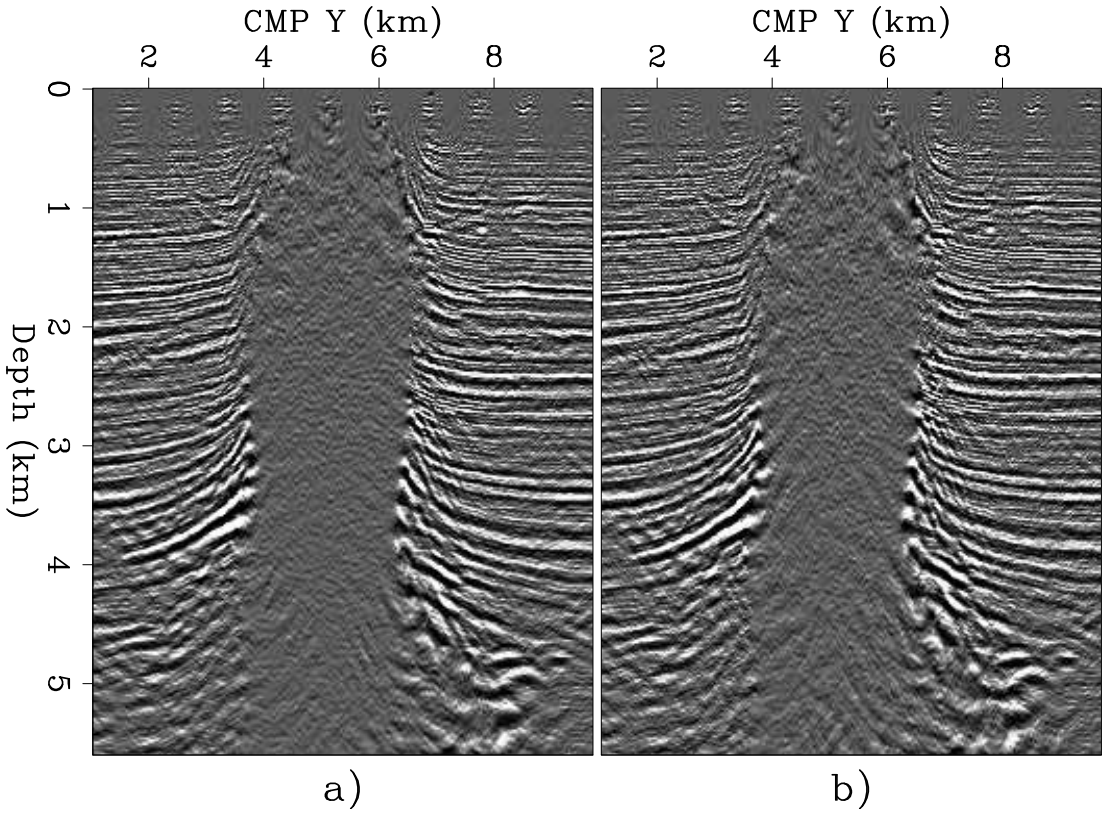


Figure 5.22: Cross-line sections obtained by McClellan migration, with $G = G_{9\text{pts}}$ and Nop=20 (left), and with $G = G_{17\text{pts}}$ and Nop=40 (right). down-xslice-mc-overn [CR,M]

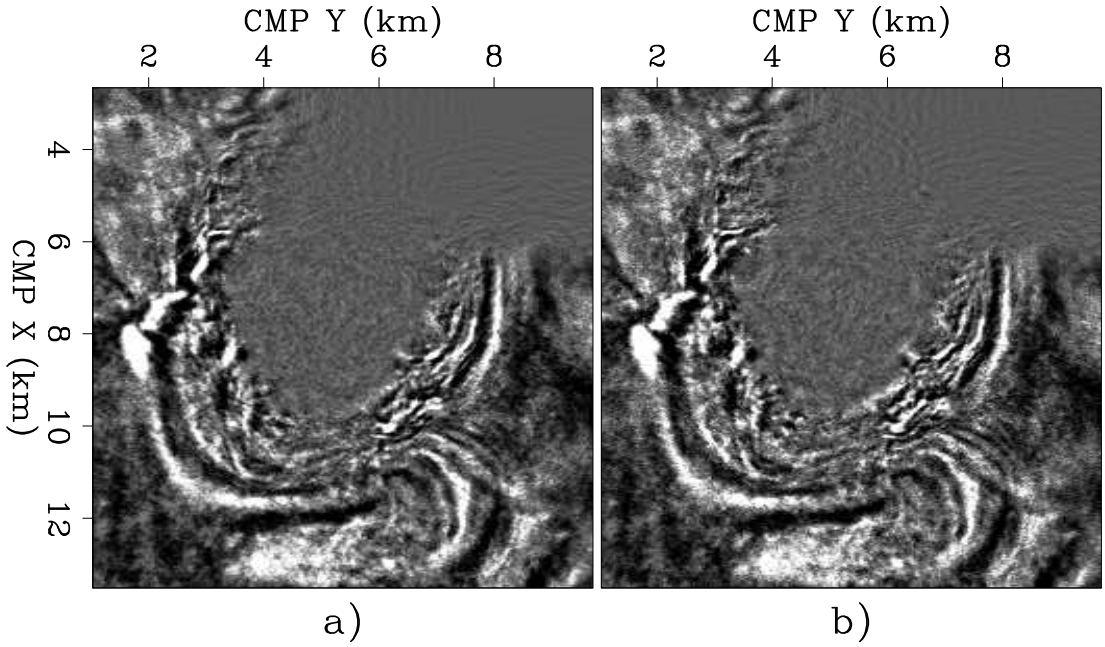


Figure 5.23: Depth slices obtained by McClellan migration, with $G = G_{9\text{pts}}$ and Nop=20 (left), and with $G = G_{17\text{pts}}$ and Nop=40 (right). down-dslice-mc-overn [CR,M]

Chapter 6

Common Image Gathers

All the migration methods presented in the previous chapters produce a migrated cube that is a function of the three spatial coordinates. To perform velocity analysis and amplitude analysis it is often useful to exploit the redundancy of seismic data and produce images with more dimensions than the three coordinates of the physical space. For example, when applying Kirchhoff migration, it is straightforward to image the data as a function of their recording offset and azimuth by subdividing the domain of integration during the Kirchhoff summation. The migrated cubes obtained by subdividing the domain of integration use only a subset of the data, and thus can be referred to as partial or prestack images. The term "prestack images" can be confusing, because it may refer to images obtained by prestack migration; therefore, in the following I will use the term **prestack partial images**. The whole image is a hypercube (usually five-dimensional) made by the ensemble of all partial images. When the partial images are created according to the data offset, the two additional dimensions are either the absolute offset and azimuth (h, θ_h), or the in-line and cross-line offsets (x_h, y_h). When the partial images are created according to the reflection angles at the reflection point, the two additional dimensions are the **reflection opening angle** (γ), and the **reflection azimuth angle** (ϕ).

For both velocity and petrophysical analysis it is particularly useful to measure the variations between the partial images at a fixed image point. The subsets of the whole image with fixed surface location are thus often displayed to perform this kind of analysis. These subsets are often called **Common Image Gathers (CIG)**, or **Common Reflection Point (CRP)** gathers. In the following I will refer to these subsets as CIGs. If the partial images are functions of the data offset, I will refer to the corresponding CIGs as **Offset Domain Common Image Gathers (ODCIGs)**. If the partial images are functions of the reflection angles, I will refer to the corresponding CIGs as **Angle Domain Common Image Gathers (ADCIGs)**.

ODCIGs are the easiest type of CIGs to compute by Kirchhoff migration. However, while this choice is appropriate for time migration problems, in the presence of strong lateral velocity variations, ODCIGs can be affected by artifacts that diminish their utility (Stolk and Symes, 2003). In these cases, ADCIGs tend to be better behaved (Xu et al., 2001; Brandsberg-Dahl et al., 2003). In contrast, ADCIGs are the more natural choice for wavefield-continuation migration methods. The computation of ADCIGs by wavefield-continuation migration is based

on a decomposition of the wavefield, either before imaging (de Bruin et al., 1990; Prucha et al., 1999), or after imaging (Sava and Fomel, 2003; Rickett and Sava, 2002; Biondi and Shan, 2002). The ADCIGs computed before imaging are parametrized as a function of the in-line and cross-line offset ray parameters $\left(p_{x_h}, p_{y_h}\right)$, whereas the ADCIGs computed after imaging are directly parametrized in reflection angles (γ, ϕ). Stolk and Symes (2003) demonstrate that the ADCIGs computed by Kirchhoff migration can still be affected by artifacts when severe multipathing of the wavefield occurs, whereas the ones computed by wavefield-continuation migration are immune to artifacts caused by multipathing. Section 6.1.3 illustrates the origin and nature of these artifacts with the help of simple examples.

6.1 Common Image Gathers by Kirchhoff migration

6.1.1 Offset-Domain Common Image Gathers

Offset-domain partial images are straightforward to produce using Kirchhoff migration. The migration integral is performed on separate subsets of the data, and the partial images are kept separate. The four-dimensional integral of equation (2.1) becomes the following two-dimensional integral over midpoints:

$$I(\xi, \mathbf{h}) = \int_{\Omega_{\xi,\mathbf{h}}} W_{\mathbf{h}}(\xi, \mathbf{m}, \mathbf{h})\, D\left[t = t_D(\xi, \mathbf{m}, \mathbf{h}), \mathbf{m}, \mathbf{h}\right]\, d\mathbf{m}. \tag{6.1}$$

The whole integration domain Ω_ξ (as used in equation 2.1) is now subdivided into several sub-domains $\Omega_{\xi,\mathbf{h}}$, where the vector offset $\mathbf{h}$ can be represented either in Cartesian coordinates (x_h, y_h), or in polar coordinates (h, θ_h). The integration is performed independently on each sub-domain, and the results are analyzed separately for velocity and amplitude analysis. The sizes of the sub-domains $\Omega_{\xi,\mathbf{h}}$ depend on the acquisition geometry; they can be partially overlapping in the offset plane, or be completely separate.

A graphical interpretation of the differences between the integral in equation (6.1) and the integral in equation (2.1) can be gained by considering the summation surfaces shown in Figure 2.3. For the integral in (2.1), the data are simultaneously integrated over both summation surfaces shown in Figure 2.3 (and all the other surfaces corresponding to the offsets within the range of the data acquisition). In contrast, for the integral in (6.1) the data are integrated over only one summation surface at a time. This graphical interpretation is more obvious if we consider the summation curves in 2-D instead of the summation surfaces in 3-D. In 2-D the summation curves for all offsets form a pyramidal surface that is often called "Cheops' pyramid" (Claerbout, 1985). Figure 6.1 shows the common-offset summation curves superimposed onto a Cheops' pyramid. The evaluation of the integral in (2.1) is equivalent to integrating the data on the whole pyramid, whereas, the evaluation of the integral in (6.1) is equivalent to integrating the data along each of the parallel lines superimposed on the pyramid.

Examples of ODCIGs extracted from prestack partial images of the SEG-EAGE salt data set are shown in Figures 6.2-6.3. Figure 6.2 shows a window from an in-line section of the stacked image. Figure 6.3 shows a set of ODCIGs; the offset axis is the absolute half-offset

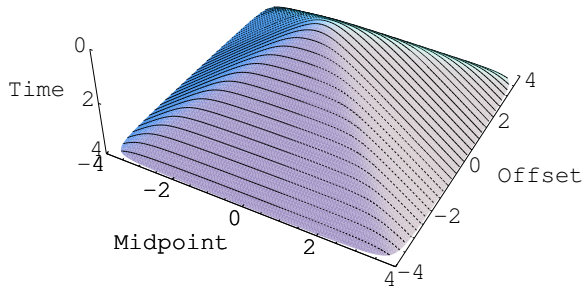


Figure 6.1: Common-offset integration curves superimposed onto the Cheops pyramid. (Courtesy of Sergey Fomel.) cig-coffset [NR]

|**h**|. All ODCIGs shown in Figure 6.3 are taken at the same surface location (CMP X=7.02 km along the section shown in Figure 6.2), but were obtained using different velocity functions, to illustrate the sensitivity of ODCIGs to velocity error. The third panel was obtained using the correct velocity above the salt, and thus the reflections are aligned along the offset axis. A migration velocity lower than the true one was used to produce the first two panels. In these panels the reflections smile upward. The misalignment is more pronounced in the first panel than in the second one, because the velocity error is larger. A migration velocity higher than the true one was used to produce the right-most panel, and the reflections frown downward.

6.1.2 Angle-Domain Common Image Gathers

The basis of angle-domain Kirchhoff imaging is a reflector-centered parametrization of the migration summation surfaces, whereas the conventional offset-domain imaging is based on a surface-centered parametrization, which is a function of the data midpoint and offset. The summation surfaces are parametrized as functions of the angles at the reflection point. Since the geometrical complexity in 3-D may hamper understanding of the basic concepts, I first analyze ADCIGs in 2-D. Section 6.2 generalizes the 2-D analysis to 3-D ADCIGs obtained by wavefield-continuation methods. Kirchhoff-based 3-D ADCIGs are well described by Brandsberg-Dahl et al. (2003).

The schematic in Figure 6.4 defines the angles relevant to 2-D ADCIGs: the reflection opening angle γ and the **geologic dip angle** α_x (along the in-line direction). The opening angle and the dip angle are related to the propagation angles of the source and receiver rays (β_s, β_r) by the following simple relationships:

$$\gamma = \frac{\beta_r - \beta_s}{2}, \quad \text{and} \quad \alpha_x = \frac{\beta_s + \beta_r}{2}. \tag{6.2}$$

To perform angle-domain Kirchhoff imaging in 2-D, the migration integral in equation (6.1) can thus be rewritten as an integration over α_x at constant γ, as follows:

$$I(\xi,\gamma) = \int_{\Omega_{\xi,\gamma}} W_\gamma(\xi,\alpha_x,\gamma)\, D\left[t = t_D(\xi,\alpha_x,\gamma), x_m(\xi,\alpha_x,\gamma), x_h(\xi,\alpha_x,\gamma)\right] d\alpha_x, \tag{6.3}$$

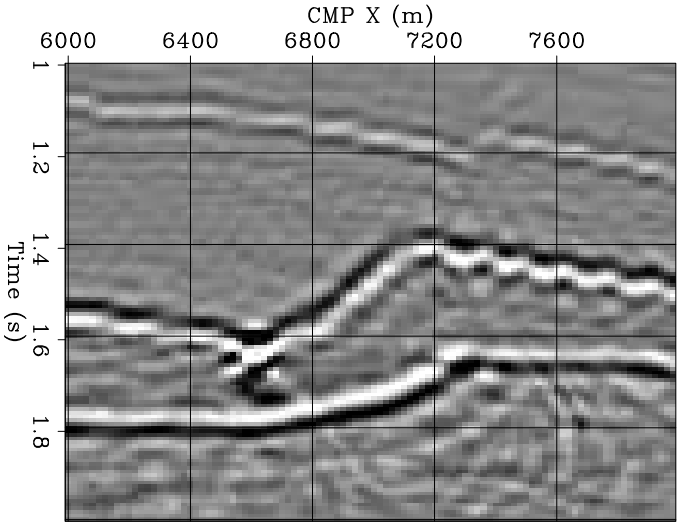


Figure 6.2: Migrated in-line section from the SEG-EAGE salt data set in the area where a major fault cuts through the salt body. cig-InLine-1-mo [CR]

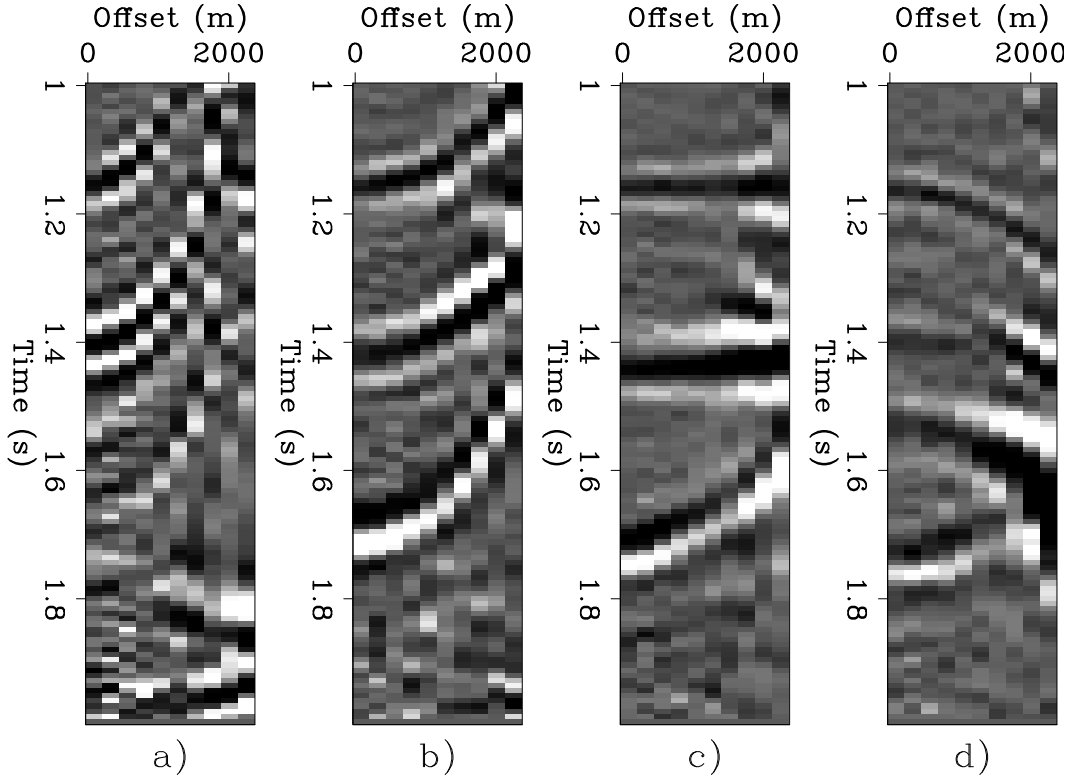


Figure 6.3: ODCIGs taken at the same surface location (CMP X=7.02 km) from the section shown in Figure 6.2. The migration velocity (v) increases from the panel on the left to the panel on the right as a constant fraction of the true velocity (v_0): a) $v = .8\ v_0$, b) $v = .92\ v_0$, c) $v = v_0$, and d) $v = 1.08\ v_0$. cig-cig-off-vel-overn [CR,M]

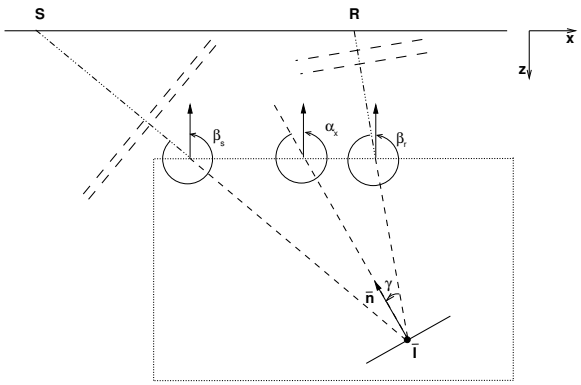


Figure 6.4: A schematic of the geometry of an angle-domain CIG gather in 2-D. Depending on the context, the angles can be either the angles formed by the propagation direction of the rays, or those formed by the propagation direction of the associated plane waves. The arrows indicate positive angles; that is, β_s, β_r, and α_x are negative, and γ is positive. This sign convention is consistent with upward propagating rays (waves). cig-cig-simple-v3 [NR]

where $\Omega_{\xi,\gamma}$ are the integration sub-domains, defined as functions of the aperture angle γ.

The summation curves depend on the velocity model, and are defined parametrically by the functions $t_D(\xi,\alpha_x,\gamma), x_m(\xi,\alpha_x,\gamma), x_h(\xi,\alpha_x,\gamma)$. In the general case, they must be computed numerically by ray tracing (Section 2.2). As in the offset-domain imaging case, when the velocity is constant, the expressions for the traveltime, midpoint and offset as functions of the reflection and dip angles can be derived analytically using trigonometry (Fomel and Prucha, 1999). In this case, the integration path for the migration integral in equation (6.3) is defined parametrically as follows:

$$t_D = \frac{2z_\xi}{V}\frac{\cos\alpha_x\cos\gamma}{\cos^2\alpha_x-\sin^2\gamma}, \tag{6.4}$$

$$x_h = z_\xi\frac{\sin\gamma\cos\gamma}{\cos^2\alpha_x-\sin^2\gamma}, \tag{6.5}$$

$$x_m = x_\xi + z_\xi\frac{\sin\alpha_x\cos\alpha_x}{\cos^2\alpha_x-\sin^2\gamma}. \tag{6.6}$$

For the denominator in equations (6.4-6.6) not to vanish, the admissible range of α_x and γ is defined by the following condition:

$$\cos^2\alpha_x > \sin^2\gamma \quad \text{or} \quad |\alpha_x|+|\gamma| < \frac{\pi}{2}\,. \tag{6.7}$$

This analytical condition corresponds to the geometric condition that both the source and receiver ray emerge at the surface.

Figure 6.5 shows the common-reflection-angle curves superimposed onto the same Cheops pyramid as in Figure 6.1. The comparison of Figure 6.5 with Figure 6.1 illustrates the similarities and differences between common-offset migration and common-angle migration. First, both families of curves cover the pyramid; therefore, the images obtained by stacking the CIGs over offset or reflection angle are equivalent, if the integration weights properly take into account the Jacobian of the transformation defined by equations (6.4-6.6). Second, the

zero-offset curve is the same as the normal-incidence curve. Third, the curves for narrow reflection angles are close to the curves for short offsets, but the curves for wider reflection angles differ substantially from the wide-offset curves.

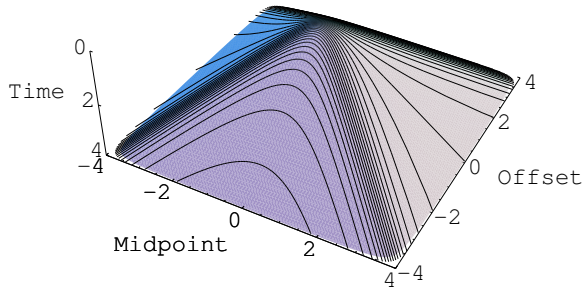


Figure 6.5: Common-reflection-angle integration curves superimposed onto the Cheops pyramid. (Courtesy of Sergey Fomel.) cig-cangle [NR]

Figure 6.5 also shows that the common-angle summation curves for one point diffractor at depth may span a wide range of offsets as a function of the traveltime. This characteristic of the summation curves creates a challenge for efficient implementation of common-angle migration algorithms. In common-offset migration, the input data are accessed according to a predictable and local pattern (each common-offset cube is accessed sequentially). In contrast, in common-angle migration, the input data are accessed according to a more global (and strongly model-dependent for depth migration) pattern. For large 3-D data sets, this quasi-random data access may make it difficult to avoid frequent data reading from disk devices.

6.1.3 Artifacts in Common Image Gathers by Kirchhoff migration

When the complexity in the velocity function causes multipathing of the reflected energy, the summation methods introduced in the previous section produce ODCIGs and ADCIGS that may be affected by artifacts. These artifacts appear because reflectors cannot be always unambiguously imaged by integrating the data along sub-domains defined by a constant parameter (either $\mathbf{h}$ or γ and ϕ). In these cases, ADCIGs tend to be better behaved (Xu et al., 2001; Brandsberg-Dahl et al., 2003) than ODCIGs, but they are still prone to artifacts (Stolk and Symes, 2003). These artifacts disappear when the partial images are summed together (assuming that the acquisition geometry is appropriate to illuminate the reflectors in the subsurface), but can mislead the interpretation of the CIGs before stacking. I will illustrate the origin and nature of these artifacts with the help of a schematic for common-offset migration and a synthetic numerical example for common-angle migration.

Artifacts in ODCIG

The origin of the artifacts in common-offset migration has a simple, direct explanation: in common-offset cubes, the propagation directions of the reflections at the surface are not unambiguously determined from their time dips along the midpoint axis alone (Xu et al., 1998;

Prucha et al., 1999; Stolk and Symes, 2003). This ambiguity problem is conceptually similar to the fact that out-of-plane reflections cannot be distinguished from in-plane reflections using 2-D seismic (Section 0.1).

The propagation directions of reflections recorded at the surface are unambiguously determined if all the components of the slowness vectors at the source location $\mathbf{p_s}$ and at the receiver location $\mathbf{p_g}$ are known. Because we assume the velocity to be known, the slowness vectors are determined if we know all horizontal components $p_{x_s}, p_{\hat{\mathbf{s}}_z}, p_{x_g}, p_{y_g}$. These slowness-vector components are related to time dips in the midpoint-offset domain as follows:

$$p_{x_s} = \frac{p_{x_m} - p_{x_h}}{2} \quad ; \quad p_{x_g} = \frac{p_{x_m} + p_{x_h}}{2}, \tag{6.8}$$

$$p_{\hat{\mathbf{s}}_z} = \frac{p_{y_m} - p_{y_h}}{2} \quad ; \quad p_{y_g} = \frac{p_{y_m} + p_{y_h}}{2}. \tag{6.9}$$

From equations (6.9) it is evident that knowing only p_{x_m} and p_{y_m} is not sufficient to uniquely determine all $p_{x_s}, p_{\hat{\mathbf{s}}_z}, p_{x_g}, p_{y_g}$. However, when no wavefront triplications occur, constant-offset migration produces good results, because the ambiguity is automatically resolved by the differences in arrival time of the reflections. Conversely, when the wavefronts triplicate, this ambiguity cannot always be resolved.

Figure 6.6 and Figure 6.7 illustrate this concept in a simple model case. Figure 6.6 shows the diffractor positions, the raypaths, and the velocity model. I consider two diffractors at the same horizontal location, but at different depths. The velocity function is a constant-velocity background with a circular slow velocity anomaly. The center of the anomaly is exactly located at the position of the shallower diffractor. The rays are all straight, because they propagate either at constant velocity (deeper diffractor) or parallel to the velocity gradient (shallower diffractor). For both diffractions, the time dips measured along the midpoint axis are the same, and equal to zero. With the appropriate choice of magnitude for the velocity anomaly, the arrival times are also the same. In this case, the two diffractors cannot be distinguished using common-offset data.

Figure 6.7 provides another perspective of the same model case by showing the constant-offset integration curves for the two diffractors. The thinner line is for the shallower diffractor (X_1), and the thicker one is for the deeper one (X_2). The two curves are tangent to each other at the apex, which is located at the midpoint analyzed in Figure 6.6. Even if only one diffractor existed in the subsurface, the reflected energy at the apex would constructively interfere during the integration along both curves. That is, common-offset migration would create an image for both diffractors, of which one would be a false image. However, the two diffractors are unambiguously distinguishable from multi-offset data, because the time dips along the offset axis are different, as it is evident from Figure 6.6 and equations (6.9).

Artifacts in ADCIG

Xu et al. (2001) and Brandsberg-Dahl et al. (2003) present convincing examples wherein ADCIGs are less prone to artifacts than ODCIGs. The intuitive reason for the robustness of ADCIGs in the presence of multipathing is that they are based on partitioning the image

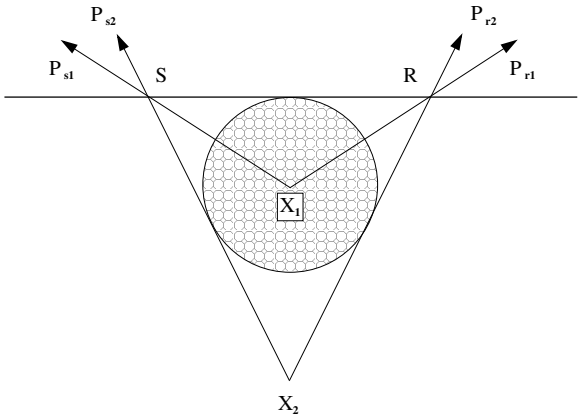


Figure 6.6: Simple model case of ambiguity in constant-offset migration. The velocity model is a constant background with a circular slow-velocity anomaly. The positions of the diffractors are marked as X_1 and X_2. (Courtesy of Marie Clapp.) cig-offset1-new [NR]

according to image-space parameters (e.g. reflection opening angle and reflection azimuth) instead of data-space parameters. However even ADCIGs obtained by summation methods are not completely immune to artifacts. Stolk and Symes (2003) demonstrate that the artifacts are related to the fundamental ambiguity in the propagation of the recorded events, which is unresolved when only subsets of the data are used for imaging. I illustrate this concept by applying downward-continuation migration using the SSR operator to the zero-offset section of a synthetic data set. As discussed in Section 4.2.2, this migration is equivalent to a common-angle migration with $\gamma = 0$, because it assumes that the incident wavefield and the reflected wavefield travel along exactly the same path, though in opposite directions. The presence of these artifacts when using a wavefield-continuation migration instead of a Kirchhoff migration confirms that is the use of only a subset of the whole data set (e.g. the zero-offset section), that causes the artifacts, not the use of a particular migration method. Since normal-incidence reflections are recorded exclusively at zero offset, a Kirchhoff migration would suffer from the same problems as the wavefield-continuation migration used in this example.

Figure 6.8 shows the zero-offset section (Figure 6.8a), and the CMP gather located at the origin of the midpoint axis (Figure 6.8b), extracted from the synthetic data set used to illustrate the problem of artifacts in ADCIGs. The data set is the same one I used in Section 4.2.3.

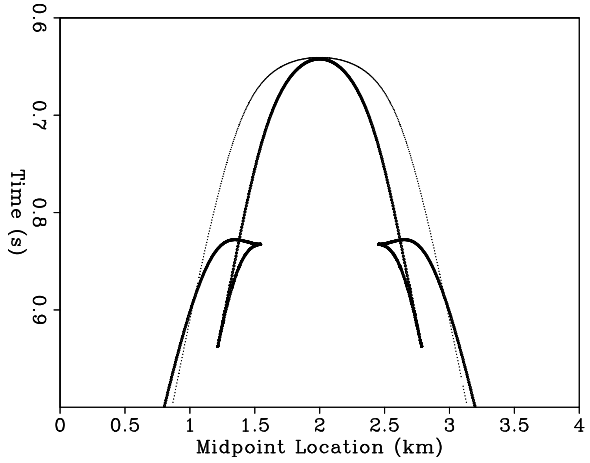


Figure 6.7: Diffraction curves corresponding to the velocity model and diffractor locations shown in Figure 6.6. Notice that the two curves are tangent to each other at the apex. (Courtesy of Marie Clapp.) cig-dualcurve1 [NR]

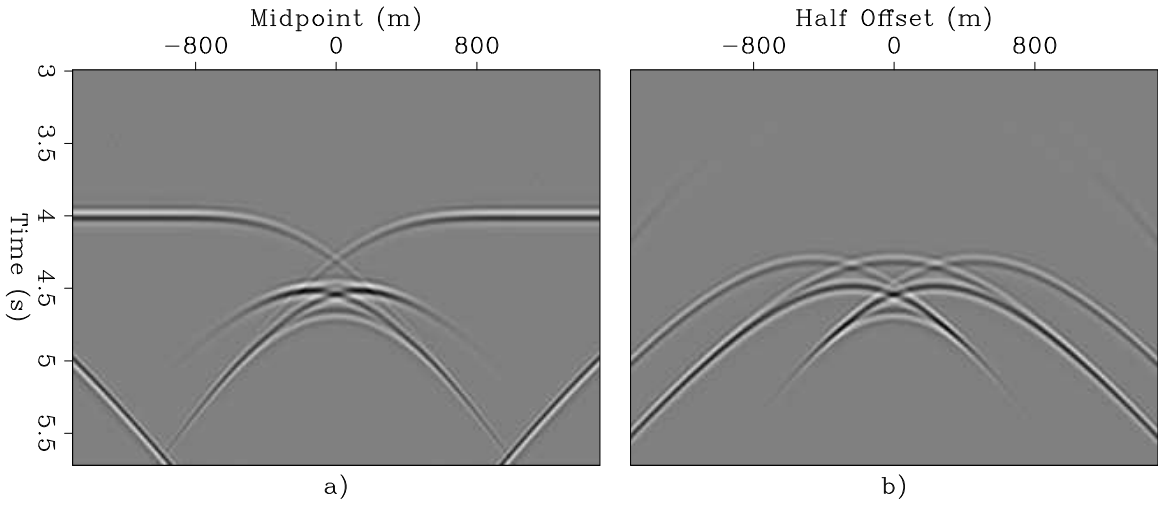


Figure 6.8: The zero-offset section (panel a) and the CMP gather located at the origin of the midpoint axis (panel b) extracted from the synthetic data set used to illustrate the problem of artifacts in ADCIGs. The artifacts are related to the curved event recorded at zero offset around 4.5 seconds and corresponding to a reflection with $\gamma \neq 0$. cig-zoff-cmp-overn-z0 [CR,M]

Figure 4.14 shows the velocity function assumed to model the data. The reflector has a simple flat geometry, but a strong velocity anomaly above it creates severe multipathing. The curved event recorded at zero offset (Figure 6.8a) around 4.5 seconds corresponds to reflections with $\gamma \neq 0$. Figure 6.9 shows two pairs of raypaths corresponding to this event. These reflections are not at normal incidence, and thus their time slope along the offset axis is not zero (Figure 6.8b), indicating that, at the surface, the propagation direction of the down-going wavefield is different from the propagation direction of the up-going wavefield. Though this information is available in the complete data set, common-angle migration with $\gamma = 0$ is insensitive to the slope along the offset axis, and consequently it images this event into a spurious reflector.

The image shown in Figure 6.10 was obtained by applying the SSR-migration to the zero-

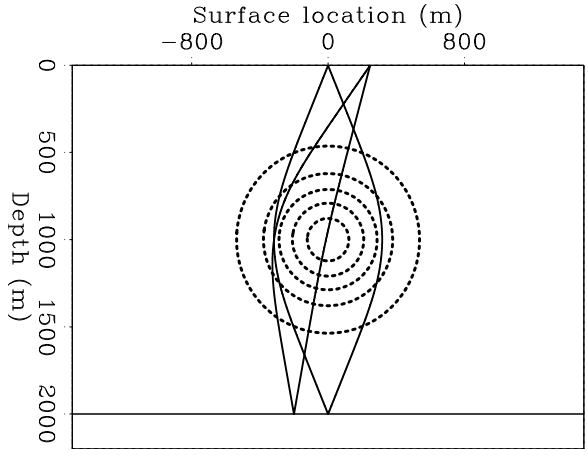


Figure 6.9: Two pairs of raypaths corresponding to zero-offset events reflected at non-normal incidence. Notice that $\gamma \neq 0$ but $\alpha_x = 0$ because at the reflection point the rays are symmetric with respect to the vertical. cig-zoff-ray-overn [CR]

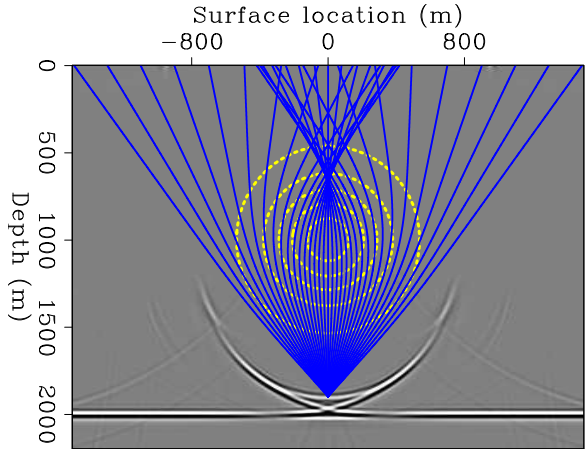


Figure 6.10: Image obtained by applying the SSR-migration to the zero-offset section shown in Figure 6.8a. The event smiling above the image of the flat reflector is an imaging artifact. Superimposed onto the seismic image is a contour representation of the velocity anomaly (light lines) and a ray fan (dark lines) traced starting from the bottom of the artifact. cig-tripl-image-overn [CR,M]

offset section; the event smiling above the image of the flat reflector is the imaging artifact caused by the event with $\gamma \neq 0$ identified above. Superimposed onto the seismic image there is a contour representation of the velocity anomaly (light lines) and a ray fan (dark lines) traced starting from the bottom of the artifact. Figure 6.11 shows a window of the zero-offset section. The line superimposed onto the data is the traveltime curve corresponding to the ray fan shown in Figure 6.10. This traveltime curve is the summation operator that a common-angle migration (with $\gamma = 0$) would use to image a point diffractor located at the bottom of the artifact. The later branch of the traveltime curve, which corresponds to the rays going straight through the anomaly, is tangent to the event with $\gamma \neq 0$. Consequently, this event is constructively integrated by the migration operator and is imaged as the bottom of the smiling artifact. Similarly, the upward branches of the spurious reflector are imaged by summation operators that constructively integrate over the event with $\gamma \neq 0$.

Further insight into the origin of the artifact can be gained by examining the wavefield at depth. Figure 6.12 shows the zero-offset section after downward continuation by the SSR operator to a depth of 1,000 meters; that is, the depth of the center of the velocity anomaly. At this depth the triplication caused by the anomaly has disappeared, and the event with $\gamma \neq 0$ has assumed the shape of an anticline. In contrast, Figure 6.13 shows the wavefield at a depth

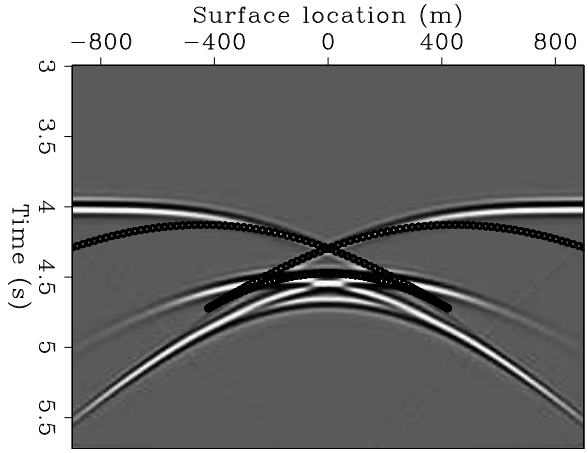


Figure 6.11: Window of the zero-offset section shown in Figure 6.8a. The line superimposed onto the data is the traveltime curve corresponding to the ray fan shown in Figure 6.10. This traveltime curve is the effective summation operator of a common-angle migration at $\gamma = 0$. cig-tripl-sum-overn [CR,M]

of 1,000 meters after full prestack downward-continuation (i.e. using the DSR operator) is applied to the whole data set. In this case the event with $\gamma \neq 0$ has correctly moved away from the zero-offset section. Direct comparison of the zero-offset sections (Figure 6.12 and Figure 6.13a) provides a clear example of the limitations of imaging only a subset of the whole prestack data in the presence of complex overburden.

Brandsberg-Dahl et al. (2003) have proposed a method for removing this kind of artifacts when producing ADCIGs with a "Kirchhoff-like" migration. Their method is based on dip-limiting the migration integral; they thus dub their methodology as "focusing in dip". Whereas the method is effective, it also requires an a priori knowledge of the dip spectrum of the image. This assumption is not needed when the whole wavefield is used to produce ADCIGs, as is the case for the wavefield-continuation methods discussed in the next section.

6.2 Angle-Domain Common Image Gathers by wavefield-continuation

Computing surface-offset CIGs using wavefield-continuation methods is challenging, since the information on the surface offset is "lost" during the wave propagation. Fortunately, the information on the propagation directions of both the incident and reflected waves is available from the wavefield at depth, and thus ADCIGs can be easily computed using wavefield-continuation migration.

This chapter presents two methods for computing ADCIGs in conjunction with wavefield-continuation migration. Both methods rely on a decomposition of the wavefield into plane-wave components; this decomposition is performed by slant stacks (Schultz and Claerbout, 1978) along the subsurface offset axes (see the definition of subsurface offset axes in Chapter 4). The main difference between the two methods is in whether the plane-wave decomposition is performed before imaging or after imaging. I first present the method based on an angle decomposition before imaging (de Bruin et al., 1990; Prucha et al., 1999), because it has the most direct physical interpretation. The physical interpretation of the ADCIGs computed after imaging (Sava and Fomel, 2003; Rickett and Sava, 2002; Biondi and Shan, 2002) follows directly from the interpretation of the ADCIGs computed before imaging.

In the presence of multipathing and complex velocity, either of these wavefield-continuation methods produces better images, and consequently better ADCIGs, than Kirchhoff migrations do, as we discussed at the beginning of Chapter 4. Furthermore, ADCIGs computed by wavefield-continuation are immune to the artifacts that I discussed in the previous section, which affect CIGs computed by Kirchhoff migration. These artifacts are generated by imaging subsets of the whole data set independently, as is usually done during Kirchhoff migration. In contrast, when using wavefield-continuation migration methods, ADCIGs are computed by simultaneously using all contributions from the whole recorded data set. On the other hand, the need to include contributions from the whole data set may add complexity to the data-handling during the computations for some of wavefield-continuation migration methods (e.g. shot-profile migration).

When the geologic dips are steep, the computation of ADCIGs using the horizontal sub-

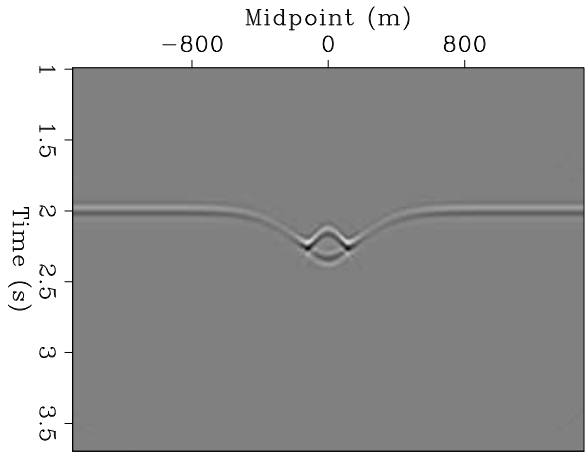


Figure 6.12: The zero-offset section after downward continuation by the SSR operator to a depth of 1,000 meters. At this depth, the event with $\gamma \neq 0$ has assumed the shape of an anticline. This event migrates into the smiling artifact shown in Figure 6.10. cig-zoff-overn-z1000-art [CR,M]

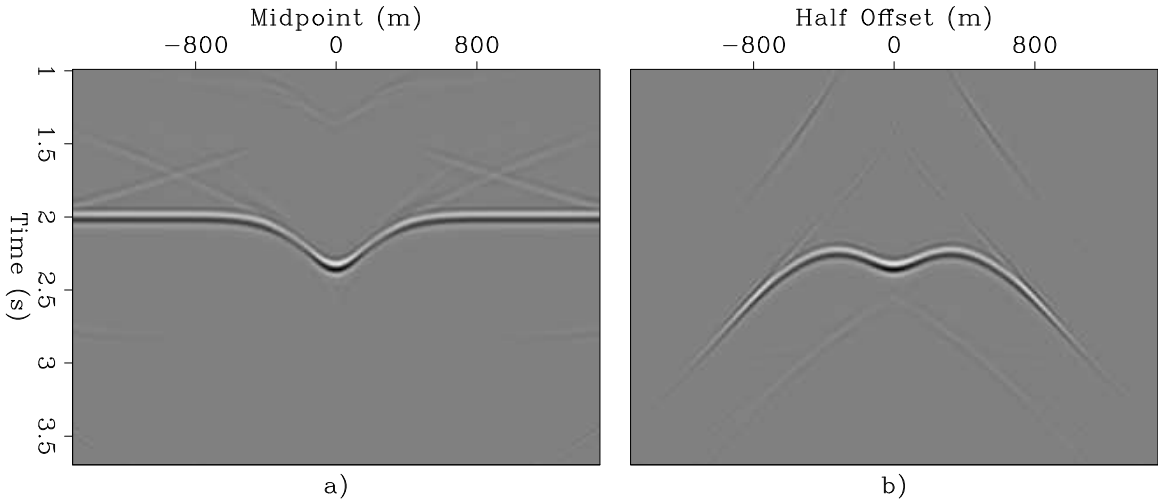


Figure 6.13: The wavefield at a depth of 1,000 meters after full prestack downward-continuation (i.e. using the DSR operator) is applied to the whole data set. In this case, the event with $\gamma \neq 0$ has correctly moved away from the zero-offset section. Propagation through half of the anomaly has greatly simplified the wavefield compared with the wavefield recorded at the surface, which is shown in Figure 6.8 cig-zoff-cmp-overn-z1000 [CR,M]

surface offset does not produce useful gathers; the horizontal-offset CIGs degenerate, and their focusing at zero offset blurs (Biondi and Shan, 2002). Therefore, in the presence of complex structure, we may need to compute CIGs as a function of the vertical subsurface offset. A detailed analysis of this case is beyond the scope of this chapter, but the interested reader can find it in Biondi and Symes (2003).

6.2.1 ADCIG before imaging – ADCIG$\left(p_{x_h}, p_{y_h}\right)$

A 3-D prestack wavefield can be decomposed according to the offset ray parameters (p_{x_h}, p_{y_h}) by applying slant stacks along the offset axes. These slant stacks produce plane-wave components called **offset plane waves**. If the velocity function is horizontally invariant, each offset plane wave can be accurately propagated and imaged independently from the others. Each prestack partial image produced by this procedure corresponds to one offset ray parameter, and thus ADCIGs functions of the offset ray parameters [ADCIG$\left(p_{x_h}, p_{y_h}\right)$] can be extracted from the whole prestack image cube. The offset ray parameters measured at the surface are functions of the surface take-off angles of the source and receiver rays, and are obviously linked with the reflection opening angle γ and the reflection azimuth ϕ.

Ottolini and Claerbout (1984) presented a procedure for obtaining angle-dependent images based on the slant-stack of 2-D CMP gathers at the surface and on the propagation of offset plane waves. Mosher et al. (1997) generalized the 2-D procedure to 3-D marine data and derived a more accurate downward-continuation method for the offset plane waves. However, there is a limit on the accuracy of a migration based on an offset plane-wave decomposition at the surface. When the velocity function is laterally varying, the source and receiver wavefields are refracted differently, according to the local velocities. Consequently, the independent propagation of each offset plane wave is inaccurate when strong lateral velocity variations occur. In these situations, the idea of decomposing the wavefield into offset plane waves is still useful, but it must be generalized; the prestack wavefield must be slant-stacked recurrently at each depth level, instead of only once at the surface.

Once we have introduced the concept of decomposing the prestack wavefield at depth, the derivation of the procedure for computing ADCIGs before imaging using source-receiver downward continuation is straightforward (Prucha et al., 1999). As we have seen in Section 4.2.2, when source-receiver downward continuation is used, the whole prestack wavefield is readily available at every depth level by downward continuing the recorded data using the DSR. The image is then obtained by extracting the values of the wavefield at zero time. The usual migration process can be schematically represented as follows:

$$P(\omega, \mathbf{m}, \mathbf{h}; z=0) \stackrel{\text{DSR}}{\Longrightarrow} P(\omega, \mathbf{m}, \mathbf{h}; z) \tag{6.10}$$

$$P(\omega, \mathbf{m}, \mathbf{h}; z) \stackrel{Imaging}{\Longrightarrow} P(t=0, \mathbf{m}, \mathbf{h}; z). \tag{6.11}$$

To compute ADCIGs, we need to introduce a slant-stack transformation between downward continuation [step (6.10)] and imaging [step (6.11)]. The results of slant stacks are functions of the offset ray parameter $\mathbf{p_h}$ and the intercept time τ. The image is then obtained by extracting

the values of the wavefield at zero intercept time; i.e. $\tau = 0$. The migration process used to produce ADCIGs can be schematically represented as follows:

$$P(\omega, \mathbf{m}, \mathbf{h}; z = 0) \stackrel{\text{DSR}}{\Longrightarrow} P(\omega, \mathbf{m}, \mathbf{h}; z) \tag{6.12}$$

$$P(\omega, \mathbf{m}, \mathbf{h}; z) \stackrel{\textit{Slant stack}}{\Longrightarrow} P(\tau, \mathbf{m}, \mathbf{p_h}; z) \tag{6.13}$$

$$P(\tau, \mathbf{m}, \mathbf{p_h}; z) \stackrel{\textit{Imaging}}{\Longrightarrow} P(\tau = 0, \mathbf{m}, \mathbf{p_h}; z). \tag{6.14}$$

Figure 6.14 illustrates the subsequent stages of the process represented in (6.12–6.14). The synthetic data set used to generate this figure is the same one I used in Section 4.1 to illustrate the principles of reverse time migration. The data were generated assuming two reflectors (one dipping and one flat) immersed in a constant-velocity medium.

The downward-continuation process [step (6.10) or step (6.12)] focuses the wavefield toward zero offset; if the continuation velocity is correct, a migrated image can be obtained by extracting the value of the wavefield at zero offset and zero time. Figure 6.14a shows the zero-offset section of the downward-continued wavefield at the depth of the flat reflectors; that is, at $z = 700$ meters. As expected, the flat reflector is well imaged at zero time. Figure 6.14b shows the wavefield at the same depth as in Figure 6.14a, but as a function of offset and at fixed midpoint ($x_m = 1,410$ meters). The flat reflector is well focused at zero offset and zero time, whereas the energy corresponding to the dipping reflector has already been defocused and thus it appears as a "time-reversed" hyperbola at negative times. The wavefield shown in Figure 6.14b is decomposed into plane waves by performing slant stacks for each midpoint location [step (6.13)], and the results of these slant stacks are imaged at zero time [step (6.14)]. The downward-continuation, slant stacks, and imaging steps are repeated for all depth levels. Figure 6.14c shows the ADCIG at $x_m = 1,410$ meters, for all depths. Both reflectors are imaged as flat events in the ADCIGs, because at each depth level, slant stacks transform band-limited impulses located at zero offset (like the one shown in Figure 6.14b) into flat events.

Figure 6.15 shows the effects of velocity errors on ADCIGs. It shows the same panels as Figure 6.14 [i.e. wavefield at zero-offset (a), wavefield at fixed midpoint (b), and ADCIG (c)], but with a migration velocity 10% higher than the correct one. At the correct depth of the flat reflector ($z = 700$ meters), the energy for both reflectors has started defocusing and it forms time-reversed hyperbolas. Most of the energy for the flat reflector has not reached the zero-time line yet, causing the flat reflector to be imaged deeper than the correct depth. In the ADCIG, both events frown downward, indicating a too high migration velocity. The residual moveout caused by velocity errors in ADCIGs is thus qualitatively similar to the moveout observed in ODCIGs (Figure 6.3). If the velocity function is too low, the reflections will smile upward; if the velocity function is too high, the reflections will frown downward. Section 11.2.1 presents a quantitative analysis of the residual moveout in ADCIGs, with the purpose of using the information contained in the ADCIGs to estimate the migration velocity, based on the theory developed by Biondi and Symes (2003).

Computing ADCIGs before imaging is more difficult when using shot-profile migration (Section 4.2.1) than when using source-receiver migration, because the prestack wavefield at depth is not easily available as a function of the subsurface offset. In theory, it could be com-

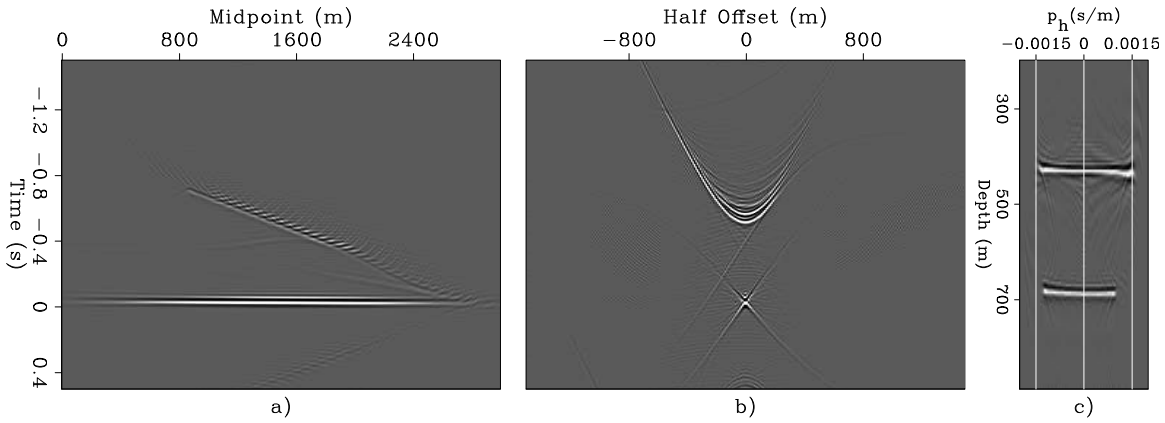


Figure 6.14: Illustration of the method for computing ADCIGs before imaging. Orthogonal slices of the prestack wavefield after downward continuation with the correct velocity to the depth of 700 meters; that is, the depth of the flat reflector: zero-offset section (panel a), and the common-midpoint gather at 1,410 meters (panel b). Panel c) shows the complete (i.e. for all depths) ADCIG at 1,410 meters. cig-zoff-cmp-cig-dip-overn-z700 [CR,M]

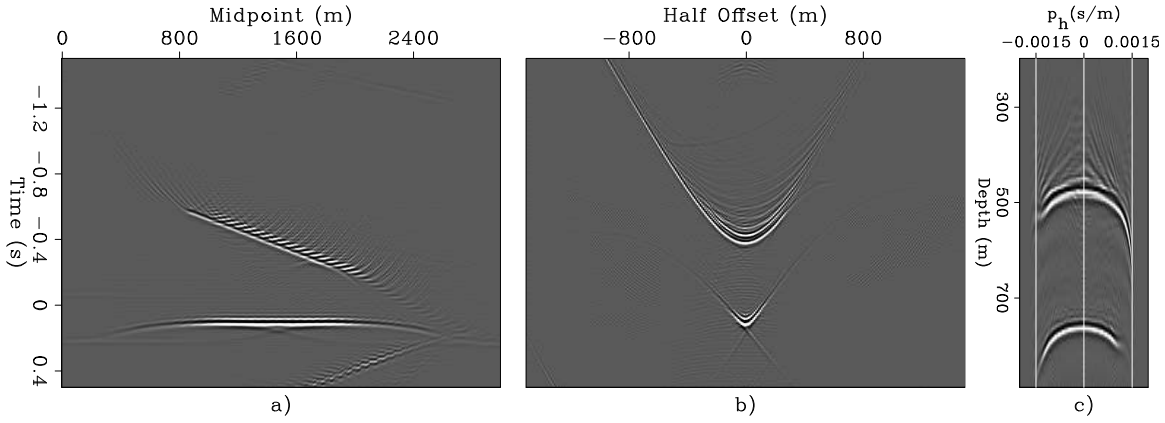


Figure 6.15: Illustration of the sensitivity of ADCIGs to velocity errors. Orthogonal slices of the prestack wavefield after downward continuation with a velocity 10% higher than the correct one to the depth of 700 meters: zero-offset section (panel a), and the common-midpoint gather at 1,410 meters (panel b). Panel c) shows the complete (i.e. for all depths) ADCIG at 1,410 meters. cig-zoff-cmp-cig-dip-overn-z700-fast [CR,M]

puted by appropriate correlations (or deconvolutions) of the source and receiver wavefields (de Bruin et al., 1990). This procedure generates ADCIGs equivalent to those obtained by source-receiver migration, as can be demonstrated by a generalization of the equivalence principle for the prestack images that I presented in Section 4.2.3. In practice, the computation of the ADCIGs before imaging for shot-profile migration is computationally demanding, because it requires either decomposing the wavefield for each shot independently and accumulating the prestack images for all shots (computationally inefficient), or accumulating the contributions of each shot to the whole prestack wavefield before decomposing it (storage and I/O inefficient). Therefore, ADCIGs are rarely computed before imaging when using shot-profile migration. In Section 6.2.2 we will see how computing ADCIGs after imaging addresses this problem.

Geometric interpretation of 2-D ADCIG$\left(p_{x_h}\right)$

Regardless of whether they have been computed using source-receiver migration or shot-profile migration, ADCIGs computed before imaging have the same geometrical interpretation. Strictly speaking, they are functions of the offset ray parameters (p_{x_h}, p_{y_h}) and not of the reflection opening angle γ and the reflection azimuth ϕ. However, in 2-D, the in-line offset ray parameter p_{x_h} is easily related to the reflection opening angle γ and the geological dip angle α_x (as defined in Figure 6.4) by a simple trigonometric relationship.

According to equation (6.9), the offset ray parameter can be expressed as a function of the source and receiver ray parameters (p_{x_s}, p_{x_g}) as follows:

$$p_{x_h} = p_{x_g} - p_{x_s}. \tag{6.15}$$

The source and receiver ray parameters (p_{x_s}, p_{x_g}) are functions of the source and receiver ray propagation angles (β_s, β_r) and of the velocity at the reflector $v(z,x)$ as follows:

$$p_{x_s} = \frac{\sin\beta_s}{v(z,x)} \quad \text{and} \quad p_{x_g} = \frac{\sin\beta_r}{v(z,x)}. \tag{6.16}$$

Substituting the relationships in (6.16) into equation (6.15), then using the angular relationship expressed in equation (6.2) and applying fundamental trigonometric identities, we obtain the desired relationship:

$$p_{x_h} = \frac{\sin\beta_r - \sin\beta_s}{v(z,x)} = \frac{2\sin\left(\frac{\beta_r-\beta_s}{2}\right)\cos\left(\frac{\beta_r+\beta_s}{2}\right)}{v(z,x)} = \frac{2\sin\gamma\cos\alpha_x}{v(z,x)}, \tag{6.17}$$

which directly links the offset ray parameter p_{x_h} to the reflection opening angle γ. The mapping of the p_{x_h} axis into the γ axis depends on both the local velocity and the geological dip. When the spatial velocity variations are significant and the geological structure is complex, unraveling this dependence might be a challenge because of the well known difficulties in reliably measuring rapidly changing local dips.

Figure 6.16 shows an example of a 2-D ADCIG and compares it with the ODCIGs extracted at the same surface location. The ODCIG was obtained by an amplitude-preserving

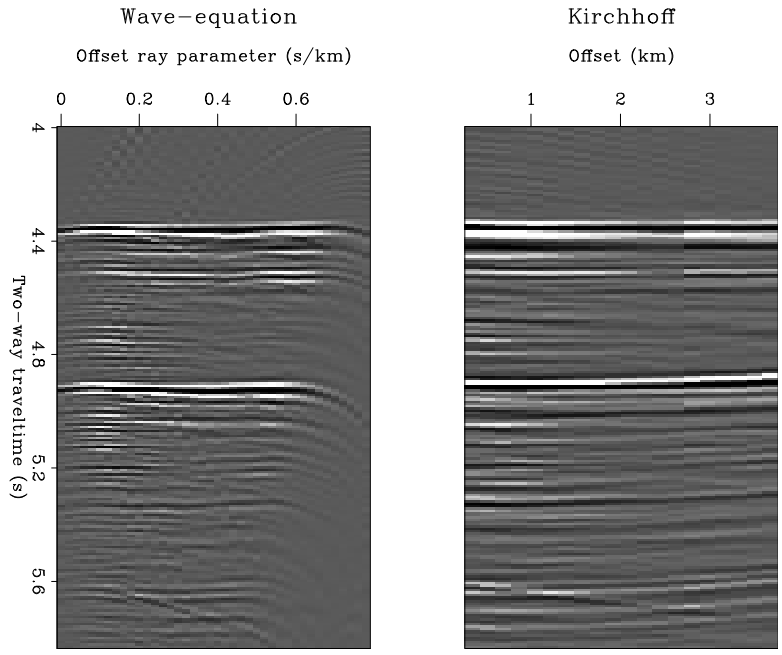


Figure 6.16: Left: ADCIG obtained by wavefield-continuation migration. Right: ODCIG gather obtained by Kirchhoff migration. (Right panel is from Ecker et al. (1996)). cig-AVO-hydrate-angle-kir [CR]

Kirchhoff migration (Ecker et al., 1996). Both the amplitudes and kinematics (including a slight upward smile) of the image function of the offset ray parameter (left panel) are in qualitative agreement with the image function of offset (right panel). However there are some important differences in the appearance of the two gathers. The range of well-illuminated offset ray parameters p_{x_h} decreases with depth (two-way traveltime), because a fixed maximum recording offset at the surface illuminates a decreasing range of reflection opening angles at depth. Also, notice the amplitude discontinuity clearly visible in the ODCIG around 1.8 km offset. This discontinuity was caused by a change in receiver-spacing in the recording cable. In the ADCIG, this amplitude discontinuity has been smoothed over a Fresnel zone.

Geometric interpretation of 3-D ADCIG$\left(p_{x_h}, p_{y_h}\right)$

The relationships in equation (6.17) are also valid in 3-D when the reflection azimuth ϕ is equal to zero. But in the general 3-D case, we need to add another relationship linking the cross-line offset ray parameter p_{y_h} to p_{x_h} and the angles γ and ϕ at the reflection point. To derive this relationship, we impose the constraint that the source and the receiver rays must become coplanar before meeting at the reflection point. I will thus refer to this relationship as the **coplanarity condition**.

The coplanarity of the two rays also simplifies the 3-D geometry of ADCIGs and enables us to generalize easily to 3-D the 2-D geometric interpretation. The schematic in Figure 6.17 is the 3-D generalization of the 2-D schematic shown in Figure 6.4. The plane of coplanarity

in 3-D corresponds to the vertical plane in 2-D. I define the reflection azimuth angle ϕ as the angle formed by the in-line axis x with the line defined by the intersection of the coplanar plane with a constant depth plane. After rotation by ϕ, the horizontal coordinates x and y become x' and y'. Once rotated by ϕ, the coplanar plane is tilted with respect to the vertical by the cross-line dip angle $\alpha_{y'}$. All the angles in Figure 6.4 that are formed between ray (plane wave) directions and the true vertical axis z ($\beta_{s'}$, $\beta_{r'}$, and $\alpha_{x'}$) are now relative to the tilted vertical axis z'; these angles have the same meaning on the tilted plane as they have on the vertical plane in 2-D. Furthermore, the horizontal ray parameters ($\mathbf{p_{s'}}, \mathbf{p_{g'}}$) are not affected by the tilt, since they depend on the angles formed by the rays with the x axis. However notice that the angles $\beta_{s'}$, $\beta_{r'}$, $\alpha_{x'}$, and $\alpha_{y'}$ are affected by the rotation of the horizontal axes by ϕ; I include the prime in the notation to indicate the angles after rotation.

Imposing the coplanarity of the source and receiver rays along a plane rotated by ϕ with respect to the original coordinate system provides us with the needed relationship that constraints the cross-line offset ray parameter $p_{y'_h}$. For the azimuth of the coplanar plane to be ϕ, the "rotated" source and receiver ray-parameters $p_{x'_s}, p_{y'_s}, p_{x'_g}, p_{y'_g}$ must be related by the following expression:

$$\left(p_{y'_g} - p_{y'_s}\right) = \left(p_{y'_g} + p_{y'_s}\right) \frac{\sqrt{\frac{1}{v^2(\mathbf{g}',z)} - p_{x'_g}^2} - \sqrt{\frac{1}{v^2(\mathbf{s}',z)} - p_{x'_s}^2}}{\sqrt{\frac{1}{v^2(\mathbf{g}',z)} - p_{x'_g}^2} + \sqrt{\frac{1}{v^2(\mathbf{s}',z)} - p_{x'_s}^2}}. \tag{6.18}$$

This relationship was derived by Biondi and Palacharla (1996) to define common-azimuth migration, and it is demonstrated in the Appendix of their paper.

So far we have interpreted the coplanarity condition in terms of source and receiver rays, but it has a similar interpretation in terms of plane waves. The phase vectors of two arbitrary plane waves are coplanar by construction, since they meet at the origin. The tilt angle $\alpha_{y'}$ and the azimuth ϕ of the plane of coplanarity for the incident and the reflected plane waves are implicitly determined by the coplanarity condition expressed in equation (6.18). The cross-correlation of the two plane waves does not define an imaging point but an imaging line that is orthogonal to the coplanar plane. The tilted plane shown in Figure 6.17 is an arbitrary plane parallel to the plane of coplanarity and the imaging point $\bar{\mathbf{I}}$ is the intersection of the imaging line with this plane. The source and receiver rays are defined as the lines parallel to the phase vectors of the corresponding plane waves and passing through the imaging point $\bar{\mathbf{I}}$.

Using the relationships among the ray parameters expressed in equations (6.9) and the relationships between the ray parameters and the angles expressed in equations (6.16), and after applying trigonometric identities, we can rewrite equation (6.18) as follows:

$$p_{y'_h} = \frac{\sin\alpha_{y'}(\cos\beta_r - \cos\beta_s)}{v(z',x',y')} = -\frac{\sin\alpha_{y'}\sin\gamma\sin\alpha_{x'}}{v(z',x',y')}. \tag{6.19}$$

The in-line offset ray parameter from equation (6.17) can also be rotated by the azimuth ϕ to give the following expression:

$$p_{x'_h} = \frac{2\sin\gamma\cos\alpha_{x'}}{v(z',x',y')}, \tag{6.20}$$

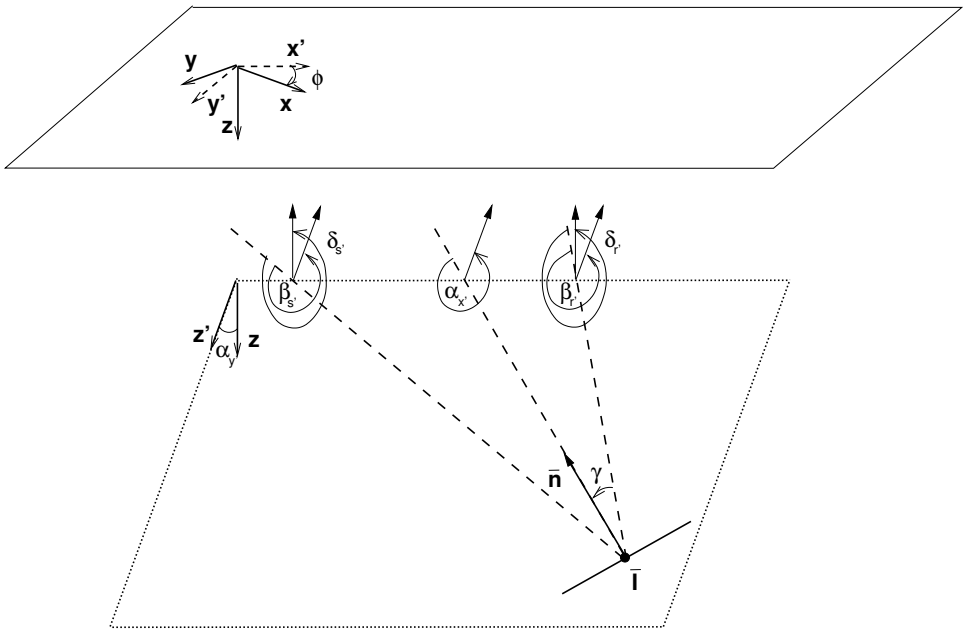


Figure 6.17: A schematic of the geometry of an angle-domain CIG gather in 3-D. The geometry is analogous to the 2-D case illustrated in Figure 6.4. In contrast with the schematic of Figure 6.4, the plane of coplanarity is not vertical but it is tilted by $\alpha_{y'}$ and rotated by ϕ with respect to the horizontal coordinates. Notice also that the angles $\delta_{s'}$ and $\delta_{r'}$ formed by the rays with the "true" vertical axis are different from the angles $\beta_{s'}$ and $\beta_{r'}$ formed by the rays with the tilted vertical axis. cig-cig-3d-v4 [NR]

which, together with equation (6.19), can be used to map the image from the offset ray parameter domain $\left(p_{x_h}, p_{y_h}\right)$ into the angle domain (γ, ϕ).

As for the 2-D case, the mapping of the ray parameters into angles depends on both the local velocity and the geological dips. For some applications this dependence does not cause difficulties. For example, in migration velocity analysis (Clapp and Biondi, 2000), there is no need to perform the mapping from ray parameters to angles, since ray parameters are adequate initial conditions for ray tracing. For other applications, the reflection angles are needed, and thus the mapping must be performed. When this mapping is difficult, it might be convenient to use the ADCIGs computed after imaging, which are presented in the next section. This kind of ADCIG circumvents the need to estimate local dips by transforming the image directly into angles.

6.2.2 ADCIG after imaging – ADCIG(γ, ϕ)

The computation of ADCIGs before imaging has a straightforward physical interpretation, and thus it was the first methodology to be developed for both and shot-profile migration (de Bruin et al., 1990) and source-receiver migration (Prucha et al., 1999). However, as we have discussed in the previous section, the computation of ADCIGs before imaging with shot-profile

migration can be either computationally or I/O demanding. Sava and Fomel (2003) recognized that for source-receiver migration, the plane-wave decomposition can be performed after applying the imaging condition, as well as before. Rickett and Sava (2002) applied this insight to compute ADCIGs after shot-profile downward continuation and imaging with the generalized imaging condition expressed in equation (4.7). Their method is theoretically based on the equivalence of shot-profile and source-receiver migrations presented in Section 4.2.3; therefore, the ADCIGs computed after imaging with source-receiver migration and shot-profile migration are equivalent. Biondi and Shan (2002) have further extended their method to the computation of ADCIGs by reverse time migration after imaging with the generalized imaging condition expressed in equation (4.2). This last extension enables the computation of ADCIGs for overturned waves.

The computation of the ADCIGs after imaging is based on a plane-wave decomposition by slant-stacks along the offset axes, exactly as in the computation before imaging that we discussed above. The only difference between the two methods (before and after imaging) is that the dips along the offset axes are affected by the transformation from time to depth that is implicit in the imaging step. Therefore, the offset-dip parameters are linked to the reflection opening angle γ and azimuth ϕ differently than in the previous case. In the next two sections I will derive and discuss the analytical relationships between reflection angles and offset dips after imaging. I will start with the simpler 2-D case (Sava and Fomel, 2003), and then address the general 3-D case (Biondi et al., 2003).

ADCIG after imaging in 2-D – ADCIG(γ)

The application of the imaging condition transforms a wavefield propagating in time into an image cube that is function of depth. The transformation from time to depth depends on the local dips in the wavefield and the local propagation velocity. In the frequency-wavenumber domain this transformation is represented by the DSR operator, which in 2-D can be expressed as a function of the angles β_s and β_r as follows:

$$k_z = -\frac{\omega}{v(z,x)}(\cos\beta_r + \cos\beta_s). \tag{6.21}$$

Recalling the relationship between the in-line offset wavenumber k_{x_h} and the in-line offset ray parameter p_{x_h}:

$$k_{x_h} = p_{x_h}\omega, \tag{6.22}$$

and substituting both equation (6.21) and equation (6.17) in equation (6.22), we obtain the following relationship:

$$k_{x_h} = -p_{x_h}\frac{k_z v(z,x)}{\cos\beta_r + \cos\beta_s} = -\frac{2\sin\gamma\cos\alpha_x}{v(z,x)}\frac{k_z v(z,x)}{2\cos\gamma\cos\alpha_x} = -k_z\tan\gamma. \tag{6.23}$$

This relationship directly links the dips in the depth-offset domain $\left(k_{x_h}/k_z\right)$ to the aperture angle ($\tan\gamma$). Notice that both the local velocity $v(z,x)$ and the in-line dip angle α_x have been canceled from equation (6.23), so that in 2-D the offset to angle transformation can be

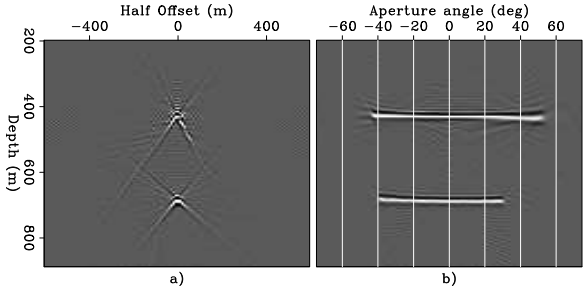


Figure 6.18: ODCIG (panel a) and ADCIG(γ) (panel b) after migration with the correct velocity. These CIGs are taken from the same data and at the same surface location as the ADCIG shown in Figure 6.14c. cig-off-ang-cig-dip-overn-dz5 [CR,M]

accurately performed in the wavenumber domain. Further, since it is independent from the geological dips, it can be performed on each ODCIG independently. However, it is important to notice that the absence of the local velocity $v(z,x)$ from the expression used to compute ADCIGs after imaging does not make the result independent from the local migration velocity. This dependence is indirect through the vertical wavenumber k_z; or, in other words, the estimate of γ depends on the apparent vertical wavelength of the imaged reflectors. Therefore, the advantages derived from the absence of $v(z,x)$ in equation (6.23) are purely computational. The sensitivity of the estimate of γ on the accuracy of the local velocity $v(z,x)$ is the same regardless of whether γ is estimated indirectly through p_{x_h} by using equation (6.17), or directly by using equation (6.23). This concept is well illustrated by the following example.

Figures 6.18–6.19 illustrate the computation of 2-D ADCIGs after imaging with a synthetic example. They are analogous to Figures 6.14–6.15, which illustrate the computation of 2-D ADCIGs before imaging. Figure 6.18a shows the ODCIG after migration with the correct velocity. The energy is well focused at zero offset for both the dipping and the flat reflectors. Slant stacks transform the impulses at zero offset into flat events in the angle domain (Figure 6.18b).

When the migration velocity is 10% higher than the correct one, the energy is defocused in the ODCIGs and the events have a hyperbolic moveout (Figure 6.19a); the apexes of the hyperbolas are deeper than the true reflectors' depths. Slant stacks transform the hyperbolas into frowns (Figure 6.19b) that are similar to the frowns shown in Figure 6.15c. However, there is a subtle but important difference between the two cases. The p_{x_h} range does not change between the ADCIG shown in Figure 6.14c and the one shown in Figure 6.15c, whereas the γ range

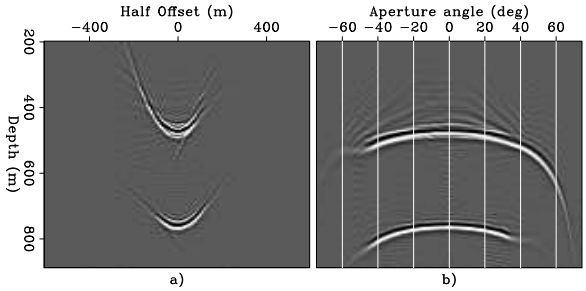


Figure 6.19: ODCIG (panel a) and ADCIG(γ) (panel b) after migration with a velocity 10% higher than the correct one. Notice that the range of γ increases from Figure 6.18b to Figure 6.19b. cig-off-ang-cig-dip-overn-fast-dz5 [CR,M]

increases when the velocity is too high (Figure 6.19b). This increase is due to the increase in the apparent vertical wavelength in the image, which correspondingly causes a decrease of k_z in equation (6.23). Or, from a different viewpoint, if we were to use equation (6.17) to map p_{x_h} into γ, the mapping would be affected by both the increase in $v(z,x)$ and the increase in apparent geological dip α_x – and the corresponding decrease in $\cos\alpha_x$.

The simple example above illustrates qualitatively the link between kinematics in the ADCIGs and migration velocity. The analysis of the prestack images in the angle-domain provides critical information on the correctness of the migration velocity function and, if needed, on how to improve it. Section 11.2 presents in details methods for extracting quantitative information on migration-velocity errors from migrated ADCIGs, and Sections 11.3 and 11.4 explain how to use this information to update the migration-velocity function.

ADCIG after imaging in 3-D – ADCIG(γ,ϕ)

To generalize the 2-D offset-to-angle transformation to 3-D, we use the coplanarity condition and the geometrical model shown in Figure 6.17. In this context, the 2-D DSR expressed in equation (6.21) describes upward-propagating waves on the tilted plane shown in Figure 6.17, instead of the vertical plane shown in Figure 6.4. Consequently, the vertical wavenumber in equation (6.21) is now the vertical wavenumber along the tilted vertical axis $k_{z'}$, and not the vertical wavenumber along the "true" vertical direction. This distinction is irrelevant in the case of ADCIGs computed before imaging because the computation is performed at each depth level independently, but it is required in this case because the slant stacks are performed in the depth domain.

If $\delta_{s'}$ and $\delta_{r'}$ are the angles that the source and receiver rays form with the "true" vertical direction (as indicated in Figure 6.17), simple trigonometry relates these angles to $\beta_{s'}$ and $\beta_{r'}$ through the tilt angle $\alpha_{y'}$ as follows: $\cos\delta_{s'} = \cos\beta_{s'}\cos\alpha_{y'}$, and $\cos\delta_{r'} = \cos\beta_{r'}\cos\alpha_{y'}$. The vertical wavenumber $k_{z'}$ is thus related to the vertical wavenumber k_z as follows: $k_{z'} = k_z/\cos\alpha_{y'} = \sqrt{k_z^2+k_{y'}^2}$. Substituting this relationship into equation (6.23) leads to its 3-D equivalent:

$$k_{x'_h} = -k_{z'}\tan\gamma = -\sqrt{k_z^2+k_{y'}^2}\tan\gamma. \tag{6.24}$$

This 3-D expression is not independent of the geological dips as was its 2-D equivalent [equation (6.23)]. In the presence of significant cross-line dips, it is thus important to use the correct 3-D expression in place of the approximate 2-D one. Figures 6.20–6.21 illustrate with a data example the effects of the cross-line dip correction in equation (6.24).

When deriving the after-imaging equivalent of equation (6.19), we need to take into account that $\alpha_{x'}$ is also measured along the tilted axis and thus that $\tan\alpha_{x'} = -k_{x'}/k_{z'}$. Using equation (6.19) and equation (6.24) we derive the following relationship, which expresses $k_{y'_h}$ as a function of the other wavenumbers in the image:

$$k_{y'_h} = -k_{y'}\tan\gamma\tan\alpha_{x'} = -\frac{k_{y'}k_{x'_h}k_{x'}}{{k_{z'}}^2} = -\frac{k_{y'}k_{x'_h}k_{x'}}{k_z^2+k_{y'}^2}. \tag{6.25}$$

The combination of equation (6.24) and equation (6.25) enables the computation of 3-D ADCIGs after imaging; using these two relationships, we can map the prestack image from the offset domain (x_h, y_h) to the angle domain (γ, ϕ). Notice that the reflection azimuth ϕ is implicitly included in these equations, since all the wavenumbers, with the exception of k_z, are dependent on ϕ. Therefore, during computations the wavenumbers need to be properly transformed by rotating the horizontal axes by the azimuth angle ϕ.

In contrast with the transformation described by equations (6.17–6.19), the transformation described by equations (6.24-6.25) is independent of the local velocity $v(z,x,y)$. Therefore there is no need to estimate the geological dips locally, but only globally. This estimate can be performed accurately and efficiently in the wavenumber domain.

There are two alternative computational algorithms to numerically perform the transformation described by equations (6.24-6.25); they differ in whether the computations are performed with the offset axes in the space or wavenumber domain. In either case, it is computationally effective to perform the computation in the wavenumber domain $\left(k_z, k_x, k_y\right)$ for the physical coordinates, because of the dependence of the mapping on the geological dips. If the ADCIGs are computed for many values of γ and ϕ, it is less expensive to perform the computation in the offset-wavenumber domain by a 3-D generalization of the 2-D radial-trace transform used by Ottolini and Claerbout (1984); Sava and Fomel (2003). However, this approach can generate artifacts, because the subsurface-offset axes are usually short, and the Fourier transforms have circular boundary conditions. The computation of the slant stack by integration in the offset-space domain avoids these artifacts and can be preferable when high-quality gathers are needed.

Figures 6.20–6.21 illustrate the error introduced by the use of the approximate 2-D relationship [equation (6.23)] to estimate the aperture angle γ when significant geological dips are present. Figure 6.20 shows a depth slice (z=580 meters) of the migrated image of the SEG-EAGE salt data set. The cross hair is centered at a horizontal location where the top of the salt dips at approximately 50 degrees in the cross-line direction and is flat along the in-line direction. Figure 6.21 shows the ADCIGs computed at the location marked by the cross hair. Figure 6.21a was computed using the 2-D relationship [equation (6.23)], whereas Figure 6.21b was computed using the 3-D relationship [equation (6.24)]. The aperture angle is overestimated in the gather on the left (apparent maximum aperture is about 60 degrees), and correctly estimated in the gather on the right (apparent maximum aperture is about 48 degrees). This error is consistent with the factor $\cos\alpha_{y'}$ that is neglected in the 2-D case. Notice that the bottom of the salt reflection ($z \simeq 2{,}100$ meters) is unchanged, because it is flat.

6.2.3 Examples of 3-D ADCIG(γ, ϕ)

ADCIGs are five-dimensional objects in 3-D, and thus it can be challenging to gain an intuitive understanding of their behavior. In this section I show 2-D slices of the 5-D space generated by computing 3-D ADCIGs from the migrated cube obtained from a simple synthetic data set. The data set contains five dipping planes, dipping at 0°, 15°, 30°, 45° and 60° toward increasing x and y. The azimuth of the planes is 45 degrees with respect to the direction of the

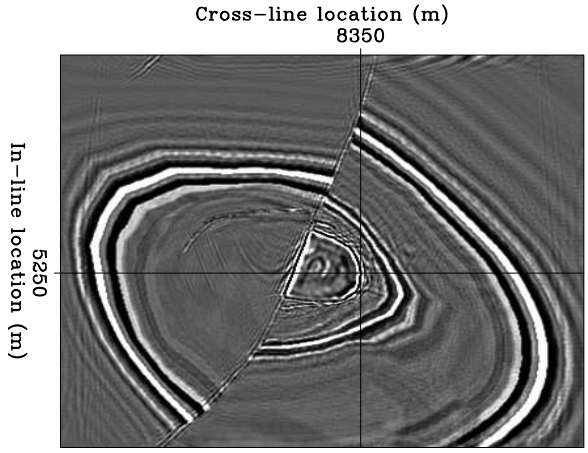


Figure 6.20: Depth slice (z=580 meters) of the migrated image of the SEG-EAGE salt data set. The cross hair is centered at a horizontal location where the top of the salt dips at approximately 50 degrees in the cross-line direction. cig-comp-2d-vs-3d-depth-slice [CR]

acquisition. The velocity is $V(z) = (1,500 + 500z)$ m/s, which roughly corresponds to typical gradients found in the Gulf of Mexico. The acquisition geometry had a single azimuth oriented along the x axis, and the maximum source-receiver offset was 3,000 meters. Figure 6.22 shows the geometry of the reflectors. The data were imaged with a full source-receiver 3-D prestack migration.

Because of the velocity gradient and the oblique azimuthal orientation, the azimuths of the reflections are not equal to the azimuth of the acquisition ($\phi = 0$). The reflection azimuths are within the range of $0^\circ \leq \phi \leq 15^\circ$ and depend on the reflector dip and on the aperture angle γ. The steeper the reflector dip and the wider the aperture angle are, the larger is the azimuth rotation at the reflection point.

All the following figures show slices of the ADCIGs at one fixed horizontal location with $x = y = 450$ meters; that is, they show slices through the 3-D image cube described as ADCIG(z, γ, ϕ). The most familiar of these slices display the image as a function of the depth (z) and the aperture angle (γ). Figure 6.23 shows two of these ADCIGs, for two different reflection azimuths: $\phi = 10$ degrees (panel a) and $\phi = 15$ degrees (panel b). The reflections from the deepest – and steepest – reflector ($z \simeq 1{,}430$ meters) are well focused within the range delimited by these two azimuths. In contrast, the reflections from the other two reflectors (dipping at 30 and 45 degrees), are not well focused at these azimuths, and thus they frown downward even if the migration velocity is correct.

Figure 6.24 shows a slice taken at the constant depth of $z = 1{,}430$ meters; this depth corresponds to the deepest reflector. The reflection amplitudes are thus shown as functions of both the aperture angle (γ) and the reflection azimuth (ϕ). Because of the poor azimuthal resolution close to normal incidence, the azimuthal range is wide for small γ; it narrows around $\phi = 15$ degrees as γ increases.

The increasing azimuthal resolution with aperture angle is clearly demonstrated in Figure 6.25. The three panels in Figure 6.25 display the image as a function of depth (z) and reflection azimuth (ϕ), and at constant aperture angle. The aperture angles are: a) γ=4 degrees, b) γ=20 degrees, and c) γ=30 degrees. The curvature of the reflectors as a function of the azimuth increases with increasing aperture angle, indicating that the azimuthal resolution

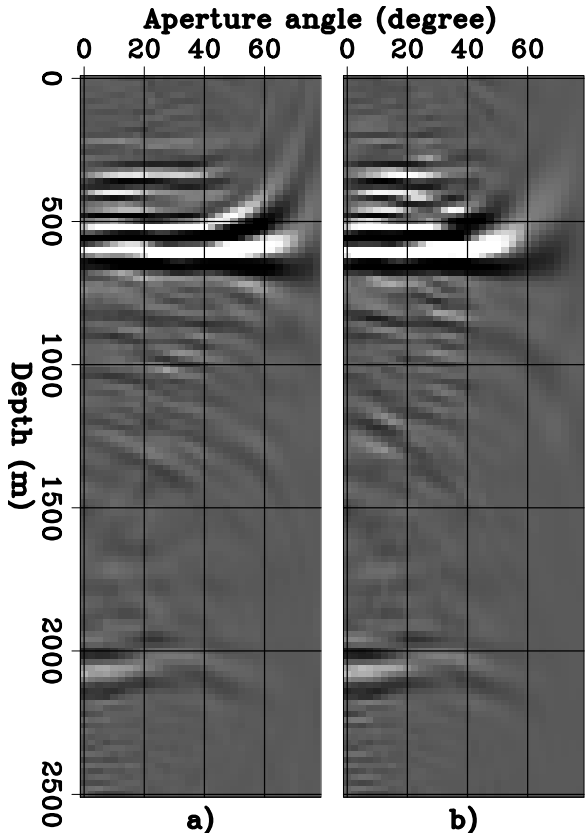


Figure 6.21: ADCIG computed using the approximate 2-D relationship (panel a), and ADCIG computed using the correct 3-D relationship (panel b). The aperture angle γ of the top of the salt reflection ($z \simeq 600$ meters) is overestimated in panel (a), whereas the bottom of the salt reflection ($z \simeq 2{,}100$ meters) is the same in the two panels. cig-comp-2d-vs-3d-adcigs [CR]

increases as the aperture angle widens. In other words, the common-azimuth data "illuminates" all the reflection azimuths for narrow aperture angles, but "illuminates" only a narrow range of reflection azimuths at wide aperture angles.

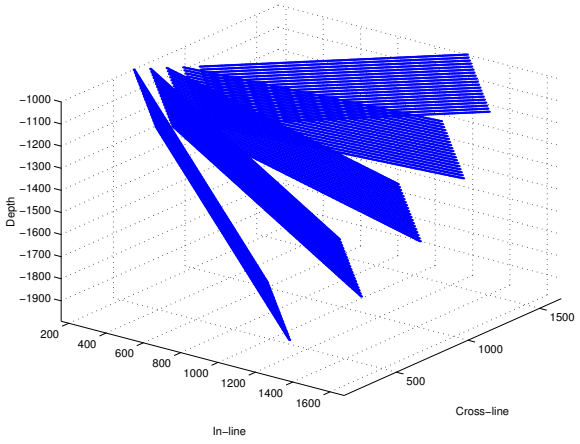


Figure 6.22: Reflectors' geometry for the synthetic data set used to illustrated 3-D ADCIGs. The reflectors are slanted planes, dipping at 0°, 15°, 30°, 45° and 60° toward increasing x and y; they are oriented with an azimuth of 45° with respect to the in-line direction. cig-planes [NR]

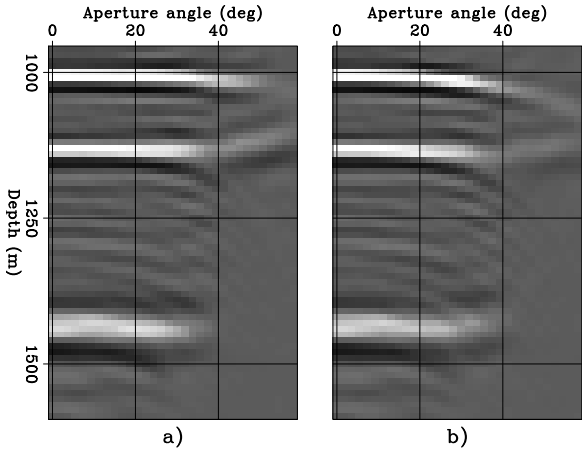


Figure 6.23: ADCIGs as functions of depth (z) and aperture angle (γ) for two different reflection azimuths and at constant horizontal location ($x = y = 450$ meters): $\phi = 10^\circ$ (a), and $\phi = 15^\circ$ (b). cig-cig-1-data8 [CR]

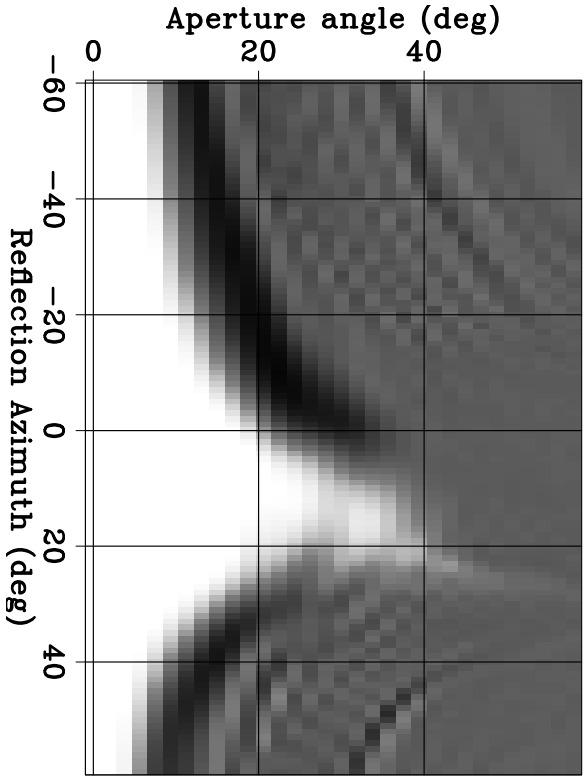


Figure 6.24: ADCIG as a function of aperture angle (γ) and reflection azimuth (ϕ) at constant depth ($z = 1{,}430$ meters) and horizontal location ($x = y = 450$ meters). cig-zaz-60-60-dense-all-v4-data8 [CR]

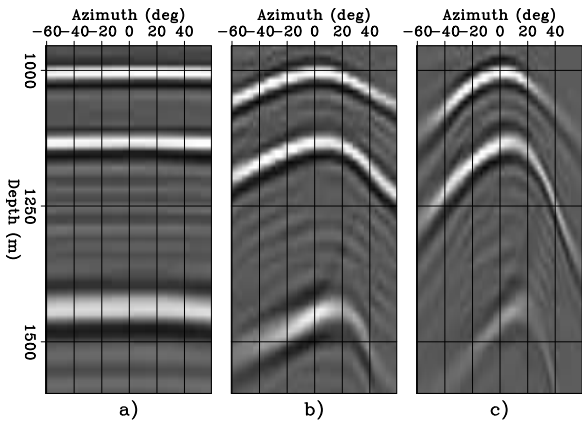


Figure 6.25: ADCIGs as functions of depth (z) and reflection azimuth (ϕ) for three different aperture angles and at constant horizontal location ($x = y = 450$ meters): $\gamma = 4^\circ$ (a), $\gamma = 20^\circ$ (b), and $\gamma = 30^\circ$ (c). cig-azim-gamma-all-data8 [CR]

REFERENCES

Biondi, B., and Palacharla, G., 1996, 3-D prestack migration of common-azimuth data: Geophysics, **61**, 1822–1832.

Biondi, B., and Shan, G., 2002, Prestack imaging of overturned reflections by reverse time migration: 72nd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 1284–1287.

Biondi, B., and Symes, W. W., 2003, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics: submitted for publication.

Biondi, B., Tisserant, T., and Symes, W. W., 2003, Wavefield-continuation angle-domain common-image gathers for migration velocity analysis: 73rd Ann. Internat. Meeting, Soc. of Expl. Geophys., 2104–2107.

Brandsberg-Dahl, S., de Hoop, M. V., and Ursin, B., 2003, Focusing in dip and AVA compensation on scattering-angle/azimuth common image gathers: Geophysics, **68**, 232–254.

Claerbout, J. F., 1985, Imaging the Earth's Interior: Blackwell Scientific Publications.

Clapp, R., and Biondi, B., 2000, Tau domain migration velocity analysis using angle CRP gathers and geologic constraints: 70th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 926–929.

de Bruin, C., Wapenaar, C., and Berkhout, A., 1990, Angle-dependent reflectivity by means of prestack migration: Geophysics, **55**, 1223–1234.

Ecker, C., Dvorkin, J., and Nur, A., 1996, Sediments with gas hydrates: Internal structure from seismic AVO: 67th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1767–1770.

Fomel, S., and Prucha, M., 1999, Angle-gather time migration: SEP–**100**, 141–150.

Mosher, C. C., Foster, D. J., and Hassanzadeh, S., 1997, Common angle imaging with offset plane waves: 67th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1379–1382.

Ottolini, R., and Claerbout, J. F., 1984, The migration of common-midpoint slant stacks: Geophysics, **49**, 237–249.

Prucha, M., Biondi, B., and Symes, W. W., 1999, Angle-domain common-image gathers by wave-equation migration: 69th Ann. Internat. Meeting, Soc. Expl. Geophys., Expanded Abstracts, 824–827.

Rickett, J., and Sava, P., 2002, Offset and angle-domain common image-point gathers for shot-profile migration: Geophysics, **67**, 883–889.

Sava, P., and Fomel, S., 2003, Angle-domain common-image gathers by wavefield continuation methods: Geophysics, **68**, 1065–1074.

Schultz, P. S., and Claerbout, J. F., 1978, Velocity estimation and downward-continuation by wavefront synthesis: Geophysics, **43**, 691–714.

Stolk, C. C., and Symes, W. W., 2003, Kinematic artifacts in prestack depth migration: Geophysics: accepted for publication.

Xu, S., Chauris, H., Lambare, G., and Noble, M. S., 1998, Common angle image gather: A new strategy for imaging complex media: 68th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1538–1541.

Xu, S., Chauris, H., Lambare, G., and Noble, M. S., 2001, Common-angle migration: A strategy for imaging complex media: Geophysics, **66**, 1877–1894.

Chapter 7

Efficient wavefield-continuation methods for prestack migration

Wavefield-continuation methods can be more accurate for 3-D prestack migration than methods based on the Kirchhoff integral. Their computational cost is the main hurdle that discourages the usage of wavefield methods in a majority of the imaging project. Wavefield-propagation algorithms are the most efficient when the computational grid is regular, and it has an horizontal extent similar to the horizontal coverage of the recorded data that are being propagated. The first, but not the most challenging, difficulty to overcome is that realistic 3-D data acquisition geometries are not regular (Chapter 1). The data geometry must be thus regularized before migration by applying a data-regularization algorithm, such as the methods described in Chapter 9, or methods that achieve similar results.

Wavefield-continuation methods can be extremely inefficient when there is a mismatch between the geometry of the recorded data and the geometry of the computational grid that the imaging algorithm requires. This problem is difficult to avoid when shot-gather migration is applied, as described in Chapter 4, in particular when the data are narrow-azimuth, such as marine streamer data. The sketch shown in Figure 7.1 illustrates the challenges faced by shot-gather migration when imaging deep and steeply dipping targets. It represents the geometry of the prestack imaging process for two targets, one shallow and the other deep. Each threesome of rays corresponds to the propagation paths for the events recorded at zero and maximum offset. The dashed boxes show the boundary of the computational domain required for imaging the events of interest. From the figure, it is apparent that for imaging the deep and steeply dipping event, a shot-gather migration would require a computational domain with horizontal extent much wider than the original maximum offset of the recorded data. In practice, the recorded shot gather must be padded all around with an enormous amount of zero traces. For example, if padding by a factor of four is needed on either side of the receiver array, and along both the in-line and the cross-line axes, the computational domain would be larger than the original shot gather by a factor of 81 ($(4+1+4)^2 = 81$). If computational cost are reduced by using a narrower computational domain, steeply dipping deep events might be lost.

Several solutions to this computational-efficiency challenge have been proposed. The

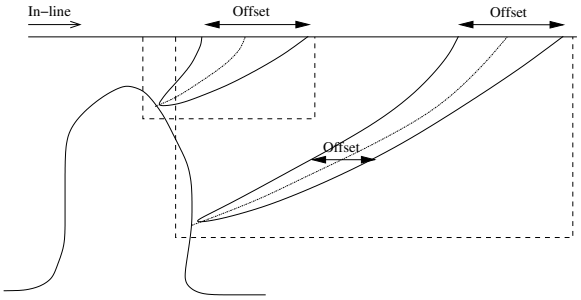


Figure 7.1: Schematic representing the ray-paths, projected along the in-line direction, for two prestack events recorded from two steeply-dipping reflectors, one shallow and the other deep. The dashed boxes show the boundaries of the minimum-sized computational domains required to image the events. The box for the deep events is substantially wider than the in-line offset, indicating potential computational inefficiencies when shot-gather migration is applied. comaz-in-line-shot [NR]

most of them fall in one of the following two categories: 1) variations of the source-receiver downward-continuation migration transformed into the midpoint-offset coordinates (Section 4.2.2), or 2) "synthesis" of common-source gathers with wide horizontal extent that are obtained by weighted sums of the shot gathers recorded in the field.

The algorithms that belong to the first category can be extremely efficient, in particular for marine streamer data, because they exploit the fact that the offset axes tend to shrink monotonically with depth. This phenomenon is illustrated by the sketch in Figure 7.1. Therefore, the offset axes at surface need to be only as wide as the data offset axes, and can be additionally trimmed down with depth. Furthermore, the cross-line offset of marine streamer data is narrow at the surface, as illustrated by the sketch in Figure 7.2, and thus the computational domain can be reduced further along the cross-line offset axis. At the limit, it can be reduced to just one sample, as in the common-azimuth algorithm presented in Section 7.2.2. In that section, I present a family of algorithms that achieve high efficiency by taking advantage of the narrow-azimuth nature of streamer data.

The main drawback of source-receiver migration is that it is not as flexible as shot-gather migration is. Its main limitation is that is intrinsically incapable of imaging overturned events, since for such events there is always at least one depth level at which the horizontal offset between the downward-continued sources and receivers is not defined. In other words, for overturned events either the source path or the receiver path overturns shallower than the other one, making its correct propagation impossible within a source-receiver migration framework, at least in its conventional definition presented in Section 4.2.2.

In contrast with the methods based on source-receiver migration, the algorithms based on the synthesis of common-source gathers enjoy the same flexibility of the original shot-gather migration. However, they do not provide as much computationally efficiency as the one belonging to the first category, because they cannot (at least in the form known at the moment of writing this book) fully exploit the narrowness of the offset axes (in particular of

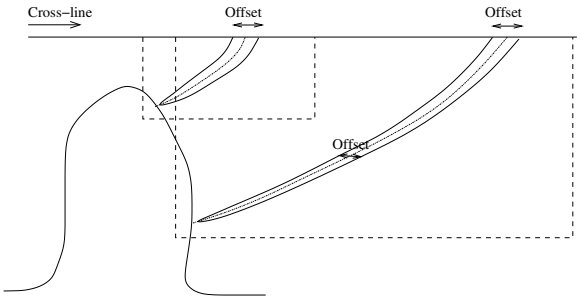


Figure 7.2: Schematic representing the geometry of the same two prestack events represented in Figure 7.1, but now projected along the cross-line direction. The cross-line offset is narrow at the surface because a marine streamer geometry is assumed. If a source-receiver migration transformed into midpoint-offset coordinates is applied to migrate the data, the computational domain can be much smaller than the one required by a shot-gather migration (dashed boxes). comaz-cross-line-shot [NR]

the cross-line axis) and the fact that the offset axes shrink with depth, as the source-receiver based algorithms do. Section 7.1 discusses the general principles underlying all the algorithms in this second category, and briefly demonstrate how they can be simply adapted to migrate overturned events by propagating the source and receiver wavefields along a tilted Cartesian grid.

7.1 Migration of synthesized common-source gathers

An effective method to tackle the computational inefficiency of shot-gather migration discussed above is to combine the data from several gathers recorded from point sources (shot gathers) and synthesize equivalent gathers from spatially-extended source. I will refer to these synthesized gathers as **common-source gathers**. The common-source gathers generated by this process have a wide horizontal extent, at the limit covering the whole survey area, for both the equivalent source function and the synthesized data. Imaging of these common-source gathers using a shot-gather migration method; that is, the independent propagation of the source and receiver wavefields [equations (4.5)–(4.6)] and imaging at depth by applying the imaging condition in equation (4.7), avoids some of the inefficiencies encountered when imaging point-source gathers.

The imaging of the combined common-source gathers is based on the linearity of the acoustic wave equation with respect to the propagating wavefield. Linearity is a common property of all approximations of the wave-equation that we commonly use, such as the one-way wave equation used for downward-continuation migration and the two-ways wave equation used for reverse-time migration. Because of linearity, the wavefield obtained at depth by propagating a common-source gather is the same as the wavefield that we would have obtained by independently propagating each recorded shot gather, and then combining the propagated

wavefields at depth. In the general case, the combination of wavefields generates "cross-talks" artifacts in the image; these artifacts are caused by spurious correlation of source and receiver wavefields that are related to different shot gathers. However, when a large number of combined common-source gathers are migrated, these cross-talks artifacts can be attenuated, and at the limit they vanish. The accuracy of the results obtained by imaging several common-source gathers depends on the method used for the combination (i.e. the choice of the algorithm to determine the summation weights), and the total number of the common-source gathers synthesized and imaged. Depending on the scheme used for determining the summation weights, the errors can be alternatively analyzed as caused by "cross-talks" or as caused by an incomplete (aliased) representation of the source functions.

The synthesis of the common-source gathers is performed by the following sum of N shot gathers weighted by the complex coefficients $w_{i,j}(\omega)$:

$$\widehat{P}^g(\omega,x,y;j)=\sum_{i=1}^{N} w_{i,j}(\omega)\,P^g(\omega,x,y;\mathbf{s}_i) \tag{7.1}$$

to create the j-th combined receiver common-source gather, $\widehat{P}^g(\omega,x,y;j)$. Similarly, the j-th combined-source function, $\widehat{P}^s(\omega,x,y;j)$, is created by the following weighted sums of N source functions:

$$\widehat{P}^s(\omega,x,y;j)=\sum_{i=1}^{N} w_{i,j}(\omega)\,P^s(\omega,x,y;\mathbf{s}_i)\,. \tag{7.2}$$

The weighted sums expressed in equations (7.1) and (7.2) can be repeated to create M common-source gathers and M combined-sources functions. In the following I will assume, for the sake of simplicity, that N is equal to the total number of shot gathers in the survey, but this assumption is not necessary. Equations (7.1) and (7.2) are in the temporal-frequency domain ω, but obviously the same procedure could be performed in the time domain if we substitute convolution along the time axis for the multiplication in the frequency domain.

The summation weights $w_{i,j}(\omega)$ in equations (7.1) and (7.2) are written as general functions of the temporal frequency ω, the original shot-gather index, i, and the combined common-source gather index, j. The choice of the functional form of these weights determine the name and the characteristics of the migration method. There are two main families of methods: the **phase-encoding migration** methods presented in the next section, and the **plane-wave migration** methods presented in Section 7.1.2.

7.1.1 Phase-encoding migration

The phase-encoding migration methods rely on the application of different phase shifts to the original shot gathers (i.e. phase encoding) to decrease the amount of cross-talks artifacts caused by migrating the combined gathers (Morton and Ober, 1998; Romero et al., 2000). The summation weights are phase-only complex numbers and are independent from the spatial coordinates; they can be expressed as follows:

$$w_{i,j}(\omega)=\frac{e^{i\phi_{i,j}(\omega)}}{\sqrt{M}}, \tag{7.3}$$

where $\phi_{i,j}(\omega)$ is a real phase function. Because the summation weights are independent from the spatial coordinates, their application to the wavefields commutes with the application of the propagation operators, and the prestack image cube that is obtained by applying shot-gather migration (either by downward continuation of by reverse-time migration) to the M common-source gathers can be written as:

$$
\begin{aligned}
& I(z,x,y,x_h,y_h) \\
&= \sum_{\omega}\sum_{j=1}^{M} \widehat{P}_z^g(\omega,x+x_h,y+y_h;j)\,\overline{\widehat{P}_z^s(\omega,x-x_h,y-y_h;j)} \qquad (7.4)\\
&= \sum_{\omega}\sum_{i=1}^{N}\sum_{j=1}^{M} \left|w_{i,j}(\omega)\right|^2 P_z^g(\omega,x+x_h,y+y_h;\mathbf{s}_i)\,\overline{P_z^s(\omega,x-x_h,y-y_h;\mathbf{s}_i)} \\
&+ \sum_{\omega}\sum_{k=1;k\neq l}^{N}\sum_{l=1}^{N}\sum_{j=1}^{M} w_{l,j}(\omega)\,\overline{w}_{k,j}(\omega)\,P_z^g(\omega,x+x_h,y+y_h;\mathbf{s}_l)\,\overline{P_z^s(\omega,x-x_h,y-y_h;\mathbf{s}_k)} \\
&= \sum_{\omega}\sum_{i=1}^{N} \widehat{w}_{i,i}(\omega)\,P_z^g(\omega,x+x_h,y+y_h;\mathbf{s}_i)\,\overline{P_z^s(\omega,x-x_h,y-y_h;\mathbf{s}_i)} \\
&+ \sum_{\omega}\sum_{k=1;k\neq l}^{N}\sum_{l=1}^{N} \widehat{w}_{l,k}(\omega)\,P_z^g(\omega,x+x_h,y+y_h;\mathbf{s}_l)\,\overline{P_z^s(\omega,x-x_h,y-y_h;\mathbf{s}_k)}, \qquad (7.5)
\end{aligned}
$$

where the subscript z indicates the wavefields propagated from the surface down to depth z, and where

$$\widehat{w}_{l,k}(\omega) = \sum_{j=1}^{M} w_{l,j}(\omega)\,\overline{w}_{k,j}(\omega), \qquad (7.6)$$

and consequently

$$\widehat{w}_{i,i}(\omega) = \sum_{j=1}^{M} \left|w_{i,j}(\omega)\right|^2. \qquad (7.7)$$

The image cube computed by equation (7.5) is equal to the image obtained by conventional shot-gather migration of the original shot-gathers [equation (4.7)] when the sum in equation (7.7) is equal to one, and when the cross-talk contributions vanish; that is, when the sum in equation (7.6) is equal to zero. The choice of the particular form for the weights expressed in equation (7.3) assures that the sum in equation (7.7) is always equal to one.

Several strategies have been proposed to minimize the "cross-talk" term. Because of the particular form of the weights, the sum in equation (7.6) can be re-written as follows:

$$\widehat{w}_{l,k}(\omega) = \frac{1}{M}\sum_{j=1}^{M} e^{\left(\phi_{l,j}(\omega)-\phi_{k,j}(\omega)\right)}. \qquad (7.8)$$

One way for assuring that the sum in equation (7.8) approaches zero as M increases is to chose the phase function $\phi_{i,j}(\omega)$ to be a random variable equally distributed in the interval

$0 - 2\pi$ (Morton and Ober, 1998; Bonomi and Cazzola, 1999; Romero et al., 2000). This choice for the weights is effective when the number, N, of shot gathers combined to create each common-source gather is large, but it is less effective when N is small. In this case deterministic algorithms for designing the weights can be more effective (Jing et al., 2000; Romero et al., 2000).

When random phase shifts are chosen for the summation weights, the cross-talk noise tends to be incoherent and zero-mean. It can be also shown that its total energy decreases as $1/\sqrt{M}$. One attractive application of random phase-encoding is the reduction of computational cost of the iterative migrations needed by Migration Velocity Analysis (MVA) procedures (Chapter 11). Many of the methods that we apply during MVA for extracting velocity information from the migrated images are based on curvature scans (e.g. semblance spectra function of ρ), which are robust to incoherent noise in the prestack image. Therefore, the application of phase encoding with small values of M is likely to be sufficient for MVA purposes.

Another intriguing application of random phase encoding is the reduction of computational cost of wavefield-based MVA methods (Chapter 12). In these methods, the result of cross-correlating two wavefields (e.g. "imaging" in shot-gather migration) is a velocity-updating function, which is usually smoothed by the introduction of a regularization term. We should be able to get accurate smooth velocity-updates by using a small value for M, because the cross-talk noise is incoherent and zero-mean and thus it should be greatly attenuated by the spatial smoothing of the velocity updates.

7.1.2 Plane-wave migration

The idea of imaging seismic data after decomposition in their plane-wave components has been introduced in the early days of seismic imaging (Schultz and Claerbout, 1978; Stoffa et al., 1981; Treitel et al., 1982; Temme, 1984). However, results of plane-waves depth migrations of real data have been published only more recently (Whitmore and Garing, 1993), and plane-waves migration methods have become popular thanks to their computational efficiency for wavefield-continuation migration of 3-D prestack marine data (Duquet et al., 2001; Notfors et al., 2003). Whitmore (1995) provides an excellent introduction to the basic concepts of plane-wave migration.

The decomposition of prestack data in plane-wave components according to the ray parameters at the source, $\left(p_{x_s}, p_{y_s}\right)$, can be seen as a particular case of phase encoding, when the resulting combined source functions are plane waves with different value of the ray parameters p_{x_s} and p_{y_s}, and the common-source gathers are computed by slant stacks of common-receiver gathers (Liu et al., 2002). To achieve this result, the phase function in equation (7.3) is expressed as follows:

$$\phi_{i,j}(\omega) = \omega\left[p_{x_s,j}\left(x_0 - x_{s,i}\right) + p_{y_s,j}\left(y_0 - y_{s,i}\right)\right], \tag{7.9}$$

where x_0 and y_0 are arbitrary horizontal coordinates that are the same for all the values of i and j.

Also in this case we can apply the quantitative analysis of the cross-talk noise developed in equations (7.4) and (7.5). The summation in equation (7.6) can be written as follows:

$$\sum_{j=1}^{M} w_{l,j}\,(\omega)\,\overline{w}_{k,j}\,(\omega) = \frac{1}{M}\sum_{j=1}^{M} e^{i\omega\left[p_{x_s,j}\left(x_{s,k}-x_{s,l}\right)+p_{y_s,j}\left(y_{s,k}-y_{s,l}\right)\right]}. \tag{7.10}$$

The sum in equation (7.10) vanishes if the ranges of the ray parameters p_{x_s} and p_{y_s} are: 1) wide enough with respect to the minimum distances between shots, $\left(x_{s,k}-x_{s,l}\right)$ and $\left(y_{s,k}-y_{s,l}\right)$, and 2) sampled densely enough to assure that the oscillatory argument of the summation over j in equation (7.10) is not aliased. Not surprisingly, the criterion for eliminating the cross-talk noise turns into a sampling criterion for a plane-wave representation of a function defined at the original shot locations of the survey. These sampling criteria are the ones usually applied when discussing the accuracy of plane-wave migration (Whitmore, 1995; Duquet et al., 2001; Notfors et al., 2003).

An interesting variations of the plane-wave migration that is cost-effective for 3-D marine streamer data is the **conical-wave migration** first discussed by Whitmore (1995), and first applied to 3-D data by Duquet et al. (2001). This method takes advantage of both the narrow-azimuth nature of marine data and of the large distance between parallel sail lines of typical 3-D marine surveys to achieve higher computational efficiency than a full 3-D plane-wave migration could achieve (Duquet et al., 2001; Notfors et al., 2003). Each sail line in a 3-D survey is assumed to be a common-azimuth line; that is, with zero cross-line offset. The 3-D acquisition geometries used in the field are not strictly common-azimuth, but we can transform the data into an ensemble of common-azimuth lines by a pre-processing step such as the AMO process presented in Section 3.3. The shot gathers in each common-azimuth line are then combined according to the in-line source ray parameter p_{x_s}, and imaged independently into a 3-D volume. The name of the method derives from the fact that the wavefield originated by the line-source functions for each p_{x_s}, propagates as a conical wave in a 3-D volume.

Imaging overturned waves by plane-wave migration in tilted coordinates

The computational advantages of plane-wave migration can be substantial when the inaccuracies caused by limiting the number of plane-wave components do not interfere with the final goals of the imaging project. However, the most promising application of plane-wave migration might be the migration of overturned events by one-way propagation operators. One-way extrapolation has many advantages over two-way propagation. It is computationally more efficient and less memory demanding because it can be easily performed in the temporal frequency domain and the depth-stepping length is not limited by stringent stability constraints as the time-stepping length is in explicit finite-difference two-ways solvers. Furthermore, when using one-way propagators, sharp velocity boundaries do not cause the artifacts that are created by reverse-time migration and that I discussed in Section 4.1. The main limitation of one-way extrapolators is that they cannot handle events that propagate along both directions of the extrapolation axis; such as overturned events.

Higginbotham et al. (1985) proposed a simple method for imaging overturned events reflected by a specific interface, such as a salt flank, by a zero-offset migration that uses a one-

way propagator. The idea is to tilt the propagation axis from the conventional depth direction to a direction that better matches the propagation of the events of interest. This idea is simple an powerful, but it is difficult to generalize to prestack migration when the overturned events have not a preferred propagation direction, as it happens in complex 3-D geology. The creation of common-source gathers identified by the take-off angle of the source wavefields at the surface achieves an approximate separation of the events according to their prevailing propagation direction, and thus enables the determination of an "optimal" tilt angle for imaging each plane wave component. Both the combined source function and the combined common-source gather can be effectively continued by one-way propagators advancing along the tilted direction defined by the take-off angle at the source (Etgen, 2003; Shan and Biondi, 2004), or even at slightly steeper angle when the velocity is known to increase monotonically with depth.

Figure 7.3 illustrates this idea for one plane wave component. It shows that both the source and the receiver legs of an overturned event have much better chance to be properly imaged by one-way propagation along an axis tilted in the same direction as the take-off angle of the source, than by conventional extrapolation along the depth axis. For events not reflected at normal-incidence, the propagation direction of the receiver wavefield does not coincide with the propagation direction of the source wavefield, but the relative "angular distance" between these two directions is approximately bound by the aperture angle at the reflection point, which is limited by the maximum offset of the recording geometry.

Shan and Biondi (2004) describe an implementation of this idea, and show that overturned events generated by vertical salt edge can be imaged by using a one-way propagator as well as by using a reverse time-migration algorithm. Furthermore, they applied the geometric understanding of the kinematics of angle-domain CIG of overturned events presented by Biondi and Shan (2002) and Biondi and Shan (2002) to define a simple algorithm for computing high-quality ADCIGs for the overturned events imaged using the **tilted-coordinates plane-wave migration** method. As discussed by Biondi and Shan (2002) and Biondi and Symes (2003), the quality of the ADCIGs obtained from horizontal-offset CIGs degrades progressively as the geological dip increases, and, at the limit, it degenerates when events overturn. The propagation in tilted coordinates enables a simple solution to this problem. The offset-domain CIGs are computed along the x' directions and then transformed to equivalent CIGs computed on both the x and z directions. The contributions from all the source plane-wave components are merged, and then transformed to to angle domain (Shan and Biondi, 2004).

Figures 7.4 and 7.5 show the quality of the stacked images and of the CIGs that can be obtained by applying a plane-wave migration in tilted coordinates. The data set is a 2-D line extracted from a 3-D marine survey recorded in the North Sea. A vertical salt edge generates overturned events that cannot be migrated by using downward-continuation migration. This data set is thus a good test bed for migration methods aiming to image overturned events.

Figure 7.4 compares the migration results for three different prestack migration methods. Figure 7.4a shows the image obtained by a conventional downward-continuation migration. Figure 7.4b shows the image obtained by a reverse-time migration (Biondi and Shan, 2002), and Figure 7.4c shows the image obtained by plane-wave migration with tilted coordinates.

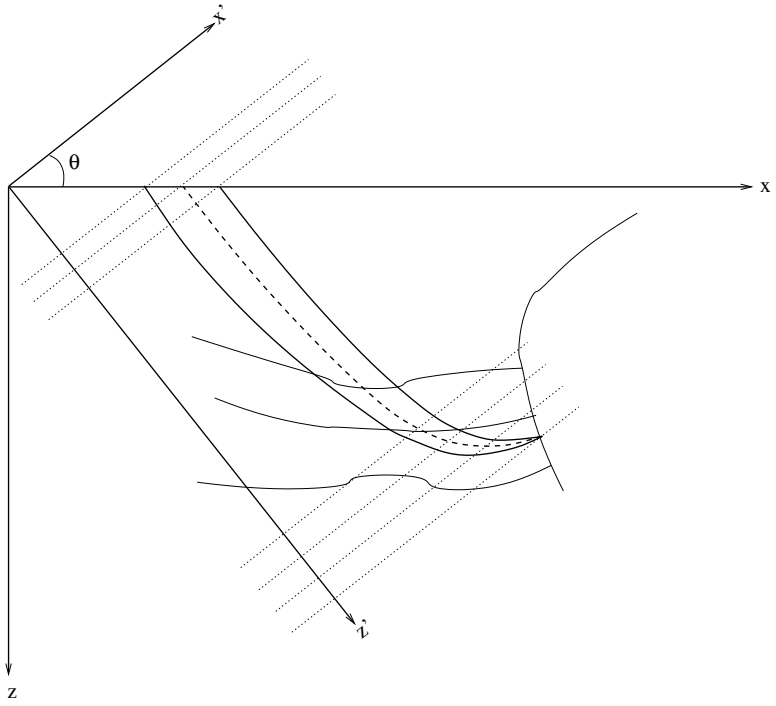


Figure 7.3: Schematic showing the geometry of an overturned event reflected from a salt flank. The event cannot be properly imaged by a downward-continuation method propagating the wavefields along the depth axis z. In contrast, it can be properly imaged by a a plane-wave migration propagating the wavefields along the z' propagation direction, which is rotated by the tilt angle θ with respect to z. The tilt angle θ can be different for different plane-wave components. The dotted lines represent the "horizontal" direction in the tilted coordinates; they help to visually verify that neither of the two legs of the event overturns with respect to the z' propagation direction. comaz-tilt-over [NR]

The fan of rays shown in Figure 7.4d is traced through the migration velocity model starting from the vertical salt-edge reflection. Notice that the reverse-time migration image (panel b) is sampled more coarsely than either of the other two images (panels a and c). The vertical salt edge is imaged equally well by the plane-wave migration (panel c) as by the reverse-time migration (panel b). Whereas the salt-edge reflectors are discontinuous and attenuated in the downward-continuation migration results (panel a), they are continuous and with strong amplitudes in the plane-wave migration (panel c).

Figure 7.5 compares the CIGs obtained by reverse-time migration with the CIGs obtained by plane-wave migration, extracted at a surface location, $x = 4,000$ meters, where the reflectors are only mildly dipping. Figure 7.5a and Figure 7.5b compare the offset-domain CIGs, whereas Figure 7.5c and Figure 7.5d compare the angle-domain CIGs. The two sets of CIGs are similar, though there are slight differences mostly due to the different spatial sampling between the images computed by reverse-time migration and the ones computed by plane-wave migration.

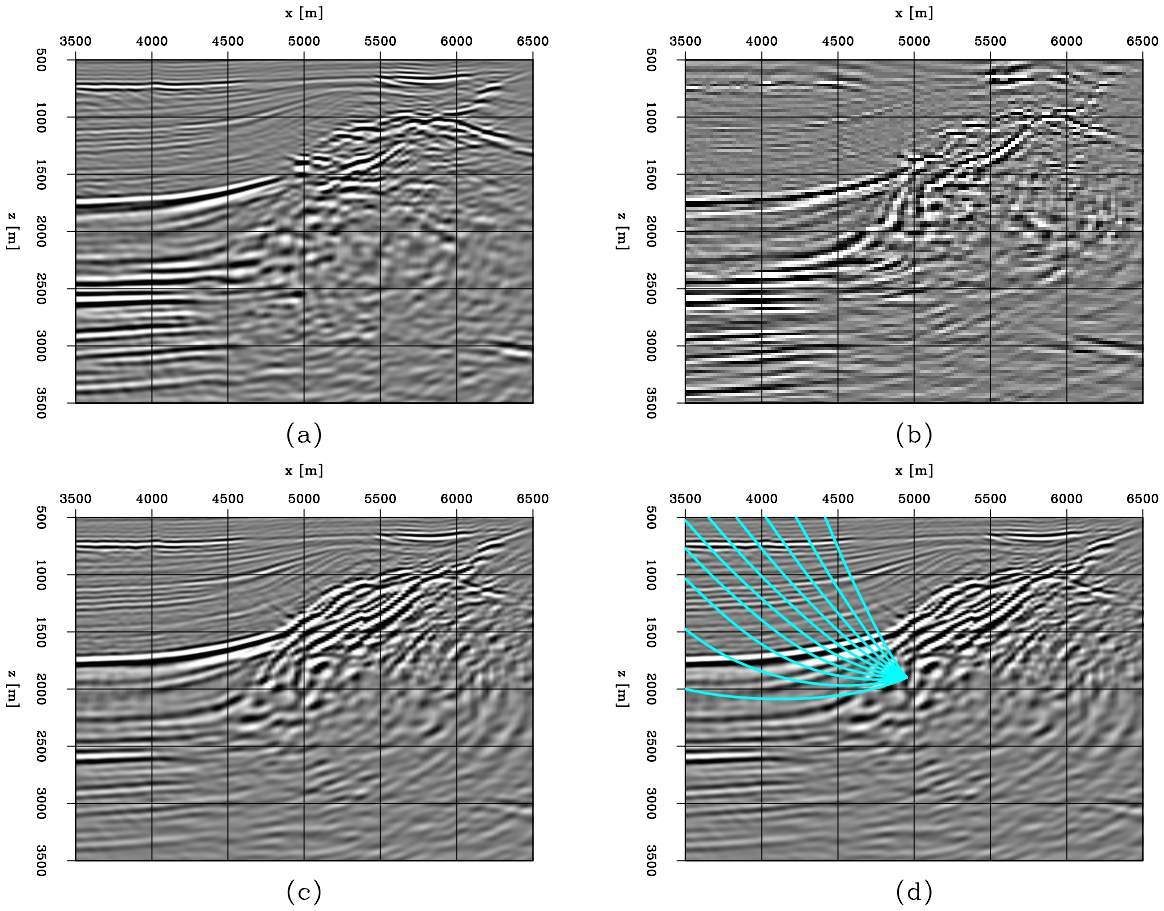
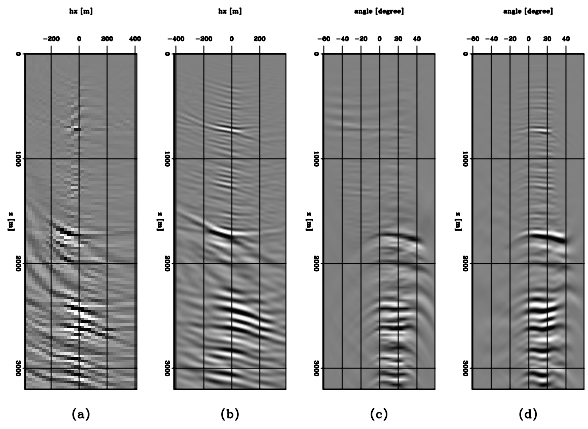


Figure 7.4: Migration of a North Sea real dataset in the proximity of a vertical salt edge: (a) the image from downward-continuation migration; (b) the image from reverse-time migration; (c) the image from plane-wave migration in tilted coordinates; (d) same image as in (c), with superimposed a fan of rays corresponding to overturned events reflected from the salt edge. (This figure is from Shan and Biondi (2004).) comaz-elf.image [NR]

Figure 7.5: CIGs extracted from the prestack images at $x = 4{,}000$ meters: (a) offset-domain CIGs obtained by reverse-time migration; (b) offset-domain CIGs obtained by plane-wave migration in tilted coordinates; (c) ADCIGs obtained by reverse-time migration; (d) ADCIGs obtained by plane-wave migration in tilted coordinates. (This figure is from Shan and Biondi (2004).) comaz-elf.cig [NR]



7.2 Source-receiver migration in midpoint-offset coordinates

The methods presented in the previous section achieve high computational efficiency when the number of synthesized common-source gathers, M, is substantially lower than the total number of shot gathers to be imaged (Cazzola et al., 2004); otherwise, the computational gain might be modest (Stork and Kapoor, 2004). Whether this condition is fulfilled depends on many parameters, such as the required image accuracy, the structure of the target, and the data-acquisition geometry.

Another effective way to reduce the computational cost of wavefield-continuation migration is to take advantage of the fact that, because of practical and economical limitations, the data are always acquired with a limited extent of the offset axes, as compared to the total horizontal extents of the survey (Figures 7.1–7.2). Furthermore, the most of the data-acquisition geometries on both land and the ocean have an anisotropic distribution of the offset vectors (Chapter 1), with the offset range substantially wider in one direction (i.e.; the in-line direction) than the other (i.e.; the cross-line direction).

Source-receiver migration transformed into midpoint-offset coordinates can take advantage of the limited offset ranges in a simple and straightforward way, because the computational domain can be easily restricted to encompass only the existing offsets. From Section 4.2.2, the whole prestack wavefield expressed in the frequency and wavenumber domain, $P_z(\omega, \mathbf{k_m}, \mathbf{k_h})$, can be downward-continued from depth z to depth $z + \Delta z$ by the following phase-shift:

$$P_{z+\Delta z}(\omega, \mathbf{k_m}, \mathbf{k_h}) = e^{-ik_z \Delta z} P_z(\omega, \mathbf{k_m}, \mathbf{k_h}), \tag{7.11}$$

where the vertical wavenumber k_z is evaluated using the following midpoint-offset wavenumber DSR equation:

$$\begin{aligned} k_z &= \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left[\left(k_{x_m} - k_{x_h}\right)^2 + \left(k_{y_m} - k_{y_h}\right)^2\right]} \\ &+ \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left[\left(k_{x_m} + k_{x_h}\right)^2 + \left(k_{y_m} + k_{y_h}\right)^2\right]}. \end{aligned} \tag{7.12}$$

The prestack image is computed by applying the imaging condition (4.12); that is, by evaluating the downward-propagated wavefield at zero time (i.e. summation over frequencies).

As demonstrated in Section 4.2.3, the prestack image obtained by source-receiver migration is equivalent to the one obtained by shot-gather migration. Therefore, I will refer to the migration algorithm defined by equations (7.11), (7.12), and (4.12) as **full-azimuth migration**. Full-azimuth migration can be fairly efficient, but there are two main practical obstacles for full-azimuth migration achieve absolute efficiency: 1) the downward continuation must be performed on 4-D data sets (constant-frequency slices) and thus require either a large amount of memory or out-of-core computations, and 2) computational efficiency demands that the cross-line offset range is as narrow as possible, but still to be sufficiently wide to "capture" all the useful propagation paths and to avoid boundary artifacts. Several variations of full-azimuth migration have been proposed for addressing these issues and improve computational

efficiency. Because of their practical importance, in particular for the imaging of 3-D marine data, I will dedicate the rest of this chapter to analyze these methods.

7.2.1 Offset plane-wave downward-continuation and migration

The **offset plane-wave migration** (Mosher et al., 1997) is the most efficient method among the variations of full-azimuth migration that we consider. This methods is a generalization to 3-D depth migration of the 2-D time migration methods originally proposed by Ottolini and Claerbout (1984). It is based on the decomposition of each CMP gather into offset plane waves by slant stacks as a function of the offset ray parameters p_{x_h} and p_{y_h}. After slant-stacking, each offset plane-wave component can be downward-continued independently by applying the following expression of the DSR operator:

$$\begin{aligned} k_z &= \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left[\left(k_{x_m} - \omega p_{x_h}\right)^2 + \left(k_{y_m} - \omega p_{y_h}\right)^2\right]} \\ &+ \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left[\left(k_{x_m} + \omega p_{x_h}\right)^2 + \left(k_{y_m} + \omega p_{y_h}\right)^2\right]}, \end{aligned} \tag{7.13}$$

that is obtained by substituting $k_{x_h} = \omega p_{x_h}$ and $k_{y_h} = \omega p_{y_h}$ in equation (7.12).

The independent imaging of the offset plane-wave components solves the practical challenges of full-azimuth migration discussed above; that is; the large-memory requirements and the boundary artifacts along narrow offset axes. The number of offset-plane waves needed to achieve the desired imaging accuracy is an important parameter influencing the computational efficiency. For narrow-azimuth data sets, the number of required offset plane waves is substantially lower than the number of source planes waves needed for the plane-waves migration presented in Section 7.1.2.

The price to pay for the efficiency gained with the independent downward-continuation of the offset plane waves is inaccuracy in depth migration with complex velocity models. The offset plane waves should actually mix during downward-continuation, to take into account the differences in ray bending between the source and the receiver legs that occur in media with laterally-varying velocity. Whereas in layered media the horizontal ray parameters, $\left(p_{x_s}, p_{y_s}, p_{x_g}, p_{y_g}\right)$, are constant with depth, the ray parameters change when later inhomogeneities are encountered, and consequently the offset ray parameters should be allowed to change as well. An alternative way for understanding the same issue is to notice that in laterally varying media the operators represented by the square roots in the DSR operator are not "diagonal" in the frequency-wavenumber domain; they become convolutional operators that mix the wavefield components corresponding to different wavenumbers; that is, they mix different offset plane waves.

7.2.2 Common-azimuth downward-continuation and migration

The vast majority of marine data sets acquired with streamers have a narrow cross-line offset range. The **common-azimuth migration** method takes advantage of this characteristic to reduce the dimensionality of the computational domain of full-azimuth migration (Biondi and Palacharla, 1996). It assumes that the all the data have zero cross-line offset; that is, all the traces in the data share the same azimuth, hence the common-azimuth name. By reducing the cross-line offset axis to only one sample, common-azimuth solves both issues of full-azimuth migration discussed above: the constant-frequency slices become smaller (3-D instead of 4-D), and the problem of boundary artifacts along the cross-line offset axis disappears.

Since the actual data are not common-azimuth, a pre-processing step is necessary to transform all the data into equivalent common-azimuth data (Section 3.3.3). In principle, this pre-processing step might introduce errors, but usually the amount of azimuth rotation required to convert real geometries into true common-azimuth geometries is very small, and consequently the errors introduced by the rotation are also small. As I will discuss later in this section, inaccuracies of common-azimuth images are less likely to be caused by the initial azimuthal rotation than by the approximation involved in forcing the downward-continued wavefield to be common-azimuth at every depth level.

The analytical derivation of the common-azimuth downward-continuation operator takes advantage explicitly of the reduced dimensionality of the data space so that it derives a continuation operator with only four dimensions. The general downward-continuation step of equation (7.11) can be rewritten as the following common-azimuth continuation:

$$
\begin{aligned}
&P_{z+\Delta z}\left(\omega, \mathbf{k_m}, k_{x_h}, y_h = 0\right) \\
&\quad = \int_{-\infty}^{+\infty} dk_{y_h} P_z\left(\omega, \mathbf{k_m}, k_{x_h}, y_h = 0\right) e^{-ik_z \Delta z} \\
&\quad = P_z\left(\omega, \mathbf{k_m}, k_{x_h}, y_h = 0\right) \left\{ \int_{-\infty}^{+\infty} dk_{y_h} e^{-ik_z \Delta z} \right\} \\
&\quad \approx P_z\left(\omega, \mathbf{k_m}, k_{x_h}, y_h = 0\right) A\left(\omega, \mathbf{k_m}, k_{x_h}\right) e^{-i\widehat{k_z} \Delta z},
\end{aligned}
\tag{7.14}
$$

where the integration over the cross-line offset wavenumber k_{y_h} evaluates the wavefield along the plane $y_h = 0$. Because the data is independent from k_{y_h} the integral can be pulled inside, and applied only to the operator itself and not to the data. The integral can be then approximated by a stationary-phase method (Bleistein, 1984); the result is a dispersion relation that does not depend on k_{y_h}, and that thus represents a four-dimensional operator instead of a five-dimensional one.

The expression for the common-azimuth operator is derived by substituting the "stationary path",

$$
\widehat{k}_{y_h} = k_{y_m} \frac{\sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left(k_{x_m} + k_{x_h}\right)^2} - \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left(k_{x_m} - k_{x_h}\right)^2}}{\sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left(k_{x_m} + k_{x_h}\right)^2} + \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left(k_{x_m} - k_{x_h}\right)^2}},
\tag{7.15}
$$

into the expression for the full DSR of equation (7.12); that is,

$$\begin{aligned}\widehat{k_z} &= \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left[\left(k_{x_m}-k_{x_h}\right)^2 + \left(k_{y_m}-\widehat{k}_{y_h}\right)^2\right]} \\ &+ \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left[\left(k_{x_m}+k_{x_h}\right)^2 + \left(k_{y_m}+\widehat{k}_{y_h}\right)^2\right]}. \end{aligned} \tag{7.16}$$

Common-azimuth migration can be accurately implemented with mixed-domain (ω-k/ω-x) downward-continuation methods, such as an extended split-step algorithm (Section 5.2), as long as the implementation takes properly into account the fact that the "stationary path" expressed in equation (7.15) is function of the interval velocities at the source and receiver locations.

The common-azimuth dispersion relation of equation (7.16) can be recast, after some algebraic manipulations, as a cascade of two dispersion relations. The first is that for 2-D prestack downward-continuation along the in-line direction,

$$k_{z_x} = \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{1}{4}\left(k_{x_m}-k_{x_h}\right)^2} + \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{1}{4}\left(k_{x_m}+k_{x_h}\right)^2}, \tag{7.17}$$

and the second is the one for 2-D zero-offset downward continuation along the cross-line axis,

$$\widehat{k_z} = \sqrt{k_{z_x}^2 - k_{y_m}^2}. \tag{7.18}$$

This in-line and cross-line separation of the common-azimuth dispersion relation is an alternative proof of the full-separability of the 3-D prestack migration operator derived in Chapter 3. If the propagation velocity in equation (7.17) and equation (7.18) is assumed to be constant, these dispersion relations can be directly used for cascading two 2-D Stolt migrations (Stolt, 1978).

The cost function of a mixed-domain implementation of common-azimuth migration is

$$\begin{aligned}&\text{PreCaz} \propto \kappa_{\text{PreCaz}} \\ &\quad \times \left(\text{Nz}_\xi \times \text{Nx}_\xi \times \text{Ny}_\xi \times \text{Nf} \times \log_2 \text{Nx}_\xi \times \log_2 \text{Ny}_\xi \times \log_2 \text{Nf}\right) \times (\text{Nt}). \end{aligned} \tag{7.19}$$

For full-volume imaging of deep targets this cost function is more attractive than the equivalent cost function for Kirchhoff migration [equation (2.12)], because it is proportional only to the square of the target depth ($\text{Nz}_\xi \times \text{Nt}$), as opposed to the cube.

Figures 7.6–7.7 demonstrate the effectiveness of common-azimuth migration to depth-image 3-D data in the presence of a complex velocity function. The data were recorded in the North Sea over a salt diapir with an overhang. The figures shows an in-line section extracted from migrated cube imaged by common-azimuth migration (Figure 7.6) and Kirchhoff migration (Figure 7.7). Notice that below the overhang the common-azimuth results are cleaner and the termination of the sediments against the salt are better imaged.

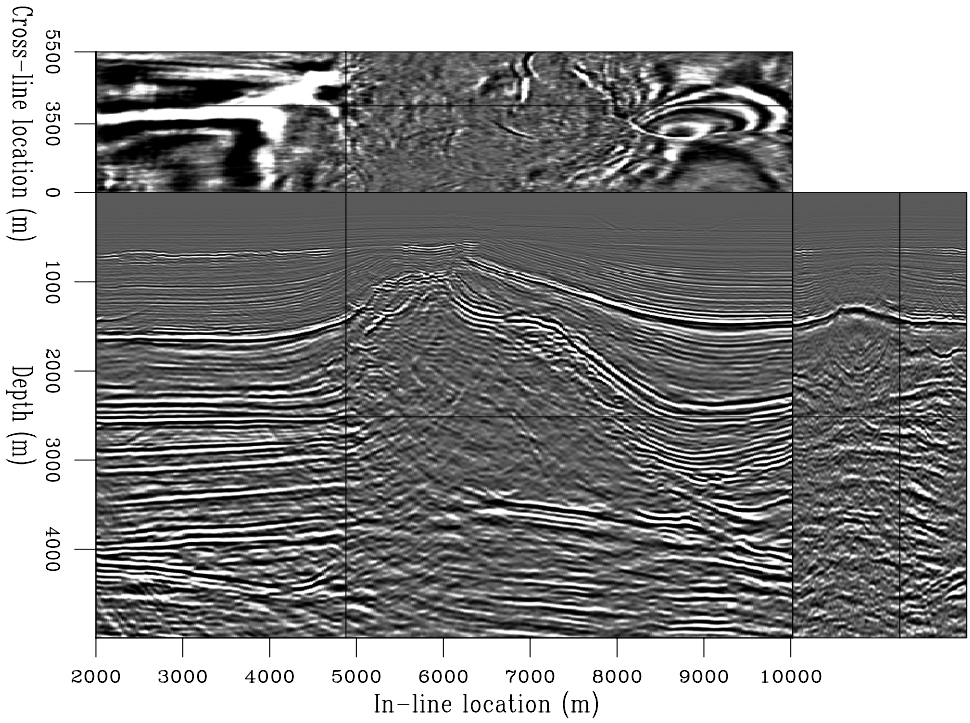


Figure 7.6: In-line section extracted from a migrated cube imaged by common-azimuth migration. (This figure is from Vaillant and Calandra (2000).) comaz-L7d-wave-120 [NR,M]

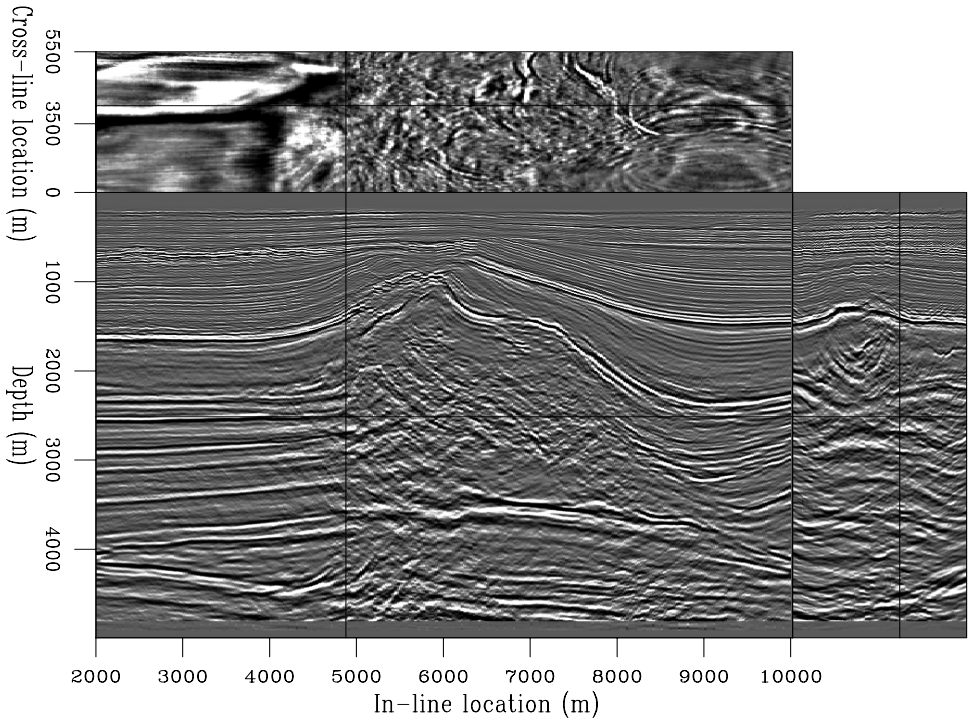


Figure 7.7: In-line section extracted from a migrated cube imaged by Kirchhoff migration. This section as the same cross-line coordinate as the section shown in Figure 7.6. (This figure is from Vaillant and Calandra (2000).) comaz-L7d-kir-120 [NR,M]

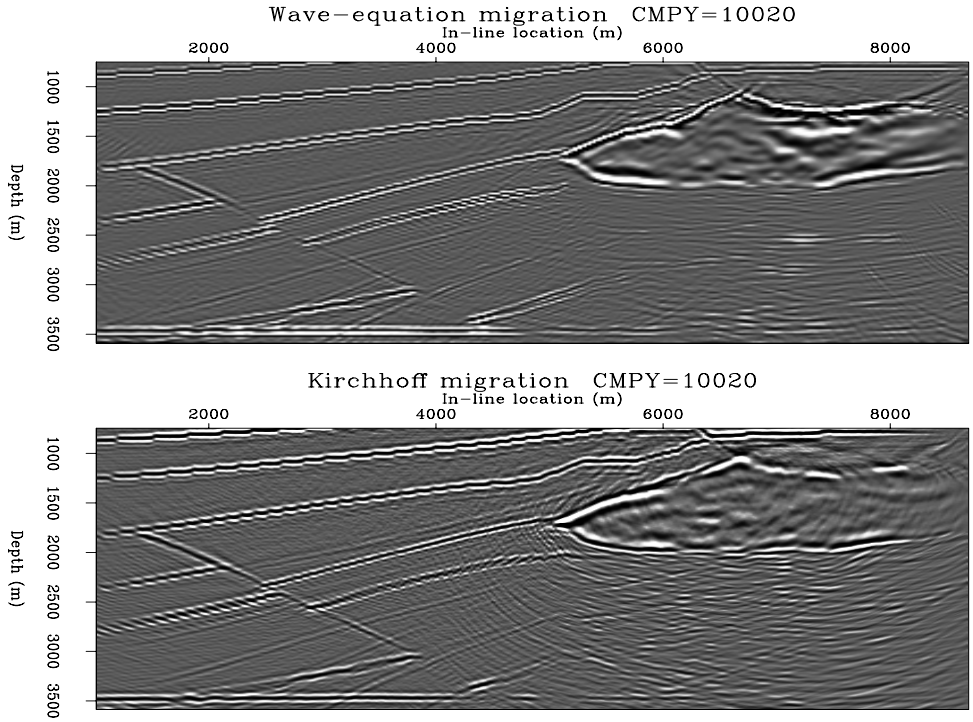


Figure 7.8: Common-azimuth migration (top) compared with Kirchhoff migration (bottom). (This figure is from Biondi (1999a).) comaz-Both-salt-under-y10020 [NR]

Figure 7.8 shows similar advantages for common-azimuth migration over Kirchhoff migration, but for the SEG-EAGE salt data set, for which the velocity model is known. Figure 7.8 compares two in-line sections. The structure under the salt is better imaged by common-azimuth migration.

Geometric interpretation of common-azimuth downward continuation

The common-azimuth downward-continuation operator, which was derived analytically by the stationary-phase method, has a straightforward geometric interpretation that relates the propagation directions of the rays of the continued wavefield. Biondi and Palacharla (1996) demonstrate in the Appendix of their paper that the expression for the stationary path of equation (7.15) is equivalent to the following relationship:

$$\frac{p_{y_s}}{p_{z_s}} = \frac{p_{y_g}}{p_{z_g}}, \tag{7.20}$$

among the ray parameters for the rays downward-propagating the sources, $(p_{x_s}, p_{y_s}, p_{z_s})$, and the ray parameters for the rays downward-propagating the receivers, $(p_{x_g}, p_{y_g}, p_{z_g})$. This relationship between the ray parameters constrains the direction of propagation of the source and receiver rays, for each possible pair of rays. In particular, the source ray and the receiver ray must lie on the same plane, with all the possible propagation planes sharing the

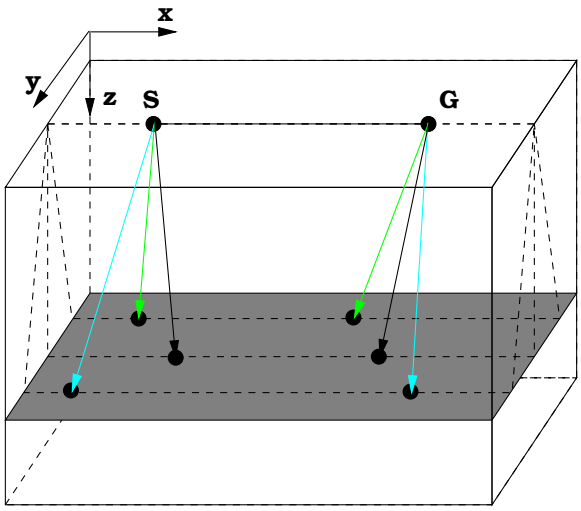


Figure 7.9: Geometry of the source and receiver rays for one depth step of common-azimuth downward continuation. For each pair of source ray and receiver ray, both rays are constrained to lie on the same slanted plane. All the propagation planes share the line connecting the source and the receiver locations. comaz-comaz-down [NR]

line that connects the source and receiver location at each depth level. This geometric relationship constrains the sources and receivers at the new depth level to be aligned along the same azimuth as the source and receivers at the preceding depth level; this constraint is consistent with the condition that we imposed when deriving analytically the common-azimuth downward-continuation operator [equation (7.14)].

Figure 7.9 is a graphical representation of the geometric interpretation of one depth step of a common-azimuth downward continuation. The source ray and the receiver ray must lie on any of the slanted planes that share the line connecting the source and receiver locations. Notice that this coplanarity constraint applies only to one depth step at the time. The propagation of any given event can occur on different planes at each depth step; that is, the tilt angles of the propagation planes can arbitrarily vary with depth.

Notice that in Section 6.2.1, I used this geometric interpretation of the relationship between the temporal frequency and the spatial wavenumbers defined by the stationary path (7.15) when discussing 3-D ADCIGs. I called this relationship the coplanarity condition, and used it to define the plane of coplanarity for events imaged by 3-D ADCIGs.

In contrast with common-azimuth downward continuation, the full-azimuth downward-continuation operator represented by the DSR equation propagates the source and the receiver rays in all directions. Figure 7.10 shows the ray geometry in the general case. The generality of this downward-continuation operator requires a full five-dimensional computational space, instead of the four-dimensional space required by common-azimuth migration.

Significance of the coplanarity condition

The coplanarity condition in equation (7.15) determines the cross-line offset dips of events as a function of their dips along the midpoint axes and the in-line offset axis. These dips are equal to zero only for events that are exactly dipping in either the in-line direction ($k_{y_m} = 0$) or the cross-line direction ($k_{x_m} = 0$ and consequently the numerator vanishes). For all other

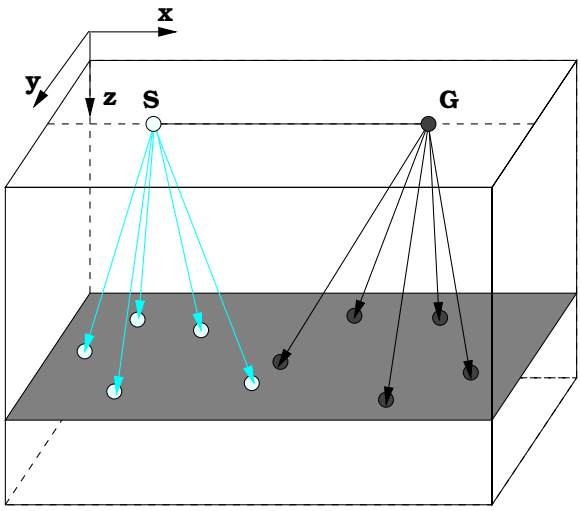


Figure 7.10: Geometry of the source and receiver rays for one depth step of a full-azimuth 3-D prestack downward continuation. The source and the receiver rays propagate in all directions. comaz-dsr-down [NR]

events $k_{y_h} \neq 0$, although $y_h = 0$. This is somewhat counter-intuitive and we could be tempted to approximate the value provided by the stationary path (7.15) with zero. If we used this approximation for $\widehat{k}_{y_h}$, the DSR operator in equation (7.16) becomes:

$$\overline{k_z} = \sqrt{\frac{\omega^2}{v^2(\mathbf{s},z)} - \frac{k_{y_m}^2}{4} - \frac{1}{4}\left(k_{x_m} - k_{x_h}\right)^2} + \sqrt{\frac{\omega^2}{v^2(\mathbf{g},z)} - \frac{k_{y_m}^2}{4} - \frac{1}{4}\left(k_{x_m} + k_{x_h}\right)^2}. \tag{7.21}$$

It is easy to verify that if we assume $v(\mathbf{s},z) \approx v(\mathbf{g},z) = v(\mathbf{m},z)$, the dispersion relation of equation (7.21) can be expressed as the cascade of a zero-offset downward continuation along the cross-line direction:

$$k_{z_y} = \sqrt{\frac{\omega^2}{v^2(\mathbf{m},z)} - \frac{k_{y_m}^2}{4}} \tag{7.22}$$

and the second is the one for 2-D prestack downward continuation along the in-line axis,

$$\overline{k_z} = \sqrt{k_{z_y}^2 - \frac{1}{4}\left(k_{x_m} - k_{x_h}\right)^2} + \sqrt{k_{z_y}^2 - \frac{1}{4}\left(k_{x_m} + k_{x_h}\right)^2}. \tag{7.23}$$

In constant velocity, the use of the dispersion relation of equation (7.21) is thus equivalent to a constant-velocity cross-line zero-offset migration, followed by a constant-velocity in-line prestack migration. The order between these migrations is thus reversed with respect to the correct order for two-pass migration derived in Section 3.4.

To analyze the implications of this order reversal I show the equivalent 3-D prestack Kirchhoff migration impulse response (spreading surface) obtained by applying two-pass migration in the wrong order, to be compared with the spreading surface shown in Figure 3.30 computed with the correct two-pass migration. Figure 7.11 compares the exact impulse response of 3-D prestack migration and the approximate impulse response corresponding to using the approximate DSR operator in equation (7.21). The gray surface shown in the left panel of Figure 7.11 is the exact spreading surface for an impulse recorded at 2.5 seconds, at an offset of 4,000 meters, and assuming a constant velocity of 2,500 m/s. The inner set of contour lines corresponds

to the exact summation surface, while the outer one corresponds to the surface defined by the approximation. The right panel of Figure 7.11 shows the same contour lines in plane view. The solid lines correspond to the exact spreading surface, while the dashed lines correspond to the approximate spreading surface. It is apparent that the approximation is worse for shallow reflectors dipping at 45 degrees with respect to the acquisition axes. This qualitative analysis is confirmed by the following numerical results obtained on the SEG-EAGE salt data set.

Figure 7.12 compares the cross-line sections cut through the images obtained by common-azimuth migration (panel a) and by using the approximate DSR operator in equation (7.21) (panel b). Both the right and left flank of the salt are better imaged by migration (a) than by migration (b). Migration (a) produces also a better image of the two normal faults.

Limitations of common-azimuth migration

Kinematics of common-azimuth migration are only approximately correct when the velocity varies. The errors are related to the departure of the reflected events wavepaths from the common-azimuth geometry. This phenomenon can be easily understood by analyzing the raypaths of reflections. Figure 7.13 shows an example of raypaths for an event bouncing off a reflector dipping at 60 degrees and oriented at 45 degrees with respect to the offset direction. The offset is equal to 2.9 km, and the velocity function is $V(z) = 1.5 + .5z$ km/s. The projections of the rays on the cross-line plane clearly show the raypaths departure from the common-azimuth geometry. Notice that the source ray (light gray) is close to overturn. A dipping reflector oriented at 45 degrees with rays close to overturn is the worst-case scenario for common-azimuth migration.

The event modeled with raytracing can also be imaged using raytracing by a simple process that I will identify as raytracing migration. Starting from the initial conditions at the surface given by modeling, both the shot and the receiver rays are traced downward until the sum of their traveltimes is equal to the traveltime of the reflected event. When raytracing migration is performed using the exact equation derived from an asymptotic approximation of the double-square root equation, the rays are exactly the same as the rays shown in Figure 7.13. On the contrary, if the common-azimuth approximation is introduced in the raytracing equations, the rays will follow the paths shown in Figure 7.14. As expected, the projections of the rays on the cross-line plane overlap perfectly, confirming that the rays follow a common-azimuth geometry. However, in Figure 7.14 it is also apparent that the common-azimuth rays do not meet at the ending points. This discrepancy in the kinematics causes errors in the migration.

To connect the kinematic analysis with migration errors, I migrated a data set with similar characteristics as the events analyzed above. The data set is the same synthetic common-azimuth data set used in Section 6.2.3 to illustrate the concept of 3-D ADCIGs. The reflectivity field consists of a set of five dipping planes, from zero dip to 60 degrees dip. The azimuth of the planes is 45 degrees with respect to the direction of the acquisition. The velocity was $V(z) = 1.5 + .5z$ km/s, which roughly corresponds to typical gradients found in the Gulf of Mexico. The maximum source-receiver offset was 3 km. Figure 6.22 shows the geometry of the reflectors.

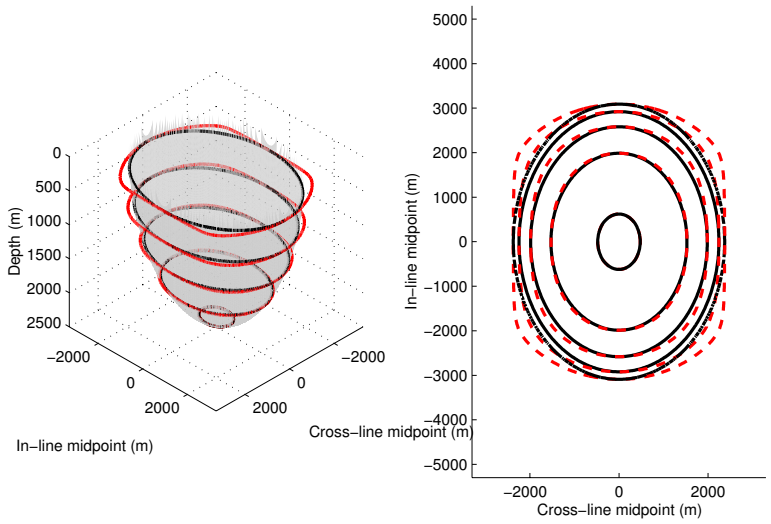


Figure 7.11: The gray surface shown in the left panel is the exact spreading surface for an impulse at at 2.5 seconds, an offset of 4,000 meters, and assuming a constant velocity of 2,500 m/s. The solid contour lines correspond to the exact spreading surface, while the dashed contour lines correspond to the approximate spreading surface. (This figure is from Biondi (1999b).) comaz-planeellips [NR]

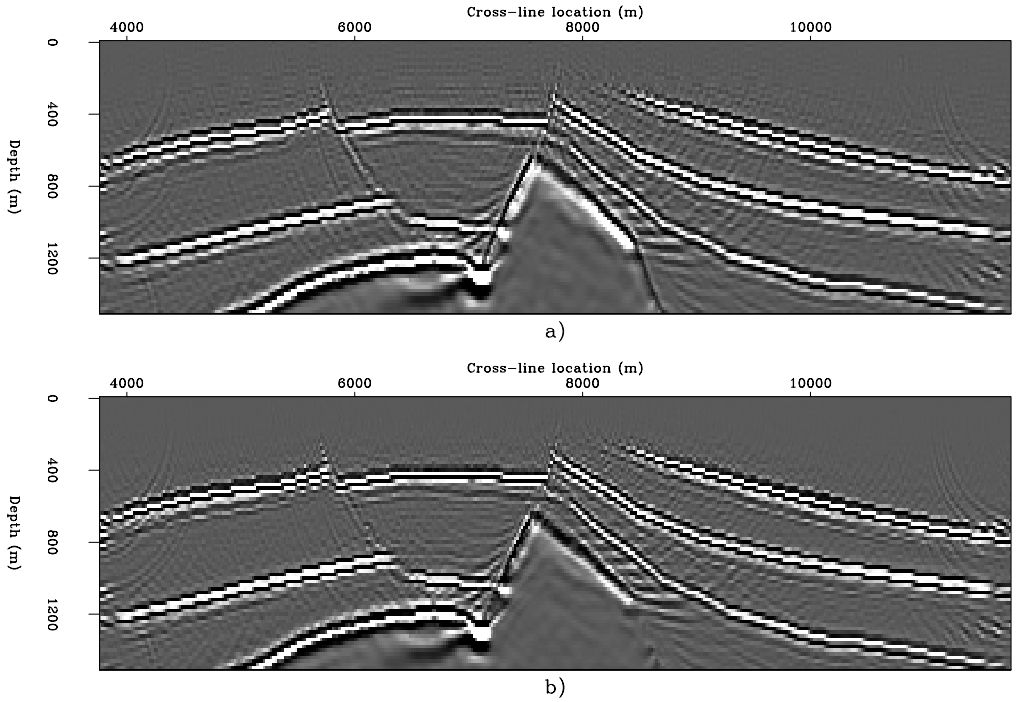


Figure 7.12: Cross-line sections ($x_m = 4{,}420$ meters) from the images obtained with a) common-azimuth migration, b) migration obtained by using the approximate DSR operator in equation (7.21). The shallow faults and the salt flanks are better imaged in (a) than in (b). (This figure is from Biondi (1999b).) comaz-Both-salt-over-y4420 [NR]

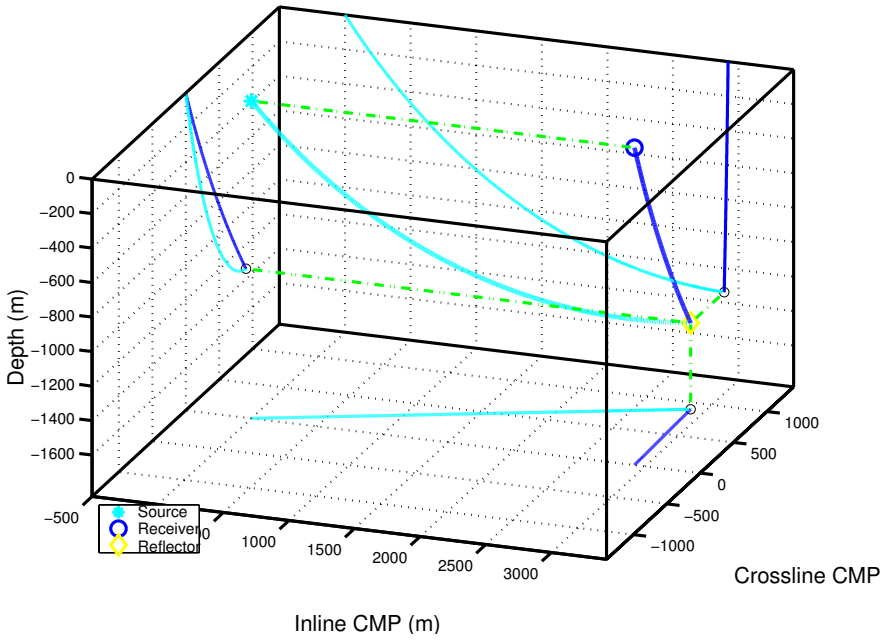


Figure 7.13: Ray corresponding to an event reflected by a reflector dipping at 60 degrees and oriented at 45 degrees with respect to the offset. The offset is 2.9 km offset, and p_{x_h}=.00045 s/m. The velocity function is $V(z) = 1.5 + .5z$ km/s. Notice the small, but finite, cross-line offset of the rays at depth. Also notice that the source ray (light gray) is close to overturn. (This figure is from Biondi (2003).) comaz-sem1 [NR]

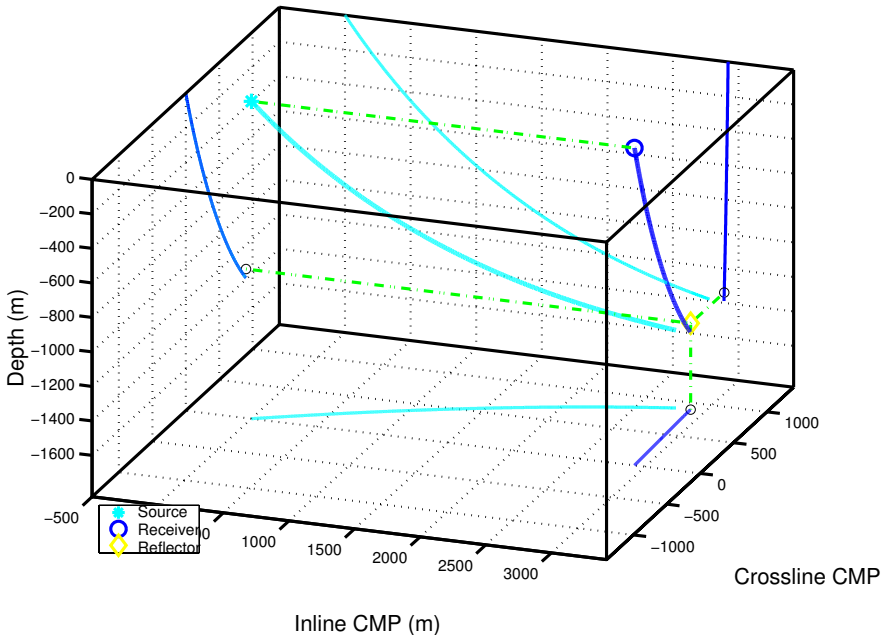


Figure 7.14: Equivalent common-azimuth rays for the same event shown in Figure 7.13. The common-azimuth rays are similar to the true rays shown in Figure 7.13, but the end points do not meet, causing a mispositioning of the migrated image. (This figure is from Biondi (2003).) comaz-sem2 [NR]

Figure 7.15 shows a subset of the migration results. The front face of the cube displayed in the figure is an in-line section through the stack. The other two faces are sections through the prestack image as a function of the offset ray parameter p_{x_h}. The three events in the right panel of Figure 7.15 correspond to the planes dipping at 30, 45 and 60 degrees. Notice that the events are almost perfectly flat as a function of the offset ray parameter p_{x_h}, except for the reflections from the 60 degrees dipping plane with large offset ray parameters (i.e. large reflection angle). Figures 7.13–7.14 show the rays corresponding to one of these events; in particular the one corresponding to p_{x_h}=.00045 s/m. Figure 7.16 shows the three orthogonal projections of these rays. The black (blue in colors) rays are the exact rays, while the light gray (red in colors) rays are the common-azimuth rays for the same events recorded at the surface. The solid (magenta in colors) dot corresponds to the imaging location for the common-azimuth migration. It is at the midpoint between the end points of the two rays. It is deeper than the correct one by Δ_z=48 m, and laterally shifted by Δ_x=-56 m and Δ_y=-2 m. However, at fixed horizontal location, the dot is shallower by $\Delta_{\text{z}-\text{plane}}$=-21 m than the reflecting plane. This is about the same vertical shift that is observable on the corresponding event in the ADCIG gather shown in Figure 7.15.

The maximum cross-line offset of the exact rays is about 200 meters. This maximum offset occurs at the intersection between the cross-line offset ray parameter (p_{y_h}) curves shown on the top-right panel in Figure 7.16. This small value for the maximum cross-line offset (about 7% of the maximum in-line offset), suggests that this event could be exactly downward continued by expanding the computational domain in a narrow strip around the zero cross-line offset. On the other hand, the value for the cross-line offset ray parameter p_{y_h} is about to .00018 s/m; that is, p_{y_h} is about 40% of the in-line offset ray parameter (p_{x_h}=.00045 s/m). To minimize the number of cross-line offsets needed to adequately sample the cross-line-offset dips (p_{y_h}), it is important to define an optimal range of p_{y_h} that is not symmetric around the origin. In the next section I will discuss how to use the common-azimuth migration equations for defining such a range.

7.2.3 Narrow-azimuth downward-continuation and migration

The kinematic analysis presented in the previous section suggests a generalization of common-azimuth migration based on the downward continuation of a narrow strip around the zero cross-line offset. The computational cost of such a generalization is obviously proportional to the number of cross-line offsets used to represent this narrow strip. The minimum width of the strip needed to properly image all events depends on the reflector geometry and on the velocity model. The maximum sampling in the cross-line offset direction depends on the cross-line-offset dip spectrum; the common-azimuth relationship (7.15), or its equivalent in term of reflection angles (6.19), provides a guideline for the cross-line dip spectrum as a function of the reflector geometry.

To achieve computational efficiency, the cross-line offset range must be as narrow as possible and still "capture" all the useful propagation paths and avoid boundary artifacts. This goal can be best accomplished by applying two complementary procedures: 1) definition of an "optimal" range of of cross-line-offset wavenumbers, k_{y_h}, for the downward continuation, and 2) window the ADCIGs along the azimuth axis to limit the effects of the boundary artifacts

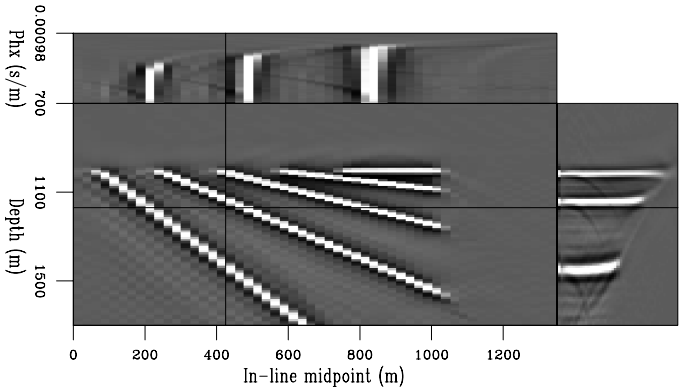


Figure 7.15: Subset of the results of common-azimuth migration of the synthetic data set. The front face of the cube is an in-line section through the stack. The other two faces are sections through the prestack image. The three events in the ADCIG (right panel) correspond to the planes dipping at 30, 45 and 60 degrees. Notice that the events are almost perfectly flat in the ADCIG, with the exception of the large offset ray parameters (i.e. large reflection angle) of the 60 degrees dipping plane. (This figure is from Biondi (2003).) comaz-CA-pull-WKBJ-stat-vp [NR]

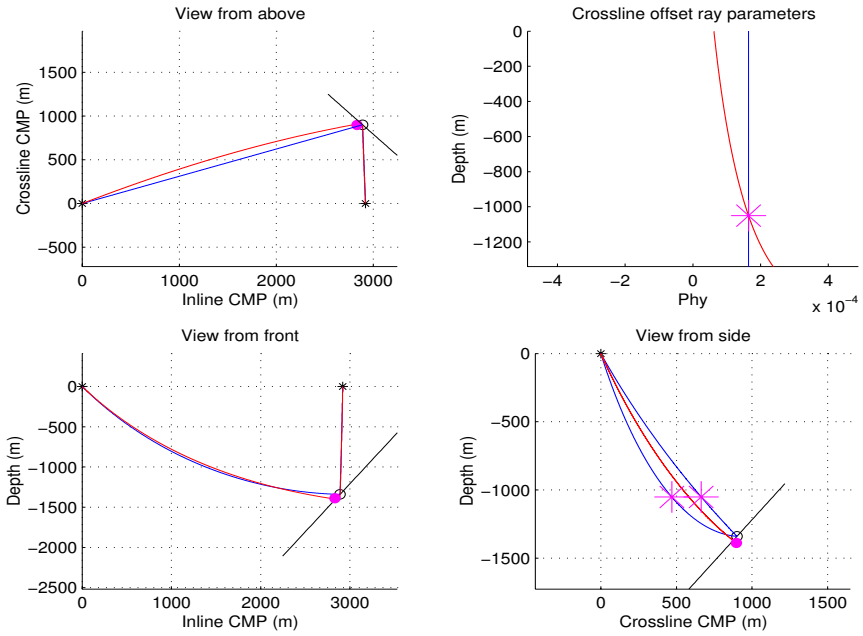


Figure 7.16: Orthogonal projections of rays shown in Figure 7.13 and Figure 7.14. The imaging point of common-azimuth migration (solid, red in colors, dot) is deeper than the correct one by Δ_z=48 m, and laterally shifted by Δ_x=-56 m and Δ_y=-2 m. However, the dot is shallower by $\Delta_{z-\text{plane}}$=-21 m than the plane at the same horizontal location. This vertical shift is consistent with the shift observed in the ADCIG gather shown in Figure 7.15. The maximum cross-line offset of the exact rays is about 200 m. The top-right panel shows the cross-line offset ray parameters as a function of depth. (This figure is from Biondi (2003).) comaz-sem3 [NR]

on the final migrated image.

To define an optimal range of cross-line-offset wavenumbers we need to take into account that, as demonstrated in the previous section, for dipping reflectors the dip spectrum is not centered around the zero dip ($k_{y_h} = 0$), and thus a symmetric range around zero would be wasteful. The information provided by the common-azimuth equation can be exploited to define a range of cross-line-offset dips that corresponds to propagation azimuths centered around the zero-azimuth direction defined by the common-azimuth stationary path. For this reason, and because we can window the ADCIGs along the azimuth axis, the resulting migration method is called **narrow-azimuth migration**.

The definition of an optimal cross-line offset-wavenumber range is based on the common-azimuth stationary path, where the center of the range, $\overline{k}_{y_h}$ is defined by equation (7.15), or an approximation of equation (7.15). In equation (7.15), the cross-line offset-wavenumber k_{y_h} is function of the velocity (at the source, $v(\mathbf{s},z)$, and at the receiver $v(\mathbf{g},z)$), the temporal-frequency, ω, and of the wavenumbers, $k_{x_m}, k_{y_m}, k_{x_h}$. The dependency on velocity makes the direct application of equation (7.15) to depth migration cumbersome; furthermore, the dependency on the temporal frequency is also inconvenient when we want to compute ADCIGs after imaging, since the dependency on ω is lost during the imaging step. We therefore use an approximation of equation (7.15).

A practical approximation of equation (7.15) can be achieved by setting the values of the velocity and the frequency to be constant and equal to average values, $\overline{v}$ and $\overline{\omega}$. In this way the ratio α between the cross-line offset-wavenumber k_{y_h} and the cross-line midpoint wavenumber k_{y_m} is dependent only on the in-line wavenumbers k_{x_m} and k_{x_h}; that is:

$$\alpha_{y_h}\left(k_{x_m},k_{x_h}\right) = \frac{\sqrt{\frac{\overline{\omega}^2}{\overline{v}^2} - \frac{1}{4}\left(k_{x_m}+k_{x_h}\right)^2} - \sqrt{\frac{\overline{\omega}^2}{\overline{v}^2} - \frac{1}{4}\left(k_{x_m}-k_{x_h}\right)^2}}{\sqrt{\frac{\overline{\omega}^2}{\overline{v}^2} - \frac{1}{4}\left(k_{x_m}+k_{x_h}\right)^2} + \sqrt{\frac{\overline{\omega}^2}{\overline{v}^2} - \frac{1}{4}\left(k_{x_m}-k_{x_h}\right)^2}}. \tag{7.24}$$

To avoid singularities when the denominator in equation (7.24) becomes small, it is also appropriate to set a limit on the maximum magnitude of α_{y_h}, so that $\left|\alpha_{y_h}\right| \leq \alpha_{y_h}^{\max}$. The center of the k_{y_h} range can thus be set as:

$$\overline{k}_{y_h}\left(k_{x_m},k_{y_m},k_{x_h},\right) = \begin{cases} -k_{y_m}\alpha_{y_h}^{\max} & \text{if } \alpha_{y_h}\left(k_{x_m},k_{x_h}\right) \leq -\alpha_{y_h}^{\max} \\ k_{y_m}\alpha_{y_h}\left(k_{x_m},k_{x_h}\right) & \text{if } \left|\alpha_{y_h}\left(k_{x_m},k_{x_h}\right)\right| < \alpha_{y_h}^{\max} \\ k_{y_m}\alpha_{y_h}^{\max} & \text{if } \alpha_{y_h}\left(k_{x_m},k_{x_h}\right) \geq \alpha_{y_h}^{\max} \end{cases}, \tag{7.25}$$

and the the range is defined as:

$$\overline{k}_{y_h}\left(k_{x_m},k_{y_m},k_{x_h},\right) - \left(\frac{N_{y_h}}{2}-1\right)dk_{y_h} \leq k_{y_h} \leq \overline{k}_{y_h}\left(k_{x_m},k_{y_m},k_{x_h},\right) + \frac{N_{y_h}}{2}dk_{y_h}, \tag{7.26}$$

where N_{y_h} is the number of cross-line offsets and the cross-line offset-wavenumber sampling dk_{y_h} is

$$dk_{y_h} = \frac{2\pi}{N_{y_h}\Delta y_h}. \tag{7.27}$$

When computing 3-D ADCIGs from the results of narrow-azimuth migration, it is important to take into account the shift in the cross-line offset-wavenumber range defined by equation (7.25). This task is straightforward to accomplish when applying equation (6.19) (Section 6.2.1) to convert offset ray parameters into angles. It is also easy to accomplish when computing ADCIGs after imaging by applying equation (6.25) (Section 6.2.2), because we approximated the stationary path (7.15) with an expression that is independent from the temporal frequency ω.

Narrow-azimuth migration of a simple synthetic data set

The migration example shown in Figure 7.15 illustrates the limitation of common-azimuth migration in a a simple situation that can be effectively analyzed by raytracing (Figures 7.13–7.14 and 7.16). The next set of figures shows the comparison between the images obtained by full-azimuth migration, narrow-azimuth migration, and common-azimuth migration on the same data set. In particular I focus on the same ADCIG shown in the right panel of Figure 7.15 and then analyzed by raytracing. To create data sets with more than one cross-line offset I padded the original common-azimuth data with zeros. The differences between the results are thus exclusively due to differences between the migration algorithms.

Figure 7.17 shows four ADCIGs, all extracted at the same horizontal location, but corresponding to four different migrations. The full-azimuth migration and the narrow-azimuth migration show the ADCIGs at the "true" reflection azimuth ($\phi = 12$ degrees) for the event of interest; that is, the reflection from the deepest reflector. The shallower events are reflected at azimuths smaller than 12 degrees, and thus they are not flat in the ADCIGs, except for the common-azimuth ADCIG.

Figure 7.17a shows the ADCIG obtained by full-azimuth migration with $N_{y_h} = 32$, $\overline{k}_{y_h} = 0$, and $\Delta y_h = 50$ meters. Figure 7.17b shows the ADCIG obtained by narrow-azimuth migration with $N_{y_h} = 32$ and $\Delta y_h = 100$ meters. The center of the k_{y_h} range, $\overline{k}_{y_h}$, was set by using equations (7.24) and (7.25) with $\overline{\omega} = 2\pi 25$ s^{-1}, $\overline{v} = 1{,}500$ m/s, and $\alpha_{y_h}^{\max} = 2$. Figure 7.17c shows the ADCIG obtained by full-azimuth migration with $N_{y_h} = 32$, $\overline{k}_{y_h} = 0$, and $\Delta y_h = 100$ meters, and Figure 7.17d shows the ADCIG obtained by common-azimuth migration. In Figure 7.17a and Figure 7.17b all events are correctly imaged, even the wide-aperture angle reflection from the reflector dipping at 60 degrees, which shows the upward departure from flatness in the common-azimuth migration. Notice that the same event in Figure 7.17c has not been correctly imaged, because the cross-line offset sampling was too coarse, $\Delta y_h = 100$ meters, for the full-azimuth migration. In contrast, narrow-azimuth migration with the same value of Δy_h (Figure 7.17b) images the event correctly, because its uses the computational domain more efficiently by shifting the center of the range according to equation (7.25).

Figures 7.18–7.20 shows slices taken from the same ADCIGs shown in Figure 7.17, at the constant depth of $z = 1{,}430$ meters; this depth corresponds to the deepest reflector shown in Figure 7.17. The reflection amplitudes are shown as functions of both the aperture angle (γ) and the reflection azimuth (ϕ). Because the common-azimuth ADCIG lacks the azimuthal dimension, the depth-slice obtained by common-azimuth migration is omitted.

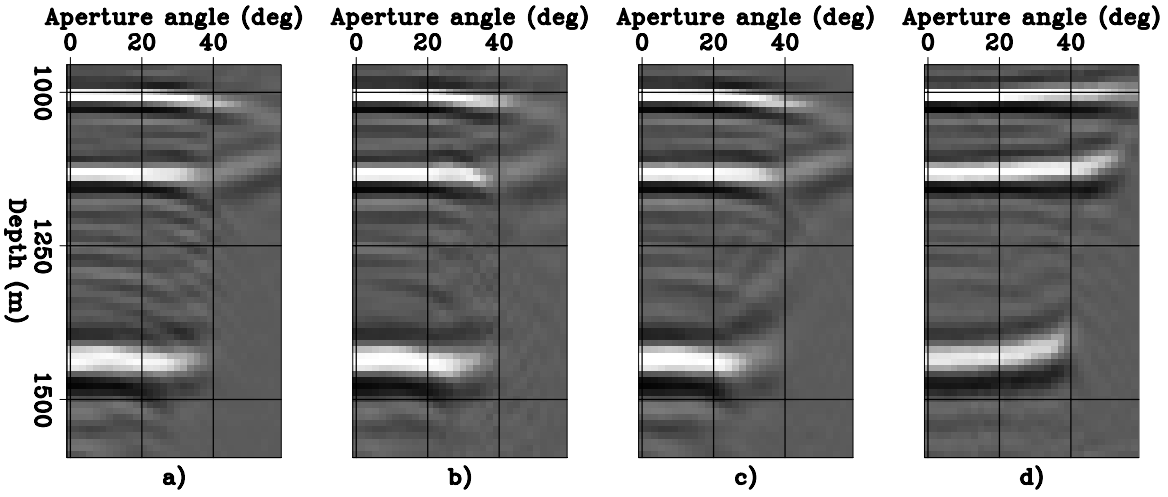


Figure 7.17: Four ADCIGs, all extracted at the same horizontal location, but corresponding to four different migrations: (a) ADCIG obtained by full-azimuth migration with $N_{y_h} = 32$, $\overline{k}_{y_h} = 0$, and $\Delta y_h = 50$ meters, (b) ADCIG obtained by narrow-azimuth migration with $N_{y_h} = 32$ and $\Delta y_h = 100$ meters, (c) ADCIG obtained by full-azimuth migration with $N_{y_h} = 32$, $\overline{k}_{y_h} = 0$, and $\Delta y_h = 100$ meters, (d) ADCIG obtained by common-azimuth migration. The full-azimuth migration and the narrow-azimuth migration show the ADCIGs at the "true" reflection azimuth for the deepest reflector; that is, $\phi = 12$ degrees. comaz-cig-az12-azim [CR]

Figure 7.18 shows the image obtained by full-azimuth migration when the cross-line offset axis is sufficiently sampled (depth slice of ADCIG shown in Figure 7.17a). Figure 7.19 shows the image obtained by narrow-azimuth migration (depth slice of ADCIG shown in Figure 7.17b). Figure 7.20 shows the image obtained by full-azimuth migration when the cross-line offset axis is not sufficiently sampled (depth slice of ADCIG shown in Figure 7.17c). The adequately-sampled full-azimuth image (Figure 7.18) and the narrow-azimuth image (Figure 7.19) are similar and show coherent events up to $\gamma \approx 40$ degrees, though the narrow-azimuth image is slightly more noisy than the wide-azimuth image. The coarsely-sampled full-azimuth image (Figure 7.20) shows coherent events only up to $\gamma \approx 30$ degrees.

REFERENCES

Biondi, B., and Palacharla, G., 1996, 3-D prestack migration of common-azimuth data: Geophysics, **61**, 1822–1832.

Biondi, B., and Shan, G., 2002, Prestack imaging of overturned reflections by reverse time migration: 72nd Ann. Internat. Meeting, Soc. of Expl. Geophys., 1284–1287.

Biondi, B., and Symes, W. W., 2003, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics: accepted for publication.

Biondi, B., 1999a, Subsalt imaging by common-azimuth migration: SEP–**100**, 113–125.

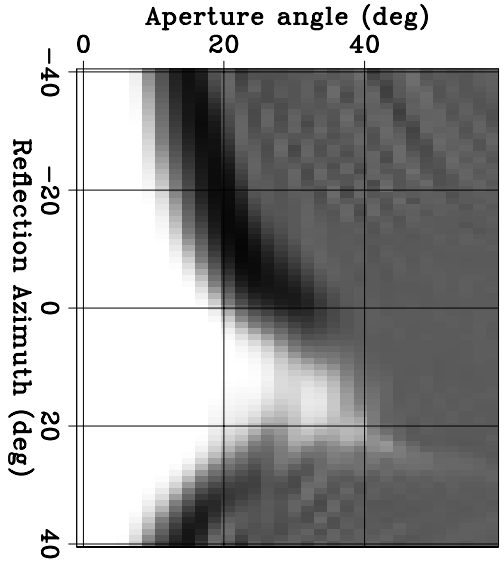


Figure 7.18: Full-azimuth ADCIG as a function of aperture angle (γ) and reflection azimuth (ϕ) at constant depth ($z = 1{,}430$ meters). The cross-line offset sampling ($\Delta y_h = 50$ meters) was adequate for imaging the events with large γ. comaz-zaz-60-60-dense-all-v4-data8 [CR]

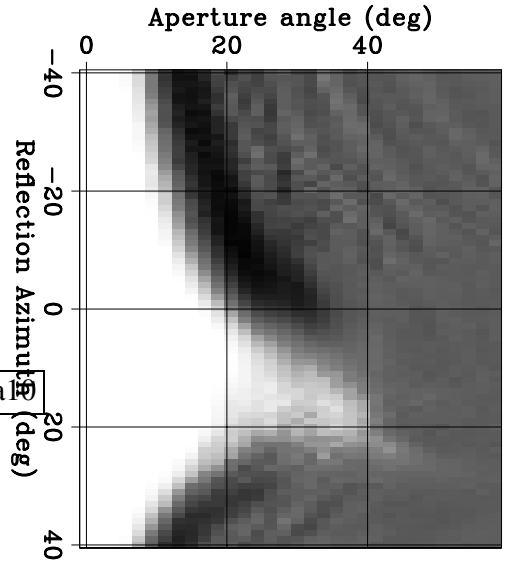


Figure 7.19: Narrow-azimuth ADCIG as a function of aperture angle (γ) and reflection azimuth (ϕ) at constant depth ($z = 1{,}430$ meters). The events with large γ are correctly imaged. comaz-zaz-60-60-dense-nam-pos-all-v5-data1 [CR]

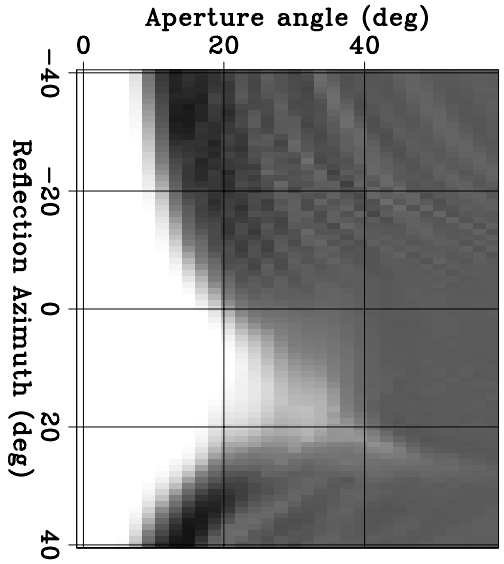


Figure 7.20: Full-azimuth ADCIG as a function of aperture angle (γ) and reflection azimuth (ϕ) at constant depth ($z = 1{,}430$ meters). The cross-line offset sampling ($\Delta y_h = 100$ meters) was too coarse for imaging the events with large γ. comaz-zaz-60-60-dense-all-v4-data9 [CR]

Biondi, B., 1999b, Offset plane waves vs. common-azimuth migration for sub-salt imaging: SEP–**102**, 15–34.

Biondi, B., 2003, Narrow-azimuth migration of marine streamer data: SEP–**113**, 107–121.

Bleistein, N., 1984, Mathematical methods for wave phenomena: Academic Press.

Bonomi, E., and Cazzola, L., 1999, Prestack imaging of compressed seismic data: A Monte Carlo approach: 69th Ann. Internat. Meeting, Soc. of Expl. Geophys., 1914–1917.

Cazzola, L., Pizzaferri, L., Ratti, L., Cardone, G., and Bonomi, E., 2004, An example of wavefield depth migration and Monte Carlo imaging in West Africa deep waters: 74th Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, accepted for publication.

Duquet, B., Lailly, P., and Ehinger, A., 2001, 3-D plane wave migration of streamer data: 71st Ann. Internat. Meeting, Soc. of Expl. Geophys., 1033–1036.

Etgen, J. T., 2003 , private communication.

Higginbotham, J. H., Shin, Y., and Sukup, D. V., 1985, Directional depth migration (short note): Geophysics, **50**, 1784–1796.

Jing, X., Finn, C., Dickens, T., and Willen, D., 2000, Encoding multiple shot gathers in prestack migration: 70th Ann. Internat. Meeting, Soc. of Expl. Geophys., 786–789.

Liu, F., Stolt, R., Hanson, D., and Day, R., 2002, Plane wave source composition: An accurate phase encoding scheme for prestack migration: 72nd Ann. Internat. Meeting, Soc. of Expl. Geophys., 1156–1159.

Morton, S. A., and Ober, C. C., 1998, Faster shot-record migrations using phase encoding: 68th Ann. Internat. Meeting, Soc. of Expl. Geophys., 1131–1134.

Mosher, C. C., Foster, D. J., and Hassanzadeh, S., 1997, Common angle imaging with offset plane waves: 67th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1379–1382.

Notfors, C., Gray, S., Sun, J., Young, J., Zhang, Y., and Chernis, L., 2003, Delayed-shot 3D prestack depth migration: 73rd Ann. Internat. Meeting, Soc. of Expl. Geophys., 1027–1030.

Ottolini, R., and Claerbout, J. F., 1984, The migration of common-midpoint slant stacks: Geophysics, **49**, 237–249.

Romero, L. A., Ghiglia, D. C., Ober, C. C., and Morton, S. A., 2000, Phase encoding of shot records in prestack migration: Geophysics, **65**, 426–436.

Schultz, P. S., and Claerbout, J. F., 1978, Velocity estimation and downward-continuation by wavefront synthesis: Geophysics, **43**, 691–714.

Shan, G., and Biondi, B., 2004, Imaging overturned waves by plane-wave migration in tilted coordinates: 74th Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, accepted for publication.

Stoffa, P. L., Buhl, P., Diebold, J. B., and Wenzel, F., 1981, Direct mapping of seismic data to the domain of intercept time and ray parameter - A plane-wave decomposition: Geophysics, **46**, 255–267.

Stolt, R. H., 1978, Migration by Fourier transform: Geophysics, **43**, 23–48.

Stork, C., and Kapoor, J., 2004, How many P values do you want to migrate for delayed shot wave equation migration?: 74th Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, accepted for publication.

Temme, P., 1984, A comparison of common-midpoint single-shot and plane-wave depth migration: Geophysics, **49**, 1896–1907.

Treitel, S., Gutowski, P. R., and Wagner, D. E., 1982, Plane-wave decomposition of seismograms: Geophysics, **47**, 1375–1401.

Vaillant, L., and Calandra, H., 2000, Common-azimuth migration and Kirchhoff migration for 3-D prestack imaging: A comparison on North Sea data: SEP–**103**, 139–147.

Whitmore, N. D., and Garing, J. D., 1993, Interval velocity estimation using iterative prestack depth migration in the constant angle domain: The Leading Edge, **12**, 757–762.

Whitmore, N. D., 1995, An imaging hierarchy for common-angle plane wave seismograms: Ph.D. thesis, University of Tulsa.

Chapter 8

Imaging and aliasing

The quality of 3-D images is strongly influenced by the spatial sampling of the data, and whether the imaging operators properly take into account the data sampling. Strong aliasing artifacts degrade the images when the data are poorly sampled and the imaging operators are not carefully implemented. The sampling problem is more acute in 3-D imaging than in 2-D imaging because the spatial axes of 3-D data are often sparsely and irregularly sampled. In this chapter I analyze the problems caused by regular data grids that are too coarsely, but regularly, sampled. Chapter 9 discusses the issues related with the irregularity of data acquisition and reflector illumination.

There are three types of aliasing that are relevant to seismic imaging: data aliasing, image aliasing, and operator aliasing. Data aliasing and image aliasing are fairly straightforward to understand using standard sampling theory, whereas operator aliasing is more peculiar to imaging operators and thus requires a detailed analysis. Since the beginning of the development of Kirchhoff migration methods (Schneider, 1978), it has been evident that aliasing artifacts appear in the image if data and operator aliasing are not considered in the implementation of the summation operator (Gardner et al., 1974). The methods for preventing operator aliasing during Kirchhoff migration are thus well established (Gray, 1992; Bevc and Claerbout, 1992; Lumley et al., 1994; Abma et al., 1998; Biondi, 2001). The first part of this chapter discusses aliasing for Kirchhoff migration.

In contrast, operator aliasing becomes an issue for wavefield-continuation migration only when migrating prestack data. Therefore, the understanding of operator aliasing for wavefield-continuation migration is less well developed. The second part of this chapter presents the basic concepts underlying our understanding of operator aliasing in wavefield-continuation migrations. Some of these concepts are relatively well established (Zhang et al., 2003), whereas others are subject of active research (Artman et al., 2003). At the end of the chapter, I apply the general concepts governing operator aliasing in wavefield-continuation migration to the analysis of aliasing in prestack migration, and to the definition of a simple procedure to avoid, or at least attenuate, operator-aliasing artifacts.

8.1 Aliasing fundamentals

The basic principles of sampling and aliasing are easily understood for plane waves. Plane waves are stationary signals in time and space, and they can thus be analyzed in the frequency-wavenumber domain, as well in the original time-space domain. For a plane wave with slope p, the frequency ω and the wavenumber k are related by the following expression:

$$p = \frac{k}{\omega}. \tag{8.1}$$

Plane waves with a given frequency ω_d are well sampled if their slopes are limited within the range determined by the condition

$$|p| \leq \frac{k_N}{\omega_d}, \tag{8.2}$$

where k_N is the Nyquist wavenumber given as a function of the spatial sampling Δx as follows:

$$k_N = \frac{\pi}{\Delta x}. \tag{8.3}$$

Aliasing occurs when condition (8.2) is violated because either the dip range (**dip bandwidth**) is too wide, or the waveform is too high-frequency, or the spatial sampling is too coarse.

Figure 8.1 shows a classic wavenumber representation of aliasing. The drawing at the top of the figure (Figure 8.1a) shows the spectrum, for a single temporal frequency, of a family of plane waves that are adequately sampled in space. Because the signal is discretized in space, the original spectrum of the continuous signal (center) is replicated at intervals of twice the Nyquist wavenumbers (left and right). The signal is not aliased because its maximum wavenumber is lower than the Nyquist wavenumber. In contrast, aliasing occurs in Figure 8.1b because the original spectrum of the signal is stretched, by either an increase in temporal frequency or a widening of the dip bandwidth of the plane waves. Consequently, plane waves that are steeply dipping, and correspondingly have high wavenumber, appear to have a negative wavenumber and to be dipping in the opposite direction. Figure 8.1c shows the aliasing caused by doubling the spatial sampling Δx, and consequently halving the Nyquist wavenumber.

The drawing in Figure 8.2 illustrates the aliasing bounds in the multi-dimensional case. For the sake of simplicity, the bounds on the data wavenumbers are set as a maximum of the absolute value of the wavenumber vector. In this case, all the data wavenumbers are contained in a circular region. The sampling along the cross-line direction y is assumed to be coarser by a factor of two than that along the in-line direction x. When the temporal frequency is low enough (Figure 8.2a), the data dips are not aliased. However, if the temporal frequency ω_d increases, the replicas of the circles overflow in the main band. To avoid aliasing artifacts, only the wavenumber components within the hourglass-shaped area should be included in the image, or, even more conservatively, the imaging should be limited to the wavenumber components within the dark rectangular area in Figure 8.2.

The principles of aliasing embodied in Figures 8.1 and 8.2 can be directly applied to the analysis of **data aliasing** and **image aliasing**. The aliasing of imaging operators is still based on the same fundamental concepts, but it is less obvious. To understand **operator aliasing** we

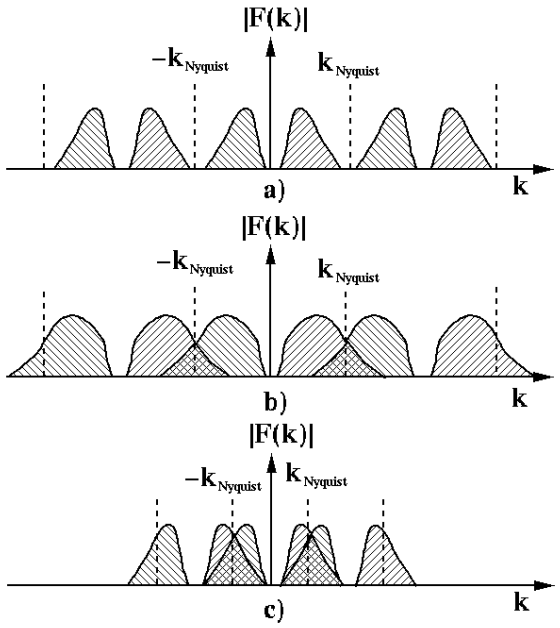


Figure 8.1: Spectra of a family of plane waves when there is a) no aliasing, b) aliasing caused by an increase in frequency or dip bandwidth, and c) aliasing caused by doubling the spatial sampling. alias-ali1d [NR]

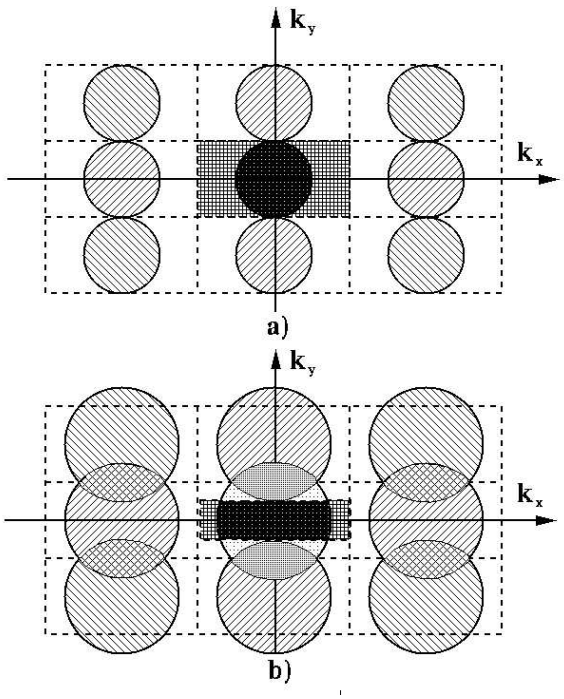


Figure 8.2: Wavenumber representation of operator aliasing in 2-D. The black-shaded areas are the admissible regions for the operator dip. alias-ali2d [NR]

need to analyze how imaging operators connect the data space with the image space. Operator aliasing is function of the local time derivatives of the operator's summation surface, often called operator dip, and of the frequency bandwidth and dip bandwidth of the data.

In this chapter we will analyze methods to avoid any of the three kinds of aliasing that cause artifacts in seismic images. The second section of this this chapter (Section 8.2) presents the theory of aliasing and anti-aliasing for Kirchhoff imaging operators. The third section (Section 8.3) builds on the understanding gained in the first part to analyze aliasing in wavefield-continuation migrations, and in particular for prestack wavefield-continuation migration.

8.1.1 The anti-aliasing dilemma

Before plunging into the theory, I discuss an example of imaging artifacts caused by operator and data aliasing, and present a fundamental dilemma that we face when tackling the task of imaging aliased data. Since prestack seismic data is almost always aliased, we constantly need to determine the appropriate trade-off between imaging artifacts and image resolution. Although the following discussion is based on results obtained by Kirchhoff migration, prestack migration by wavefield-continuation methods present similar challenges.

In Kirchhoff imaging, operator aliasing is usually avoided by the suppression of some high-frequency components during the summation process. Unfortunately, this low-pass filtering decreases the image resolution when the data are aliased. We therefore face a dilemma when imaging aliased data: on one hand, to avoid aliasing noise we need to apply anti-aliasing filters; on the other hand, we do not want to lose image resolution. Figure 8.3 exemplifies the problem. It shows the results of 3-D zero-offset time migration of a salt-dome flank in the Gulf of Mexico. The section on the left (Figure 8.3a) was 3-D migrated without the use of any anti-aliasing filter, while the section on the right (Figure 8.3b) was obtained with a standard anti-aliased migration. Whereas the image obtained without anti-aliasing is much noisier than the anti-aliased one, it also has higher resolution. In the shallow part of the section shown in Figure 8.3a, the aliasing noise is so strong that it is impossible to appreciate the higher resolution of Figure 8.3a compared with Figure 8.3b. But when zooms into the deeper part of the sections are compared (Figure 8.4), it is apparent that we lose resolution by applying anti-aliasing. In particular, the high-frequency dipping event at about CMP X=700 meters and Time=2.2 seconds is poorly resolved in the anti-aliased migration (Figure 8.4b). The anti-aliased migration misses a whole wavelet cycle of the sediment truncation against the salt flank. If we consider that hydrocarbon reservoirs are often located at the sediment-salt interfaces, we appreciate the potential advantages of improving the resolution of such events.

In this example, migration without anti-aliasing achieves higher resolution than anti-aliased migration because it images data components that are aliased in the data space. In particular, the steeply dipping energy reflected from the salt flanks, visible in the data window shown in Figure 8.5, is aliased. Figure 8.6 shows the frequency-wavenumber spectrum of the data window in Figure 8.5. Figure 8.6 shows the central band of the spectrum and its two replicas on either side. The vertical white lines correspond to the Nyquist wavenumbers. The two "replicated bands" are caused by the spatial discretization, and their existence is at the root of the

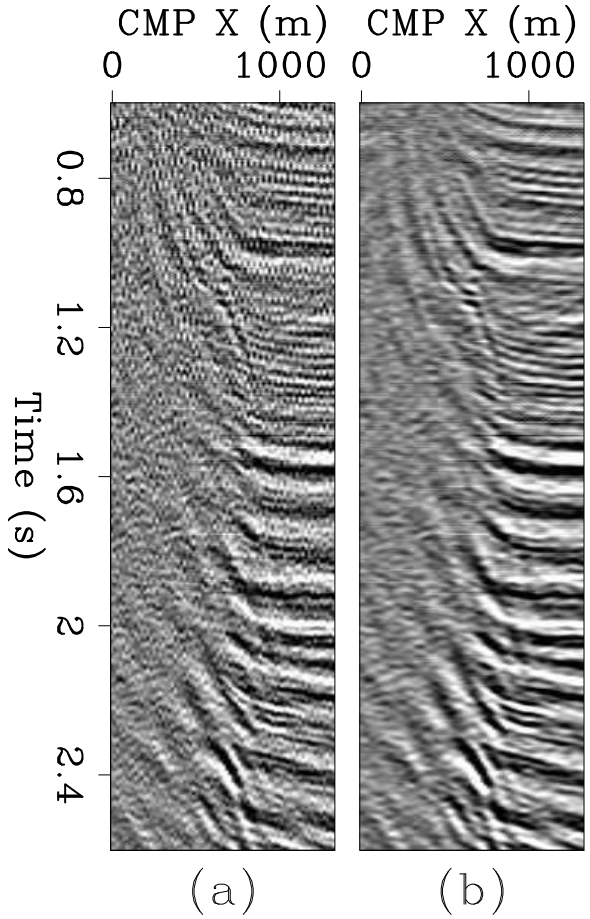


Figure 8.3: 3-D migrations of salt-dome flanks in the Gulf of Mexico: (a) migration obtained without the use of any anti-aliasing filter, (b) migration obtained with the application of a "standard" anti-aliasing filter. alias-Comp-WL-intro [ER]

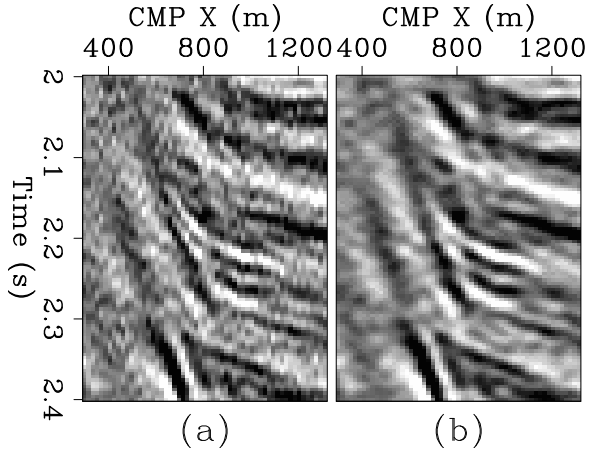


Figure 8.4: A zoom into the 3-D migrations of salt-dome flanks in the Gulf of Mexico shown in Figure 8.3: (a) migration obtained without the use of any anti-aliasing filter, (b) migration obtained with the application of a "standard" anti-aliasing filter. alias-Comp-WB-intro [ER]

aliasing problem. The dipping events that appear aliased in the data window correspond to the "clouds" in the spectrum contoured with ellipses in Figure 8.6. The ellipse marked as **B** contours the energy that can be correctly migrated. The cloud starts in the main band but crosses the positive Nyquist line and trespasses upon the replicated band on the positive side. The portion of the cloud that crosses the positive Nyquist corresponds to the signal components that are lost when conventional anti-aliased migration is applied (Figures 8.3b and 8.4b), and that are responsible for the high resolution achieved by migration without anti-aliasing (Figures 8.3a and 8.4a). The ellipse marked as **A** contours the energy that cannot be correctly migrated and should be avoided during the summation.

8.2 Aliasing in Kirchhoff migration

In the previous sections we have reviewed the basic concepts of aliasing and illustrated the effects of aliasing on the migrated images. In this section we apply these concepts to the problem of eliminating artifacts in the images created by the application of imaging operators applied with Kirchhoff methods. We analyze in detail the aliasing of 3-D prestack migration, but the same concepts can be applied to all other integral operators presented in this book.

8.2.1 Aliasing in image space

The simplest type of aliasing related to the imaging operator is **image-space aliasing**. It occurs when the spatial sampling of the image is too coarse to represent adequately the steeply dipping reflectors that the imaging operator attempts to build during the imaging process. In other words, image-space aliasing is caused by excessively coarse sampling of the image space, with consequent aliasing of the migration ellipsoid. With Kirchhoff imaging, the choice of image sampling is arbitrary, and thus image aliasing can be avoided easily by increasing the density of the image. However, the cost of migration grows proportionally with image density. For a given spatial sampling of the image, to avoid image aliasing we need to limit during the migration the frequency content of the image as a function of reflectors' dips. This goal can be accomplished by the application of a dip-dependent temporal low-pass filtering of the input data during the summation process.

An efficient method to low-pass filter time-domain data with variable frequency is the triangle-filters method described in *Basic Earth Imaging* (Claerbout, 1995). An alternative method is to precompute low-passed versions of the input traces, and select the appropriate input data during summation (Gray, 1992). This second method is potentially more accurate than triangle filtering, and it is more computationally efficient when each input trace is summed into many output traces, as happens for 3-D migrations (Abma et al., 1998). However, this method may require storing many versions of the input data. To reduce the storage requirements without affecting the accuracy, I linearly interpolate the low-passed traces along the frequency axis during the summation.

The anti-aliasing constraints used to avoid image aliasing can be easily derived from basic

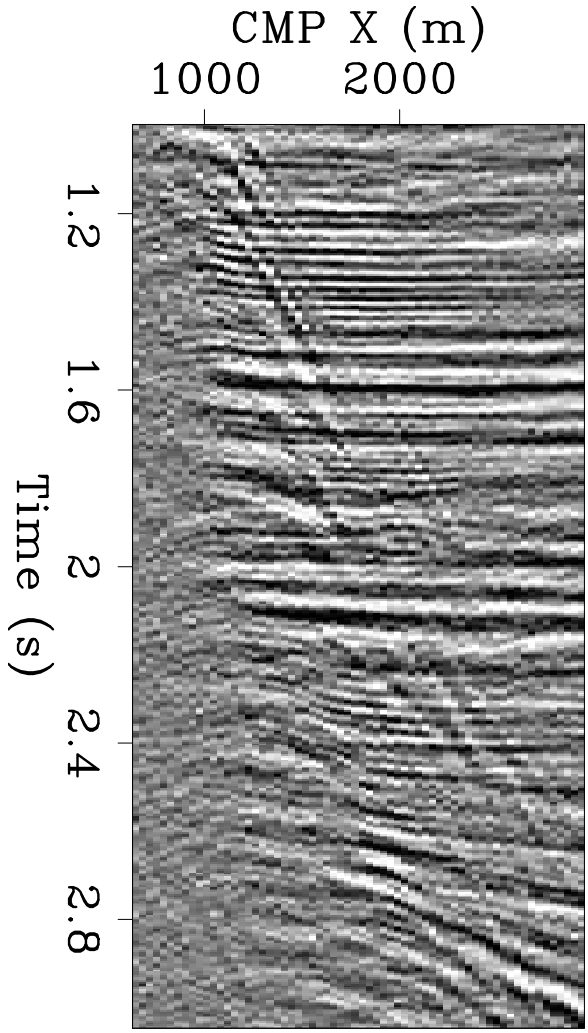


Figure 8.5: Data window containing aliased reflections from the salt flanks. alias-Wind-data [ER]

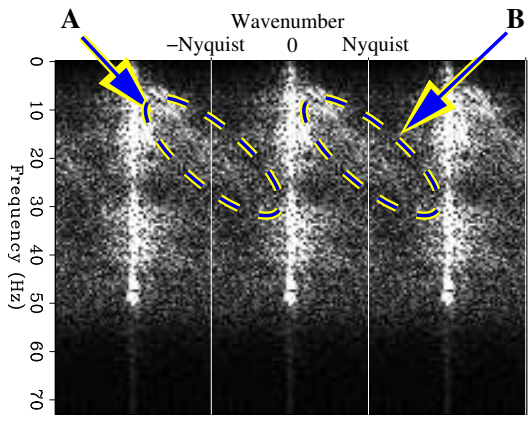


Figure 8.6: Frequency-wavenumber spectrum of the data window shown in Figure 8.5. Notice that the aliased events correspond to the "clouds" in the spectrum contoured with ellipses in the Figure. The ellipse marked as **B** contours the energy that can be correctly migrated. The ellipse marked as **A** contours the energy that cannot be correctly migrated and should be avoided during the summation. alias-Wind-spec-geo2 [NR]

sampling theory. For the case of time migration, when the coordinates of the image space are $\xi = (\tau_\xi, x_\xi, y_\xi)$, the pseudo-depth frequency ω_τ^ξ must fulfill the following inequalities:

$$\begin{aligned} \omega_\tau^\xi &< \frac{\pi}{\Delta x_\xi \left| p_x^\xi \right|}, \\ \omega_\tau^\xi &< \frac{\pi}{\Delta y_\xi \left| p_y^\xi \right|}; \end{aligned} \tag{8.4}$$

where Δx_ξ and Δy_ξ are the image sampling rates of the x_ξ and y_ξ axes, and p_x^ξ and p_y^ξ are the operators' dips in image space.

As discussed above, it is convenient to apply an anti-aliasing filter as a band-pass filter of the input traces, before their contributions are summed into the output image. Therefore, we need to recast the constraints on the pseudo-depth frequency ω_τ^ξ of equations (8.4) into constraints on the input data frequency ω_d. This distinction is important, because stretching or compression of the time axis changes the frequency content of the seismic wavelet during the imaging process. In particular, during time migration the wavelet is always stretched; neglecting this stretch would lead to anti-aliasing constraints that are too stringent. Notice that the seismic wavelet may get compressed, instead of stretched, by other imaging operators, such as inverse DMO and AMO (Chapter 3). The pseudo-depth frequency of the image and the temporal frequency of the data are thus linked by the **wavelet-stretch factor** $dt_D/d\tau_\xi$, as $\omega_\tau^\xi = \omega_d dt_D/d\tau_\xi$, where t_D is the time coordinate in the data space. Taking into account the wavelet-stretch factor, we can write the constraints on the data frequency that control image aliasing, as a function of the image sampling rates Δx_ξ and Δy_ξ, the image dips p_x^ξ and p_y^ξ, and the wavelet-stretch factor $dt_D/d\tau_\xi$:

$$\begin{aligned} \omega_d &< \frac{\pi}{\Delta x_\xi \left| p_x^\xi \right| \frac{dt_D}{d\tau_\xi}}, \\ \omega_d &< \frac{\pi}{\Delta y_\xi \left| p_y^\xi \right| \frac{dt_D}{d\tau_\xi}}. \end{aligned} \tag{8.5}$$

Image anti-aliasing for 3-D prestack time migration

For 3-D prestack time migration, the reflector dips p_x^ξ and p_y^ξ and the wavelet-stretch factor $dt_D/d\tau_\xi$ can be analytically derived as functions of the input and output trace geometry and the input time. This analytical derivation starts from the expression of the spreading function presented in equation (3.13). This equation represents an ellipsoid as a parametric function of the angles α and β. To derive the expression for the dips in the model space, we start by differentiating the coordinates of the image point with respect to the angles α and β; that is,

$$\begin{aligned} dz_\xi &= \frac{t_N V}{2}(\cos\alpha\cos\beta\, d\alpha - \sin\alpha\sin\beta\, d\beta), \\ dx_\xi &= \frac{t_D V}{2}(-\sin\alpha\, d\alpha), \\ dy_\xi &= \frac{t_N V}{2}(\cos\alpha\sin\beta\, d\alpha + \sin\alpha\cos\beta\, d\beta). \end{aligned} \tag{8.6}$$

The first step to obtain useful relationships is to eliminate the differentials $d\alpha$ and $d\beta$ from this set of equations by setting dy_ξ equal to zero when evaluating the dip in the in-line direction p_x^ξ, and setting dx_ξ equal to zero when evaluating the dip in the cross-line direction p_y^ξ. The second step is to eliminate the angles themselves and express the image dips as functions of the image coordinates (z_ξ, x_ξ, y_ξ):

$$p_x^\xi = \frac{dz_\xi}{dx_\xi} = \frac{t_N}{t_D}\frac{\cot\alpha}{\cos\beta} = \frac{t_N^2 x_\xi}{t_D^2 z_\xi}, \tag{8.7}$$

$$p_y^\xi = \frac{dz_\xi}{dy_\xi} = \tan\beta = \frac{y_\xi}{z_\xi}. \tag{8.8}$$

The wavelet-stretch factor for 3-D prestack migration can be easily derived by differentiating the summation surfaces expressed in equation (2.2), and obtaining

$$\frac{dt_D}{dz_\xi} = \frac{z_\xi}{V^2}\left(\frac{1}{t_s} + \frac{1}{t_g}\right). \tag{8.9}$$

To gain intuition about the effects of incorporating an anti-aliasing filter in a migration operator, it is instructive to analyze the images generated when one single input trace is migrated into a cube. Figure 8.7 shows the result produced when one trace is migrated without the application of any anti-aliasing filter. The input trace was recorded at an offset of 2.4 km and the velocity was assumed to be constant and equal to 1.5 km/s. The image sampling was 20 m in each direction $(\Delta x_\xi = \Delta y_\xi = 20\text{m})$. Strong aliasing artifacts are visible in both the time slices and the vertical section.

Figure 8.8 shows the result produced when the same data trace is migrated with an appropriate anti-aliased operator. Notice that the aliasing artifacts disappear as the frequency content of the imaged reflectors progressively decreases, that is, as the dips increase. Figure 8.9 shows the result of migrating the same data trace when the effects of the wavelet stretch are not taken into account; that is, by setting $dt_D/d\tau_\xi = 1$. In this case, the anti-aliasing filter over-compensates for the image dip, and valuable resolution is lost at steep dips. Examining the time slice shown on the top of Figure 8.9, we notice that the loss of resolution is larger for regions of the migration ellipses with a steep dip along the cross-line direction.

Figure 8.10 shows the effects of image-space aliasing on the migrated results from the salt-dome data set shown in Figure 8.3. Figure 8.10a shows the migration results obtained without the use of any anti-aliasing and with $\Delta x_\xi = 36$ m; that is, the original sampling of the zero-offset data. Shallow dipping reflectors, and the steep high-frequency event at about 2.2 seconds, are badly aliased. The quality of the image improves when the image spatial sampling is halved to $\Delta x_\xi = 18$ m (Figure 8.10b). It is useful to notice that the traces in Figure 8.10a are exactly the same as the odd traces in Figure 8.10b. Therefore, image aliasing does not add noise to the image, it just makes the interpretation of the image more difficult and ambiguous. The application of the image-space anti-aliasing constraints, expressed in equation (8.5), further improves the image quality, in particular for the shallower events (Figure 8.10c). Although the image in Figure 8.10c is less noisy than the images in Figures 8.10a and 8.10b, it

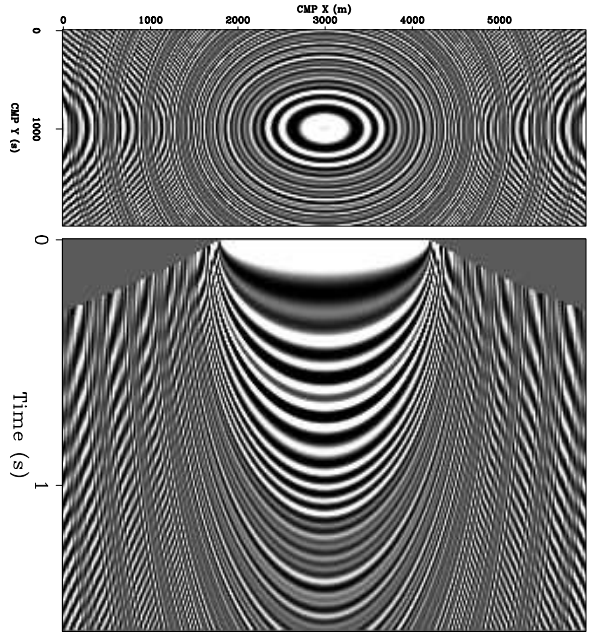


Figure 8.7: Image obtained by constant-velocity Kirchhoff migration without the use of any anti-aliasing ($\Delta x_\xi = \Delta y_\xi = 20$ m, $V = 1.5$ km/s, and Off $= 2.4$ km). The time slice is cut at Time=1 s, and the vertical section is cut at CMP Y=1000 m. alias-Imp-noantialias [ER,M]

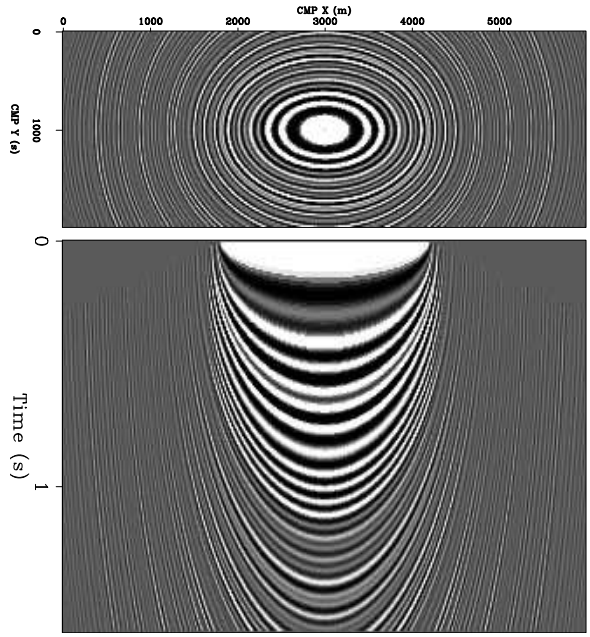


Figure 8.8: Image obtained by constant-velocity Kirchhoff migration with the use of image-space anti-aliasing. ($\Delta x_\xi = \Delta y_\xi = 20$ m, $V = 1.5$ km/s, and Off $= 2.4$ km). The time slice is cut at Time=1 s, and the vertical section is cut at CMP Y=1000 m. alias-Imp-antialias-equal [ER,M]

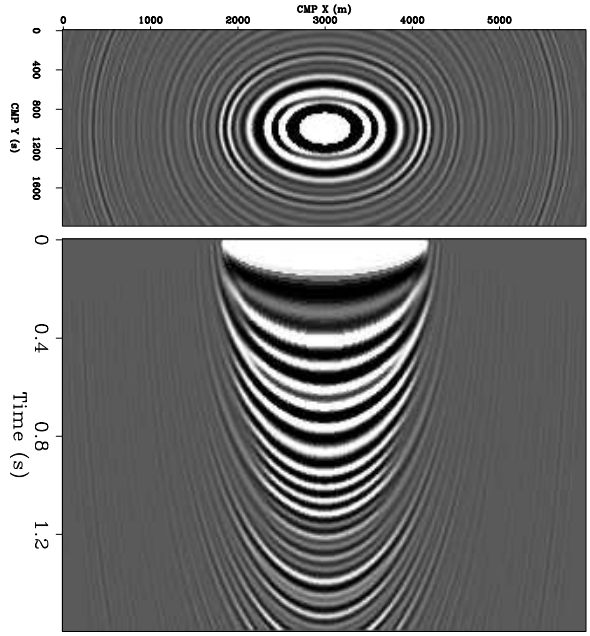


Figure 8.9: Image obtained when too strong anti-aliasing is applied because the effects of the wavelet stretch on the frequency content of the image are ignored. ($\Delta x_\xi = \Delta y_\xi = 20$ m, $V = 1.5$ km/s, and Off $= 2.4$ km). The time slice is cut at Time=1 s, and the vertical section is cut at CMP Y=1000 m. alias-Imp-nostretch [ER,M]

is still contaminated by aliasing artifacts. These artifacts are caused by **operator aliasing**, or data-space aliasing. The next section analyzes the causes of operator aliasing, and presents anti-aliasing constraints used to eliminate it.

8.2.2 Operator aliasing

The previous section analyzed aliasing in the image space, which occurs when the spreading representation of the imaging operator is aliased. Data-space aliasing, the counterpart of image-space aliasing, occurs when the gathering representation of the imaging operator is aliased in the data space. This data-space aliasing is commonly called **operator aliasing**. Operator aliasing is different than data aliasing, and it may occur even when the data are sufficiently sampled. When operator aliasing occurs, noise is actually added to the image, in contrast to the image-space aliasing case when the interpretability of the image is hampered by missing data, not additive noise. To illustrate the issues related to operator aliasing, the next set of figures analyze a simple example in which the data are plane waves and the summation operator is a slant stack.

Figures 8.11 and 8.12 show two plane waves. The plane waves are adequately sampled when the waveform is a 30 Hz sinusoid (Figure 8.11a), but the one with positive time dip is aliased when the waveform is a 60 Hz sinusoid (Figure 8.12a). The data aliasing is obvious in the time-space domain, where the data appears to be dipping in the opposite direction.

Data summation along a given trajectory is equivalent to a two-step process; first, the data are shifted so that the events align along the desired trajectory. Second, the traces are stacked together. In the case of the slant-stack operator, the summation trajectories are lines and the first step is equivalent to the application of linear moveout (LMO) with the desired dip.

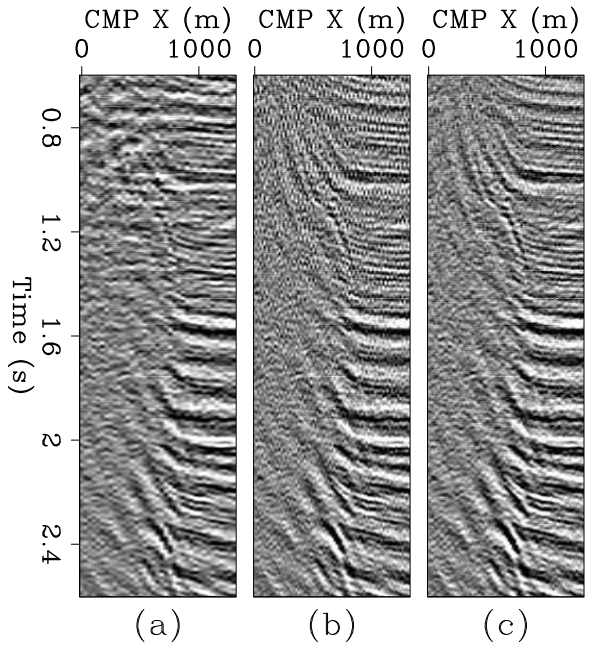


Figure 8.10: Migrated section from the Gulf of Mexico salt dome: (a) $\Delta x_\xi = 36$ m and no anti-aliasing, (b) $\Delta x_\xi = 18$ m and no anti-aliasing, (c) $\Delta x_\xi = 18$ m and image-space anti-aliasing. alias-Comp-noanti-mod [ER]

Figures 8.11b and 8.12b show the results of applying LMO with the slowness of the positive-dip plane wave to the data respectively in Figures 8.11a and 8.12a. Figures 8.11c and 8.12c show the results of applying LMO with the slowness of the negative-dip plane wave to the data respectively in Figures 8.11a and 8.12a. The traces on the right side of the sections are the results of stacking the corresponding data, each plane wave independently. The solid line is the stack of the positive-dip plane wave, and the dotted line is the stack of the negative-dip plane wave. At 30 Hz no aliasing occurs, and only the desired plane wave stacks coherently after applying LMO with either slowness. In contrast, at 60 Hz both plane waves stack coherently after applying LMO with either slowness (Figures 8.12b and 8.12c). In general, artifacts are generated when data that are not aligned with the summation path stack coherently into the image. This phenomenon is the cause of the **aliasing noise** that degrades the image when operator aliasing occurs.

To avoid adding aliasing noise to the image, we could low-pass-filter the input data according to the operator dips. A reasonable strategy is to limit the migration operator to the data components in the main band of the data spectrum (e.g. between the negative Nyquist and the positive Nyquist lines in the spectrum shown in Figure 8.6). This strategy leads to the application of the following anti-aliasing constraints on the temporal frequency of the summation operator:

$$\begin{aligned} \omega_d &< \frac{\pi}{\Delta x_D \left| p_x^{\text{op}} \right|}, \\ \omega_d &< \frac{\pi}{\Delta y_D \left| p_y^{\text{op}} \right|}; \end{aligned} \tag{8.10}$$

where Δx_D and Δy_D are the sampling rates of the data axes, and p_x^{op} and p_y^{op} are the operator dips in the data space, that is, the dips of the summation operator (i.e hyperboloid for migra-

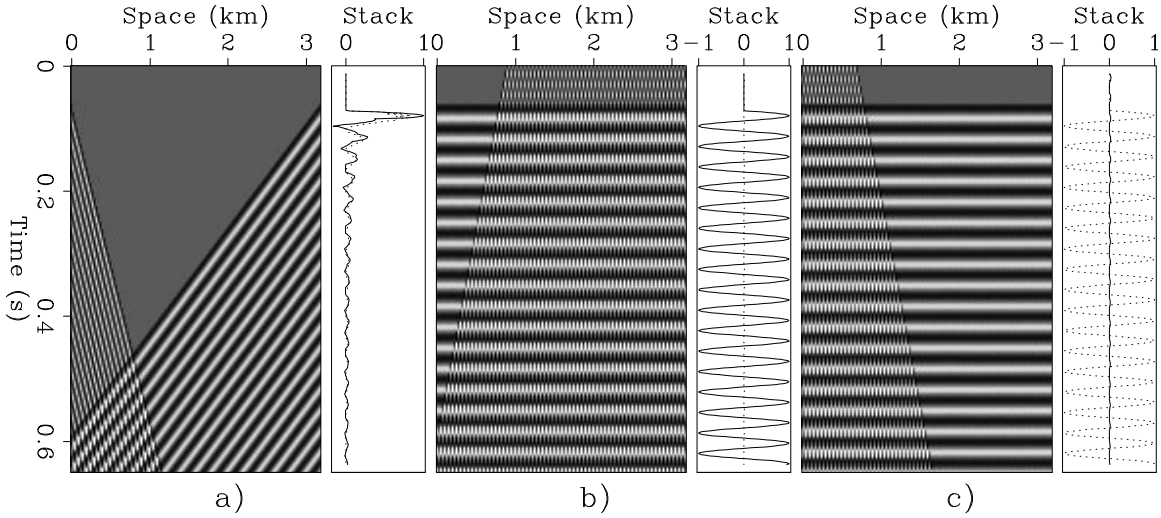


Figure 8.11: Two plane waves with dips of 0.5 s/km and -0.166 s/km, 30 Hz waveform and Δx_D=25 m: a) before linear moveout, b) after linear moveout with positive dip, c) after linear moveout with negative dip. The traces on the right side of the sections are the results of stacking the corresponding data, each plane wave independently. The solid line is the stack of the positive-dip plane wave, and the dotted line is the stack of the negative-dip plane wave. alias-Fig-30-sec-rot [ER]

tion). These, or equivalent, relationships have been presented by a number of authors (Gray, 1992; Bevc and Claerbout, 1992; Lumley et al., 1994; Abma et al., 1998).

The application of these constraints to the simple slant stack example of Figures 8.11 and 8.12 (Δx_D=25 m, p_+^{op}=0.5 s/km, p_-^{op}=-0.166 s/km) would result in the suppression of all the frequencies above 40 Hz when summing along the positive dip, and the suppression of all the frequencies above 120 Hz when summing along the negative dip. At 60 Hz, both the signal and the operator aliasing noise would be suppressed when summing along the positive dip (p_+^{op}=0.5 s/km), but both would be preserved when summing along the negative dip (p_-^{op}=-0.166 s/km). The outcome of applying the "standard" anti-aliasing criteria is thus sub-optimal.

To examine further the relationship between operator aliasing and the dip bandwidth in the data, we consider the two plane waves shown in Figure 8.13. In this case, the two plane waves have a 60 Hz waveform, as in Figure 8.12, but the second plane wave is dipping less than in the previous case (p_-^{op}=-.025 s/km). The two plane waves have conflicting dips, but even at 60 Hz they do not interfere with each other during summation, because the difference between the conflicting dips is smaller. However, the anti-aliasing conditions expressed in equation (8.10) would apply exactly in the same way as in the previous case. Their application would cause the suppression of the signal when summing along the positive dip, even if the stack is free of operator aliasing noise.

The previous examples show that the anti-aliasing conditions expressed in equation (8.10) can be either too restrictive or too lax. Although these conditions may be correct for several

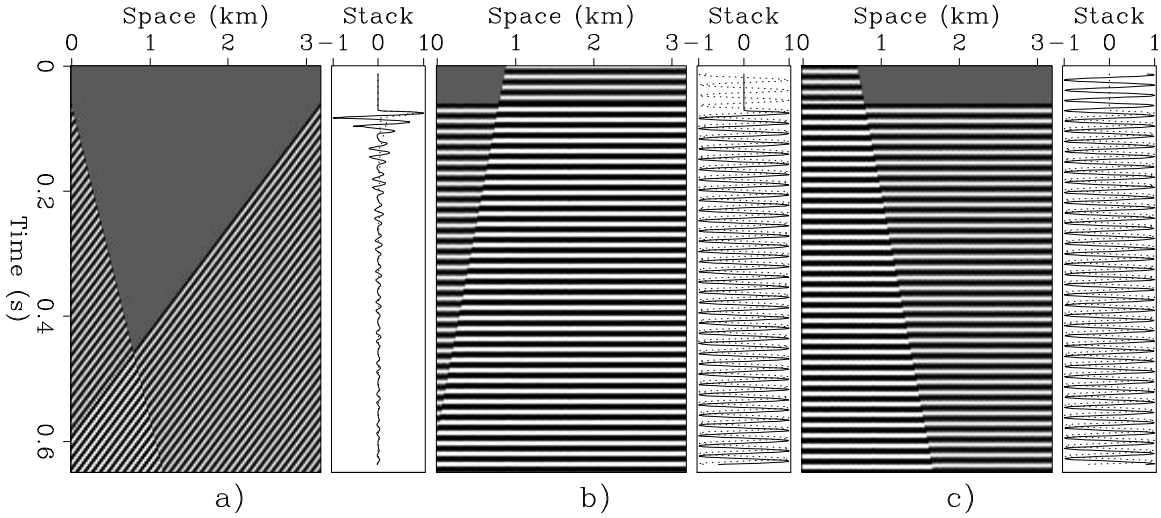


Figure 8.12: Two plane waves with dips of 0.5 s/km and -0.166 s/km, 60 Hz waveform and Δx_D=25 m: a) before linear moveout, b) after linear moveout with positive dip, c) after linear moveout with negative dip. The traces on the right side of the sections are the results of stacking the corresponding data, each plane wave independently. The solid line is the stack of the positive-dip plane wave, and the dotted line is the stack of the negative-dip plane wave. alias-Fig-60-sec-rot [ER]

important cases, they do not take into account the fact that operator aliasing depends on the presence of conflicting, and potentially aliased, dips in the data. They implicitly assume that no dips are aliased in the data. When this implicit assumption is violated, as for the positive dip in the previous example, it can be harmful to apply the "standard" anti-aliasing conditions. To avoid the problems with "standard" anti-aliasing conditions, Biondi (2001) introduced more general anti-aliasing conditions that take into account the existence of conflicting dips. Pica (1996) presented a model-based anti-aliasing method that is similar to the one proposed by Biondi (2001), though it yields different anti-aliasing conditions.

Figure 8.14 graphically shows the idea at the basis of more general anti-aliasing criteria. The shaded area is the widest possible area of the spectrum where there is no interference with aliased energy; that is, the data components in the shaded areas can be safely imaged without generating operator-aliasing noise. The boundaries of this "pass region" are determined as a function of the minimum data dips, $\mathbf{p}^{\min} = \left(p_x^{\min}, p_y^{\min}\right)$, and maximum data dips, $\mathbf{p}^{\max} = \left(p_x^{\max}, p_y^{\max}\right)$. On the negative wavenumber side, the pass region is bounded by a line representing a replica of the maximum dip line. That is, it is bounded by a line that starts at -2k_{nyq} (where k_{nyq} is the Nyquist wavenumber) and has a slope corresponding to the maximum dip. Similarly, on the positive wavenumber side, the pass region is bounded by a line starting at 2k_{nyq} and with slope corresponding to the minimum dip.

The boundaries for the pass region are analytically expressed by the following inequalities:

$$p_x^{\max} - \frac{2\pi}{\omega_d \Delta x_D} < p_x^{\text{op}} < p_x^{\min} + \frac{2\pi}{\omega_d \Delta x_D}, \tag{8.11}$$

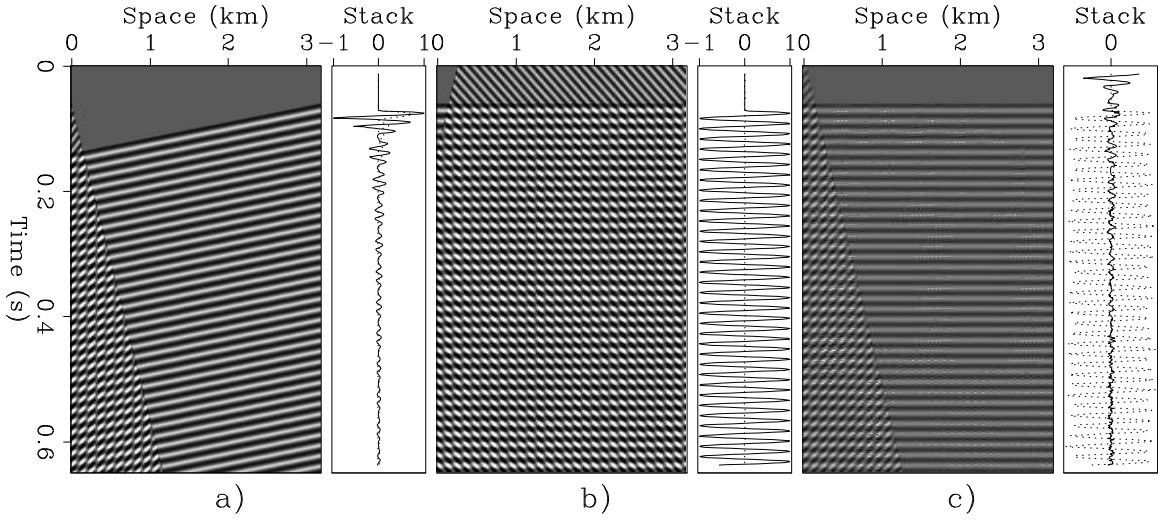


Figure 8.13: Two plane waves with dips of 0.5 s/km and -0.025 s/km, 60 Hz waveform and Δx_D=25 m: a) before linear moveout, b) after linear moveout with positive dip, c) after linear moveout with negative dip. The traces on the right side of the sections are the results of stacking the corresponding data, each plane wave independently. The solid line is the stack of the positive-dip plane wave, and the dotted line is the stack of the negative-dip plane wave. alias-Fig-60-gath-rot [ER]

$$p_y^{\max} - \frac{2\pi}{\omega_d \Delta y_D} < p_y^{\text{op}} < p_y^{\min} + \frac{2\pi}{\omega_d \Delta y_D}. \tag{8.12}$$

The inequalities expressed in equations (8.11-8.12) can be easily recast as anti-aliasing constraints on the maximum frequency in the data, as follows:

$$\omega_d < \frac{2\pi}{\Delta x_D \left(p_x^{\text{op}} - p_x^{\min}\right)}, \tag{8.13}$$

$$\omega_d < \frac{2\pi}{\Delta x_D \left(p_x^{\max} - p_x^{\text{op}}\right)}, \tag{8.14}$$

$$\omega_d < \frac{2\pi}{\Delta y_D \left(p_y^{\text{op}} - p_y^{\min}\right)}, \tag{8.15}$$

$$\omega_d < \frac{2\pi}{\Delta y_D \left(p_y^{\max} - p_y^{\text{op}}\right)}. \tag{8.16}$$

The constraints expressed in equations (8.13–8.16) can be used as alternatives to, or in conjunction with, the constraints expressed in equation (8.10), to design anti-alias summation operators.

Comparing equation (8.10) with equations (8.13–8.16), we can notice that the two sets of constraints are equivalent when, for each frequency, the data dip limits $p^{\min}$ and $p^{\max}$ are respectively set to $-k_{\text{nyq}}/\omega_d$ and k_{nyq}/ω_d. This is easy to verify by substituting $p_x^{\min} = -\pi/(\Delta x_D \omega_d)$ into equation (8.13), and $p_x^{\max} = \pi/(\Delta x_D \omega_d)$ into equation (8.14), and performing analogous substitutions into equations (8.15–8.16). This result implies that when we

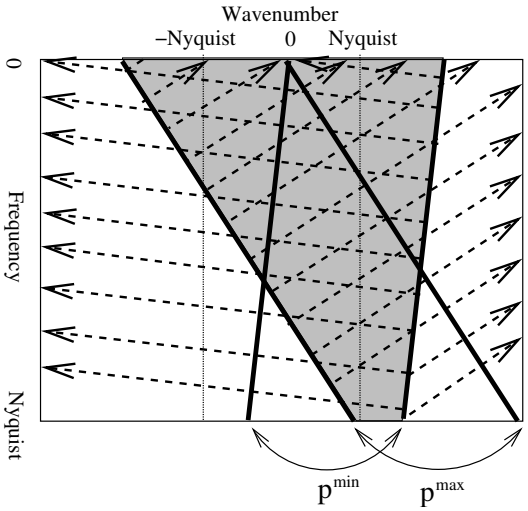


Figure 8.14: Graphic representation of the general anti-aliasing criteria [equations (8.11-8.12)]. The shaded area correspond to the "pass region". It is the widest possible area of the spectrum where there is no interference with aliased energy. On the negative wavenumber side, the pass region is bounded by a line that starts at -2k_{nyq} and has a slope corresponding to the maximum dip p^{max}. On the positive wavenumber side, the pass region is bounded by a line starting at 2k_{nyq} and with slope corresponding to the minimum dip p^{min}. The dashed lines cover the areas where at least one of of the inequalities in equation (8.11) is satisfied. alias-Spec-sketch [ER]

can assume that there is no spatial aliasing in the data, the new anti-aliasing conditions are equivalent to the "standard" conditions.

Applying the constraints of equations (8.13–8.16) to the plane-waves example of Figures 8.11–8.13 leads to the "correct" choice of frequency cutoffs. In the case with wide dip bandwidth (Figures 8.11–8.12; p_+^{op}=0.5 s/km, p_-^{op}=-0.166 s/km), the frequency cutoff is 60 Hz when summing along both the positive and the negative dips. The plane waves in Figure 8.12 are thus just outside the pass region and should be suppressed. In the case with narrow dip bandwidth (Figure 8.13; p_+^{op}=0.5 s/km, p_-^{op}=-0.025 s/km), the frequency cutoff is about 84 Hz when summing along both the positive and the negative dips. Therefore, both plane waves in Figure 8.12 can be safely imaged.

The dip limits of the data, $\mathbf{p}^{\text{min}}$ and $\mathbf{p}^{\text{max}}$, can vary in both time and space, according to the local dips in the data. Therefore, the anti-aliasing filters applied to the data as a consequence of the constraints in equations (8.13–8.16) can be fairly complex, and dependent on factors including local dips, time, and spatial coordinates. If no *a priori* knowledge of the local dips is available, and the summation is carried out along the midpoint axes, twice the inverse of propagation velocity is a reasonable bound on the absolute values of both $\mathbf{p}^{\text{min}}$ and $\mathbf{p}^{\text{max}}$. In contrast, in the case where the summation is performed along the offset axes, as for CMP stacks, $\mathbf{p}^{\text{min}}$ can be safely assumed to be positive, and at worst equal to zero. Therefore, imaging operators that sum data over the offset axes, such as NMO followed by stack, are less prone to spatial aliasing than imaging operators that sum data over midpoints, such as migration.

The most substantial benefits of applying the more general constraints expressed in equations (8.13–8.16) are achieved when asymmetric bounds on the dips in the data enable imaging the high-frequency components that are present in the data as aliased energy, and still minimizing the aliasing artifacts. These components would be filtered out if the constraints in equation (8.10) were applied. Asymmetric bounds on the data dips are realistic in the important application of imaging steep salt-dome flanks, as in the Gulf of Mexico data set shown above. In this case, we can assume that the negative time dips in the data are small. According to equations (8.11-8.12), the increase in $p^{\min}$, i.e., the decrease of its absolute value, increases the maximum positive dip that can be imaged without operator aliasing. In practice, the application of the generalized constraints in equations (8.13–8.16), with $\mathbf{p}^{\min} \neq \mathbf{p}^{\max}$, causes the migration operator to be asymmetric, with dip bandwidth dependent on reflector direction.

Figures 8.15 and 8.16 show an example of the effects of asymmetric dip bounds on the migration operator. The image sampling is the same as in Figure 8.8 $\left(\Delta x_\xi = \Delta y_\xi = 20\text{m}\right)$ for both images, but the data sampling is assumed to be coarser than the image sampling by a factor of two; that is, $\Delta x_D = \Delta y_D = 40$m. When the constraints in equation (8.10) are applied (see Figure 8.15), the operator has lower resolution than in Figure 8.8. But if we assume that $p_x^{\min} = 0$ s/m, and apply the constraints in equations (8.13–8.16), the pass region for the operator is augmented by the band comprised between k_{nyq} and $2k_{\text{nyq}}$ (see Figure 8.14). Consequently, in Figure 8.16 the positive time dips are imaged with the same resolution as in Figure 8.15.

The link between operator anti-aliasing and image anti-aliasing

The constraints derived to avoid operator aliasing and image aliasing are different in general; both sets of constraints must be respected if a high-quality image is to be assured. However, the two sets of constraints are equivalent in many practical situations. In these situations only one set of constraints needs to be applied to assure that the image is both adequately sampled and free of operator-aliasing noise. The analysis is simple for time migration of zero-offset or constant-offset data, where we can assume that the summation surfaces $t_D = t_D(\xi, x_m, y_m)$ are functions only of the relative distance between the image-point ξ and the midpoint of the data trace (x_m, y_m).

The first condition for linking image aliasing to operator aliasing is that the data are not spatially aliased, and thus the operator anti-aliasing constraints expressed in equation (8.10) are equivalent to the constraints expressed in equations (8.13–8.16). Comparing the constraints for operator anti-aliasing [equation (8.10)] with the constraints for image anti-aliasing [equation (8.5)], we can easily notice that a necessary condition for their being uniformly equivalent is that the data sampling rates Δx_D and Δy_D be equal to the image sampling rates Δx_ξ and Δy_ξ.

The other necessary conditions are that $p_x^{\text{op}} = p_x^\xi dt_D/d\tau_\xi$ and $p_y^{\text{op}} = p_y^\xi dt_D/d\tau_\xi$; that is, that the ratio between the operator dips in the data space and the operator dips in the image space are exactly equal to the wavelet-stretch factor discussed in the section on image aliasing. These conditions are fulfilled in the important case of spatially invariant imaging

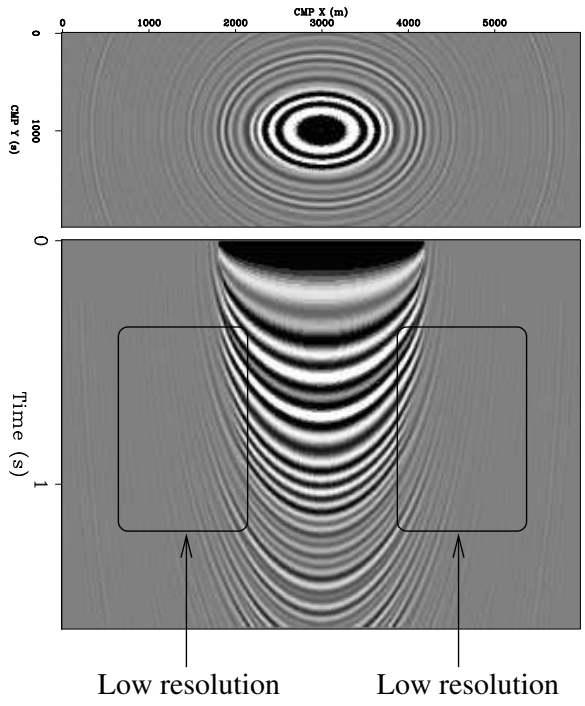


Figure 8.15: Image obtained by constant-velocity Kirchhoff migration using "standard" anti-aliasing. ($\Delta x_\xi = \Delta y_\xi = 20$ m, $\Delta x_D = \Delta y_D = 40$ m, $V = 1.5$ km/s, and Off $= 2.4$ km). The time slice is cut at Time=1 s, and the vertical section is cut at CMP Y=1000 m. alias-Imp-antialias-nodirect-ann [ER]

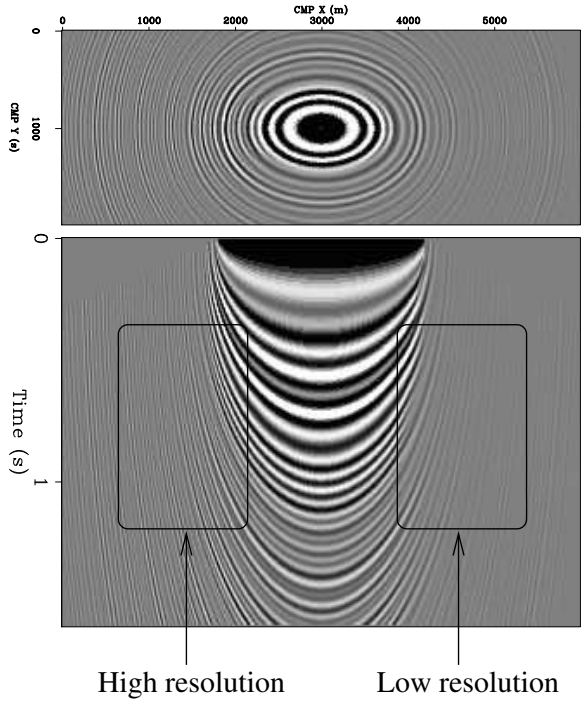


Figure 8.16: Image obtained by constant-velocity Kirchhoff migration with use of the "directed" anti-aliasing, assuming $p_x^{\min} = 0$ s/m ($\Delta x_\xi = \Delta y_\xi = 20$ m, $\Delta x_D = \Delta y_D = 40$ m, $V = 1.5$ km/s, and Off $= 2.4$ km). The time slice is cut at Time=1 s, and the vertical section is cut at CMP Y=1000 m. alias-Imp-antialias-direct-ann [ER]

operators, as can be shown by applying the chain rule to the derivative of the summation surfaces $t_D = t_D(\xi, x_m, y_m)$, with respect to the midpoint coordinates of the data trace (x_m, y_m):

$$\begin{aligned} p_x^{\text{op}} = \frac{dt_D}{dx_m} &= \frac{dt_D}{d\tau_\xi}\frac{d\tau_\xi}{dx_\xi}\frac{dx_\xi}{dx_m} = \frac{dt_D}{d\tau_\xi}p_x^\xi, \\ p_y^{\text{op}} = \frac{dt_D}{dy_m} &= \frac{dt_D}{d\tau_\xi}\frac{d\tau_\xi}{dy_\xi}\frac{dy_\xi}{dy_m} = \frac{dt_D}{d\tau_\xi}p_y^\xi. \end{aligned} \tag{8.17}$$

The equalities in (8.17) rely on the horizontal invariance of the imaging operator; they require that the derivatives of the horizontal coordinates of the image point with respect to the horizontal coordinates of the data trace (dx_ξ/dx_m and dy_ξ/dy_m) be equal to one. For migration, these conditions are strictly fulfilled only in horizontally layered media, though they are approximately fulfilled when the migration velocity varies smoothly. Another way to interpret this result is to recall that migration in the frequency-wavenumber domain is equivalent to a down-shift in temporal frequency at constant wavenumber (Stolt, 1978); that is, if both the data and the operator are not aliased, the image is guaranteed to be not aliased. Because the equalities in (8.17) do not require any other assumptions on the shape of the summation surfaces, the same link between operator aliasing and image aliasing exists for all spatially-invariant integral operators, such as DMO (Hale, 1991) and AMO (Biondi et al., 1998).

The distinction between operator aliasing and image aliasing can thus be safely ignored when well sampled zero-offset data are *time* migrated (Bevc and Claerbout, 1992; Lumley et al., 1994; Claerbout, 1995). In this case only one set of constraints between equation (8.10) and equation (8.5) must be applied. However, the distinction should be respected when irregularly sampled prestack data are *depth* migrated. It is also important when *a priori* assumptions about the dips in the data permit less stringent operator anti-aliasing constraints to be set, so that reflectors can be imaged with high resolution but without operator-aliasing artifacts. In these cases it is important to use a finer spatial sampling for the image than for the data to avoid aliasing the image. The next section shows an example of this situation.

High-resolution imaging of salt flanks

The high-resolution imaging of salt-dome flanks is an important application of the general anti-aliasing method described in the previous section. In this case, we can often assume that in the proximity of the salt flanks, the data contains little or no energy dipping in the direction opposite to the reflections from the flanks. According to the theory developed in the previous section, this assumption enables the salt flanks to be imaged with higher resolution than was otherwise possible. However, it is important to be aware that when the image frequency content is increased by the application of the constraints in equations (8.13–8.16), rather than those in equation (8.10), we run the risk of aliasing the image. As discussed in the previous section, the conditions that avoid operator aliasing do not guarantee the avoidance of image aliasing. Therefore, the constraints to avoid image aliasing [equation (8.5)] must also be taken into account, and the image sampling must be reduced so that both goals – avoiding image aliasing and preserving high-frequency components – can be achieved. Because of image-aliasing considerations, the images of the salt flanks from the Gulf of Mexico data that

are shown in the following figures are sampled twice as densely $\left(\Delta x_\xi = \Delta y_\xi = 18\text{m}\right)$ as the zero-offset data ($\Delta x_D = \Delta y_D = 36\text{m}$).

The first step in the process of achieving high-resolution by application of the general anti-aliasing condition, is to determine the appropriate values for the bounds on the data dips. For the sake of simplicity, I choose constant bounds for this example; that is, $p_x^{\min} = -0.082$ s/km, $p_x^{\max} = 0.48$ s/km, $p_y^{\min} = -0.8$ s/km, $p_y^{\max} = 0.8$ s/km. In more geologically complex cases, it may be advantageous to allow the data-dip bounds to vary in both time and space. Figure 8.17 shows the same data spectrum as Figure 8.6; the dashed lines superimposed onto the spectrum cover the areas determined by the inequalities of equations (8.11) and (8.12), according to the chosen bounds for $p_x^{\min}$ and $p_y^{\max}$. The areas that honor all anti-aliasing constraints, and that thus represent data components used by the imaging, are covered by crossing dashed lines. A large swath of the aliased energy with positive time dips is used by the high-resolution imaging, whereas it would be discarded if standard anti-aliasing methods were used.

The improvements in image resolution that are made possible by the high-resolution anti-aliasing method are demonstrated in Figures 8.18–8.20. Figure 8.18a shows the results of 3-D post-stack migration obtained without any anti-aliasing. Figure 8.18b shows the results obtained when the high-resolution anti-aliasing constraints in equations (8.13–8.16) are applied. And Figure 8.18c shows the results produced when the standard anti-aliasing constraints in equation (8.10) are applied. The differences between the results are better appreciated by a comparison of windows zooming into smaller parts of the sections. Figure 8.19 shows the comparison for the shallower part of the section. The image obtained without anti-aliasing is uninterpretable because of the strong aliasing noise. The image obtained with the high-resolution method shows better resolution of several dipping reflectors and of the steep salt flank. Figure 8.20 demonstrates (for the same area shown in Figure 8.4) that the high-frequency sediment truncation against the salt flank (CMP X=700 m and Time=2.2 s) is well resolved in the image obtained with the high-resolution method, whereas it is poorly resolved in the image obtained with the traditional methods.

8.3 Aliasing in wavefield-continuation migration

For a long time, wavefield-continuation migration has been considered immune to operator aliasing artifacts. In contrast with Kirchhoff operators, which can sum the data on the side bands of the spectra in addition to the central band (see discussion on Kirchhoff operator aliasing in Section 8.2.2), wavefield-continuation operators are intrinsically limited to propagating the wavefield in the central band of the discretely sampled data. However, wavefield-continuation migrations generate artifacts similar to the ones generated by Kirchhoff migrations when we interleave the recorded traces with zero traces. This interleaving is seldom done for zero-offset migration, but it is done, either implicitly or explicitly, for the most prestack migration processes.

An example of implicit interleaving is when we perform shot-profile migration on data that have coarser spatial sampling for the shots than for the receivers; a characteristic that is prevalent in 3-D acquisition geometries. If we propagate both the source wavefield and the

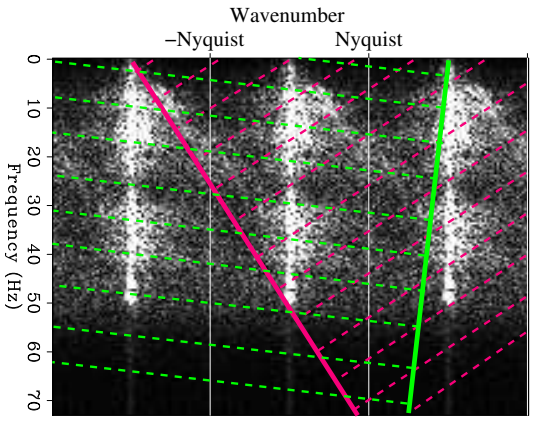


Figure 8.17: Frequency-wavenumber spectrum of a data window (same as in Figure 8.6). The dashed lines superimposed onto the spectrum cover the areas determined by the inequalities of equations (8.11) and (8.12). The areas that respect all the operator anti-aliasing constraints, and thus that represent data components used by the imaging, are covered by dashed lines along both directions. alias-Wind-spec-sep1 [NR]

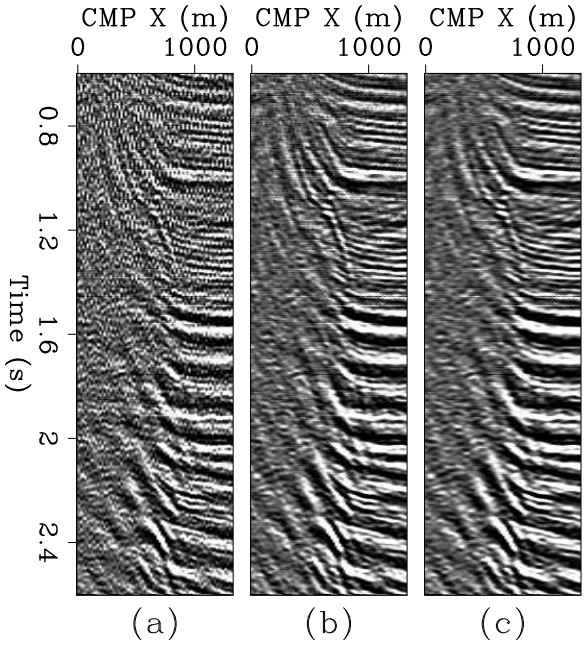


Figure 8.18: 3-D migrations of a salt-dome flank in the Gulf of Mexico: (a) migration obtained without any anti-aliasing filter, (b) migration obtained with the application of the high-resolution anti-aliasing filter, and (c) migration obtained with the application of a "standard" anti-aliasing filter. alias-Comp-WL-anti [ER,M]

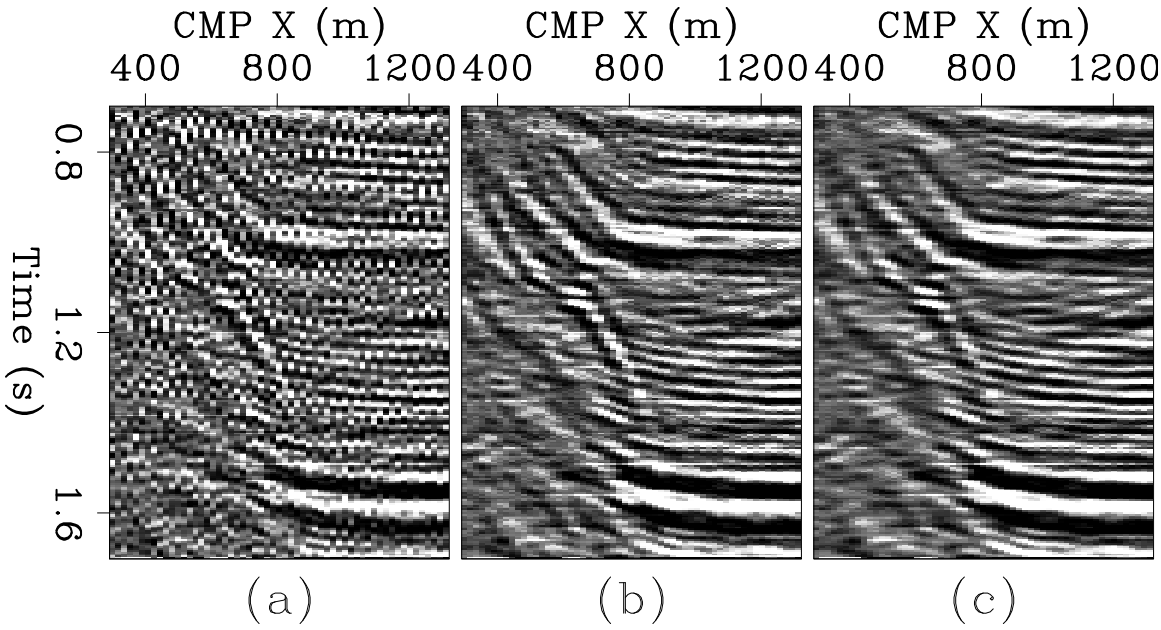


Figure 8.19: Magnification of the shallower part of the 3-D migrations of a salt-dome flank in the Gulf of Mexico shown in Figure 8.18: (a) migration obtained without any anti-aliasing filter, (b) migration obtained with the application of the high-resolution anti-aliasing filter, and (c) migration obtained with the application of a "standard" anti-aliasing filter. alias-Comp-WT-anti [ER,M]

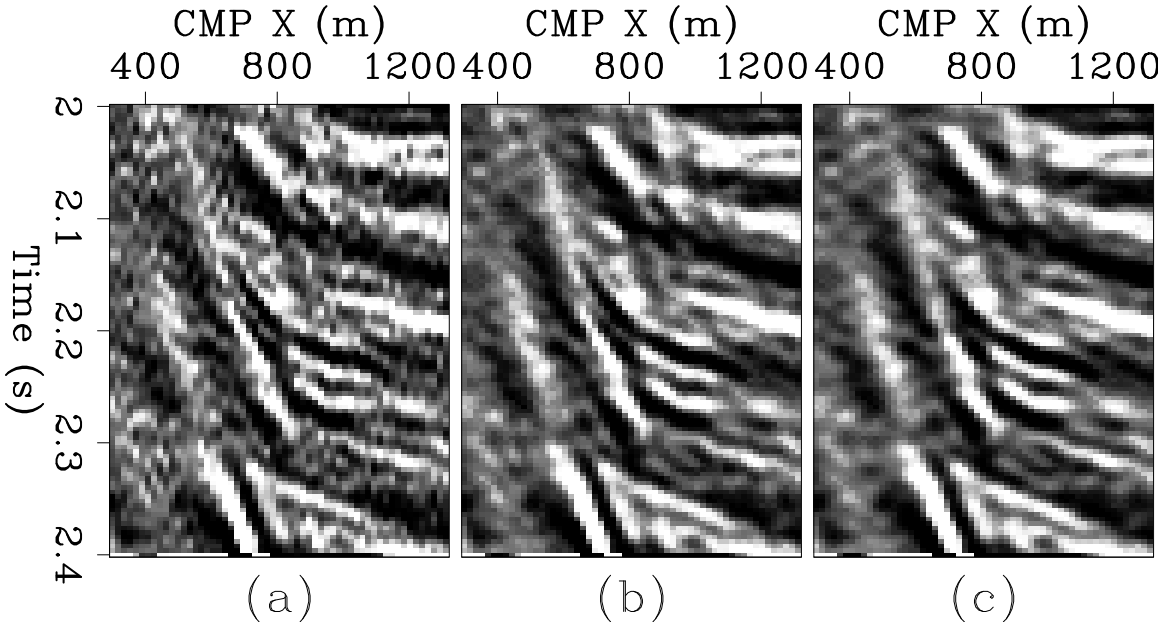


Figure 8.20: Magnification of the deeper part of the 3-D migrations of a salt-dome flank in the Gulf of Mexico shown in Figure 8.18: (a) migration obtained without any anti-aliasing filter, (b) migration obtained with the application of the high-resolution anti-aliasing filter, and (c) migration obtained with the application of a "standard" anti-aliasing filter. alias-Comp-WB-anti [ER,M]

receiver wavefield on a grid sampled with the receiver sampling rate (finer sampling), and then evaluate the image using the standard imaging condition in equation (4.7), we are effectively interleaving the shot axes with as many zero traces as needed to match the sampling rate of the receiver axes.

Basic Fourier-analysis theory teaches us that interleaving the data with zero traces is equivalent to replicating the central band in as many side bands as the interleaving factor (Bracewell, 1999). Figures 8.21 and 8.22 illustrate the effects of interleaving applied to the Gulf of Mexico data set used in the previous section. The left panel in Figure 8.21 shows the same data window shown in Figure 8.5, but with three zero traces inserted for each live trace. The right panel in Figure 8.21 shows the frequency-wavenumber spectrum of the interleaved data. Because the interleaved data sampling rate is four times finer than the sampling rate of the original data, the spatial bandwidth is also four times larger. However, by adding zero traces we have not added any new information, and the high (positive and negative) wavenumbers contain copies of the main band in the original data.

When we migrate the interleaved data using a wavefield-continuation algorithm, the energy in the side copies of the main band is propagated incorrectly, and generates "aliasing" artifacts in the image. Figure 8.22 shows the results of this experiment. Figure 8.22a displays the image generated by performing zero-offset downward-continuation migration on the data which were obtained by interpolating the original traces onto a grid four times finer than the original sampling grid. Figure 8.22b displays the image generated by using the same migration algorithm and migration velocity, but on the data interleaved with zero traces shown in Figure 8.21. The addition of the spurious copies of the energy at high (positive and negative) wavenumbers creates the aliasing artifacts visible in Figure 8.22b.

As for Kirchhoff migration, adding the contributions of the side bands to the image improves the resolution of steeply-dipping sediments that are truncating against the salt body. Figure 8.22c shows the difference between the panels in Figure 8.22a and 8.22b. The difference panel is mostly incoherent noise, but there are few coherent events corresponding to the same dipping reflectors high-lighted in the previous section; for example, the sediment truncation at about 3,500 meters depth is clearly visible in Figure 8.22c. Therefore, with wavefield-continuation migration we face the same dilemma discussed in Section 8.1.1 for Kirchhoff migration: how to optimally set the trade-off parameters between aliasing artifacts and image resolution.

The aliasing artifacts that interleaving adds to the image in Figure 8.22b are similar to the aliasing artifacts visible in Figure 8.18a, which was obtained by Kirchhoff migration without applying any anti-aliasing filter. Although the two images are similar, the amount of aliased energy is lower in the section obtained by wavefield-continuation migration than in the section obtained by Kirchhoff migration. The image obtained by wavefield-continuation is equivalent to the image obtained by applying anti-aliased Kirchhoff migration using the "standard" anti-aliasing constraints, and assuming the sampling rate to be equal to the sampling rate of the interleaved data. That is, by applying the constraints expressed in equations (8.13–8.16), with the assumption that the the data dip limits $p^{\min}$ and $p^{\max}$ are respectively set to $-k_{\text{Nyq}}/4\omega_d$ and $k_{\text{Nyq}}/4\omega_d$.

The previous example shows that the aliasing artifacts caused by interleaving zero traces with the data are similar to the artifacts caused by operator aliasing with Kirchhoff migration. Furthermore, if we interleave data that are not aliased, and then migrate the interleaved data using a wavefield-continuation migration, aliasing artifacts will be present in the image. Conversely, by interleaving the original data with zero traces, we enable the migration algorithm to migrate wavefield components that were aliased in the original data, and thus we have the possibility of increasing the resolution of the image. In summary, the aliasing caused by interleaving data with zero traces has similar characteristics to Kirchhoff-migration operator aliasing, and consequently I will refer to this kind of aliasing as **wavefield-continuation operator aliasing**.

Operator aliasing is seldom a problem for zero-offset migration, but it is widespread in 3-D prestack migration. Effects of operator aliasing are particularly noticeable in the prestack partial images (e.g. ADCIGs) created by wavefield-continuation methods. The effects on the stacked image are less visible, because the artifacts tend to be incoherent over aperture angles, and thus the redundancy in the data contributes to the attenuation of the artifacts in the stacked image. The next section analyzes operator aliasing for prestack migration and presents some simple solution for reducing the effects of operator aliasing on the final image.

8.3.1 Aliasing of wavefield-continuation imaging operators

To understand the effects of operator aliasing on prestack migration by wavefield-continuation methods, we analyze sketches of the wavenumber-domain footprint of the whole data set on a plane (at least in 2-D). Because source-receiver migration and shot-profile migration are equivalent, as demonstrated in Section 4.2.3, our analysis is valid for both source-receiver migration and shot-profile migration.

Before analyzing the effects of undersampling the field coordinates (either shots or receivers), we need to analyze the effects of the transformation from source-receiver coordinates into midpoint-offset coordinates. This analysis is useful because the data are recorded, and often are propagated, in source-receiver coordinates, but the final image is built in midpoint-offset coordinates for both source-receiver migration [imaging condition in equation (4.12)] and shot gather migration [imaging condition in equation (4.7)]. The sketches in Figure 8.23 are intended to support this analysis.

The sketch on the left in Figure 8.23 represents a 2-D prestack data set (one live trace per dot) equally sampled in shots and receivers. The middle square in the sketch on the right of the figure represents the wavenumber footprints for the same data set shown on the left. However, the re-sorting of a data set with equal sampling intervals in source and receivers, $\Delta_s = \Delta_g$, into midpoint-offset coordinates is not straightforward. If we were willing to discard half of the original traces, we could keep the original sampling interval; that is $\Delta_m = \Delta_h = \Delta_s$. The inner square in the sketch on the right of the figure represents the wavenumber footprints for this case. Since we want to preserve all the traces, we must sample the midpoint and offset axes with half of Δ_s; that is $\Delta_m = \Delta_h = \Delta_s/2$. To avoid aliasing caused by interleaving, we should interpolate both the source and receiver axes before applying the transformation. The

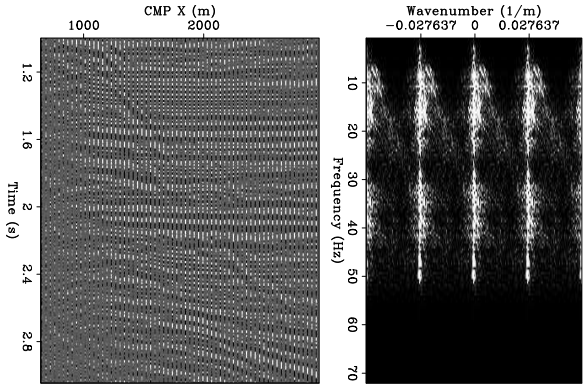


Figure 8.21: The same data window of the Gulf of Mexico data shown in Figure 8.5, but with three zero traces inserted for each live trace (left panel); frequency-wavenumber spectrum of the interleaved data (right panel). alias-Wind-comb-data-spec [ER]

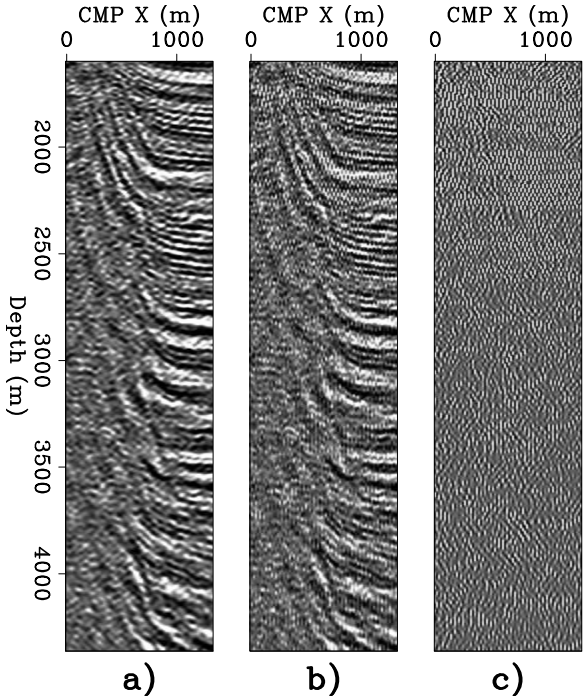


Figure 8.22: Images obtained by performing zero-offset downward-continuation migration: (a) on the data obtained by interpolating the original data on a grid four times finer, (b) on the data obtained by interleaving the original data, and with three zero traces for each live trace, and (c) difference between panel a) and panel b). alias-Mig-interp-comb-diff-overn [ER]

outer square in the sketch on the right of the figure represents the wavenumber footprints for this case. Notice that the inner and middle square in the right sketch in Figure 8.23 represent datasets sampled equally in their respective coordinate systems, but the inner square covers a smaller area of the wavenumber plane. This apparent contradiction is resolved if we note the transformation between a source-receiver coordinate system and a midpoint-offset coordinate system is not a simple rotation, but includes also a stretch of the axes by a $\sqrt{2}$ factor. Of course, this also true for the transformation from source-receiver wavenumbers, $(\mathbf{k_s}, \mathbf{k_g})$, to midpoint-offset wavenumbers, $(\mathbf{k_m}, \mathbf{k_h})$.

The considerations above apply not only to data sorting, but also to building the image using the imaging conditions in equations (4.12) and (4.7). Since we are not willing to throw away potentially valuable data, I assume that we build the image on the outer square in the right sketch in Figure 8.23. The triangular areas at the corners of the outer square are likely to be mostly empty in the prestack image, because the wavefields are band-limited to the middle square. In this case, with with equal sampling of the source and receiver axes, and band-limited source and receiver wavefields, the outer square does not contain any side band of the signal, and thus the image will not be aliased (assuming that the wavefield was adequately sampled at the start).

Figure 8.24 represents the case when the shots are sampled four times less densely than the receivers, as depicted by the sketch on the left of the figure. As shown in the sketch on the right of Figure 8.24, the wavenumber footprint of the image (outer square) now contains the aliased bands of the source axis (filled with an hexagonal pattern in the figure). These side bands contain both useful energy and energy that creates aliasing artifacts. There are two possible strategies for imaging the energy present in these side bands.

A "conservative" strategy would be to eliminate the contributions to the image from the side bands. This task can be achieved by spatially band-passing the wavefield either at the imaging point or directly at the source. If the source is band-passed, and shot-profile migration is employed, computational efficiency could be further increased by propagating the source wavefield on the coarse grid defined by the shot spacing. However, this computationally efficient solution would not be equivalent to band-passing the source when the velocity function has strong lateral velocity variation. Unfortunately the conservative approach causes the progressive disappearance of dipping reflectors from the image as the source undersampling increases.

Figure 8.25 demonstrates this loss of energy from the migrated image of the dipping reflectors in the shallow section of the Marmousi model (Versteeg and Grau, 1990). Figure 8.25a shows the results of prestack migration of all the shots in the data; that is $\Delta_s = \Delta_g = 25$ meters. The other panels in the figure show the progressive degradation of the image as the shots are progressively decimated by a factor of two; that is, $\Delta_s = 50$ meters for panel b), $\Delta_s = 100$ meters for panel c), and $\Delta_s = 200$ meters for panel d). All these images were obtained by source-receiver migration using the nominal sampling interval of 25 meters for the wavefield propagations. The decimated results were computed by interleaving the decimated shots with zero traces along the shot axis, followed by band-passing along the source axis before migration. Performing source-receiver migration after this process of interleaving and band-passing

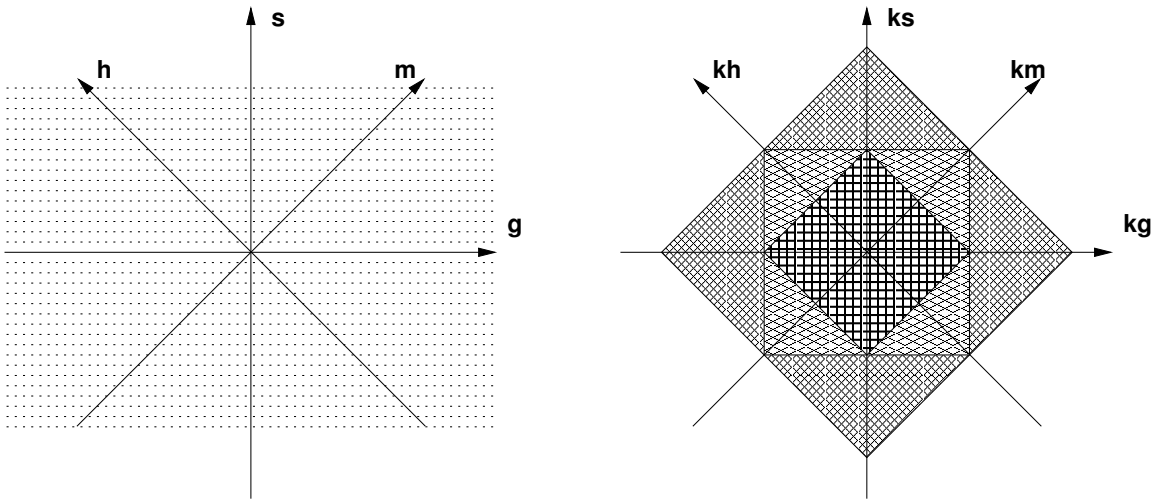


Figure 8.23: Graphical representation of a data set with $\Delta_s = \Delta_g$ in the space domain (left panel), and in the wavenumber domain (right panel). In the right panel, the inner square represents the data set in the midpoint-offset wavenumber domain with $\Delta_m = \Delta_h = \Delta_s$. The middle square represents the data set in the source-receiver wavenumber domain with $\Delta_s = \Delta_g$. The outer square represents the data set in the midpoint-offset wavenumber domain with $\Delta_m = \Delta_h = \Delta_s/2$. alias-sr-basic-v2 [NR]

is equivalent to performing shot-profile migration with a band-passed source function. Notice that because the band-pass was performed after decimation (I assumed that the missing shots had not been recorded), we introduces aliasing artifacts in addition to losing the dipping events. The most notable aliasing artifacts are in Figure 8.25d, at $z \approx 600$ meters and $x_m \approx 4,500$ meters.

As the migration examples shown in Figure 8.25 demonstrate, the "conservative" approach described above yields unsatisfactory results as soon as the undersampling becomes larger than a factor of two. An alternative strategy is to ignore the potential of creating aliasing artifacts, and migrate the decimated shots as if the shot-sampling interval were the same as the receiver-sampling interval. For source-receiver migration, this strategy is implemented by simply interleaving the decimated shots with zero traces along the shot axis. For shot-profile migration, this strategy is implemented by propagating the source wavefield on a grid with the same sampling interval as the receivers. Figure 8.26 shows the results of this strategy and of the "anti-aliasing" technique described below. In this figure I show the ADCIGs at two midpoint locations ($x_m = 4,425$ meters and $x_m = 7,425$ meters), in addition to the section obtained by stacking the ADCIGs over angles.

Figure 8.26a shows the results obtained with all the shots ($\Delta_s = 25$ meters), Figure 8.26b shows the results obtained using one shot in every four ($\Delta_s = 100$ meters), and Figure 8.26d shows the results obtained using one shot in every eight ($\Delta_s = 200$ meters). As expected, both the stacked section and the ADCIGs are well imaged in panel a). The stacked section in panel b) is almost as good as the one in panel a), and significantly better than the one obtained using the conservative approach with the same subset of shots (Figure 8.25d). The

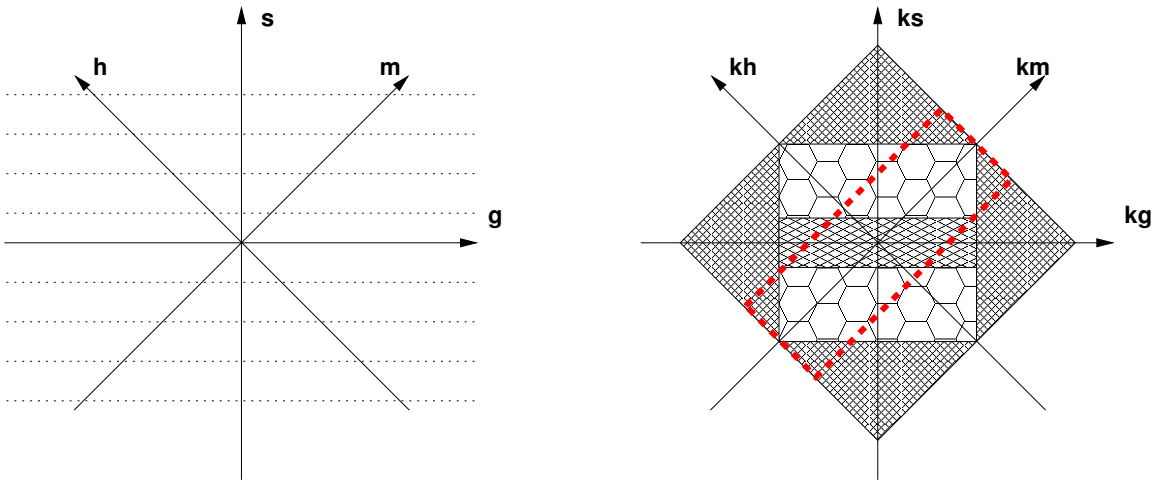


Figure 8.24: Graphical representation of a data set with $\Delta_s = 4\Delta_g$ in the space domain (left panel), and in the wavenumber domain (right panel). In the right panel, the middle rectangle represents the main band of the data set in the source-receiver wavenumber domain with $\Delta_s = 4\Delta_g$. The areas filled with an hexagonal pattern are the side bands along the source wavenumber axis. The dashed box indicates the area contributing to the image when the anti-aliasing procedure described in the text is applied. alias-sr-under-v2 [NR]

comparison between Figure 8.26b and Figure 8.25c demonstrates that when the subsampling is substantial, the side bands contain crucial energy and the elimination of their contributions to the image is damaging. Although the stacked section is almost free of aliasing artifacts, the ADCIG has strong coherent artifacts dipping along the aperture angle axis. These artifacts would prevent any meaningful AVA analysis, and would severely hamper velocity updating by using the kinematic information contained in the ADCIGs (Section 11.2).

As the decimation factor is increased to eight, aliasing artifacts begin to appear also in the stacked section (Figure 8.26d), and the ADCIGs become too affected by aliasing artifacts to be of any practical use. Many of the coherent events related to aliasing, which are visible in the stacked section shown in Figure 8.26d, are the same events visible in the stacked section shown in Figure 8.25d, which was obtained using the "conservative" approach.

To preserve as much of the useful energy as possible and attenuate the strong aliasing artifacts, we can window the image along both the offset and the offset-wavenumber axes. This anti-aliasing strategy has the advantage of being a post-processing task, which is performed on the prestack image cube after migration. Therefore the anti-aliasing parameters (i.e. the bounds of the pass-region in subsurface-offsets and offset-wavenumbers domains) do not need to be determined *a priori*, but can be easily determined by trial and error.

When we window the prestack image cube in the offset domain around zero offset before transformation to the angle domain, we remove a large amount of the aliased energy that creates the coherent dipping events in the ADCIGs. The implicit assumption behind windowing in the offset domain is that the energy corresponding to primary reflections has been focused around zero offset, or equivalently, that the migration velocity is approximately correct. By

excluding the high offset wavenumbers, we effectively band-pass the image at the wide aperture angles, and thus remove some of the high-frequency aliasing artifacts. This windowing is equivalent to selecting the energy contained within the dashed box in Figure 8.24. For example, the strong aliasing artifact that appears around $x_m = 4,425$ meters in the stacked section obtained by taking one shot in every eight (panel d), corresponds to the artifact that appears at wide aperture angles (well beyond 40 degrees) in the ADCIG. As with windowing in offset, also windowing in offset wavenumber may create problems when we want to use the ADCIGs for velocity estimation. The resolution of the measurements of the velocity errors from migrated ADCIGs is dependent on the availability of information at wide-aperture angles.

Figures 8.26c and 8.26e show the result of applying this anti-aliasing methodology to the prestack images shown in Figures 8.26b and 8.26d, respectively. In both cases the prestack image was windowed around zero offset between -200 meters and 200 meters. To create the ADCIGs shown in Figure 8.26c, only half of the available offset-wavenumber bandwidth was used, and only a quarter of the bandwidth was used in Figure 8.26e. The quality of the ADCIGs has significantly improved because of the application of anti-aliasing, and the aliasing artifact present at $x_m = 4,425$ meters in the stacked section in Figure 8.26d has been strongly attenuated in Figure 8.26e. On the other hand, the stacked image of the flat shallow events around the ADCIG located at $x_m = 7,425$ meters has lost coherency in Figure 8.26e, compared with the same events in Figure 8.26d. This loss of coherency can be explained by analyzing the corresponding ADCIG in the fully sampled image (Figure 8.26a), and noticing that a large part of the energy contributing to the stacked image of the shallow events is reflected at fairly wide angles, whereas the narrow angles have little energy, because the minimum acquisition offset in the data is 200 meters.

Aliasing and velocity estimation

Velocity estimation is one of the main uses of prestack images like the ADCIGs shown in Figure 8.26. It is therefore important to analyze the effects of aliasing on the prestack images when the migration velocity is not correct, and to verify under which conditions we can accurately extract velocity information from the migrated ADCIGs. Figure 8.27 shows the images obtained when the migration slowness is 5% higher than the correct one. Each of the panels in Figure 8.27 is obtained by applying the same procedure followed to compute the respective panels in Figure 8.26, except that in Figure 8.27 the migration slowness was uniformly scaled by a factor of 1.05.

As expected, the ADCIGs obtained with full shot sampling (Figure 8.27a) smile upward and provide good velocity information. Unfortunately, the images obtained when the velocity is too slow appear to be more susceptible to aliasing artifacts than the images obtained when the velocity is correct. The undersampling of the shots by a factor of four causes severe artifacts in the stacked section shown in Figure 8.27b, whereas the equivalent section shown in Figure 8.26b is almost artifact-free. In particular, the aliasing artifacts that appear in Figure 8.27b are similar to the artifacts visible in the stacked section obtained with the correct velocity when the undersampling factor is doubled; that is, eight instead of four (Figure 8.27d). Both the stacked section and the ADCIGs obtained by decimating the shots by a factor of four

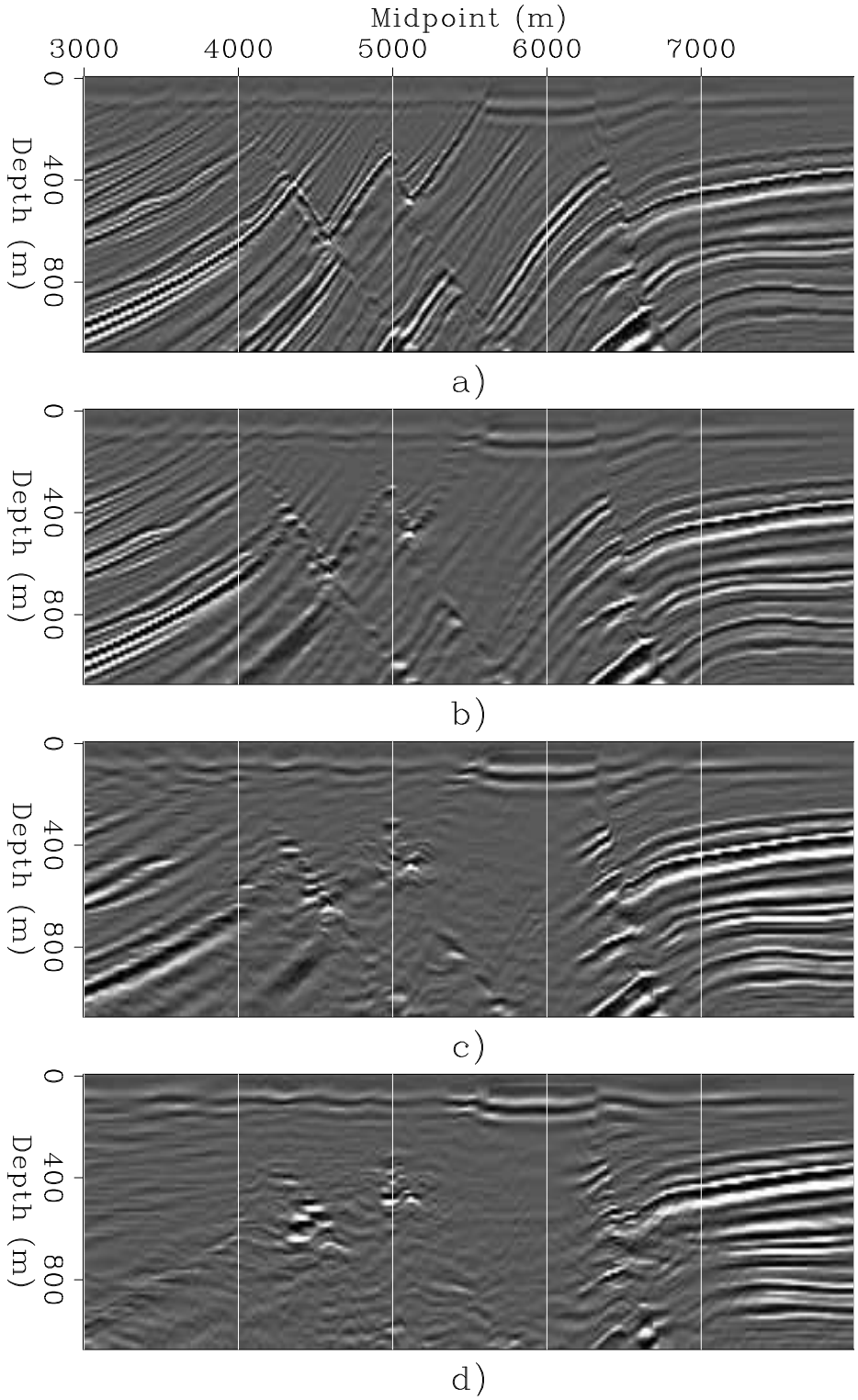


Figure 8.25: The panel on the top displays the migrated section obtained by using all the shots in the data ($\Delta_s = 25$ meters). The other migrated sections are obtained by decimating the shots, followed by interleaving the decimated shots with zero traces along the shot axis, and then band-passing along the source axis before migration. The shot intervals were: $\Delta_s =$ 50 meters for panel b), $\Delta_s = 100$ meters for panel c), and $\Delta_s = 200$ meters for panel d). alias-zoff-all-bpass-overn [ER]

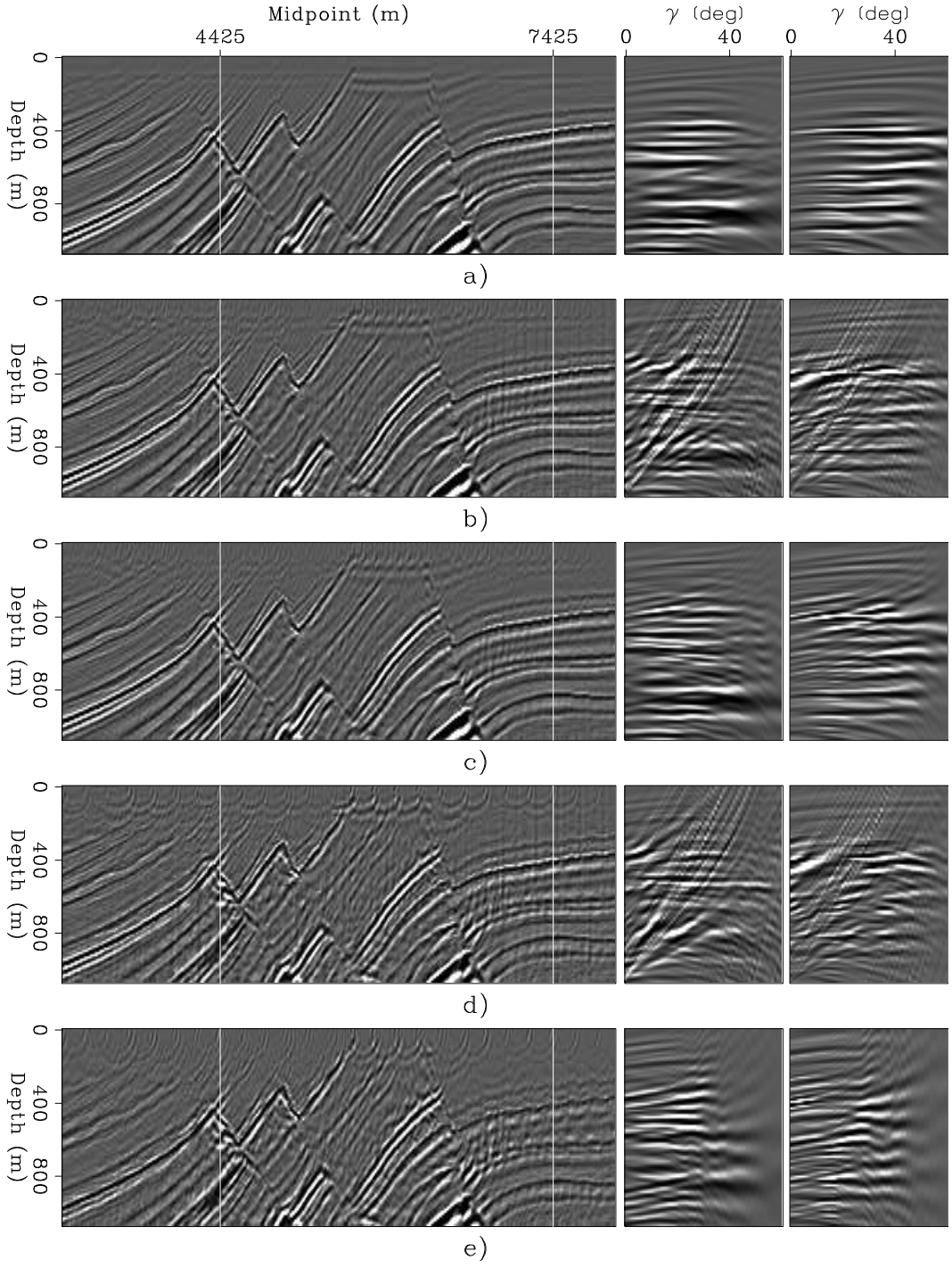


Figure 8.26: Migration results obtained with the correct velocity. The panel on the top displays the migrated section and two ADCIGs obtained by using all the shots in the data ($\Delta_s = 25$ meters). The other migrated results are obtained by first decimating the shots and then interleaving the decimated shots with zero traces along the shot axis. The shot intervals were: $\Delta_s = 100$ meters for panels b) and c), and $\Delta_s = 200$ meters for panels d) and e). alias-zoff-cig-all-overn [ER]

substantially improve after the application of the anti-aliasing procedure described above (Figure 8.27c). The application of the anti-aliasing procedure has fairly well preserved the velocity information in the ADCIGs, with the exception of the aliased event that appears in the stacked section and that is flat along the angle axis in the ADCIG located at $x_m = 7,425$ meters. The flatness of the aliased event along the angle axis erroneously indicates a correct migration velocity for that event, and consequently may mislead a velocity-updating procedure.

When the shots are undersampled by a factor of eight (Figure 8.27d), the stacked image is so severely affected by aliasing artifacts that the interpretation of the reflectors is seriously impaired. The application of the anti-aliasing procedure described above substantially improves the interpretability of the reflectors in the stacked image (Figure 8.27e), and thus could help the velocity estimation process. Unfortunately, even after the application of anti-aliasing, the ADCIGs are so severely affected by aliasing that it is unlikely that reliable velocity information can be extracted from prestack images.

8.3.2 Aliasing in image space

Zhang et al. (2003) describe another type of aliasing artifact that can be inadvertently created when performing wavefield-continuation migrations, by either source-receiver migration or shot-profile migration. These aliasing artifacts are caused by undersampling the image space, and consequently the phenomenon is called **image-space aliasing**. This type of aliasing occurs even when the source and receiver spacings are the same. It is caused by the fact that, as we discussed in the previous section, the final image is built in midpoint-offset coordinates for both source-receiver migration [imaging condition in equation (4.12)] and shot gather migration [imaging condition in equation (4.7)], even if the data are recorded, and often propagated, in source-receiver coordinates. Figure 8.23 shows that if the midpoint-offset axes in the image are sampled with the same sampling rate as the wavefields in source-receiver coordinates, the imaging condition introduces aliasing in the image. Referring to the left sketch in Figure 8.23, image aliasing can be visualized as the folding of the middle square, which is the footprint of the propagating wavefields, into the inner square, which is the footprint of the image.

An alternative way to explain the occurrence of image-space aliasing is to notice that the space-domain multiplication between the source and receiver wavefields, which is required for evaluating the imaging condition, is equivalent to a convolution in the wavenumber domain. This convolution generates a signal with twice the bandwidth of the original one; hence the need for the image to be sample twice as densely as the source and receiver wavefields.

REFERENCES

Abma, R., Sun, J., and Bernitsas, N., 1998, Anti-aliasing methods in Kirchhoff migration: Geophysics, **64**, no. 6, 1783–1792.

Artman, B., Biondi, B., and Shragge, J., 2003, Operator aliasing in wavefield continuation

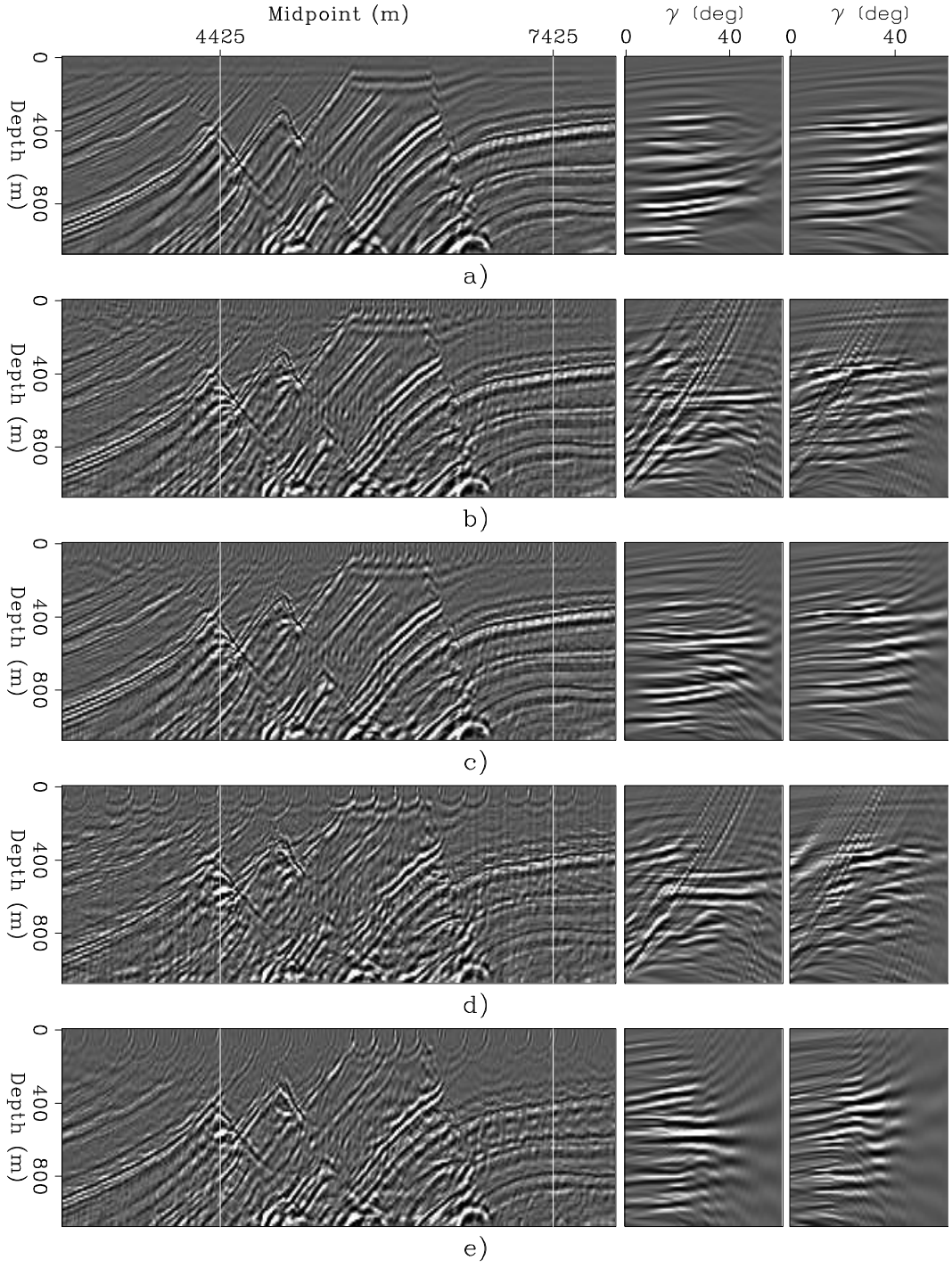


Figure 8.27: Migration results obtained with the slowness 5% higher than the correct one. The panel on the top displays the migrated section and two ADCIGs obtained by using all the shots in the data ($\Delta_s = 25$ meters). The other migrated results are obtained by first decimating the shots and then interleaving the decimated shots with zero traces along the shot axis. The shot intervals were: $\Delta_s = 100$ meters for panels b) and c), and $\Delta_s = 200$ meters for panels d) and e). alias-zoff-cig-all-slow-overn [ER]

migration: 73rd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1091–1093.

Bevc, D., and Claerbout, J., 1992, Fast Anti-Aliased Kirchhoff Migration and Modeling: SEP–**75**, 91–96.

Biondi, B., Fomel, S., and Chemingui, N., 1998, Azimuth moveout for 3-D prestack imaging: Geophysics, **63**, no. 2, 574–588.

Biondi, B., 2001, Kirchhoff imaging beyond aliasing: Geophysics, **66**, 654–666.

Bracewell, R. N., 1999, The Fourier Transform & Its Applications: McGraw-Hill Science/Engineering/Math, 3rd edition.

Claerbout, J. F., 1995, Basic Earth Imaging: Stanford Exploration Project.

Gardner, G. H. F., French, W. S., and Matzuk, T., 1974, Elements of migration and velocity analysis: Geophysics, **39**, 811–825.

Gray, S. H., 1992, Frequency-selective design of the Kirchhoff migration operator: Geophys. Prosp., **40**, no. 5, 565–572.

Hale, D., 1991, A nonaliased integral method for dip moveout: Geophysics, **56**, no. 6, 795–805.

Lumley, D. E., Claerbout, J. F., and Bevc, D., 1994, Anti-aliased Kirchhoff 3-D migration: 64th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1282–1285.

Pica, A., 1996, Model-based anti-aliased Kirchhoff 3D PSDM: 58th Mtg. Eur. Assoc. Expl Geophys., Eur. Assoc. Expl. Geophys.

Schneider, W. A., 1978, Integral formulation for migration in two-dimensions and three-dimensions: Geophysics, **43**, no. 1, 49–76.

Stolt, R. H., 1978, Migration by Fourier transform: Geophysics, **43**, 23–48.

Versteeg, R., and Grau, G., 1990, Practical aspects of seismic data inversion, The Marmousi experience: 52th Mtg. Eur. Assoc. Expl Geophys., 1–194.

Zhang, Y., Sun, J., and Gray, S., 2003, Aliasing in wavefield extrapolation prestack migration: Geophysics, **68**, 629–633.

Chapter 9

Imaging and partial subsurface illumination

In Chapter 8, we analyze the effects of the data spatial sampling rate on the quality of the image. If the data sampling is not sufficiently dense, the seismic image may lose resolution and/or may be affected by artifacts. Unfortunately, density of spatial sampling is not the only problem encountered with realistic 3-D acquisition geometries. An even more common problem is the irregularity of the spatial sampling. Often this irregularity is a product of practical constraints that prevent us from recording data with regular geometries; examples include cable feather in marine acquisition and surface obstacles in land acquisition. In other cases (e.g. button patch geometries), the irregular geometry might be inherent to the survey design.

The main effect of irregular geometries is **uneven illumination** or **incomplete illumination** of the subsurface. This partial illumination causes distortions in the image. In the milder cases, the distortions are limited to the image amplitudes, and they are clearly visible in depth (or time) slices. These distortions are often described as **acquisition footprint**. Figure 9.1 shows an example of acquisition footprints in a migrated depth slice taken from a marine data set. On the right-hand side, horizontal striping is clearly visible superimposed over the image of a complex turbidite system with crossing channels. The horizontal striping is not linked with geology but is along the direction of the sailing lines of the recording vessel.

When the subsurface illumination is incomplete, besides being uneven, the phase of the image is distorted, and strong artifacts are created. At the limit, when the acquisition geometry has holes, the data are aliased, at least locally. In these cases the distinction between the effects of coarse sampling (which we called aliasing in Chapter 8) and those of irregular geometries is obviously artificial. However, it is useful to analyze the effects separately and develop independent methods to alleviate the problems.

Uneven, or incomplete, illumination can be also caused by complexity of the velocity function in the overburden, in addition to irregular acquisition geometries. Imaging under salt edges is an example of an important imaging task that suffers from partial illumination of the reflectors. The problem is often caused by sharp variations in the velocity model that prevent the seismic energy either from reaching the reflectors, or from propagating back to the surface. Although the immediate causes of the partial illumination are different in the

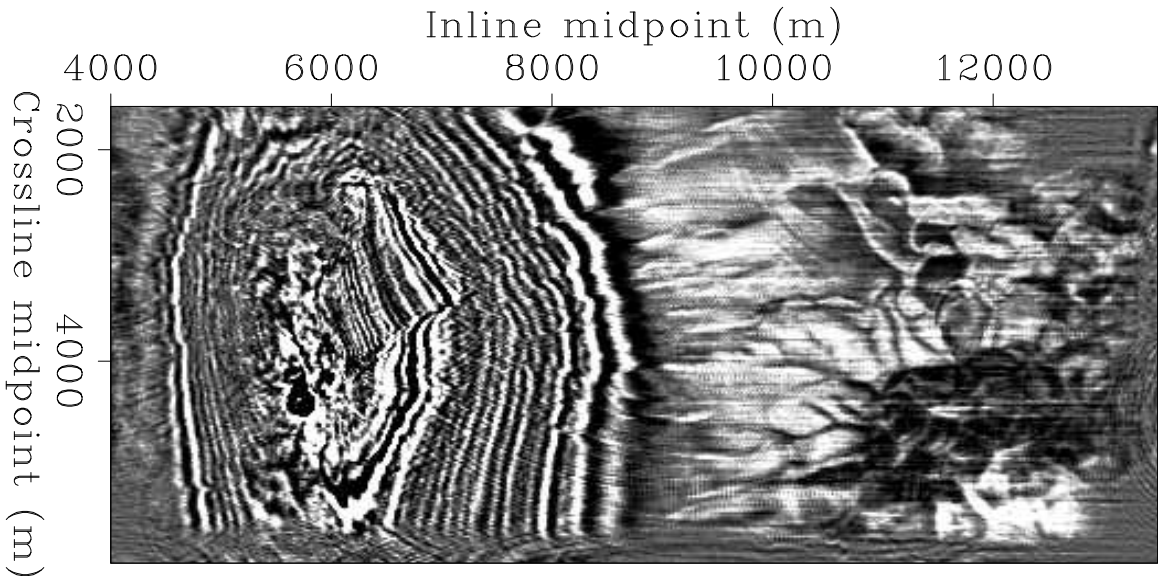


Figure 9.1: Example of acquisition footprint in a migrated depth slice. The horizontal stripes are related to the acquisition sail lines. Notice that the stripes bend when the reflectors start dipping in the vicinity of the salt ($x_m \approx 5,500$ meters), bin-footprint [CR]

two cases – irregular acquisition geometry and complex overburden – the final manifestation is the same: the wavefield is not sufficiently sampled at depth for migration to be able to image the reflectors without artifacts. The concepts and methods used to address the uneven illumination problem are similar, regardless of its origin, and consequently I present them in a unified manner.

When the illumination is uneven, but there are no gaps, the image can be substantially improved by a simple normalization of the imaging operator, or as it is often called, by an **operator equalization**. In this chapter we introduce the basic concepts of operator equalizations, using a simple imaging operator – interpolation followed by partial stacking – as proxy for more complex imaging operators. In the cases when the uneven illumination of the reflectors is mostly related to the irregular acquisition geometry, and the velocity in the overburden is fairly simple, the DMO or AMO operators (Chapter 3) are normalized (Beasley and Mobley, 1988; Canning and Gardner, 1998; Chemingui, 1999). In more complex situations, where the velocity in the overburden is sufficiently complex to distort the wavefield, or even to cause illumination gaps, the normalization should be applied in the image domain after full prestack migration (Bloor et al., 1999; Rickett, 2003).

A simple normalization of the imaging operators is not sufficient to remove the imaging artifacts when the illumination gaps are large. In these conditions, the data-modeling operator – which is usually defined as the adjoint of the imaging operator – should be inverted by a regularized inversion methodology. As in the cases of operator equalization, the methods proposed in the literature for inverting imaging operators can be divided between algorithms based on partial-prestack migration (Ronen, 1987; Ronen and Liner, 2000; Chemingui and Biondi, 2002), and those based on full-prestack migration: using either a Kirchhoff operator

(Nemeth et al., 1999; Duquet et al., 2000) or a wavefield-continuation operator (Prucha and Biondi, 2002; Kuehl and Sacchi, 2002).

Iterative inversion is expensive, in particular when a full prestack migration operator is inverted. In this chapter I present a non-iterative method for regularizing the model space that improves the quality of the reconstructed data without the computational cost of an iterative inversion. However, in the presence of large acquisition gaps, or when the complexity of the overburden is responsible for the incomplete illumination of the reflectors, expensive iterative regularized inversion is unavoidable. At the end of this chapter, we discuss some potential applications of iterative inversion.

9.1 Equalization of imaging operators

To explore the methods used to equalize imaging operators, I use interpolation followed by partial stacking as a proxy of more complex imaging operators. As a proxy, interpolation has the advantage of being simple, easy to understand, and easy to manipulate analytically. Its analysis will lead us to discuss fundamental issues regarding spatial interpolation of seismic traces, and normalization, or equalization, of imaging operators. The lessons learned by using interpolation are applicable to the equalization of several imaging operators.

Stacking is the operation of averaging seismic traces by summation. It is an effective way to reduce the size of data sets and to enhance reflections while attenuating noise. To avoid attenuating the signal together with the noise, the reflections need to be coherent among the traces that are being stacked. To increase trace coherency we can apply simple Normal Moveout (NMO) before stacking, or a partial prestack migration operator such as DMO or AMO (Chapter 3).

Global stacking of all the traces recorded at the same midpoint location, no matter their offset and azimuth, is the most common type of stacking. Partial stacking averages only those traces with offset and azimuth within a given range. It is useful when we want to preserve differences among traces that are functions of the trace offset and azimuth, and thus we must avoid global averaging. AVO studies are a good example of a useful application of partial stacking. Partial stacking is also useful when simple transformations, such as NMO, are not sufficient to correct for the differences in time delays among traces with very different offsets and azimuths. This situation is common when velocity variations cause non-hyperbolic moveouts in the data. Because in partial stacking the data redundancy is low, the results of partial stacking are more likely to be affected by artifacts related to irregular acquisition geometries than are the results of global stacking. In this section I will thus focus my analysis on partial stacking, but the methods presented here obviously can be applied to global stacking operators, too.

To start our analysis, I define a simple linear model that links the recorded traces (at arbitrary midpoint locations) to the stacked volume (defined at on a regular grid). Each data trace is the result of interpolating the stacked traces, and it is equal to the weighted sum of the neighboring stacked traces. The interpolation weights are functions of the distance between

the midpoint location of the model trace and the midpoint location of the data trace. The sum of all the weights corresponding to one data trace is usually equal to one. Because the weights are independent of time along the seismic traces, for the sake of notation simplicity, we collapse the time axis and consider each element d_i of the data space (recorded data) **d**, and each element m_j of the model space **m** (stacked volume), as representing a whole trace. The relationship between data and model is linear and can be expressed as follows:

$$d_i = \Sigma_j \, l_{ij} \, m_j; \text{ subject to the constraint } \Sigma_j \, l_{ij} = 1. \tag{9.1}$$

In matrix notation, equation (9.1) becomes

$$\mathbf{d} = \mathbf{L}\mathbf{m}. \tag{9.2}$$

The simplest, and crudest, spatial interpolation is a nearest-neighborhood interpolation. For example, if we have three model traces and four data traces, and we use a simple nearest-neighborhood interpolator, equation (9.2) becomes

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix}. \tag{9.3}$$

Linear interpolation is more accurate than nearest-neighborhood interpolation and almost as simple. If we use linear interpolation instead of nearest-neighborhood interpolation, the example above becomes

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix}. \tag{9.4}$$

Stacking is the summing of the data traces into the model traces, weighted by the interpolation weights. In operator notation, stacking can be represented as the application of the **adjoint operator** $\mathbf{L}'$ to the data traces (Claerbout, 2004); that is,

$$\mathbf{m} = \mathbf{L}'\mathbf{d} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{d}. \tag{9.5}$$

Unfortunately, the application of simple stacking as an adjoint operator does not yield satisfactory results when the fold is unevenly distributed among midpoint bins. In the stack, the amplitudes of the bins with higher fold will be artificially higher than the amplitudes of the bins with lower fold. To compensate for this unevenness in the fold, it is common practice to divide the stacked traces by the inverse of the fold. This **fold normalization** also can be expressed in operator notation, when a diagonal operator $\mathbf{W}^m$ is added in front of the adjoint operator in the computation of the stack:

$$\mathbf{m} = \mathbf{W}^m\mathbf{L}'\mathbf{d} = \begin{bmatrix} w_1^m & . & . \\ . & w_2^m & . \\ . & . & w_3^m \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{d}. \tag{9.6}$$

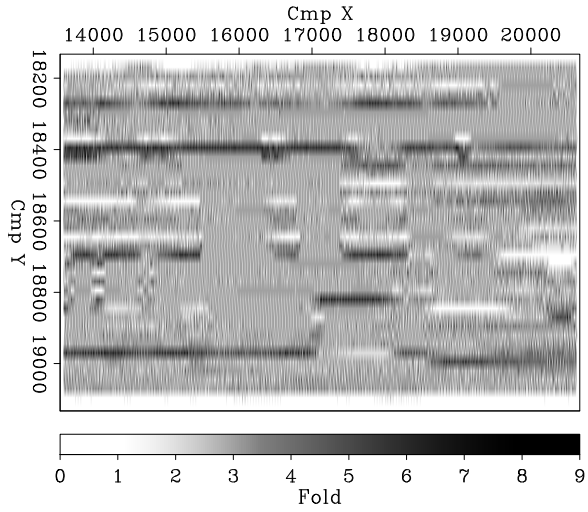


Figure 9.2: Fold of partial stacking by nearest-neighborhood interpolation of a marine data set over offsets ranging from 400 m to 600 m. bin-F-nsea [ER,M]

The weights w_j^m are given by the inverse of the fold, which can be simply computed by a summation of the elements in each column of $\mathbf{L}$; that is,

$$w_j^m = \left(\Sigma_i \, l_{ij}\right)^{-1}. \tag{9.7}$$

Figures 9.2 and 9.3 show the fold as a function of the midpoint axes, computed for the data set shown in the examples presented in Section 9.1.2. This is a marine data set recorded in the North Sea. Details on the data geometry are presented in Chapter 1. The data was partially stacked over offsets ranging from 400 m to 600 m. The fold resulting from linear interpolation (Figure 9.3) is smoother and shows fewer gaps (white spots) than the fold resulting from nearest-neighborhood interpolation (Figure 9.2). Gaps in the fold present a problem for fold normalization, because they make the weights diverge to infinity in equation (9.7). To avoid instability, it is common practice to add a small number ϵ_w to the actual fold, or to set the weights to zero when the fold is smaller than ϵ_w, as in the following revised expression for the weights:

$$w_j^m = \begin{cases} \left(\Sigma_i \, l_{ij}\right)^{-1} & \text{if } \Sigma_i \, l_{ij} \geq \epsilon_w, \\ \\ 0 & \text{elsewhere.} \end{cases} \tag{9.8}$$

We derived the fold normalization by a simple heuristic, and it may seem an *ad hoc* solution to the problem of normalizing the output of stacking. However, it can be shown that the weights used by fold normalization can be derived from applying the general theory of inverse least-squares to the stacking normalization problem. It turns out that fold normalization is equal to the exact least-squares inverse when nearest-neighborhood interpolation is used, while it is equal to an approximate inverse when more accurate interpolators are used.

In the next section I show the relationship between fold normalization and inverse theory, and derive simple principles that will be useful for the analysis of operators that are more complex than stacking.

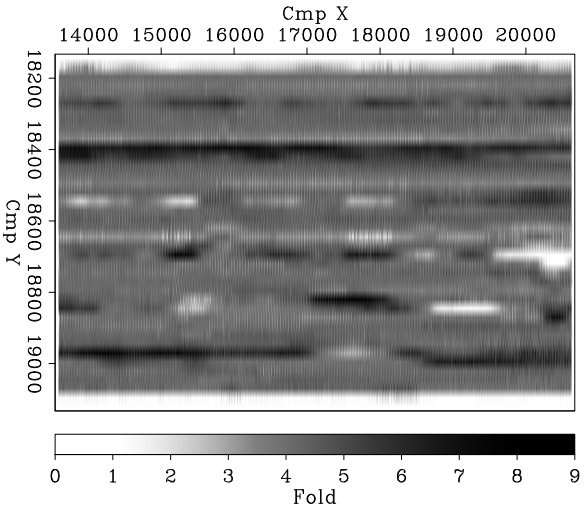


Figure 9.3: Fold of partial stacking by linear interpolation of a marine data set over offsets ranging from 400 m to 600 m. bin-F-nsea-lin [ER,M]

9.1.1 Least-squares inverses

The simple solution to the normalization of stacking that we derived in the previous section can be seen as a crude approximation to the solution of a least-squares problem, where the interpolation operator $\mathbf{L}$ is the forward modeling operator. This least-squares problem can be expressed as the minimization of the following objective function:

$$J(\mathbf{m}) = \|\mathbf{L}\mathbf{m} - \mathbf{d}\|_2 . \tag{9.9}$$

In general, the operator $\mathbf{L}$ is not square and its inverse is not defined; therefore, we use its least-squares inverse (Strang, 1986). The least-squares inverse can be expressed in terms of the operator $\mathbf{L}$ as follows:

$$\mathbf{L}^{\ddagger}_{m} = \left(\mathbf{L}'\mathbf{L}\right)^{-1}\mathbf{L}'. \tag{9.10}$$

Applying the least-squares inverse is equivalent to applying the adjoint operator $\mathbf{L}'$, followed by a *spatial filtering* of the model space given by the inverse of $\mathbf{L}'\mathbf{L}$. The fold normalization can be seen as a particular approximation of the inverse of $\mathbf{L}'\mathbf{L}$ with a diagonal operator. When nearest-neighborhood interpolation is used, $\mathbf{L}'\mathbf{L}$ is indeed diagonal, and each element is equal to the fold, as it can be easily verified in the particular case of our example:

$$\mathbf{L}'\mathbf{L} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{9.11}$$

But it can be also easily demonstrated in the general case. The elements of $\mathbf{L}'\mathbf{L}$ are given by

$$M_{ij} = \Sigma_k \, l_{ki} \, l_{kj}. \tag{9.12}$$

If nearest-neighborhood interpolation is used, l_{ki} is different from zero, and equal to one, for only one value of i for each k. Therefore,

$$M_{ii} = \begin{cases} \Sigma_k \, l_{ki}^2 & = \quad \Sigma_k \, l_{ki} \\ 0 \end{cases} . \tag{9.13}$$

Only the diagonal elements M_{ii} are different from zero, and equal to the sum of the columns of $\mathbf{L}$, that is, the fold in each model bin. When more accurate interpolators than nearest-neighborhood are used, $\mathbf{L}'\mathbf{L}$ is not diagonal and cannot be easily inverted. In the next section we will see that fold normalization is equivalent to a particular choice of an approximate inverse of $\mathbf{L}'\mathbf{L}$.

The least-squares inverse can be alternatively expressed as

$$\mathbf{L}^{\ddagger}_{d} = \mathbf{L}' \left(\mathbf{L}\mathbf{L}' \right)^{-1} . \tag{9.14}$$

In this case, the adjoint operator is applied after the data have been *filtered* with the inverse of $\mathbf{L}\mathbf{L}'$.

When the operator $\mathbf{L}$ is not square, either $\mathbf{L}'\mathbf{L}$ or $\mathbf{L}\mathbf{L}'$ is singular. When the number of data traces n_d is larger than the number of the model traces n_m, as in our simple example, $\mathbf{L}\mathbf{L}'$ is singular, as it can be easily verified in the example,

$$\mathbf{L}\mathbf{L}' = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} . \tag{9.15}$$

The right-hand matrix is singular because the first and last rows are identical. In this case the problem is called **overdetermined**. Because $\mathbf{L}\mathbf{L}'$ is singular, the least-squares inverse expressed in equation (9.10) is commonly used for overdetermined problems. In these cases the equations may be inconsistent, and no model vector exists that exactly satisfies equation (9.2).

On the contrary, if $n_d < n_m$ the problem is **underdetermined**, and there is no unique solution of equation (9.10). In this case, the operator $\mathbf{L}'\mathbf{L}$ is singular, and the least-squares inverse expressed in equation (9.14) is used. This least-squares inverse is the one that among all the infinite possible solutions has the minimum length; that is, it minimizes the norm $\|\mathbf{m}\|_2$. The two least-squares inverses showed in equations (9.10) and (9.14) become analytically equivalent if the pseudo-inverses of $\mathbf{L}'\mathbf{L}$ and $\mathbf{L}\mathbf{L}'$ are substituted for their regular inverses (Strang, 1986).

Because the average nominal fold is higher than one, the stacking problem is usually overdetermined when the spatial sampling of the stack is equal to the nominal bin sizes of the acquisition. But it is also often locally underdetermined where irregularities in acquisition geometry create gaps in the spatial coverage of the midpoint plane. Further, in many interesting problems, the size of the model space is not fixed, but is arbitrarily determined according to the desired resolution of the resulting image. At the limit, the model space can be a continuum with infinite dimensionality. The imaging problem is thus never genuinely overdetermined. For this reason, I will not refer to the two least-squares inverses, as is done often

in the literature, as underdetermined and overdetermined. Rather, I will distinguish them according to the space in which the filtering is applied. Therefore, we will refer to the inverse of equation (9.10) as the **model-space inverse** and we will refer to the inverse of equation (9.14) as the **data-space inverse**. In most of the practical cases, both $\mathbf{L}'\mathbf{L}$ and $\mathbf{L}\mathbf{L}'$ are singular, and care must be taken during the approximation of either inverse.

9.1.2 Diagonal approximations of least-squares inverses

The derivation of the analytical expression of the least-squares model-space inverse is straightforward when we consider simple stacking by nearest-neighborhood interpolation. However, when more accurate interpolators are used for stacking, $\mathbf{L}'\mathbf{L}$ is not diagonal, and computing its inverse is not straightforward. We are also interested in imaging operators that are more complex and expensive than stacking, for which even computing and storing the elements of $\mathbf{L}'\mathbf{L}$ can be too expensive, and directly inverting the operators is impossible. In these cases we have two choices: 1) to compute an analytical approximation to the inverse, 2) to use an iterative method, such as conjugate gradient, to compute a numerical approximation to the inverse (Claerbout, 2004). Even if we follow the second strategy, the availability of an analytical approximation to the inverse is useful, because the approximate inverse can be used as a preconditioner to accelerate the convergence of the iterative inversion by conjugate gradient (Golub and Loan, 1983).

We will discuss two methods for approximating the inverse of $\mathbf{L}'\mathbf{L}$. The first method is based on the direct manipulation of the elements of $\mathbf{L}'\mathbf{L}$, such as extracting its diagonal or summing the elements in its columns (or rows). The second method is based on the idea that, for capturing the most significants properties of $\mathbf{L}'\mathbf{L}$, we can *sound* it by measuring its effects when applied to a reference model ($\mathbf{m}_{\text{ref}}$) (Claerbout and Nichols, 1994; Rickett, 2003). The approximation is then evaluated as

$$\mathbf{L}'\mathbf{L} \approx \frac{\mathbf{diag}\left(\mathbf{L}'\mathbf{L}\,\mathbf{m}_{\text{ref}}\right)}{\mathbf{diag}\left(\mathbf{m}_{\text{ref}}\right)}. \tag{9.16}$$

This method has the important advantage that it requires not the explicit evaluation (and storage) of the elements of $\mathbf{L}'\mathbf{L}$, but just its application to the reference model. Of course, the resulting approximation is strongly dependent on the choice of the reference model. The closer the reference model is to the true model, the better the approximation is. In theory, if the reference model is equal to the true model we will achieve perfect results.

Although these two methods seems unrelated, they yield equivalent results for specific choices of $\mathbf{m}_{\text{ref}}$. Therefore, the second method can be used to analyze the assumptions that underly the possible choices of approximations.

Approximation of $\mathbf{L}'\mathbf{L}$ by the sum of its columns

We now analyze the properties of the approximate inverse that is defined by the substitution of $\mathbf{L}'\mathbf{L}$ with a diagonal matrix $\widetilde{\mathbf{L}'\mathbf{L}}$, where each element of $\widetilde{\mathbf{L}'\mathbf{L}}$ is equal to the sum of the

corresponding row of $\mathbf{L}'\mathbf{L}$. Notice that $\mathbf{L}'\mathbf{L}$ is symmetric, and thus summing over the columns would be equivalent to summing over the rows. This particular approximation for $\mathbf{L}'\mathbf{L}$ is often applied in the numerical solution of finite-element problems; in that literature it is known as mass-lumping approximation (Hughes, 1987).

For the linear-interpolation example introduced in equation (9.4), $\mathbf{L}'\mathbf{L}$ becomes the following banded matrix:

$$\mathbf{L}'\mathbf{L} = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{70}{36} & \frac{17}{36} \\ 0 & \frac{17}{36} & \frac{13}{36} \end{bmatrix}. \tag{9.17}$$

The banded matrix $\mathbf{L}'\mathbf{L}$ can be approximated with the diagonal operator $\widetilde{\mathbf{L}'\mathbf{L}}$, as follows:

$$\mathbf{L}'\mathbf{L} \approx \widetilde{\mathbf{L}'\mathbf{L}} = \begin{bmatrix} \frac{1}{2} & \Leftarrow \Sigma & \Leftarrow \Sigma \\ \Sigma \Rightarrow & \frac{8}{3} & \Leftarrow \Sigma \\ \Sigma \Rightarrow & \Sigma \Rightarrow & \frac{5}{6} \end{bmatrix}. \tag{9.18}$$

It can be noticed that the diagonal elements of $\widetilde{\mathbf{L}'\mathbf{L}}$ are equal to the sum of the columns of the original operator $\mathbf{L}$, as defined in equation (9.4); i.e., they are equal to the fold computed for each corresponding model trace. This is not an accident of the particular example, but it is true in general, as it can be shown analytically by expressing the inverse of $\widetilde{\mathbf{L}'\mathbf{L}}$ as

$$^{\Sigma}w_i^m = \left(\Sigma_j\ M_{ij}\right)^{-1} = \left(\Sigma_j\ \Sigma_k\ l_{ki}\ l_{kj}\right)^{-1} = \left(\Sigma_k\ \ l_{ki}\,\Sigma_j\ l_{kj}\right)^{-1}. \tag{9.19}$$

The weights l_{kj} are interpolator weights, and they fulfill the constraint $\Sigma_j\ l_{kj} = 1$, expressed in equation (9.1). It immediately follows that

$$^{\Sigma}w_i^m = (\Sigma_k\ \ l_{ki})^{-1}, \tag{9.20}$$

which is equivalent to equation (9.7). In summary, we just derived, for the model-space inverse, an approximation that is easy to compute, and it is equal to the fold normalization that was defined by heuristic considerations. However, the definition of the approximate inverse can be applied when more complex imaging operators are involved, and for which we do not have a simple heuristic to define the weights that improve the results obtained by application of the adjoint operator.

It is also easy to show that this approximation is equivalent to the choice of a constant vector as $\mathbf{m}_{\text{ref}}$ in equation (9.16). Therefore, it biases the imaging process toward models that are constant. In the case of stacking, it privileges flat reflectors, which is consistent with the flat-reflector assumptions underlying the stacking process. In the case of imaging operators aimed at imaging complex structure, this bias toward flat reflectors may be less appropriate (Rickett, 2003).

Approximation of $\mathbf{L}'\mathbf{L}$ by its diagonal

The matrix $\mathbf{L}'\mathbf{L}$ is often diagonally dominant. It is thus natural to approximate it with its diagonal (Nemeth et al., 1999; Duquet et al., 2000). For the stacking operator, the evaluation

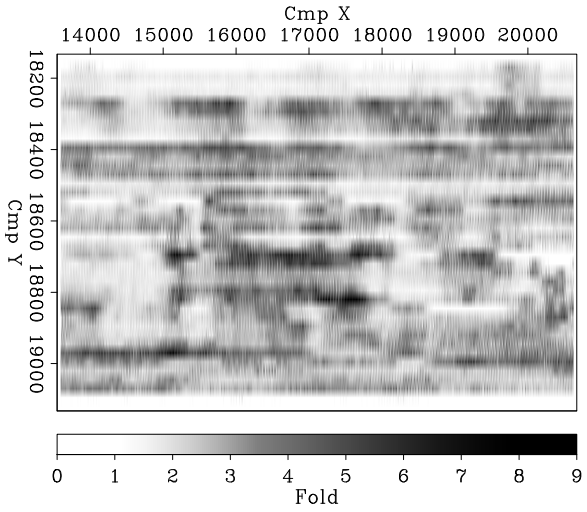


Figure 9.4: Values of the diagonal elements of $\mathbf{L}'\mathbf{L}$ when performing partial stacking by linear interpolation of a marine data set over offsets ranging from 400 m to 600 m. Compare with Figure 9.3. bin-J-nsea-lin [ER,M]

of the diagonal elements is as simple as the evaluation of the sum of the columns (or rows) $\mathbf{L}'\mathbf{L}$. Instead of being the sum of the elements in each column of $\mathbf{L}$ [equation (9.20)], it is the sum of the square of the elements in each column of $\mathbf{L}$; that is,

$$ {}^{d}w_i^m = \left(\Sigma_k \; l_{ki}^2\right)^{-1}. \tag{9.21}$$

In case of nearest-neighborhood interpolation, $\mathbf{L}'\mathbf{L}$ is diagonal (and the elements l_{ki} are equal to either 0 or 1), and thus equation (9.20) and equation (9.21) are equivalent. In the case of linear interpolation, ${}^{\Sigma}w_i^m \leq {}^{d}w_i^m$ for every i, because $0 \leq l_{ki} \leq 1$. In this case, the values yielded by equation (9.20) and equation (9.21) can be significantly different. Figure 9.4 shows the values of the diagonal of $\mathbf{L}'\mathbf{L}$ for the North Sea data example. This figure should be compared with the fold values shown in Figure 9.3. It is apparent that the field shown in Figure 9.4 is rougher and closer to zero in most locations than the field shown in Figure 9.3.

No single reference model could be inserted in equation (9.16) to obtain the approximation equivalent to equation (9.21). However, the use of the diagonal of $\mathbf{L}'\mathbf{L}$ is equivalent to evaluating equation (9.16) with n_m reference models, each with a delta function at the corresponding model location. Therefore, the choice of approximating $\mathbf{L}'\mathbf{L}$ by its diagonal biases the imaging process toward models that are made of delta functions far apart from each other with respect to the wavelength of the seismic signal. This is an assumption that is seldom realistic.

The theoretical analysis is confirmed by the results of applying the two kinds of approximate model-space inverses to the partial stacking of the North Sea data set. Figure 9.5 shows in-line sections obtained by using linear interpolation to partial stacking the data. The section on the top of the figure was obtained without any normalization weights; for the one in the middle, ${}^{d}w_i^m$ were used, and for the one at the bottom, ${}^{\Sigma}w_i^m$ were used. For both approximate inverses, the inversion was stabilized by setting $\epsilon_w = 0.1$ [equation (9.8)]. The gap in the coverage visible at the right end of the section is well compensated by the application of the ${}^{\Sigma}w_i^m$ weights, though at the expense of a lower signal-to-noise ratio. However, the gap is very poorly compensated by the use of the ${}^{d}w_i^m$ weights, which introduce rapid fluctuations in the data amplitudes. Figure 9.6 shows the corresponding time slices. Again, the bottom image $\left({}^{\Sigma}w_i^m\right)$ is smoother and more continuous than the middle one $\left({}^{d}w_i^m\right)$.

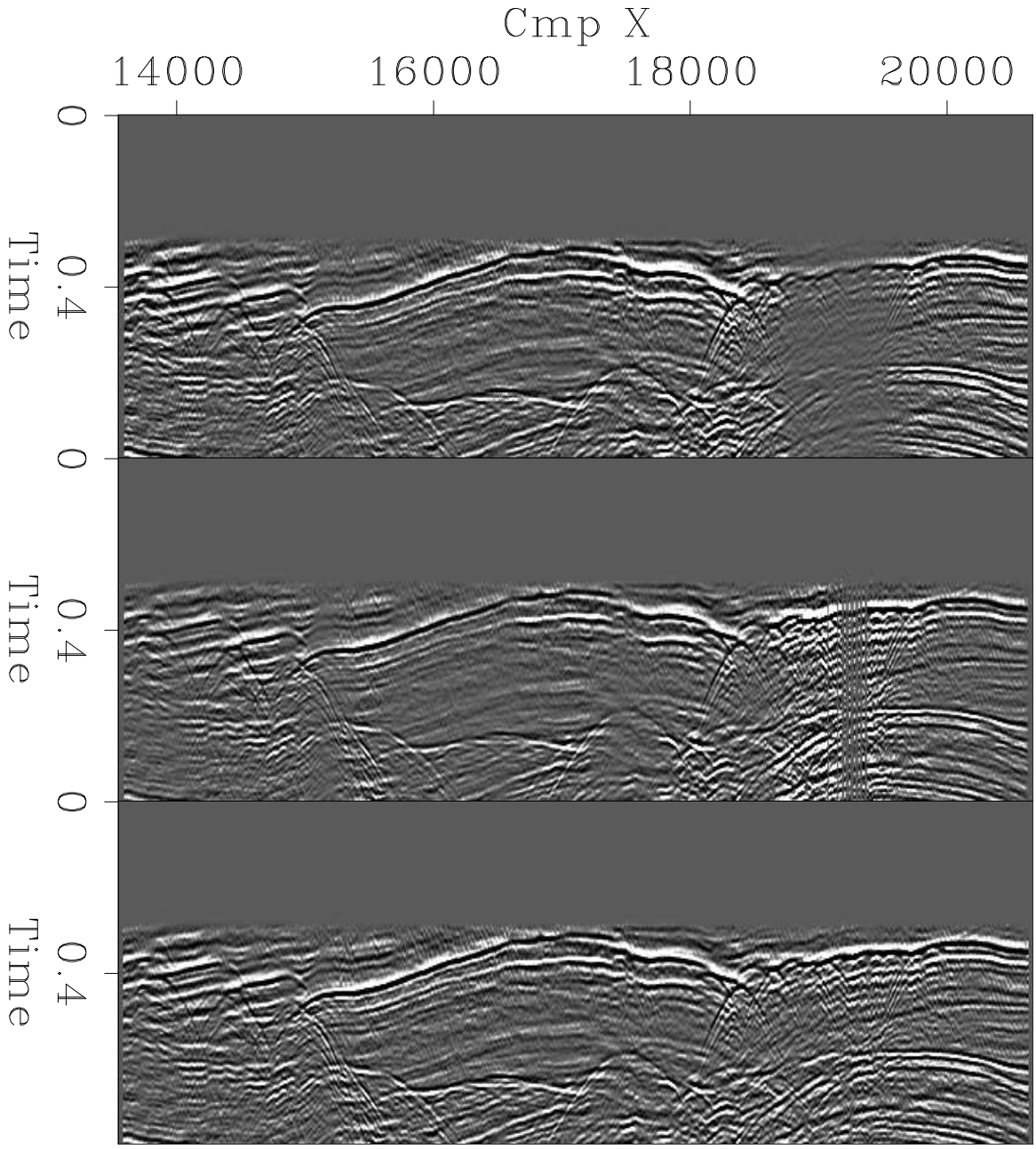


Figure 9.5: In-line sections obtained by partial stacking of the North Sea data set. Linear interpolation was used with the application of no weights (top), ${}^{d}w_i^m$ weights (middle), and ${}^{\Sigma}w_i^m$ weights (bottom). bin-CYj-nsea-lin [ER,M]

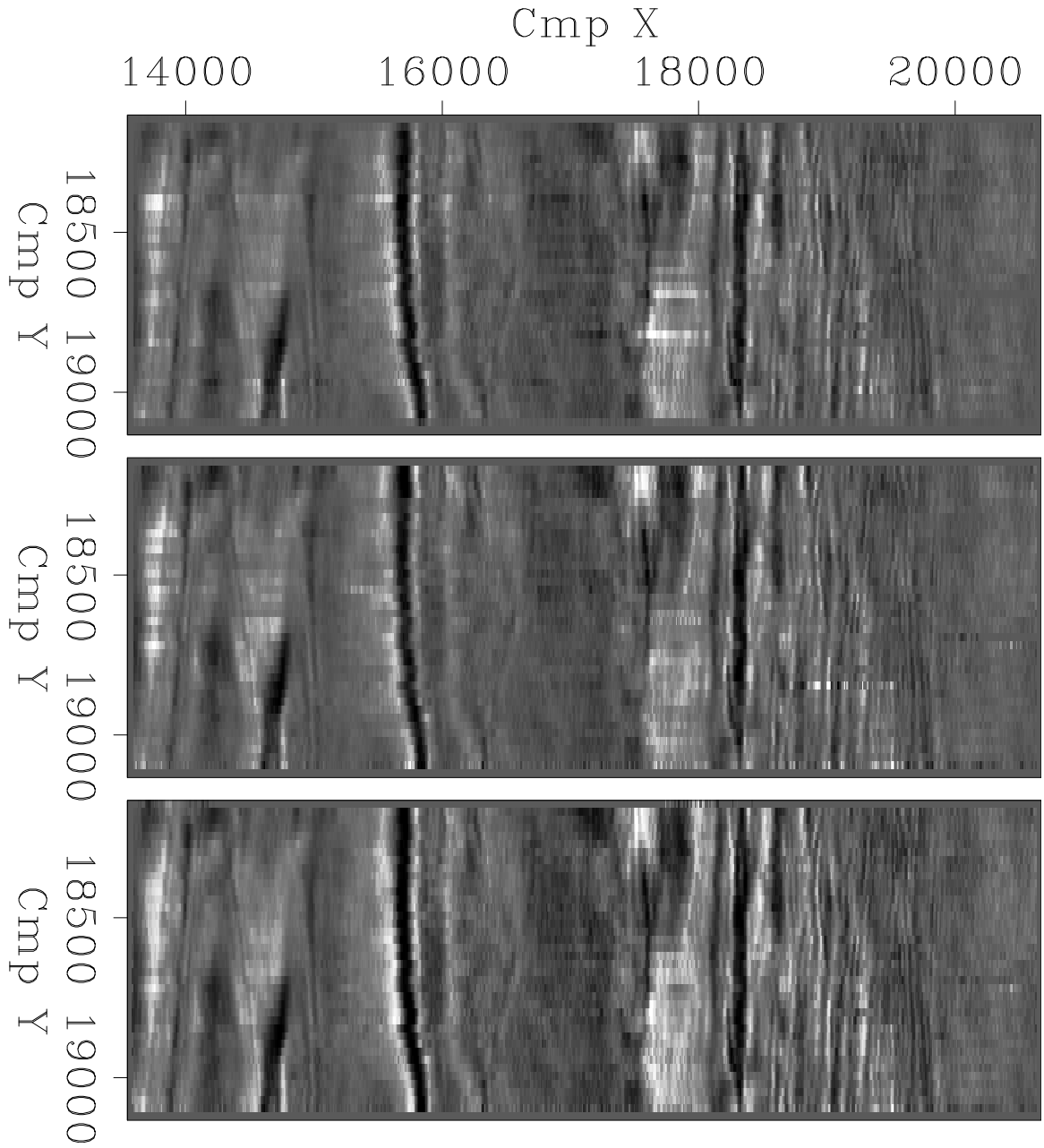


Figure 9.6: Time slices obtained by partial stacking of the North Sea data set. Linear interpolation was used with the application of no weights (top), ${}^{d}w_i^m$ weights (middle), and ${}^{\Sigma}w_i^m$ weights (bottom). bin-CTj-nsea-lin [ER,M]

Approximation of $\mathbf{LL}'$ by the sum of its columns

Next we will apply the same principles used to derive the approximate model-space inverse to derive an approximate data-space inverse.

The data-space inverse can be approximated by replacing $\mathbf{LL}'$ with a diagonal matrix $\widetilde{\mathbf{LL}}'$; the diagonal elements of $\widetilde{\mathbf{LL}}'$ are equal to the sum of the corresponding rows of $\mathbf{LL}'$. The analytical expression for the elements of the inverse of $\widetilde{\mathbf{LL}}'$ is

$$w_i^d = \left(\Sigma_j\ D_{ij}\right)^{-1} = \left(\Sigma_j\ \Sigma_k\ l_{ik}\ l_{jk}\right)^{-1} = \left(\Sigma_k\ \ l_{ik}\Sigma_j\ l_{jk}\right)^{-1} = \left(\Sigma_k\ \ l_{ik}(w_k^m)^{-1}\right)^{-1}. \tag{9.22}$$

If $0 \le l_{ik} \le 1$ and $\Sigma_k\ l_{ik} = 1$, then $\Sigma_j\ l_{jk} \ge l_{ik}$ and $(w_k^d)^{-1} \ge \Sigma_k\ l_{kk}^2 > 0$, and thus the elements w_i^d never diverge to infinity. In contrast, it may happen that the fold normalization weights, as defined by equation (9.7), do diverge. This stability property is an attractive characteristic of the approximate data-space inverse. Also, notice that the computation of the elements w_i^d can be recast as a weighted sum of the elements $(w_k^m)^{-1}$; this observation suggests a straightforward algorithm for the implementation of the data-space weighting.

The corresponding matrices for the linear-interpolation example are

$$\mathbf{LL}' = \begin{bmatrix} 1 & \frac{1}{2} & \frac{2}{3} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{2}{3} & \frac{1}{3} & \frac{5}{9} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{9.23}$$

and

$$\widetilde{\mathbf{LL}}' = \begin{bmatrix} \frac{8}{3} & \Leftarrow\Sigma & \Leftarrow\Sigma & \Leftarrow\Sigma \\ \Sigma\Rightarrow & \frac{19}{12} & \Leftarrow\Sigma & \Leftarrow\Sigma \\ \Sigma\Rightarrow & \Sigma\Rightarrow & \frac{37}{18} & \Leftarrow\Sigma \\ \Sigma\Rightarrow & \Sigma\Rightarrow & \Sigma\Rightarrow & \frac{7}{4} \end{bmatrix}. \tag{9.24}$$

For the nearest-neighborhood interpolation example,

$$\widetilde{\mathbf{LL}}' = \begin{bmatrix} 2 & \Leftarrow\Sigma & \Leftarrow\Sigma & \Leftarrow\Sigma \\ \Sigma\Rightarrow & 1 & \Leftarrow\Sigma & \Leftarrow\Sigma \\ \Sigma\Rightarrow & \Sigma\Rightarrow & 1 & \Leftarrow\Sigma \\ \Sigma\Rightarrow & \Sigma\Rightarrow & \Sigma\Rightarrow & 2 \end{bmatrix}. \tag{9.25}$$

It can be shown analytically that in the case of nearest-neighborhood interpolation, the following identity always holds:

$$\mathbf{L}'\left(\widetilde{\mathbf{LL}}'\right)^{-1} = \left(\widetilde{\mathbf{L}'\mathbf{L}}\right)^{-1}\mathbf{L}', \tag{9.26}$$

though in general,

$$\widetilde{\mathbf{L}'\mathbf{L}} \ne \widetilde{\mathbf{LL}}'. \tag{9.27}$$

Equations (9.26) and (9.27) are consistent, though apparently contradictory, because the difference between $\widetilde{\mathbf{L}'\mathbf{L}}$ and $\mathbf{LL}'$ belongs to the null space of $\mathbf{L}'$.

The equivalent of equation (9.16) for the data-space inverse is

$$\mathbf{LL}' \approx \frac{\mathbf{diag}\left(\mathbf{LL}'\ \mathbf{d}_{\text{ref}}\right)}{\mathbf{diag}\left(\mathbf{d}_{\text{ref}}\right)}, \tag{9.28}$$

where $\mathbf{d}_{\text{ref}}$ is a reference data set. As for the model-space inverse, the approximation of $\mathbf{L}\mathbf{L}'$ by the sum of its columns (or rows) is equivalent to choosing a reference data set with constant values.

Similarly, the approximation of $\mathbf{L}\mathbf{L}'$ by its diagonal would be equivalent to choosing a reference data set with isolated spikes. This choice would be even more unrealistic in this case, because the mechanism of wave propagation enforces some degree of spatial correlation between data points.

Model-space inverse vs. data-space inverse

We derived a convenient approximation to both the data-space and the model-space inverses. It is natural to ask when we should use one and when we should use the other. One important factor to consider is the nature of $\mathbf{L}'\mathbf{L}$ and $\mathbf{L}\mathbf{L}'$, and in particular how difficult is to compute their (approximate) inverses. To suppress noise and estimate velocities, we tend to collect data that is highly redundant, therefore the problems can be often considered overdetermined. The model-space inverse is usually advised for use in overdetermined problems. However, the data-space inverse has an inherent advantage over the model-space inverse when we take into account the irregular sampling of 3-D data. To better understand this concept, we digress to analyze the role of $\mathbf{L}'\mathbf{L}$ and $\mathbf{L}\mathbf{L}'$ in the least-squares inverse process.

As mentioned before, the inverse of $\mathbf{L}'\mathbf{L}$ ($\mathbf{L}\mathbf{L}'$) filters the model (data) space to *correct* the adjoint operator for the interdependencies between model (data) elements. Each element M_{ij} (D_{ij}) of $\mathbf{L}'\mathbf{L}$ ($\mathbf{L}\mathbf{L}'$) describes the correlation between the model parameter m_i (data parameter d_i) and the model parameter m_j (data parameter d_j). This correlation is equal to zero if the parameters are totally uncorrelated, and equal to one if they are fully correlated. For example, the operator $\mathbf{L}'\mathbf{L}$ is diagonal when all the model parameters are independent, as in the case of nearest-neighborhood interpolation. When simple stacking is used, the correlation between model parameters (data parameters) is limited to a small number of neighboring elements and is thus fairly cheap to evaluate numerically. But for more complex imaging operators that have wider spatial aperture, the correlation is more complex, and $\mathbf{L}'\mathbf{L}$ ($\mathbf{L}\mathbf{L}'$) is a denser matrix. In such cases, the computation of each element M_{ij} requires the evaluation of an inner product in the data space. This inner product must be computed numerically, because of the irregularity of the data geometry. In many cases the numerical computation is too expensive. In contrast, the computation of each element D_{ij} requires the evaluation of an inner product in the model space. Because the model space is regularly sampled, for several imaging operators the inner products can be computed analytically, leading to a fast and affordable evaluation of the elements of $\mathbf{L}\mathbf{L}'$.

Notwithstanding some of the theoretical advantages of the data-space inverse, the approximation of the model-space inverse yields better results than the equivalent approximation of the data-space inverse for the partial stacking of the North Sea data set. Figures 9.7 and 9.8 show in-line sections cut at the same location as the sections shown in Figure 9.5. Figures 9.7 was obtained by nearest-neighborhood interpolation, while 9.8 was obtained by linear interpolation. In both figures, the sections on the top were obtained without any normalization

weights; for the ones in the middle, w_i^d were used, and for the ones at the bottom, ${}^{\Sigma}w_i^m$ were used. For the model-space inverses, the inversion was stabilized by setting $\epsilon_w = 0.1$ [equation (9.8)]. As predicted by the theoretical discussion [equation (9.26)], with nearest-neighborhood interpolation (Figure 9.7), the middle and bottom sections are the same. In this case, the coverage gap is only partially compensated by the weights. When linear interpolation is used (Figure 9.8), the coverage gap is better compensated by the model-space approximate inverse (bottom) than by the data-space approximate inverse (middle). However, the data-space inverse produces a better model than the model-space inverse obtained by using the diagonal of $\mathbf{L}'\mathbf{L}$ (middle panel of Figure 9.5). Figure 9.9 shows time slices cut at the same time as in Figure 9.6. The model-space inverse is slightly better than the data-space inverse.

9.2 Filling illumination gaps by model regularization

In the previous section we have seen that data gaps are a challenge for simple operator equalization. Near the gaps, the normalization weights may become large, and the signal-to-noise ratio of the results substantially degrades. Even worse, there might be no data available to fill the gaps. The divergence of the normalization weights is a sign that the least-squares estimation problem defined by equation (9.9) is poorly constrained. To constrain the solution, we can add a regularization term to the data-fitting term of equation (9.9).

There are several choices of regularization term to constrain the data interpolation problem. Often regularization terms are based on *a priori* assumptions on the reflectors geometry, or at least on the variability of the reflectivity in the spatial directions. When the data coverage is really poor, these "statistical" constraints are often unavoidable. However, using the information from traces at the surrounding midpoints can cause a resolution loss, because it may smooth true reflectivity changes. The challenge is to devise a method that maximizes the image resolution and minimizes artifacts.

If at all possible, we would like to constrain the estimation of data in the gaps using "physical" criteria, instead of statistical criteria. Because of the nature of the reflection mechanism, the reflection amplitudes can be assumed to be a smooth function of the reflection angle and azimuth. This observation leads to the idea that smoothing the data over offset and azimuth could be performed without losing resolution. Ideally, such smoothing should be done over reflection angles (aperture and azimuth) at the imaging point, not over offset and azimuth at the surface (Bloor et al., 1999; Prucha and Biondi, 2002; Kuehl and Sacchi, 2002). However, smoothing at the reflectors would require a full migration of the data. The migration step would make the method dependent on the accurate knowledge of the interval-velocity model. This reliance on the velocity model is inescapable when the gaps in the image are caused by the complexities in the velocity model itself (e.g. subsalt illumination (Bloor et al., 1999; Prucha and Biondi, 2002)), but it ought to be avoided when the imaging problems are caused by irregularities in the acquisition geometries.

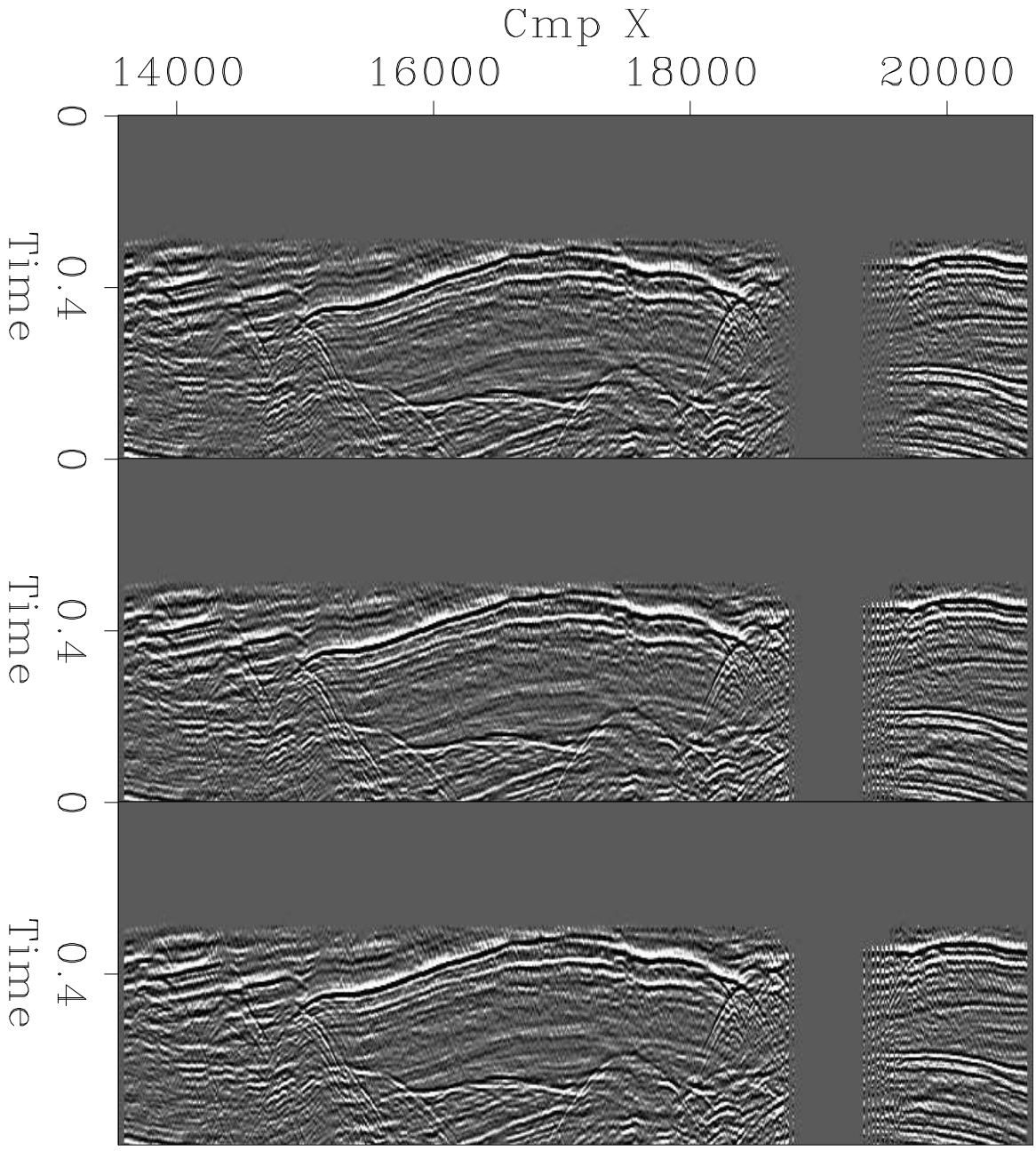


Figure 9.7: In-line sections obtained by partial stacking of the North Sea data set. Nearest-neighborhood interpolation was used with the application of no weights (top), w_i^d weights (middle), and ${}^{\Sigma}w_i^m$ weights (bottom). bin-CY-nsea [ER,M]

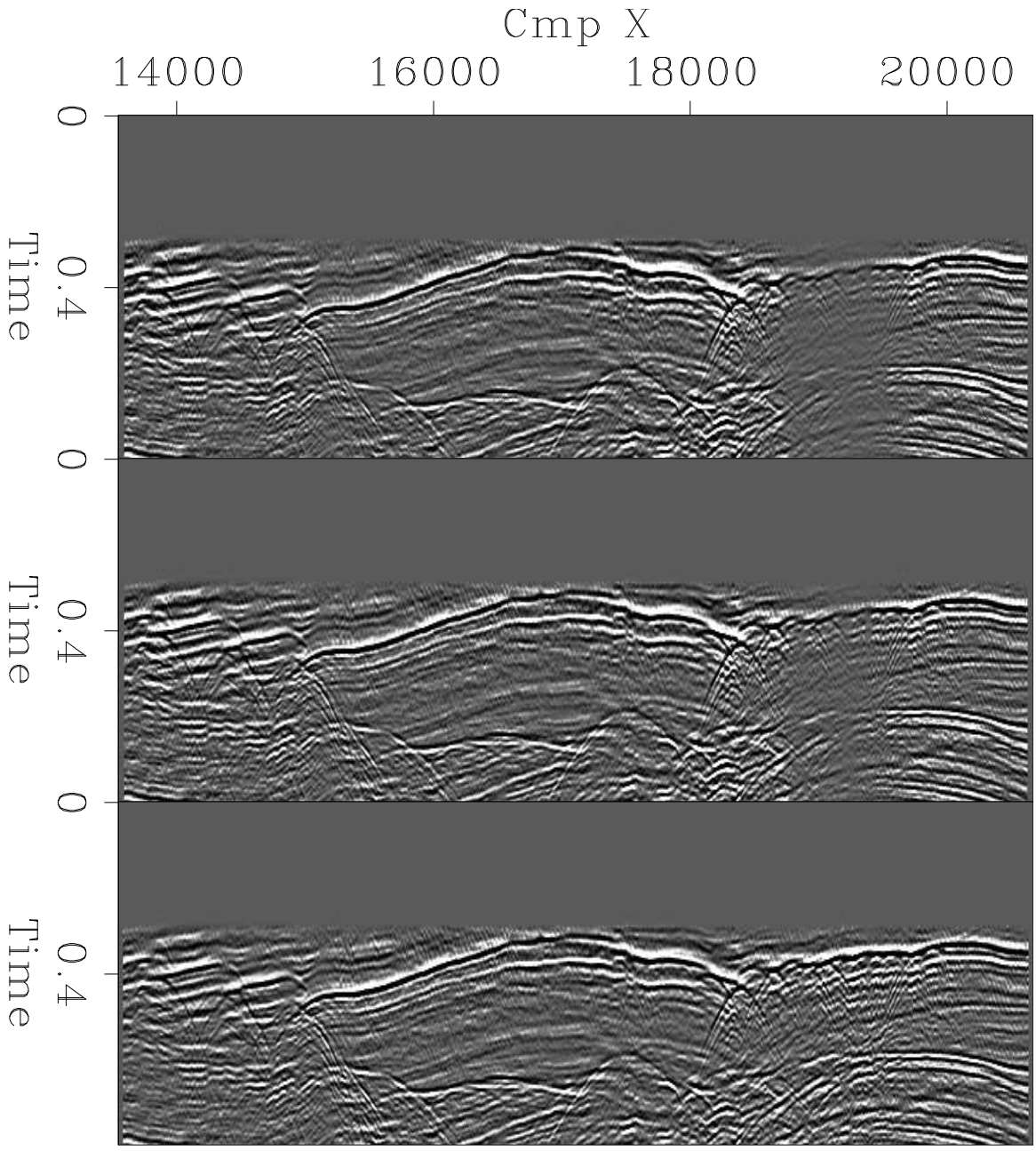


Figure 9.8: In-line sections obtained by partial stacking of the North Sea data set. Linear interpolation was used with the application of no weights (top), w_i^d weights (middle), and ${}^{\Sigma} w_i^m$ weights (bottom). bin-CY-nsea-lin [ER,M]

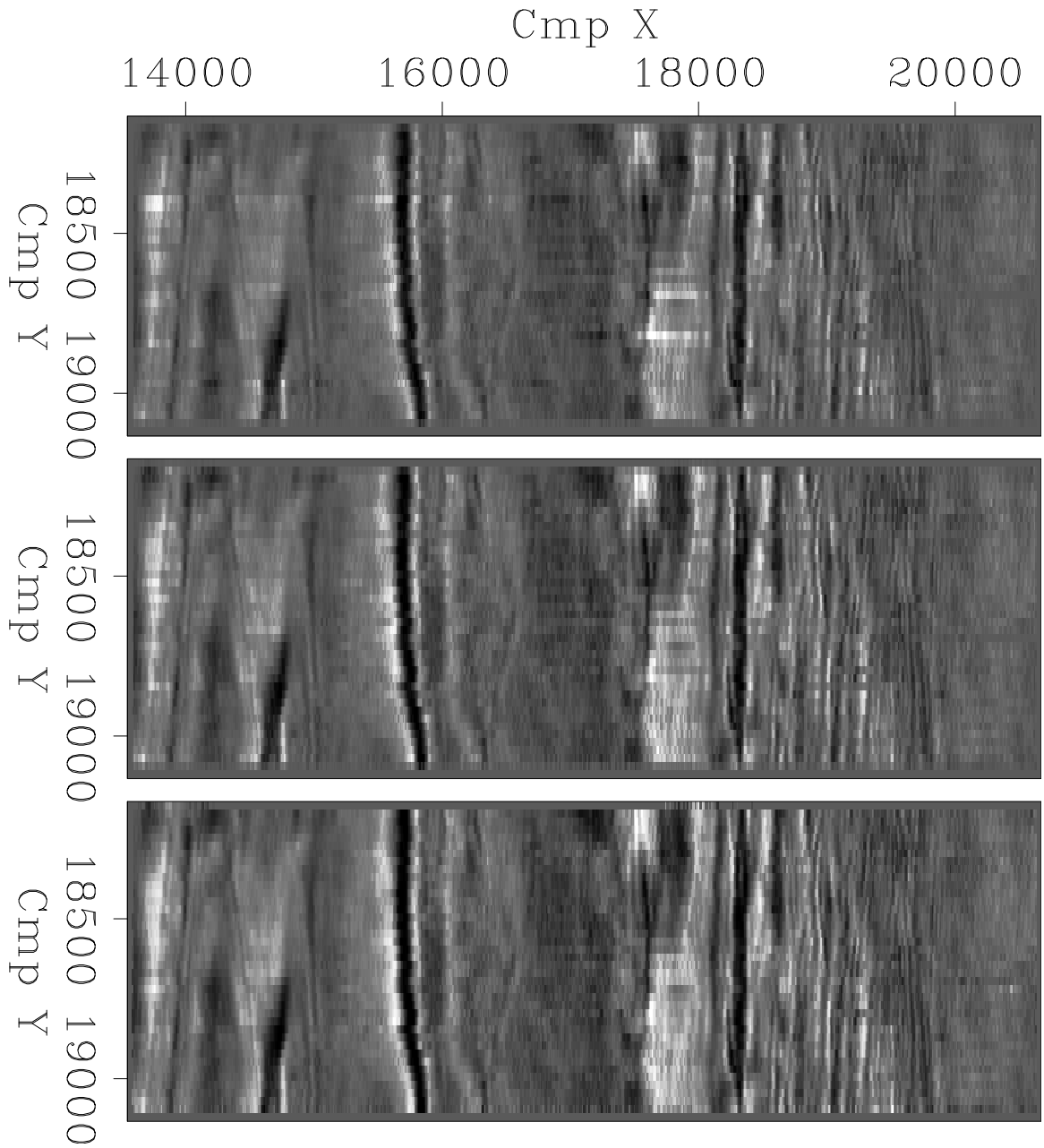


Figure 9.9: Time slices obtained by partial stacking of the North Sea data set. Linear interpolation was used with the application of no weights (top), w_i^d weights (middle), and ${}^{\Sigma}w_i^m$ weights (bottom). bin-CT-nsea-lin [ER,M]

9.2.1 Regularization by imposing continuity over offsets and azimuths

In the context of least-squares inversion, smoothing along offset/azimuth in the model space (e.g. uniformly sampled offset/azimuth cubes) can be accomplished by introducing a model-regularization term that penalizes variations of the seismic traces between the cubes. The simple least-squares problem of equation (9.9) then becomes

$$J_{\text{reg}}(\mathbf{m}) = \|\mathbf{L}\mathbf{m} - \mathbf{d}\|_2 + \epsilon_D \mathbf{m}' \left(\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}\right) \left(\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}\right) \mathbf{m}, \tag{9.29}$$

where the *roughener* operator $\mathbf{D_h}$ is

$$\mathbf{D_h} = \frac{1}{1-\rho_D} \begin{bmatrix} 1-\rho_D\mathbf{I} & 0 & 0 & \vdots & 0 & 0 \\ -\rho_D\mathbf{I} & \mathbf{I} & 0 & \vdots & 0 & 0 \\ 0 & -\rho_D\mathbf{I} & \mathbf{I} & \vdots & 0 & 0 \\ \ldots & \ldots & \ddots & \vdots & 0 & 0 \\ 0 & 0 & \ldots & \ddots & -\rho_D\mathbf{I} & \mathbf{I} \end{bmatrix}. \tag{9.30}$$

The coefficient ρ_D must be between 0 and 1. It determines the range over which we smooth the offset/azimuth cubes. The smaller we set the value of ρ_D, the narrower the smoothing range is.

Regularization with a roughener operator such as $\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}$ has the computational drawback that it substantially worsens the conditioning of the problem, and consequently slows down the convergence of an iterative inversion (Fomel, 2001). Several iterations would be needed to smooth the solution sufficiently, if we were to solve the least-squares problem expressed in equation (9.29). Even worse, if we could not afford the cost of an iterative solution, and attempted to approximate the least-squares inverse by a diagonal matrix, as discussed in the previous section, the regularization term would have almost no effect on the solution.

Fortunately, the problem is easy to precondition, because $\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}$ is easy to invert, since it is already factored into a lower block-diagonal operator $\mathbf{D_h}$ and an upper block-diagonal operator $\mathbf{D}_\mathbf{h}'$. Each of these factors can be easily inverted by recursion. Therefore, we can write the following preconditioned least-squares problem:

$$J_{\text{prec}}(\mathbf{p}) = \left\| \mathbf{L} \left(\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}\right)^{-1} \mathbf{p} - \mathbf{d} \right\|_2 + \epsilon_D \mathbf{p}' \mathbf{I} \mathbf{p}, \tag{9.31}$$

where $\mathbf{p} = \mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h} \mathbf{m}$ is the preconditioned model vector.

To take into account fold variations, we can introduce a diagonal scaling factor by applying the same theory discussed in the previous section. The weights for the regularized and preconditioned problem are thus computed as follows:

$$\mathbf{W}_\mathbf{I}^{-1} = \frac{\mathbf{diag}\left\{ \left[\left(\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}\right)^{-1} \mathbf{L}' \mathbf{L} \left(\mathbf{D}_\mathbf{h}' \mathbf{D}_\mathbf{h}\right)^{-1} + \epsilon_D \mathbf{I} \right] \mathbf{p}_{\text{ref}} \right\}}{\mathbf{diag}\left(\mathbf{p}_{\text{ref}}\right)}. \tag{9.32}$$

Notice that because of the nature of $\mathbf{D_h'D_h}$, $\mathbf{p}_{\text{ref}} = \mathbf{D_h'D_h m}_{\text{ref}} = \mathbf{D_h'D_h 1} = \mathbf{1}$; and $\left(\mathbf{D_h'D_h}\right)^{-1}\mathbf{1} = \mathbf{1}$; therefore some computation can be saved by computing the weights as follows:

$$\mathbf{W_I}^{-1} = \frac{\mathbf{diag}\left\{\left[\left(\mathbf{D_h'D_h}\right)^{-1}\mathbf{L'L} + \epsilon_D\mathbf{I}\right]\mathbf{1}\right\}}{\mathbf{diag}(\mathbf{1})}. \tag{9.33}$$

In this case, the computational cost of twice applying the leaky integrator $\left(\mathbf{D_h'D_h}\right)^{-1}$ to the model vector is small, and thus the computational saving is trivial. However, when we introduce more expensive operators to smooth the data over offset/azimuth, substituting equation (9.32) with equation (9.33) halves the computational cost of evaluating the weights.

The solution of the problem obtained by normalizing the preconditioned adjoint is

$$\widetilde{\mathbf{m}} = \left(\mathbf{D_h'D_h}\right)^{-1}\mathbf{W_I}\left(\mathbf{D_h'D_h}\right)^{-1}\mathbf{L'd}. \tag{9.34}$$

The normalization weights could be used for another preconditioning step, by using them in another change of variables, $\mathbf{q} = \mathbf{W_I}^{-\frac{1}{2}}\mathbf{p}$. This final step would yield the following least-squares problem:

$$J_{\text{prec}}(\mathbf{q}) = \left\|\mathbf{L}\left(\mathbf{D_h'D_h}\right)^{-1}\mathbf{W_I}^{\frac{1}{2}}\mathbf{q} - \mathbf{d}\right\|_2 + \epsilon_D\mathbf{q'W_I q}. \tag{9.35}$$

If the problem expressed in equation (9.35) were to be solved iteratively, it would likely converge faster than either the original regularized problem [equation (9.29)] or the preconditioned problem [equation (9.31)].

9.2.2 Regularization by AMO

The main drawback of the method described above is that smoothing over offset/azimuth cubes by the inverse of the simple roughener operator expressed in equation (9.30) may result in loss of resolution when geological dips are present. It is well known that reflections from dipping interfaces are not flattened by NMO with the same velocity as reflections from flat interfaces. However, the method is easily generalized by substitution of the identity matrix in the lower diagonal of $\mathbf{D_h}$ with an appropriate operator that correctly transforms a common-offset/azimuth cube into an equivalent cube with a different offset and azimuth. This can be accomplished by using the AMO operator (Chapter 3), or by using an offset-continuation operator (Fomel, 2001). In this section I show examples of the application of the AMO operator in the regularization term.

Fortunately, since the cubes to be transformed are uniformly sampled, we can use a Fourier-domain formulation of AMO that is both efficient and straightforward to implement (Biondi

and Vlad, 2002). The roughener operator that includes AMO is then expressed as follows:

$$\widetilde{\mathbf{D}}_{\mathbf{h}} = \frac{1}{1-\rho_D} \begin{bmatrix} 1-\rho_D\mathbf{I} & 0 & 0 & \vdots & 0 & 0 \\ -\rho_D\mathbf{T}_{\mathbf{h}_{1,2}} & \mathbf{I} & 0 & \vdots & 0 & 0 \\ 0 & -\rho_D\mathbf{T}_{\mathbf{h}_{2,3}} & \mathbf{I} & \vdots & 0 & 0 \\ ... & ... & \ddots & \vdots & 0 & 0 \\ 0 & 0 & ... & \ddots & -\rho_D\mathbf{T}_{\mathbf{h}_{n-1,n}} & \mathbf{I} \end{bmatrix}, \tag{9.36}$$

where $\mathbf{T}_{\mathbf{h}_{i,i+1}}$ is the AMO operator that transforms the offset/azimuth cube i into the offset/azimuth cube $i+1$. The $\widetilde{\mathbf{D}}'_{\mathbf{h}}\widetilde{\mathbf{D}}_{\mathbf{h}}$ operator can also be easily inverted by recursion, and thus the least-squares problem obtained by substituting $\widetilde{\mathbf{D}}_{\mathbf{h}}$ for $\mathbf{D}_{\mathbf{h}}$ in equation (9.29) can also be easily preconditioned and normalized using the same techniques described in equation (9.31-9.35).

9.2.3 Imaging of a 3-D land data set

I illustrate the geometry-regularization methods presented in the first part of this section on a land data set recorded in South America. The data were shot with a cross-swath geometry. The shot lines are not perfectly aligned along the perpendicular to the receiver lines, but rather are oriented at an angle (about 18 degrees). Figure 9.10 shows the plot of the absolute offset vs. azimuth for a random sample of traces in the data set. For land data, the data set has fairly narrow-azimuth acquisition geometry. Figure 9.11 shows the fold map for the offset bin between 2.4 and 2.6 kilometers before regularization. Figure 9.12 shows the fold after regularization, for the same offset bin as in Figure 9.11.

Figure 9.13 shows three sections (depth slice at the top, cross-line section in the middle, and in-line section on the right) of the near-angle stack of the migrated data. At a depth of about 3.2 kilometers there is a producing reservoir with numerous wells. The only structural feature in the data is a region of steep folding/faulting at the reservoir level. There are subtle stratigraphic features at a shallower depth, such as the channel visible in the upper-left corner of the depth slice in the figure ($z = 1.9$ kilometers).

The processing sequence is comprised of the following steps: a) NMO, b) geometry regularization, c) inverse NMO, and d) 3-D prestack common-azimuth wave-equation migration (Chapter 7), with the imaging step designed to preserve relative amplitudes, as discussed by Sava et al. (2001). The velocity model was fairly simple. The NMO velocity was slightly varying in the lateral direction. However, I migrated the data using an interval velocity that was a function of depth alone.

To analyze the relative performances of the different geometry-regularization methods, I migrated the data after regularizing the geometries with three different methods. The first method is normalization by partial stack fold [equation (9.8)]; I will simply call this method *normalization*. The second method is normalization of the regularized and preconditioned solution without AMO [equation (9.34)]; I will simply call this method *regularization*. The

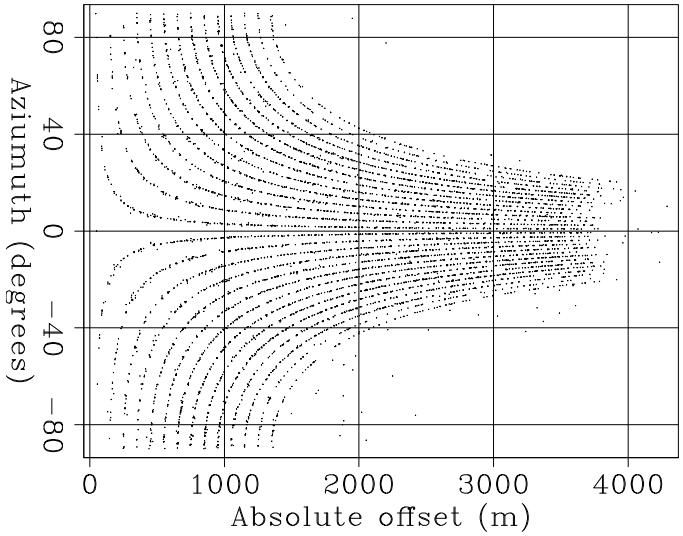


Figure 9.10: Offset-azimuth distribution of the land data set from South America. bin-OffAz [NR]

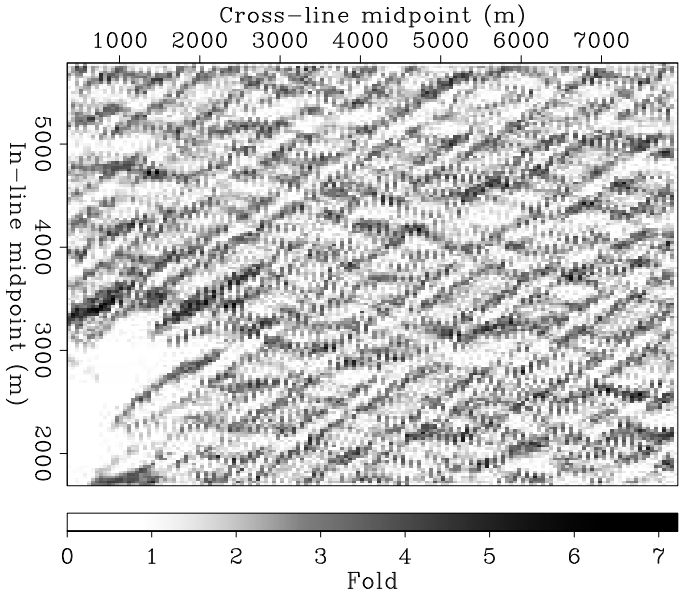


Figure 9.11: Fold map before regularization for the the offset bin between 2.4 and 2.6 kilometers. bin-F_lin_all_off_2535 [NR,M]

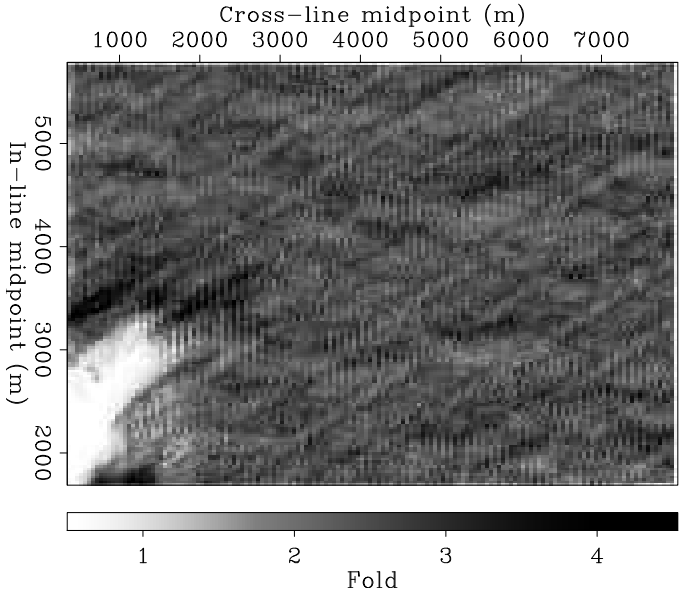


Figure 9.12: Fold map after regularization for the the offset bin between 2.4 and 2.6 kilometers. bin-F_lin_all_reg_off_2535 [NR,M]

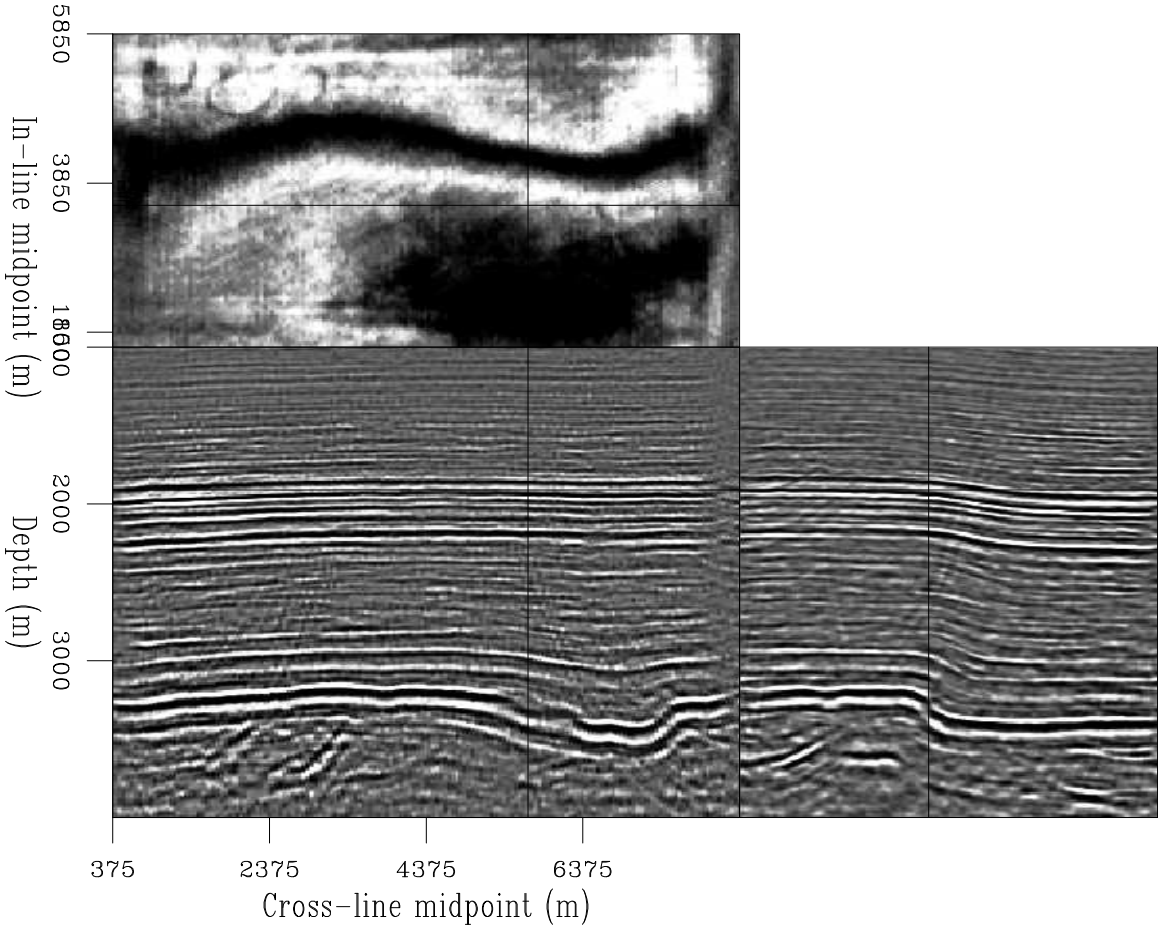


Figure 9.13: The orthogonal slices cut through the near-angle stack of the migrated cube. bin-Wm_lin_all_reg_vz_mig_nang_stack [NR]

third method is normalization of the regularized and preconditioned solution with AMO [equation (9.34) with $\mathbf{D_h} = \widetilde{\mathbf{D}}_\mathbf{h}$, as in equation (9.36)]; I will call this method *AMO regularization*.

The geometry regularizations produced common offset/azimuth cubes at zero azimuth, and with offset sampling of 195 m, in-line midpoint sampling of 25 m, and cross-line midpoint sampling of 50 m. By assuming reciprocity, we could ignore the offset sign to increase the effective fold of the common offset/azimuth cubes. Our first tests to regularize the data geometry produced common offset/azimuth cubes at zero azimuth (i.e. the data azimuth was ignored) because of the fairly limited azimuthal range at far offsets.

The offset axis was resampled at 65 m by simple interpolation before migration, to avoid offset aliasing in the downward continuation. This offset sampling was selected to avoid aliasing artifacts at the reservoir level. Finer sampling would be necessary to migrate shallower events without aliasing the higher frequencies present in the seismic signal. The effects of irregular geometries are more evident at shallow depth, and interesting stratigraphic features are visible in the migrated stack as shallow as 1 kilometer.

Effects of geometry regularization on the data

Figure 9.14 compares the three geometry-regularization methods discussed above by displaying one in-line section of the regularized data sets. Figure 9.14a shows the results for normalization, Figure 9.14b shows the results for regularization, and Figure 9.14c shows the results for AMO regularization. Comparing the in-line sections at one offset (3.38 kilometers) shows the advantages of both regularization and AMO regularization over simple normalization. The data amplitudes after normalization are still fairly uneven, and thus they are likely to produce artifacts during the migration. On the contrary, in the data obtained using regularization, the amplitudes are better balanced. The steeply dipping reflection from the fold at the reservoir level is better preserved in the AMO regularization results than in the simple regularization results. The reason is quite apparent when examining the data as a function of offset for one particular midpoint location. The dipping event is smiling upward after NMO, and thus it is attenuated by simple smoothing over offset.

Figure 9.15 shows a detail of the same line that illustrates the effects of the regularization term during the process. As in Figure 9.14, Figure 9.15a shows the results for normalization, Figure 9.15b shows the results for regularization, and Figure 9.15c shows the results for AMO regularization. An acquisition gap is clearly visible in the middle of the constant-offset (2.275 kilometers) section in panel (a). Simple normalization cannot fill the gap. On the contrary, the gap is filled in the regularized results, which exploit the information from the neighboring offsets. The gap in the dipping event is better filled by the AMO regularization, because the information from neighboring offsets is moved to the missing data consistently with their kinematics. The differences in behavior between the two regularization methods are apparent when analyzing the time slices in the upper part of the figure. As the dipping event moves toward higher offset (up), it also moves toward further midpoint (right). This movement is well seconded by the AMO regularization, while it is smoothed over by the simpler regularization scheme.

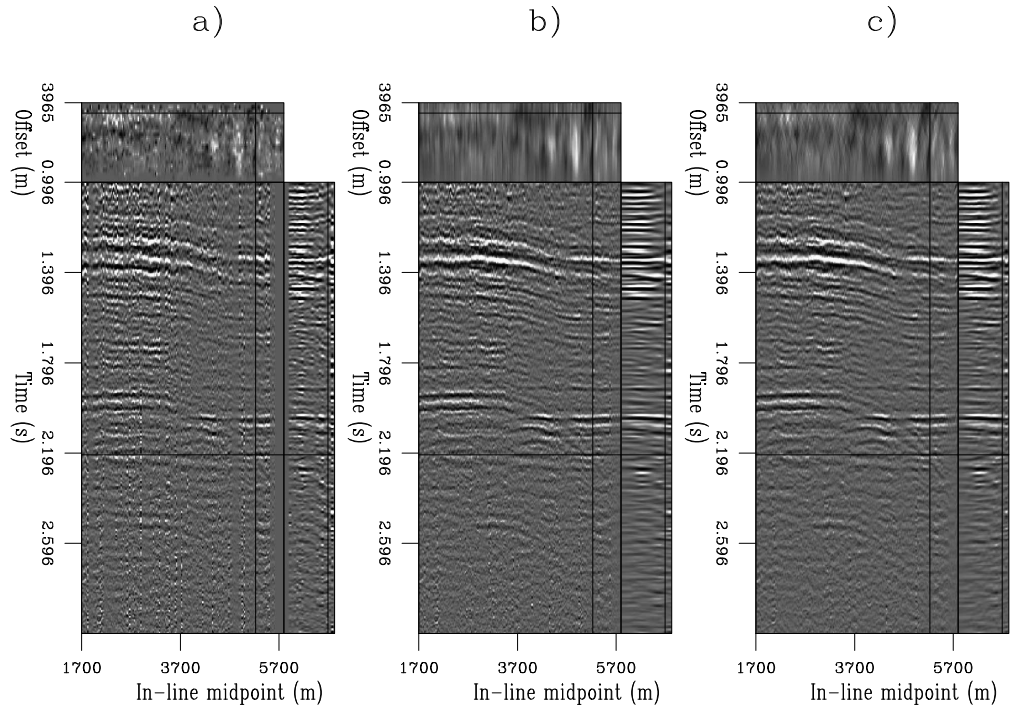


Figure 9.14: A prestack line ($y_m = 5.775$ kilometers) after geometry regularization with a) normalization, b) regularization, c) AMO regularization. bin-Data_abc [NR]

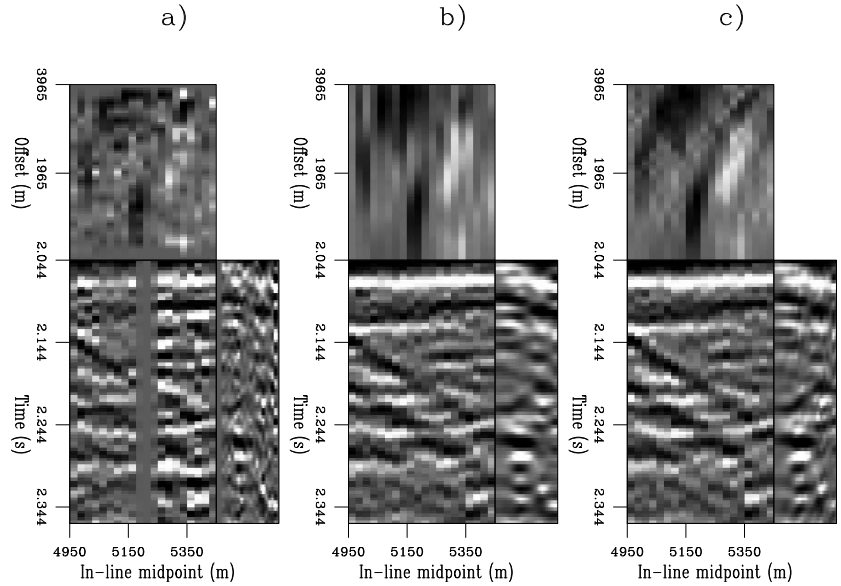


Figure 9.15: A detail of the prestack line shown in Figure 9.14. Notice the data gap in the middle of the common-offset (2.275 kilometers) section, and how it has been interpolated differently by regularization (panel b) and AMO regularization (panel c). bin-Data_win_abc [NR]

Although the individual common offset/azimuth cubes had some gaps along the line shown in the previous figures, these gaps could be filled by a regularization method that used the data recorded at nearby offsets and azimuths. However, it would be difficult to fill large gaps by using a non-iterative regularization algorithm. The next step for improving the data in areas with large acquisition gaps is to iteratively solve the least-squares problem defined in equation (9.35). Clapp (2003) compares the results obtained by applying the non-iterative regularization and normalization methodology described above [equation (9.34)], with the results of an iterative solution by a conjugate-gradient algorithm of the least-squares problem defined in equation (9.31). In both cases, in the regularization term was defined using the AMO operator; that is, with $\mathbf{D_h} = \widetilde{\mathbf{D}}_\mathbf{h}$.

Figure 9.16 shows three orthogonal slices cut through a common offset/azimuth cube obtained by applying the non-iterative regularization procedure. Notice the loss of amplitude clearly visible in the bottom-left corner of the time slice shown in Figure 9.16. In this area, there is a substantial gap in the coverage, as confirmed by the fold map shown in Figure 9.11. In contrast, the results of the iterative solution show a substantial improvement in the equalization of the amplitudes in the reconstructed data (Figure 9.17), even if only a small number of iterations – five in this example – were used. In the area where there is a substantial gap in the coverage (e.g. in the bottom-left corner), the iterative inversion was successful in equalizing the amplitudes, though the signal-to-noise ration of the reconstructed data is substantially worse in this area than in the rest of the data cube.

Effects of geometry regularization on the image

I migrated the data after the non-iterative geometry regularization using common-azimuth migration (Chapter 7); this produced three different prestack migrated images with the ADCIGs function of the in-line offset ray parameter, p_{x_h} (Section 6.2.1). Figure 9.18 compares one line extracted at $y_m = 5.775$ kilometers, for a wide reflection angle. As before, panel (a) displays the result obtained using normalization, panel (b) displays the result obtained using regularization, and panel (c) displays the result obtained using AMO regularization. The image obtained by normalization is more noisy than the ones obtained using regularization. The fold is slightly better-imaged in the results using AMO regularization (panel c).

These differences are confirmed in the depth slices shown in Figure 9.19. It shows two depth slices taken at the reservoir level ($z = 3.27$ kilometers) obtained by migration after geometry regularization with normalization (top), regularization (middle), and AMO regularization (bottom). Panels (a) show the images for a narrow reflection angle, while panels (b) show the images for a wide reflection angle. The depth slices obtained using normalization are noisier than the ones obtained using regularization. The dipping event (marked as "Fold" in the figures) is better-imaged in both the normalization results and in the AMO regularization results than in the regularization results.

Figure 9.20 shifts our attention to a shallower level, where there are no obvious structural features, but meandering channels are visible. The depth slices were taken at a depth of 1.91 kilometers. One channel is clearly visible in the upper left corner of the images, while another

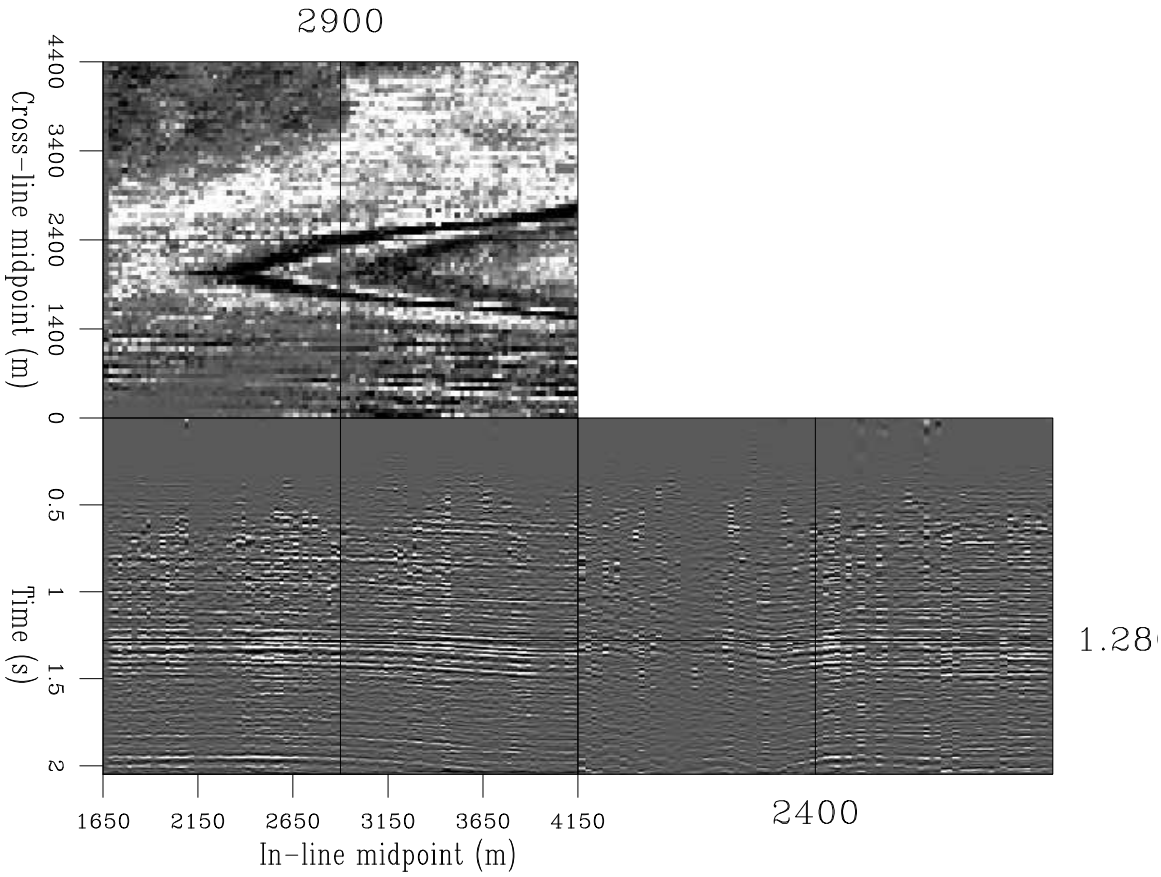


Figure 9.16: Three orthogonal slices cut through a common offset/azimuth cube obtained by applying the non-iterative regularization and normalization procedure summarized in equation (9.34). Comparing with the results of iterative inversion (Figure 9.17), notice the low amplitudes in the bottom-left corner of the time slice. (This figure is adapted from Clapp (2003).) bin-approx [NR,M]

one is (less clearly) visible around the lower right corner of the images. As for the previous figures, the slices on the left are taken for a narrow reflection angle, and the slices on the right are taken for a wide reflection angle. The images obtained using normalization are noisier and show more clearly the oblique acquisition footprints. In the narrow-angle image, the noise is so overwhelming that no channels are visible. The two regularization results are comparable for narrow angles. At wider angles, the AMO regularization image is slightly better focused. Subtle differences are noticeable in the imaging of both channels, where indicated by the arrows. The images of the upper channel are unfortunately affected by edge effects caused by the migration.

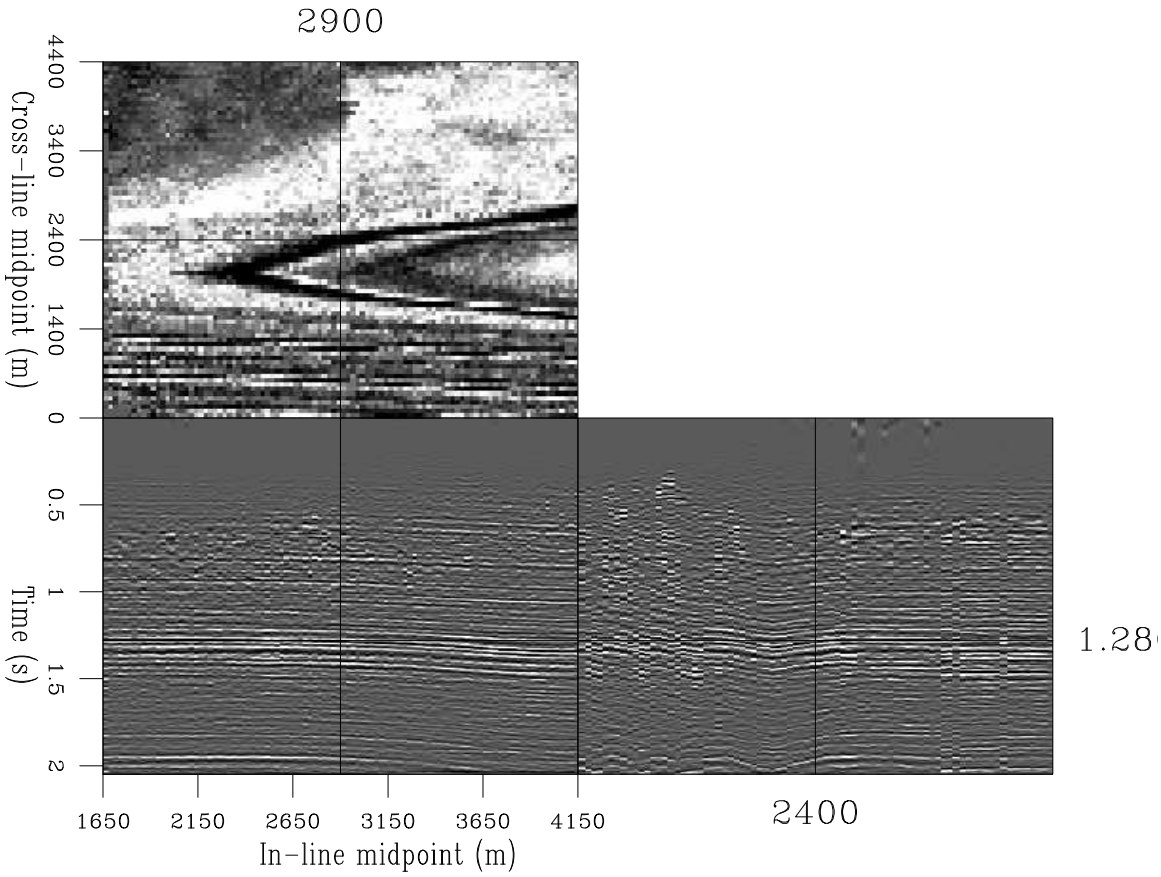


Figure 9.17: Three orthogonal slices cut through a common offset/azimuth cube obtained by 5 iterations of conjugate-gradient applied to the least-squares problem defined in equation (9.31). Comparing with the results of non-iterative regularization (Figure 9.16), notice the higher, but noisy, amplitudes in the bottom-left corner of the time slice. (This figure is adapted from Clapp (2003).) bin-inverse [NR,M]

9.3 Regularized inversion of prestack migration

The data-domain methods that I presented in the previous sections are appropriate when the incomplete illumination of the reflectors is caused by irregular acquisition geometries. Unfortunately, when the challenge is posed by a complex velocity function in the overburden that distorts the wavefield, we must switch to image-domain regularization methods, which are based on full prestack migration. There are two main factors that conspire to make image-domain regularization a computational problem much more challenging than data-domain regularization. First, the computational cost of prestack migration is substantially larger (by orders of magnitude) than the computational cost of the imaging operators (e.g. interpolation, AMO) that are used in the data-domain methods. Second, the non-iterative normalization methods, which are often successful in the data domain, are less effective in eliminating imaging artifacts when applied in the image domain. This second characteristic is a direct consequence of the fact that the zone of influence of prestack migration (i.e. the migration aperture) is much

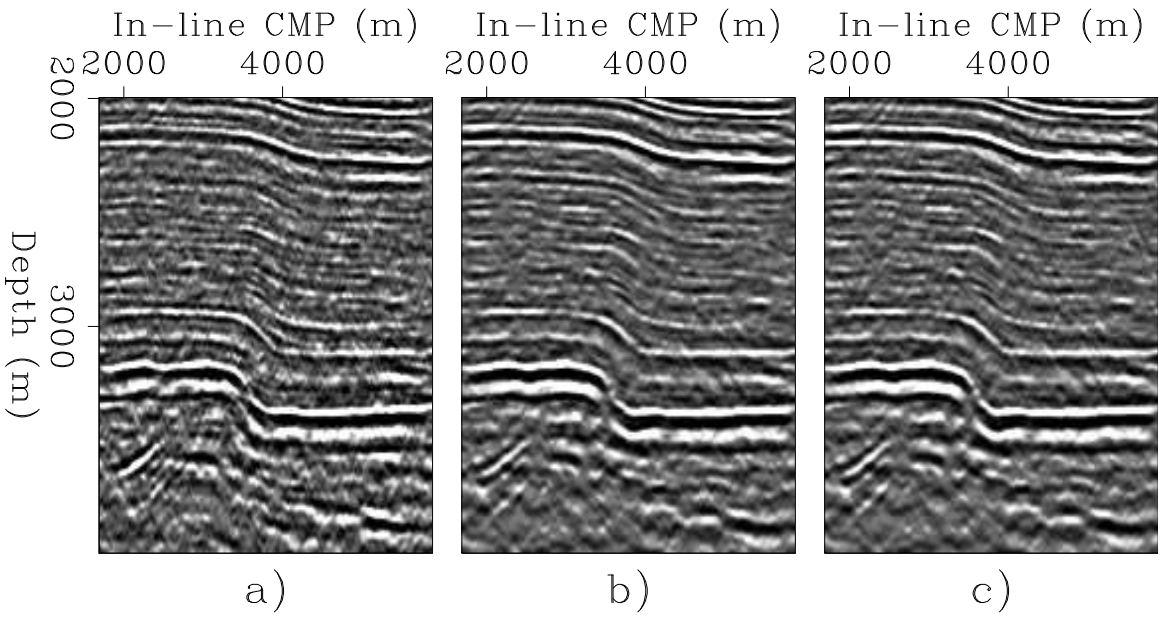


Figure 9.18: Images for wide reflection angle at constant cross-line location ($y_m = 5.775$ kilometers). The images were obtained after geometry regularization with a) normalization, b) regularization, and c) AMO regularization. bin-Line_abc [NR]

wider than the corresponding aperture of data-domain operators. When the model space of the imaging operator $\mathbf{L}$ is the image space, gaps in the wavefield, caused by either irregular acquisition or transmission effects, make the matrix $\mathbf{L}'\mathbf{L}$ even less diagonally dominant than when the model space of $\mathbf{L}$ is still in the data space. Consequently, the diagonal approximations of the inverse of $\mathbf{L}'\mathbf{L}$, which are at the basis of the non-iterative regularization methods discussed in the previous sections, become less effective, and we must perform iterative inversion of the computationally expensive prestack migration operator. The compounding of these two problems makes the computational cost of image-domain regularization methods exorbitant, and they might be difficult to justify with today's (2004) computers. However, the expected improvement in the cost-effectiveness of computer technology is likely to make these methods affordable a few years from now.

The imaging under salt edges is an application of image-domain regularization methods that is likely to have a practical impact in the future. Figure 9.21 shows a migrated in-line section (left panel) and an ADCIG (right panel) obtained from 2-D prestack migration of a 3-D data set acquired in the deep waters of the Gulf of Mexico. The vertical lines superimposed on the two panels mark the midpoint and aperture angle, respectively, where the other panel is cut through the prestack-image cube. The in-line section shows a clear illumination gap below the "nose" of the salt, and a less obvious, but nonetheless troublesome, gap further right, under the salt edge ($x_m \approx 24$ kilometers). Within the gap, and on both of its sides, the image is affected by migration artifacts caused by the discontinuities of the wavefield. The ADCIG has a gap near the middle of the angular range. The artifacts are even more evident in the ADCIG than in the in-line section; they appear as diffractions cutting across the illumination

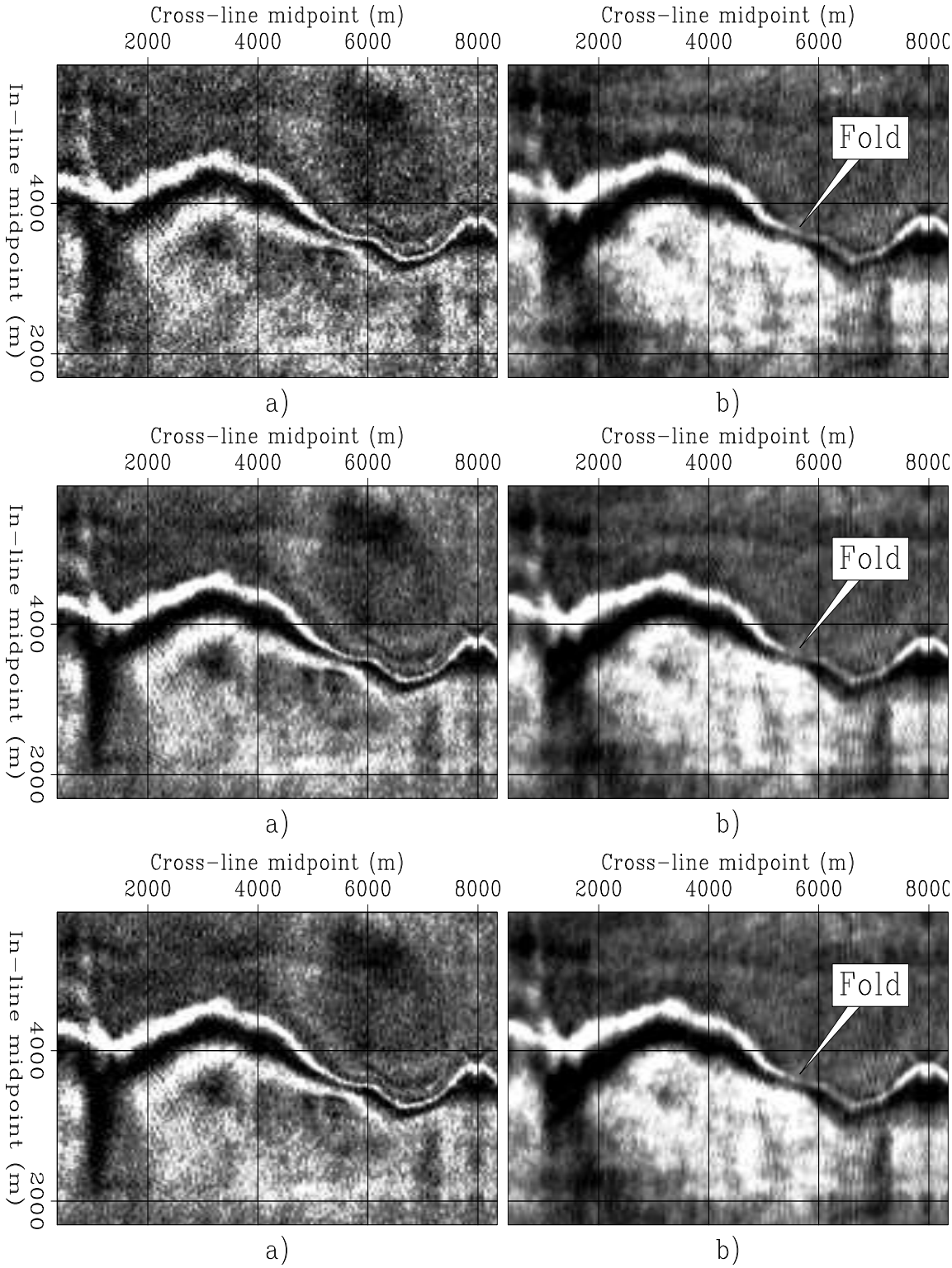


Figure 9.19: Depth slices, at a depth of 3.27 kilometers, obtained by migration after geometry regularization with normalization (top), regularization (middle), and AMO regularization (bottom). Narrow reflection angle (left), and wide reflection angle (right). bin-3270 [NR,M]

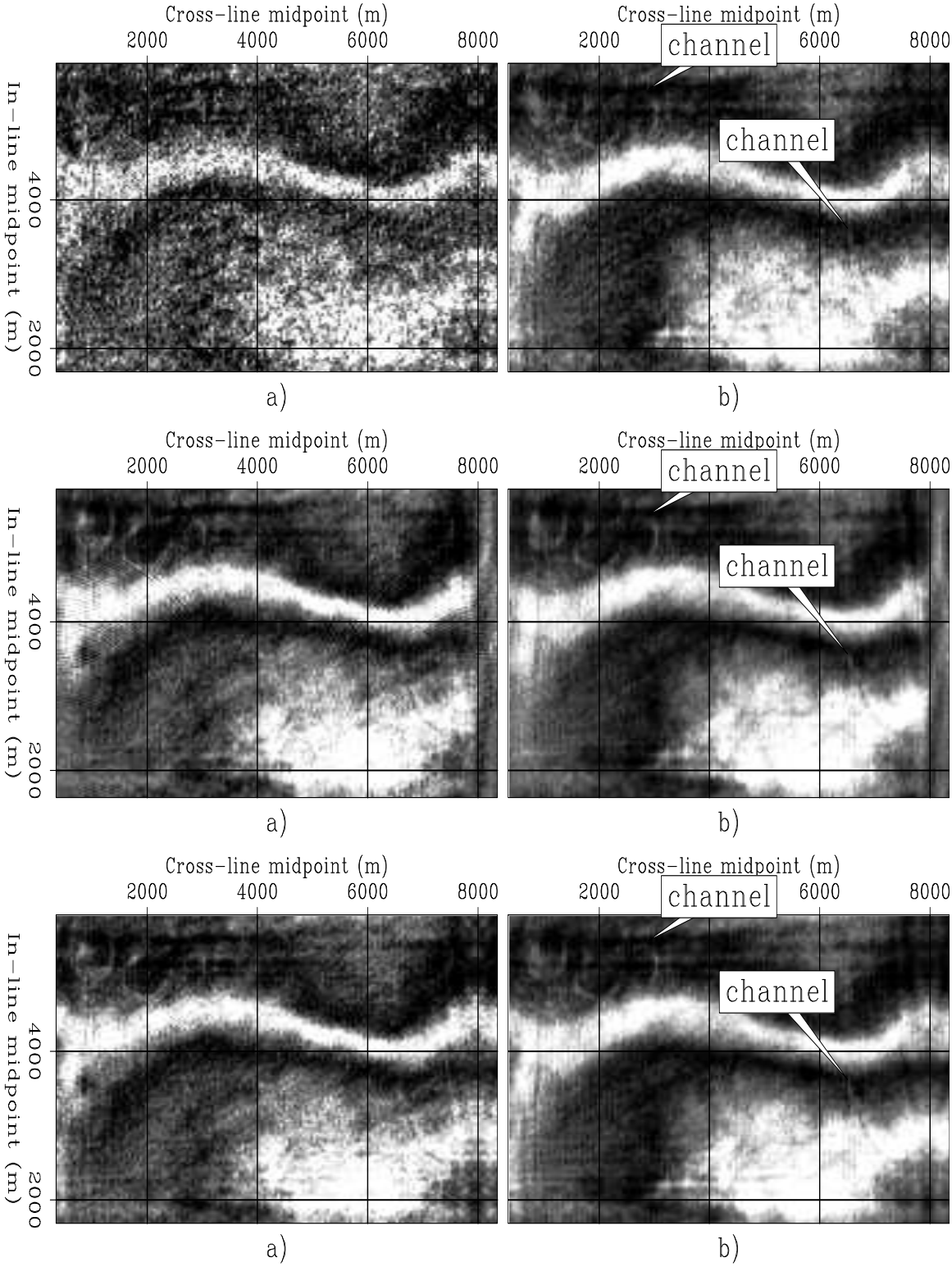


Figure 9.20: Depth slices, at a depth of 1.91 kilometers, obtained by migration after geometry regularization with normalization (top), regularization (middle), and AMO regularization (bottom). Narrow reflection angle (left), and wide reflection angle (right). bin-1910 [NR,M]

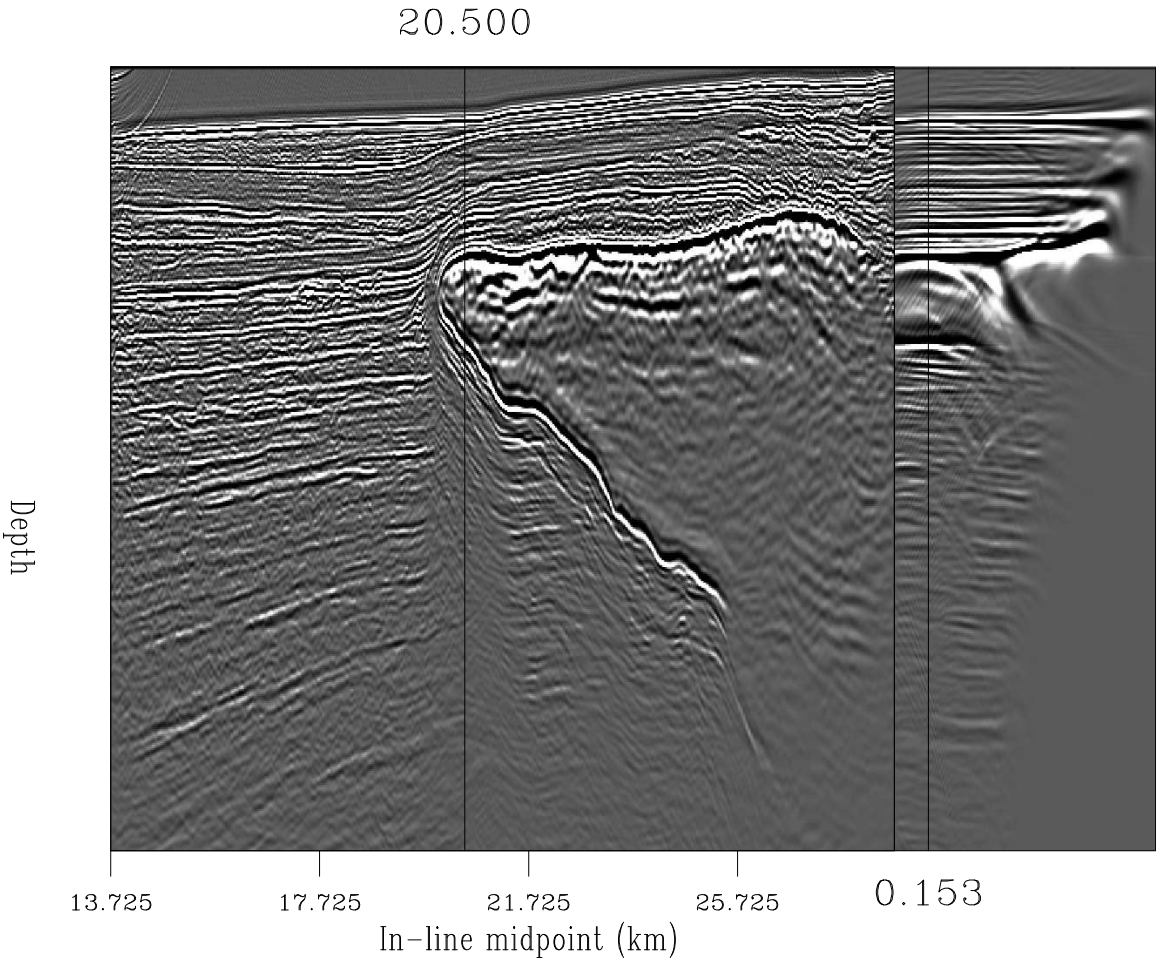


Figure 9.21: In-line section (left panel) and ADCIG (right panel) extracted from the prestack-image cube of a 3-D data set acquired in the deep waters of the Gulf of Mexico. Notice the illumination gaps and the imaging artifacts in both the in-line section and the ADCIG. (This figure is adapted from Clapp and Clapp (2004).) bin-subsalt-cube-mig2d [NR,M]

gap. A simple non-iterative normalization may be capable of equalizing the aperture across the illumination gaps, but it cannot remove the imaging artifacts.

Figure 9.22 shows an example of incomplete illumination under a salt edge from a synthetic data set (Paffenholz et al., 2002). The illumination problem is similar to the real Gulf of Mexico example shown in Figure 9.21. As Figure 9.21, Figure 9.22 shows an in-line section (left panel) and a ADCIG (right panel). However, in contrast to Figure 9.21, the in-line section shown in Figure 9.22 is the result of the stack of all the aperture angles.

The advantage of analyzing a synthetic data set is that the velocity function is perfectly known. The imaging problems cannot be attributed to an inaccurate velocity estimation, as it could in a real data case. Furthermore, we can gain valuable insight by accurately modeling wave propagation through the known velocity model with both ray-tracing and full-waveform modeling. Figure 9.23 shows the results of such modeling for a point source located in the

poorly illuminated area of the subsurface, at the in-line location of the ADCIG shown in Figure 9.22. The figure shows a snapshot of the wavefield at constant time; the corresponding rays and kinematic wavefront are superimposed onto the wavefield. In agreement with the imaging results shown in Figure 9.22, the energy corresponding to the middle of the aperture-angle range never reaches the surface. The left propagation lag of these events is deflected downward by the bottom of the salt nose.

Prucha and Biondi (2002) propose to address the illumination problem under salt edges by a regularized inversion of wavefield-continuation migration. They propose to minimize the following objective function:

$$J_{\text{reg}}(\mathbf{m}) = \|\mathbf{Lm} - \mathbf{d}\|_2 + \epsilon_D \mathbf{m}' \mathbf{A}' \mathbf{Am}, \tag{9.37}$$

where the modeling operator, $\mathbf{L}$, is the adjoint of full prestack wavefield-continuation migration, and the regularization operator, $\mathbf{A}$, is a roughener along the reflection-angle axes of the prestack image (see Chapter 6). Conceptually, this objective function is similar to the data-domain objective function expressed in equation (9.29). The regularization along the reflection angles is similarly motivated by the continuity of reflection coefficients as a function of aperture angle and azimuth, at least for precritical reflections.

The objective function in equation (9.37) needs a model-vector preconditioning for accelerating the convergence of iterative solutions. Assuming that we can efficiently invert the roughener operator $\mathbf{A}$, the preconditioned model vector is $\mathbf{p} = \mathbf{Am}$, and the objective function in equation (9.37) becomes the following objective function:

$$J_{\text{prec}}(\mathbf{p}) = \left\|\mathbf{LA}^{-1}\mathbf{p} - \mathbf{d}\right\|_2 + \epsilon_D \mathbf{p}' \mathbf{Ip}. \tag{9.38}$$

Figure 9.24 shows the results of ten iterations of a conjugate-gradient algorithm applied to the minimization of the objective function expressed in equation (9.38). The in-line section (left panel) and the ADCIG (right panel) in Figure 9.24 are cut at the same locations as the corresponding panels in Figure 9.21. The imaging artifacts are mostly suppressed in the inversion results, and some of the reflectors show better continuity at the edges of the gaps.

However, while iterative inversion has improved the image, the illumination gaps have not been filled. If more than ten iterations were performed, possibly the illumination gaps could be healed. On the other hand, the computational cost of ten conjugate-gradient iterations is already high, roughly equivalent to the computational cost of twenty prestack migrations. Prucha and Biondi (2002) propose to add an additional regularization term – dubbed "geological regularization" – that rewards interpreted geological dips. They show that this alternative strategy has the potential of filling in the illumination gaps. However, it has the drawback of imposing more subjective constraints than the smoothness along the reflection-angle axes imposed by the "geophysical" constraints applied to obtain the image shown in Figure 9.24.

Clapp et al. (2004) propose the alternative strategy of performing joint inversion of two or more 3-D data sets acquired along different directions. The illumination gaps are strongly dependent on the prevailing data azimuth, and thus two data sets acquired along different directions are likely to complement each other in the poorly illuminated areas. This strategy is viable in practice, because many subsalt areas of commercial interest have been covered with several data sets.

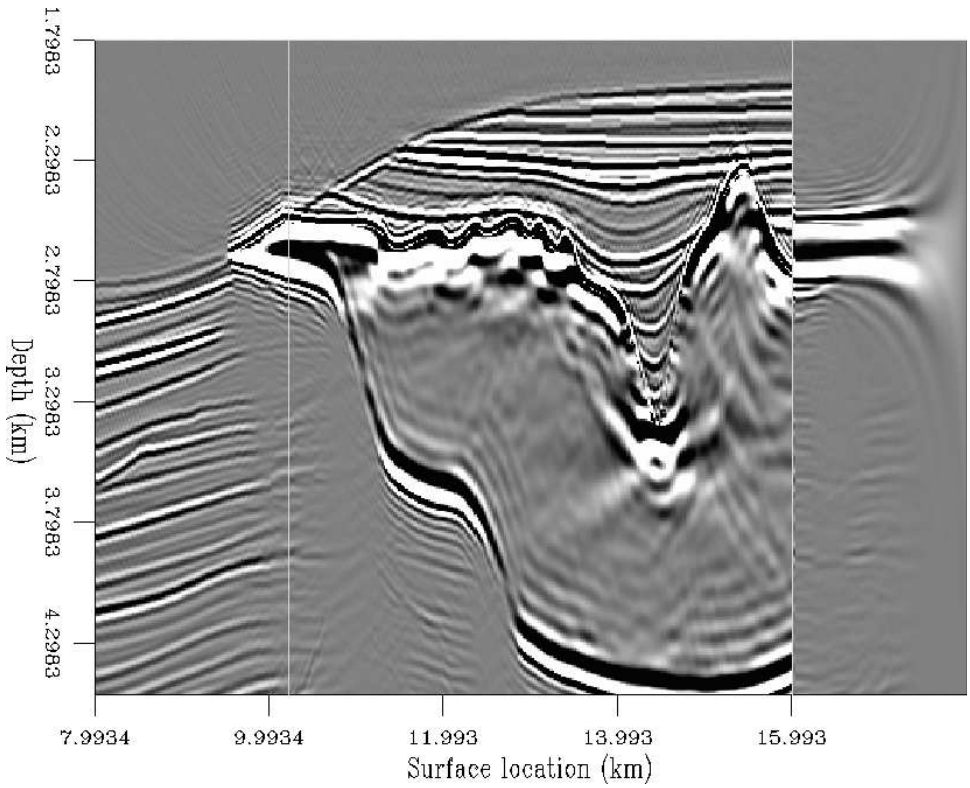


Figure 9.22: Migrated in-line section (left panel) and ADCIG (right panel) for a 2-D synthetic data set. The illumination gaps and the imaging artifacts are similar to the real-data case shown in Figure 9.21. bin-Cig-cube-win-7-new [NR]

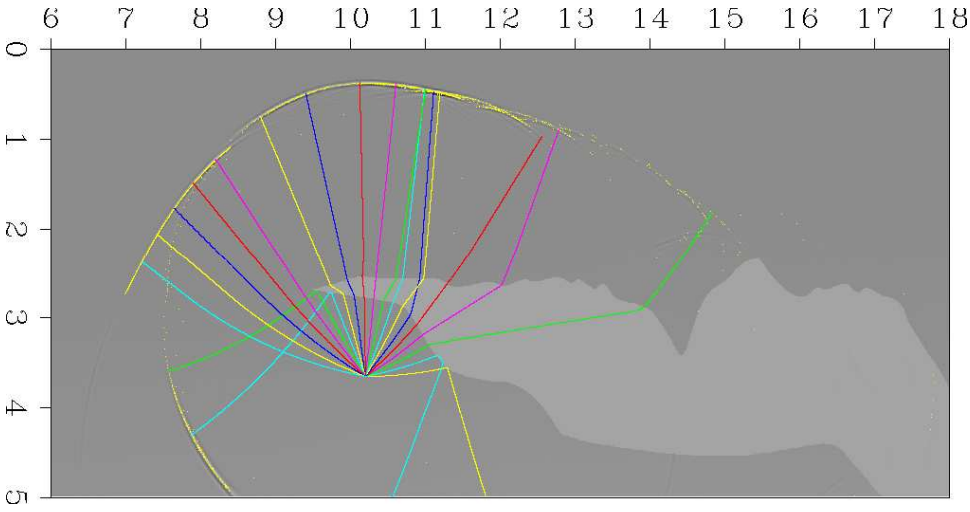


Figure 9.23: Waveform and ray-tracing modeling of seismic propagation from a point source located in the poorly illuminated area of the subsurface. Notice that the energy corresponding to the middle of the aperture-angle range never reaches the surface. bin-ccwft3-new-col [NR]

REFERENCES

Beasley, C. J., and Mobley, E., 1988, Amplitude and antialiasing treatment in $(x-t)$ domain DMO: Soc. Expl. Geophys., 58th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1113–1116.

Biondi, B., and Vlad, I., 2002, Amplitude preserving prestack imaging of irregularly sampled 3-D data: 72nd Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, 2170–2173.

Bloor, R., Albertin, U., Jaramillo, H., and Yingst, D., 1999, Equalised prestack depth migration: Soc. Expl. Geophys., 69th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1358–1361.

Canning, A., and Gardner, G. H. F., 1998, Reducing 3-d acquisition footprint for 3-D DMO and 3-D prestack migration: Geophysics, **63**, 1177–1183.

Chemingui, N., and Biondi, B., 2002, Seismic data reconstruction by inversion to common offset: Geophysics, **67**, 1575–1585.

Chemingui, N., 1999, Imaging irregularly sampled 3D prestacked data: Ph.D. thesis, Stanford University.

Claerbout, J., and Nichols, D., 1994, Spectral preconditioning: SEP–**82**, 183–186.

Claerbout, J. F., 2004, Image Estimation by Example: `http://sepwww.stanford.edu/sep/prof/index.html`.

Clapp, M. L., and Clapp, R. G., 2004, Regularized inversion for subsalt imaging: real data example: SEP–**115**, 283–294.

Clapp, M. L., Biondi, B., and Clapp, R. G., 2004, Regularized least-squares inversion for 3-D subsalt imaging: Soc. of Expl. Geophys., Expanded Abstracts, submitted.

Clapp, R. G., 2003, AMO regularization: Effective approximate inverses for amplitude preservation: SEP–**114**, 159–170.

Duquet, B., Marfurt, K. J., and Dellinger, J. A., 2000, Kirchhoff modeling, inversion for reflectivity, and subsurface illumination: Geophysics, **65**, no. 4, 1195–1209.

Fomel, S., 2001, Three-dimensional seismic data regularization: Ph.D. thesis, Stanford University.

Golub, G. H., and Loan, C. F. V., 1983, Matrix computations: John Hopkins University Press.

Hughes, T. J. R., 1987, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis: Prentice-Hall.

Kuehl, H., and Sacchi, M., 2002, Robust AVP estimation using least-squares wave-equation migration: Soc. of Expl. Geophys., 72nd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 281–284.

Nemeth, T., Wu, C., and Schuster, G. T., 1999, Least-squares migration of incomplete reflection data: Geophysics, **64**, no. 1, 208–221.

Paffenholz, J., McLain, B., Zaske, J., and Keliher, P., 2002, Subsalt multiple attenuation and imaging: Observations from the Sigsbee2B synthetic dataset: Soc. of Expl. Geophys., 72nd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 2122–2125.

Prucha, M. L., and Biondi, B. L., 2002, Subsalt event regularization with steering filters: 72nd Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, 1176–1179.

Rickett, J., 2003, Illumination-based normalization for wave-equation depth migration: Geophysics, **68**, 1371–1379.

Ronen, S., and Liner, C. L., 2000, Least-squares DMO and migration: Geophysics, **65**, 1364–1371.

Ronen, J., 1987, Wave equation trace interpolation: Geophysics, **52**, 973–984.

Sava, P., Biondi, B., and Fomel, S., 2001, Amplitude-preserved common image gathers by wave-equation migration: 71st Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, 296–299.

Strang, G., 1986, Linear Algebra and its Applications: Harcourt Brace Jovanovich College Publishers.

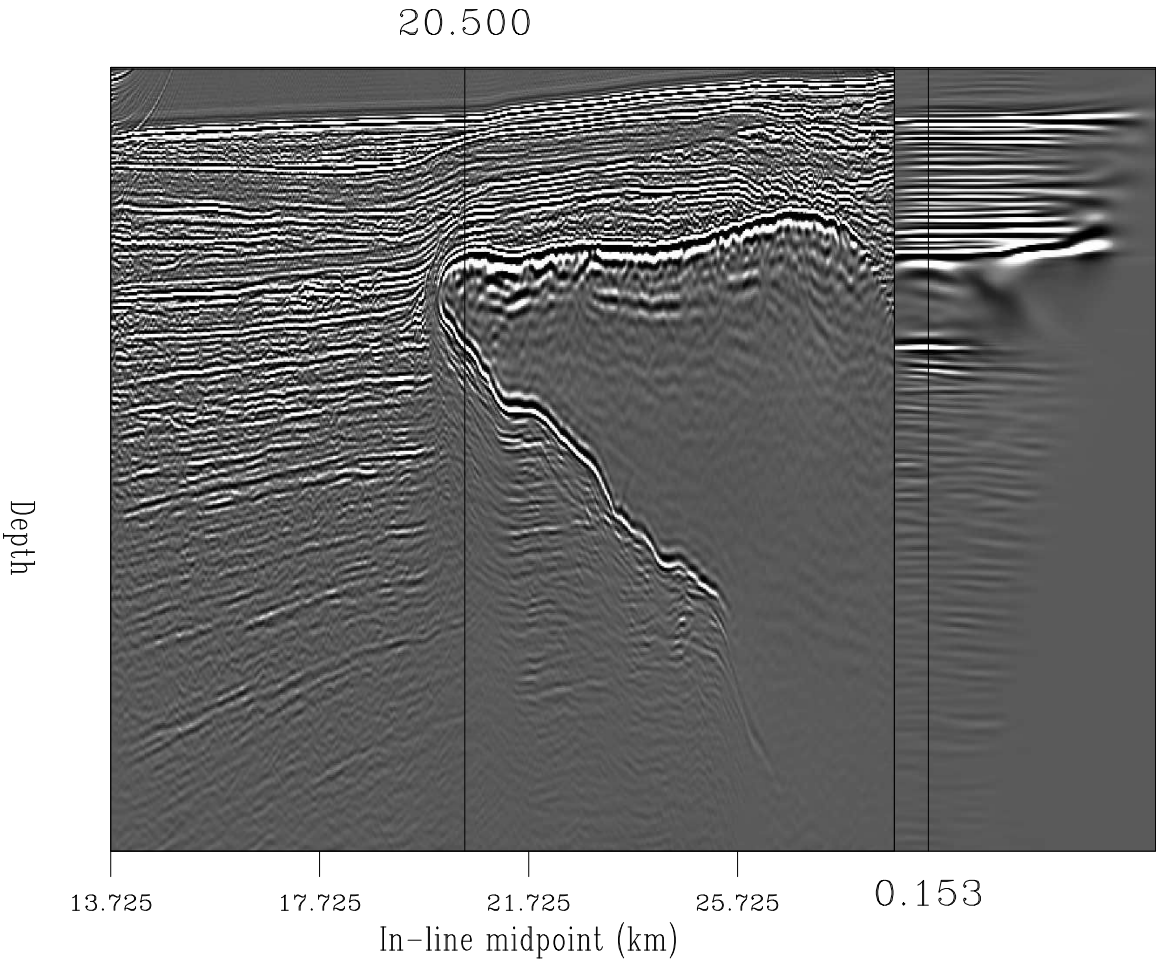


Figure 9.24: In-line section (left panel) and ADCIG (right panel) extracted from the prestack-image cube obtained by 10 iterations of the regularized inversion formulated in equation (9.38). These panels correspond to the panels shown in Figure 9.21. Notice the substantial reduction of imaging artifacts and increased continuity of the imaged events along both the midpoint and aperture-angle axes. (This figure is adapted from Clapp and Clapp (2004).) bin-subsalt-cube-1dprec.10it [NR,M]

Chapter 10

Principles of velocity analysis

In the previous chapters, we assumed that the velocity function, either RMS or interval velocity, was known when we analyzed methods to image zero-offset and prestack data. Of course, in reality that is not true, and we need to estimate velocity from the seismic data. The estimation of velocity from seismic data is an ill-posed inversion problem, in the sense that the data do not contain all the necessary information to define a velocity function that varies arbitrarily with depth and along the horizontal directions. Fortunately, in many practical situations we have a priori knowledge of the behavior of the velocity function that complements the information contained in the seismic data. The combination of these two sets of information; that is, the information contained in the data and the a priori one, are often sufficient to adequately constrain the problem. A typical example is the velocity estimation in a horizontally layered medium using the Dix inversion formula. If we assume that the subsurface is horizontally layered and isotropic, we can measure the RMS velocity function from the data, and we can apply the Dix formula to estimate the interval velocity function. However, in the presence of structural dips and lateral velocity variations, we must use more sophisticated analysis methods, or risk estimating grossly inaccurate velocity functions. The drawback of using more general methods than the Dix formula is that velocity estimation becomes poorly constrained as we add degrees of freedom to the set of admissible solutions.

Velocity plays a dual role in seismic imaging: 1) it affects the focusing of the data, and 2) it determines the positioning of the imaged reflectors in the physical space. Consequently, we call **focusing velocity** the velocity component that affects the data focusing, and **mapping velocity** the velocity component that affects the mapping of the reflectors. The V_{rms} function used by Kirchhoff time migration is a well-known example of a focusing-velocity function (Chapter 2). The interval velocity used to perform map migration of time-migrated images is an example of a mapping-velocity function. The focusing and mapping velocities are obviously correlated, and, in some cases, they are defined by a single velocity function and are estimated simultaneously. For example, in depth migration the interval velocity function determines both the focusing and the mapping of the data. However, it is important to distinguish the two velocity components, because their estimations have different characteristics. Furthermore, they are often estimated sequentially, with the estimation of the focusing velocity preceding the estimation of the mapping velocity.

The estimation of the focusing velocity is strictly linked with an imaging operator. For example, the estimation of interval velocity using the Dix formula is based on the simplest imaging procedure; that is, NMO+Stack (Chapter 3). This link is the main reason why this book covered imaging methods (stacking, DMO, migration, ...) before studying velocity estimation methods.

The focusing velocity is non-uniquely determined from the data; that is, many choices of a velocity function may focus the data equally well. Furthermore, because the imaging operators are not linear functions of the focusing velocity, and because of the presence of coherent noise in the data (e.g. multiple reflections), several locally optimal solutions may exist, and the convergence to a globally optimal solution is not assured. However, at least in principle, the data contain all the information required to determine a velocity function that properly focuses the reflections. In contrast, the mapping velocity is determined from the seismic data only through its correlation with the focusing velocity. Therefore, the definition of a velocity function that correctly maps reflectors in depth often requires other information (e.g. well data, geological models) in addition to seismic data. In this book, we will cover methods that estimate the mapping velocity from seismic data; but we will not discuss in detail ways to constrain the inversion for the mapping velocity using information other than the one contained in the seismic data. This is an important and challenging problem, but it is beyond the scope of this book.

The presence of anisotropy in the velocity function (Tsvankin, 2001) is another important challenge that I do not address in this book, and that is strictly related with the need of using additional data. Anisotropy has the largest effects on the mapping velocity, though, in some cases, it has some effects on the focusing velocity. Estimating velocity including an anisotropic term is therefore a problem even more poorly constrained than estimating an isotropic velocity model. It is often an impossible task without using additional geophysical or geological data.

Most velocity-estimation methods are based on the measurements of the kinematics of the reflections. An important differentiation among the wide-variety of methods is whether the kinematics are measured directly from the data in the time domain, or they are measured in the image domain from the results of migration. This chapter presents the methods defined in the data domain, whereas the following chapter presents some of the methods based on migration. When the geological structures are mild and the lateral velocity variations are smooth, the data-domain methods usually provide robust and accurate velocity estimates of both the focusing and mapping velocities. However, when the geological structures and/or the velocity function are complex, data-domain methods often fail; then we need to leverage the power of migration to focus and "simplify" the data and be able to reliably measure the kinematics of the reflections and converge toward accurate velocity estimates.

Data-domain methods are simple, and are well suited for introducing several of the concepts needed to develop the more complex method based on migration. This chapter presents the data-domain methods in order of increasing complexity of both the methodology and the geological setting in which they are appropriate to use. I start with the simplest method for inverting stacking velocities, which was introduced by Dix (1955) and it is valid for horizontally layered media. Second I examine the situations when the use of the Dix formula can be ex-

tended beyond its assumption of a layered Earth, and show examples illustrating the limitations and perils of such extensions. Finally I introduce the basic concepts of reflection tomography (Bishop et al., 1985; Stork and Clayton, 1991), which can be applied in the presence of mild structures and velocity variations with wavelengths of the same order as the data's maximum offset. The concepts and challenges of reflection tomography are illustrated in the analysis of the method introduced by Toldi (1985) for tomographically inverting stacking velocities.

10.1 Flat reflectors in $V_{rms}(\tau)$ media

We define the focusing velocity as the velocity function that enables an imaging algorithm to focus the data. Therefore, the focusing velocity needs to describe accurately the relative delays between events reflected from the *same* discontinuity in the subsurface and recorded at *different* surface locations. The first step of a velocity estimation procedure is to measure relative delays of reflections recorded within an ensemble of seismic traces, as a function of the trace source/receiver coordinates. A robust solution to this problem in the presence of complex geology is not by any means trivial.

The simplest method for measuring relative delays in the data is to measure the moveout along the offset axis. When the reflections are originated from a flat interface, and the medium above the interface is isotropic and homogeneous, the moveout is hyperbolic with offset. The inverse of the velocity is given by the time dip of the hyperbola's asymptote. In this case the focusing velocity is the velocity parameter that describes the hyperbola over which we can sum the data and produce a well-focused image of the planar interface; this velocity parameter is often called the **stacking velocity**. Stacking velocities are measured from the data by using **stacking-velocity spectra** (Taner and Koehler, 1969). A velocity spectrum is computed from a CMP gather by applying an NMO correction with several constant velocities, within a given velocity range. The flatness of the NMO-corrected gathers is measured by using a coherency measure, such as semblance or stacking power. The stacking velocities corresponding to the maxima of the spectra are then picked, either by hand or automatically.

When the subsurface is horizontally stratified the reflections from flat interfaces are still well described by hyperbolas, but now the stacking velocity varies according to the depth of the interface; that is, the stacking velocity is a function of the two-way traveltime, τ_ξ, to the apex of the hyperbolas. The stacking-velocity function is an average velocity, and it cannot be directly used to transform two-way traveltimes into depth; if we know just the stacking velocity, we cannot properly image the reflectors at depth. According to the terminology introduced above, the stacking-velocity function is the focusing velocity. To convert time to depth, we need to estimate the mapping-velocity function; that is, the propagation velocity, or **interval velocity**. Often the estimation of interval velocities is a much tougher problem than the estimation of average velocities, such as stacking velocity. Fortunately, in a layered medium the stacking velocity is related to the interval-velocity function by the simple relationship that follows:

$$V_{rms}(\tau_N)^2 = \frac{\sum_{i=1}^{N} v(\tau_i)^2 \, \Delta\tau_i}{\sum_{i=1}^{N} \Delta\tau_i}, \tag{10.1}$$

where $\Delta\tau_i$ is the "time thickness" of the i-th layer, and $\tau_N = \sum_i \Delta\tau_i$ is the total two-way traveltime to the bottom of the N-th layer. Because equation (10.1) expresses the root-mean-square of the interval velocity, the resulting velocity function is called RMS velocity. This formula is easily invertible, and the interval velocity in each layer $v\left(\tau_\xi\right)$ can be estimated from the RMS velocity function, $V_{rms}\left(\tau_\xi\right)$, by applying the following formula, also known as the **Dix formula** (1955),

$$v(\tau_i)^2 = \frac{\tau_i V_{rms}(\tau_i)^2 - \tau_{i-1} V_{rms}(\tau_{i-1})^2}{\Delta\tau_i}. \tag{10.2}$$

Once the interval velocity is known as a function of the vertical traveltime τ_ξ, the time-to-depth conversion is defined by the transformation that follows:

$$z_j = \frac{1}{2}\sum_{i=1}^{j} v(\tau_i)\,\Delta\tau_i, \tag{10.3}$$

where z_j is the depth at the bottom of the j-th layer.

In horizontally layered media, the RMS-velocity function can be directly measured from the data as the stacking-velocity function, because in a layered medium the two velocities are approximately equivalent. Therefore, in such simple media, the estimation of the mapping velocity from the data is a well-posed problem, although, in practice, the recursive inversion of equation (10.1) by the Dix formula in equation (10.2) can be often unstable, unless the RMS-velocity function has been properly smoothed. In contrast, in more complex media the stacking velocity is not equal to the RMS velocity, and the estimation of an interval-velocity function from an average-velocity function is far from unique. In the next section, we explore some of the issues arising in complex media, with the help of examples from the SEG-EAGE salt data set.

10.2 Dipping reflectors in $V_{rms}(\tau)$ media

The next step in the direction of more complex and realistic media is to consider media that have depth-dependent velocity [$V_{rms}(\tau)$ media], but contain dipping reflectors. In Chapter 3, we have seen that even in a homogeneous medium, the stacking velocity measured from dipping reflectors is not equal to the medium velocity. Stacking velocities are biased by a factor proportional to the inverse of the cosine of the dip [equation (3.12)]. In a homogeneous medium, this velocity bias is removed after the application of DMO (Chapter 3). Stacking velocities are similarly biased in a medium with depth-varying velocity, and DMO approximately corrects this bias. Therefore the quality of stacking-velocity analysis greatly improves after the application of DMO. The stacking velocities measured from the data after DMO are often called **DMO stacking velocities**. However, though DMO removes from the stacking-velocity spectra the bias caused by dips, it does not remove all the sources of errors and ambiguities in the velocity analysis. In particular, it does not resolve the ambiguities related to the fact that the zero-offset time of reflection from a dipping reflector is larger than the two-way vertical

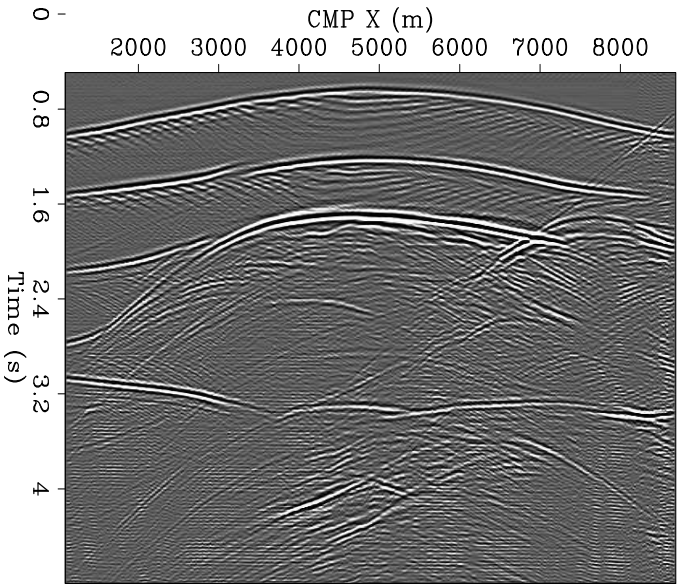


Figure 10.1: S1-S1' in-line section of the DMO-stacked cube of the SEG-EAGE salt data set. vel-Dmo-secw-2-stack [ER,M]

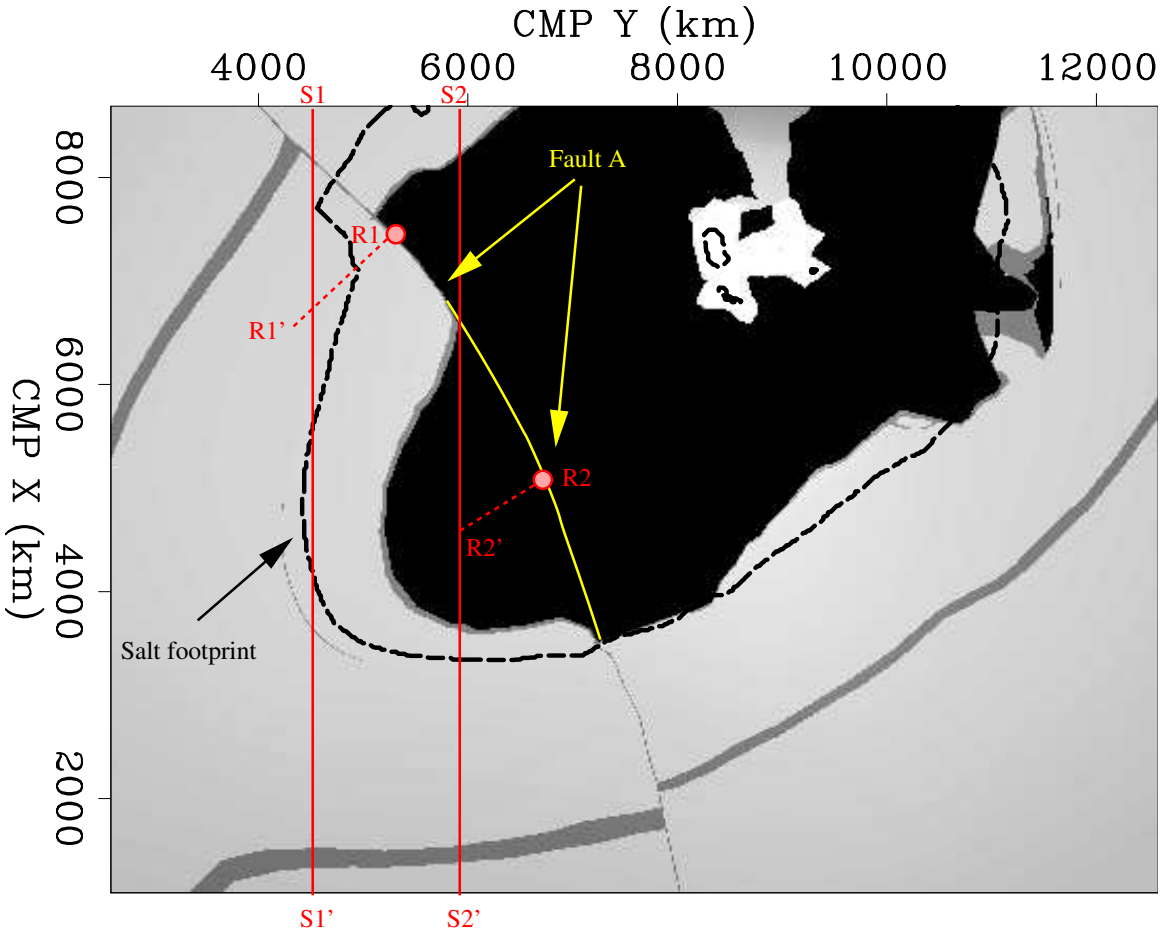


Figure 10.2: Depth slice of the velocity model for the SEG-EAGE salt data set. The two vertical lines (S1-S1' and S2-S2') superimposed onto the velocity function indicate the locations of the two data lines used for the examples discussed in Sections 10.2 and 10.3. vel-Depth-vel-stack [NR]

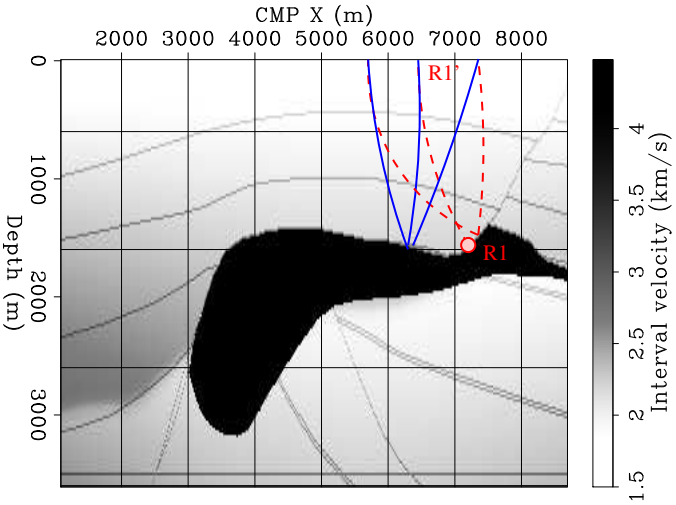


Figure 10.3: In-line section of the velocity model for the SEG-EAGE salt data set. This in-line section was taken at the cross-line location approximately corresponding to the the reflection point for the dipping event, indicated as R1 in Figure 10.2. vel-InLine-vel-3-ann [NR]

traveltime from the same dipping reflector to the surface. The following example illustrates this issue.

Figure 10.1 shows an in-line section of the DMO-stacked cube of the SEG-EAGE salt data set. Above the salt body the velocity function increases linearly with depth, but varies only slowly along the horizontal directions. Therefore, the data for reflections that are above the salt can be considered to be recorded in a $V_{rms}(\tau)$ medium. Figure 10.2 shows a depth slice of the velocity model taken at a depth of 1.5 kilometers. The location of the DMO-stack section shown in Figure 10.1 is indicated by the S1-S1' vertical line in Figure 10.2. We concentrate our attention on the reflections from the salt-sediment interface along the fault plane indicated by Fault A in Figure 10.2. These top-of-salt reflections correspond to the dipping event located around CMP X=6.5 kilometers in the stacked section S1-S1'. Notice that the dipping event arrives later than the almost flat event corresponding to another reflection from the top of the salt. The raypaths of these two top-of-salt events are shown in the schematics of Figure 10.3; the raypaths are superimposed onto an in-line section of the velocity model. This in-line section was taken at the cross-line location approximately corresponding the the reflection point for the dipping event, indicated as R1 in both Figure 10.2 and Figure 10.3. The raypath of the zero-offset reflection from the fault plane corresponds to the R1-R1' lines in the Figures. Notice that for the dipping event the reflection point of the non-zero-offset reflections is moved updip with respect to the zero-offset reflection point (R1). This is well-known effect is called **reflection-point dispersal**; DMO corrects the reflection-point dispersal by moving the dipping energy along the midpoint axes and horizontally aligning non-zero-offset events with zero-offset events.

Figure 10.4 shows the CMP gather located at CMP X=6.6 kilometers after simple binning and spatial interpolation. Panel (a) on the left shows the original CMP gather. Panel (b) in the middle shows the CMP gather after the application of NMO with the RMS velocity at the CMP location. Panel (c) on the right shows the velocity spectrum for the gather in (a). The moveout for the fault-plane reflection (zero-offset time is about 2 seconds) is overcorrected in panel (b) because of the dip effect. Also, the moveout for the other top-of-salt reflection (1.8

seconds) is slightly overcorrected in panel (b) because of the dip effects. Figure 10.5 shows the same CMP gather as in Figure 10.4, but after 3-D DMO has been applied to the data. DMO made the events in Figure 10.5b flatter than the events in Figure 10.4b. The moveout of the fault-plane reflection after DMO is now pointing slightly down because of the approximations involved in applying constant-velocity DMO to a $V_{rms}(\tau)$ medium.

Comparing Figure 10.5c with Figure 10.4c we can notice that DMO reduces the stacking velocity of the fault-plane reflection by about 250 m/s. Further, because DMO corrects the reflection-point dispersal, in the CMP gather after DMO, the reflections from the same event really originated at the same point in the subsurface, independent of their offsets. However, the velocity spectra after DMO is still misleading, because of the timing of the fault-plane reflection. This reflection arrives later than the other top-of-salt reflection, but it has a slightly lower velocity because the reflection point is actually shallower. This inversion of the stacking-velocity function does not correspond to a velocity inversion in the true RMS-velocity function. Figure 10.6 shows the true RMS velocity corresponding to the in-line section shown in Figure 10.1. Notice that at the CMP of interest, the RMS velocity is not influenced by the presence of the salt body. This is because the cross-line dip of the fault plane causes the acquisition plane S1-S1' to be displaced with respect to the reflection point R1. If we assumed the stacking-velocity function measured from the velocity spectrum in Figure 10.5c were equal to the RMS-velocity function, and then converted it to interval velocity by inverting equation (10.1), the resulting interval velocity function would be inaccurate. In particular, it would erroneously show a low-velocity layer between 1.8 seconds and 2 seconds, since the measured RMS velocity decreases with time. When the structural dips are mild, a careful picking of the stacking-velocity spectra and vertical smoothing of the RMS-velocity function can mitigate the negative effects of this phenomenon. However, when lateral velocity variations are present, in addition to dipping reflectors, the dangers of estimating velocity in the unmigrated domain are more significant, as the next example demonstrates.

10.3 Dipping reflectors in smooth $V_{rms}(\tau, x, y)$ media

The problems induced by the mispositioning of the stacking velocity are even more troublesome when the medium is laterally inhomogeneous [$V_{rms}(\tau, x, y)$ media]. The zero-offset arrivals from dipping reflectors are not only delayed but also shifted horizontally, and when the velocity is laterally varying, these horizontal shifts add further ambiguity to the interpretation of the stacking-velocity measurements.

These problems are illustrated by the following example. Figure 10.7 shows an in-line section of the DMO-stacked cube taken closer to the salt body than the in-line section shown in Figure 10.1. The location of this in-line section is indicated by the S2-S2' vertical line in Figure 10.2. As for the previous example, we concentrate on the fault-plane reflections at the salt-sediment interface. The zero-offset reflections from this interface are recorded at the surface locations around CMP X=4.5 kilometers. These reflections interfere with the reflections from the bottom of the salt body, as shown by the raypath schematics superimposed onto the velocity model in Figure 10.8.

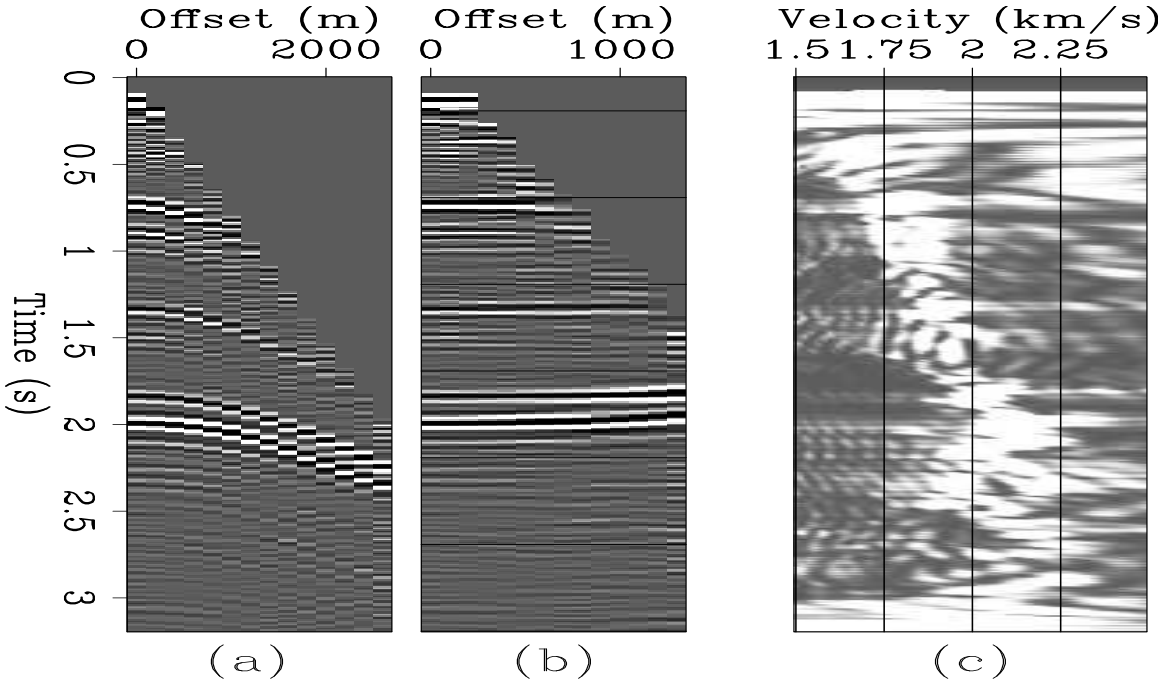


Figure 10.4: CMP gather taken at CMP X=6.6 kilometers in the in-line section shown in Figure10.1: (a) original CMP gather, (b) CMP gather after NMO, (c) velocity spectrum. vel-Bin-secw-2-2-all [ER,M]

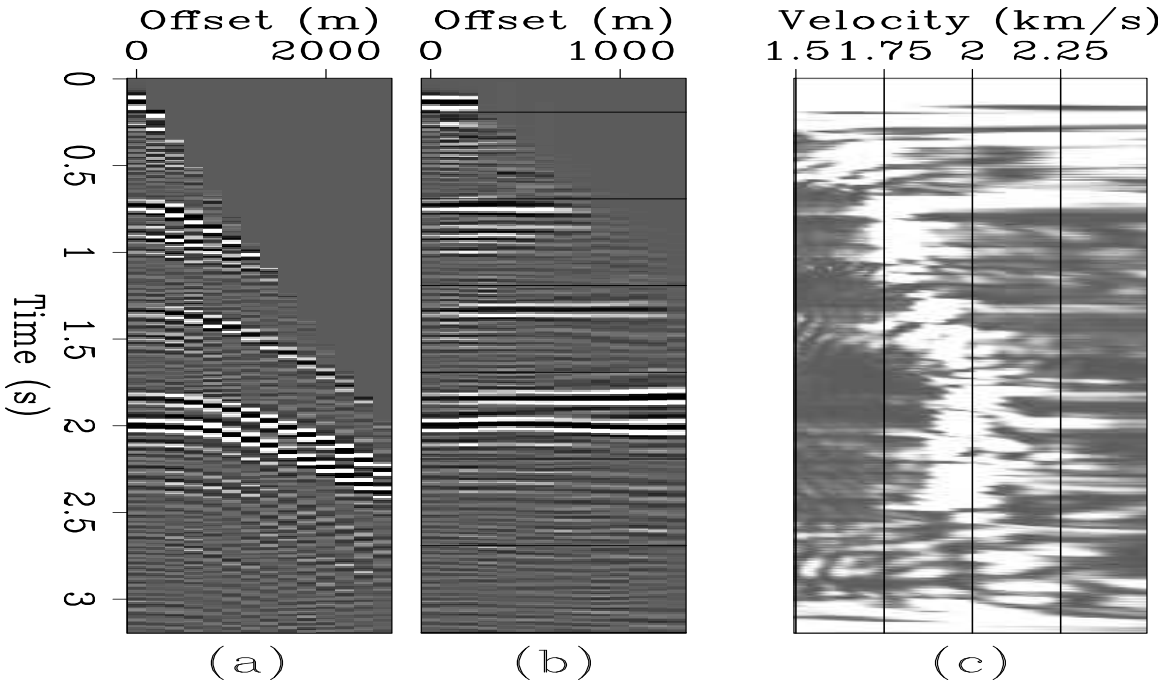


Figure 10.5: CMP gather after DMO taken at CMP X=6.6 kilometers in the in-line section shown in Figure10.1: (a) original CMP gather, (b) CMP gather after NMO, (c) velocity spectrum. vel-Dmo-secw-2-2-all [ER,M]

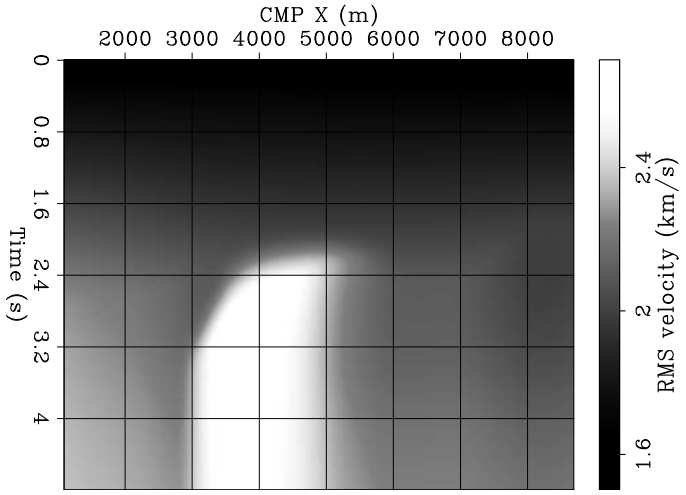


Figure 10.6: In-line section of the RMS velocity function corresponding to the stack section shown in Figure10.1. vel-Vrms-wind-2-overn [ER]

Figure 10.9 shows the CMP gather located at CMP X=4.5 kilometers after simple binning and spatial interpolation. Figure 10.10 shows the same CMP gather as in Figure 10.9, but after 3-D DMO has been applied to the data. The CMP gathers before and after DMO are fairly similar. In both gathers, the fault-plane reflections (zero-offset time is about 1.75 seconds) have been grossly undercorrected after NMO [panel (b)]. And both velocity spectra show the incongruity between the velocity of the fault-plane reflections (slow) and the bottom of the salt reflections (fast). This incongruity can be easily explained by comparing the stacked section shown in Figure 10.7 with the corresponding RMS-velocity function shown in Figure 10.11. The fault-plane reflections are recorded at the surface at a traveltime and surface location where the RMS velocity is very high because of the presence of the salt body, and thus they are undercorrected by NMO. In contrast, the salt-bottom reflections travel almost vertically, and thus their stacking velocity is in agreement with the RMS velocity.

Although the fault-plane and the salt-bottom reflections are recorded at zero offset at similar times and the same midpoint, they have originated at a distant location in the subsurface, and thus they propagated along different paths and "sample" different regions of the velocity function. If the stacking velocity from the dipping reflector were to be used for estimating the RMS velocity, large errors would be introduced in the velocity estimation. The origin of the problem is that, although DMO corrects for the effects of the reflector dips, it does not move the reflections to the time and lateral position of the reflection point. Events that reflect from different locations in the subsurface can cross in the unmigrated data. Only full migration (in contrast with partial migration) fully corrects these shifts and resolves the resulting ambiguities.

As for simple vertical variation, careful picking and smoothing (both vertical and horizontal) of the RMS velocity function can mitigate the problem. An even safer strategy would be to estimate interval velocity from stacking velocities, by inverting the relation defined by equation (10.1) with a smoothness constraint on the interval velocity (Clapp et al., 1998). If these simple precautions do not work, there are two possible ways of moving forward in the velocity-estimation process. If lateral velocity variations are the main problem, a tomographic method in the data domain may be the appropriate choice; the next section introduces such

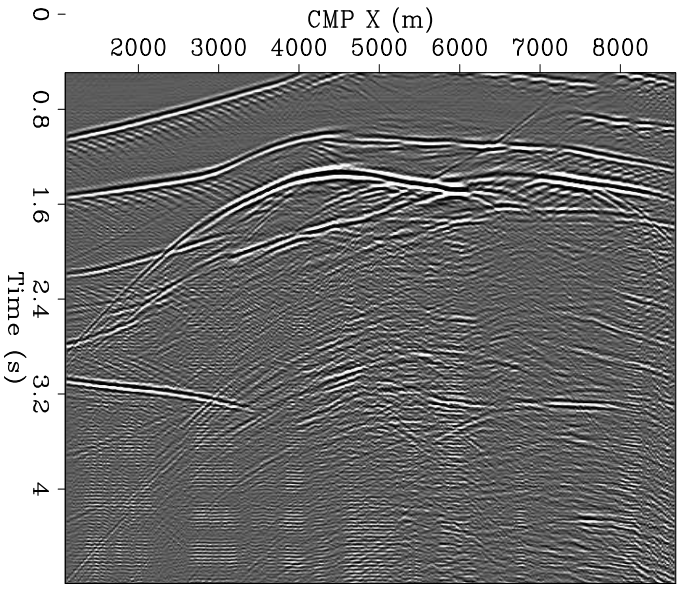


Figure 10.7: S2-S2' in-line section of the DMO-stacked cube of the SEG-EAGE salt data set. vel-Dmo-secw-1-stack [ER,M]

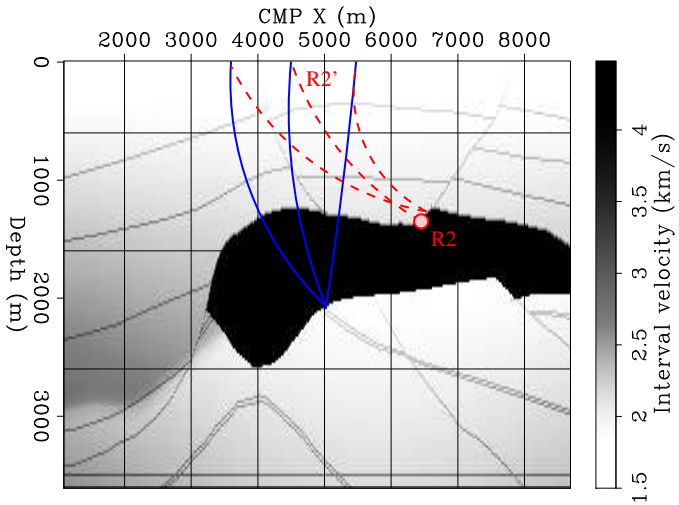


Figure 10.8: In-line section of the velocity model for the SEG-EAGE salt data set. This in-line section was taken at the cross-line location approximately corresponding to the the reflection point for the dipping event, indicated as R2 in Figure 10.2. vel-InLine-vel-1-ann [NR]

methods. If structural complexity is present, the velocity analysis methods based on migration, which I present in the next chapter (Chapter 11), are likely to be the best solution.

10.4 Traveltime reflection tomography

The Dix formula is based on the assumption that the Earth is horizontally layered. When lateral velocity variations are sufficiently strong, traveltime delays cannot be modeled analytically as well as they are modeled in a horizontally stratified Earth. The relationship between interval velocity and RMS velocity expressed in equation (10.1) becomes inaccurate, and consequently it is inappropriate to estimate interval velocity by inverting equation (10.1); that is, by applying the Dix formula in equation (10.2). In these cases, the estimation of interval velocity must be based on more accurate forward modeling of the traveltimes and an explicit fitting of the modeled traveltimes to "measured" traveltimes. Methods that estimate interval velocity by fitting modeled traveltimes to measured traveltimes using an inversion procedure are generally categorized as **reflection tomography** methods.

There are many, almost infinitely many, flavors of reflection tomography methods, but they are all based on the same principle and face similar challenges. In this section we will review the basic principles of reflection tomography and its main shortcomings. In its simplest form, reflection tomography assumes that the traveltimes of the reflections are readily available. This is not true in practice, and actually obtaining accurate and robust measures of reflected traveltimes is one of the main challenges of reflection tomography.

Picking individual traveltimes directly from the data is conceptually simple, and it was the method of choice of the earliest reflection tomography methods (Bishop et al., 1985; Stork and Clayton, 1991). However, picking traveltime is not only time-consuming, but also prone to errors, even with relatively simple and high-quality data. Instead of picking individual traveltimes, it is more robust to pick traveltimes and traveltime derivatives (or ray parameters) from ray-parameter spectra computed using local slant-stacks (Sword, 1987; Biondi, 1992; Billette et al., 1997). The algorithmic details for inverting these picks differ from the details for inverting individual traveltimes, but all these methods share the same fundamental principles. At the limit, the tomographic principles can be applied to invert stacking velocities measured from velocity spectra, as velocity spectra are even less noisy and easier to pick than local slant stacks (Toldi, 1985). In general, there is a trade-off between theoretical resolution and data uncertainty among the data-domain tomographic methods. Tomography based on the picks of individual traveltimes has a higher theoretical resolution than does tomography based on the picks of average parameters, such as stacking velocity. On the other hand, the loss of resolution caused by the increased uncertainty in the picks may outweigh the theoretical advantages.

The inversion of the information extracted by local stacking operator, such as local slant stacks (Sword, 1987; Billette et al., 1997) or beam stack (Biondi, 1992), is a potentially useful compromise in this trade-off. It has been recently revived by the development of the Common-Reflection-point Stacking (CRS) methodology (Perroud et al., 1999) and of the multifocusing imaging method (Gurevich and Landa, 2002). The parameter picked from the spectra obtained

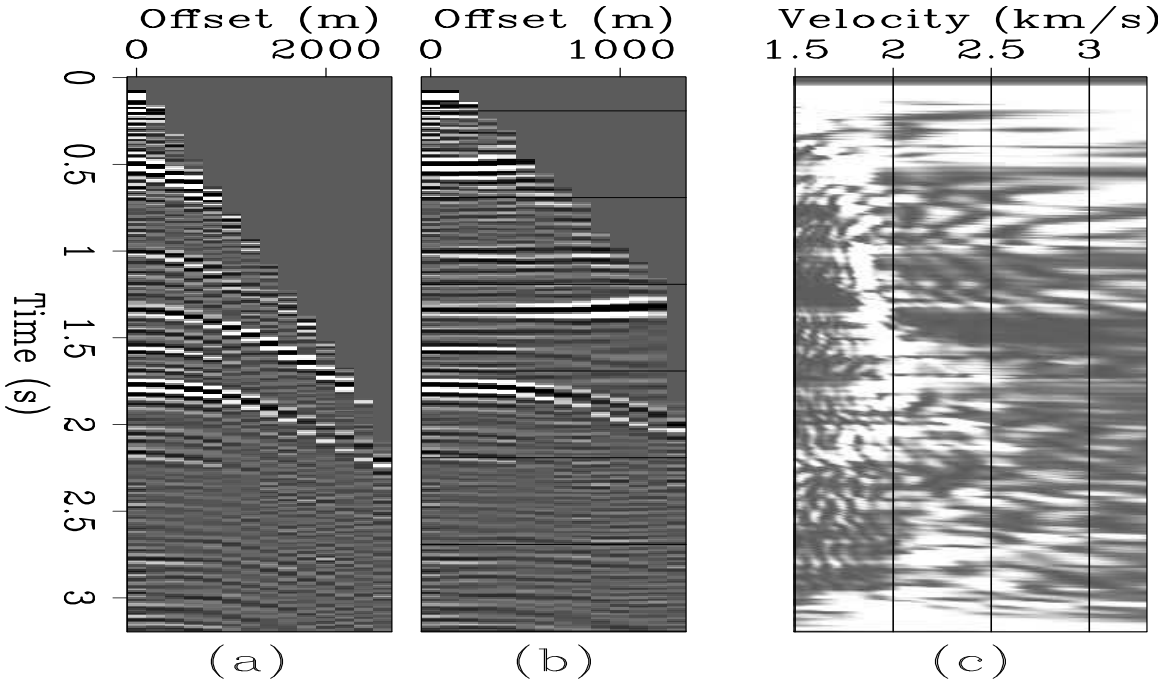


Figure 10.9: CMP gather taken at CMP X=4.5 kilometers in the in-line section shown in Figure10.7: (a) original CMP gather, (b) CMP gather after NMO, (c) velocity spectrum. vel-Bin-secw-1-1-all [ER,M]

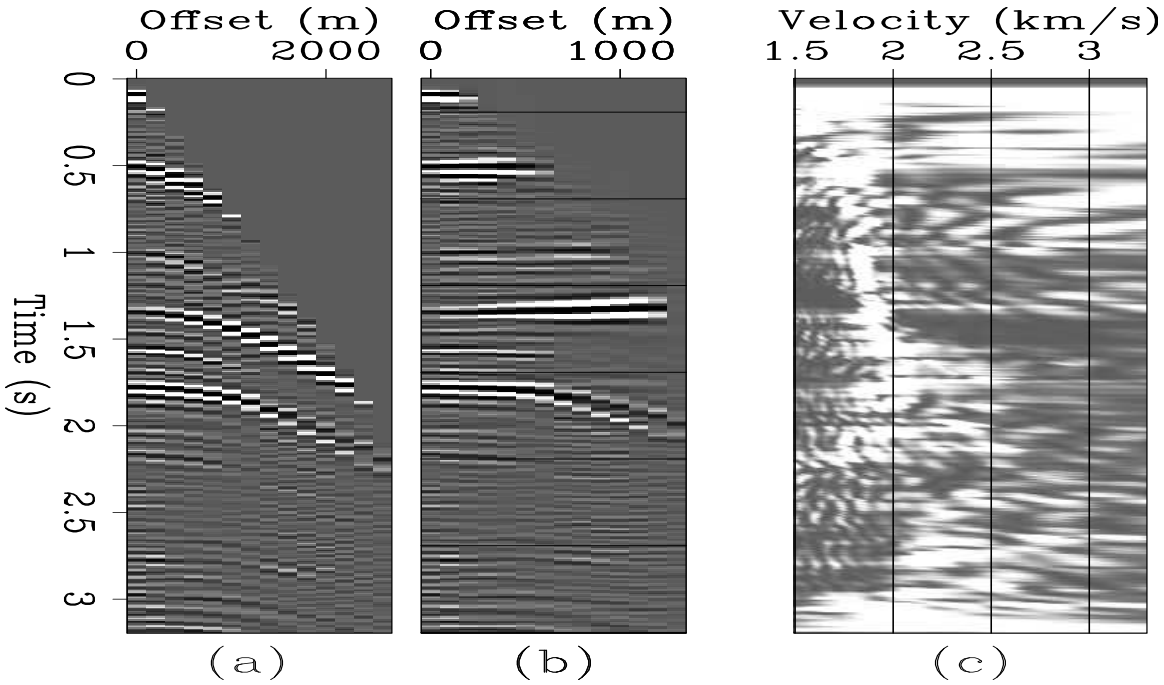


Figure 10.10: CMP gather after DMO taken at CMP X=4.5 kilometers in the in-line section shown in Figure10.7: (a) original CMP gather, (b) CMP gather after NMO, (c) velocity spectrum. vel-Dmo-secw-1-1-all [ER,M]

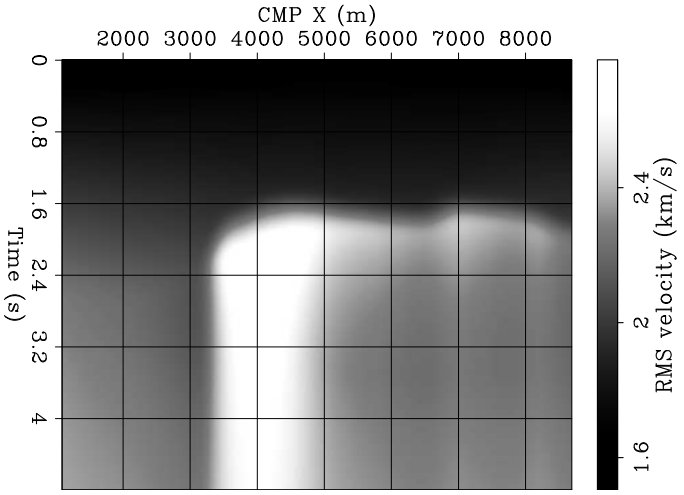


Figure 10.11: In-line section of the RMS-velocity function corresponding to the stacked section shown in Figure10.7. vel-Vrms-wind-1-overn [ER]

from these local stacking operators can be easily inverted for interval velocity, as Duveneck and Hubral (2002) have shown.

10.4.1 Formalization of traveltime tomography as an inverse problem

Traveltime tomography estimates the interval-velocity model by minimizing the differences between the traveltimes measured from the data, (t_{obs}), and the traveltimes computed using a forward modeling operator, (t_m). The forward modeling of the reflected traveltimes is usually accomplished by solving the Eikonal equation, either by ray tracing or by direct solutions using Eikonal solvers. More details on the numerical solution of the Eikonal equation can be found in Section 2.2.

An important characteristic of reflection tomography that distinguishes it from transmission tomography is that the traveltimes are functions not only of the velocity model, but also of the reflectors' geometry. The reflectors' geometry is itself a complex function of the velocity model and is usually obtained by interpreting the migrated image. This indirect dependence of the traveltimes on the velocity model, through the reflectors' geometry, increases the degree of non-linearity of the modeling operator with respect to the unknown parameters (velocity function), and consequently increases the difficulties of reflection tomography.

In mathematical terms, we formulate the velocity-estimation problem as the minimization of the following non-quadratic objective function:

$$J_{\text{s}}(\mathbf{s}) = \|\mathbf{t}_m(\mathbf{s},\mathbf{I}) - \mathbf{t}_{\text{obs}}\|_2\,, \tag{10.4}$$

as a function of the slowness vector $\mathbf{s}$. I have explicitly highlighted in my notation the dependence of the vector of modeled traveltimes, $\mathbf{t}_m$, on a vector of parameters, $\mathbf{I}$, which describes the reflectors' geometry.

The minimization problem expressed in equation (10.4) is poorly constrained. In most practical situations, many different slowness vectors minimize the objective function equally

well, within the accuracy of the data. To constrain the solution and avoid divergence during the numerical solution of the optimization problem, we add a model-regularization term to the simple objective function in equation (10.4). The objective function of the "regularized" problem is the following:

$$J_{\text{reg}}(\mathbf{s}) = \|\mathbf{t}_m(\mathbf{s},\mathbf{I}) - \mathbf{t}_{\text{obs}}\|_2 + s' \mathbf{C_s}^{-1} s, \tag{10.5}$$

where $\mathbf{C_s}$ is an estimate of the covariance of the slowness function. The rapidly varying components of the velocity functions are poorly determined from reflection data, because they have little influence on the kinematics of the reflections. Therefore, $\mathbf{s}$ is often constrained to be a smooth function, and consequently $\mathbf{C_s}^{-1}$ is set to be a roughener operator. Section 11.4.4 discusses in more detail how the model covariance can be actually estimated from the data in order to improve the geological likelihood of the final velocity model. An alternative approach for constraining the velocity function, is to define it by using smooth basis functions, such as cubic splines . In some areas (e.g. North Sea), it is also advantageous to use a geologically-driven sparse parametrization of the velocity function. In Section 11.4.4 I will also discuss the relative advantages and disadvantages of a gridded parametrization of the velocity function compared with a sparse parametrization.

The optimization problem in equation (10.5) is usually solved by applying gradient-based optimization methods (Claerbout, 2004). However, the reflection traveltimes are not a linear function of the slowness model because of both ray bending and the movement of the estimated reflectors. To apply gradient-based optimization algorithms, we must repeatedly linearize the problem around the current best estimate of the slowness model, ${}_i\hat{\mathbf{s}}$, and reflector geometry, ${}_i\widehat{\mathbf{I}}$, where the left subscript i indicates the iteration number. The first linearization is based on the a priori estimate ${}_0\mathbf{s}$; therefore, given the high degree of non-linearity of the modeling operator, the final solution is strongly dependent on ${}_0\mathbf{s}$.

The objective function for the linearized problem at the i-th iteration can thus be written as a function of the slowness perturbation: ${}_i\Delta\mathbf{s} = {}_i\hat{\mathbf{s}} - {}_{i-1}\hat{\mathbf{s}}$

$$\begin{aligned} J_{\text{reg}}({}_i\Delta\mathbf{s}) &= \left\| \left({}_{i-1}\widehat{\mathbf{t}_m} + {}_i\Delta\mathbf{t}_m\right) - \mathbf{t}_{\text{obs}} \right\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})' \mathbf{C_s}^{-1} ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}) \\ &= \left\| {}_i\mathbf{L}\,{}_i\Delta\mathbf{s} - \left(\mathbf{t}_{\text{obs}} - {}_{i-1}\widehat{\mathbf{t}_m}\right) \right\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})' \mathbf{C_s}^{-1} ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}), \end{aligned} \tag{10.6}$$

where ${}_{i-1}\widehat{\mathbf{t}_m}$ are the modeled traveltimes given the slowness function, ${}_{i-1}\hat{\mathbf{s}}$, and ${}_i\Delta\mathbf{t}_m$ are the perturbations in the modeled traveltimes caused by the perturbations, ${}_i\Delta\mathbf{s}$, in the slowness function. These perturbations (${}_i\Delta\mathbf{s}$ and ${}_i\Delta\mathbf{t}_m$) are linked by the operator ${}_i\mathbf{L}$, which is evaluated by linearizing the forward-modeling functional at $\mathbf{s} = {}_{i-1}\hat{\mathbf{s}}$. The next section presents the principles for evaluating the linear operator ${}_i\mathbf{L}$.

10.4.2 Evaluation of the linearized reflection-tomography operator

When evaluating ${}_i\mathbf{L}$, it is important to take into account that the modeled traveltimes, ${}_i\widehat{\mathbf{t}_m}$, are functions of the current estimate of the slowness model, ${}_i\hat{\mathbf{s}}$, both directly and indirectly, through the current estimate of the reflectors' geometry, ${}_i\widehat{\mathbf{I}}$ (van Trier, 1990; Etgen, 1990;

Stork and Clayton, 1991). The operator ${}_i\mathbf{L}$ is made of two distinct components: ${}_i\mathbf{L}_s$, which models the perturbations in reflected traveltimes, ${}_i\Delta_s\mathbf{t}_m$, at *fixed* reflectors' geometry, and the chain operator ${}_i\mathbf{L}_I = {}_i\mathbf{G}\,{}_i\mathbf{H}$, which models the perturbations in reflected traveltimes, ${}_i\Delta_I\mathbf{t}_m$, caused by perturbations in the reflectors' geometry, $\Delta\,{}_i\widehat{\mathbf{I}}$. The total traveltime perturbations, ${}_i\Delta\mathbf{t}_m$, are equal to the sum of these two terms, as follows:

$$ {}_i\Delta\mathbf{t}_m = {}_i\Delta_s\mathbf{t}_m + {}_i\Delta_I\mathbf{t}_m = {}_i\mathbf{L}_s\,{}_i\Delta\mathbf{s} + {}_i\mathbf{G} = ({}_i\mathbf{L}_s + {}_i\mathbf{G}\,{}_i\mathbf{H})\,{}_i\Delta\mathbf{s}. \tag{10.7} $$

The schematic in Figure 10.12 shows the raypaths used for evaluating the two components of ${}_i\mathbf{L}$ that correspond to one element in the traveltime vectors. The source ray (ray_s) and the receiver ray (ray_r) are used to evaluate ${}_i\mathbf{L}_s$, whereas the normal-incidence ray (ray_{n0}) is used to evaluate ${}_i\mathbf{H}$.

We apply Fermat principle to all the relevant rays to numerically evaluate ${}_i\mathbf{L}_s$ and ${}_i\mathbf{H}$. We can apply Fermat principle because in each case we assume that the end points of the rays are fixed. According to Fermat principle, the first-order perturbations in the traveltimes between two fixed points are given by the integral of the slowness perturbations along the ray connecting the two points. The application of the adjoint of ${}_i\mathbf{L}$ to traveltime residuals spreads (back-projects) these traveltime residuals along the raypaths into the slowness model. The adjoint of ${}_i\mathbf{L}$ is thus called the **back-projection operator**.

Traveltime perturbations caused by velocity perturbations

The evaluation of ${}_i\mathbf{L}_s$ is straightforward. The perturbation of the modeled traveltime, ${}_i\Delta_s t_m$, for one event is computed by integrating the slowness perturbations along both the down-going ray and the up-going ray; that is,

$$ {}_i\Delta_s t_m = \int_{\text{ray}_s} {}_i\Delta\mathbf{s}\; dl_s + \int_{\text{ray}_r} {}_i\Delta\mathbf{s}\; dl_r. \tag{10.8} $$

Traveltime perturbations caused by reflector movements

The second component of ${}_i\mathbf{L}$ is given by the chain of two operators: ${}_i\mathbf{G}$ and ${}_i\mathbf{H}$. The tomographic operator, ${}_i\mathbf{H}$, relates slowness perturbations to reflectors' shifts. The "local" ${}_i\mathbf{G}$ operator converts reflectors' shifts into traveltime perturbations along the source and receiver rays (ray_s and ray_r).

A challenge in evaluating this second component is the linearization of the relationship between the estimated reflectors' geometry and the estimated slowness model. The geometry of the reflectors is determined by identifying reflectors in a migrated image. The geometry of the imaged reflectors is related in a complex non-linear way to the slowness model. To linearize this relationship, we make the simplifying assumption that the reflectors' geometry is given by the **map migration** of the zero-offset events. Following this assumption, the perturbations in the reflectors' geometry, $\Delta\,{}_i\widehat{\mathbf{I}}$, caused by perturbations in the slowness model can be considered equal to the residual zero-offset map migration of the reflector. Residual zero-offset

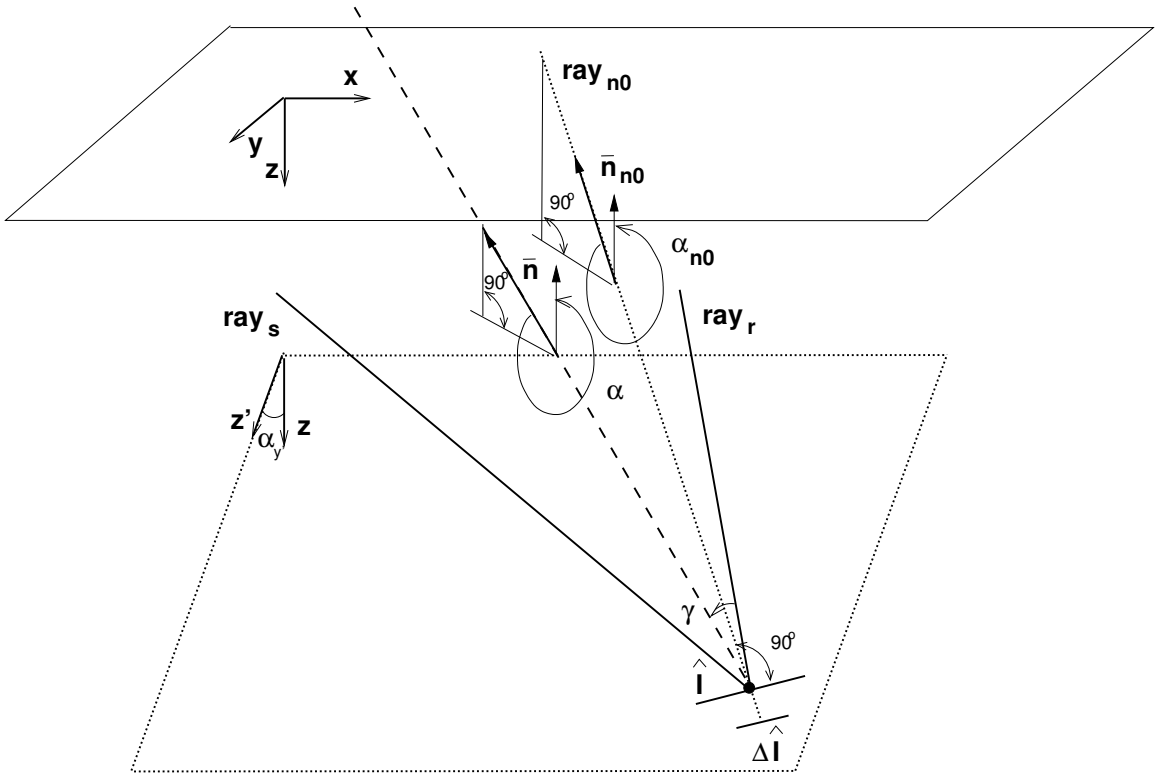


Figure 10.12: Raypaths used for the evaluation of the two components ($_i\mathbf{L}_s$ and $_i\mathbf{L}_I$) of the linearized tomographic operator, $_i\mathbf{L}$. The source and receiver rays (ray$_s$ and ray$_r$) are used for $_i\mathbf{L}_s$. The normal-incidence ray (ray$_{n0}$) is used for $_i\mathbf{L}_I$. Notice that in general the direction orthogonal to the reflector interpreted from the migrated image ($\mathbf{n}_{n0}$) does not coincide with the bisector of the source and receiver rays (**n**). vel-tomo-rays-3d [NR]

migration moves the reflectors along the zero-offset ray; that is, the ray normal to the reflector. The magnitude of this movement is given by the two-way traveltime perturbations along the normal-incidence raypath ($_i\Delta_s t_m^0$) scaled by the slowness at the reflection point ($_{i-1}\hat{s}_I$); that is by the following integral expression:

$$\Delta_i \widehat{I} = -\frac{_i\Delta_s t_m^0}{2\ _{i-1}\hat{s}_I} = -\frac{1}{_{i-1}\hat{s}_I}\int_{\text{ray}_{n0}} {}_i\Delta\mathbf{s}\ dl_{n0}. \tag{10.9}$$

The projection of the reflector perturbation on the vertical axis ($\Delta_i \widehat{I}_z$) is derived directly from equation (10.9) as follows:

$$\Delta_i \widehat{I}_z = -\frac{1}{p_{z_{n0}}}\int_{\text{ray}_{n0}} {}_i\Delta\mathbf{s}\ dl_{n0}, \tag{10.10}$$

where $p_{z_{n0}}$ is the vertical ray parameter of the normal-incidence ray at the reflection point. The adjoint of $_i\mathbf{H}$ is a back-projection operator because it spreads slowness perturbations along raypaths.

In contrast, the relation between a reflectors' geometry perturbations and perturbations in traveltime depends only on the rays' geometry and the slowness at the reflection point as follows:

$$ {}_i\Delta_I t_m = \left(p_{z_s} + p_{z_g}\right) \Delta_i \widehat{I}_z, \tag{10.11} $$

where p_{z_s} and p_{z_g} are the vertical ray parameter of the down-going ray and up-going ray, respectively, at the reflection point.

Substituting equation (10.9) into equation (10.11) yields the following expression for the traveltime perturbations:

$$ {}_i\Delta_I t_m = -\frac{p_{z_s} + p_{z_g}}{p_{z_{n0}}} \int_{\text{ray}_{n0}} {}_i\Delta\mathbf{s}\, dl_{n0}, \tag{10.12} $$

which can be expressed in terms of the aperture angle γ and the dip angles α and α_{n0} (see Figure 10.12), by using the relationships between ray parameters and and angles at the reflection point (Chapter 6) as follows:

$$ {}_i\Delta_I t_m = -2\cos\gamma \frac{\cos\alpha}{\cos\alpha_{n0}} \int_{\text{ray}_{n0}} {}_i\Delta\mathbf{s}\, dl_{n0}. \tag{10.13} $$

When the velocity is correct, the dip angle of the direction orthogonal to the imaged reflector (α_{n0}) is equal to the dip angle of the bisector of the angle formed by the source and receiver rays at the reflection point (α). When the velocity is not correct, these angles may differ. In keeping with the assumption that the reflectors' geometry is given by the map migration of the zero-offset events, we usually assume that α is equal to α_{n0}. With this assumption, equation (10.13) simplifies as follows:

$$ {}_i\Delta_I t_m = -2\cos\gamma \int_{\text{ray}_{n0}} {}_i\Delta\mathbf{s}\, dl_{n0}. \tag{10.14} $$

Notice that the two components of the back-projection operator have opposite signs; a positive perturbation in slowness contributes to both an increase in traveltimes through equation (10.8), and a decrease in traveltimes through equation (10.13). The total traveltime perturbations (${}_i\Delta t_m = {}_i\Delta_s t_m + {}_i\Delta_I t_m$) usually decrease as the aperture angle decreases, and are exactly zero for normal-incidence events. For these events, the source and receiver rays coincide with the normal-incidence ray and thus the sum of the integrals in equation (10.8) is exactly twice the integral in equation (10.13), and $\gamma = 0$ and $\alpha = \alpha_{n0}$ in equation (10.13).

The insensitivity to velocity perturbations of the measurable perturbations in the traveltimes for normal-incidence reflections is directly related to the fact that the positions of the reflectors are unknown. Therefore, from reflection data there is no velocity information in absolute traveltimes, but only in the the relative delays between events reflected from the same reflector at different angles. This limitation causes the well-known **depth-velocity ambiguity**, which is another way to say that the mapping velocity component is poorly determined from reflection data. The next section analyzes this problem by examining the characteristics of a simple tomographic operator that exemplifies many of the issues encountered when we attempt to invert more realistic and complex operators.

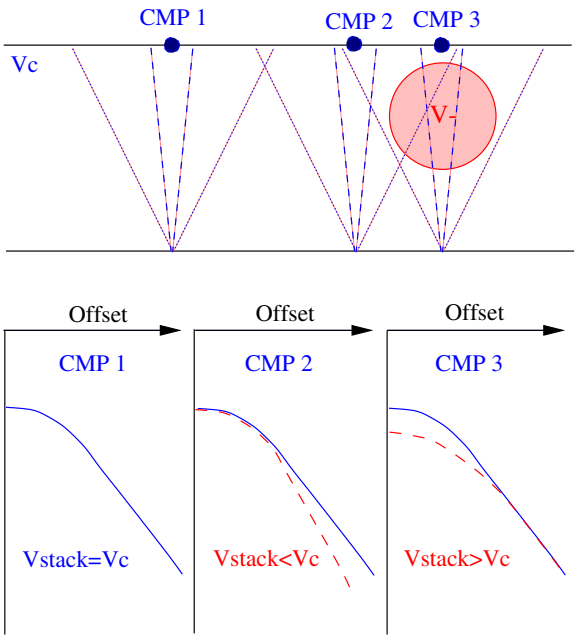


Figure 10.13: Effects of a velocity anomaly on stacking velocity. Three CMP gathers are considered: far from the anomaly (CMP1), on the side of the anomaly (CMP2), and right above the anomaly (CMP3). vel-stackvel [NR]

10.4.3 Tomographic inversion of stacking velocities

The Dix formula is not accurate when substantial lateral velocity variations occur within the area "illuminated" by the individual CMP gathers. However, even in these situations, the picking of stacking-velocity spectra computed from CMP gathers is a robust and efficient procedure for measuring the kinematics in the data. The kinematics information contained in the stacking velocities can be extracted by inverting a tomographic operator that relates interval-velocity perturbations to stacking-velocity perturbations. This section presents a simple example of such a tomographic operator that is often referred as the **Toldi operator** (1985; 1989).

The Toldi operator is derived assuming flat reflectors and constant background velocity; because these assumptions are rarely met in practice, its applicability to real data is limited in scope. But, notwithstanding its simplicity, it exemplifies well the effects that lateral velocity variations have on the kinematics that are measurable from the data. In particular, the analysis of the Toldi operator clearly illustrates the dangers of disregarding the presence of lateral velocity variations while inverting the picks from velocity spectra – stacking velocity in this case, residual-moveout parameters in Section 11.3. Furthermore, since it is derived assuming a constant velocity background, it is stationary with respect to the horizontal coordinates and can be analyzed in the horizontal-wavenumber domain. In this domain, useful insights can be gained on the resolution of reflection tomography, and how multi-azimuth acquisition and processing can potentially improve the resolution of velocity estimation.

The Toldi operator quantitatively models the relationship between stacking and interval velocities that is qualitatively illustrated by the schematics shown in Figure 10.13. The top part of the Figure 10.13 shows the schematics of the raypaths for reflections from a flat reflector recorded at three CMP gathers. The reflections recorded at CMP 1 are not affected at all

by the negative velocity anomaly, and therefore the moveout is perfectly hyperbolic, and the stacking velocity measured at CMP 1 is equal to the constant background velocity, V_c, in the medium (bottom part of the Figure). In contrast, the reflections recorded at CMP 2 and CMP 3 are affected by the velocity anomaly, though in different ways. At CMP 2, the far-offset reflections (dotted raypaths in the Figure) are delayed by the anomaly, while at CMP 3 the near-offset reflections (dashed raypaths in the Figure) are delayed by the anomaly. In both cases, the resulting moveouts are not hyperbolic, but they can be well approximated with hyperbolas, if the anomaly is not too strong. However, the perturbation in the stacking velocity of the hyperbola at CMP 2 has the opposite sign from the perturbation in the stacking velocity of the hyperbola at CMP 3. The moveout at CMP 2 fits a hyperbola with lower velocity than V_c, because the time dip of the asymptote is increased. In contrast, the moveout at CMP 3 fits a hyperbola with higher velocity than V_c, because the top of the hyperbola is flattened, while the time dip of the asymptote is unchanged.

The phenomenon illustrated in Figure 10.13 may cause large errors in a 1-D vertical inversion of stacking velocities. If we were to update the velocity function at the CMP 3 location based on the measured stacking velocity, we would increase the velocity above the reflector, when just the opposite is needed. To update the velocity function in the right direction, a joint inversion of the measures of moveouts at several midpoints is needed. The Toldi method is based on the inversion of a linearized relationship between the perturbations in the interval slowness function and the corresponding perturbations in the measured stacking slowness. This linear relationship quantitatively describes the effects illustrated in Figure 10.13, and it is expressed as the following convolution:

$$\Delta s_{\text{st}}\left(x_m, y_m, \theta_h, z_\xi\right) = \int_z \int_x \int_y G\left(x_m, y_m, \theta_h, z_\xi, z, x, y\right) \Delta s\left(z, x, y\right) dz\, dy\, dx, \tag{10.15}$$

where G is the kernel of the linear operator relating stacking-slowness perturbations, Δs_{st}, to interval-slowness perturbations, Δs. In matrix notation, equation (10.15) can be more compactly expressed as follows:

$$\Delta \mathbf{s}_{\text{st}} = \mathbf{L}_{\text{st}} \Delta \mathbf{s}. \tag{10.16}$$

To numerically evaluate $\mathbf{L}_{\text{st}}$, we apply the principles of reflection tomography presented in Section 10.4.2. The kernel in equation (10.15) describes two chained relationships: 1) the link between the perturbations in the curvature of the hyperbola that best fits, in the least squares sense, the moveout of the perturbed reflections, and 2) the relation between the traveltime perturbations and the interval-velocity perturbations. This second relationship is evaluated using the expression in equation (10.7).

In the case of a flat-layered Earth and assuming no ray bending, the evaluation of $\mathbf{L}_{\text{st}}$ is analytical. Toldi (1989) developed and applied the relevant theory in 2-D, and Clapp and Biondi (1995) extended it to 3-D. The 3-D version of G is a simple extension of the 2-D operator evaluated along the vertical plane connecting the source and receiver. The expression for G in 3-D is the following:

$$G\left(x_m, y_m, \theta_h, z_\xi, z, x, y\right) = \frac{15 z_\xi}{L^2 L'} \left\{ 3 \left(\frac{2\sqrt{(x-x_m)^2 + (y-y_m)^2}}{L'} \right)^2 - 1 \right\}$$

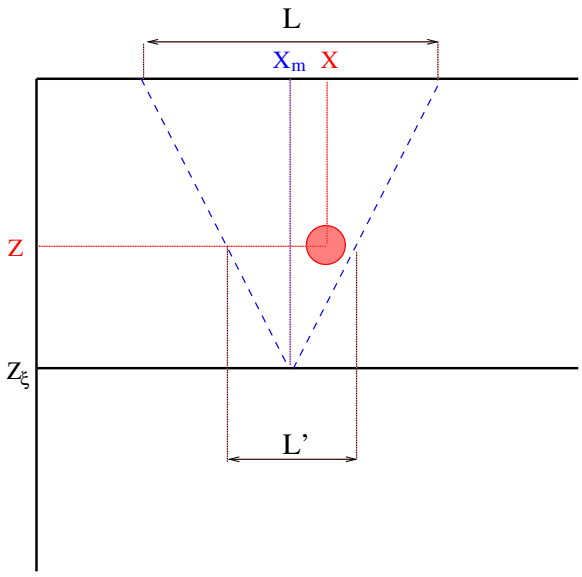


Figure 10.14: Schematic illustrating the concept of effective cable length. L is the length of the seismic cable at the surface. L' is the effective cable length for a velocity anomaly at depth z, "illuminated" by a reflector at depth z_ξ. vel-toldi-oper [NR]

$$\left\{1+\frac{L^2}{4z_\xi^2}\left(\frac{2\sqrt{(x-x_m)^2+(y-y_m)^2}}{L'}\right)^2\right\}$$
$$\text{for } \sqrt{(x-x_m)^2+(y-y_m)^2} < \frac{L'}{2}$$
$$\text{and } \tan^{-1}\frac{(x-x_m)}{(y-y_m)} = \theta_h$$
$$= \quad 0 \qquad \text{elsewhere,} \tag{10.17}$$

where L represents the recording cable length and L' represents the **effective cable length** for a reflector located at depth z_ξ and a velocity perturbation located at depth z. The effective cable length L' is defined as follows:

$$L' = \left[\frac{z_\xi - z}{z_\xi}\right] L; \tag{10.18}$$

its physical meaning is illustrated in Figure 10.14.

The following example illustrates the properties of this operator when applied to predict stacking-velocity perturbations and when inverted for estimating interval-velocity perturbations from measured stacking velocities. Figure 10.15 shows an in-line section of the velocity function used for this example. A Gaussian-shaped positive anomaly is superimposed onto a constant velocity background of 3 km/s. Figure 10.16 shows the corresponding stacking velocities for a flat reflector at a depth of 2.3 kilometers, when the data have been recorded along only one azimuth, θ_h, aligned with the in-line direction and with a maximum offset of 1.5 kilometers. The solid line in Figure 10.16 shows the stacking-velocity function as predicted by the Dix formula. As expected, the Dix formula predicts a positive anomaly peaking right below the interval-velocity anomaly. The dashed line in the Figure shows the stacking-velocity function as predicted by the Toldi operator expressed in equation (10.17). Consistent with the

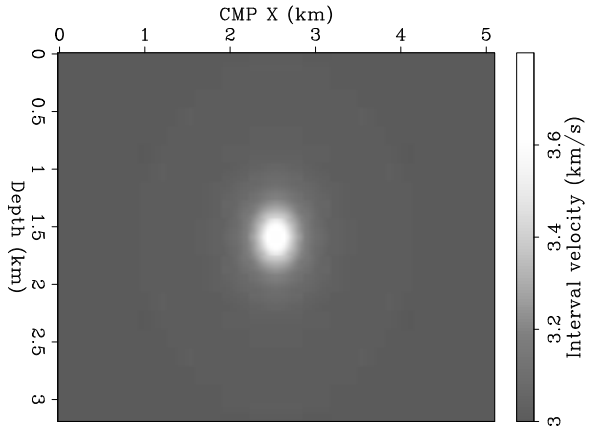


Figure 10.15: In-line section of the interval-velocity function cut through the peak of the Gaussian-shaped anomaly. vel-new-model [ER]

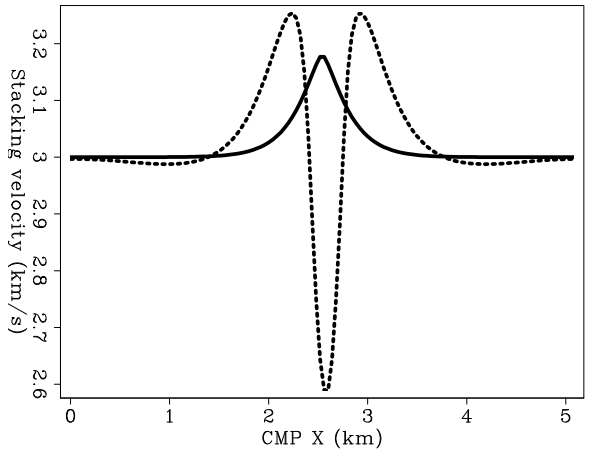


Figure 10.16: Predicted stacking velocities: Dix formula (solid). Toldi's operator (dashed), vel-all.stacking-gr [ER,M]

analysis illustrated in Figure 10.13, the stacking velocity has a large negative perturbation right below the interval-velocity anomaly, and two positive side-lobes on the sides of the anomaly.

Figure 10.17 shows an in-line section of the stacking-velocity function predicted by the Dix formula for the reflector depth z_ξ varying continuously from the surface to 3.2 kilometers. This vertical section was cut through the the peak of the anomaly. Above the anomaly the stacking velocity is constant; it reaches a peak just below the anomaly, and then slowly declines. Figure 10.18 shows the corresponding in-line section of the stacking-velocity function predicted by the Toldi operator for the same varying reflector depth, z_ξ. Notice that the interval-velocity anomaly's zone of influence grows in the horizontal direction with the depth of the reflector, because the effective cable length L' [equation (10.18)] increases with increasing z_ξ.

Figure 10.19 shows an in-line section of the results of using the Dix formula to invert the stacking velocities that were predicted using the Toldi operator (Figure 10.18). As expected, the estimated velocity function is far from the true one (Figure 10.15). In contrast, the velocity anomaly estimated by inverting the Toldi operator (Figure 10.20) has a shape similar to the true anomaly. However, the estimated anomaly is stretched in the vertical direction and the peak amplitude is only about 17% of the peak amplitude of the true anomaly. Unfortunately, these discrepancies between the true model and the estimated model are not particular to the inversion of stacking velocities using the Toldi operator, but rather they are typical of tomographic velocity-estimation methods. They are a consequence of the underconstrained nature of the problem.

10.4.4 Multi-azimuth velocity estimation

The estimation results shown in the previous section are negatively affected by the non-uniqueness of estimating velocity from reflection seismic data. However, the data used in the previous section did not fully exploit the 3-D nature of the problem, because all the offset azimuths were oriented along only one direction. In this section, we briefly discuss how a multi-azimuth acquisition can help to improve the results of velocity estimation, as shown by Clapp and Biondi (1995) for the Toldi operator, and then by Kosloff and Sudman (2002) for a more general reflection tomography operator. The considerations that follow are not applicable to marine data acquired with a standard surface-streamer geometry, but they are relevant to data acquired with a wide-azimuth acquisition geometry, such OBC and land data (Chapter 1). However, it is still standard practice to ignore the azimuth when performing velocity estimation, even for data acquired with wide-azimuth acquisition geometry. The following analysis shows that this simplification might unnecessarily reduce the resolution of velocity estimation.

We first analyze the properties of the Toldi operator as a function of the azimuthal coverage in the data, and then we discuss some inversion results. To quantify the degree of non-uniqueness in the inversion of a linear operator, such as the one represented by equation (10.17), we introduce the concept of *effective rank* . The concept of effective rank of a matrix is linked to the concept of *condition number* (Golub and Loan, 1983). The condition

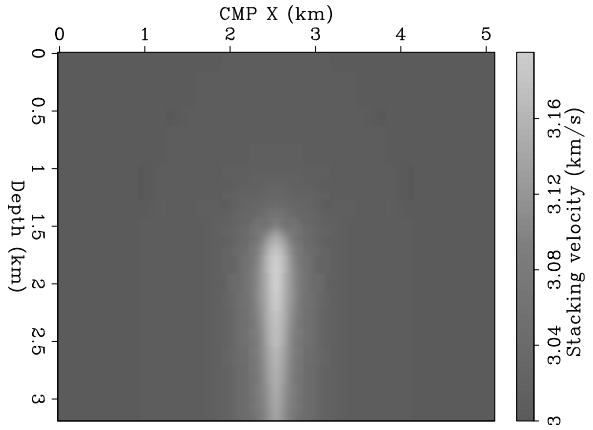


Figure 10.17: In-line section of the stacking-velocity function predicted by the Dix formula. vel-dix.stacking [ER,M]

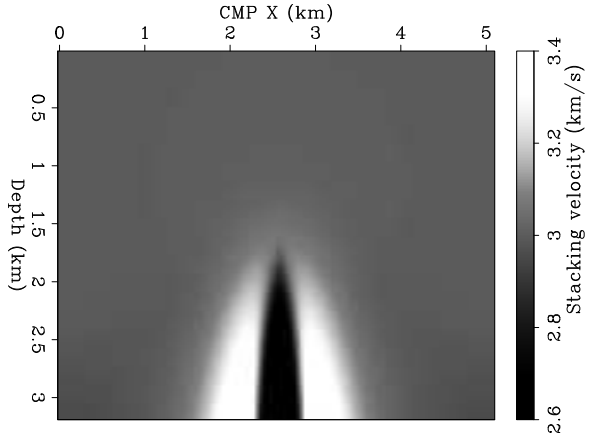


Figure 10.18: In-line section of the stacking-velocity function predicted by the Toldi operator. vel-toldi.stacking [ER,M]

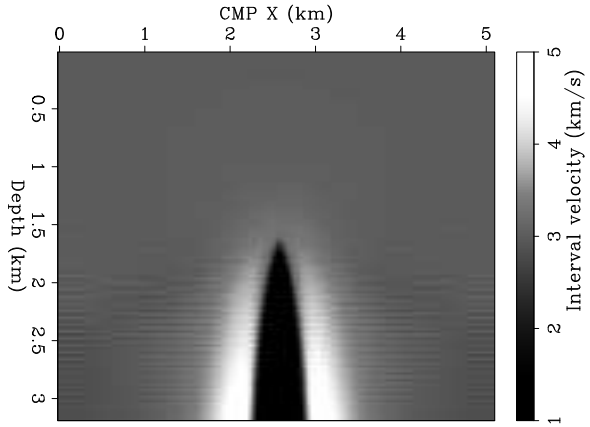


Figure 10.19: In-line section of the interval-velocity function estimated by inverting the Dix formula. Compare with the true model shown in Figure 10.15. vel-dix-inv [ER,M]

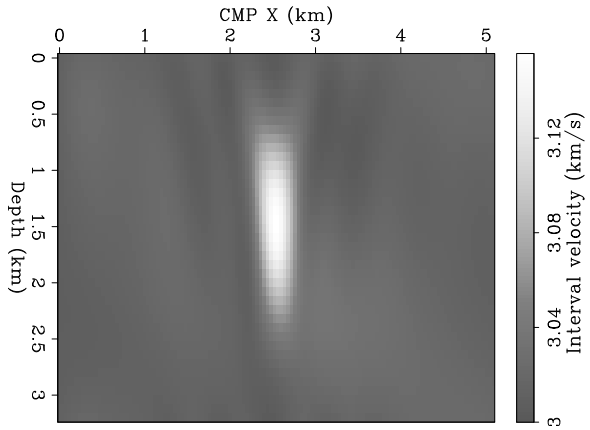


Figure 10.20: In-line section of the interval-velocity function estimated by inverting the Toldi operator. Compare with the true model shown in Figure 10.15. vel-toldi-inv [ER,M]

number is defined as the ratio between the largest and the smallest singular values. When inverting real data, a matrix with too large of a condition number is effectively rank deficient; an upper limit must be set on the condition number. The effective rank of a matrix, then is equal to the order for which the ratio between the largest singular value and the singular value of that order is less than the set upper limit on the condition number. For the examples shown below, we have set the upper limit on the condition number equal to 10.

Because the Toldi operator is convolutional and spatially invariant with respect to the horizontal axes (x, y) of the model, it can be conveniently analyzed in the Fourier domain as a function of the horizontal wavenumbers k_x and k_y. Therefore, we applied a Singular Value Decomposition (SVD) (Golub and Loan, 1983) to the wavenumber-domain representation of the operator in equation (10.17), and estimated the rank of the operator for each value of (k_x, k_y). Figure 10.21 shows the rank of the operator, as a function of the wavenumbers, when the data were acquired with only one azimuth. Because we assume the single-azimuth acquisition to be oriented along the in-line direction, the rank is independent of the cross-line wavenumber. Low rank indicates a poorly-determined component of the solution. In particular, the components of the solution with in-line wavenumber around 1 km^{-1} appear to be poorly determined. Figure 10.22 shows the rank of the operator when the data are acquired along three different azimuths ($0°, 60°,$ and$120°$). The rank is uniformly higher than in Figure 10.21, indicating a better-constrained inversion problem.

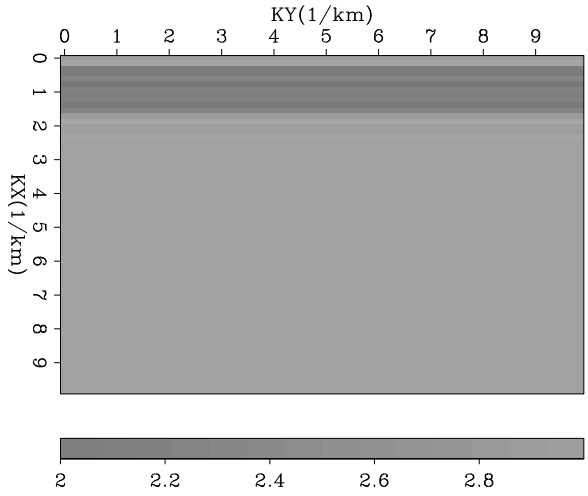


Figure 10.21: Rank of $\mathbf{L}_{st}$ when the data are collected along one azimuth (0°). vel-Rank-1 [ER,M]

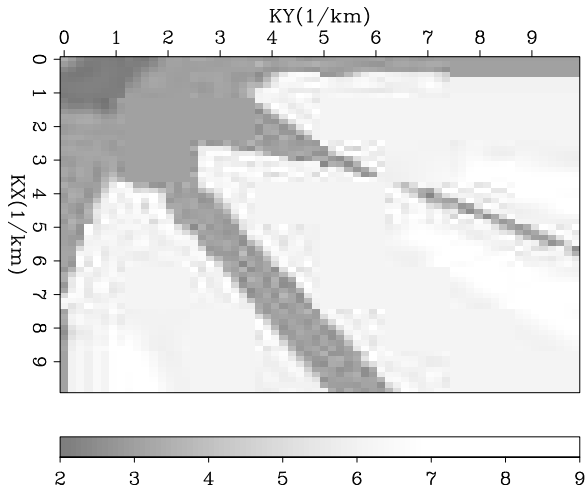


Figure 10.22: Rank of $\mathbf{L}_{st}$ when the data are collected along three azimuths: $0°, 60°,$ and$120°$. vel-3dview2 [ER,M]

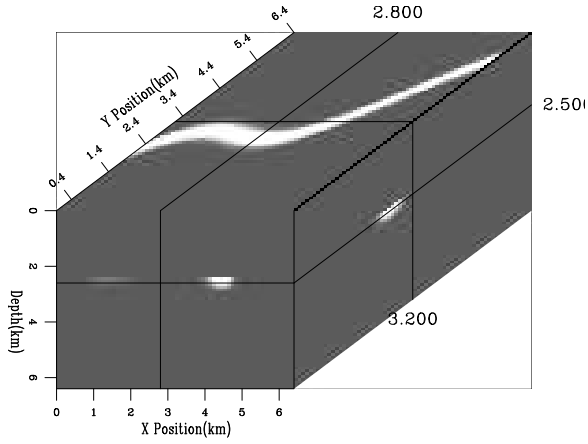


Figure 10.23: True velocity model. The channel is parabolic in shape, with amplitude decaying exponentially as a function of the distance from the center of the channel. vel-river-overn [ER,M]

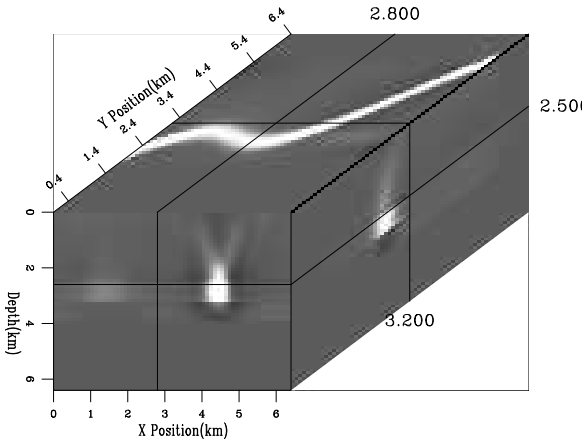


Figure 10.24: Estimated velocity model when the data are collected along one azimuth. Notice the poor definition of the segment of the channel approximately in the middle of the model. vel-chanel1-overn [ER,M]

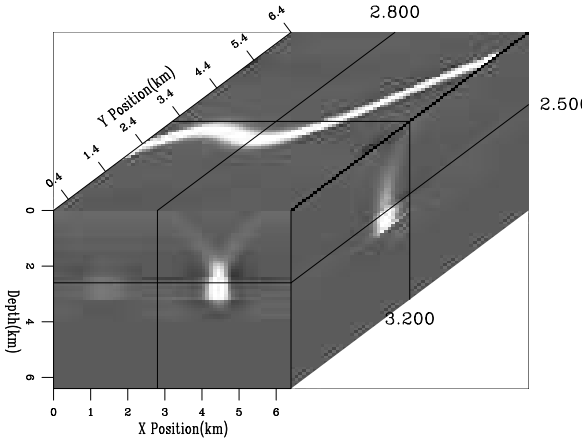


Figure 10.25: Estimated velocity model when the data are collected along three azimuths: $0°, 60°,$ and $120°$. Notice that the segment of the channel approximately in the middle of the model is better defined than in Figure 10.24. vel-chanel3-overn [ER,M]

The analysis of the operator rank is confirmed by the results of inverting for the negative velocity perturbations caused by a meandering channel in the subsurface. Figure 10.23 shows the true velocity model. Figure 10.24 shows the estimated velocity model when the data are collected along one azimuth. Notice the poor definition of the segment of the channel approximately in the middle of the model. This part of the velocity model corresponds to smooth variations along the in-line direction ($k_x \approx 1$ km^{-1} in Figure 10.21) that are not well constrained by the data. Figure 10.25 shows the estimated velocity model when the data are collected along three azimuths: $0°, 60°,$ and $120°$. Notice that the segment of the channel that was poorly resolved in Figure 10.24 is better defined in Figure 10.25.

REFERENCES

Billette, F., Lambare, G., and Podvin, P., 1997, Velocity macromodel estimation by stereotomography: 67th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 1889–1892.

Biondi, B., 1992, Velocity estimation by beam stack: Geophysics, **57**, 1034–1047.

Bishop, T. N., Bube, K. P., Cutler, R. T., Langan, R. T., Love, P. L., Resnick, J. R., Shuey, R. T., Spindler, D. A., and Wyld, H. W., 1985, Tomographic determination of velocity and depth in laterally varying media: Geophysics, **50**, 903–923.

Claerbout, J. F., 2004, Image Estimation by Example: `http://sepwww.stanford.edu/sep/prof/index.html`.

Clapp, R. G., and Biondi, B. L., 1995, Multi-azimuth velocity estimation: 65th Ann. Internat. Meeting, Soc. Expl. Geophys., Expanded Abstracts, 1014–1017.

Clapp, R. G., Sava, P., and Claerbout, J. F., 1998, Interval velocity estimation with a null-space: SEP–**97**, 147–156.

Dix, C. H., 1955, Seismic velocities from surface measurements: Geophysics, **20**, 68–86.

Duveneck, E., and Hubral, P., 2002, Tomographic velocity model inversion using kinematic wavefield attributes: 72nd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 862–865.

Etgen, J., 1990, Residual prestack migration and interval velocity estimation: Ph.D. thesis, Stanford University.

Golub, G. H., and Loan, C. F. V., 1983, Matrix computations: John Hopkins University Press.

Gurevich, B., and Landa, E., 2002, Multifocusing imaging with controlled reflection-point dispersal: Geophysics, **67**, 1586–1592.

Kosloff, D., and Sudman, Y., 2002, Uncertainty in determining interval velocities from surface reflection seismic data: Geophysics, **67**, 952–963.

Perroud, H., Hubral, P., and Hocht, G., 1999, Common-reflection-point stacking in laterally inhomogeneous media: Geophys. Prosp., **47**, 01–24.

Stork, C., and Clayton, R. W., 1991, Linear aspects of tomographic velocity analysis: Geophysics, **56**, 483–495.

Sword, C. H., 1987, Tomographic determination of interval velocities from reflection seismic data: The method of controlled directional reception: Ph.D. thesis, Stanford University.

Taner, M. T., and Koehler, F., 1969, Velocity spectra - digital computer derivation and applications of velocity functions: Geophysics, **34**, 859–881.

Toldi, J., 1985, Velocity analysis without picking: Ph.D. thesis, Stanford University.

Toldi, J. L., 1989, Velocity analysis without picking: Geophysics, **54**, 191–199.

Tsvankin, I., 2001, Seismic signatures and analysis of reflection data in anisotropic media: Elsevier Science.

van Trier, J., 1990, Tomographic determination of structural velocities from depth migrated seismic data: Ph.D. thesis, Stanford University.

Chapter 11

Migration Velocity Analysis

In the presence of complex structures and/or strong lateral velocity variations, extracting velocity information in the data space is both inaccurate and time consuming. In these situations, the image space is a more appropriate domain for extracting kinematic information, since migration focuses and greatly simplifies the events. Even when the migration velocity is far from the true velocity, the incomplete focusing of the reflections is a step in the right direction. Velocity-estimation methods that use the focusing capabilities of migration to extract kinematic information more reliably, are commonly known as **Migration Velocity Analysis** (**MVA**) methods.

MVA is an iterative process where each iteration comprises two distinct steps: 1) the data are imaged by prestack migration, and 2) the velocity function is updated based on the migration results. The global convergence of this process is a first-order concern. The relationship between the velocity model and the focusing-quality of the image is highly non-linear, and consequently, the objective function optimized during MVA is non-convex and has several local minima. The quality of the starting velocity model is crucial to assure global convergence. The starting model is usually estimated using one of the data-domain velocity-estimation methods that I presented in Chapter 10. Therefore, the development of MVA methods has not made the data-domain methods obsolete. On the contrary, data-domain velocity estimation methods are a needed preamble to MVA.

The first challenge of MVA is the extraction of kinematic information from the migrated image cube. In the data domain, the accuracy of the velocity function is determined by measuring the coherency of the data in CMP gathers along the offset axes, after the application of NMO. In the image domain, the accuracy of the velocity function is determined by measuring the focusing of the reflections in the migrated image. The coherence of the image in CIGs after migration is the most common criterion used to judge the focusing. When Offset-Domain CIGs (ODCIGs) are generated (such as the ones generated by a Kirchhoff common-offset prestack migration), coherency is measured along the offset axes. When Angle-Domain CIGs are generated, (such as the ones generated by a wavefield-continuation migration), coherency is measured along the aperture angle (γ) and reflection azimuth (ϕ) axes. The departure from flatness of the events in the CIGs is called **residual moveout**, or **RMO**.

As in the data-domain case, in the migrated domain it is often more robust to measure the residual moveout as a function of only one parameter, or a few parameters, instead of measuring the depth errors for each offset or angle. The choice of parametrization of the residual moveout is important for obtaining accurate measurements. In Section 11.2.1 I discuss possible choices for the RMO function, concentrating on the derivation of a functional expression for the RMO function in ADCIGs.

Extracting kinematic information in the migrated domain instead of the in data domain has another advantage: in the migrated domain we can evaluate both the focusing of the reflections in the physical space (midpoint and depth), and the residual moveout along the offset (or angle) axes. The additional information provided by the focusing can be useful in the presence of numerous strongly diffracted events, such as those created by a rugose top-salt interface, or when the recorded range of reflection-aperture angles is too narrow to provide sufficient velocity resolution. The "residual focusing" required to optimally image the data can be measured by applying **residual prestack migration** and creating velocity spectra analogous to the RMO spectra. Section 11.2.2 presents the basic concepts of residual prestack migration and examples of its application.

When the goal of MVA is simply to estimate an accurate RMS velocity function for prestack time migration (Section 11.1), the measurements of RMO parameters are directly applied to update the V_{rms} function. When the goal of MVA is more ambitious, and an interval-velocity function is sought to perform depth migration, the kinematics errors measured from the migrated gathers must be converted into interval-velocity updates. There are several methods to accomplish this conversion. The simplest ones are based on a generalization to the migrated domain of the Dix formula presented in Section 10.1. The RMO functions picked from the migrated gathers are interpreted as errors in the RMS velocity function. These errors are then inverted into interval-velocity updates along the vertical direction, using an updating procedure based on the Dix formula. Section 11.3 presents the basic concepts underlying vertical-updating methods, and discusses the limitations of this approach.

In complex situations, tomography is often needed for inverting the measured depth errors into interval velocity updates. The basic principles presented in Section 10.4 are directly applicable to **tomographic migration velocity analysis**, after the depth errors are converted into time errors. Section 11.4 presents the theory behind these methods. Tomographic MVA is used in the most challenging imaging projects, where the simpler methods fail. However, even tomographic methods may be challenged by these complex problems, because of the intrinsic limitations of tomographic velocity estimation: non-linearity of the problem and non-uniqueness of the solution. Several methods have been proposed to address these limitations and ensure that the estimation process converges to a velocity function that is consistent with our geological models of the subsurface. I review some of these advanced methods at the end of the chapter.

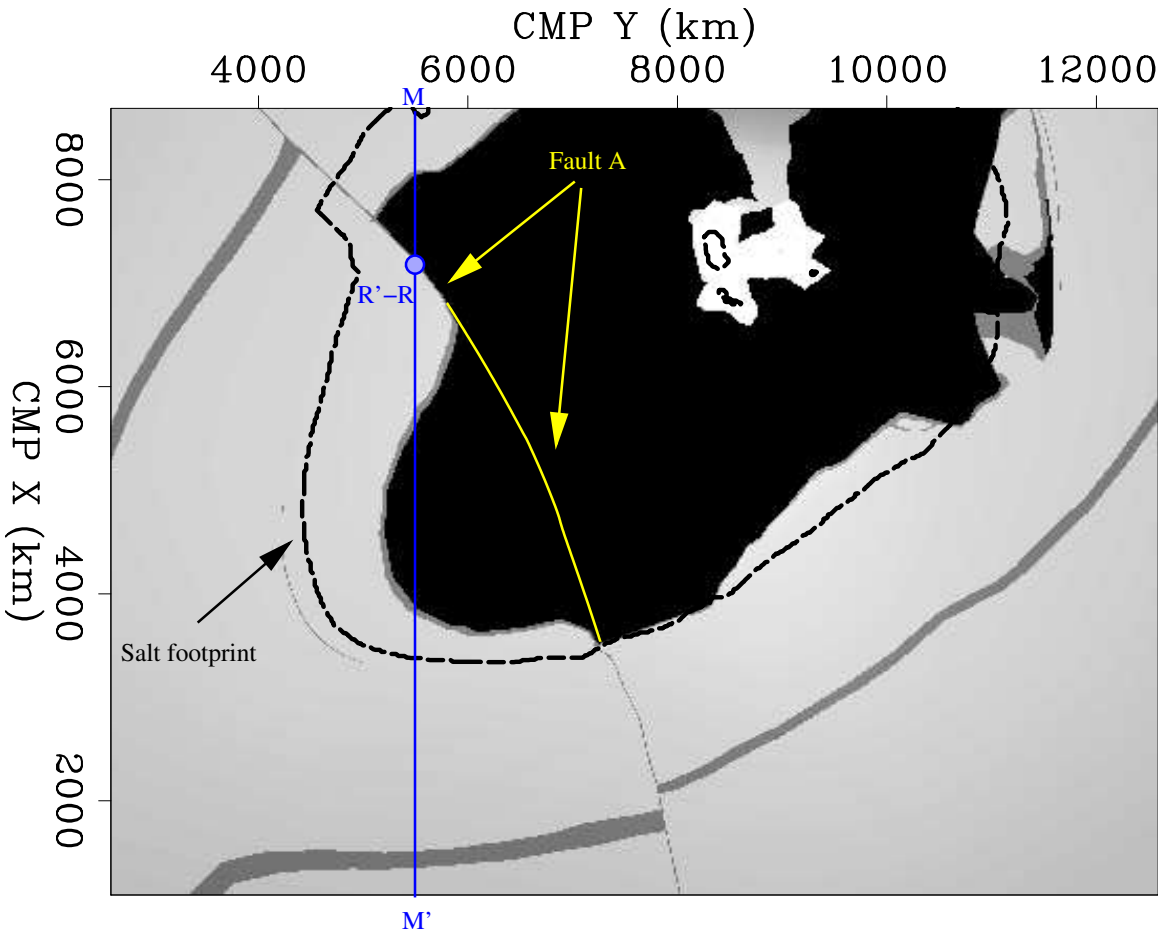


Figure 11.1: Depth slice of the velocity model for the SEG-EAGE salt data set. The vertical line (M-M') superimposed onto the velocity function indicates the location of migrated section shown in the following figures. migvel-Depth-vel-mig [NR]

11.1 Time-migration velocity analysis

Sections 10.2 and 10.3 present examples where the zero-offset events are shifted vertically and horizontally with respect to the reflectors' true positions, causing ambiguities in the interpretation of the stacking velocity analysis. These ambiguities cause severe problems not only when we try to invert stacking velocities into interval velocities, but also when we attempt the simpler task of defining a V_{rms} function that can be used for a prestack time migration. We can successfully unravel the ambiguities in the data domain by performing the velocity analysis in the migrated domain. Gardner et al. (1974) and Sattlegger (1975) first recognized that many of the problems of conventional velocity analysis in CMP gathers can be addressed by comparing the images obtained by migration of different data offsets. The following example illustrates this concept by analyzing the migration results of the SEG-EAGE salt data set along one line.

The location of the migrated line analyzed in the next set of figures is indicated in Fig-

ure 11.1 by the M-M' vertical line superimposed onto the the depth slice ($z = 1.5$ kilometers) of the velocity function. We concentrate our attention on the reflections from the salt-sediment interface along the fault plane indicated by Fault A in Figure 11.1.

Figure 11.2 shows the migrated section in a window taken around the image of the fault-plane interface. This section is the result of migrating the data within an offset range 240 meters wide and centered at 500 meters. Figure 11.3a shows the ODCIG at CMP X=7.02 kilometers. As with gathers after NMO, if the velocity function used to image the data is correct, reflections should be flat along the offset axis of the CIGs. In the CIG shown in Figure 11.3a, the fault-plane reflections have been correctly flattened by migration, as have the reflections from the interface directly above it. In contrast, the migration velocity is too slow for the salt-bottom reflections, and thus their moveout curves points upward.

We can us the ODCIGs to improve the migration-velocity function by performing a move-out analysis. CIGs have an advantage over CMP gathers after DMO, in that the events have been moved, at least approximately, to their proper location in the subsurface. Therefore, the events in a CIGs are less likely to have traveled through different regions in the subsurface, causing inconsistency in the moveout analysis. A straightforward procedure to update the velocity function using the CIGs is the following. First, an inverse NMO correction is applied to the gathers, according to the RMS velocity used to migrate the data. Figure 11.3b shows the results of inverse NMO applied to the CIGs. Second, conventional velocity spectra are computed for the CIGs after inverse NMO. Figure 11.3c shows the velocity spectra computed from the gather in Figure 11.3b. Third, the stacking velocities measured from the velocity spectra are used to generate a new RMS-velocity function. The new RMS-velocity function is used to migrate the data again, and, if needed, the process can be repeated to further improve the velocity.

The procedure outlined above is better illustrated when applied to CIGs obtained with an inaccurate velocity. Figure 11.4 shows the migration of the same data as in Figure 11.2, but when the velocity function has been uniformly scaled by a factor of .92. The data have been incompletely focused and show clear signs of **undermigration**. In particular, residual diffraction hyperbolas are visible below the point where the fault-plane terminates against the salt body (CMP X=6.6 kilometers). Also notice that the shape of the fault-plane is distorted in the undermigrated image.

Figure 11.5a shows the ODCIG at CMP X=7.02 kilometers, corresponding the migrated section shown Figure 11.4. Figure 11.5b shows the results of inverse NMO applied to the CIG, and Figure 11.5c shows the velocity spectra computed from the CIG shown in Figure 11.5b. Comparing the velocity spectra in Figure 11.5c with the spectra in Figure 11.3c, we notice that the values of the V_{rms} at the semblance peaks corresponding to the two shallower reflections are approximately the same. As expected, the zero-offset time of the dipping event (fault reflection) is slightly lower in the CIG obtained using the lower migration velocity. However, the velocity spectra computed from the CIGs obtained with the inaccurate velocity would yield a fairly accurate V_{rms} function.

Similar results can be obtained when the migration velocity is higher than the true velocity, that is, in presence of **overmigration**. But in this case, the mispositioning of dipping reflectors

is even more apparent. Figure 11.6 shows the migration results when the correct velocity function has been scaled by a factor of 1.08. In this case, the imaged dipping reflector is too deep and too steep. Typical overmigration smiles are also visible in the section. These smiles are caused by the overmigration of the diffractions generated by the step-like discontinuity in the top of the salt. Now the moveouts in the CIG (Figure 11.7) are undercorrected and point downward. Notice the appearance of a spurious event in the CIG at around 1.4 s, which could be erroneously interpreted as a real event. This event is the result of the overmigration smiles visible in the in-line section (Figure 11.6).

In this section, I have used the cascade of inverse and forward NMO to measure the RMO in the CIGs after migration. This methodology can be applied to the results of time migration because they are defined in the time domain. However, for analyzing depth-migrated images, we need to derive new expressions for the RMO function, expressing the RMO curves as functions of depth instead of time. This task is accomplished in the next section.

11.2 Extracting velocity information from prestack images

The MVA methods I present later in this chapter update the interval-velocity function using the velocity information contained in prestack partial images. The vast majority of MVA methods are based on measurements of residual moveouts in CIGs (Section 11.2.1). However, prestack residual migration can be used to analyze the focusing of the reflections in physical space (midpoint and depth), in addition to the residual moveout along the offset (or angle) axes. Indeed, prestack residual migration is a generalization of RMO, and between these two end members there is a continuum of possible approximations that are potentially useful. Section 11.2.2 presents prestack residual migration and its relation to residual moveout.

11.2.1 Residual Moveout (RMO) Analysis

Residual moveout analysis of migrated CIGs is analogous to the stacking velocity analysis of CMP gathers in the data domain. Methods have been developed to perform this analysis on both offset-domain CIGs and angle-domain CIGs (see Chapter 6). Since ODCIGs are easily computed by Kirchhoff migration, offset-domain RMO analysis methods were developed first (Al-Yahya, 1989; Etgen, 1990). Several methods for RMO analysis after migration in the angle domain have also been proposed (Ottolini and Claerbout, 1984; Whitmore and Garing, 1993; Jiao et al., 2002); however, the application of RMO analysis of ADCIGs parametrized as a function of the reflection opening angles is a recent development (Biondi and Sava, 1999; Clapp and Biondi, 2000; Mosher et al., 2001).

Regardless of the domain in which the prestack partial images are defined, the RMO function is usually parametrized by a one single parameter. As discussed in the traveltime tomography section (Section 10.4), the more parameters we use to define an RMO function, the noisier the measurements of these parameters are likely to be. Therefore, there is a trade-off between data uncertainty and theoretical resolution. The most common choice is to parametrize the

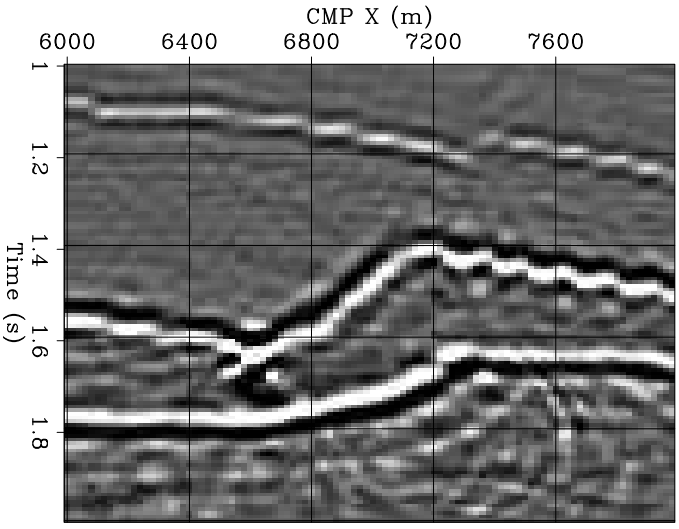


Figure 11.2: Migration of the fault-plane interface with the correct velocity. This section is the result of migrating the data within an offset range 240 meters wide and centered at 500 meters. migvel-InLine-1-mo [ER,M]

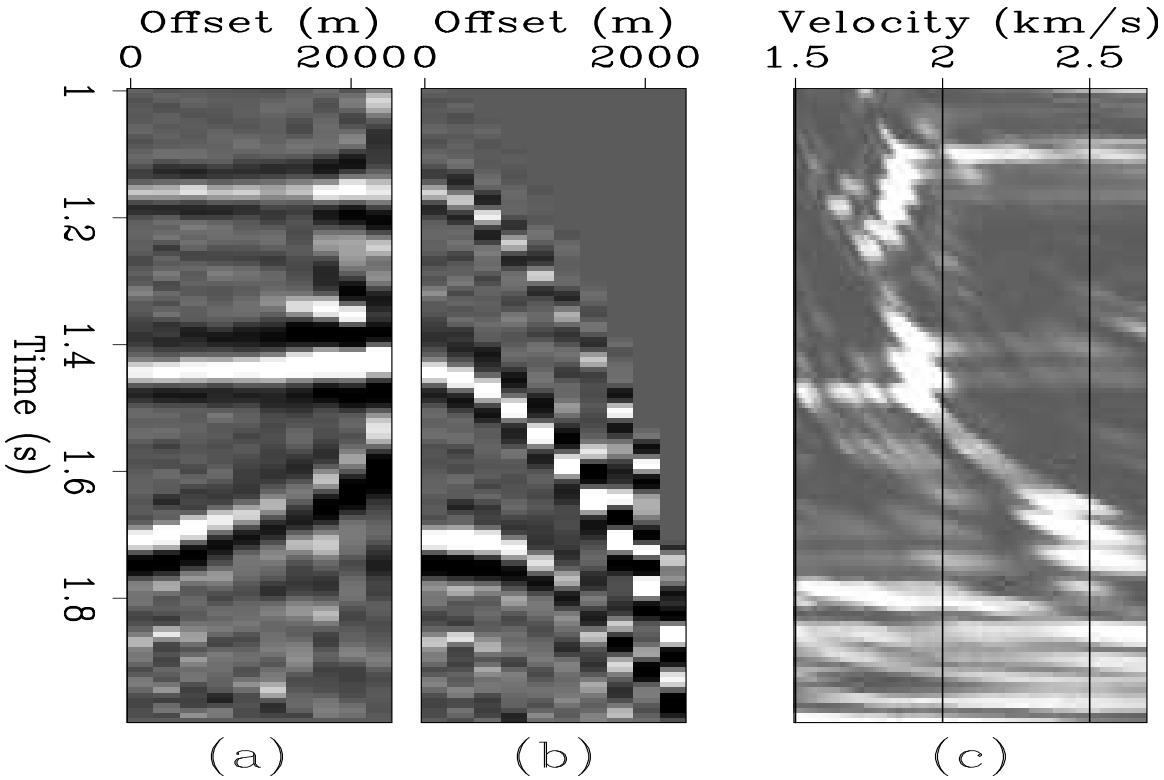


Figure 11.3: CIG after migration with the correct velocity function. This CIG is located at CMP X=7.02 kilometers in the migrated section shown in Figure11.2: (a) original CIG, (b) CIG after inverse NMO, (c) velocity spectrum of CIG after inverse NMO. migvel-Cmp-1-all [ER,M]

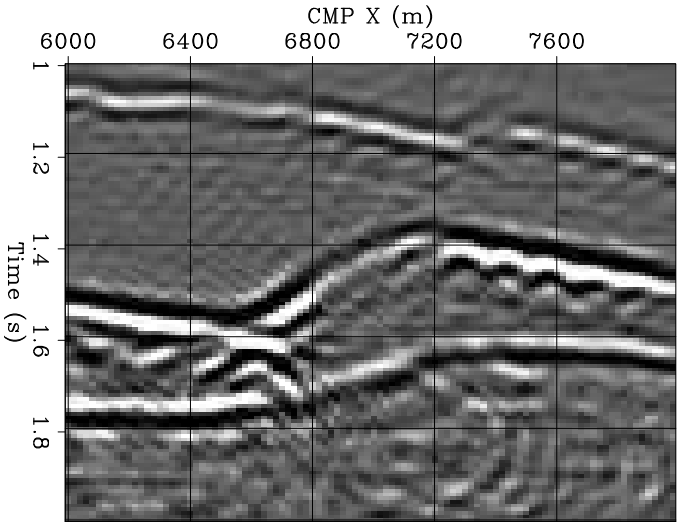


Figure 11.4: Migration of the fault-plane interface with the velocity scaled by a factor .92. This section is the result of migrating the data within an offset range 240 meters wide and centered at 500 meters. migvel-InLine-.92-mo [ER,M]

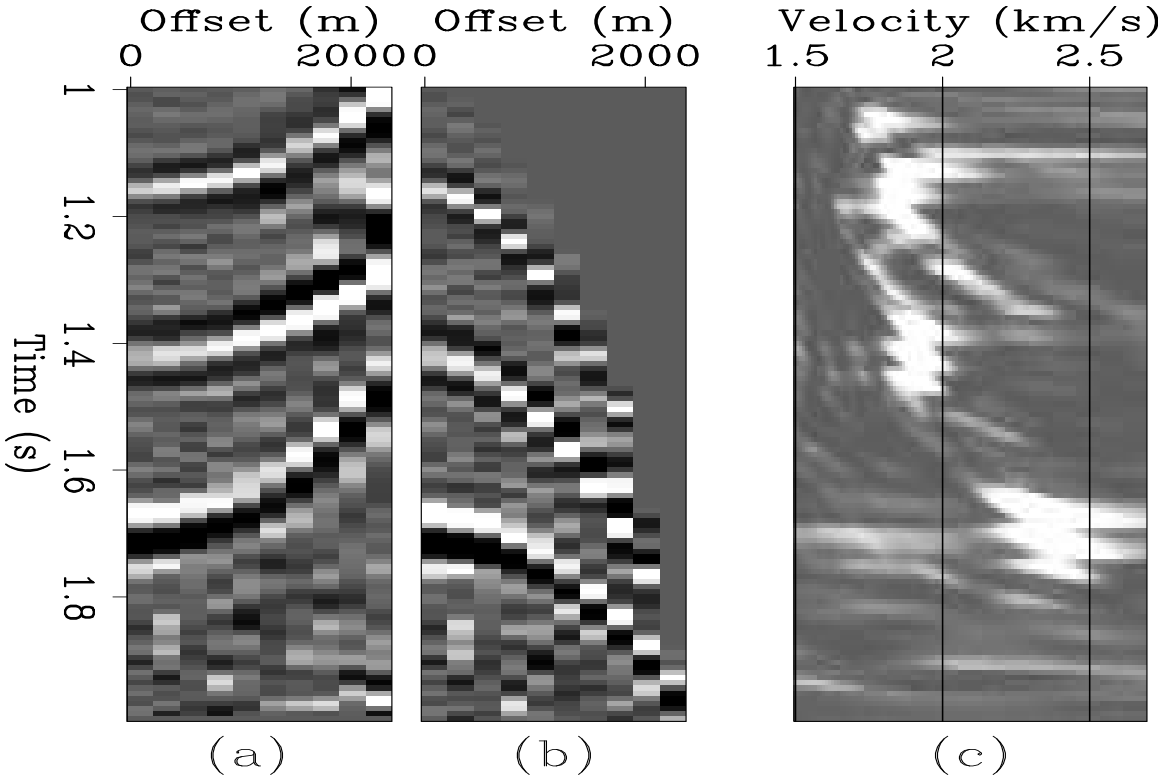


Figure 11.5: CIG after migration with the velocity scaled by a factor .92. This CIG is located at CMP X=7.02 kilometers in the migrated section shown in Figure11.2: (a) original CIG, (b) CIG after inverse NMO, (c) velocity spectrum of CIG after inverse NMO. migvel-Cmp-.92-all [ER,M]

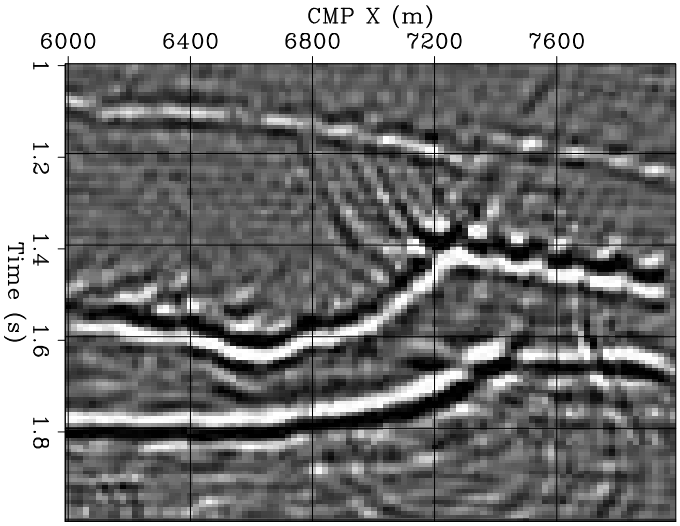


Figure 11.6: Migration of the fault-plane interface with the velocity scaled by a factor 1.08. This section is the result of migrating the data within an offset range 240 meters wide and centered at 500 meters. migvel-InLine-1.08-mo [ER,M]

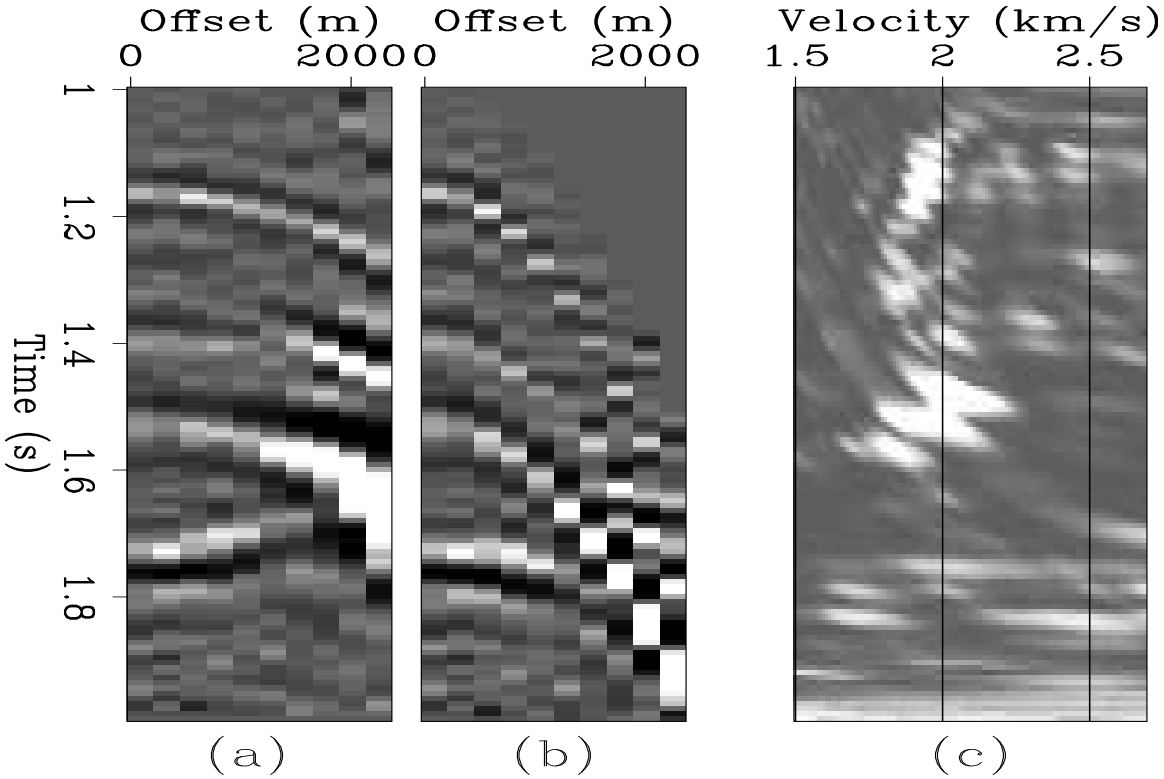


Figure 11.7: CIG after migration with the velocity scaled by a factor 1.08. This CIG is located at CMP X=7.02 kilometers in the migrated section shown in Figure11.2: (a) original CIG, (b) CIG after inverse NMO, (c) velocity spectrum of CIG after inverse NMO. migvel-Cmp-1.08-all [ER,M]

RMO function with only one parameter that describes the "average" of the migration-velocity error. This parameter is often chosen to be the ratio, ρ, between the average of the true migration slowness, s, and the average of the current migration slowness, s_ρ; that is, $\rho = s_\rho/s$.

In this section, we focus our attention on the methods that parametrize the RMO function with only one parameter, but the reader should be aware that more flexible parameterizations are possible. At the limit, the depth errors could be picked for each value of the offsets and reflection angles (van Trier, 1990). An intermediate solution was proposed by Lafond and Levander (1993), who define a two-parameter functional form for the RMO function. An interesting solution, which preserves the information contained in the full prestack partial images but avoids data picking, is the Differential Semblance Optimization (**DSO**) method introduced by Symes and Carazzone (1991). DSO minimizes the inconsistency between the images at contiguous offsets (or aperture angles), for all the values of the offset (or angle).

RMO functions in 2-D

We start with the derivation of RMO functions in 2-D, and then generalize them to 3-D. The simplest case is when the reflectors are flat and the RMO is defined as a function of the data absolute offset, $|\mathbf{h}|$. This RMO function was first presented by Al-Yahya (1989), and derived according to the following simple procedure.

In a constant-velocity medium, the reflection traveltimes from a flat interface at depth z are given by the following expression:

$$t = 2s\sqrt{z^2 + |\mathbf{h}|^2}. \tag{11.1}$$

Since traveltimes are an "observed" quantity, they are independent from the migration velocity. When the data are migrated with the wrong slowness s_ρ, the same traveltimes are thus expressed as follows:

$$t = 2s_\rho\sqrt{z_\rho^2 + |\mathbf{h}|^2}, \tag{11.2}$$

where z_ρ is the migrated depth. Equating the right-hand sides of equation (11.1) and (11.2), and performing simple algebraic manipulations, we derive the following expression for z_ρ:

$$z_\rho = \sqrt{\frac{z^2}{\rho^2} + \frac{1-\rho^2}{\rho^2}|\mathbf{h}|^2} = \frac{z}{\rho}\sqrt{1 + \left(1-\rho^2\right)\frac{|\mathbf{h}|^2}{z^2}}. \tag{11.3}$$

From equation (11.3) it is trivial to derive the residual moveout with respect to the zero-offset depth $z_{\rho 0} = z_\rho(h=0) = z/\rho$:

$$\mathbf{z}_{\text{RMO}} = \left(\frac{z}{\rho}\sqrt{1 + \left(1-\rho^2\right)\frac{|\mathbf{h}|^2}{z^2}} - \frac{z}{\rho}\right)\mathbf{z} = z_{\rho 0}\left(\sqrt{1 + \frac{1-\rho^2}{\rho^2}\frac{|\mathbf{h}|^2}{z_{\rho 0}^2}} - 1\right)\mathbf{z}, \tag{11.4}$$

where is $\mathbf{z}$ is the unit vector in the depth direction. Equation (11.4) can be directly applied to measure the value of ρ from ODCIGs, by computing ρ-spectra and picking the semblance

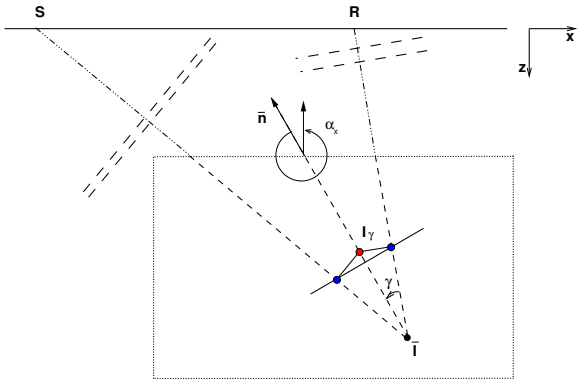


Figure 11.8: Geometry of a 2-D ADCIG for a single event migrated with the wrong velocity. The geological dip angle is labeled as α_x, and the aperture angle as γ. The velocity error causes the shift of the image point from $\bar{\mathbf{I}}$ to $\mathbf{I}_\gamma$. Compare this figure with Figure 6.4. migvel-cig-2d-mva-v1 [NR]

peaks in the ρ-spectra. However, Equation (11.4) is non-linear with respect to ρ, and its generalization to dipping events is challenging to apply (Al-Yahya, 1989).

To overcome these difficulties we derive a RMO function that is valid for arbitrary reflectors' dips but is linear with respect to ρ. The linearization of equation (11.4) around $\rho = 1$ yields the following RMO expression:

$$\Delta \mathbf{z}_{\text{RMO}} \approx (\rho - 1) \left. \frac{\partial \mathbf{z}_{\text{RMO}}}{\partial \rho} \right|_{\rho=1} = -(\rho - 1) \frac{|\mathbf{h}|^2}{z_{\rho 0}}, \tag{11.5}$$

which is linear in ρ but parabolic in offset. Notice that the addition of Δ in the notation indicates that it is a linearized expression. Equation (11.5) is much simpler than equation (11.4); it is also fairly accurate and is often applied in practice.

To generalize equation (11.5) to dipping reflectors, I switch to the angle domain, because the derivation and the expressions are simpler there than in the offset domain. An angle-domain RMO function is of immediate applicability to ADCIGs computed by either integral migration methods (Section 6.1.2) or wavefield-continuation methods (Section 6.2). Liu and Bleistein (1995) and Meng and Bleistein (2001) discuss the application of angle-domain relationships to the ODCIGs obtained by Kirchhoff migration.

The sketch in Figure 11.8 represents the geometry of a ADCIG in 2-D. It is analogous to Figure 6.4, but it illustrates the case where the migration velocity is inaccurate (too low). Biondi and Symes (2003) demonstrated that in the angle domain, to first order, the movement of a dipping event is along the normal **n**. In Figure 6.4, the velocity error causes the image point to shift along the normal **n** from $\bar{\mathbf{I}}$ to $\mathbf{I}_\gamma$. We will thus derive an expression for the relative shift along the normal direction ($\Delta \mathbf{n}_{\text{RMO}}$), instead of the relative shift along the vertical direction ($\Delta \mathbf{z}_{\text{RMO}}$).

By applying the chain rule, the linearized RMO function can be represented as the product of two functions: one linking linearized perturbations in average slowness, $\Delta s = s(\rho - 1)$, to linearized traveltime perturbations, Δt, and one linking traveltime perturbations to the perturbations in the reflectors' relative shifts along the normals to the reflectors, $\Delta \mathbf{n}_{\text{RMO}}$. The RMO shift, $\mathbf{n}_{\text{RMO}}$ is given by the difference between the shift at arbitrary aperture angle, $\mathbf{n}_\gamma$, and the shift at normal incidence, $\mathbf{n}_0$. The chain rule must be applied separately to these two

components, as indicated by the following expression:

$$\begin{aligned}\left.\frac{\partial \mathbf{n}_{\text{RMO}}}{\partial \rho}\right|_{rho=1} &= \left.\frac{\partial \mathbf{n}_{\gamma}}{\partial \rho}\right|_{rho=1} - \left.\frac{\partial \mathbf{n}_{0}}{\partial \rho}\right|_{rho=1} \\ &= \left.\frac{\partial \mathbf{n}_{\gamma}}{\partial t_{\gamma}}\right|_{rho=1} \left.\frac{\partial t_{\gamma}}{\partial \rho}\right|_{rho=1} - \left.\frac{\partial \mathbf{n}_{0}}{\partial t_{0}}\right|_{rho=1} \left.\frac{\partial t_{0}}{\partial \rho}\right|_{rho=1}. \end{aligned} \tag{11.6}$$

The physical interpretation of this linearization is that both terms in the chain rule are evaluated assuming stationary raypaths. The second terms in the chain rule ($\partial t/\partial \rho$) yield the traveltime perturbations caused by the scaling of the slowness along the ray paths. The first terms ($\partial \mathbf{n}/\partial t$) determine the movements of the end points of the raypaths: these are necessary to compensate for the traveltime perturbations along the raypaths, and maintain constant the total traveltimes. These derivatives ($\partial \mathbf{n}/\partial t$) can be easily evaluated by trigonometric considerations based on the sketch in Figure 11.8, as proven by Bishop et al. (1985) for ODCIGs and by Biondi and Symes (2003) for ADCIGs, and they are given by the following expressions:

$$\left.\frac{\partial \mathbf{n}_{\gamma}}{\partial t_{\gamma}}\right|_{rho=1} = \frac{\mathbf{n}}{2\hat{s}\cos\gamma}, \tag{11.7}$$

and

$$\left.\frac{\partial \mathbf{n}_{0}}{\partial t_{0}}\right|_{rho=1} = \frac{\mathbf{n}}{2\hat{s}}, \tag{11.8}$$

where $\hat{s}$ is the current slowness around which we are linearizing the RMO function.

The second terms in the chain rule ($\partial t/\partial \rho$) can be evaluated by differentiating the expression for the prestack traveltimes given in equation (6.4), as follows:

$$\left.\frac{\partial t_{\gamma}}{\partial \rho}\right|_{rho=1} = 2\hat{s}\frac{\cos\alpha_x\cos\gamma}{\cos^2\alpha_x - \sin^2\gamma} z_{\rho 0}, \quad \text{and} \quad \left.\frac{\partial t_{0}}{\partial \rho}\right|_{rho=1} = 2\hat{s}\frac{1}{\cos\alpha_x} z_{\rho 0}. \tag{11.9}$$

Substituting equations (11.7), (11.8), and equations (11.9) into equation (11.6), we obtain the general expression for the linearized 2-D RMO function:

$$\Delta \mathbf{n}_{\text{RMO}} = \frac{\rho - 1}{\cos\alpha_x} \frac{\sin^2\gamma}{\left(\cos^2\alpha_x - \sin^2\gamma\right)} z_{\rho 0}. \tag{11.10}$$

According to equation (11.10), the RMO shift increases as a function of the apparent geological dip $|\alpha_x|$. The intuitive explanation for this behavior is that the specular rays become longer as the apparent geological dip increases, and consequently the effects of the slowness scaling increase. In the special case of flat reflectors ($\alpha_x = 0$), equation (11.10) simplifies into

$$\Delta \mathbf{n}_{\text{RMO}} = (\rho - 1) z_{\rho 0} \tan^2\gamma, \tag{11.11}$$

which is the angle-domain equivalent of equation (11.5). Notice that the two equations have opposite signs because in Figure 11.8 the normal vector $\mathbf{n}$ is defined as directed upward.

To illustrate the different behavior of the two RMO expressions as a function of the reflectors' dips, I will use the migration results of a synthetic data set acquired over a circular

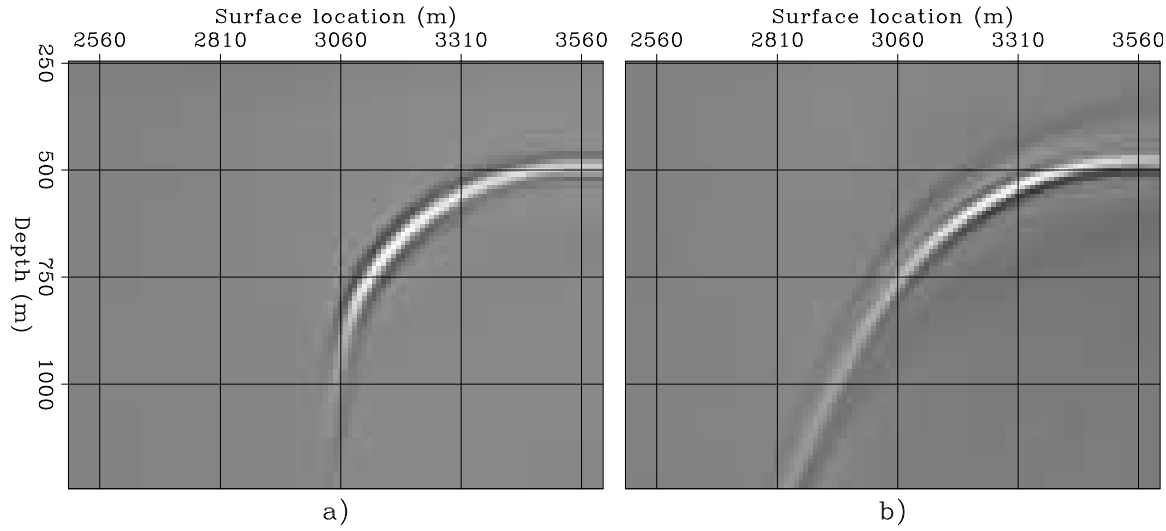


Figure 11.9: Images of the synthetic data set obtained with a) correct velocity, b) too low velocity ($\rho = 1.04$). migvel-Mig-zo-overn [ER]

reflector. The whole data set was migrated twice: first using the correct velocity (2,000 m/s), and second after scaling the slowness function by a constant factor $\rho = 1.04$ (corresponding to a velocity of 1,923 m/s). Figure 11.9a shows the zero-offset section (stack) of the migrated cubes with the correct velocity, and Figure 11.9b shows the zero-offset section obtained with the low velocity.

Figure 11.10 illustrates the accuracy of the two RMO functions when predicting the actual RMO in the migrated images obtained with a constant slowness function with $\rho = 1.04$. The four panels show the ADCIGs corresponding to different apparent reflector dip: a) α = 0; b) α = 30; c) α = 45; d) α = 60. Notice that the vertical axes change across the panels; in each panel the vertical axis is oriented along the direction normal to the respective apparent geological dip **n**. The solid lines superimposed onto the images are computed using equation (11.10), whereas the dashed lines are computed using equation (11.11). The solid lines overlap the migration results for all dip angles. This figure demonstrates that, when the slowness perturbation is sufficiently small (4 % in this case), the assumption of stationary raypaths causes only small errors in the predicted RMO. In contrast, the dashed lines predicted by the "flat-reflectors" RMO function [equation (11.11)] are an acceptable approximation of the actual RMO function only for small dip angles (up to 30 degrees). For large dip angles, a value of ρ substantially higher than the correct one would be necessary to fit the actual RMO function with equation (11.11). If this effect of the reflector dip is not properly taken into account, the false indications provided by the inappropriate use of equation (11.11) may prevent the MVA process from converging.

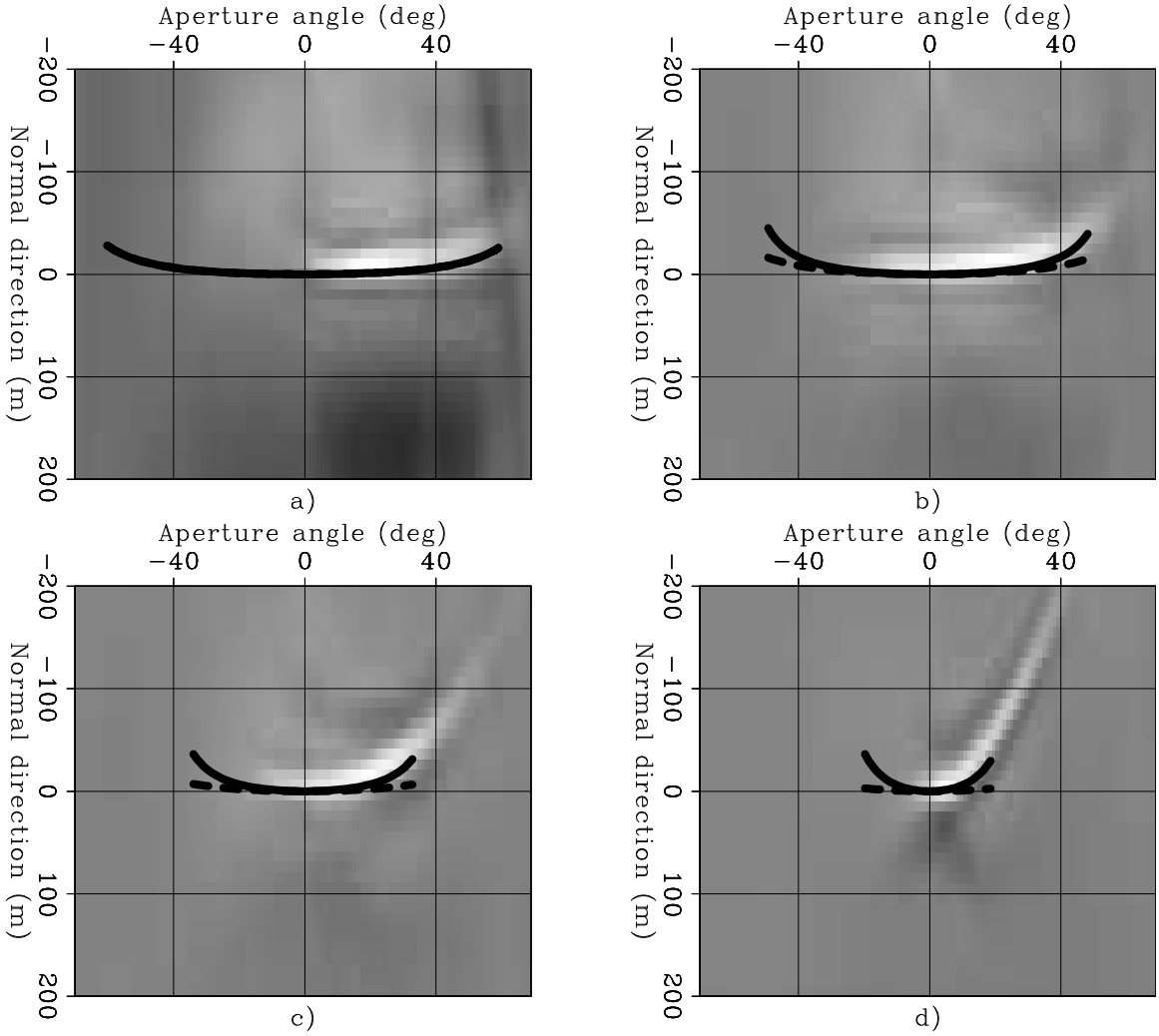


Figure 11.10: ADCIGs for four different apparent reflector dips: a) α = 0; b) α = 30; c) α = 45; d) α = 60 with $\rho = 1.04$. Superimposed onto the images are the RMO functions computed using equation (11.10) (solid lines), and using equation (11.11) (dashed lines). Notice that the vertical axes change across the panels; in each panel the vertical axis is oriented along the direction normal to the respective apparent geological dip. migvel-Ang-Cig-slow-4p-overn [ER]

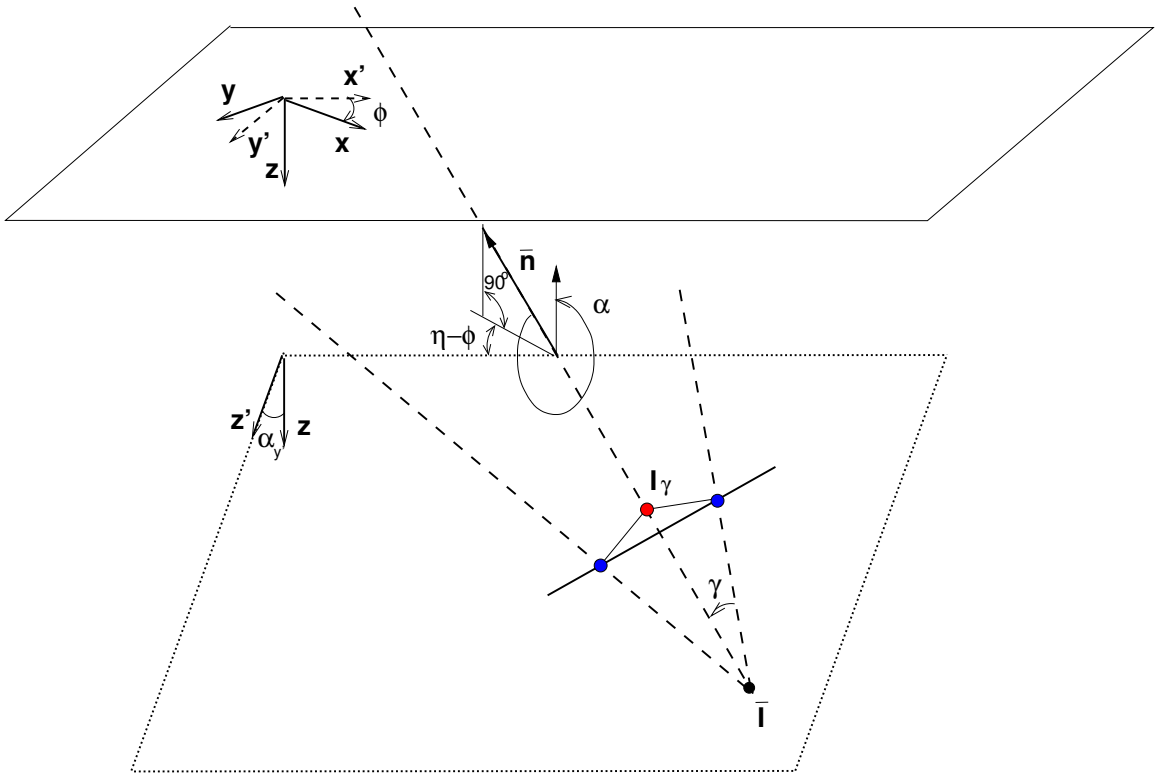


Figure 11.11: Geometry of an ADCIG for a single event migrated with the wrong velocity. The velocity error causes the shift of the image point from $\bar{\mathbf{I}}$ to $\mathbf{I}_\gamma$. Compare this figure with Figure 6.17. migvel-cig-3d-mva-v6 [NR]

RMO functions in 3-D

The 2-D analysis can be extended to 3-D by following the same geometrical considerations I used in the Section 6.2 to generalize ADCIGs from 2-D to 3-D (Biondi and Tisserant, 2004). Figure 11.11 graphically illustrates this idea and shows the geometry of angle-domain CIG for a single event migrated with the wrong (low in this case) velocity. However, there is an important conceptual difference between the 2-D case and the 3-D case. In 2-D we can assume that the source and receiver rays cross, even when the data were migrated with the wrong velocity; below the imaging point when the migration velocity is too low and above the imaging point in the opposite case. In 3-D, this assumption is easily violated, because two rays are not always coplanar.

Fortunately, the plane-wave interpretation that we discussed in Section 6.2.1 is still valid. The coplanar plane of Figure 11.11 is determined by the coplanarity condition expressed in equation (6.18) applied to the phase vectors of the incident and reflected plane waves, or, alternatively by the coplanarity condition expressed in equation (6.25) applied to the offset dips present in the prestack image. The source and receiver plane waves constructively interfere along an imaging line; the angle-domain image point $\mathbf{I}_\gamma$ is defined as the intersection of this imaging line with the tilted coplanar plane. Once the coplanar plane is defined, the geomet-

rical relations between the objects (rays and imaging points) that lie on the vertical plane in 2-D directly apply to 3-D on the tilted coplanar plane, as is schematically represented in Figure 11.11. In particular, the relative positions of the angle-domain image point $\mathbf{I}_\gamma$ and the crossing point between the source and receiver ray $\overline{\mathbf{I}}$ are defined on the tilted plane as if they were on the 2-D vertical plane.

It is important to notice that when the migration velocity is inaccurate, the azimuth ϕ of the tilted plane is not necessarily the true azimuth of the reflection. The discrepancies between the apparent azimuth and the true azimuth are analogous to the discrepancies between apparent aperture angle and true aperture angle that is illustrated by the numerical example presented in Section 6.2.2. These "geometrical" errors are related to both the ray bending caused by the velocity errors during propagation, and the dependency of the transformation to angle domain on the local velocity [through k_z in equations (6.25) and (6.24)]. These errors are not taken into account in our linear analysis of the RMO function [equation 11.6]. As with all non-linear effects, they are not problematic when the migration velocity is close to the correct one, but they may slow convergence when the velocity estimation process starts too far from the correct solution.

Given the geometrical insight discussed above, the expression for the 2-D RMO function [equation (11.10)] can be easily generalized to 3-D. In 3-D, equation (11.10) is valid on the tilted plane shown in Figure 11.11, where the migrated depth of the normal-incidence event z_0 is now the depth along the tilted plane z_0'. Therefore, the migrated depth z_0 needs to be scaled by a factor of $1/\cos\alpha_{y'}$, where $\alpha_{y'}$ is the inclination of the tilted plane. In 3-D, equation (11.10) becomes

$$\begin{aligned}\Delta\mathbf{n}_{\text{RMO}} &= \frac{\rho-1}{\cos\alpha_{x'}}\frac{\sin^2\gamma}{\left(\cos^2\alpha_{x'}-\sin^2\gamma\right)}z_0'\,\mathbf{n} \\ &= \frac{\rho-1}{\cos\alpha_x\cos\alpha_y}\frac{\sin^2\gamma}{\left(\cos^2\alpha_{x'}-\sin^2\gamma\right)}z_0\,\mathbf{n}. \end{aligned} \tag{11.12}$$

When $\alpha_{x'}$ and $\alpha_{y'}$ are available, the RMO function could be directly evaluated using the expression in equation (11.12). However, in several situations it is more useful to express the RMO function explicitly as a function of the geological dip α, the azimuth angle of the normal to the geological dip η, and the azimuth angle of the reflected event ϕ (see Figure 11.11), with the following expression:

$$\Delta\mathbf{n}_{\text{RMO}} = \frac{\rho-1}{\cos\alpha}\frac{\sin^2\gamma}{\left(1-\sin^2\alpha\cos^2(\eta-\phi)-\sin^2\gamma\right)}z_0\,\mathbf{n}. \tag{11.13}$$

Equation (11.13) is the general expression of the RMO function in 3-D.

In practice, it is computationally simpler to apply the RMO function at a fixed surface location, that is, as a vertical shift ($\Delta\mathbf{z}_{\text{RMO}}$) instead as a normal shift ($\Delta\mathbf{n}_{\text{RMO}}$). To derive the $\Delta\mathbf{z}_{\text{RMO}}$ functions from the $\Delta\mathbf{n}_{\text{RMO}}$ functions, we just apply the relationship $\Delta\mathbf{z}_{\text{RMO}} = \Delta\mathbf{n}_{\text{RMO}}\cos\alpha$. In the presence of dipping events, the vertical RMO correction might not be as accurate as the normal RMO correction, because of the **image-point dispersal** phenomenon. After migration with an inaccurate velocity, events with different aperture angles are imaged

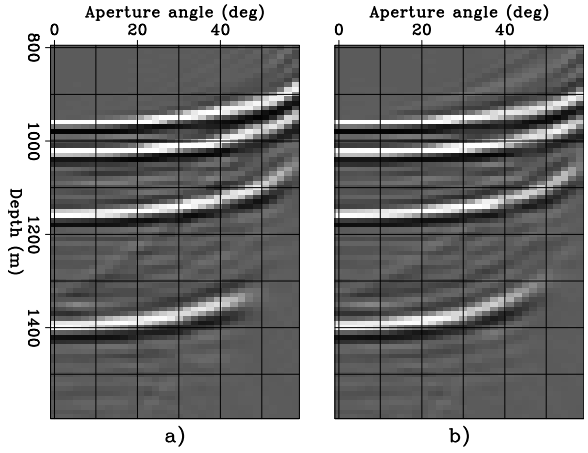


Figure 11.12: ADCIGs as functions of depth (z) and aperture angle (γ) for two different reflection azimuths and at constant horizontal location ($x = y = 700$ meters): $\phi = 0°$ (a), and $\phi = 7°$ (b). migvel-cig-2-data7 [ER]

at different horizontal locations, even if they originated at the same reflecting point in the subsurface.

Example of 3-D RMO analysis

The following example illustrates the application of the 3-D RMO corrections derived above, and the effects of their respective approximations. Figure 11.11 shows a section cut through an ADCIG obtained from migrating the same synthetic common-azimuth data set used in Section 6.2.3 to illustrate the concept of 3-D ADCIGs. The data set contains five planes, dipping at 0°, 15°, 30°, 45° and 60° toward increasing x and y as shown in Figure 6.22. The azimuth of the planes is 45 degrees with respect to the direction of acquisition. In contrast with the ADCIGs shown in Section 6.2.3, which were obtained by using the correct migration velocity, in this case the migration velocity was too slow. The data were migrated with a velocity function 3% slower than the correct one; that is, the slowness function was scaled by 1.03 at every depth level.

All the following figures show slices of the ADCIGs at one fixed horizontal location with $x = y = 700$ meters. At this horizontal location there are four reflectors. Starting from the shallower reflector, their respective dips are: 0, 15, 30 and 45 degrees. Figure 11.12 shows two of these ADCIGs, for two different reflection azimuths: $\phi = 0$ degrees (panel a) and $\phi = 7$ degrees (panel b). As expected the ADCIG smiles upward, because the migration velocity is too slow.

The three panels in Figure 11.13 display the image as a function of depth (z) and reflection azimuth (ϕ), and at constant aperture angle. The aperture angles are a) γ=4 degrees, b) γ=20 degrees, and c) γ=40 degrees. The curvature of the events as a function of the azimuth increases with the aperture angle, indicating that the RMO depends on the reflection azimuth. However, as with the correct velocity ADCIGs shown in Figure 6.25, this dependency occurs not because the general expression of the RMO function [equation (11.13)] depends on azimuth, but because the common-azimuth data "illuminate" only a narrow range of reflection azimuths at wide aperture angles.

The apparent reflection azimuth corresponds to the apex of the curves shown in Figure 11.13. From the figure it is evident that it increases with the reflectors' dip. On one end, the apparent azimuth of the flat reflector (top) is zero, and on the other end, the apparent azimuth of the steepest reflector is about 7 degrees. *A priori*, the reflection azimuth is an unknown. Therefore, to avoid ambiguities, the RMO analysis of narrow-azimuth data, like the ones in this example, should be performed on ADCIGs obtained by averaging over a range of reflections azimuths. This stacking over azimuths would strongly attenuate the events with wide aperture angles, unless a proper Jacobian correction is applied to the ADCIGs before stacking (Biondi, 2003). The ADCIG shown in Figure 11.14a was obtained by averaging the ADCIGs over the azimuthal range $-20^o \leq \phi \leq 20^o$, after the application of the appropriate Jacobian correction.

The panels in Figure 11.14b-d show ρ-spectra computed by using three different RMO functions. The panels are centered at $\rho = 1.3$, which corresponds to the actual scaling of the migration velocity. Figure 11.14b shows the ρ-spectrum computed using the flat-reflectors approximation [equation (11.11)]. As the reflector dip increases, the semblance peaks shift toward the right; that is, the value of ρ is overestimated. This bias introduced by the flat-event RMO function is consistent with the results shown in Figure 11.10.

Figure 11.14c shows the application of the full 3-D RMO function [equation (11.13)] with the correct values of the geological dip α, but ignoring the relative rotation between the azimuth of the reflectors and the azimuth of the reflections (i. e. assuming $\eta = \phi$). In this case, the RMO function used for computing the ρ-spectrum has a larger curvature than the actual RMO in the ADCIG. Therefore, the semblance peaks shift toward the left, and the ρ of the dipping reflectors is underestimated.

Finally, Figure 11.14d shows the application of the full 3-D RMO function with the correct value of η (45 degrees), and approximately the correct value of ϕ (0 degrees). The semblance peaks are approximately in the right positions. There is a residual error caused by the fact that the background velocity increases linearly with depth; rather than being constant as the derivation of equation (11.13) assumes. Because of ray bending, the true rays are shorter than the straight rays assumed by the derivation of equation (11.13). Consequently, the traveltime errors accumulated along the true rays are lower than those predicted along straight rays, and

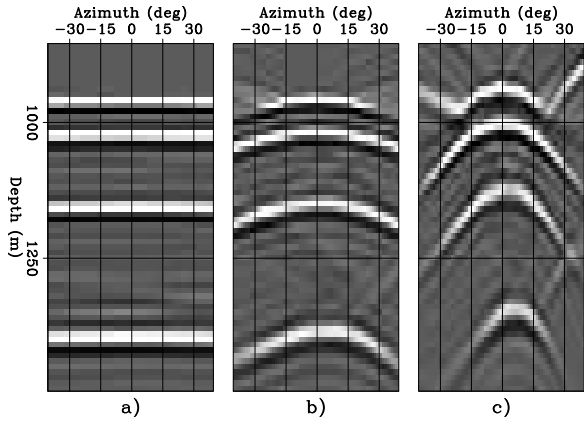


Figure 11.13: ADCIGs as functions of depth (z) and reflection azimuth (ϕ) for three different aperture angles and at constant horizontal location ($x = y = 700$ meters): $\gamma = 4°$ (a), $\gamma = 20°$ (b), and $\gamma = 40°$ (c). migvel-azim-gamma-all-data7 [ER]

ρ is slightly underestimated.

As I remarked above, the azimuth of the reflections is not known *a priori*. It is likely to fall within a range centered around the data-acquisition azimuth, but, in general, it is dependent on the velocity model in the overburden as well as the orientations of the data azimuth and of the geological dips. I will now describe a simple procedure for estimating the reflection azimuth by exploiting its effects on the RMO in the ADCIGs. The ability to estimate the reflection azimuth directly from the ADCIGs is attractive. It has the potential to improve the RMO analysis, because equation (11.13) depends on the difference between the reflection azimuth and the azimuth angle of the normal to the geological dip ($\eta - \phi$). Furthermore, the accuracy of a tomographic updating of the interval-velocity function (Section 11.4) would benefit from the knowledge of the reflection azimuth.

The estimation of the reflection azimuth starts with the computation of ρ-spectra, like the one shown in Figure 11.14b-d. Once the parameters of the RMO function are determined, a ϕ-spectrum can be computed by applying the fixed RMO function to all the ADCIGs with the azimuths within a range of interest, and computing the semblance along the γ axis of the moved-out gathers. Figure 11.15 shows an example of a ϕ-spectrum. For each reflector, the value of the apparent reflection azimuth corresponds to the semblance peaks in the panel. As expected, the semblance peak for the flat reflector is at $\phi = 0$, whereas the semblance peaks shift to the right as the reflectors' dips increase. The semblance peak for the most steeply dipping reflector (45 degrees) is at about 7 degrees, which is consistent with the position of the apex of the curve shown in Figure 11.13.

11.2.2 Residual prestack migration

The RMO analysis presented in the previous section provides information on the migration-velocity errors by measuring the consistency of the prestack partial images as a function of the reflection angles (or data offsets). This CIG-flatness criterion has the advantage of being independent from the structure of the subsurface. However, it is not the only criterion for judging the quality of the migration velocity. The focusing of the reflections in the physical space (midpoint and depth) often carries additional valuable velocity information. This information can be extracted form the prestack partial images by analyzing the results of **residual prestack migration**. This process is more interpretative and structure-dependent than the RMO analysis; therefore, it is more challenging to automate and is used more rarely. In areas where diffracted events are numerous (e.g. rugose salt bodies), analyzing the residual focusing of diffractions can provide useful information for updating interval velocities (Sava et al., 2004).

Another advantage of residual migration over residual moveout is that it further reduces the effects of the image-point dispersal (discussed in previous section) between events imaged at the same physical location but with different aperture angles. The removal of the image-point dispersal can improve the accuracy of the semblance spectra computed after residual migration, compared to the ones computed after residual moveout. One major disadvantage of residual migration is computational complexity. Residual migrating large 3-D prestack partial images can be both computationally intensive and a data-handling challenge.

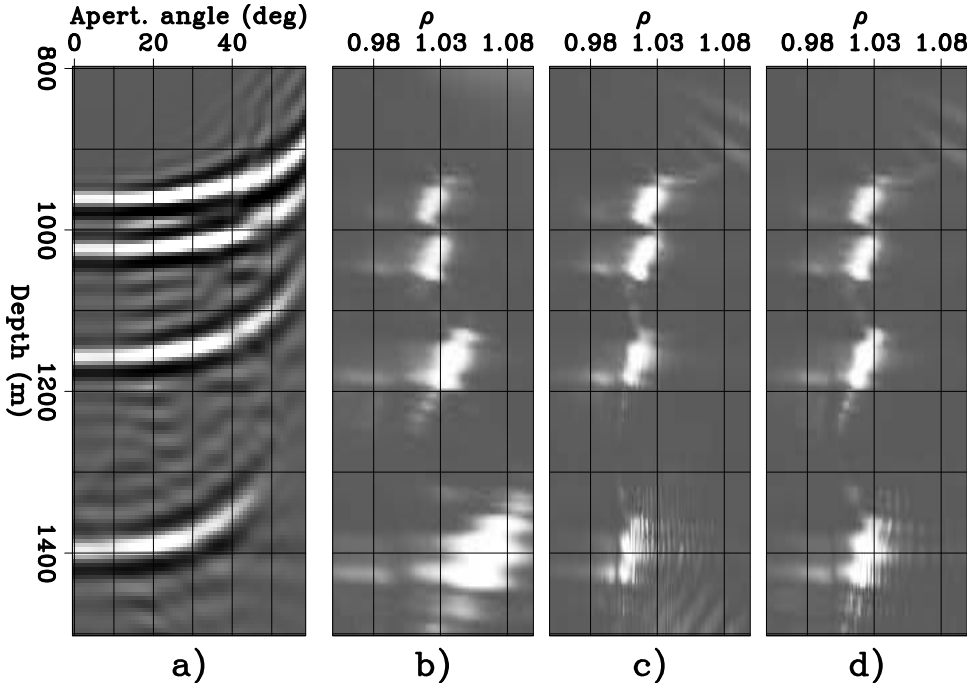


Figure 11.14: ADCIG and corresponding ρ-spectra computed using the 3-D RMO function in equation (11.13) with different sets of parameters: $\alpha = 0$ (b), $\eta = \phi$ (c), $\eta = 45^o$ and $\phi = 0^o$ (d). migvel-zg_zs_zsa-overn [ER]

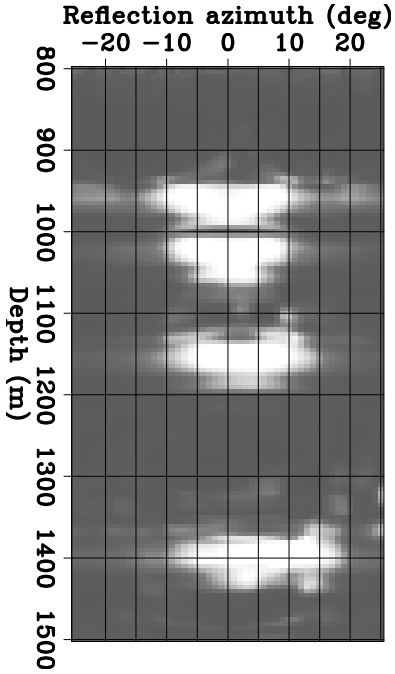


Figure 11.15: Semblance plot used to estimate the apparent reflection azimuth of reflections. For each reflector, the position of the semblance peaks corresponds to the estimated reflection azimuth. migvel-pfan-overn [ER]

Rocca and Salvador (1982) introduced the concept of residual migration for zero-offset time migration. As for conventional full migration, residual migration can be performed either by applying an integral operator or by downward continuation (Rothman et al., 1985; Etgen, 1990; Fomel, 2003). The CRP-scan methodology developed by Audebert et al. (1996) is another method that is related to residual migration and that is applicable to Kirchhoff-like migrations. However, the most computationally efficient residual migration methods are based on the frequency-wavenumber domain algorithm first proposed by Stolt (1978) to perform constant-velocity migration. Both the prestack residual time-migration method proposed by Stolt (1996) and the prestack residual depth-migration method proposed by Sava (2003) are based the Stolt migration algorithm. In this section, we discuss the residual-migration method presented by Sava (2003), because it is the most closely related to the angle-domain residual moveout methodology presented in the previous section.

Prestack residual depth migration is conveniently parametrized using the same parameter used for RMO analysis; that is, the ratio, ρ, between the average of the true migration slowness, s, and the average of the actual migration slowness, s_ρ. As for conventional Stolt migration, residual migration is performed by a remapping of the vertical wavenumber (frequency) axis. Conventional full migration remaps the temporal frequency axis, ω, into the vertical wavenumber axis, k_z. Residual migration remaps the vertical wavenumber axis of the migrated partial images, k_{z_ρ} into the vertical wavenumber of the residual migrated partial images, k_z. This remapping is performed completely in the wavenumber domain, according to the following expression:

$$
\begin{aligned}
k_z \;=\; & \sqrt{\frac{1}{\rho^2}\frac{\left(k_{z_\rho}^2+|\mathbf{k_m}|^2\right)\left(k_{z_\rho}^2+|\mathbf{k_h}|^2\right)}{4k_{z_\rho}^2}-\frac{1}{4}(\mathbf{k_m}+\mathbf{k_h})\cdot(\mathbf{k_m}+\mathbf{k_h})} \\
& \sqrt{\frac{1}{\rho^2}\frac{\left(k_{z_\rho}^2+|\mathbf{k_m}|^2\right)\left(k_{z_\rho}^2+|\mathbf{k_h}|^2\right)}{4k_{z_\rho}^2}-\frac{1}{4}(\mathbf{k_m}-\mathbf{k_h})\cdot(\mathbf{k_m}-\mathbf{k_h})}\;. \qquad (11.14)
\end{aligned}
$$

To perform a residual-migration analysis we can apply equation (11.14) for several values of the ρ parameters, and then compute ρ-spectra by semblance analysis over the aperture angle axis. The semblance peaks in the ρ-spectra indicate the values of ρ that improve focusing the most. Alternatively, or as a complement to the semblance analysis, the stacks over the aperture angles and azimuths can be interpreted to determine the values of ρ which correspond to the most geologically plausible image, or the one with the best-focused diffracted energy.

Figures 11.16 and 11.17 show an example of this "structural" analysis. Figure 11.16 shows an in-line section taken from a 3-D marine data set recorded in the North Sea. The image shows some clear signs of undermigration. There is some unfocused diffracted energy from the tip of the salt body, and the sediments under the salt overhang are not well imaged. The thick box in Figure 11.16 indicates the area corresponding to residual migration results shown in Figure 11.17. This figure shows nine panels corresponding to different values of ρ starting from $\rho = 1.08$ to $\rho = 1$. The shape of the salt tip changes with ρ as steeply dipping energy that was not well focused in the original image comes into focus, and defocuses again at higher

values of ρ. The focusing of the bottom of the salt and of the sediments terminating against the salt overhang also improves in the residual migrated results.

Residual prestack migration and residual moveout

The residual migration expression in equation (11.14) and the residual moveout expression in equation (11.3) have a similar structure. Equation (11.3) performs a remapping of the vertical axis z, whereas equation (11.14) performs a remapping of its dual, k_z. The similarity is better highlighted when we simplify equation (11.14), assuming flat reflectors ($\mathbf{k_m} = 0$), and identifying $|\mathbf{k_h}|/k_{z_\rho}$ with $\tan\gamma_\rho$ (according to equation (6.24) with $\mathbf{k_m} = 0$), where $\tan\gamma_\rho$ is the apparent aperture angle with the actual migration slowness s_ρ. Under these assumptions, equation (11.14) becomes

$$k_z = \frac{k_{z_\rho}}{\rho}\sqrt{1+\left(1-\rho^2\right)\frac{|\mathbf{k_h}|^2}{k_{z_\rho}^2}} = \frac{k_{z_\rho}}{\rho}\sqrt{1+\left(1-\rho^2\right)\tan^2\gamma_\rho}. \tag{11.15}$$

Equation (11.3) can also be written easily as a function of the aperture angle γ as follows:

$$z_\rho = \frac{z}{\rho}\sqrt{1+\left(1-\rho^2\right)\frac{|\mathbf{h}|^2}{z^2}} = \frac{z}{\rho}\sqrt{1+\left(1-\rho^2\right)\tan^2\gamma}. \tag{11.16}$$

The right-hand sides of equations (11.15) and (11.16) are reciprocal scalings of Fourier dual variables (notice that we have hidden the non-stationarity of the scaling factor in the angles), and thus they represent the same transformation. This simple analytical exercise demonstrates that the non-linear RMO function for flat events is equivalent to residual migration of flat events. In general, the RMO functions that we derived in the previous section are approximations of the general residual migration expression in equation (11.14). There is a continuum of approximations that are possible between applying the flat-event RMO function and full residual migration. The choice of the method depends on the particular problem at hand.

To gain some intuition on how results differ between the two end members of this continuum, Figure 11.18 shows the results of applying residual moveout with equation (11.16) to the North Sea prestack partial images. These panels should be compared with the ones shown in Figure 11.17. The application of residual moveout improves the coherency and the power of the stack, but it does not correct the misfocusing of the salt tip, nor improve the image of the sediments below the salt overhang as much as residual migration does.

11.3 Vertical interval-velocity updates from measured average-velocity errors

The information on the average-velocity errors measured using the methodology presented in the previous section can be used to estimate interval-velocity updates in an iterative velocity-estimation scheme. The three steps of migration, RMO analysis, and velocity updating can

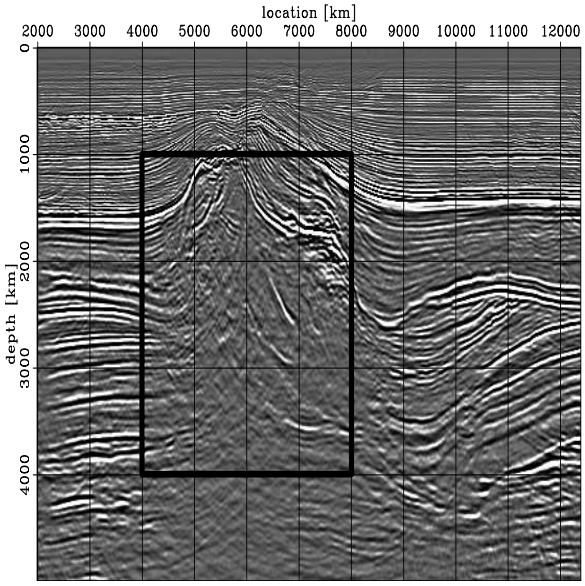


Figure 11.16: Migrated image of a North Sea data set. The thick box indicates the area corresponding to the residual migrations and residual moveouts results respectively shown in Figure 11.17 and Figure 11.18. (This figure is from Sava (2003)) migvel-saltreal.raw [ER]

be repeated until they converge to a velocity function that yields a well-focused image and migrated CIGs with an acceptable degree of flatness.

The simplest way to update the interval-velocity function from the average-velocity errors is to apply the Dix formula in equation (10.2). However, the Dix formula is expressed in two-way traveltimes τ, whereas the velocity-error measurements obtained from depth-migrated CIGs are expressed in depth. To reconcile this inconsistency, we convert the current migration velocity and the error measurements into time, perform the updates, and then convert the updated interval velocity back into depth. The main steps of this process can be summarized as follows:

1. Evaluate the mapping between depth and two-way traveltime using the current slowness model ${}_i\hat{s}$ (where the left subscript i indicates the iteration number) in the following relationship:
$$\tau(z) = 2\int_0^z {}_i\hat{s}\left(z'_\xi\right) dz'_\xi. \tag{11.17}$$

2. Transform the current slowness model ${}_i\hat{s}$ and the error measurements ρ from depth to time.

3. Transform the interval-velocity function into the RMS-velocity function ${}_i\hat{V}_{rms}(\tau)$ using equation (10.1).

4. Compute the new RMS-velocity function ${}_{i+1}\hat{V}_{rms}(\tau) = {}_i\hat{V}_{rms}(\tau)/\rho(\tau)$.

5. Apply the Dix formula to convert the new ${}_{i+1}\hat{V}_{rms}(\tau)$ into the new interval-velocity function.

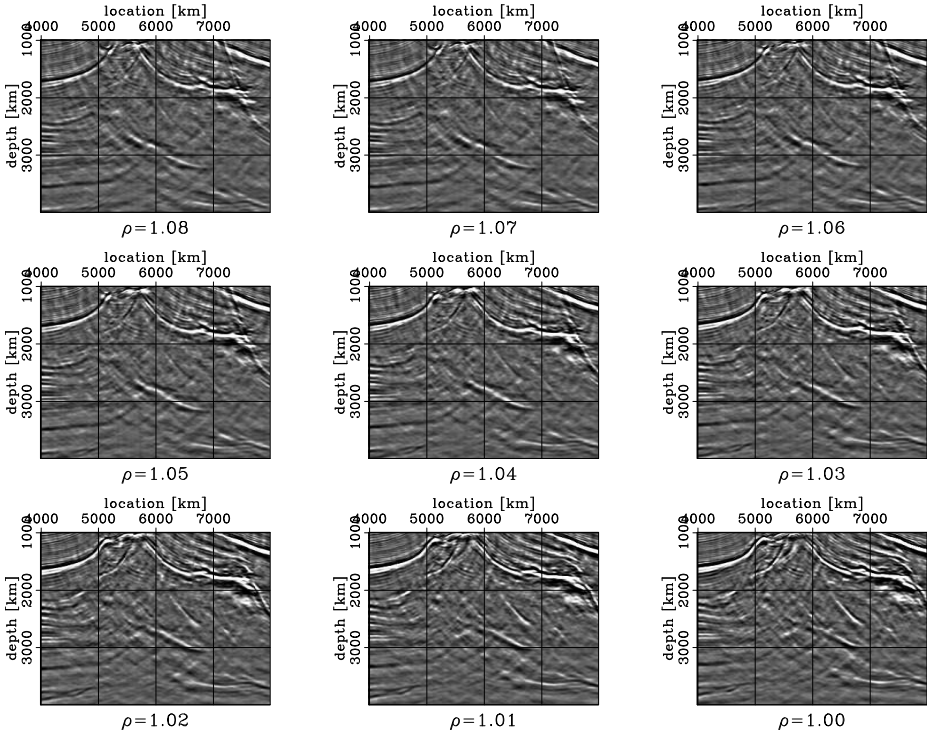


Figure 11.17: Stacks after residual migrations for $1 \leq \rho \leq 1.08$. (This figure is adapted from Sava (2003)) migvel-saltreal.srm.bio [ER]

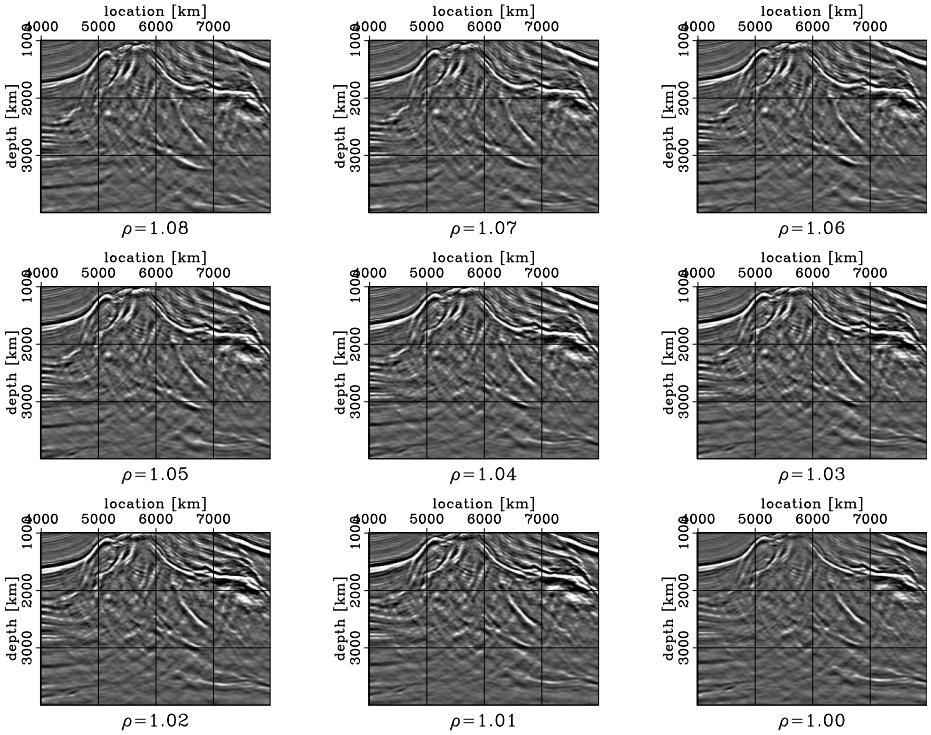


Figure 11.18: Stacks after residual moveout for $1 \leq \rho \leq 1.08$. (This figure is adapted from Sava (2003)) migvel-saltreal.rmo.bio [ER]

6. Recompute the time-to-depth mapping function using the new interval-slowness function ${}_{i+1}\hat{s}$ in the inverse of equation (11.17), that is,

$$z(\tau) = \frac{1}{2}\int_0^{\tau} \frac{d\tau'}{{}_{i+1}\hat{s}(\tau')}, \tag{11.18}$$

7. Map the new interval slowness function from time to depth using the new mapping function.

This process may seem circuitous compared to updating the interval velocity directly in depth by applying a depth-version of the Dix formula. However, by updating the interval velocity in time, and then using the *new* interval-velocity function to map it back to depth, we "reposition" the whole interval-velocity function according to the updated (and supposedly more accurate) time-to-depth mapping. We discuss this issue in more detail in the next section (Section 11.4), since it is important also for tomographic MVA methods (Kosloff et al., 1996; Biondi et al., 1998; Clapp, 2001).

An iterative process that includes the updating scheme outlined above, or a similar updating scheme, is often referred as a **Deregowski loop** (Deregowski, 1990). The Deregowski loop is adequate for depth migration when the velocity function does not vary rapidly in the lateral directions, and the reflectors' geometry is not too complex. However, the vertical back-projection of the residual-moveout analysis using the Dix formula is based on two assumptions that are not necessarily fulfilled in depth-migration problems: first, the velocity does not change rapidly within the offset span of the data. Second, the vertical direction used for the updating is a good approximation of the true normal to the reflectors. The first assumption is similar to the assumption made when inverting stacking velocity using the Dix formula, and encounters the same problems in the presence of strong lateral velocity variations. In these cases, the assumption that interval velocities and average velocities are linked by the simple relation in equation (10.1) is incorrect, and the use of the Dix formula may cause problems similar to the ones described in Section 10.4.3 when applied to the inversion of stacking velocities.

The schematic shown in Figure 11.19 illustrates the problems caused by the second assumption. In the Deregowski loop, the interval velocity is updated along the vertical line at the same horizontal location as the CIG used for the analysis. As is apparent from the figure, when the reflectors are dipping, this vertical line is a poor approximation of the true zero-offset rays. Furthermore, the zero-offset ray from each reflector may be distant from the zero-offset rays of the other reflectors. Consequently, average velocities measured from reflectors with different dips may sample different areas of the velocity field. When there are lateral velocity variations, the measured average-velocity errors can be inconsistent among reflectors. This inconsistency may cause problems when the measured errors in average velocities are used for interval-velocity updates. This problem could be addressed by inverting the average-velocity errors along the normal rays instead of along vertical lines. However, normal rays may cross, and the subsequent inconsistencies in the updates must be resolved by a tomographic inversion method (Audebert et al., 1997). Therefore, the simplicity of the Deregowski loop, which is one the main reasons it is widely used, is lost when normal-ray updates are employed.

The accuracy and convergence of the Deregowski loop may be improved by applying it in a **layer-stripping** fashion; that is, by estimating the velocity layer-by-layer, starting from the shallow ones and moving down in depth. The two assumptions discussed above are more valid when the vertical updating is performed only on a thin layer above the reflectors. Figure 11.20 graphically shows this concept. The vertical update assumes that the velocity perturbations are centered around the vertical ray (gridded area), while the measured effects are caused by perturbations in a wider area around the zero-offset ray (solid area). The two regions overlap significantly only in the layer just above the reflector.

In the presence of steep dips and strong lateral variations, a layer-stripping procedure must proceed slowly, with relatively thin layers, because the errors introduced by the Deregowski updating increase with the thickness of the layer. An even more substantial limitation of a layer-stripping procedure is that it estimates the velocity one layer at a time, assuming that the velocity for all the layers above is correct. Therefore, a layer velocity is estimated by using the information contained in the reflections from the interfaces just below it. In reality, the focusing and the moveout of deep reflectors are strongly influenced by shallow velocity anomalies and thus they contain valuable information on these anomalies. Layer-stripping neglects this information. Even worse, if the velocity function "frozen" in the shallow layers contains errors, these errors propagate down in the estimated velocity for all the layers below, often preventing the process from converging to a satisfactory solution.

The reader interested in examples of the application of the Deregowski loop can find several excellent case studies in Yilmaz (2001), and in numerous articles in the literature. The Deregowski loop combined with layer stripping is often used in practice because of its simplicity, and because it can be driven by the geological understanding of the interpreter. However, in complex media it has substantial drawbacks, and may require more iterations to converge than the more accurate and global approach of tomographic MVA presented in the next section.

Historically, vertical velocity-updating for prestack depth migration was defined for the wavefield-continuation methods, and, in particular, for source-receiver migration (Doherty and Claerbout, 1976; Yilmaz and Chambers, 1984; MacKay and Abma, 1993), earlier than for Kirchhoff-migration methods (Deregowski, 1990). When performing source-receiver migration, migration-velocity errors prevent the prestack wavefield from focusing at zero time and zero offset (6.15 and 6.19); however, it still focuses, at least approximately, at a time different from zero. The measurements of this focusing-time error (hence the generic name for these methods of **focusing analysis**) provide information on the average-velocity errors, similar to the information that measurements of the RMO parameter ρ provide for the Deregowski loop described in this section. The time shifts of the foci can thus be inverted into interval-velocity updates along the vertical line. These focusing-analysis methods are a valid alternative to RMO analysis when simple vertical updating is performed, However, focusing analysis does not allow tomographic updating as readily as the RMO analysis of ADCIGs described above.

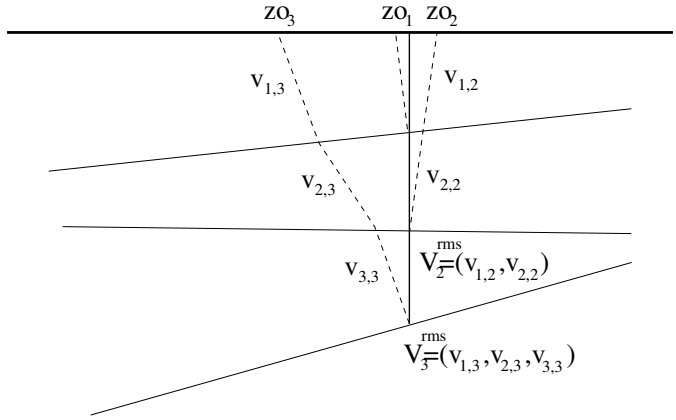


Figure 11.19: Schematic showing how the average velocities V_{rms} measured from two dipping reflectors may sample different interval velocities in the same layers, because of the different reflectors' dips. This inconsistency may cause problems when the average velocities are inverted into interval velocities along the same vertical line. migvel-deregowski-v2 [NR]

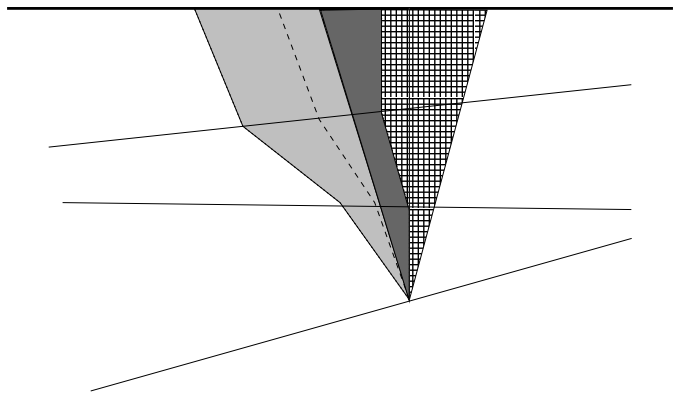


Figure 11.20: Schematic showing that the error introduced by layer stripping increases with the thickness of the layer. The amount of overlap between the area of influence around the vertical ray (gridded) and the true normal ray (solid) is higher when only the layer just above the reflector is considered than when all the layers are considered. migvel-layer-stripping-v2 [NR]

11.4 Tomographic migration velocity analysis

The simple migration velocity analysis procedure presented in the previous section may yield biased estimates, or even diverge, when the velocity function varies rapidly in the horizontal directions and the reflectors' structure is complex. To solve these challenging imaging problems we must update the migration-velocity function using a full-fledged tomographic method. Tomographic MVA methods are based on the conceptual framework that we developed in Section 10.4 for traveltime tomography. The linear tomographic operators that are inverted to estimate velocity updates are actually equivalent in the two cases (Stork, 1992). The crucial difference between MVA tomography and traveltime tomography is that in MVA tomography the traveltime errors are measured indirectly from the migrated images and not from the data. In complex areas, extracting kinematic information in the data space is practically impossible, whereas, the RMO analysis of CIG gathers is not only possible, but also robust, even if the current migration velocity is not particularly close to the true velocity.

The first formulations of tomographic MVA were based on the kinematic information extracted from ODCIGs obtained by Kirchhoff migration (van Trier, 1990; Etgen, 1990; Stork, 1992). More recently, the MVA method has been adapted to invert the velocity information extracted from ADCIGs obtained by either Kirchhoff migration (Brandsberg-Dahl et al., 1999) or wavefield-continuation migration (Biondi and Sava, 1999; Clapp and Biondi, 2000). In this section I derive the ray-based tomographic operator that is used to invert the kinematic information contained in ADCIGs; the mathematical formalism for deriving the MVA tomographic operator in the offset domain is very similar.

11.4.1 Objective function of tomographic migration velocity analysis

The goal of MVA is to improve the quality of the migrated images. In particular, we aim at flattening the migrated CIGs, or, in other words, zeroing the RMO function measured from the CIGs, $\widetilde{\mathbf{n}}_{\mathbf{RMO}}$. To achieve our goal, we minimize the following objective function:

$$J_{\text{reg}}(\mathbf{s}) = \|\widetilde{\mathbf{n}}_{\mathbf{RMO}}(\mathbf{s})\|_2 + \mathbf{s}^{'}\mathbf{C_s}^{-1}\mathbf{s}, \tag{11.19}$$

which includes a slowness regularization term based on the estimated covariance of the slowness function, $\mathbf{C_s}$. The objective function expressed in equation (11.19) is highly non-quadratic. As in the case of traveltime tomography (Section 10.4), this non-quadratic optimization problem is usually solved by applying gradient-based optimization methods (Claerbout, 2004), and thus it needs to be repeatedly linearized.

The measured RMO function must be extracted from the migrated images each time a new migration is performed. It is thus a function of the non-linear iteration index, i, and it is express as ${}_i\widetilde{\mathbf{n}}_{\mathbf{RMO}}$. In practice, ${}_i\widetilde{\mathbf{n}}_{\mathbf{RMO}}$ is measured from a set of interpreted reflectors, and it is thus defined as a function of the reflection angles (or offsets) along the picked reflectors. At each point along the reflectors, the RMO function can be directly picked from the migrated CIGs for each value of the reflection angle(s) or offset(s) (van Trier, 1990). However, a more robust way for measuring this RMO function is to use either the RMO or the residual-migration analysis

described in Section 11.2. When we apply simple RMO analysis (Section 11.2.1), the RMO function is directly determined by the ρ values picked from ρ-spectra (Etgen, 1990).

At each non-linear iteration, the tomographic operator, ${}_i\mathbf{M}$, is obtained by linearizing the relation between the RMO function and the velocity around the current velocity model, ${}_{i-1}\hat{\mathbf{s}}$, as described in Section 11.4.2. The linearized operator is used to compute the perturbations, ${}_i\widehat{\Delta\mathbf{n}}_{\mathbf{RMO}}$, in the modeled RMO function caused by perturbations, ${}_i\Delta\mathbf{s}$, in the slowness function. The estimation of the covariance of the slowness function, ${}_i\mathbf{C_s}$, should also vary with the iterations, since it ought to be based on the structural interpretation of the migrated image (Section 11.4.4).

Comparing the objective function expressed in equation (11.19) with its homologous for traveltime tomography, equation (10.5), we notice that in equation (11.19) there is no observed data vector ($\mathbf{t}_{\text{obs}}$ in traveltime tomography) as an explicit "target" for the modeled data vector ($\mathbf{t}_m$ in traveltime tomography, $\widetilde{\mathbf{n}}_{\mathbf{RMO}}$ in tomographic MVA). Although it does not explicitly appear in the objective function, the implicit target of tomographic MVA is a null RMO function. In other words, at each non-linear iteration, the objective function of ray-based MVA aims to drive the measured RMO function for the interpreted reflections toward the constant "global target" of null RMO function, regardless of whether the measured RMO function is far from being flat. At each non-linear iteration, the velocity updates are estimated by solving a quadratic optimization problem that minimizes the difference between ${}_i\widetilde{\Delta\mathbf{n}}_{\mathbf{RMO}} = \mathbf{0} - {}_i\widetilde{\mathbf{n}}_{\mathbf{RMO}}$ and the modeled perturbations, ${}_i\widehat{\Delta\mathbf{n}}_{\mathbf{RMO}}$, in the RMO function.

The objective function for the velocity updating problem at the i-th iteration is written as follows:

$$\begin{aligned} J_{\text{reg}}({}_i\Delta\mathbf{s}) &= \left\| {}_i\widehat{\Delta\mathbf{n}}_{\mathbf{RMO}} - {}_i\widetilde{\Delta\mathbf{n}}_{\mathbf{RMO}} \right\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})' \, {}_i\mathbf{C_s}^{-1} ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}) \\ &= \left\| {}_i\mathbf{M} \, {}_i\Delta\mathbf{s} - {}_i\widetilde{\Delta\mathbf{n}}_{\mathbf{RMO}} \right\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})' \, {}_i\mathbf{C_s}^{-1} ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}). \quad (11.20) \end{aligned}$$

The minimization of this objective function provides an estimate of the slowness perturbations (${}_i\Delta\mathbf{s}$) that cause the kinematics of the next prestack image (${}_{i+1}\widetilde{\mathbf{n}}_{\mathbf{RMO}}$) to be as close as possible, in a least-squares sense, to the kinematics of a "target" image that has flat CIGs.

As noted above, all the terms in the linearized objective function expressed in equation (11.20) change at each non-linear iteration. These variations can be substantial, since both the measured RMO function (${}_i\widetilde{\mathbf{n}}_{\mathbf{RMO}}$) and the estimate of the slowness covariance (${}_i\mathbf{C_s}$) result from an interpretative process. Furthermore, even the domain of definition of the objective function changes at each iteration, as reflectors are added (and subtracted) from the interpretation because they come in and out of focus. From an optimization-theory point of view, the linearized problems are probably better analyzed as semi-independent problems than as subsequent steps of an iterative solution to a non-quadratic optimization problem.

Differential Semblance Optimization (DSO) (Symes and Carazzone, 1991) is another viable approach for flattening the RMO function that, unlike the conventional MVA method described in this section, relies not on the definition of a "distant" target but only on "local" measurements of non-flatness in the CIGs. The basic principle of DSO is similar to that of conventional MVA; that is, the quality of a velocity model is judged by measuring the departure from flatness of the migrated CIGs. However, instead of minimizing the RMO of the

image at each angle (or offset) with respect to the normal-incidence (zero-offset) image, the DSO method minimizes the relative differences in imaging depth between contiguous angles (or offsets). Mulder and ten Kroode (2002) demonstrated that minimizing the DSO objective function can successfully improve the migration velocity in a real-data case. DSO has the potential to be effective with complex imaging problems, but more tests are necessary. The similarities and differences between conventional MVA and DSO are further discussed in Chapter 12 in the context of wavefield-based MVA methods, instead of ray-based methods.

11.4.2 Evaluation of the linearized tomographic-MVA operator

The linearized tomographic operator ${}_i\mathbf{M}$ can be evaluated using the chain rule as in equation (11.9). But in this case, the traveltime perturbations are functions of the perturbations in interval slowness instead of the perturbations in average slowness. These traveltime perturbations are computed by integrating slowness perturbations along the raypaths. The traveltime perturbations at normal incidence (${}_i\Delta_s\mathbf{t}_m^0$) are computed according to equation (10.14), whereas equation (10.8) is used to compute the traveltime perturbations for an arbitrary aperture angle (${}_i\Delta_s\mathbf{t}_m$). The derivative of of the modeled RMO function with respect to the slowness model is thus expressed as follows:

$$\begin{aligned}\left.\frac{\partial\widehat{\mathbf{n}_{\mathbf{RMO}}}}{\partial s}\right|_{s={}_{i-1}\hat{\mathbf{s}}} &= \left.\frac{\partial\mathbf{n}_\gamma}{\partial s}\right|_{s={}_{i-1}\hat{\mathbf{s}}} - \left.\frac{\partial\mathbf{n}_0}{\partial s}\right|_{s={}_{i-1}\hat{\mathbf{s}}} \\ &= \left.\frac{\partial\mathbf{n}_\gamma}{\partial t_m}\right|_{s={}_{i-1}\hat{\mathbf{s}}}\left.\frac{\partial t_m}{\partial s}\right|_{s={}_{i-1}\hat{\mathbf{s}}} - \left.\frac{\partial\mathbf{n}_0}{\partial t_m^0}\right|_{s={}_{i-1}\hat{\mathbf{s}}}\left.\frac{\partial t_m^0}{\partial s}\right|_{s={}_{i-1}\hat{\mathbf{s}}}. \end{aligned} \tag{11.21}$$

The derivatives of the normal shifts with respect to the traveltimes are evaluated using the relationships in equations (11.7-11.8). Substituting these equations into equation (11.21), we obtain the following expression for the RMO perturbations:

$${}_i\widehat{\Delta\mathbf{n}}_{\mathbf{RMO}} = \left(\frac{{}_i\Delta_s\mathbf{t}_m}{\cos\gamma} - {}_i\Delta_s\mathbf{t}_m^0\right)\frac{\mathbf{n}}{2\ {}_{i-1}\hat{s}}. \tag{11.22}$$

The same value for the derivative of the RMO function can be directly derived from the theory developed in Section 10.4 for traveltime tomography. We can start from the expression for the total traveltime perturbations in equation (10.7), and convert the traveltime perturbations into residual-moveout perturbations using equations (11.7). Following this path we obtain the following expression:

$${}_i\widehat{\Delta\mathbf{n}}_{\mathbf{RMO}} = \frac{{}_i\Delta t_m}{2\ {}_{i-1}\hat{s}\cos\gamma}\mathbf{n} = \frac{{}_i\Delta_s t_m - {}_i\Delta_s\mathbf{t}_m^0\cos\gamma}{2\ {}_{i-1}\hat{s}\cos\gamma}\mathbf{n}, \tag{11.23}$$

which is obviously equivalent to equation (11.22).

This result demonstrates that the tomographic operator is the same (ignoring a scaling factor $2\ {}_{i-1}\hat{s}\cos\gamma$), regardless of whether we apply tomography in the data domain or in the migrated domain. In the first case, we subtract from the traveltime perturbations the contribution of the reflector movement at normal incidence; in the second case, we directly match

the difference between the image movements at an arbitrary offset and the image movements at normal incidence. A direct consequence of the equivalence between the tomographic operators is that traveltime tomography and tomographic MVA suffer from similar problems related to the non-uniqueness of the solution, which we discussed in Section 10.4, and analyzed in Sections 10.4.3 and 10.4.4 for the simple case of the Toldi operator. Section 11.4.4 discusses methods for addressing this problem, but next we recapitulate the whole process of tomographic MVA with the help of an example.

11.4.3 Example of 3-D tomographic MVA

All the steps performed in a tomographic MVA procedure are best illustrated on a concrete case study. The data used in this example are the same North-Sea data used in the residual-migration example shown in Section 11.2.2. The results are taken from Clapp (2001). The data set is challenging because of the presence of a shallow chalk layer and of a salt body piercing through the chalk.

The first step of a tomographic MVA is to estimate a starting velocity model that is close enough to the true model that the process has the chance of converging in a limited number of iterations. In this case, the starting velocity model was estimated using a layer-based parametrization of the model (see the discussion on model parametrization in Section 11.4.4), and thus it displays sharp discontinuities at the salt and chalk boundaries, but smooth gradients inside the layers. Figure 11.21 shows three orthogonal sections cut through the starting model.

The second step is to perform a prestack depth migration and create prestack partial images. A wavefield-continuation migration is applied in this case, because of the difficulties of imaging the subsalt areas with Kirchhoff migration. The common-azimuth algorithm described in Chapter 7 is employed because of its computational efficiency that allows the performance of multiple iterations. The ADCIGs are computed before imaging, as described in Section 6.2.1.

The third step is to analyze the prestack partial images. The reflectors' geometry is determined by picking reflectors from the image cube. Figure 11.22 shows three orthogonal sections of the image cube obtained with the starting velocity. The asterisks superimposed onto the images indicate the interpreted reflectors. The RMO semblance spectra are computed and analyzed along the interpreted horizons. The twelve cubes displayed in Figure 11.23 are RMO-spectra extracted at the locations in physical space corresponding to the interpreted reflectors. The vertical axis of these cubes represents the RMO parameter, whereas the horizontal axes correspond to the image horizontal coordinates. The RMO analysis for the shallow reflectors (top-left) is sharp and clean, but, as expected, the resolution decreases as the depth of the reflectors increases. The RMO analysis becomes difficult for the bottom-of-salt reflectors (middle column, second from the top), because the actual residual moveouts in the image are substantially different from the one-parameter function used for computing the RMO spectra (Section 11.2.1).

The next step is to set up and solve the tomographic estimation problem. The reflector geometry and the aperture-angle information in the ADCIGs are used to determine the ray-

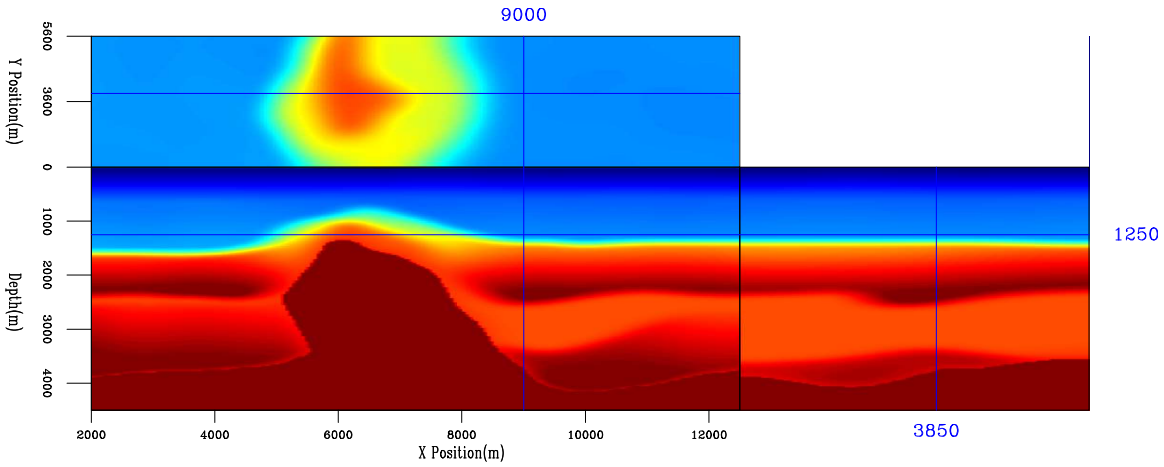


Figure 11.21: Orthogonal slices cut through the initial velocity cube (This figure is adapted from Clapp (2001).) migvel-elf3d.vel0-one-overn-col [ER]

paths corresponding to the RMO values measured from the prestack images. The velocity covariance matrix used to constraint the optimization problem [equation (11.20)] is estimated using the procedure described in Section 11.4.4. The tomographic MVA operator is evaluated according to either equation (11.22) or equation (11.23). The solution of the optimization problem yields the velocity function shown in Figure 11.24. This new velocity function is used for a new depth migration, and the process is repeated until a satisfactory result is obtained.

The velocity cube obtained after two iterations is displayed in Figure 11.25. Comparing this "final" velocity with the initial velocity (Figure 11.21), we can notice differences in the interpretation of the salt boundaries, as well as in the layers velocities, in particular to the right of the salt body. The chalk layer and the sediments below it are significantly slower in the new velocity model than in the initial one.

These changes in velocity function substantially improve the migrated results, as can be appreciated by comparing the orthogonal slices shown in Figure 11.26 (initial velocity) and Figure 11.27 (final velocity). These slices are cut through the image cube at the same locations as the slices cut through the velocity cubes shown in the previous figures. The most noticeable differences between the images are highlighted by ellipses. The sediments below the salt edge (ellipse A) are coherently imaged further in the direction of the salt in Figure 11.27 than in Figure 11.26. The lowering of the velocity in the chalk, and in the layer below it, leads to a substantial improvement in the image of the two faults visible in the cross-line section (ellipse D), and of the synclinal structure highlighted by ellipse C.

11.4.4 Improving the convergence of tomographic MVA

Tomographic MVA is applied to the most challenging imaging project, where the difficulties caused by the non-unique and non-linear nature of the velocity-estimation problem are magnified and they may seriously impact the final success of the method. The non-uniqueness of the

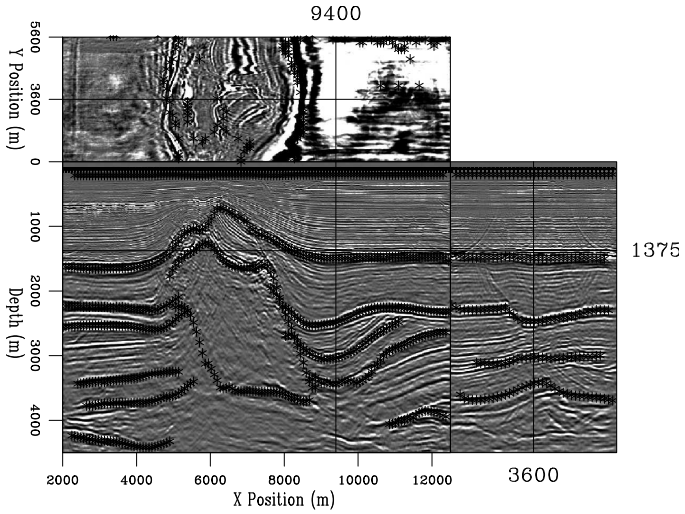


Figure 11.22: Orthogonal slices cut through the migrated image obtained using the starting velocity. The asterisks superimposed onto the image indicate the interpreted reflectors. (This figure is from Clapp (2001).) migvel-elf3d.reflectors.vel0 [ER]

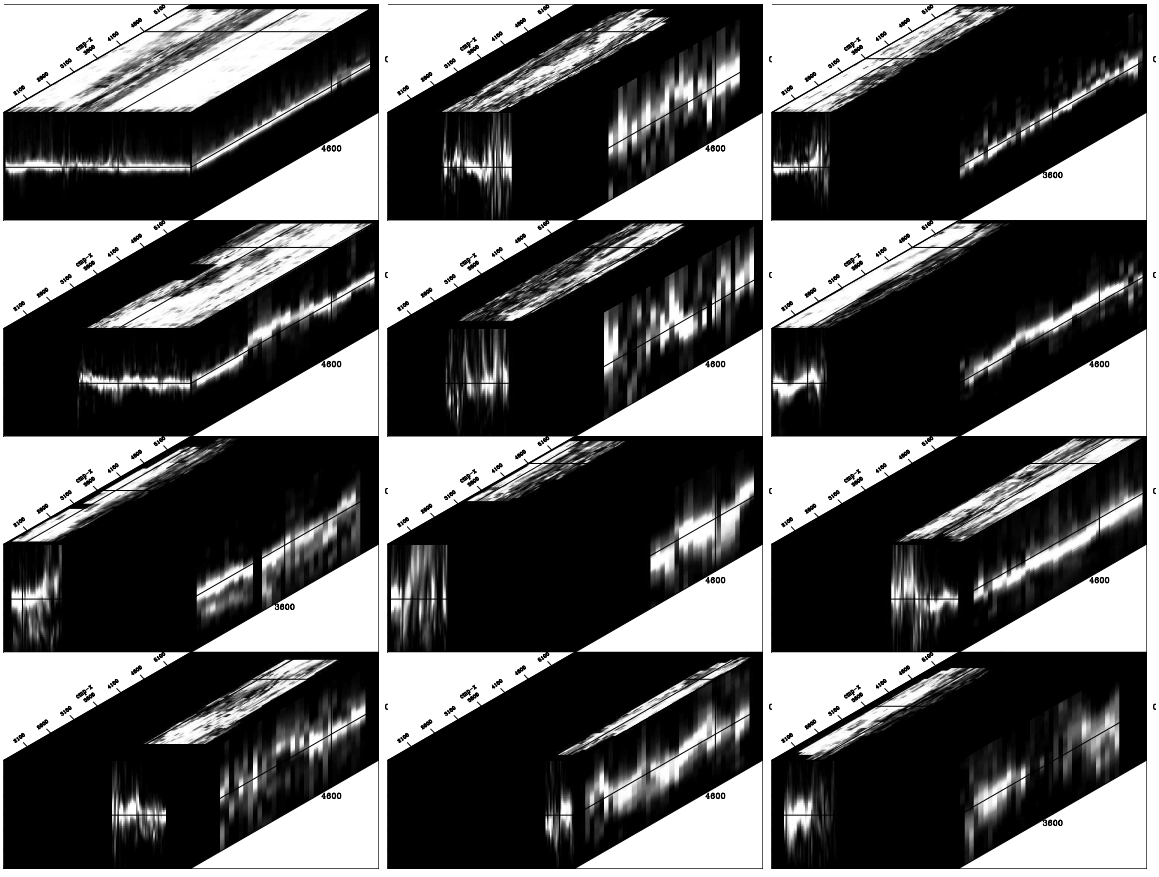


Figure 11.23: RMO semblance-analysis cubes for the interpreted reflectors. The vertical axis of these cubes represents the RMO parameter, whereas the horizontal axes correspond to the image horizontal coordinates. (This figure is from Clapp (2001)) migvel-sem-ref.3d-overn [ER]

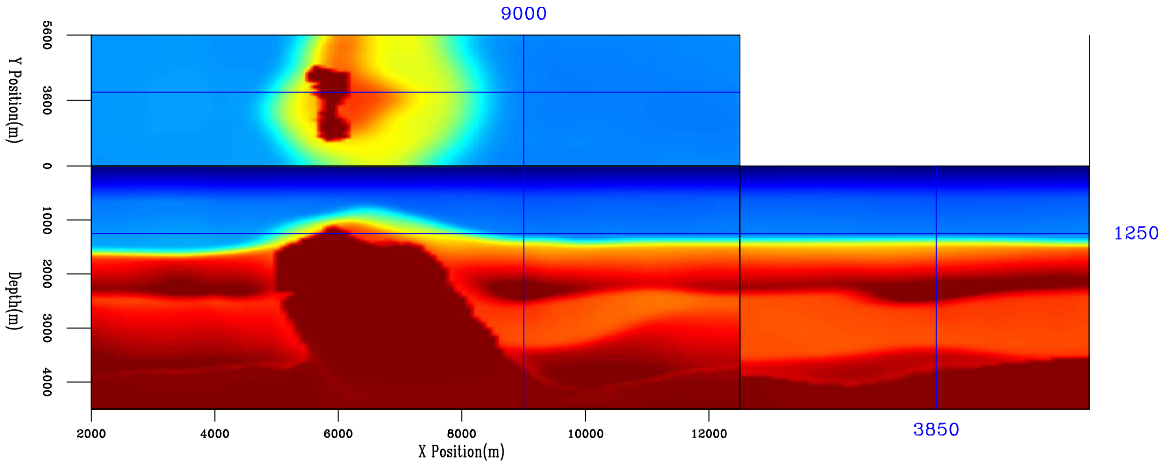


Figure 11.24: Orthogonal slices cut through the velocity cube after one iteration of the MVA tomographic updating. (This figure is adapted from Clapp (2001).) migvel-elf3d.vel1-one-overn-col [ER]

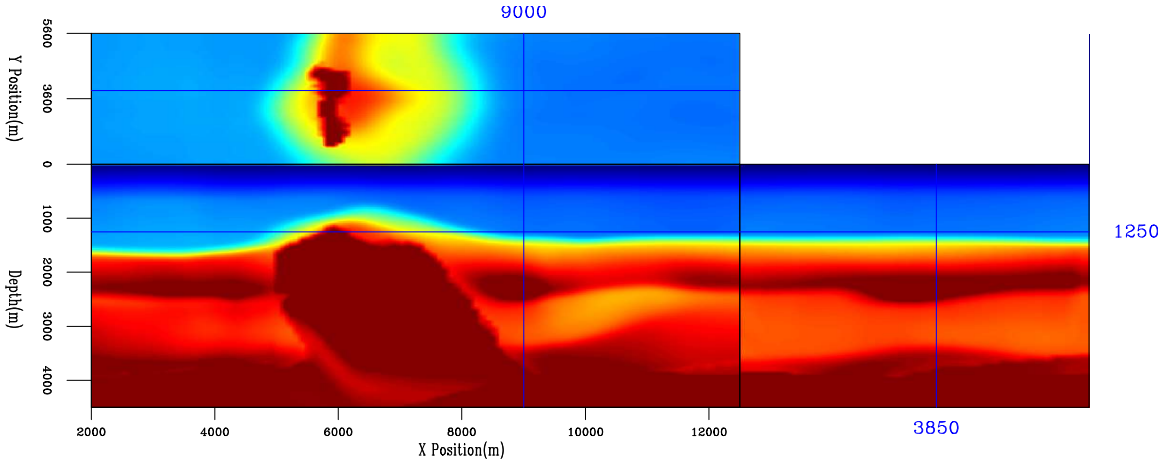


Figure 11.25: Orthogonal slices cut through the velocity cube after two iterations of the MVA tomographic updating. (This figure is adapted from Clapp (2001).) migvel-elf3d.vel2-one-overn-col [ER]

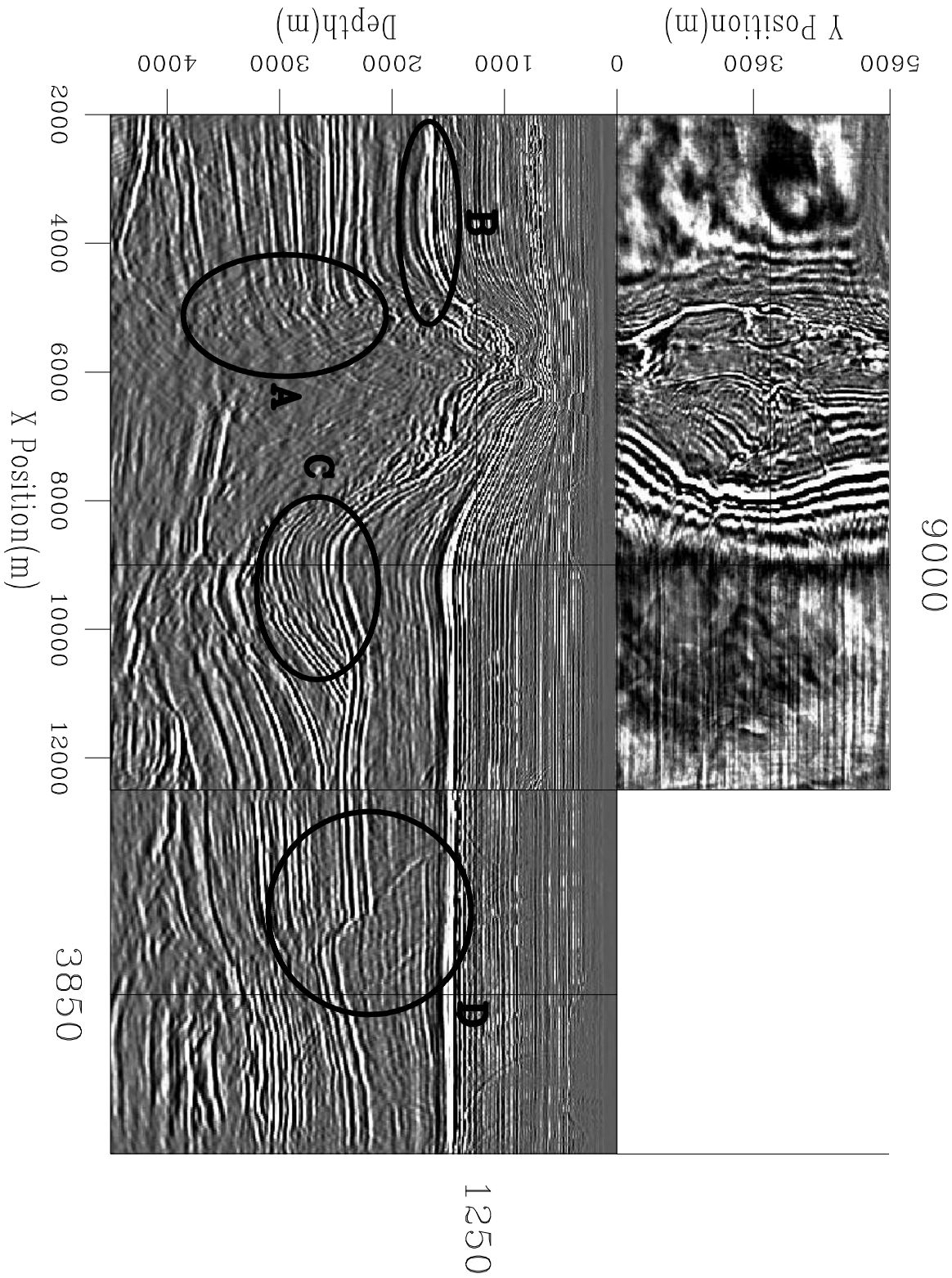


Figure 11.26: Orthogonal slices cut through the migrated image obtained using the starting velocity. (This figure is from Clapp (2001).) migvel-cube.mig0.1-overn [ER]

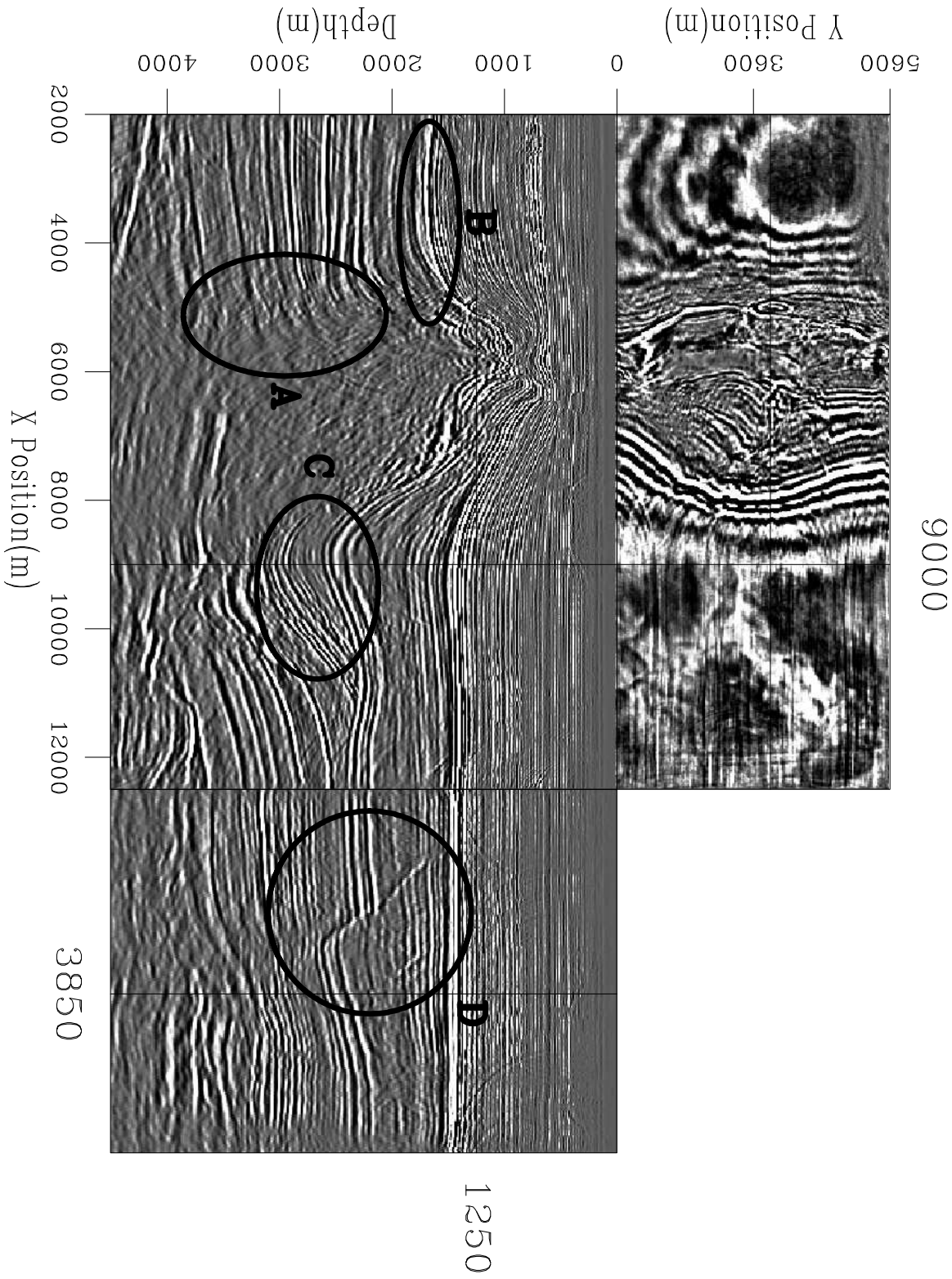


Figure 11.27: Orthogonal slices cut through the migrated image obtained using the velocity function obtained after two iterations of the MVA tomographic updating. (This figure is from Clapp (2001).) migvel-cube.mig2.1-overn [ER]

MVA solution causes poor local convergence during the solution of the linearized problems encountered at each non-linear iteration. The non-linearity of the relationship between the migrated images and the velocity model creates multiple local minima in the objective function. Global convergence to the optimal solution is not assured, and it is strongly dependent on the starting model. The two problems are interrelated, but for the sake of simplicity, I will discuss them separately.

Improving the local convergence of tomographic MVA

The velocity perturbations are not uniquely defined by the available data because several velocity functions yield migrated CIGs that are equally flat, at least in a least-squares sense. This problem could be addressed either by making better use of the information in the seismic data or by integrating additional information into the inversion process. For example, to make a better use of the information contained in the seismic data we could analyze the focusing of the image in the physical space (Sava et al., 2004), in addition to the flatness of the CIGs. Or we could update the velocity function by using a wavefield-continuation operator that is capable of exploiting the frequency dispersion caused by complex wave propagation (Chapter 12).

In this section, I discuss more conventional methods that use geological information to better constrain the linearized optimization problem. These **geological constraints** are designed to guide the inversion process toward solutions that satisfy a geological model of how the velocity should vary between layers and within each layer. There are two main methods for guiding the velocity-estimation process toward geologically plausible solutions: 1) use a parsimonious parametrization of the velocity function that, by construction, allows only geologically plausible solutions, 2) define the regularization term in the objective function of equation (11.20) to privilege velocity functions that are consistent with our geological model. These two methods are not mutually exclusive, and can be used in combination.

The parametrization of the velocity model by discrete layers of constant velocity is natural, and has been employed since the beginning of the development of velocity-analysis techniques. It has often been described as the **velocity macro model** strategy (van de Made et al., 1984; Wapenaar and Berkhout, 1985). The constant-velocity-layers parametrization can be easily generalized to non-constant velocity layers, with a polynomial defining the velocity function within the layers. The macro-model approach has the advantage of being simple and requiring very few parameters to define the velocity function; the solution is thus tightly constrained and, thanks to the small number of parameters used, requires few iterations to converge. In areas where the underlying assumptions of a strong correlation between lithology and velocity are appropriate (e.g. North Sea), the macro-model approach has been successful. Its main disadvantage is its lack of “flexibility”. Because the parametrization is fixed *a priori*, it does not automatically adapt to situations where “optimal” focusing of the data requires velocity variations that are more complex than the given parametrization allows. When there are unexpected velocity variations, the solution defined with a macro-model approach may completely miss them. In other words, the macro-model approach does not allows for surprises whereas, the real world is full of them.

The parametrization of the velocity model by cubic splines is a parsimonious velocity representation that shares with the macro-model approach the advantage of requiring few parameters, but is more flexible where the data require (Biondi, 1990; van Trier, 1990; Etgen, 1990). The simplest way of using cubic splines is to use use them globally; that is, to parametrize the whole velocity model as one single set of cubic splines. When this approach is clearly insufficient, like when representing velocity functions with known sharp discontinuities (e.g salt and sediments), the cubic-spline approach can be combined with the macro-model approach and a set of spline functions for each macro-layer can be used (Perdrizet and Sinoquet, 2003).

When we use a dense grid of cubic-spline nodes, the cubic-splines approach behaves similarly to a straightforward gridded representation of the velocity model. In this approach the velocity model is defined on a dense Cartesian grid, and the task of stabilizing the inversion is entirely left to the regularization term ($\mathbf{C_s}$) in the objective function [equation (11.20)]. In Bayesian estimation theory, the matrix $\mathbf{C_s}$ is interpreted as the *a priori* covariance matrix of the model parameters (Tarantola, 1987). A common assumption is that the velocity model is smooth, and consequently $\mathbf{C_s}^{-1}$ is set to be an isotropic roughener operator, like a Laplacian operator (Toldi, 1985). However, the choice of an isotropic roughener is driven more by the knowledge of the null space of the tomographic operator, and by the goal of assuring stability to the inversion procedure, than by geological considerations.

Geological information can be introduced in the regularization operator by making it directional; that is, by rewarding velocity functions that are smooth along some particular directions. Because velocity is often driven by a combination of compaction and sedimentation, the smoothing directions are chosen according to the geological dips and/or the known compaction direction. The important advantage of directional regularization operators is that we let geology (e.g. the structural image together with its geological interpretation) drive the determination of $\mathbf{C_s}$. Therefore, the final model is likely to satisfy the geologist, in addition to satisfying the geophysicist (e.g. fitting the seismic data). Because the geological interpretation may vary with the non-linear iterations, as reflectors come in and out of focus, the estimate of the covariance matrix also changes with iterations, and I thus add the left subscript i to the notation; i.e., I write ${}_i\mathbf{C_s}$.

Two methods have been proposed to introduce a directivity in the regularization term: 1) penalize the dot product of the velocity gradient with the smoothing direction (Sinoquet, 1993), 2) use non-stationary **steering filters** aligned along the smoothing direction (Clapp, 2001; Clapp et al., 2004). The non-stationarity of the directional regularization operators is essential when imaging complex geological structure with rapidly varying geological dips. It also makes possible the modeling of constant velocity layers (e.g. salt bodies or water columns); a task that is easily accomplished with the macro-model approach.

The ability to guide the smoothing of a gridded velocity model according to the structural image gives the gridded-model approach some of the attractive characteristics of the macro-model approach. However, the simplistic approach to smoothing regularization formalized in the objective function [equation (11.20)] is likely to slow the convergence of the optimization problem.

When we penalize roughness in the velocity functions by using a roughener operator such

as the inverse of the model covariance matrix, ${}_i\mathbf{C_s}^{-1}$, the resulting quadratic optimization problem is severely ill-conditioned, because the matrix ${}_i\mathbf{C_s}^{-1}$ itself is ill-conditioned (Golub and Loan, 1983). The ill-conditioning is caused by the widely variable scaling of the spatial wavenumbers performed by ${}_i\mathbf{C_s}^{-1}$; the low wavenumbers are strongly attenuated and, at the limit, the constant component is in the null space of ${}_i\mathbf{C_s}^{-1}$. An alternative, and more intuitive, way of describing this problem, is that the regularization term is a short operator. Therefore, a conjugate-gradient method (Claerbout, 2004) requires many iterations to apply the regularization term (and its adjoint) a sufficient number of times to spread velocity perturbations that are localized in space.

The slow-convergence problem can be solved by preconditioning the optimization problem with a change of variable in the model parameters (Harlan, 1995). If we know how to factor an approximate inverse of ${}_i\mathbf{C_s}^{-1}$, such that ${}_i\mathbf{C_s} \approx {}_i\mathbf{C_p} = {}_i\mathbf{F_p^T}\,{}_i\mathbf{F_p}$, we can apply the change of variables from ${}_i\Delta\mathbf{s} = {}_i\mathbf{F_p}\,{}_i\Delta\mathbf{p}$. Equation (11.20) then becomes,

$$J_{\text{prec}}({}_i\Delta\mathbf{p}) = \left\| {}_i\mathbf{M}\,{}_i\mathbf{F_p}\,{}_i\Delta\mathbf{p} - {}_i\widetilde{\Delta\mathbf{n}}_{\mathbf{RMO}} \right\|_2 + \left({}_{i-1}\hat{\mathbf{s}} + {}_i\mathbf{F_p}\,{}_i\Delta\mathbf{p}\right)' {}_i\mathbf{C_s}^{-1}\left({}_{i-1}\hat{\mathbf{s}} + {}_i\mathbf{F_p}\,{}_i\Delta\mathbf{p}\right). \tag{11.24}$$

After the change of variables, the optimization problem expressed in equation (11.24) is well-conditioned, since ${}_i\mathbf{F_p^T}\,{}_i\mathbf{C_s}\,{}_i\mathbf{F_p} \approx \mathbf{I}$. Once the vector of parameters ${}_i\widetilde{\Delta\mathbf{p}}$ is estimated by using equation (11.24), the final solution in the original space can be be easily recovered as ${}_i\widetilde{\Delta\mathbf{s}} = {}_i\mathbf{F_p}\,{}_i\widetilde{\Delta\mathbf{p}}$. The speed-up in convergence is achieved even when the factorized matrix ${}_i\mathbf{C_p}$ is not the exact inverse. However, to avoid boundary artifacts, ${}_i\mathbf{C_p}$ should be an exact inverse of ${}_i\mathbf{C_s}^{-1}$ at the boundaries (Clapp, 2001), although it may be only an approximation in the interior of the velocity model.

Clapp (2001) describes an efficient numerical algorithm for computing the factors ${}_i\mathbf{F_p}$ of the exact inverse of ${}_i\mathbf{C_s}^{-1}$, when ${}_i\mathbf{C_s}^{-1}$ is defined in terms of steering filters. Meyerholtz et al. (1989) describe an analogous methodology based on convolutional quelling that has the same computational advantages of the Clapp's steering filter methodology, but it is applied to isotropic smoothing instead of to directional smoothing. The possibility of preconditioning the regularization term enables the linearized problem to converge quickly when solved with a conjugate-gradient algorithm. The speed-up of the convergence decreases the computational cost of velocity updating in tomographic MVA, by reducing the number of times the linearized tomographic operator $\mathbf{M}$ is applied.

The combination of directional regularization and preconditioning makes the gridded-velocity approach as capable of including structural-geology information, and as computationally effective, as the macro-model approach. Given the flexibility of a gridded velocity model to adapt to expected and unexpected velocity variations, it is being used in place of, or in combination with, the macro-model approach in an increasing number of imaging projects.

The gain in computational efficiency achieved by preconditioning the regularization term is useful when the velocity updating is performed with ray-operators, as for the method described in this section. However, for complex imaging projects, the relative cost of applying a ray-based tomographic operator is constantly decreasing, compared with the cost of performing expensive wavefield-continuation migrations. Indeed, for these complex imaging problems, we would like to update the velocity model by using full-wavefield operators, as

discussed in (Chapter 12). When using wavefield operators to update the model, reducing the number of iterations by preconditioning becomes essential.

Improving the global convergence of tomographic MVA

The issue of assuring global convergence of the velocity-estimation problem is even more challenging than the improvement of the local convergence. Solutions are less well-developed and they tend to be based more on heuristic considerations rather than on firm theoretical understandings. At the present, the main discriminants between local minima in the objective function are geological criteria used by the interpreters to decide whether the solutions provided by the estimation process make geological sense. However, there are lot of incentives not to rely exclusively on the subjectivity of human judgment. One of the still unexplored areas of seismic-imaging technology is the inclusion of quantitative geological constraints to guide global convergence in the velocity-estimation process, in addition to the directional-smoothing operators described in this section.

There are two main approaches for improving global convergence. The first is to apply a robust optimization algorithm for minimizing the objective function defined in equation (11.19). The hope is that a more robust optimization algorithm should be able to avoid local minima and find the global minimum. The second approach is to modify the problem by defining a new objective function that is more quadratic in the global sense; that is, it has fewer local minima than the objective function defined in equation (11.19).

In theory, we always have the option of using a global optimization algorithm to solve a non-quadratic optimization problem (i.e a genetic algorithm (Holland, 1975) or simulated annealing (Kirkpatrick et al., 1983)). However, these algorithms require too many (thousands or more) evaluations of the forward modeling operator to be even close to practical in 3-D velocity-estimation problems (Jin and Madariaga, 1994; Jervis et al., 1996). One heuristic approach that improves the chances of avoiding local minima is to start from a simple low-resolution model and let complexities enter the velocity function only gradually, over several non-linear iterations (Biondi, 1990). The rationale for this approach is that by adding model complexities when the current model is far from the true model, we risk locking the estimation process into the neighbor of a local minima and never being able to escape. This argument is based on the nature of the non-linearities of the relationship between kinematics in the data and the velocity model.

A modification of the conventional tomographic MVA problem that is proving successful for improving global convergence is the definition of the interval velocity model as a function of the two-way traveltime τ instead of depth (Kosloff et al., 1996; Biondi et al., 1998; Clapp, 2001; Alkhalifah, 2003). This modification does not require a change in the structure of the objective function, but only in the domain of definition of the velocity function and the reflectors' geometry. The tomographic velocity updating is based on the information measured at the reflectors, and the mapping of the reflectors in depth is a function of the current migration velocity. Consequently, the locations of the velocity updates are also functions of the current velocity. Ideally, we would like to remap the velocity updates in depth using the

mapping function defined by the updated velocity. This remapping introduces a non-linear relation between the velocity function and the kinematic errors measured from the data (or the image), and thus cannot be captured by a linearized tomographic operator defined in the depth domain. In contrast, by defining both the reflectors and the velocity in the τ domain instead of the depth domain, we substantially reduce the dependency of the reflector geometry on the current velocity function and remove the need for the remapping procedure just described. The vertical updating process described in Section 11.3 is a simple situation where the advantage of performing the updates in the τ domain is evident. In that case, the updating of the interval-velocity function is performed in the τ domain, after transforming the measured average-velocity errors into the τ domain from the depth domain. The interval-velocity function is mapped back to depth for a new migration only as the last step, and using the mapping determined by the new velocity.

The velocity updating in the τ domain does not involve a change in the structure of objective function for tomographic MVA. In contrast, the (DSO) method (Symes and Carazzone, 1991) defines a (slightly) different objective function from the one optimized by conventional MVA methods (see Section 11.4.1, for a brief discussion of the DSO objective function). Because of the differential nature of its objective function, DSO has the potential of avoiding some of the local minima caused by inaccurate interpretation of the migrated image. Chauris and Noble (1998) demonstrated that the DSO functional is convex in fairly simple cases. Mulder and ten Kroode (2002) show that coherent noise (e.g. multiple reflections) can cause the presence of local minima in the DSO functional, but global convergence can be achieved if the effect of the multiple reflections is removed from the migrated CIGs.

Example of improved convergence of tomographic MVA

In the following example, I illustrate two of the concepts introduced in this section for improving the convergence of tomographic MVA. I compare two results of one velocity-updating iteration on a 2-D slice of the 3-D North Sea data set shown in Section 11.4.3. The two results were obtained as follows: 1) with the velocity model defined in the depth domain and using a conventional isotropic regularization operator (e.g a Laplacian operator), and 2) with the velocity model defined in the τ domain and using a directional regularization operator (e.g steering filters). In both cases, the regularization term was preconditioned; that is, the objective function in equation (11.24) was minimized, instead of the objective function in equation (11.20).

Figure 11.28 shows the starting velocity model. It is a smooth velocity function, with the addition of a salt body interpreted from a preliminary migration of the data. Figure 11.31 shows the migrated image obtained using the initial model. Figure 11.29 shows the velocity updated using the conventional MVA methods of defining the velocity function in depth and using an isotropic regularization operator. In contrast, Figure 11.30 shows the velocity updated using two of the methods discussed in this section for improving the convergence of tomographic MVA. A directional regularization operator was used in the objective function, and the velocity model was defined in the τ domain; notice that for the sake of comparison all the figures are displayed in the depth domain, even when they were computed in the τ

domain. The regularization directions were determined in the τ domain by a dip analysis of the migrated image in Figure 11.31, mapped to the τ domain (Clapp, 2001). The benefits of the directional regularization operator on the velocity updating are evident when comparing the two velocity functions. In particular, both the chalk layer on the top of the salt body and the synclinal structure to the right of the salt body are better represented in Figure 11.30 than in Figure 11.29.

Figures 11.32 and 11.33 show the migrated images obtained by using the velocity models shown in Figures 11.29 and 11.30, respectively. The bottom of the salt reflector is better imaged when using the methodologies described in this section (Figure 11.33), and the salt sediments under the edge of the salt are more coherent. Furthermore, the focusing of the synclinal structure to the right of the salt body has also improved. Notice that 2-D migration and MVA have inherent limitations in this area, because the subsurface geology has many 3-D features.

REFERENCES

Al-Yahya, K. M., 1989, Velocity analysis by iterative profile migration: Geophysics, **54**, 718–729.

Alkhalifah, T., 2003, Tau migration and velocity analysis: Theory and synthetic examples: Geophysics, **68**, 1331–1339.

Audebert, F., Diet, J. P., and Zhang, X., 1996, CRP-scans from 3-D pre-stack depth migration: A powerful combination of CRP-gathers and velocity scans: 66th Ann. Internat. Meeting, Soc. of Expl. Geophys., 515–518.

Audebert, F., Diet, J. P., Guillaume, P., Jones, I. F., and Zhang, X., 1997, CRP-scans: 3-D preSDM velocity analysis via zero-offset tomographic inversion: 67th Ann. Internat. Meeting, Soc. of Expl. Geophys., 1805–1808.

Biondi, B., and Sava, P., 1999, Wave-equation migration velocity analysis: 69th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 1723–1726.

Biondi, B., and Symes, W. W., 2003, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics: accepted for publication.

Biondi, B., and Tisserant, T., 2004, 3-D angle-domain common-image gathers for migration velocity analysis: Geophisical Prospecting: submitted for publication.

Biondi, B. L., Clapp, R. G., Fomel, S. B., and Alkhalifah, T. A., 1998, Robust reflection tomography in the time domain: 68th Ann. Internat. Meeting, Soc. of Expl. Geophys., 1847–1850.

Biondi, B., 1990, Velocity analysis using beam stacks: Ph.D. thesis, Stanford University.

Biondi, B., 2003, Amplitude balancing of 3-D angle-domain common-image gathers: SEP–**114**, 45–56.

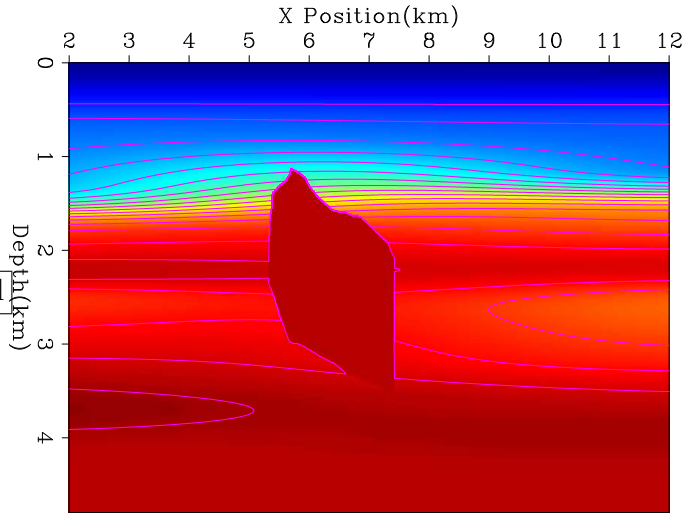


Figure 11.28: Initial velocity model. (This figure is from Clapp (2001).) migvel-vel-smooth-init-salt-overn-col [ER]

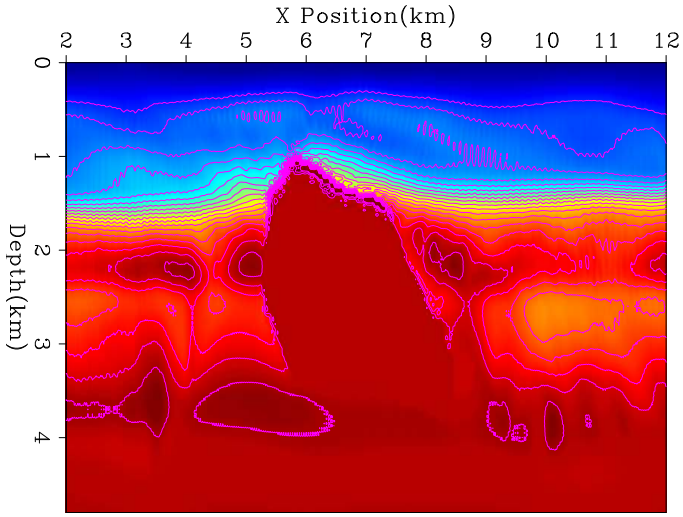


Figure 11.29: Velocity model after velocity-updating in the depth domain with an isotropic (Laplacian) regularization operator. (This figure is adapted from Clapp (2001).) migvel-vel-smooth-depth-overn-col [ER]

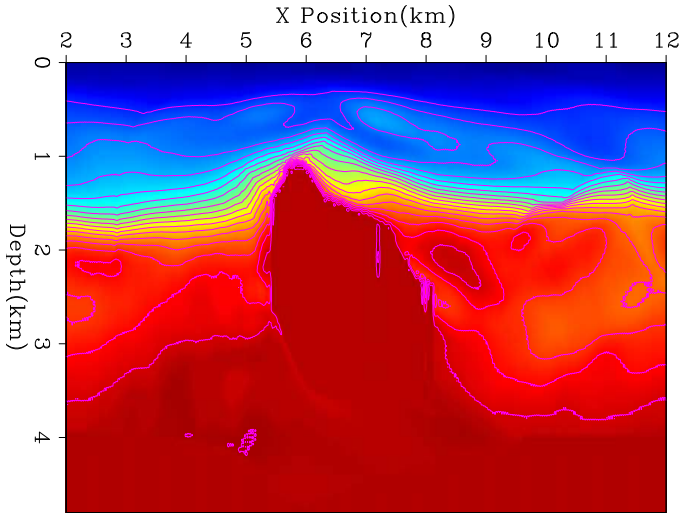


Figure 11.30: Velocity model after velocity-updating in the τ domain with a directional (steering filter) regularization operator. (This figure is adapted from Clapp (2001).) migvel-vel-smooth-tau-overn-col [ER]

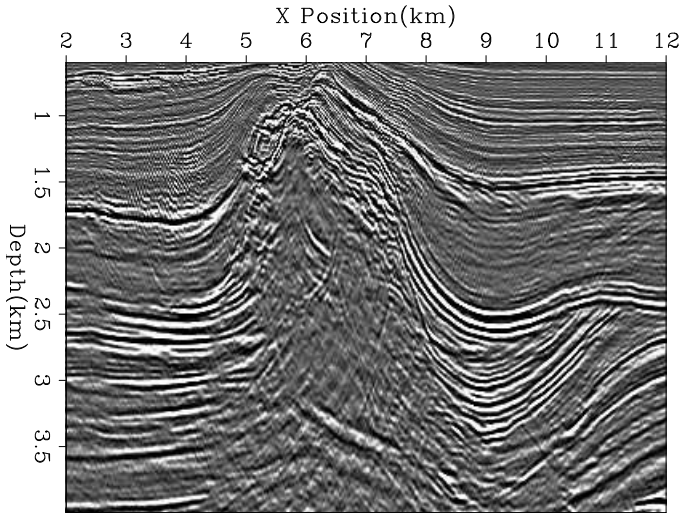


Figure 11.31: Depth migration using the initial velocity model shown in Figure 11.28. (This figure is adapted from Clapp (2001).) migvel-elf-mig-smooth-init-overn [ER]

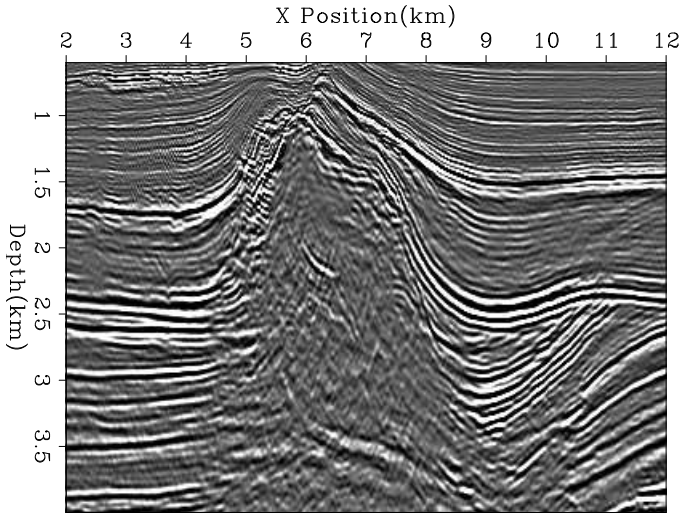


Figure 11.32: Depth migration using the velocity model shown in Figure 11.29, which was obtained by velocity-updating in the depth domain with an isotropic (Laplacian) regularization operator. (This figure is adapted from Clapp (2001).) migvel-elf-mig-smooth-depth-overn [ER]

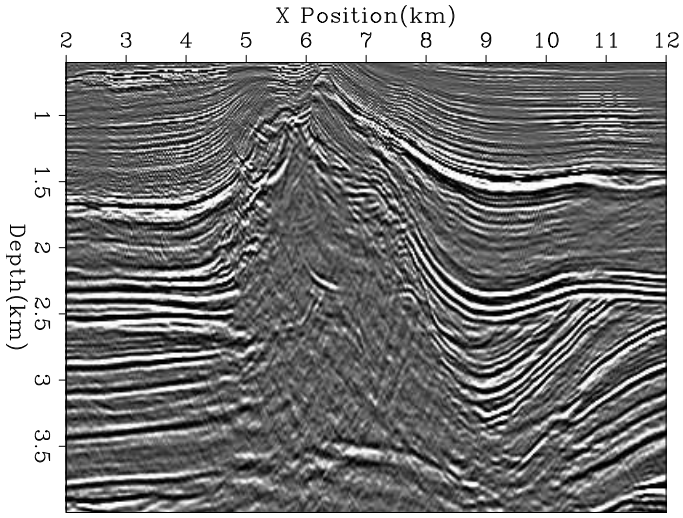


Figure 11.33: Depth migration using the velocity model shown in Figure 11.30, which was obtained by velocity-updating in the τ domain with a directional (steering filter) regularization operator. (This figure is adapted from Clapp (2001).) migvel-elf-mig-smooth-tau-overn [ER]

Bishop, T. N., Bube, K. P., Cutler, R. T., Langan, R. T., Love, P. L., Resnick, J. R., Shuey, R. T., Spindler, D. A., and Wyld, H. W., 1985, Tomographic determination of velocity and depth in laterally varying media: Geophysics, **50**, 903–923.

Brandsberg-Dahl, S., de Hoop, M., and Ursin, B., 1999, Velocity analysis in the common scattering-angle/azimuth domain: 69th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1715–1718.

Chauris, H., and Noble, M. S., 1998, Testing the behavior of differential semblance for velocity estimation: 68th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 1305–1308.

Claerbout, J. F., 2004, Image Estimation by Example: `http://sepwww.stanford.edu/sep/prof/index.html`.

Clapp, R., and Biondi, B., 2000, Tau domain migration velocity analysis using angle CRP gathers and geologic constrains: 70th Ann. Internat. Meeting, Soc. of Expl. Geophys., 926–929.

Clapp, R. G., Biondi, B., and Claerbout, J. F., 2004, Incorporating geologic information in reflection tomography: Geophysics, **69**, 533–546.

Clapp, R. G., 2001, Geologically constrained migration velocity analysis: Ph.D. thesis, Stanford University.

Deregowski, S. M., 1990, Common-offset migrations and velocity analysis: First Break, **8**, no. 6, 224–234.

Doherty, S. M., and Claerbout, J. F., 1976, Structure independent velocity estimation: Geophysics, **41**, 850–881.

Etgen, J., 1990, Residual prestack migration and interval velocity estimation: Ph.D. thesis, Stanford University.

Fomel, S., 2003, Velocity continuation and the anatomy of residual prestack time migration: Geophysics, **68**, 1650–1661.

Gardner, G. H. F., French, W. S., and Matzuk, T., 1974, Elements of migration and velocity analysis: Geophysics, **39**, 811–825.

Golub, G. H., and Loan, C. F. V., 1983, Matrix computations: John Hopkins University Press.

Harlan, W. S., 1995, Regularization by model reparameterization: `http://www.billharlan.com/pub/papers/`.

Holland, J., 1975, Adaptation in natural and artificial systems: University of Michigan Press.

Jervis, M., Sen, M. K., and Stoffa, P. L., 1996, Prestack migration velocity estimation using nonlinear methods: Geophysics, **61**, 138–150.

Jiao, J., Stoffa, P. L., Sen, M. K., and Seifoullaev, R. K., 2002, Residual migration-velocity analysis in the plane-wave domain: Geophysics, **67**, 1252–1269.

Jin, S., and Madariaga, R., 1994, Nonlinear velocity inversion by a two-step monte carlo method: Geophysics, **59**, 577–590.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., 1983, Optimization by simulated annealing: Science, **220**, 671–680.

Kosloff, D., Sherwood, J., Koren, Z., MacHet, E., and Falkovitz, Y., 1996, Velocity and interface depth determination by tomography of depth migrated gathers: Geophysics, **61**, 1511–1523.

Lafond, C. F., and Levander, A. R., 1993, Migration moveout analysis and depth focusing: Geophysics, **58**, 91–100.

Liu, Z., and Bleistein, N., 1995, Migration velocity analysis: Theory and an iterative algorithm: Geophysics, **60**, 142–153.

MacKay, S., and Abma, R., 1993, Depth-focusing analysis using a wavefront-curvature criterion: Geophysics, **58**, 1148–1156.

Meng, Z., and Bleistein, N., 2001, On velocity/depth ambiguity in 3-D migration velocity analysis (short note): Geophysics, **66**, 256–260.

Meyerholtz, K. A., Pavlis, G. L., and Szpakowski, S. A., 1989, Convolutional quelling in seismic tomography: Geophysics, **54**, 570–580.

Mosher, C., Jin, S., and Foster, D., 2001, Migration velocity analysis using common angle image gathers: 71th Ann. Internat. Mtg., Soc. of Expl. Geophys., 889–892.

Mulder, W. A., and ten Kroode, A. P. E., 2002, Automatic velocity analysis by differential semblance optimization: Geophysics, **67**, 1184–1191.

Ottolini, R., and Claerbout, J. F., 1984, The migration of common-midpoint slant stacks: Geophysics, **49**, 237–249.

Perdrizet, T., and Sinoquet, D., 2003, Adaptive model parameterization designed for reflection tomography: 73rd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 674–677.

Rocca, F., and Salvador, L., 1982, Residual migration: 52nd Ann. Internat. Meeting, Soc. of Expl. Geophys., Session:S1.4.

Rothman, D. H., Levin, S. A., and Rocca, F., 1985, Residual migration - applications and limitations: Geophysics, **50**, 110–126.

Sattlegger, J. W., 1975, Migration velocity determination - Part I: Philosophy: Geophysics, **40**, 1–5.

Sava, P., Biondi, B., and Etgen, J., 2004, Diffraction-focusing migration velocity analysis with application to seismic and GPR data: Geophysics: submitted for publication.

Sava, P. C., 2003, Prestack residual migration in frequency domain: Geophysics, **68**, 634–640.

Sinoquet, D., 1993, Modeling a priori information on the velocity field in reflection tomography: 63rd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 591–594.

Stolt, R. H., 1978, Migration by Fourier transform: Geophysics, **43**, 23–48.

Stolt, R. H., 1996, Short note - A prestack residual time migration operator: Geophysics, **61**, 605–607.

Stork, C., 1992, Reflection tomography in the postmigrated domain: Geophysics, **57**, 680–692.

Symes, W. W., and Carazzone, J. J., 1991, Velocity inversion by differential semblance optimization: Geophysics, **56**, 654–663.

Tarantola, A., 1987, Inverse problem theory: Elsevier. Methods for Data Fitting and Model Parameter Estimation.

Toldi, J., 1985, Velocity analysis without picking: Ph.D. thesis, Stanford University.

van de Made, P. M., van Riel, P., and Berkhout, A. J., 1984, Velocity and subsurface geometry inversion by a parameter estimation in complex inhomogeneous media: 54th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, Session:S1.5.

van Trier, J., 1990, Tomographic determination of structural velocities from depth migrated seismic data: Ph.D. thesis, Stanford University.

Wapenaar, C. P. A., and Berkhout, A. J., 1985, Velocity determination in layered systems with arbitrarily curved interfaces by means of wave-field extrapolation of common-midpoint data: Geophysics, **50**, 63–76.

Whitmore, N. D., and Garing, J. D., 1993, Interval velocity estimation using iterative prestack depth migration in the constant angle domain: The Leading Edge, **12**, 757–762.

Yilmaz, O., and Chambers, R. E., 1984, Migration velocity analysis by wave-field extrapolation: Geophysics, **49**, 1664–1674.

Yilmaz, O., 2001, Seismic data analysis, *in* CooperM. R.; Doherty, S. M., Ed., Seismic Data Analysis Vol. 2: Soc. of Expl. Geophys., 02, 1001–2027.

Chapter 12

Migration Velocity Analysis by wavefield methods

Every velocity estimation method presented in Chapters 10 and 11 is based, explicitly or implicitly, on a ray-tracing modeling of the kinematics of the reflections. When estimating velocity, a ray approximation is convenient for several reasons. First, it is faster to trace rays than propagating waves, and thus an inversion method based on rays has a large computational advantage. An even more important advantage is the intuitive nature of the link that rays establish between the velocity function and the kinematics of the reflections.

Ray-based velocity-estimation methods, however, have serious drawbacks for complex depth imaging problems. In Chapter 11 we have seen that when the subsurface geology is structurally complex, the velocity estimation and the migration steps are interdependent, and velocity estimation and migration are applied iteratively in a Migration Velocity Analysis (MVA) procedure. In the presence of complex wave propagation (i.e. multipathing and geometrical dispersion), ray-based migration methods are often not capable of producing high-quality images, while wavefield-continuation methods, such as the ones introduced in Chapter 4, yield better images. To ensure that the MVA iterative imaging process converges to a satisfactory model, it is crucial that the migration and the velocity estimation are as consistent with each other as possible. Consistency with migration is an important attribute for velocity-estimation methods, because the final goal is to improve the quality of the migrated image. When wavefield-continuation migration is required, we should consider using velocity-estimation methods that go beyond the asymptotic approximations required by ray-based methods.

Imaging under rugged salt bodies is an important case where ray-based MVA methods are not reliable. An important practical difficulty encountered when using rays to estimate velocity below rugose salt bodies is the instability of ray tracing. Rough salt topography creates poorly illuminated areas, or even shadow zones, in the subsalt region (Section 9.3). The spatial distribution of these poorly illuminated areas is very sensitive to the velocity function. Therefore, it is often extremely difficult to trace rays connecting a given point in the poorly illuminated areas with a given point at the surface (two-point ray-tracing). In contrast,

wavefield-extrapolation methods are also stable in poorly illuminated areas.

A related and more fundamental problem with ray-based MVA is that rays poorly approximate actual **wavepaths** when a band-limited seismic wave propagates through the rugose top of a salt body. Figure 12.1 illustrates this issue by showing three band-limited ($1-26$ Hz) wavepaths, also known in the literature as **fat rays** (Woodward, 1992; Pratt, 1999; Dahlen et al., 2000), for subsalt events propagating through a synthetic model of a salt body (Sigsbee 2A). The fat rays link perturbations in the image with perturbations in the velocity, and are thus the **sensitivity kernels** of wavefield MVA. Each of these three wavepaths is associated with the same point source located at the surface but corresponds to a different sub-salt "event". The top panel in Figure 12.1 shows a wavepath that could be reasonably approximated using the method introduced by Lomax (1994) to trace fat rays using asymptotic methods. In contrast, the wavepaths shown in both the middle and bottom panels in Figure 12.1 cannot be well approximated using Lomax's method. The amplitude and shapes of these wavepaths are significantly more complex than the simple fattening of a geometrical ray could ever describe. The bottom panel illustrates the worst-case scenario for ray-based tomography, because the variability of the top salt topology is at the same scale as the spatial wavelength of the seismic wave. The fundamental reason why true wavepaths cannot be approximated using fattened geometrical rays is that the wavepaths are frequency dependent. Figure 12.2 illustrates this dependency by depicting the wavepath shown in the bottom panel of Figure 12.1 as a function of the temporal bandwidth: $1-5$ Hz (top), $1-16$ Hz (middle), and $1-64$ Hz (bottom). The width of the wavepath decreases as the frequency bandwidth increases, and the focusing/defocussing of energy varies with the frequency bandwidth.

In 3-D, wavepaths are even more complex than in 2-D. Figure 12.3 displays three orthogonal slices of a 3-D slowness model containing a salt body. Figure 12.4 shows the sensitivity kernel related to a perturbation of the phase of subsalt event, imaged from a point source on the surface with a frequency range of $1-16$ Hz. The shapes of the kernels are complicated, which is an expression of the multipathing occurring as waves propagate through rough salt bodies. The horizontal slice indicates multiple paths linking the source point on the surface with the image perturbation in the subsurface. One noticeable characteristic is that the sensitivity kernels shown in Figure 12.4 shows little or no sensitivity along the central path; therefore, fat rays look more like "doughnuts" than "bananas" This phenomenon was discussed by Dahlen et al. (2000) and by Marquering et al. (1999) in the context of finite-frequency traveltime tomography.

Notwithstanding the potential advantages of using wavefield-continuation methods to estimate propagation velocity, these methods are seldom applied in practice. Computational cost has been a major hurdle in the past, but it is going to be less and less important in the future as the cost-effectiveness of high-performance computers increases. A more fundamental limitation of wavefield-based tomography or MVA is represented by the challenges of linearizing the wave equation for evaluating velocity updates. As we will discuss in detail in Section 12.2, the most convenient, and therefore popular, linearization of the wave equation is based on the truncation of the Born scattering series to the first order term (Born approximation). If the phase differences between the modeled and recorded wavefields are larger than a fraction of the wavelet, then the assumptions made under the Born approximation are violated, and the

velocity-inversion methods diverge (Woodward, 1992; Pratt, 1999; Dahlen et al., 2000; Hung et al., 2000). Overcoming these limitations is crucial for a practical MVA tool. This goal is easier to accomplish with methods that optimize an objective function defined in the image space (Symes and Carazzone, 1991; Biondi and Sava, 1999; Shen et al., 2003) than with methods that optimize an objective function defined in the data space (Tarantola, 1984; Mora, 1987; Pratt, 1999). However, even methods that optimize an image-space objective function are not automatically immune from the pitfalls related to the Born approximation, and therefore the objective function needs to be carefully designed.

In this chapter I present a general framework for improving the migrated image by updating the migration velocity using a wavefield-continuation operator. This general methodology could be used to optimize any of the image-space objective functions presented in the literature, such as the Differential Semblance Optimization (DSO) (Symes and Carazzone, 1991; Shen et al., 2003), the Multiple Migration Fitting (Chavent and Jacewitz, 1995), and the Wave-Equation Migration-Velocity Analysis (WEMVA) (Biondi and Sava, 1999; Sava and Biondi, 2004a,b). In the second part of this chapter, I focus on the WEMVA method, because it is conceptually related to the ray-based MVA methods described in Chapter 11, and it can be applied by leveraging the same robust work-flows currently used for conventional MVA.

12.1 Objective function of wavefield migration velocity analysis

The goal of all MVA methods is to maximize the quality of the migrated image by iteratively updating the velocity function. The coherency between partial prestack images in Common-Image Gathers (CIG) is customarily used as a measure of image quality, but the focusing of the reflections along the physical dimensions can be also a useful metric (Section 11.2.2). The "optimal" migration velocity is estimated by solving an optimization problem defined by an objective function that mathematically expresses the goals of MVA. Since Angle-Domain CIGs (ADCIGs) are the most natural CIGs for wavefield-continuation migration (Chapter 6), the objective functions of wavefield-based MVA methods are usually defined for prestack images transformed to the angle domain.

The objective function of ray-based MVA methods [equation (11.19)] is defined in terms of the kinematics of the migrated images. In particular, it aims at minimizing the RMO function measured from the CIGs, or, as we discussed at the end of Section 11.4.1, it drives the RMO function toward a "target" RMO function that is zero for all reflectors and at all angles (offsets). The objective function of wavefield-based MVA methods has a similar goal (i.e. flattening the ADCIGs), but is defined in terms of the whole image, including kinematics *and* amplitudes. It aims at driving the migrated image, $\mathbf{I}$, toward a "target" image, $\widehat{\mathbf{I}}$, that has the desired properties (i.e.; flat ADCIGs and well-focused reflections). Conceptually, such an objective function could be written as follows:

$$J_{\text{reg}}(\mathbf{s}) = \left\| \mathbf{I}(\mathbf{s}) - \widehat{\mathbf{I}} \right\|_2 + \mathbf{s}' \mathbf{C_s}^{-1} \mathbf{s}, \tag{12.1}$$

where $\mathbf{s}$ is the slowness function, and $\mathbf{C_s}$ is an estimate of the covariance matrix of the slowness function, as discussed for ray-based MVA in Chapter 11.

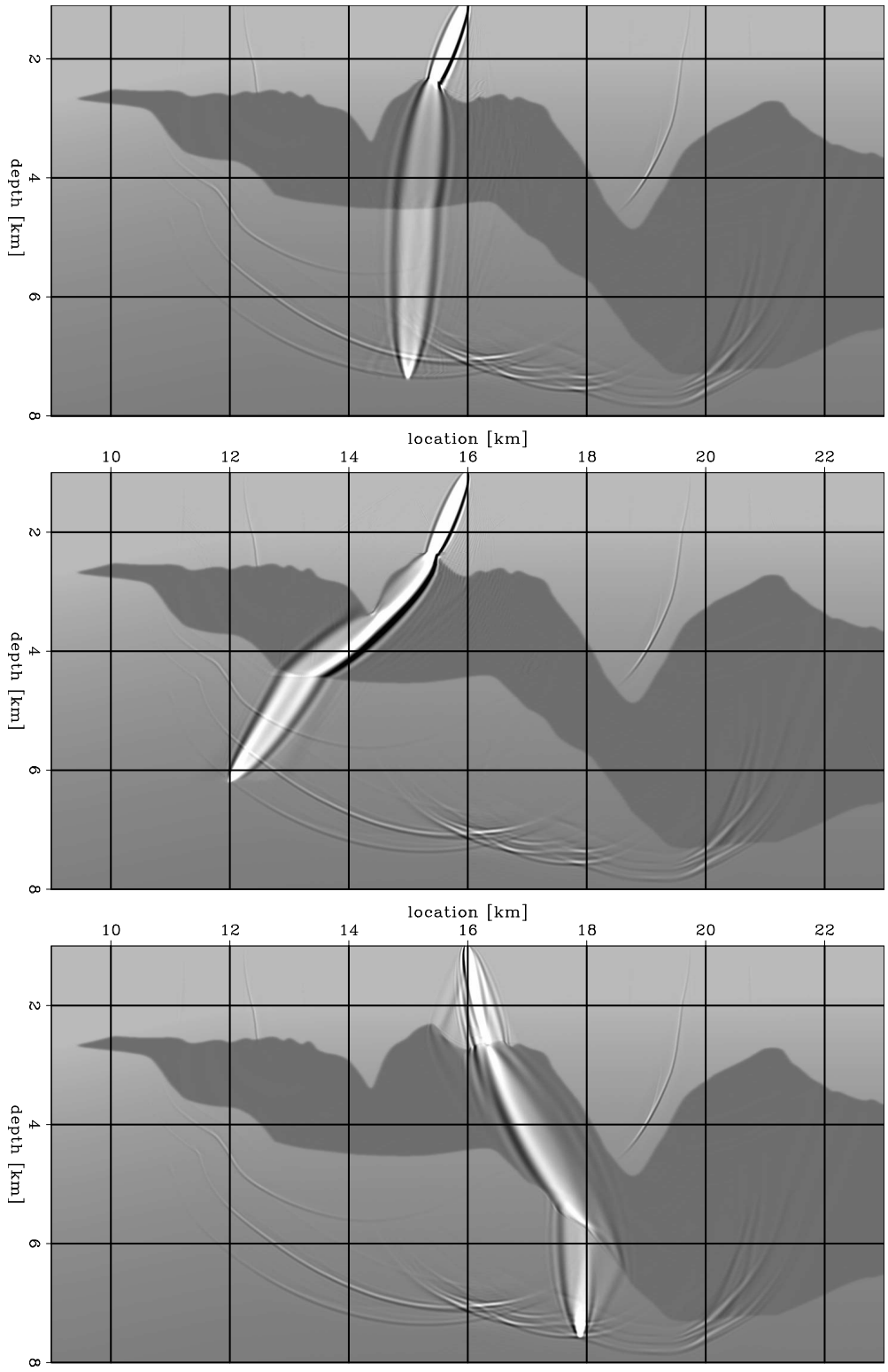


Figure 12.1: Wavepaths for frequencies between 1 and 26 Hz connecting a point on the surface with various locations in the image. Each panel is an overlay of three elements: the slowness model, the wavefield corresponding to a point source on the surface at $x = 16$ km, and wavepaths from a point in the subsurface to the source. (This figure is from Sava and Biondi (2004c).) wemva-zifat [NR]

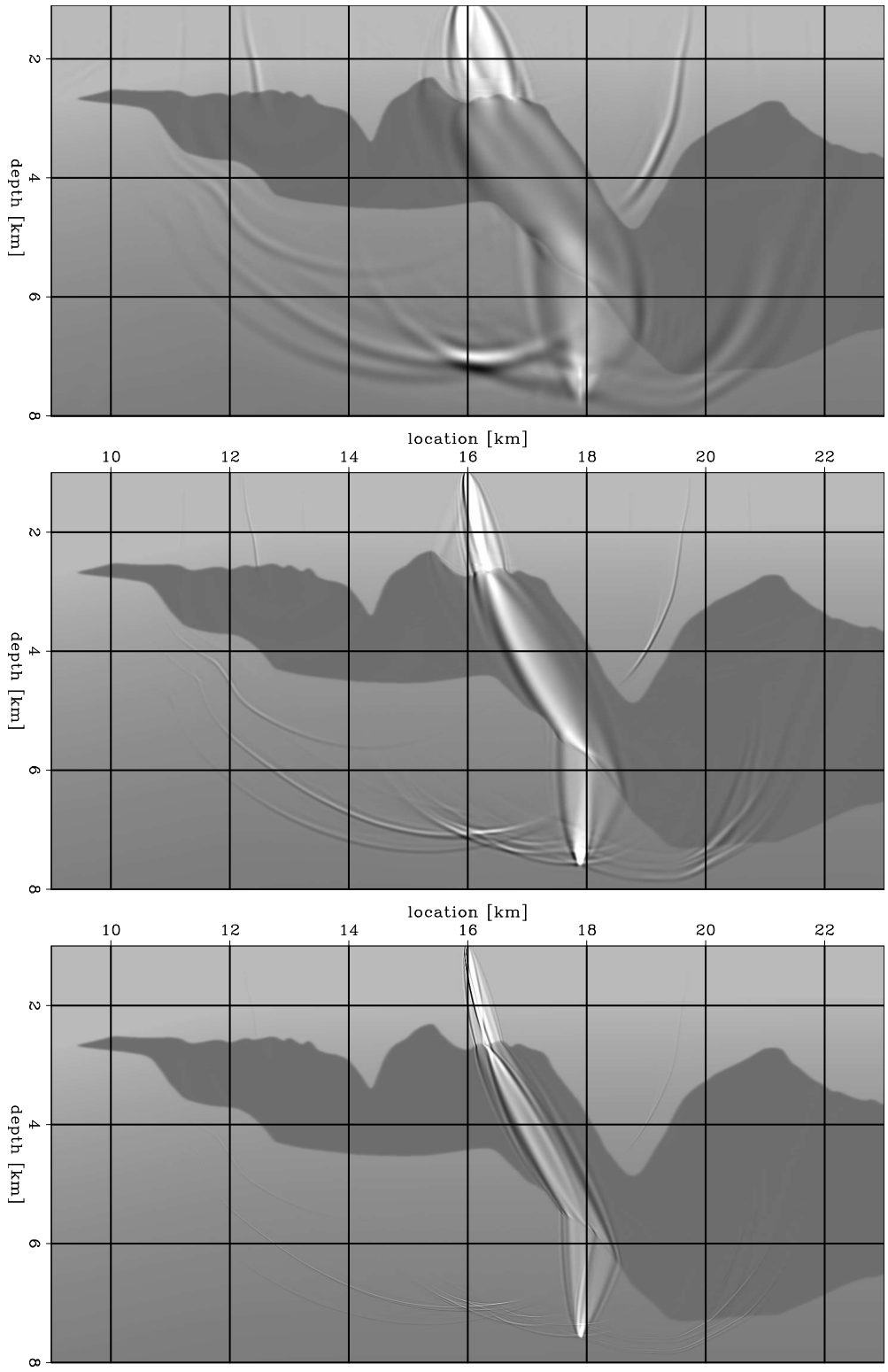


Figure 12.2: Frequency dependence of wavepaths between a location in the image and a point on the surface. Each panel is an overlay of three elements: the slowness model, the wavefield corresponding to a point source on the surface at $x = 16$ km, and wavepaths from a point in the subsurface to the source. The different wavepaths correspond to frequency bands of $1-5$ Hz (top), $1-16$ Hz (middle) and $1-64$ Hz (bottom). The larger the frequency band, the narrower the wave-path. The end member for an infinitely wide frequency band corresponds to an infinitely thin geometrical ray. (This figure is from Sava and Biondi (2004c).) wemva-zifrq2 [NR]

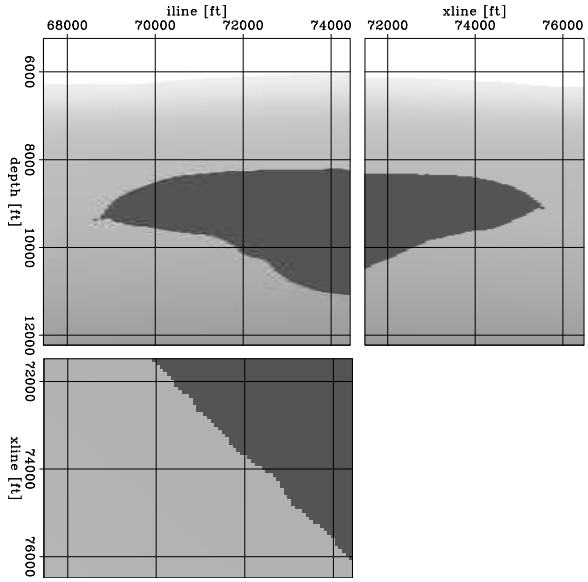


Figure 12.3: Orthogonal slices cut through a 3-D slowness model containing a salt body. (This figure is from Sava and Biondi (2004c)) wemva-fat3.sC [NR]

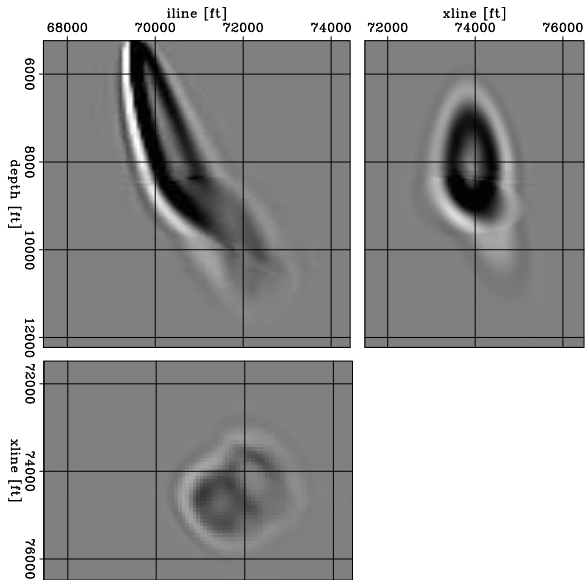


Figure 12.4: Orthogonal slices cut through the 3-D sensitivity kernel for wave-equation MVA. The frequency range is $1-16$ Hz. The kernels are complicated by the multipathing occurring as waves propagate through the salt body. (This figure is from Sava and Biondi (2004c).) wemva-fat3.fp3 [NR]

In practice, defining an objective function for both the amplitudes and the kinematics of the image that can be effectively optimized is more challenging than for the simple kinematics. In the case of ray-based MVA, the target is defined as an RMO function that is zero for all angles, but the target image in equation (12.1) cannot be simply set to zero. In practice, the target image must be a function of the migrated image. In general, the target image is the result of enhancing the migrated image by the application of a non-linear "image-enhancing" operator $\mathcal{K}_w$ to the migrated image. Therefore, the objective function of an effective MVA method must be written in a slightly more general form:

$$J_{\text{reg}}(\mathbf{s}) = \|\mathbf{W}_w \{\mathbf{I}(\mathbf{s}) - \mathcal{K}_w [\mathbf{I}(\mathbf{s})]\}\|_2 + \mathbf{s}^{'} \mathbf{C_s}^{-1} \mathbf{s}, \tag{12.2}$$

where $\mathbf{W}_w$ is a weighting operator that enhances the image components that we want to penalize (or reward), so that the optimization problem defined by the objective function (12.2) becomes tractable with gradient-based methods.

The properties of the operators $\mathcal{K}_w$ and $\mathbf{W}_w$ are the main distinguishing elements among existing wavefield-based MVA methods. The following list summarizes the choices of three well-known approaches:

- In the Differential Semblance Optimization (DSO) method (Symes and Carazzone, 1991; Shen et al., 2003), the linear operator $\mathbf{W}_w$ is a differential operator acting along the angle axes in the ADCIGs, and $\mathcal{K}_w = 0$. The rationale for this choice of objective function is that by minimizing the differences between the migrated images of contiguous angles we drive the optimization process toward flat ADCIGs.

- In the Multiple Migration Fitting (MMF) method (Chavent and Jacewitz, 1995) the linear operator $\mathbf{W}_w$ is a stacking operator along the angle axes in the ADCIGs, and $\mathcal{K}_w = 0$. The MMF objective function is maximized, instead of being minimized. By maximizing the stack power along angles, we reward well-focused images, and images that are coherent along the angle axes.

- In the Wave-Equation MVA (WEMVA) method (Biondi and Sava, 1999; Sava and Biondi, 2004a,b), the target image $\widehat{\mathbf{I}}$ is an improved version of the migrated image $\mathbf{I}$, and $\mathbf{W}_w$ is the identity operator. In this case, $\mathcal{K}_w$ is represented by an operator performing residual moveout (Section 11.2.1) or residual migration (Section 11.2.2), as a function of spatially varying values of ρ. For a fixed value of ρ, these residual imaging operators are linear with respect to the image, and thus they can be represented by the linear operator $\mathbf{K}_w(\rho)$. The non-linearity of $\mathcal{K}_w$ derives from the fact that the values of ρ are determined based on the migrated image itself.

The objective function expressed in equation (12.2) is highly non-quadratic. It could be optimized by applying algorithms specifically designed for non-quadratic problems. A good choice could be a version of the quasi-Newton algorithm (Gill et al., 1981) that does not require the storage of the complete Hessian; Guitton and Symes (2003) describe a useful seismic application of such an algorithm (BFGS-L). Alternatively, a version of the conjugate-gradient algorithm that is effective for non-quadratic problems, such as the conjugate-direction algorithm, can also be used (Claerbout, 2004).

As I discussed in Section 11.4.4, the convergence of the velocity estimation process greatly benefits when some crucial tasks, such as the determination of RMO parameters and the estimation of the covariance of the slowness function, are performed in conjunction with the interpretation of the migrated images. Because the interpretation of prestack images can be realistically carried out only a limited number of times during the estimation process, the use of fully automated optimization algorithms is challenging in practice. Therefore, in my analysis I follow the same approach used for ray-based MVA (Section 11.4), and address the non-quadratic nature of the problem by repeatedly solving the quadratic optimization problems obtained by linearizing the objective function (12.2).

After each linearization, the perturbed image can be approximated as a linear function of the slowness perturbation ${}_i\Delta\mathbf{s}$, as follows:

$$I({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}) \approx {}_{i-1}\mathbf{I} + {}_i\mathbf{M}_w\,{}_i\Delta\mathbf{s}, \tag{12.3}$$

where the left subscript i indicates the iteration number, ${}_{i-1}\hat{\mathbf{s}}$ is the current background slowness, ${}_{i-1}\mathbf{I}$ is the background image obtained by prestack migration with ${}_{i-1}\hat{\mathbf{s}}$, and ${}_i\mathbf{M}_w$ is a linearization of the prestack migration functional at ${}_{i-1}\hat{\mathbf{s}}$.

Substituting the expression (12.3) into the objective function (12.2), we obtain the following quadratic objective function:

$$\begin{aligned} J_{\text{reg}}({}_i\Delta\mathbf{s}) \;=\; & \|\mathbf{W}_w({}_{i-1}\mathbf{I} + {}_i\mathbf{M}_w\,{}_i\Delta\mathbf{s} - {}_i\mathbf{K}_w\,{}_{i-1}\mathbf{I})\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})'\,{}_i\mathbf{C_s}^{-1}({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}) = \\ & \|\mathbf{W}_w({}_i\mathbf{M}_w\,{}_i\Delta\mathbf{s} - {}_i\Delta\mathbf{I})\|_2 + ({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s})'\,{}_i\mathbf{C_s}^{-1}({}_{i-1}\hat{\mathbf{s}} + {}_i\Delta\mathbf{s}), \end{aligned} \tag{12.4}$$

where ${}_i\mathbf{C_s}$ is the current estimate of the slowness covariance, ${}_i\mathbf{K}_w$ is the image-enhancing operator determined by analyzing the current migrated image ${}_{i-1}\mathbf{I}$, and

$${}_i\Delta\mathbf{I} = ({}_i\mathbf{K}_w\,{}_{i-1}\mathbf{I} - {}_{i-1}\mathbf{I}) = ({}_i\mathbf{K}_w - \mathbf{1})\,{}_{i-1}\mathbf{I} \tag{12.5}$$

is the actual image perturbation fitted during each linearized iteration. In equation (12.5) the symbol **1** stands for the identity operator, to avoid confusion with the image **I**.

To assure that a gradient-based optimization converges to a useful estimate of the slowness perturbations, it is necessary for ${}_i\Delta\mathbf{I}$ to be consistent with the linearization of the wave-propagation process that we use to evaluate ${}_i\mathbf{M}_w$. The next section presents a method for evaluating ${}_i\mathbf{M}_w$ for downward-continuation migration, and the following section addresses the issue of computing a ${}_i\Delta\mathbf{I}$ that is consistent with ${}_i\mathbf{M}_w$.

12.2 Linearization of wave propagation with respect to the velocity function

To solve the quadratic optimization problem defined by the objective function (12.4), we need an algorithm for applying the linearized wavefield-MVA operator, $\mathbf{M}_w$, to a slowness-perturbations vector, $\Delta\mathbf{s}$, and computing the corresponding image-perturbations vector, $\Delta\mathbf{I}$. Similarly, we must be able to apply its adjoint, $\mathbf{M}_w'$, to an image-perturbations vector, $\Delta\mathbf{I}'$, and compute the corresponding slowness-perturbations vector, $\Delta\mathbf{s}'$.

A crucial component of the evaluation of $\mathbf{M}_w$ is the linearization of wave propagation. Two ways to linearize wave propagation with respect to velocity perturbations (**wavefield scattering**) have been developed and applied in the literature: the **Born linearization** and the **Rytov linearization**. The two methods can be derived using the same mathematical formalism; both methods are derived by truncating after the first-order term the Taylor series of the total perturbed wavefield, expanded as a function of the velocity perturbation. The zero-order term in the Taylor expansion represents the **background wavefield**, and the first-order term represents the **single-scattered wavefield**.

The main difference between the two approximations is that when the Born approximation is derived, the Taylor expansion is defined in terms of the amplitudes (real and imaginary) of the complex-valued wavefield, whereas for the Rytov approximation, the Taylor expansion is defined in terms of the complex phase (i.e. complex logarithm of the complex amplitude) of the wavefield (Devaney, 1981).

The Born approximation underlies most of the migration methods (Cohen and Bleistein, 1979; Clayton and Stolt, 1981; Stolt and Weglein, 1985; Stolt and Benson, 1986). However, for the purposes of velocity analysis, the Rytov approximation is more appealing at first sight, because it better approximates phase delays caused by large velocity anomalies (Devaney, 1981; Woodward, 1990). However, the Rytov approximation has severe shortcomings when complex multipathing occurs in the background wavefield, because phase perturbations are not additive. Furthermore, practical applications of the Rytov linearization require phase unwrapping, which is a task prone to errors with complex and noisy data. In contrast, the Born approximation has no difficulties with multipathing in the background wavefield, and does not require phase unwrapping. Furthermore, the evaluation and propagation of the scattered wavefield is fairly straightforward. However, the Born linearization fails when the phase differences between the perturbed and background wavefields are larger than a fraction of the wavelet, making it difficult to properly handle large velocity perturbations. The preceding brief discussion of the issues relating the Born and Rytov linearizations of the wave-equation is by necessity incomplete: *Fundamentals of Seismic Tomography* (Lo and Inderweisen, 1994) provides a succinct but clear description of both approximations, and Woodward (1990) clearly discusses the advantages and disadvantages of both approximations.

Most of the wavefield-MVA methods that have been presented in the literature are based on the Born approximation as is the evaluation of the linearized wavefield-MVA operator presented in this section. The challenge is to avoid the Born approximation's limitations, and to devise an MVA method that is capable of correctly handling large velocity perturbations. Section 12.3 analyzes this issue for the DSO, MMF, and WEMVA methods, applying the insight into the Born approximation gained by the detailed analysis presented in this section. Section 12.4 presents a solution that solves this challenge for the WEMVA method. In this section I present the method introduced by Biondi and Sava (1999), for evaluating $\mathbf{M}_w$ and its adjoint for source-receiver migration. The same concepts can be easily applied for shot-profile migration (Shen et al., 2003), or any other downward-continuation migration.

12.2.1 Application of $\mathbf{M}_w$ and $\mathbf{M}'_w$ by downward continuation

The development of the algorithm for the evaluation of $\mathbf{M}_w$ is somewhat intricate, because the two perturbations vectors, $\Delta\mathbf{s}$ and $\Delta\mathbf{I}$, are not directly linked. They are connected through two intermediary vector fields: the background wavefield, $\mathbf{P}$, and the scattered wavefield, $\Delta\mathbf{P}$. Three operators also play a role in this algorithm: the downward-continuation operator, $\mathbf{E}$, the imaging operator, $\mathbf{\Sigma}$, and the **scattering operator**, $\mathbf{G}$. The downward-continuation operator and the imaging operator are the familiar components of downward-continuation migration (Chapter 4), whereas the scattering operator is described in detail in this section.

The flow-chart in Figure 12.5 summarizes the algorithm for evaluating $\mathbf{M}_w$ and its adjoint. The three components are labeled as A, B and C on the chart. Box A corresponds to the computation of the background wavefield, based on the surface data, $\mathbf{D}$, and background slowness, $\hat{\mathbf{s}}$. Boxes B and C correspond respectively to the forward and adjoint $\mathbf{M}_w$ operator.

Computation of the background wavefield

Imaging by source-receiver migration (Section 4.2.2) is based on the following recursive continuation of the wavefields, $\mathbf{P}_z$, from a given depth level, z, to the next, $z+\Delta z$, by means of the Double Square Root operator, $\mathbf{E}(\hat{\mathbf{s}}_z)$, which is function of the slowness vector, $\hat{\mathbf{s}}_z$:

$$\mathbf{P}_{z+\Delta z} = \mathbf{E}_z(\hat{\mathbf{s}}_z)\mathbf{P}_z \ . \tag{12.6}$$

All the steps of this recursive extrapolation can also be expressed in matrix form as follows:

$$\begin{pmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_n \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{E}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_{n-1} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_n \end{pmatrix} + \begin{pmatrix} \mathbf{D}_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{12.7}$$

where $\mathbf{D}_0$ is the data recoded at the surface. Or, in a more compact notation,

$$\mathbf{P} = \mathbf{E}(\hat{\mathbf{s}})\mathbf{P} + \mathbf{D}. \tag{12.8}$$

Here and hereafter, I use the subscripts z and $z+\Delta z$ to distinguish the quantities measured at a particular depth level (e.g. $\mathbf{P}_z$), from the corresponding vectors denoting such quantities at all depth levels (e.g. $\mathbf{P}$). The recursive relationship in equation (12.8) can be formally solved as follows:

$$\mathbf{P} = [\mathbf{1} - \mathbf{E}(\hat{\mathbf{s}})]^{-1}\mathbf{D}, \tag{12.9}$$

where $\mathbf{1}$ stands for the identity operator.

After wavefield extrapolation, we obtain an image by applying, at every depth level, an imaging operator, $\mathbf{\Sigma}_z$, to the extrapolated wavefield $\mathbf{P}_z$:

$$\mathbf{I}_z = \mathbf{\Sigma}_z\mathbf{P}_z \ , \tag{12.10}$$

where $\mathbf{I}_z$ stands for the image at the z level. A commonly used imaging operator involves summation over the temporal frequencies (Section 4.2.2). We can write the same relation in compact matrix form:

$$\mathbf{I} = \mathbf{\Sigma P} = \mathbf{\Sigma}\,[\mathbf{1} - \mathbf{E}(\hat{\mathbf{s}})]^{-1}\,\mathbf{D}, \tag{12.11}$$

where $\mathbf{I}$ stands for the image, and $\mathbf{\Sigma}$ stands for the imaging operator which is applied to the extrapolated wavefield $\mathbf{P}$ at all depth levels.

The downward-continuation expression (12.8) and the imaging expression (12.11) summarize the computation of the background wavefield and the background image that is represented by Box A in Figure 12.5. The background wavefield is an important component of the wavefield-MVA operator. This wavefield plays a role analogous to the one played in traveltime tomography by the ray-field obtained by ray tracing in the background model. The wavefield is the carrier of information and defines the wavepaths along which we spread the velocity errors measured from the migrated images, which are obtained using the background slowness function. Unlike a ray-field, which describes propagation of waves with an infinite frequency band, the wavefield is band-limited. It provides a more accurate description of wave propagation through complicated media than a corresponding ray-field (Figures 12.1 and 12.2). Typical applications are salt bodies characterized by large velocity contrasts, where ray tracing is both unstable and inaccurate.

Application of the forward operator

At each depth level, the interaction of the propagating wavefield with a slowness perturbation creates a scattered wavefield, which is summed to the scattered wavefield created at the depth levels above. When computing the scattered wavefield at each depth level, we should consider the interaction between the total propagating wavefield (background and scattered) and the velocity perturbations. However, to maintain the linearity of the relationship, we limit the computations only to the contributions of the background wavefield to the scattering process, neglecting the contributions of the scattered wavefield created at the previous depth levels. These contributions are considered second-order; their omission constitutes the first-order Born approximation discussed above. Because of the first-order approximation, we can also neglect the scattered wavefield propagating in the direction opposite to the propagation direction of the background wavefield. Therefore, we consider only the forward-scattered wavefield, and we propagate the scattered wavefield in the same direction as the background wavefield; that is, only toward increasing depths.

The contribution to the scattered wavefield, $\Delta\mathbf{U}_{z+\Delta z}$, is generated by the interaction of the background wavefield, $\mathbf{P}_z$, with a perturbation of the velocity model, $\Delta\mathbf{s}_z$, which is then propagated from z to $z+\Delta z$; that is,

$$\Delta\mathbf{U}_{z+\Delta z} = \mathbf{E}_z(\hat{\mathbf{s}}_z)\,\mathbf{G}_z(\mathbf{P}_z, \hat{\mathbf{s}}_z)\,\Delta\mathbf{s}_z, \tag{12.12}$$

where the scattering operator $\mathbf{G}_z$ is function of both the incident background wavefield and the local background slowness function $\hat{\mathbf{s}}_z$. After scattering occurs, the scattered wavefield is propagated downward using the same operator, $\mathbf{E}_z$, and background slowness, $\Delta\mathbf{s}_z$, that are

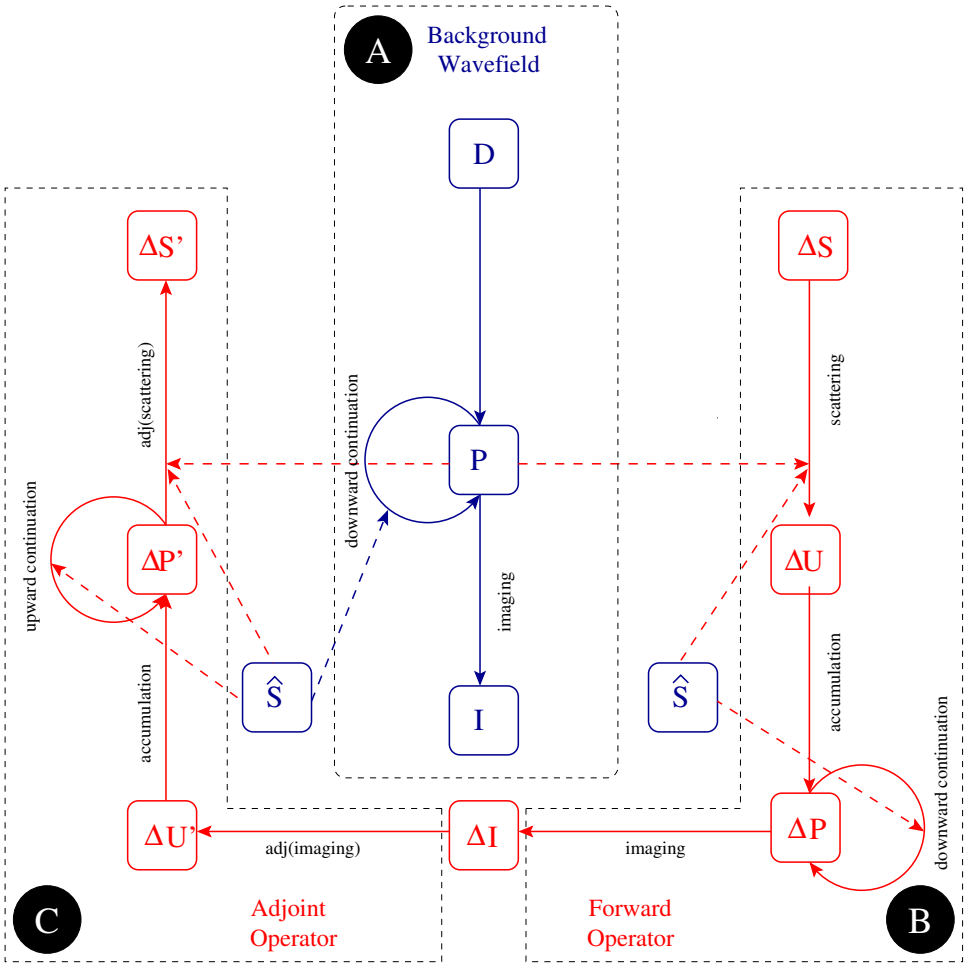


Figure 12.5: Box A – Background Wavefield: The data recorded at the surface (**D**) are extrapolated in depth using the background slowness ($\hat{\mathbf{s}}$), generating the background wavefield (**P**); the background wavefield (**P**) is transformed into the background image (**I**) using an imaging operator. Box B - Forward Operator: The background wavefield (**P**) interacts with a slowness perturbation ($\Delta\mathbf{s}$), generating a scattered wavefield ($\Delta\mathbf{U}$); after depth extrapolation, the locally scattered wavefield is accumulated into a total scattered wavefield ($\Delta\mathbf{P}$), which is transformed into an image perturbation ($\Delta\mathbf{I}$) using an imaging operator. Box C - Adjoint Operator: The image perturbation ($\Delta\mathbf{I}$) is transformed into the scattered wavefield ($\Delta\mathbf{U}'$) using the adjoint of the imaging operator. The locally scattered wavefield is accumulated into the adjoint-state scattered wavefield ($\Delta\mathbf{P}'$), which is upward continued, and at every depth level it is transformed into an adjoint slowness perturbation ($\Delta\mathbf{s}'$), by applying the adjoint of the scattering operator. wemva-wemva_chart-v5 [NR]

used to propagate the background wavefield. Therefore, the scattered wavefield propagates along the same wavepaths as the background wavefield.

The total scattered wavefield at depth $z+\Delta z$, $\Delta\mathbf{P}_{z+\Delta z}$, is the result of downward continuation from z to $z+\Delta z$ of the sum of $\Delta\mathbf{U}_{z+\Delta z}$ with the total scattered wavefield at depth z, $\Delta\mathbf{P}_z$; that is,

$$\Delta\mathbf{P}_{z+\Delta z} = \mathbf{E}_z(\hat{\mathbf{s}}_z)\,\Delta\mathbf{P}_z + \Delta\mathbf{U}_{z+\Delta z}. \tag{12.13}$$

Substituting equation (12.12) into equation (12.13) we obtain,

$$\Delta\mathbf{P}_{z+\Delta z} = \mathbf{E}_z(\hat{\mathbf{s}}_z)\,\Delta\mathbf{P}_z + \mathbf{E}_z(\hat{\mathbf{s}}_z)\,\mathbf{G}_z(\mathbf{P}_z,\hat{\mathbf{s}}_z)\,\Delta\mathbf{s}_z, \tag{12.14}$$

which also can be written in matrix form as follows:

$$\begin{pmatrix}\Delta\mathbf{P}_0\\ \Delta\mathbf{P}_1\\ \Delta\mathbf{P}_2\\ \vdots\\ \Delta\mathbf{P}_n\end{pmatrix} = \begin{pmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \mathbf{E}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{E}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_{n-1} & \mathbf{0}\end{pmatrix}\begin{pmatrix}\Delta\mathbf{P}_0\\ \Delta\mathbf{P}_1\\ \Delta\mathbf{P}_2\\ \vdots\\ \Delta\mathbf{P}_n\end{pmatrix} + \begin{pmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \mathbf{E}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{E}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_{n-1} & \mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{G}_0 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_1 & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{G}_2 & \cdots & \mathbf{0}\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{G}_n\end{pmatrix}\begin{pmatrix}\Delta\mathbf{s}_0\\ \Delta\mathbf{s}_1\\ \Delta\mathbf{s}_2\\ \vdots\\ \Delta\mathbf{s}_n\end{pmatrix},$$

or in a more compact notation as follows:

$$\Delta\mathbf{P} = \mathbf{E}(\hat{\mathbf{s}})\,\Delta\mathbf{P} + \Delta\mathbf{U} = \mathbf{E}(\hat{\mathbf{s}})\,\Delta\mathbf{P} + \mathbf{E}(\hat{\mathbf{s}})\,\mathbf{G}(\mathbf{P},\hat{\mathbf{s}})\,\Delta\mathbf{s}, \tag{12.15}$$

where the vector $\Delta\mathbf{s}$ stands for the slowness perturbation at all depths, and the vector $\Delta\mathbf{U}$ stands for the contributions to the scattered wavefields at all depths. The formal solution of the recursive equation (12.15) is

$$\Delta\mathbf{P} = [\mathbf{1}-\mathbf{E}(\hat{\mathbf{s}})]^{-1}\,\mathbf{E}(\hat{\mathbf{s}})\,\mathbf{G}(\mathbf{P},\hat{\mathbf{s}})\,\Delta\mathbf{s}. \tag{12.16}$$

By applying the imaging operator to the scattered wavefield, $\Delta\mathbf{P}$, we obtain a linear relationship between the image perturbations, $\Delta\mathbf{I}$, and the slowness perturbations, $\Delta\mathbf{s}$, and consequently, an expression for the forward operator $\mathbf{M}_w$:

$$\Delta\mathbf{I} = \mathbf{\Sigma}\,[\mathbf{1}-\mathbf{E}(\hat{\mathbf{s}})]^{-1}\,\mathbf{E}(\hat{\mathbf{s}})\,\mathbf{G}(\mathbf{P},\hat{\mathbf{s}})\,\Delta\mathbf{s} = \mathbf{M}_w\,\Delta\mathbf{s}. \tag{12.17}$$

Notice the parallelism between equation (12.11) and equation (12.17), or analogously between the recursive relationships (12.8) and (12.15). In equation (12.15), the scattered wavefield $\Delta\mathbf{U}$ that is generated by the interactions of the background wavefield with the slowness perturbations acts as "source term" at each depth level. In the conventional downward-continuation process, equation (12.8), the recorded data acts as a source at the surface. The process for evaluating the forward operator, $\mathbf{M}_w$, is summarized in Box B in Figure 12.5.

Application of the adjoint operator

To optimize the MVA objective function (12.4), we need to evaluate the adjoint operator, $\mathbf{M}'_w$, that backprojects the image perturbations $\Delta\mathbf{I}'$ into slowness perturbations $\Delta\mathbf{s}'$. Its application could be computed by applying the general **adjoint-state methodology**, as defined by Lions (1971), but the formal derivation of the adjoint operator from equation (12.17) is the most

straightforward. If we take the adjoint of the operator expressed in equation (12.17), we can write the following expression:

$$\Delta \mathbf{s}' = \mathbf{G}'(\mathbf{P},\hat{\mathbf{s}})\mathbf{E}'(\hat{\mathbf{s}})\left[\mathbf{1}-\mathbf{E}'(\hat{\mathbf{s}})\right]^{-1}\mathbf{\Sigma}'\Delta\mathbf{I}'. \tag{12.18}$$

This formal expression for the adjoint operator can be unraveled and written as an upward continuation of an **adjoint-state scattered wavefield**, $\Delta\mathbf{P}'$,

$$\Delta\mathbf{P}' = \mathbf{E}'(\hat{\mathbf{s}})\,\Delta\mathbf{P}' + \Delta\mathbf{U}', \tag{12.19}$$

where $\Delta\mathbf{U}' = \mathbf{\Sigma}'\Delta\mathbf{I}'$, followed by an "imaging" step,

$$\Delta\mathbf{s}' = \mathbf{G}'(\mathbf{P},\hat{\mathbf{s}})\mathbf{E}'(\hat{\mathbf{s}})\,\Delta\mathbf{P}'. \tag{12.20}$$

The combination of equations (12.19) and (12.20) provides an algorithm for applying the adjoint operator $\mathbf{M}'_w$ to image perturbations $\Delta\mathbf{I}'$ and computing slowness perturbations $\Delta\mathbf{s}'$. Box C in Figure 12.5. summarizes the process for evaluating $\mathbf{M}'_w$.

Although it is not obvious at first sight, the operation represented by equation (12.20) is indeed an imaging-like step. It represents a correlation between the adjoint-state scattered wavefield, $\Delta\mathbf{P}'$, and the background wavefield phase-rotated by 90 degrees, $i\mathbf{P}$, as shown by the expression of the scattering operator, $\mathbf{G}$, in equation (12.23). If the phases of $i\mathbf{P}$ (or $-i\mathbf{P}$) and $\Delta\mathbf{P}'i$ are close to each other, approximately within a ± 45 degree interval, the correlation generates coherent positive (or negative) slowness perturbations that provide useful search directions for the optimization algorithm. If the phases are not close to one another, the correlation generates incoherent slowness perturbations, possibly even with the wrong sign.

In equation (12.19) the image perturbations $\Delta\mathbf{I}'$ act as source term, and thus the phase of $\Delta\mathbf{P}'$ is related to the phase of $\Delta\mathbf{I}'$, whereas the phase of $\mathbf{P}$ is related to the phase of $\mathbf{I}$ through the imaging condition (12.10). Therefore, the relative phase delay of the adjoint-state scattered wavefield with respect to the background wavefield depends on the distance between the events in the image perturbation $\Delta\mathbf{I}'$ and the background image $\mathbf{I}$. If the reflectors in the two images are sufficiently close, a velocity-estimation process based on the Born approximation converges; otherwise it may diverge.

The scattering operator

The scattering operator, $\mathbf{G}$, enters into the computation of both the forward and the adjoint wavefield-MVA operators. It can be derived by physical considerations or can be simply derived by a formal linearization of the depth-propagation operator. This formal derivation is related to the Taylor expansion of the SSR equation introduced in Section 5.2.2, but in this case the entire wavefield, and not just the exponent, is linearized. From the approximation of the SSR operator in equation 5.4 we write

$$\mathbf{P}_{z+\Delta z} = e^{ik_z\Delta z}\mathbf{P}_z \approx e^{\left(i\Delta z\widehat{k_z}+i\Delta z\,\frac{dk_z}{ds}\Big|_{\mathbf{s}=\hat{\mathbf{s}}}\Delta\mathbf{s}\right)}\mathbf{P}_z \approx e^{i\Delta z\widehat{k_z}}\mathbf{P}_z + e^{i\Delta z\widehat{k_z}}\left(i\Delta z\,\frac{dk_z}{ds}\bigg|_{\mathbf{s}=\hat{\mathbf{s}}}\right)\mathbf{P}_z\Delta\mathbf{s}, \tag{12.21}$$

where

$$\left.\frac{dk_z}{ds}\right|_{\mathbf{s}=\hat{\mathbf{s}}} = \frac{\omega}{\sqrt{1-\frac{|\mathbf{k}|^2}{\omega^2\hat{\mathbf{s}}^2}}}. \tag{12.22}$$

By comparing the last term in equation (12.21) with the last term in equation (12.14), we can write the scattering operator as follows:

$$\mathbf{G}(\mathbf{P},\hat{\mathbf{s}}) = i\,\Delta z\,\left.\frac{dk_z}{ds}\right|_{\mathbf{s}=\hat{s}}\mathbf{P}_z = \frac{i\omega\Delta z}{\sqrt{1-\frac{|\mathbf{k}|^2}{\omega^2\hat{\mathbf{s}}^2}}}\mathbf{P}_z. \tag{12.23}$$

The scattering operator expressed in equation (12.23) is a linear function of the background wavefield $\mathbf{P}_z$, and a non-linear function of the background slowness $\hat{\mathbf{s}}$. Because the background slowness is a spatially varying function, the dip-dependent weight expressed by the denominator in equation (12.23) is spatially non-stationary, and the expression in equation (12.23) is a mixed-domain representation of $\mathbf{G}$. Its numerical implementation presents challenges similar to those encountered implementing the SSR operator for downward-continuing wavefields in laterally varying media. However, in contrast with the downward-continuation case, the scattering operator is not applied recursively. Therefore, the implementation of $\mathbf{G}$ is not subject to the same stringent stability requirements as the SSR operator is; the computations of $\mathbf{G}$ do not diverge even if its magnitude is not uniformly less than one for all wavenumbers.

A straightforward method for applying the scattering operator was proposed by Huang et al. (1999) in the context of approximating the SSR for downward continuation. Whereas this numerical method is unstable when applied to downward continuation, it is appropriate for applying $\mathbf{G}$. The method is based on the following Taylor series expansion (Sava and Biondi, 2004a):

$$i\,\Delta z\,\left.\frac{dk_z}{ds}\right|_{\hat{s}}\mathbf{P}_z \approx i\omega\Delta z\left[1+\frac{1}{2}\left(\frac{|\mathbf{k}|}{\omega\hat{\mathbf{s}}}\right)^2+\frac{3}{8}\left(\frac{|\mathbf{k}|}{\omega\hat{\mathbf{s}}}\right)^4+\frac{5}{16}\left(\frac{|\mathbf{k}|}{\omega\hat{\mathbf{s}}}\right)^6+\frac{35}{128}\left(\frac{|\mathbf{k}|}{\omega\hat{\mathbf{s}}}\right)^8+...\right]\mathbf{P}_z. \tag{12.24}$$

The even powers of $|\mathbf{k}|$ represent spatial-derivative operators that can be applied either in the space domain by finite-difference approximations, or in the Fourier domain.

Simple examples of the application of $\mathbf{M}_w$ and $\mathbf{M}'_w$

Figures 12.6 and 12.7 illustrate the flow-chart in Figure 12.5, by showing its application to two simple examples. In the first example (Figure 12.6), I use a monochromatic wavefield, whereas in the second one (Figure 12.7), I use a wide-band wavefield. For both examples the data are recorded above a planar horizontal reflector.

Figure 12.6a shows a snapshot (taken at time zero) of the monochromatic background wavefield obtained by downward continuation of an incident plane wave in a constant medium. Figure 12.6b shows a slowness perturbation that, under the influence of the incident wavefield (a), generates a wavefield perturbation (c). The snapshots at zero time shown in panels (a) and

(c) can also be regarded as images. Finally, I back-propagate the image perturbation (c) and obtain the adjoint slowness perturbation (d).

Figure 12.7 shows panels analogous to those shown in Figure 12.6, but for wide-band data. Figure 12.7a shows the image obtained by wavefield extrapolation of a wide-band plane wave in the background medium. From the same slowness perturbation (b) as in the preceding example, I obtain an image perturbation (c), from which I generate an adjoint slowness perturbation (d) using the background wavefield used to compute the background image.

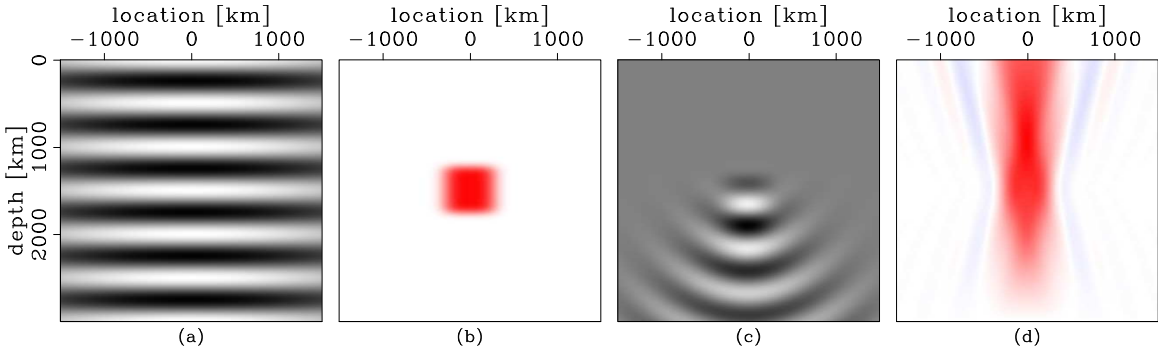


Figure 12.6: Monochromatic WEMVA example: background wavefield (a), slowness perturbation (b), wavefield perturbation (c), slowness backprojection (d). (This figure is from Sava (2004).) wemva-SCATbas1.scat [NR]

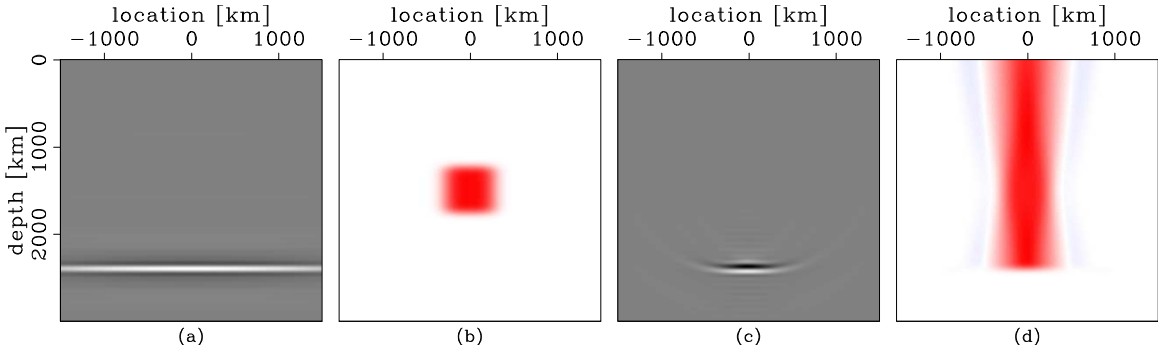


Figure 12.7: Wide-band WEMVA example: background image (a), slowness perturbation (b), image perturbation (c), slowness backprojection (d). (This figure is from Sava (2004).) wemva-SCATbasN.scat [NR]

12.3 Convergence of wavefield migration velocity analysis

In the previous section (Section 12.2), I presented a general approach for computing the linearized MVA operator, $\mathbf{M}_w$, associated with downward-continuation migration, and a specific algorithm to for source-receiver migration. This methodology can be applied to the velocity-estimation problem defined by the the non-quadratic objective function (12.2), and by the

linearized objective function (12.4). These objective functions define a family of wavefield-MVA methods; a specific method within this family is determined by the choice of the two operators $\mathcal{K}_w$ and $\mathbf{W}_w$. In Section 12.1 I briefly discussed three choices for the operators $\mathbf{W}_w$ and $\mathcal{K}_w$ that correspond to three known methods for wavefield MVA: DSO, MMF and WEMVA. In this section I will further discuss these three methods in light of the properties of $\mathbf{M}_w$ discussed in Section 12.2. In particular, I address the important issue of whether a specific choice for $\mathcal{K}_w$ and $\mathbf{W}_w$ yields an MVA method that converges even in presence of large errors in the background velocity function.

To analyze the issue of convergence, I consider the gradient direction computed by back-projecting the image perturbations into the slowness model. This back-projection is performed by applying the adjoint operator $\mathbf{M}'_w$ to the image perturbations $\Delta\mathbf{I}$. As discussed in (Section 12.2.1), we expect a useful gradient direction when the phase difference between the background image $\mathbf{I}$ and the image perturbation $\Delta\mathbf{I}$ is sufficiently small to fulfill the assumptions of the Born approximation.

The general form of the image perturbation ${}_i\widetilde{\Delta\mathbf{I}}$ that is actually back-projected at the $i-$th linearized iteration can be directly derived from the objective function (12.4) as follows:

$$ {}_i\widetilde{\Delta\mathbf{I}} = \mathbf{W}'_w\mathbf{W}_w\,({}_i\mathbf{M}_w\,{}_i\Delta\mathbf{s} - {}_i\Delta\mathbf{I}) = \mathbf{W}'_w\mathbf{W}_w\left[{}_i\mathbf{M}_w\,{}_i\Delta\mathbf{s} - ({}_i\mathbf{K}_w - \mathbf{1})\,{}_{i-1}\mathbf{I},\right]. \tag{12.25}$$

For the sake of simplicity, I analyze the gradient direction at the first linearized iteration, when ${}_i\Delta\mathbf{s} = 0$. In this case, equation (12.25) simplifies into

$$ {}_i\widetilde{\Delta\mathbf{I}} = \mathbf{W}'_w\mathbf{W}_w\,({}_i\mathbf{K}_w - \mathbf{1})\,{}_{i-1}\mathbf{I}. \tag{12.26}$$

In both the DSO and MMF cases, ${}_i\mathbf{K}_w = 0$, and equation (12.26) further simplifies into

$$ {}_i\widetilde{\Delta\mathbf{I}} = \mp\mathbf{W}'_w\mathbf{W}_w\,{}_{i-1}\mathbf{I}. \tag{12.27}$$

The sign is negative for DSO and positive for MMF, because in one case the objective function is minimized, whereas in the other it is maximized.

In the DSO case, $\mathbf{W}_w$ is a differential operator operating along the angle axes in the ADCIGs. Shen et al. (2003) demonstrate that this differential operator in the angle domain is equivalent to a zero-phase (i.e real diagonal) operator in the subsurface-offset domain. Consequently, the image perturbation ${}_i\widetilde{\Delta\mathbf{I}}$ and the background image ${}_{i-1}\mathbf{I}$ are, by construction, in phase with each other, and the back-projection operator provides useful slowness perturbations.

In contrast, in the MMF case the $\mathbf{W}_w$ operator stacks the background image over angles, whereas its adjoint $\mathbf{W}'_w$ spreads the result of stacking back along the angles. In the presence of large errors in the background velocity function, the ADCIGs show a significant curvature, and thus at wide-aperture angles the reflectors in the stacked image are shifted substantially with respect to the reflectors in the background image. Under these conditions, the relative phase differences between ${}_i\widetilde{\Delta\mathbf{I}}$ and ${}_{i-1}\mathbf{I}$ can exceed the narrow range in which the Born approximation is valid, and the MVA method based on the MMF objective function may never converge toward a satisfactory solution.

In the WEMVA method, the choice of the "image enhancing" operator $\mathcal{K}_w$ plays an important role, whereas $\mathbf{W}_w$ is set to be the identity operator. Ideally the target image would be a an image with flat ADCIGs and well-focused reflections in the physical space. Biondi and Sava (1999) proposed to use residual prestack migration (Section 11.2.2) to obtain the target image. Alternatively, residual moveout can be used in place of residual migration to flatten the ADCIGs, albeit without improving the focusing of the reflections (Section 11.2.1). In either case, an "optimal" value for the parameter ρ is measured for all the locations in the image space, and the background image is transformed by using a residual imaging operator ${}_i\mathbf{K}_w(\rho)$, which is a function of ρ. However, a residual imaging operator significantly shifts the image of the reflectors for which $\Delta\rho = \rho - 1$ is sufficiently large. For these reflectors, the relative phase differences between ${}_i\widetilde{\Delta\mathbf{I}}$ and ${}_{i-1}\mathbf{I}$ can easily exceed the narrow range in which the Born approximation is valid, and the WEMVA method based on the this choice of $\mathcal{K}_w$ may never converge toward a satisfactory solution. This problem is illustrated in the following sequence of four figures, and its solution is described in Section 12.4.

Illustration of wavefield MVA challenges in the presence of large velocity errors

Large velocity errors present challenges to both the MMF and WEMVA methods as defined in the previous section. The following four figures illustrate the problem faced by WEMVA with a simple example. Figure 12.8 shows three impulse responses of the migration operator. For the sake of simplicity the following figures show the results of zero-offset experiments. The velocity is constant, and the data are represented by an impulse in space and time. We consider two slowness models: the background slowness s, and the "correct" (perturbed) slowness $s_c = s + \Delta s$. The two slownesses are related by a scale factor $\rho = s/s_c$. For Figures 12.8 and 12.9 we consider $\rho = .999$ to ensure that we do not violate the limits imposed by the Born approximation.

Figure 12.8a shows the migration impulse response computed with the background slowness. Figure 12.8b shows the difference between the migration with the background slowness (Figure 12.8a) and the migration with the correct slowness s_c. Figure 12.8c shows the image perturbation computed by applying the forward operator $\mathbf{M}_w$ to the slowness perturbation Δs. Since the slowness perturbation is very small, the conditions imposed by the Born approximation are fulfilled, and the two images in Figures 12.8b and 12.8c are identical. The image perturbations are phase-rotated by 90 degrees relative to the background image.

Figure 12.9 illustrates the back-projection process. The image perturbations shown in Figures 12.9b and 12.9c are created by extracting a small subset from the impulse responses shown in Figures 12.8b and 12.8c, respectively. In this way, the data correspond to a single point on the surface, and the image perturbation corresponds approximately to a single point in the subsurface. Panel (b) shows the image perturbation computed as an image difference, whereas panel (c) shows the image perturbation computed with the forward operator $\mathbf{M}_w$. By back-projecting the image perturbations in Figures 12.9b and 12.9c with the adjoint operator $\mathbf{M}_w'$, we obtain identical "fat rays" shown in Figures 12.9d and 12.9e, respectively. The results of the back-projection of the image perturbations shown in Figures 12.9d and 12.9e are identical, because the slowness ratio ρ is close to one ($\rho = .999$). The phase differences

between the background image (Figure 12.9a) and the image perturbation computed as a difference between images (Figure 12.9b) are small, well within the limits imposed by the Born approximation.

In contrast, if we decrease ρ to $\rho = .83$, we obtain a significantly different, and troublesome, result, as shown in Figures 12.10 and 12.11. The panels in these two figures are homologous with the panels shown in Figures 12.8 and 12.9. In this case, the phase differences between the background image (Figure 12.11a) and the image perturbation computed as a difference between images (Figure 12.11b) are large. The image perturbation shown in Figure 12.11b contains two distinct events, whereas the image perturbation obtained by the forward operator $\mathbf{M}_w$ (Figure 12.11c) shows only one event. The only difference between the image perturbations shown in Figure 12.11c and in Figure 12.9c is a scale factor proportional to the ratio between the respective magnitudes of the slowness perturbations.

Because of the large phase differences between the background image and the image perturbation computed as a difference between images, the result computed by applying the adjoint operator $\mathbf{M}'_w$ to the image perturbation, shown in Figure 12.11b, is negatively affected by the limitation of the Born approximation. The polarity of the central part of the fat ray shown in Figure 12.11c is opposite to the polarity of the fat ray shown in Figure 12.11e, which is the same as the true slowness perturbation. We also see the two characteristic migration ellipsoidal side-events, indicating cycle-skipping (Woodward, 1992).

12.4 A robust WEMVA method

The root of the problem illustrated by the previous example is that the Born approximation is derived by linearizing the wavefield scattering phenomenon in terms of the amplitudes of the wavefield, whereas large velocity errors cause large phase errors. Therefore, when the events in the background image are significantly shifted with respect to the image perturbation, the Born approximation breaks down.

In this section, I present an alternative method for computing the image perturbation (Sava and Biondi, 2004a), that overcomes the limitations imposed by the Born approximation. The events in the image perturbation produced by this method are phase-rotated by ± 90 degrees, not phase shifted, with respect to the background image. The crucial component of the new method is the approximation of the residual prestack migration operator by its linearization with respect to the amplitude of the image. This amplitude linearization "parallels" the amplitude linearization performed when defining the linearized forward modeling operator $\mathbf{M}_w$. It produces image perturbations that can be effectively inverted by using $\mathbf{M}_w$ in a gradient-based algorithm.

The residual prestack migration operator $\mathbf{K}_w(\rho)$ can be expanded in Taylor series with respect to ρ as follows:

$$\mathbf{K}_w(\rho) \approx \widehat{\mathbf{K}}_w(\rho) = \mathbf{1} + \Delta\rho \left.\frac{d\mathbf{K}_w}{d\rho}\right|_{\rho=1} = \mathbf{1} + \Delta\rho \frac{d\mathbf{K}_w}{dk_z} \left.\frac{dk_z}{d\rho}\right|_{\rho=1} = \mathbf{1} + \widehat{\mathbf{K}}^d_w, \tag{12.28}$$

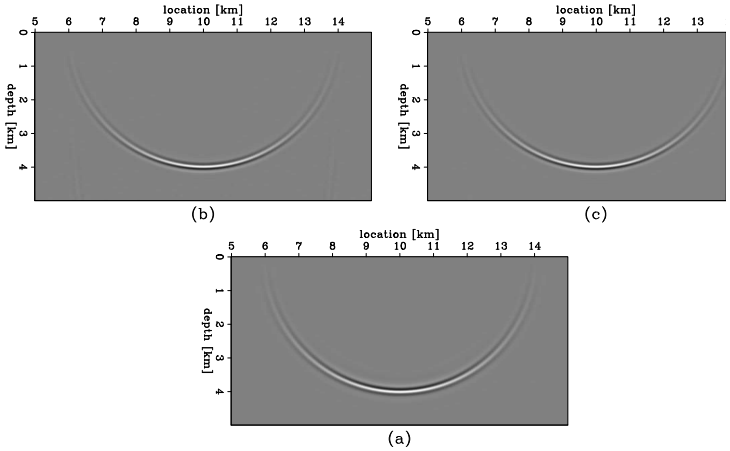


Figure 12.8: Comparison of image perturbations obtained as a difference between two migrated images (b) and as the result of the forward operator $\mathbf{M}_w$ applied to the known slowness perturbation (c). Panel (a) shows the background image corresponding to the background slowness. Since the slowness perturbation is small (0.1%), the image perturbations in panels (b) and (c) are practically identical. (This figure is from Sava (2004).) wemva-WEP1.imag [NR]

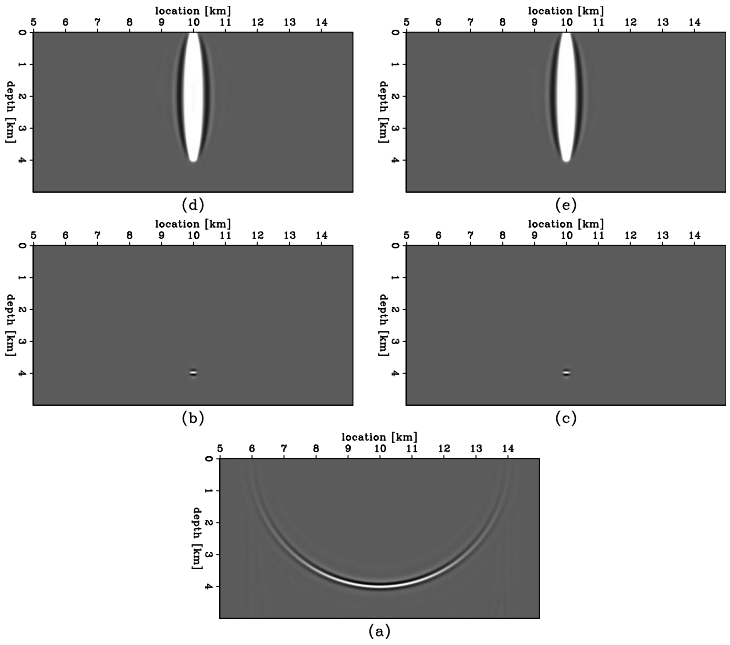


Figure 12.9: Comparison of back-projections of image perturbations into the slowness model using the adjoint operator $\mathbf{M}_w'$. Slowness perturbation (d) computed from image perturbations obtained as a difference between two migrated images (b). Slowness perturbation (e) computed from image perturbations obtained by applying the forward operator $\mathbf{M}_w$ to a known slowness perturbation (c). Panel (a) shows the background image corresponding to the background slowness. Since the slowness perturbation is small (0.1%), the image perturbations in panels (b) and (c), and the fat rays in panels (d) and (e) are practically identical. (This figure is from Sava (2004).) wemva-WEP1.rays.scaled [NR]

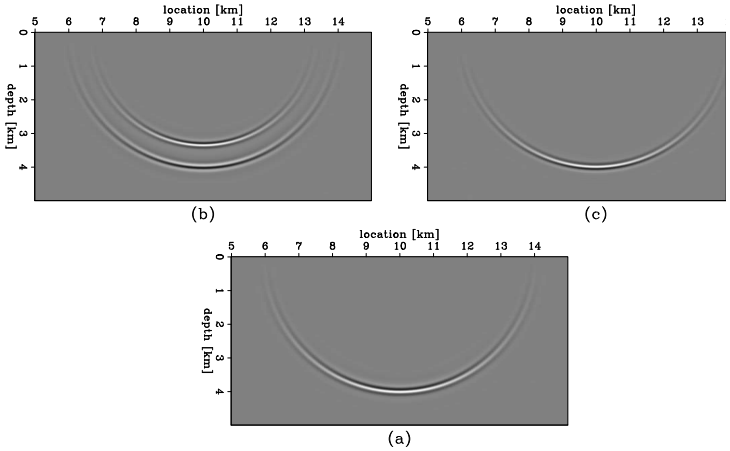


Figure 12.10: Comparison of image perturbations obtained as a difference between two migrated images (b) and as the result of the forward operator $\mathbf{M}_w$ applied to the known slowness perturbation (c). Panel (a) shows the background image corresponding to the background slowness. Since the slowness perturbation is large (20%), the image perturbations in panels (b) and (c) are different from each other. (This figure is from Sava (2004).) wemva-WEP2.imag [NR]

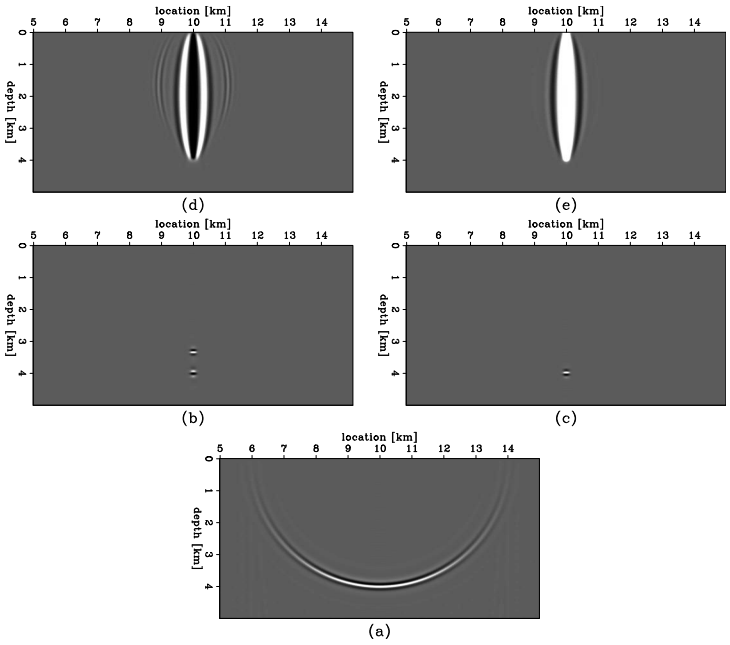


Figure 12.11: Comparison of back-projections of image perturbations into the slowness model using the adjoint operator $\mathbf{M}'_w$. Slowness perturbation (d) computed from image perturbations obtained as a difference between two migrated images (b). Slowness perturbation (e) computed from image perturbations obtained by applying the forward operator $\mathbf{M}_w$ to a known slowness perturbation (c). Since the slowness perturbation is large (20%), the image perturbations in panels (b) and (c) and the fat rays in panels (d) and (e) are substantially different. Panel (d) shows the typical behavior associated with the breakdown of the Born approximation. (This figure is from Sava (2004).) wemva-WEP2.rays.scaled [NR]

where the notation $\widehat{\mathbf{K}}_w^d$ reminds us that the operator is the result of differentiation. From expression (12.5), the image perturbation ${}_i\Delta\mathbf{I}$ can be expressed as the operator ${}_i\widehat{\mathbf{K}}_w^d$ applied to the background image ${}_{i-1}\mathbf{I}$; that is,

$$ {}_i\Delta\mathbf{I} = ({}_i\mathbf{K}_w - \mathbf{1})\ {}_{i-1}\mathbf{I} = {}_i\widehat{\mathbf{K}}_w^d\ {}_{i-1}\mathbf{I}. \tag{12.29}$$

Expression (12.28) suggests that the linearized residual migration operator ${}_i\widehat{\mathbf{K}}_w^d$ can be applied to the background image ${}_{i-1}\mathbf{I}$ as the cascade of two operators: the first is applied in the wavenumber domain, and the second is applied in the space domain.

The full expression for the first operator can be easily derived by taking the first derivative of the Stolt remapping of the vertical wavenumber, k_z, expressed in equation (11.14). This remapping can be written as follows:

$$k_z(\rho) = \sqrt{\frac{1}{\rho^2}\lambda - \frac{1}{4}\left|\mathbf{k_m}+\mathbf{k_h}\right|^2} + \sqrt{\frac{1}{\rho^2}\lambda - \frac{1}{4}\left|\mathbf{k_m}-\mathbf{k_h}\right|^2}\ , \tag{12.30}$$

where

$$\lambda = \frac{\left(k_{z_\rho}^2 + \left|\mathbf{k_m}\right|^2\right)\left(k_{z_\rho}^2 + \left|\mathbf{k_h}\right|^2\right)}{4k_{z_\rho}^2}. \tag{12.31}$$

The first derivative with respect to ρ of the mapping (12.30) is

$$\frac{dk_z}{d\rho} = -\frac{\lambda}{\rho^3}\left(\frac{1}{\sqrt{\frac{\lambda}{\rho^2} - \frac{1}{4}\left|\mathbf{k_m}+\mathbf{k_h}\right|^2}} + \frac{1}{\sqrt{\frac{\lambda}{\rho^2} - \frac{1}{4}\left|\mathbf{k_m}-\mathbf{k_h}\right|^2}}\right), \tag{12.32}$$

which when evaluated at $\rho = 1$ simplifies into

$$\left.\frac{dk_z}{d\rho}\right|_{\rho=1} = \frac{-\lambda}{\sqrt{\lambda - \frac{1}{4}\left|\mathbf{k_m}+\mathbf{k_h}\right|^2}} + \frac{-\lambda}{\sqrt{\lambda - \frac{1}{4}\left|\mathbf{k_m}-\mathbf{k_h}\right|^2}}. \tag{12.33}$$

Equation (12.33) expresses a wavenumber-dependent scaling of the migrated image. The scaling depends on the propagation direction of the source and receiver wavepaths in a constant-velocity medium. For each reflected event, this scaling is proportional to the sum of the downgoing and upgoing wavepath lengths, divided by the depth of the reflector. It is straightforward to verify that in the special case of flat reflectors, equation (12.33) is the linearization around $\rho = 1$ of the mapping functions presented in equations (11.15) and (11.16), and thus is related to the angle-domain RMO function expressed in equation (11.11).

The second operator, $d\mathbf{K}_w/dk_z$, is applied by taking the derivative of the weighted image with respect to k_z. It is efficiently applied in the space domain as follows:

$$\frac{d\mathbf{K}_w}{dk_z} = -iz. \tag{12.34}$$

The crucial component of the operator expressed in equation (12.34) is the 90-degree phase rotation performed by i. This phase rotation ensures coherent cross-correlation between the

background wavefield $\mathbf{P}$ and the scattered wavefield $\Delta\mathbf{P}'$, which is part of the "imaging" step of the adjoint operator $\mathbf{M}'_w$ expressed in equation (12.20). The depth-scaling is simply to restore the proper scaling of the downgoing and upgoing wavepath lengths that were "lost" in equation (12.33).

Finally, according to equation (12.28), the phase-rotated image is multiplied in the space domain by the value of $\Delta\rho$. This multiplication scales the image perturbation according to the measured curvature of the migrated ADCIGs. The events with large curvature in the ADCIGs (i.e. large velocity errors) are emphasized over events with a small curvature. At the limit, where the ADCIGs are flat in the migrated image, the image perturbation is zeroed.

The substitution of the linearized residual migration operator, $\mathbf{1}+\widehat{\mathbf{K}}_w$, for the residual migration operator, $\mathbf{K}_w$, achieves our goal of making the WEMVA method robust to large velocity errors. The shifts of the reflectors performed by $\mathbf{K}_w$ are replaced with the phase rotations of the reflectors performed by $\widehat{\mathbf{K}}_w$ This exchange makes the image perturbation computed using $\widehat{\mathbf{K}}_w$ consistent with the Born approximation used to define the linearized operator $\mathbf{M}_w$. The next numerical example demonstrates that, in the simple case of constant background slowness and constant velocity errors, the image perturbation computed by applying $\widehat{\mathbf{K}}_w$ is identical to the one computed using the forward linearized operator $\mathbf{M}_w$.

The application of the linearized residual prestack migration to the computation of image perturbations is illustrated by Figure 12.12. The panels in this figure are homologous with the panels shown in Figures 12.9 and 12.11. The slowness error is large, with $\rho = .83$, the same as for Figure 12.11. However, in this case the residual prestack migration operator is approximated by the the linearized residual migration operator $\widehat{\mathbf{K}}^d_w$. The image perturbation computed by applying $\widehat{\mathbf{K}}^d_w$ to the background image (panel b) is identical to the image perturbations computed by applying the forward operator $\mathbf{M}_w$ to the known slowness perturbation (c). Similarly, the results of the back-projection of the image perturbations using the adjoint operator $\mathbf{M}'_w$ are also identical. Figure 12.12. demonstrates that, at least for the simple example under study, the computation of the image perturbation by $\widehat{\mathbf{K}}^d_w$ overcomes the problems related to the Born approximation.

It should be noticed that applying the linearized residual migration operator $\widehat{\mathbf{K}}_w$ as a cascade of operators in the wavenumber and space domain assumes that these operators are stationary along both the wavenumber and space axes. This assumption is approximately fulfilled for the operators that are derived analytically and expressed in equations (12.33) and (12.34). In contrast, the measured $\Delta\rho$ function is not guaranteed to be smooth. In practical applications, it should be smoothed to assure its semi-stationarity on the scale of the wavelength of the reflectors.

The concepts applied to the derivation of the linearized residual-migration operator presented in this section can be also applied to the linearization of several residual prestack-imaging operators, such as the RMO functions introduced in Chapter 11, and the common-azimuth migration introduced in Chapter 7, as presented by Sava (2004). When the residual-imaging operator measures relative shifts between the normal-incidence events and the events at finite aperture and azimuth (e.g. RMO function), the expression of the linearized operator $\mathbf{M}_w$ should be modified to take into account this characteristic of the measured data. This

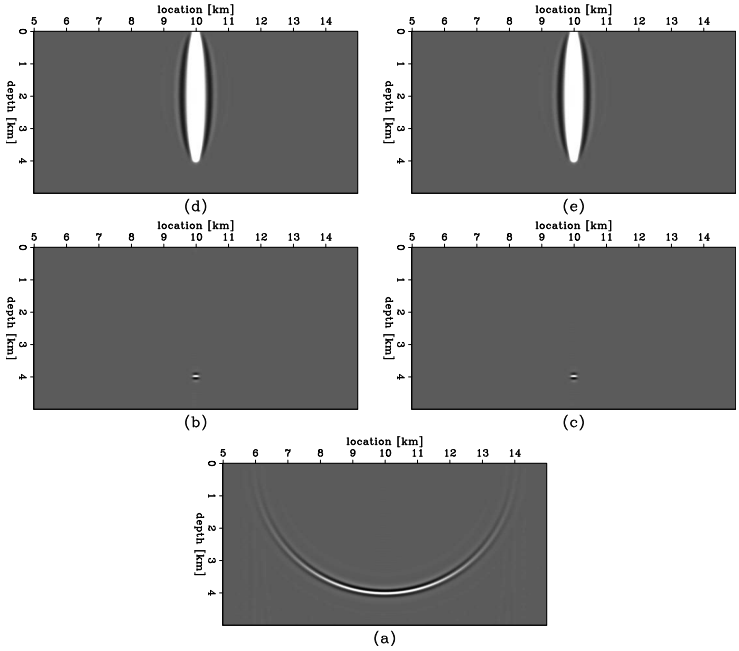


Figure 12.12: Comparison of image perturbations obtained by applying the linearized residual migration operator $\widehat{\mathbf{K}}_w^d$ (b), and by applying the forward operator $\mathbf{M}_w$ to the known slowness perturbation (c). Despite the fact that the slowness perturbation is large (20%), the image perturbations in panels (b) and (c) and the fat rays in panels (d) and (e) are practically identical, both in shape and in magnitude. (This figure is from **?**. wemva-WEP2.raan.scaled [NR]

task can be accomplished by adding another component to $\mathbf{M}_w$ that subtracts the image perturbation at normal incidence, following a theoretical development that is similar to the one followed in Section 10.4.2 for the traveltime tomography method, and in Section 11.4.2 for the ray-based MVA method.

Wavefield-continuation MVA methods hold the promise of improving the resolution of MVA methods, because they are, at least in theory, capable of using all the velocity information present in band-limited data, as illustrated by the wavepaths shown in Figure 12.2. However, by measuring the velocity errors using a residual-imaging operator defined by only one parameter (ρ), even if the parameter is spatially varying, we substantially limit our ability to exploit this potential advantage of wavefield methods over ray methods. This limitation can be an important shortcoming of the WEMVA method presented in this section, when compared with the DSO and MMF objective functions discussed previously in this chapter.

As I briefly discussed in Section 12.3, the DSO objective function has attractive properties of global convergence, because the gradient direction is well behaved even in presence of large velocity errors. In contrast, the MMF objective function cannot be used in presence of large velocity errors, but it has attractive local convergence properties, and it is sensitive to small relative shifts in the ADCIGs. These characteristics are complementary to the robust global convergence properties of the WEMVA method presented in this section. We can combine the stacking operator with a global residual imaging operator, $\mathbf{K}_w$, which removes the large shifts

from the migrated ADCIGs, and derive the following expression for computing the image perturbation ${}_i\Delta\mathbf{I}$:

$$ {}_i\Delta\mathbf{I} = \left\{ {}_i\widehat{\mathbf{K}}_w^d({}_i\rho) + \epsilon \left[{}_i\mathbf{K}_w{}^{-1}({}_i\rho)\mathbf{S}_w'\mathbf{S}_w\,{}_i\mathbf{K}_w({}_i\rho) - \mathbf{1} \right] \right\} {}_{i-1}\mathbf{I}, \tag{12.35} $$

where $\mathbf{S}_w$ its a stacking operator over angles, like the one used by the MMF objective function, and ϵ is an appropriate scaling factor. Notice that both the residual-imaging operator ${}_i\mathbf{K}_w$ and the linearized residual-imaging operator ${}_i\widehat{\mathbf{K}}_w^d$ are determined by the same slowness-ratio function ${}_i\rho$.

Expression (12.35) has attractive characteristics that should yield a velocity-estimation method that avoids the pitfalls of the Born approximation while achieving the high-resolution potential of wavefield methods. When the gathers display a curvature, but the residual moveouts are well corrected by the one-parameter residual-imaging operator ${}_i\mathbf{K}_w$, the image perturbation defined by equation (12.35) is equivalent to the one defined by equation (12.29). However, when the residual moveouts are not sufficiently corrected by the one-parameter residual-imaging operator ${}_i\mathbf{K}_w$ – that is, when after the application of ${}_i\mathbf{K}_w$ the migrated ADCIGs display small positive and negative shifts – the use of the image perturbation defined by equation (12.35) should enable the estimation of the velocity anomalies responsible for these small shifts.

At the limit, if the ADCIGs are close to flat, and the measured ${}_i\rho$ function is approximately equal to one everywhere, the contributions from the first operator $\left({}_i\widehat{\mathbf{K}}_w^d\right)$ in equation (12.35) become negligible, and the operator ${}_i\mathbf{K}_w$ becomes close to the identity operator. In this case, the gradient direction computed starting from the perturbation expressed in equation (12.35), and the gradient direction computed for the MMF functional [equation (12.27) with the plus sign and $\mathbf{W}_w = \mathbf{S}_w$] are the same, since the application of the adjoint operator $\mathbf{M}_w'$ to the background image ${}_{i-1}\mathbf{I}$ is approximately zero.

12.5 Examples of subsalt wave-equation MVA

Subsalt imaging is one of the most promising applications for wavefield MVA methods, as discussed in the introduction to this Chapter. Primarily, wavefield methods should provide more robust and stable convergence than ray methods in the subsalt environment because of the challenges presented by multi-pathing and shadow zones. Secondarily, they may provide higher-resolution estimates because of their potential at exploiting the frequency dispersion that is caused by complex velocity functions. Therefore, I will use two subsalt examples, one from a synthetic 2-D data set (Sigsbee 2A) and another from a 2-D line taken from a 3-D data set acquired in the deep waters of the Gulf of Mexico.

12.5.1 Synthetic example

The data set that I use in this example is based on the same velocity model (Sigsbee 2A) that I used to compute the wavepaths shown in Figures 12.1 and 12.2. This model has been designed

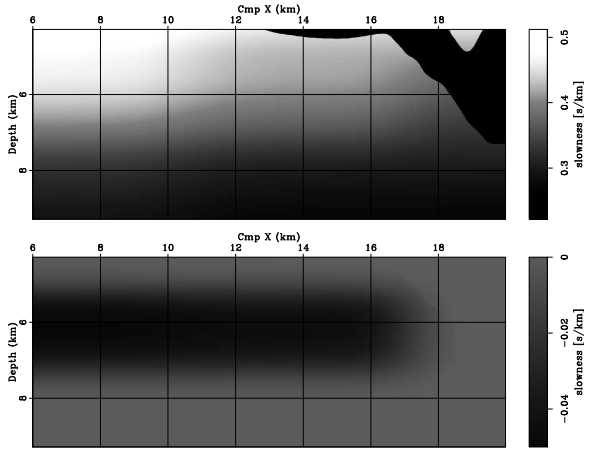


Figure 12.13: Sigsbee 2A synthetic model. The background slowness model (top) and the correct slowness perturbation (bottom). (This figure is from Sava (2004).) wemva-SIG.slo [NR]

to be similar to real salt bodies found in the deep waters of the Gulf of Mexico (Paffenholz et al., 2002). In this section, I concentrate on the lower part of the model, under the salt body. The top panel in Figure 12.13 shows the background slowness model, and the bottom panel shows the slowness perturbation of the background model relative to the correct slowness. Therefore, I simulate a common subsalt velocity analysis situation, where the shape of the salt is known, but the smoothly varying subsalt slowness is not fully known.

The original data set was computed with a typical marine off-end recording geometry. Preliminary studies of the data demonstrated that in some areas the complex overburden causes events to be reflected with negative reflection angles (i.e. the source and receiver wavepaths cross before reaching the reflector). To avoid losing these events, I applied the reciprocity principle and created a split-spread data set from the original off-end data set. This modification of the data set leads to the computation of symmetric ADCIGs that are easier to visually analyze than the typical one-sided ADCIGs obtained from marine data. Therefore, I display the symmetric ADCIGs in Figure 12.16 and Figures 12.20-12.22. Doubling the dataset also doubles the computational cost of the velocity-estimation process.

Figure 12.14 shows the migrated image using the correct slowness model. The top panel shows the zero offset of the prestack migrated image, and the bottom panel shows ADCIGs at equally spaced locations in the image. Each ADCIG corresponds roughly to the location right above it.

This image highlights several characteristics of this model that make it a challenge for migration velocity analysis. Most of them are related to the complicated wavepaths in the subsurface under rough salt bodies. First, the angular coverage under the salt ($x_m > 11$ km) is much smaller than in the sedimentary section uncovered by salt ($x_m < 11$ km). Second, the subsalt region is marked by many illumination gaps or shadow zones, the most striking being located at $x_m = 12$ and $x_m = 19$ km. In Section 11.2.2 I used the same synthetic dataset to illustrate the illumination problem caused by salt edges. The main consequence is that velocity analysis in the poorly illuminated areas is much less constrained than in the well illuminated zones.

The process starts by migrating the data with the background slowness (Figure 12.15). As

before, the top panel shows the zero offset of the prestack migrated image, and the bottom panel shows ADCIGs at equally spaced locations in the image. Since the migration velocity is incorrect, the image is defocused, and the angle-gathers show significant moveout. Furthermore, the diffractors at depths $z = 7.5$ km and the fault at $x_m = 15$ km are defocused.

As described in Section 11.2.2, prestack residual migration is performed for several values of the slowness-ratio parameter ρ within the range of 0.625 and 1.1, which ensures that a fairly wide range of the velocity space is spanned. Although residual migration operates on the entire image globally, for illustration purposes I extract one gather at $x_m = 10$ km. Figure 12.16 shows at the top the ADCIGs for all velocity ratios and at the bottom the semblance panels computed from the ADCIGs. The semblance maxima are picked at all locations and all depths (Figure 12.17), together with an estimate of the reliability of every picked value. This estimate is used to weight the image residuals during inversion. These weights enter the WEMVA objective function expressed in equation (12.5) as a diagonal weighting matrix, $\mathbf{W}_w$. The panel at the bottom of Figure 12.17 shows the weights. Where the image is poorly illuminated, the weights are low, because the ADCIGs are affected by illumination artifacts. This a typical example where poor illumination interferes with our ability to estimate velocity.

Based on the picked velocity ratio, the image perturbation $\Delta\mathbf{I}$ is computed by using equation (12.29), and the slowness perturbation is estimated by minimizing the objective function (12.5). The bottom panel of Figure 12.18 shows the estimated slowness perturbation. For comparison, the top panel of Figure 12.18 shows the correct slowness perturbation relative to the correct slowness. We can clearly see the effects of different angular coverage in the subsurface: at $x_m < 11$ km, the inverted slowness perturbation is better constrained vertically than it is at $x_m > 11$ km.

Finally, the data are remigrated with the updated slowness model (Figure 12.19). As before, the top panel shows the zero offset of the prestack migrated image, and the bottom panel shows ADCIGs at equally spaced locations in the image. With this updated velocity, the reflectors have been repositioned to their correct locations, the diffractors at $z = 7.5$ km are focused, and the ADCIGs are flatter than in the background image, indicating that the slowness update has improved the quality of the migrated image.

Figures 12.20-12.22 enable a more detailed analysis of the results of the WEMVA method by displaying the ADCIGs at three locations in the image. In each figure, the panels correspond to migration with the correct slowness (left), the background slowness (center), and the updated slowness (right). Figure 12.20 corresponds to an ADCIG at $x_m = 8$ km, in the region that is well illuminated. The angle gathers are clean, with clearly identifiable moveouts that are corrected after inversion. Figure 12.21 corresponds to an ADCIG at $x_m = 10$ km, in the region with illumination gaps, clearly visible on the strong reflector at $z = 9$ km, at a reflection aperture angle of about 20 degrees. The gaps are preserved in the ADCIG from the image migrated with the background slowness, but the moveouts are still easy to identify and correct. Finally, Figure 12.22 corresponds to an ADCIG at $x_m = 12$ km, in a region that is poorly illuminated. In this case, the ADCIG is much noisier and the moveouts are harder to identify and measure. This region also corresponds to the lowest reliability, as indicated by the low weight

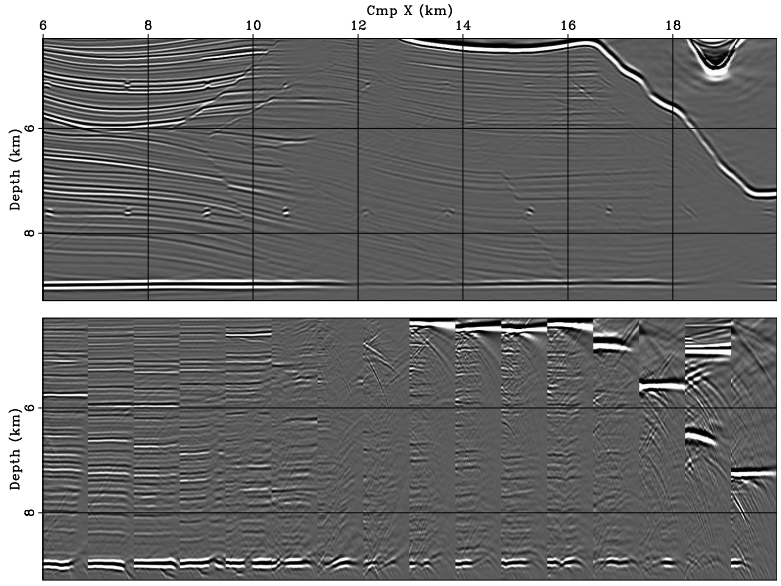


Figure 12.14: Migration with the correct slowness. The zero offset of the prestack migrated image (top) and ADCIGs at equally spaced locations in the image (bottom). Each ADCIG corresponds roughly to the location right above it. (This figure is from Sava (2004).) wemva-SIG.imgC [NR]

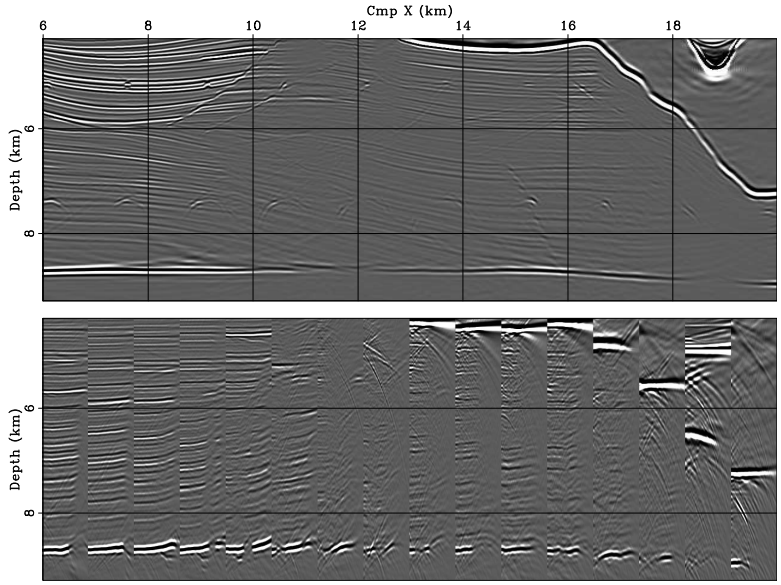


Figure 12.15: Migration with the background slowness. The zero offset of the prestack migrated image (top) and ADCIGs at equally spaced locations in the image (bottom). Each ADCIG corresponds roughly to the location right above it. (This figure is from Sava (2004).) wemva-SIG.img1 [NR]

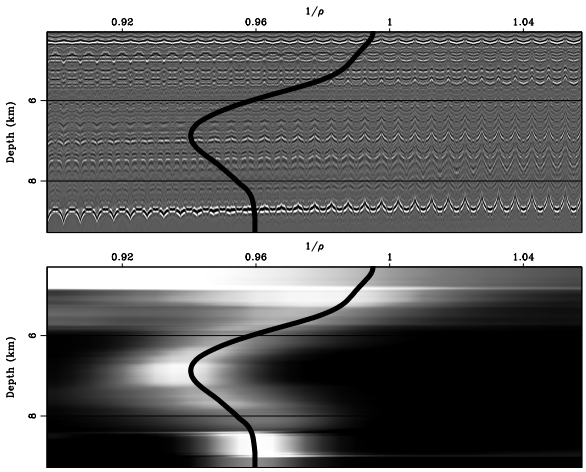


Figure 12.16: Residual migration for an ADCIG at $x_m = 10$ km. The top panel shows ADCIGs for all values of the inverse of the velocity ratio $1/\rho$, and the bottom panel shows semblance panels used for picking. All gathers are stretched along the depth axis to eliminate the vertical movement corresponding to different migration velocities. The overlaid line indicates the picked values at all depths. (This figure is from Sava (2004).) wemva-SIG.srm.fix [NR]

of the picks (Figure 12.17). The gathers in this region contribute less to the inversion, and the resulting slowness perturbation is mainly controlled by regularization. Despite the noisier gathers, after slowness update and re-migration, we recover an image reasonably similar to the one obtained by migration with the correct slowness.

A simple visual comparison of the middle panels with the right and left panels in Figures 12.20-12.22 unequivocally demonstrates that the WEMVA method overcomes the limitations related to the linearization of the wave equation with the first-order Born approximation. The images obtained using the initial velocity model (middle panels) are vertically shifted by several wavelengths with respect to the images obtained using the true velocity (left panels) and the estimated velocity (right panels). If the Born approximation limited the magnitude and spatial extent of the velocity errors that could be estimated with the WEMVA method, we would have been unable to estimate a velocity perturbation sufficient to improve the ADCIGs from the middle panels to the right panels.

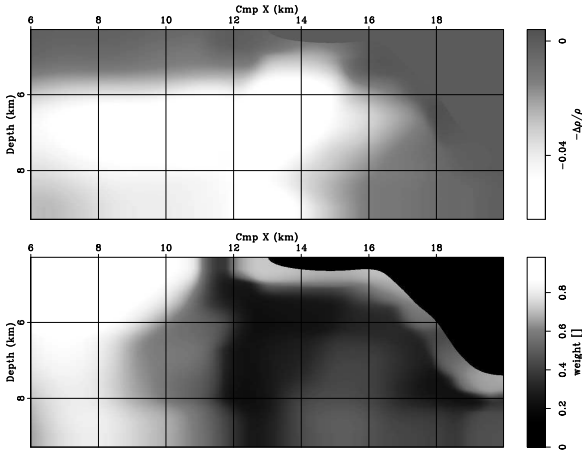


Figure 12.17: The top panel shows the values of $-\Delta\rho/\rho$ at all locations, and the bottom panel shows a weight indicating the reliability of the picked values at every location. The picks in the shadow zone around $x_m = 12$ km are less reliable than the picks in the sedimentary region around $x_m = 8$ km. All picks inside the salt are disregarded. (This figure is from Sava (2004).) wemva-SIG.pck [NR]

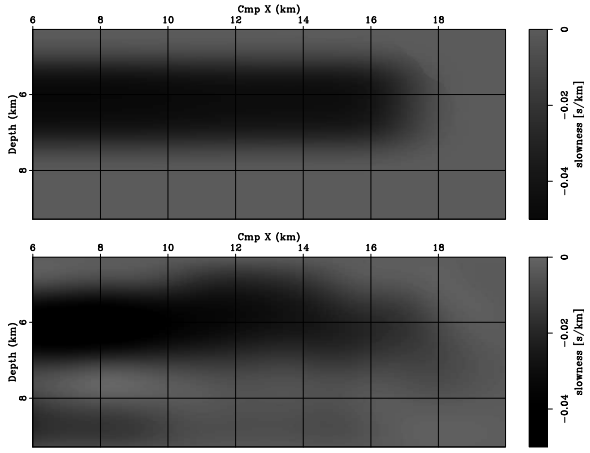


Figure 12.18: The correct slowness perturbation (top) and the inverted slowness perturbation (bottom). (This figure is from Sava (2004).) wemva-SIG.dsl [NR]

12.5.2 Gulf of Mexico data example

The next example concerns a 2-D line extracted from a 3-D subsalt dataset from the Gulf of Mexico. I follow the same methodology as used for the preceding synthetic example. In this case, however, we run several non-linear iterations of WEMVA, each involving wavefield linearization, residual migration and inversion.

Figure 12.23 (top) shows the image migrated with the background velocity superimposed on the background slowness. This image serves as a reference against which we check the results of our velocity analysis. Two regions of interest are labeled A and B in the figure. The right edge of the model corresponds to a salt body. The top edge of the image is not at the surface, because the surface data were datumed to a depth below the well-imaged overhanging salt body.

The procedure is like the one in the previous example. We run residual migration and analyze the moveouts of ADCIGs. This analysis is repeated at every location, from which we obtain two maps: a map of the residual migration parameter $\Delta\rho$ at every location in the image (Figure 12.23 middle), and a map of the weight indicating the reliability of the measured values of $\Delta\rho$ (Figure 12.23 bottom). In the map shown in the middle panel of Figure 12.23, the whiter regions indicate flatter ADCIGs. The stack of the background image is overlaid on to both of these panels to facilitate identification of image features. Next, we generate an image perturbation based on the $\Delta\rho$ values shown in Figure 12.23 (middle) and estimate the slowness perturbation using the weights shown in Figure 12.23 (bottom), as a measure of the reliability of the measured image perturbation.

The results obtained after two non-linear iterations of WEMVA are shown in Figure 12.24. As in Figure 12.23, the three panels show the migrated image superimposed on slowness (top), the values of $\Delta\rho$ measured from the migrated image (middle), and weights of the image perturbations (bottom). Two regions in which changes occur are labeled A and B.

The residual migration picks converge toward $\Delta\rho = 0$, indicating flatter ADCIGs, and therefore better-focused images. Reflectors in both regions shift vertically, according to the slowness changes. A notable feature is the improved continuity of the strongest reflectors in

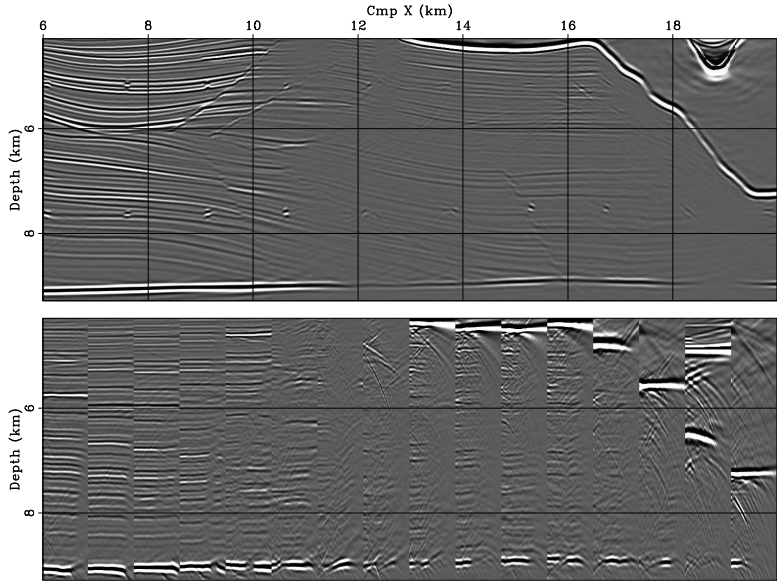


Figure 12.19: Migration with the updated slowness. The zero offset of the prestack migrated image (top) and ADCIGs at equally spaced locations in the image (bottom). Each ADCIG corresponds roughly to the location right above it. (This figure is from Sava (2004).) wemva-SIG.img2 [NR]

the region labeled B.

The comparison of the images and ADCIGs shown in Figures 12.25 and 12.26 confirm that the two non-linear iterations of the WEMVA process yield an improvement in the velocity function. The ADCIG obtained by using the starting migration velocity (bottom of Figure 12.25) "smiled" upward in region A, and "frowned" downward in region B. The ADCIG obtained by using the estimation results (bottom of Figures 12.26) are flatter in both of these regions. In the poorly illuminated area (left of "B") the ADCIGs are affected by artifacts, but they seem to be slightly more coherent in Figure 12.26 than in Figure 12.25. Correspondingly, the weights shown in the bottom panel of Figure 12.24 are slightly higher than the weights shown the bottom panel of Figure 12.23. However, in the poorly illuminated area, the weights are still fairly low, indicating that the WEMVA process is only partially using the velocity information provided in the reflectors.

As in the synthetic example shown in the previous section, poor illumination hampers the velocity estimation. Whereas WEMVA provides a robust method for inverting velocity measurements of reflections that have propagated through complex overburden, it is generally less effective in poorly illuminated regions.

REFERENCES

Biondi, B., and Sava, P., 1999, Wave-equation migration velocity analysis: 69th Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 1723–1726.

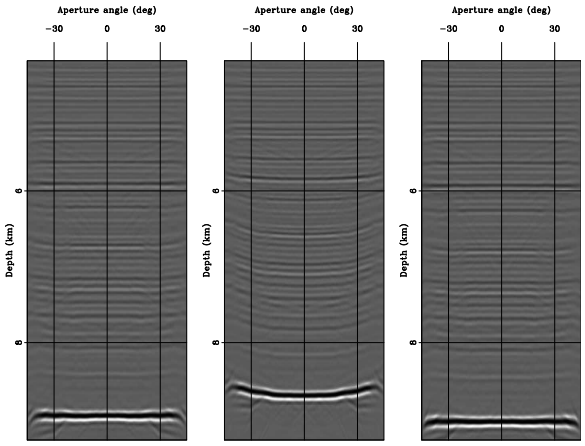


Figure 12.20: Angle-domain common-image gathers at $x = 8$ km. Each panel corresponds to a different migration velocity: migration with the correct velocity (left), migration with the background velocity (center) and migration with the updated velocity (right). (This figure is from Sava (2004).) wemva-SIG.ang08.fix [NR]

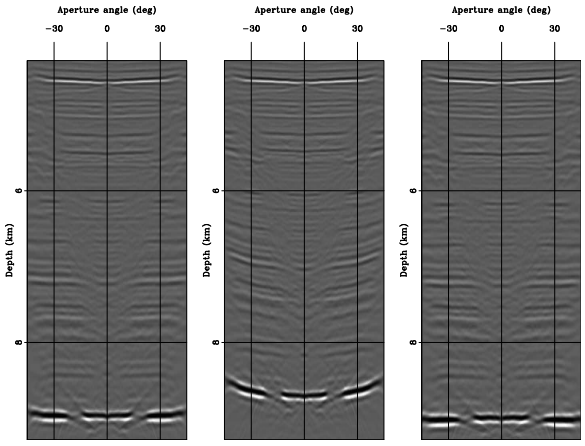


Figure 12.21: Angle-domain common-image gathers at $x = 10$ km. Each panel corresponds to a different migration velocity: migration with the correct velocity (left), migration with the background velocity (center) and migration with the updated velocity (right). (This figure is from Sava (2004).) wemva-SIG.ang10.fix [NR]

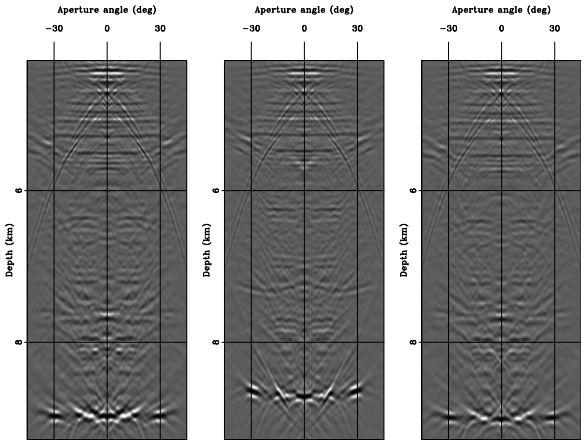


Figure 12.22: Angle-domain common-image gathers at $x = 12$ km. Each panel corresponds to a different migration velocity: migration with the correct velocity (left), migration with the background velocity (center) and migration with the updated velocity (right). (This figure is from Sava (2004)). wemva-SIG.ang12.fix [NR]

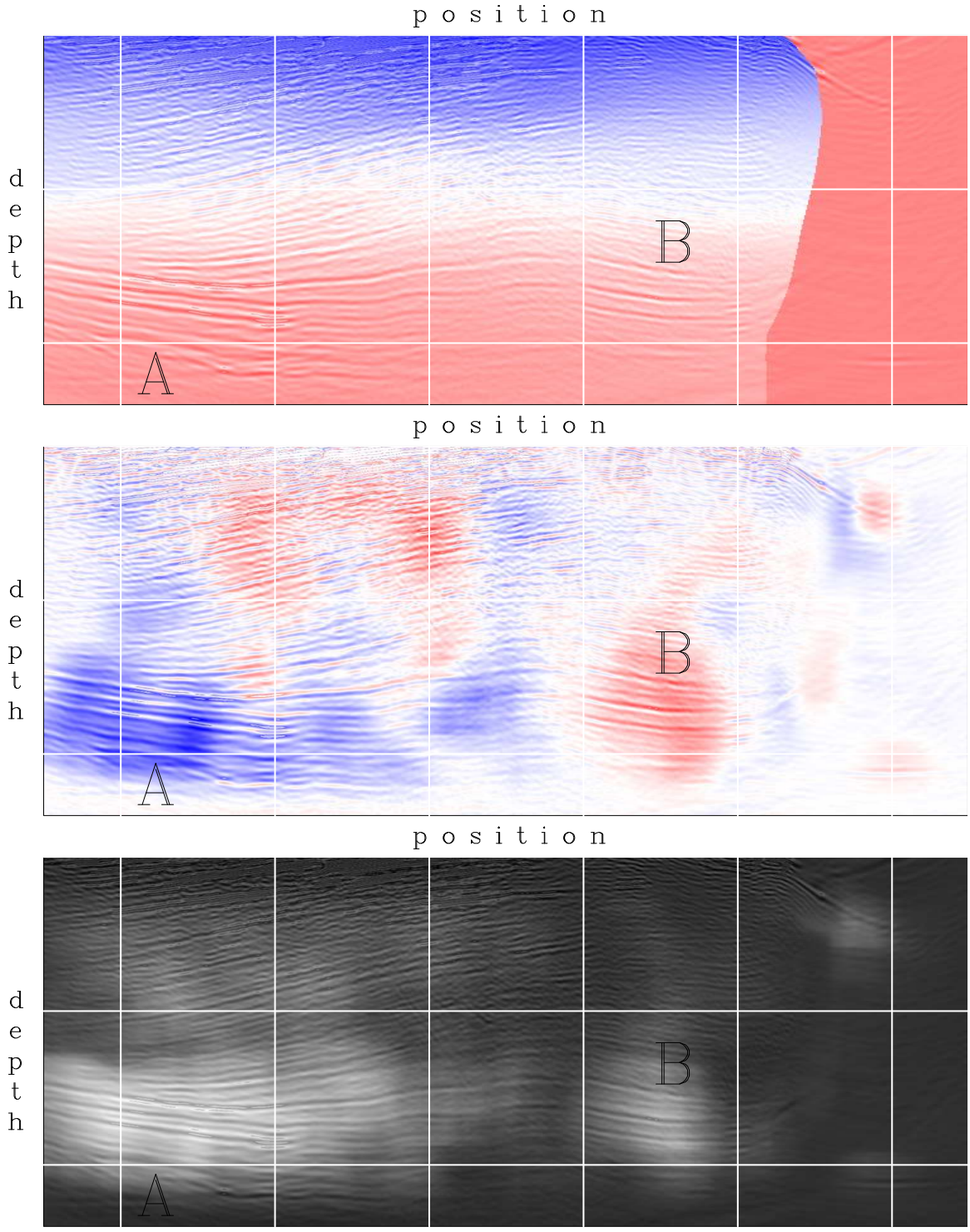


Figure 12.23: Gulf of Mexico data. Migrated image superimposed on slowness (top), residual migration picks (middle), and picking weight (bottom). The migration corresponds to the background slowness. (This figure is from Sava (2004).) wemva-BPGOM.it0 [NR]

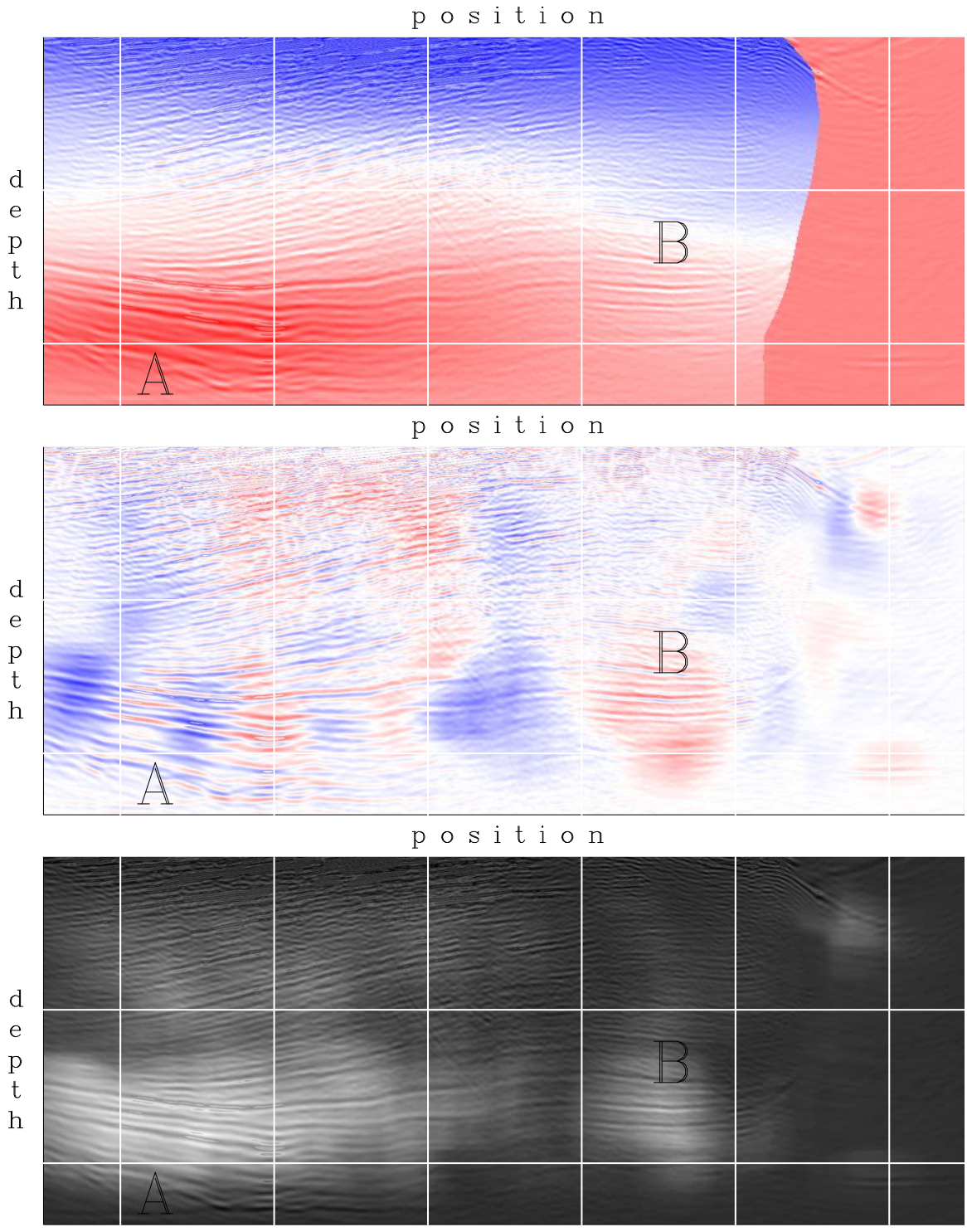


Figure 12.24: Gulf of Mexico data. Migrated image superimposed on slowness (top), residual migration picks (middle), and picking weight (bottom). The migration corresponds to the updated slowness after two iterations. Compare with Figure 12.23. (This figure is from Sava (2004).) wemva-BPGOM.it2 [NR]

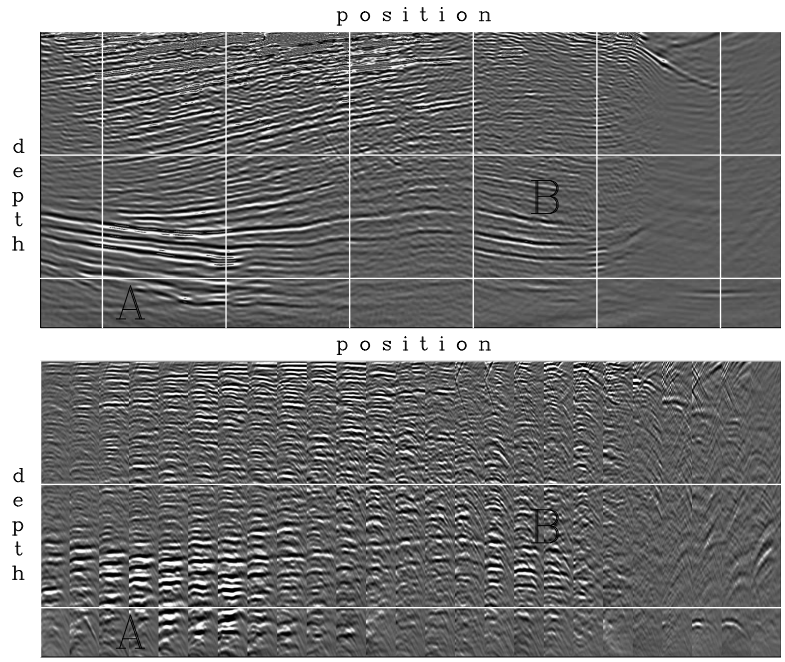


Figure 12.25: Migration results of the Gulf of Mexico data using the background slowness. Stack of the migrated image (top), and ADCIGs at regularly spaced locations (bottom). (This figure is from Sava (2004).) wemva-BPGOM.iman0 [NR]

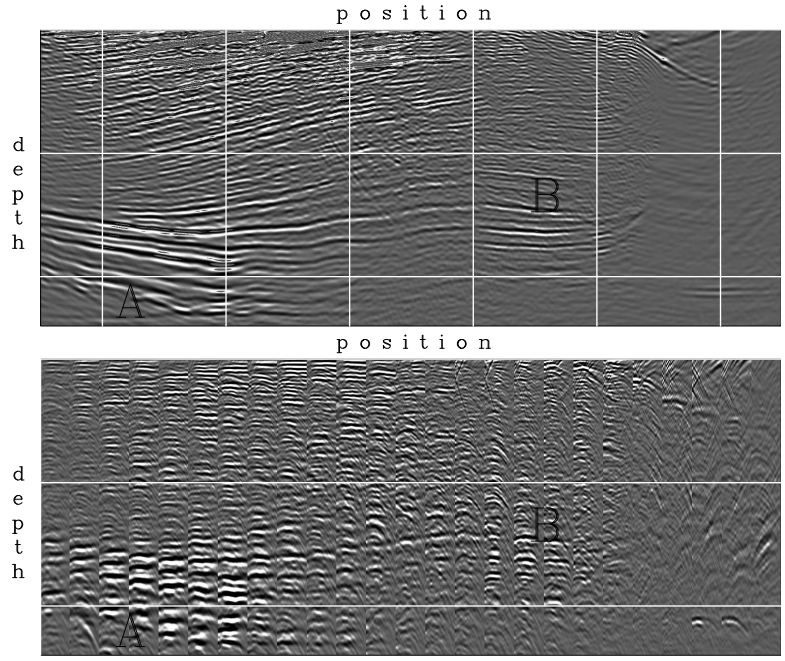


Figure 12.26: Migration results of the Gulf of Mexico data using the updated slowness after two WEMVA iterations. Stack of the migrated image (top), and ADCIGs at regularly spaced locations (bottom). (This figure is from Sava (2004).) wemva-BPGOM.iman2 [NR]

Chavent, G., and Jacewitz, C. A., 1995, Determination of background velocities by multiple migration fitting: Geophysics, **60**, 476–490.

Claerbout, J. F., 2004, Image Estimation by Example: `http://sepwww.stanford.edu/sep/prof/index.html`.

Clayton, R. W., and Stolt, R. H., 1981, A Born-WKBJ inversion method for acoustic reflection data: Geophysics, **46**, 1559–1567.

Cohen, J. K., and Bleistein, N., 1979, Velocity inversion procedure for acoustic waves: Geophysics, **44**, 1077–1087.

Dahlen, F. A., Hung, S. H., and Nolet, G., 2000, Frechet kernels for finite frequency traveltimes—I. Theory: Geophys. J. Int., **141**, 157–174.

Devaney, A. J., 1981, Inverse-scattering theory within the Rytov approximation: Optics Letters, **6**, 374–376.

Gill, P. E., Murray, W., and Wright, M. H., 1981, Practical optimization: Academic Press.

Guitton, A., and Symes, W. W., 2003, Robust inversion of seismic data using the Huber norm: Geophysics, **68**, 1310–1319.

Huang, L. Y., Fehler, M. C., and Wu, R. S., 1999, Extended local Born Fourier migration method: Geophysics, **64**, 1524–1534.

Hung, S. H., Dahlen, F. A., and Nolet, G., 2000, Frechet kernels for finite frequency traveltimes—II. Examples: Geophys. J. Int., **141**, 175–203.

Lions, J. L., 1971, Optimal control of systems governed by partial differential equations: Spring Verlag.

Lo, T. W., and Inderweisen, P. L., 1994, Fundamentals of seismic tomography:, Tulsa, OK.

Lomax, A., 1994, The wavelength-smoothing method for approximating broad-band wave propagation through complicated velocity structures: Geoph. J. Int., **117**, 313–334.

Marquering, H., Dahlen, F. A., and Nolet, G., 1999, Three-dimensional sensitivity kernels for finite-frequency traveltimes: the banana-doughnut paradox: Geophys. J. Int., **137**, 805–815.

Mora, P. R., 1987, Nonlinear two-dimensional elastic inversion of multioffset seismic data: Geophysics, **52**, 1211–1228.

Paffenholz, J., McLain, B., Zaske, J., and Keliher, P., 2002, Subsalt multiple attenuation and imaging: Observations from the Sigsbee2B synthetic dataset: 72nd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 2122–2125.

Pratt, R. G., 1999, Seismic waveform inversion in the frequency domain, part 1: Theory and verification in a physical scale model: Geophysics, **64**, 888–901.

Sava, P., and Biondi, B., 2004a, Wave-equation migration velocity analysis—I: Theory: Geophysical Prospecting, accepted for publication.

Sava, P., and Biondi, B., 2004b, Wave-equation migration velocity analysis—II: Examples: Geophysical Prospecting, accepted for publication.

Sava, P., and Biondi, B., 2004c, Sensitivity kernels for wave-equation migration velocity analysis: SEP-Report, **115**, 199–212.

Sava, P., 2004, Migration and velocity analysis by wavefield extrapolation: Ph.D. thesis, Stanford University.

Shen, P., Symes, W. W., and Stolk, C. C., 2003, Differential semblance velocity analysis by wave-equation migration: 73rd Ann. Internat. Meeting, Soc. of Expl. Geophys., Expanded Abstracts, 2132–2135.

Stolt, R. H., and Benson, A., 1986, Seismic migration - theory and practice: Geophysical Press, London - Amsterdam.

Stolt, R. H., and Weglein, A. B., 1985, Migration and inversion of seismic data: Geophysics, **50**, 2458–2472.

Symes, W. W., and Carazzone, J. J., 1991, Velocity inversion by differential semblance optimization: Geophysics, **56**, 654–663.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.

Woodward, M., 1990, Wave equation tomography: Ph.D. thesis, Stanford University.

Woodward, M. J., 1992, Wave-equation tomography: Geophysics, **57**, 15–26.

Chapter 1

Seplib3d: a software package for processing 3-D data

To process real 3-D data sets, that are typically recorded with irregular geometries and have huge sizes, it is necessary to use software tools that are specifically designed for 3-D data. Older software packages that were designed for 2-D data are either not flexible enough to handle irregular geometries, or they allow only a sequential access to the data, that is too inefficient.

Seplib3d is an extension of Seplib (Dellinger and Tálas, 1992; Claerbout, 1991), a SEP's software package that has been successfully used for prototyping 2-D seismic algorithms for many years. However, Seplib usefulness is fundamentally limited to processing regularly sampled data. This limitation is too restrictive when tackling problems in 3-D seismic and problems that involve geophysical data other than seismic. In Seplib, few parameters defined in the history file are sufficient to describe the data geometry, since it is assumed to be regular. In Seplib3d, to describe the irregular data geometry, we associate each seismic trace with a trace header, as is done in the SEGY data format, and in its many derivatives. However, to enable users and programmers to deal with irregularly sampled data with the same simplicity and efficiency that is characteristic of Seplib, Seplib3d introduces the following two principles:

- Separate the geometry information from the seismic data. This simple but powerful idea is crucial for efficiently processing large amount of data, such as in 3-D prestack data sets. It allows to minimize the access to the usually bulky seismic data files, while performing many useful operations on the trace headers and on specific subsets of the seismic traces.

- Exploit as much as possible the existing regularity in the data geometry. Regularity is important when analyzing and visualizing the data; further, it helps the development of simple and efficient code. Seplib3d "regularizes" an irregularly sampled data sets by associating the data traces with a uniformly sampled multi-dimensional grid. This gridding information is then exploited by Seplib3d application and utility programs to efficiently select and access the seismic traces.

Another important characteristic of Seplib3d is that it is a generalization of Seplib and not a completely new system. There are many good reasons for this choice. From the user point of view, it enables users familiar with Seplib to quickly master Seplib3d. Further, it enables Seplib3d to leverage the considerable amount of coding and brain power that went into Seplib. In particular we use the Seplib routines for accessing files (both ASCII and binary), and build Seplib3d capabilities on the top of these routines.

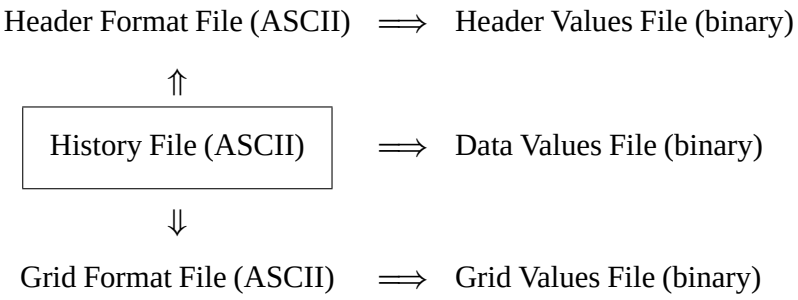
In this Appendix we introduce the fundamental concepts underlying Seplib3d. We define the data format, describe the essential utilities, and the application program interface (API). To illustrate the concept we also present an example of using Seplib3d on a subset of a real 3-D marine survey.

1.1 Data Format

This section describes the data format of a Seplib3d data set. A Seplib3d data set is defined as the collection of all the files (ASCII and binary) that contain the information relevant to the Geophysical data set.

1.1.1 Structure of a Seplib3d data set

A "complete" Seplib3d data set is made of 6 files, three ASCII files and three binary files. With the exception of the **History File**, the existence of all the other files is optional. The six files are connected to each other through pointers contained in the ASCII files. These pointers can be either simple UNIX file paths or more general URLs (the URL option is not implemented yet!). The path to the **Header Format File** (HFF) is specified by the value of the `hff` parameter in the History File. The path to the **Grid Format File** (GFF) is specified by the value of the `gff` parameter in the History File. Following is a brief "graphical" description of the connectivity among the 6 files, with the arrows $\Longrightarrow$ representing the ASCII pointers.



In addition to the links to the other files, the History File contains the processing history of the data set. The **Data Values File** (DVF) is defined as collection of fixed length records (**data records**) that contain the data values. Typically the data records are seismic traces. The header values are stored in the **Header Values File** (HVF), that is defined as a collection of fixed length records (**header records**) describing the geometry and properties of the associated data

records. The header parameters are described in the Header Format File by a table of **header keys**. A header keys specifies the name of the header parameter (**key name**), its data type (**key type**), and its position in the header record (**key index**). The association between the header records and the data records is described below in Section 1.1.3. If the data set has been binned on a regularly sampled grid, the **Grid Format File** contains the description of the grid. The **Grid Value File** contains the mapping information between the **grid cells** and the corresponding header records. The Grid Value File does not exist if the gridding is regular; that is, there is a one-to-one correspondence between grid cells and header records.

1.1.2 Data and Headers Coordinate Systems

The History File describes the **Data Coordinate System**; that is, it defines the time axis of the seismic traces, and their organization as collection of traces. To define the Data Coordinate System the History File contains the usual Seplib parameters `n`*i*, `o`*i*, `d`*i*, `label`*i* (where *i*=[1,2,3,...]). The length of the axes in the Data Coordinate System must be constant and is given by the values of the respective `n`*i* parameter. The number of data values in a data record is given by `n1` and the the number of data records is equal to the product (`n2`$*$...$*$`n`*i*$*$...).

The Header Format File contains also the usual Seplib parameters `n`*i*, `o`*i*, `d`*i*, `label`*i* (where *i*=[1,2,3,...]) describing the **Header Coordinate System**. The number of header keys in the header records is given by `n1` and the the number of header records is equal to the product (`n2`$*$...$*$`n`*i*$*$...).

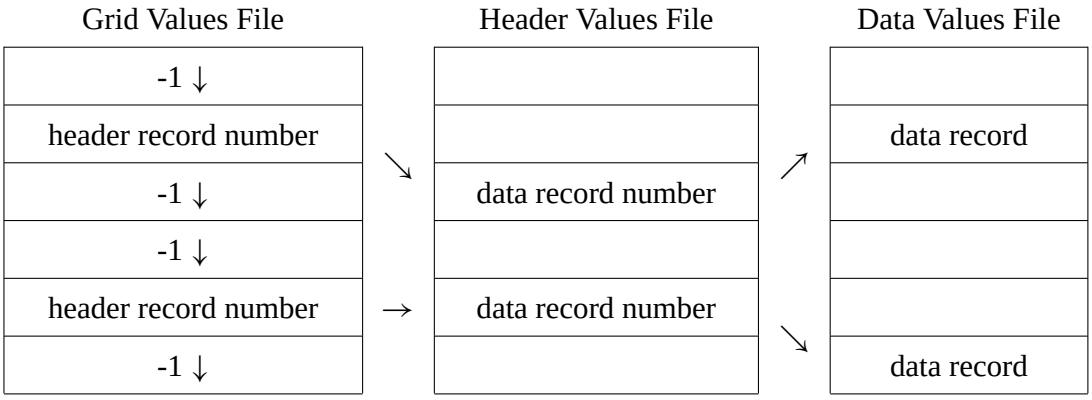
1.1.3 Mapping between the header records and the data records

In general, the order and number of the data records stored in the Data Values File may be different than the order and number of the header records stored in the Header Values File. This happens, for example, if the Header Values File has been reshuffled (e.g. sorted or transposed) while the Data Values File was left untouched. Whether the data and header records are in the same order is indicated by the value of the integer parameter `same_record_number` in the History File. The value of `same_record_number` is equal to `1` if the records are in the same order, and equal to `0` if they are not. If `same_record_number` is missing from the History File it is defaulted to `1`. When the data and header records are in the same order (`same_record_number=1`), the association between header records and data records is given by the positions of the records in the respective binary files, and the Data Coordinate System coincides with the the Header Coordinate System. If data and header records are in different order the association between data records and header records is assured by the reserved header key `data_record_number`, that contains the **data record number** of the associated data record. The value of the data record number is defined as equal to the position of the data record in the Data Values File.

1.1.4 Gridding information

The Grid Format File and the associated Grid Values File define the **Grid Coordinate System** and they contain the information about coordinates of each header records in the Grid Coordinate System. The Grid Coordinate System is a regularly sampled coordinate system defined by the the parameters `n`*i*`_grid, o`*i*`_grid, d`*i*`_grid, label`*i*`_grid` (where *i*=[1,2,3,...]) in the conventional Seplib style. The mapping between the grid cells and the header record can be either regular or irregular. A gridding is regular if for each grid cell in the Grid Coordinate System exists a header record, and vice versa, for each header record exists a grid cell. The grid cells and the header records are connects by tables of pointers to header records. These tables have an entry for each grid cell, containing the **header record number** of the corresponding header record. The value of the header record number is equal to the position of the corresponding header record in the Header Values File. Notice that if `same_record_number=0` the header record numbers are different from the data record numbers of the associated data record.

If the gridding is irregular, there are grid cell for which there is no associated header record. For these cells the pointer in the header record number tables are null (equal to `-1`). The following schematic illustrates the double-pointer mechanism that connects grid cells to data records, through the header records.



If the data is irregularly gridded the header record number tables are encoded in the Grid Values File. The format of the encoded tables is variable, and can be different from file to file. However, the programming interfaces for accessing the header record number are well defined and independent on the encoding. The encoding is variable because the optimal encoding strongly depends on the sparsity of the grid cells within the Grid Coordinate System, and thus depends on the particular way the header records were binned into the Grid Coordinate System. The gridding tables can be stored and retrieved by using the functions `sep_get_grid_window` and `sep_put_grid_window` described in Section 1.3.4. The Grid Format File and the Grid Values File are optional. When no Grid Format File is associated with a data set, the Grid Coordinate System is assumed to be the same as the Header Coordinate System, and the grid coordinates of the header records are assumed to be regular.

1.1.5 Standard Header Keys

To make different programs work together seamlessly we defined a set of standard header keys for common trace parameters. The following table describes these standard header keys.

Name	*Type*	*Description*
sx	integer	Source x coordinate
gx	integer	Receiver x coordinate
sy	integer	Source y coordinate
gy	integer	Receiver y coordinate
offset	integer	offset
cdp	integer	CDP ensemble number
cdpi	integer	Trace number within ensemble
trid	integer	Trace status (1=recorded, 0=zero padded)
data_record_number	integer	trace position in data file (Section 1.1.3)
s_x	float	Source x coordinate
g_x	float	Receiver x coordinate
s_y	float	Source y coordinate
g_y	float	Receiver y coordinate
cmp_x	float	CDP x coordinate
cmp_y	float	CDP y coordinate
offset_x	float	x component of source-receiver offset vector
offset_y	float	y component of source-receiver offset vector
aoffset	float	Length of source-receiver offset vector
azimuth	float	Azimuth of source-receiver offset vector

1.2 Utilities

A set of basic utility programs have been developed together with the library. These programs can be used to perform useful operations on Seplib3d data sets. The following is a partial list of these utilities, that are in different stages of development.

- Synch3d - Reorder the data traces according to the value of the

`data_record_number` in the trace headers.

- `Sort3d` - Sort the headers and the traces in a dataset, and create the gridding information.
- `Window3d` - Window a data set according to the header values and/or the grid information.
- `Transp3d` - Transpose the axes of the Data Coordinate System and/or of the Grid Coordinate System.
- `Infill3d` - Regularize a data set by placing zero traces in the empty grid cells.
- `Headermath` - Perform arithmetic operations on header values.
- `In3d` - Print information on a data set.
- `Cp3d` - Copy a data set.
- `Rm3d` - Remove a data set.
- `Create3d` - Create a Seplib3d data set from Seplib files that contains the data traces and the geometry information
- `Su2sep3d` - Convert a SU data set into a Seplib3d data set.
- `Sep3d2su` - Convert a Seplib3d data set into a SU data set.
- `Promax3d` - Converts a Promax data set into a Seplib3d data set.

1.3 ACCESSOR ROUTINES

This sections describe the routines that can be used by the application programmer to access the information contained in a Seplib3d data set: data, headers and grids.

1.3.1 Accessors to coordinate system parameters

These routines access the parameters that describe the data axes, the header axes and the grid axes contained in the History File, the Header Format File, and the Grid Format File (e.g. `n2_grid, o2_grid, d2_grid, label2_grid`).

1.3.2 Accessors to header formats

Routines that access the table of header keys stored in ASCII format in the Header Format File

1.3.3 Accessors to header values

These routines access the header values either by key name or by key index. The values can be accessed either one header key at the time, or a specified number of whole headers at the time. The pointers to the desired header record can be computed using the headers navigation routines.

1.3.4 Headers navigation routines

One of the main additional complexities caused by the introduction of irregularly sampled data is that navigating the data set, that is computing the pointers into the Data Values File and Header Values File of the records of interest, becomes a non-trivial problem. Since pointers to the data record are stored in the data record number header key of the corresponding header record, the problem is reduced to compute the pointers to the header record, that we call header record number. This task is facilitated if the data set has been gridded. In this case the programmer is provided with two simple headers navigation routines (`sep_get_grid_window and sep_put_grid_window`), for accessing the header record number given their grid coordinates. The modality for storing and retrieving the information on the grid coordinates from the database is straightforward. The header record number are accessed for regular sub-windows of the grid coordinates. The sub-windows are specified by using the following windowing parameters for each axis of the Grid Coordinate System: `n_wind` (length of the window), `f_wind` (origin of the window) and `j_wind` (under-sampling rate of the window).

1.3.5 Accessors to file links

These routines provide access to the links between the History File, the Header Format File, and the Grid Format File, that are specified as ASCII strings in the History File.

1.4 3-D prestack example

The following example shows how to use Seplib3d programs and libraries to analyze and manipulate 3-D prestack seismic data sets. The data set is a single sail line belonging to a 3-D marine data set recorded in the North Sea. The data were recorded by shooting from two sources located at the stern of the recording ship into three streamers being pulled behind the ship. There are about 120,000 traces with 500 time samples in the data. The total size of the Data Values File is about 250 Mbytes. Figure 1.1 shows the source locations, for one trace out of ten in the data set. Figure 1.2 shows the receiver location for the same subset of traces in Figure 1.1. The receiver locations are scattered along the cross-line direction (y) because of a fairly strong cable feather, probably caused by ocean currents during acquisition. The corresponding CMP locations are shown in Figure 1.3. The cable feather caused a noticeable irregularity in the CMP coverage, with cross-lines overlapping in many places. Figure 1.4

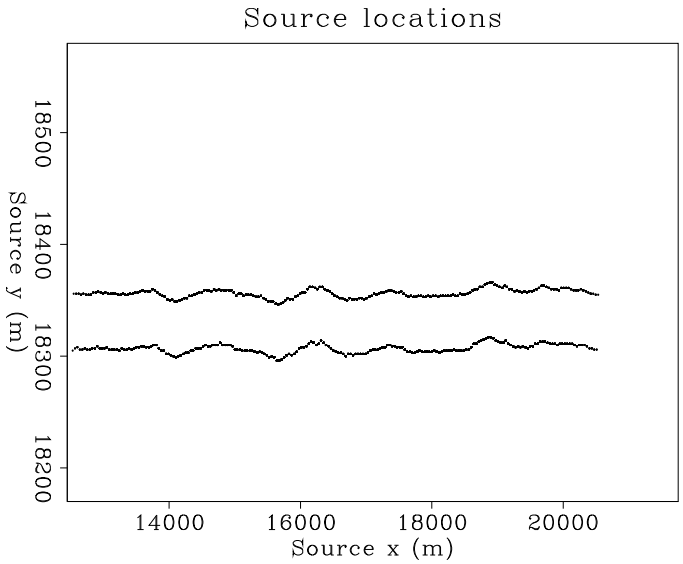


Figure 1.1: Source locations of data traces. Sep3d-Sxy [ER]

shows the trace offset-azimuth distribution. Each of the six different trends distinguishable at small offsets corresponds to a source-streamer pair.

1.4.1 Exploring the data

At the start the data set is in Seplib3d ungridded format. The seismic traces and the trace headers are in arbitrary order. Our goal is to analyze the information on the geometry of the data and to extract meaningful subset of the data to be displayed. However, since the data set is relatively bulky (250 Mbytes), we want to explore it without replicating the seismic data themselves; further we want to avoid to perform the sorting an binning operation every time we choose to display a different subset.

The first step is to sort and bin the trace headers on a regular grid, by using the `Sort3d` program. We define a three-dimensional grid, with the grid axes being the absolute value of the offset, the in-line midpoint coordinate, and the cross-line midpoint coordinate. Since for 3-D data the source-receiver offset is a vector, we could define azimuth of this vector as an additional grid axes; we leave this possibility as an exercise to the reader. The midpoint distribution is irregular (Figure 1.3), therefore it is likely that more than one trace belong to the same grid cell. If this overlap happens, `Sort3d` automatically adds an axis (called `trace_in_bin`) to the grid, making the grid a four-dimensional object. Notice that the sorting and binning is performed on the trace headers and does not require access to the seismic traces. However, as an option, `Sort3d` can reorder the data themselves after having sorted and binned the headers.

Once the grid has been computed we can extract many interesting subsets of the data by using the windowing program `Window3d`. This program can extract specified slices from the four-dimensional grid created by `Sort3d`. For example we can display at the subset of traces that correspond to grid cells with the same offset and cross-line midpoint. Figure 1.5 show

one of this "common-offset" sections. We can see the complex structures below the surface, with plenty of faulting and diffractions. Similarly we can extract a CMP gather at any desired locations, as the one shown in Figure 1.6. Because the sorting and the binning has been done once by `Sort3d`, and the pointers to the data traces corresponding to the grid cells is stored in the Grid Value File, `Window3d` can extract these subsets very quickly without the need to go through the whole data set, but just by extracting the desired traces from the Data Value File.

1.4.2 Analyzing binning parameters

The choice of the binning parameters for a data set depend on many factors, such as the acquisition geometry and the planned subsequent processing flow. Using `Sort3d` we can study the effects of choosing different binning parameters in order to select optimal ones. For example, we can vary the bin width along the midpoint axes to determine the analyze the the trade-off between bin resolution and uniformity in the offset and azimuth coverage within the bins. `Sort3d` is an efficient tool for this study, because it bins the trace headers and create the gridding information, without accessing the bulky Data Values File. To assess the quality of a set of binning parameters we can display subsets of the data, as described in the previous section. We can also examine the distribution of the trace headers on the grid by measuring the fold within each of the grid cells. A simple way of measuring the fold is to count the number of elements along the `trace_in_bin` axes that `Sort3d` creates when encounter multiple traces belonging to a single grid cell. The following subroutine performs this task, and outputs the results into a Seplib data set, which can then be displayed and examined. Notice that the subroutine is coded in Fortran 90, and uses a few of the powerful features of the language (e.g. automatic allocation of arrays, reduction intrinsics) that enables the writing of simple and compact code.

```
! subroutine that computes the fold from the gridding information
subroutine comp_fold(num_grid_axes,tot_grid_cells,n_in_bin,n_grid)

implicit none

integer sep_get_grid_window,srite
integer num_grid_axes,tot_grid_cells,n_in_bin
integer n_grid(num_grid_axes)

integer f(num_grid_axes),j(num_grid_axes) ! notice automatic allocation
integer grid_val(n_in_bin,tot_grid_cells) ! notice automatic allocation
real fold(tot_grid_cells) ! notice automatic allocation

f=0 ! notice array syntax for initilization
j=1

! get the pointers to header record number for the whole grid
if (sep_get_grid_window('in',num_grid_axes,n_grid,n_grid,f,j,grid_val) /= 0) &
        call seperr('error in sep_get_grid_window')

where (grid_val > 0)
   grid_val =1 !There is a trace if the pointer is non-negative
elsewhere
```

```
   grid_val =0 !There is NO trace if the pointer is negative
end where

fold=float(sum(grid_val,dim=1)) !notice use of reduction intrinsic SUM

if(srite('out',fold,tot_grid_cells*4) /= tot_grid_cells*4) &
         call seperr('error in srite')

return
end
```

This subroutine is the main subroutine of a Seplib3d application program (`Fold3d`). Figure 1.7 shows the fold values for cross-line of the data set computed using `Fold3d`.

1.4.3 Efficient processing

In addition to fast inspection of the data, the Seplib3d data format enables also efficient processing of large data sets. Because of practical limitations on the number of traces that can be held in memory at any given time, the processing of large data sets is typically carried out on blocks of traces. The gridding mechanism of Seplib3d enables the processing programs to access the data in a random-access fashion, instead of the usual sequential-order access required by conventional data formats. The capability of selecting the traces to be read according to specified header keys reduces the number of multiple passes on the data necessary to complete the processing, and consequently it drastically reduces the amount of data I/Os, with a obvious gain in efficiency.

For example, an efficient prestack depth migration using the Kirchhoff method can be quickly developed if the application program can easily select traces with source and receiver coordinates within given ranges. Such data access method enables the migration program to handle the traveltime tables necessary for Kirchhoff depth migration in a straightforward manner. A traveltime table is computed, and stored, for each node on a grid of surface locations. The reading of these tables can be minimized when the data are accessed in ranges of source and receiver locations, as enabled by the Seplib3d grid.

Figure 1.8 shows the source (top) and receiver (bottom) location of a subset of the North Sea data set. Figure 1.9 shows the same locations as in Figure 1.8, but with a $500m \times 500m$ grid superimposed onto the plot. The grid is four-dimensional, thus it is difficult to visualize, and Figure 1.9 shows the two-dimensional projections of this four-dimensional grid. In other words, a trace belongs to one grid cell only if both the source locations and the receiver locations are within the given ranges. Figure 1.10 show an example of such subset of traces. The source-receiver locations (top) are nicely localized, while the CMP locations (bottom), that would be the conventional way of accessing the data, are dispersed.

REFERENCES

Claerbout, J. F., 1991, SEPlib and SEP software-computer: SEP–**71**, 283–286.

Dellinger, J., and Tálas, S., 1992, A tour of SEPlib for new users: SEP–**73**, 461–502.

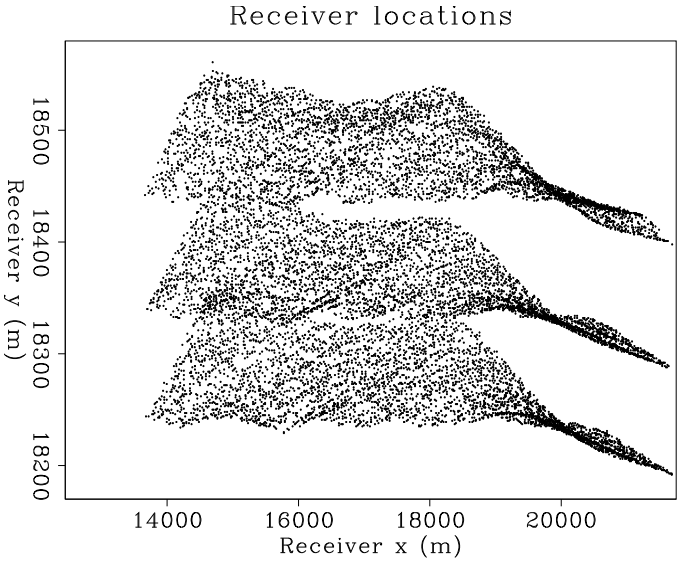


Figure 1.2: Receiver locations of data traces. Sep3d-Rxy [ER]

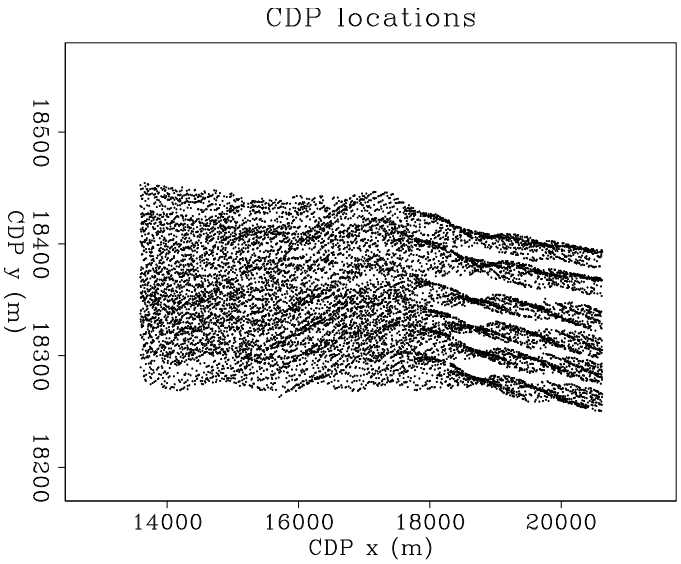


Figure 1.3: CDP locations of data traces. Sep3d-CDPxy [ER]

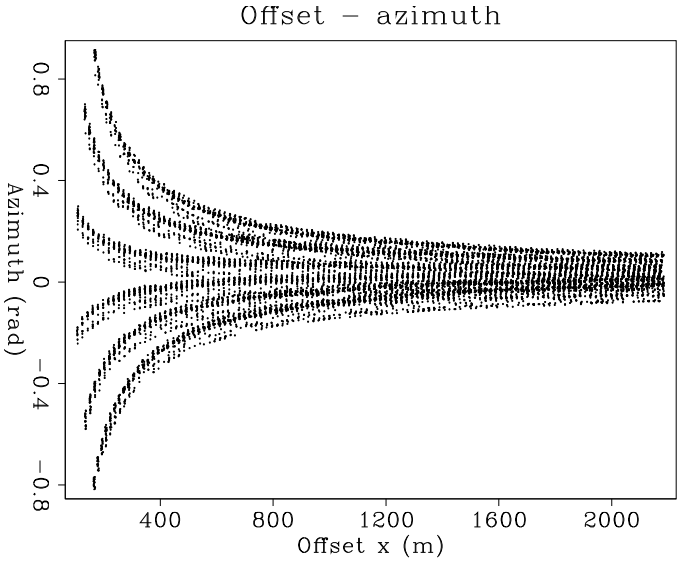


Figure 1.4: Offset-azimuth of data traces. Sep3d-OFFAZ [ER]

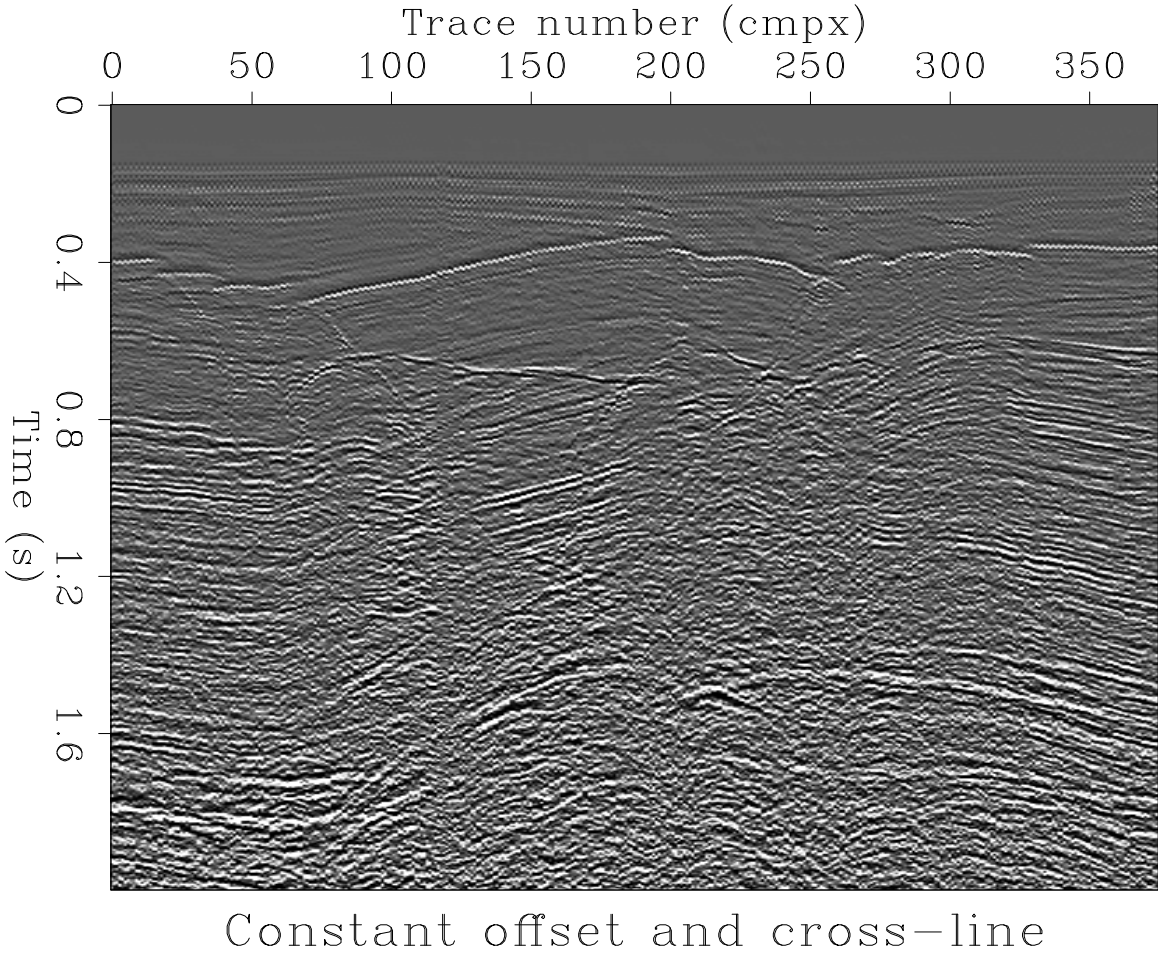


Figure 1.5: Constant offset and in-line midpoint section Sep3d-Coff [ER]

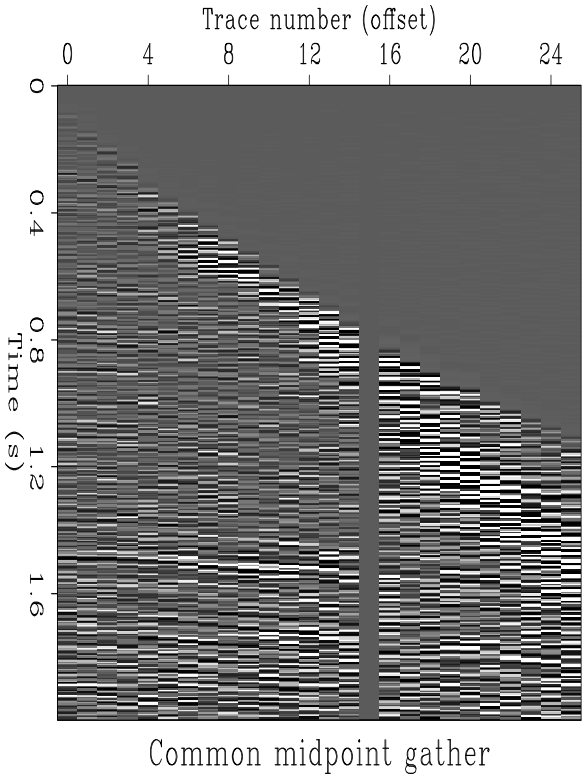


Figure 1.6: CDP gather Sep3d-Cdp [ER]

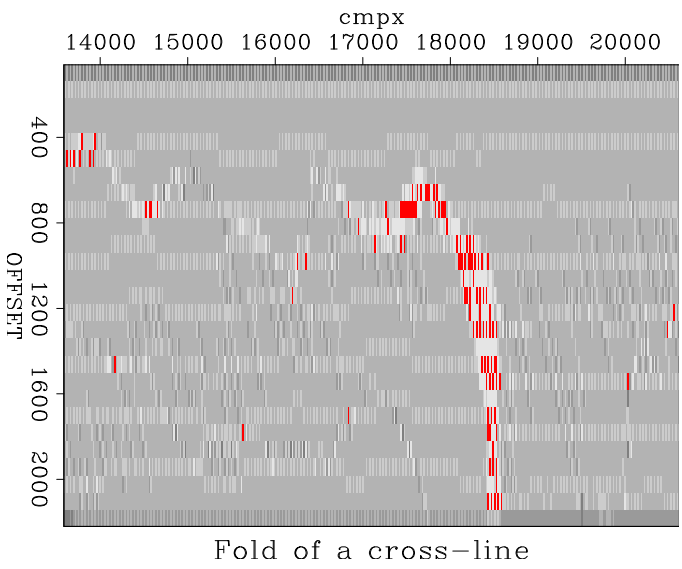


Figure 1.7: Fold values for constant in-line midpoint. Sep3d-Fold [ER]

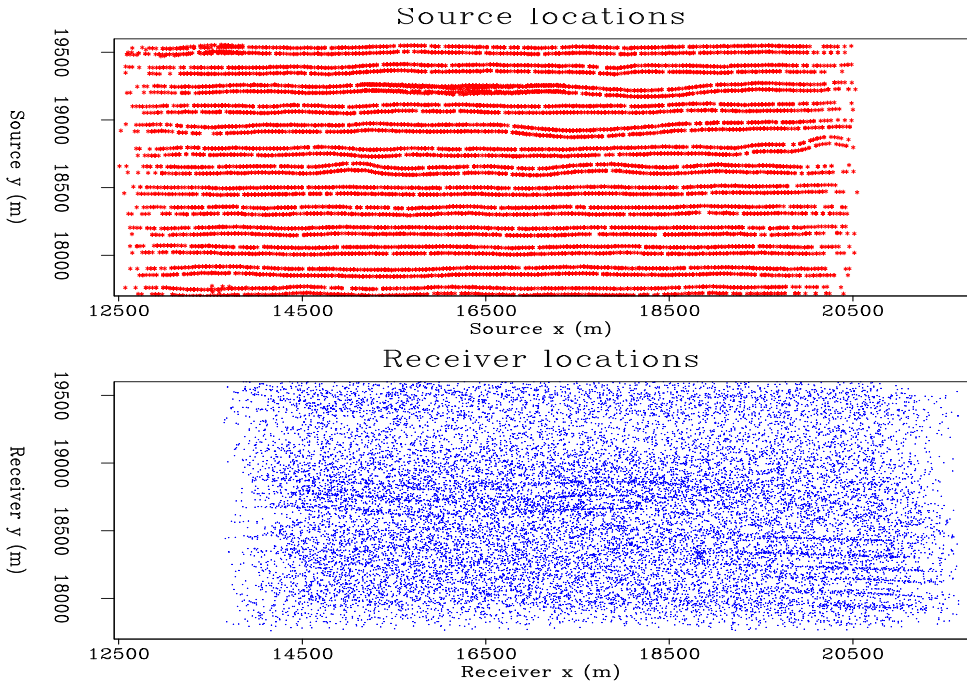


Figure 1.8: Source (top) and receiver (bottom) locations of a marine data set. Sep3d-RSxy-big [NR]

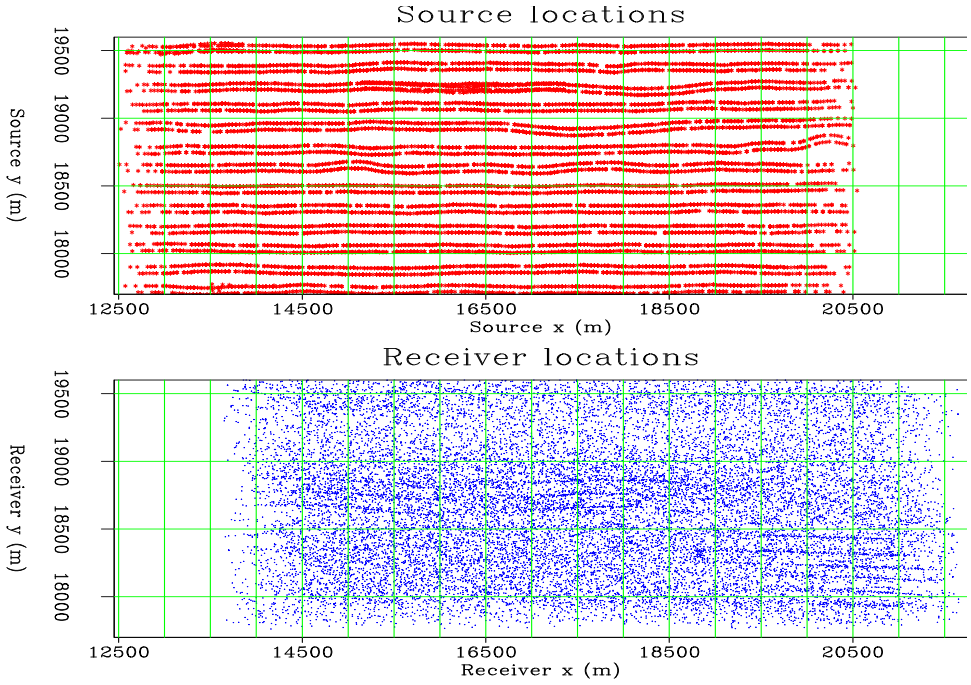


Figure 1.9: Source (top) and receiver (bottom) locations of a marine data set after gridding. Sep3d-RSxy-grid [NR]

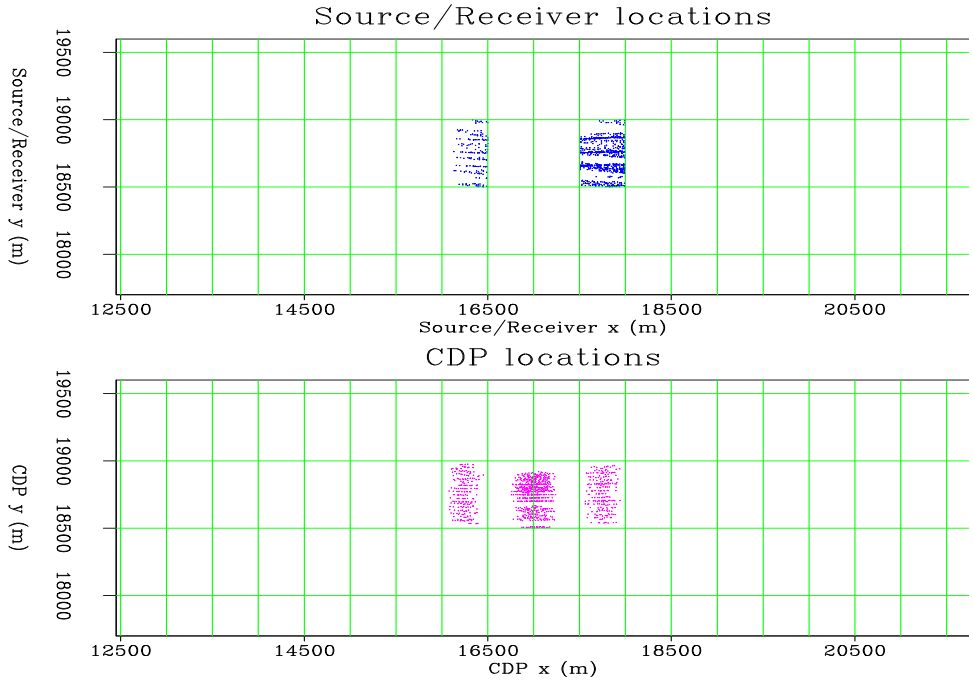


Figure 1.10: Source/receiver (top) and CDP (bottom) locations of a selected subset of the data. Sep3d-RSCDPxy-win [NR]

Chapter 2

The SEG-EAGE salt data set - (*to be written*)

This is a short description of the SEG-EAGE salt data set that I used all throughout this book.

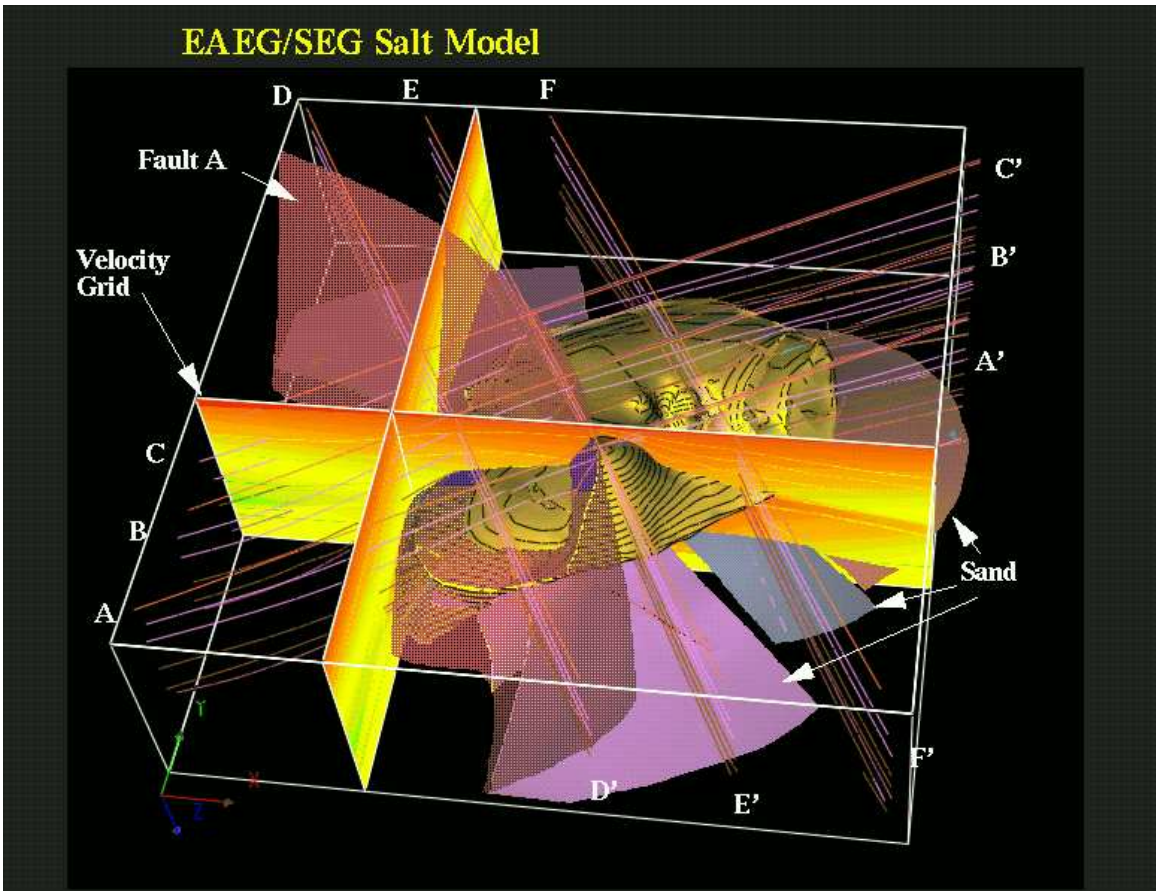


Figure 2.1: 3-D representation of the SEG-EAGE salt model. seg-salt-salt [NR]

REFERENCES