CWP-704
December 2011



Blended source imaging by amplitude encoding

Jeffrey A. Godwin

— Master of Science Thesis —
Geophysics

Defended on December 9, 2011

Committee Chair:	Prof. Luis Tenorio
Advisor:	Prof. Paul Sava
Committee members:	Prof. Yaoguo Li
	Prof. Dave Hale

Center for Wave Phenomena
Colorado School of Mines
Golden, Colorado 80401
(1) 303 273-3557

Abstract

In the last decade the seismic imaging industry has begun collecting data volumes with a substantial amount of data redundancy through new acquisition geometries including: wide-azimuth, rich-azimuth and full-azimuth geometries. The increased redundancy significantly improves image quality in areas with complex geology, but requires considerably greater computational power to construct an image because of the additional data and the need to use advanced imaging algorithms in complex geology. One way to reduce the computational cost of processing such datasets is to blend shot-records, using shot-encoding, together prior to imaging which reduces the number of migrations necessary for imaging. The downside to doing so, is that blending introduces strong, non-physical, cross-talk noise into the final image. By carefully choosing the shot-encoding scheme, I can reduce the additional noise inserted into the image and maximally reduce the number of migrations necessary. To find an optimal encoding scheme, I develop a theory of blended imaging that describes all shot-encoding schemes, and design a new class of encodings that use amplitude weights instead of phase-shifts or time-delays. We are able to use amplitude encoding to produce blended images of the same quality as previous encoding schemes at the same computational cost and find that the result of most encodings is governed by spatial sampling theory. Also, I show that blended imaging can be used to produce common image point gathers which may significantly decrease the cost of constructing such gathers. Overall, I find that the cost advantage of blended imaging is much less than previously stated, and that it is not clear whether or not blending is useful in most circumstances when a fair cost comparison is conducted.

Table of Contents

Abstract i

Acknowledgements vi

Chapter 1 Introduction 1

Chapter 2 A theory of blended source imaging 11

2.1 Matrix representation of seismic imaging 12
2.2 The encoding matrix 13
2.3 Matrix representation of blended imaging 17
2.4 Critical assumptions in the matrix representation theory 21
2.5 Designing optimal shot-encoding schemes 22
2.6 The forward problem 22
2.7 The inverse problem 24
2.8 Cost comparison 25
2.9 Connection to other fields 27
2.10 Data compression 27
2.11 Telecommunications 28
2.12 Digital filter design 28

Chapter 3 Amplitude encoding using singular value decomposition 31

3.1 Singular value decomposition 31
3.2 Choice of crosstalk matrix 32

3.3 Results 34
3.4 Discussion 35

Chapter 4 Amplitude encoding by basis projection 37

4.1 Converting an orthgonal basis to an encoding 38
4.2 Fourier-related encodings 39
4.3 Spatial-sampling and periodization distance 39
4.4 Modulated-shot migration (DFT) 42
4.5 Plane-wave migration 44
4.6 Discrete cosine basis (DCT-II) 45
4.7 Discrete sine basis (DST-II) 48
4.8 Hartley basis 49
4.9 Discrete trigonometric basis (DCS) 50
4.10 Wavelet bases 51
4.11 Haar wavelet basis 52
4.12 Daubechies-4 53
4.13 Other orthogonal bases 53
4.14 Hadamard basis 54
4.15 Walsh basis 55
4.16 Results 56
4.17 Discussion 57

Chapter 5 Plane-wave equivalent amplitude encoding 65

5.1 Theory 65
5.2 Results 66
5.3 Discussion 70

Chapter 6 Frequency-dependent amplitude encoding by basis selection 73

6.1 Frequency-dependent spatial sampling theory 73

6.2 Results 76

6.3 Discussion 77

Chapter 7 Conclusions 83

References 87

Appendix A Comparison of various encodings 91

Acknowledgements

I would like to thank my family for their support and love during my time here in graduate school: anytime I needed someone to talk to they were always there for me. I particularly want to thank Arianne for pretending to be interested in this project over the last two years, but mostly just for listening. I would like to thank my advisor Paul Sava, for his guidance and insight on this project. I would like to thank all of my committee members: Yaoguo, Luis and Dave for serving on my committee and their insights on this project. Also, I want to thank all of my colleagues in CWP for all of their support and encouragement. Lastly, I want to thank the many nameless contributors to the following open-source software projects, without which most of this project would not have been possible: Madagascar, LaTeX, Numpy, Scipy, Matplotlib, OpenMP, OpenMPI, Ubuntu, Linux and countless others.

Chapter 1

Introduction

Today's seismic exploration challenges include imaging areas with complex geology, such as salt domes and overthrust regions. The major issues for imaging areas with complex geology are poor data quality and lack of seismic illumination due to the severe deformation of the seismic wavefield. One approach to resolve these issues is to obtain large amounts of redundant information from various acquisition directions via wide-azimuth or full-azimuth seismic surveys (Michell *et al.* , 2006; Shoshitaishvili *et al.* , 2006; Howard, 2007; Kapoor *et al.* , 2007; Ting & Zhao, 2009). However, wide-azimuth surveys require significantly more time to acquire and even greater amounts of time to process due to the large amounts of data. Subsequently, the cost of acquiring and processing a wide-azimuth survey is significantly more expensive than the cost of a conventional survey. Additionally, the cost of imaging in complex geology is much greater because advanced imaging algorithms such as reverse-time migration must be used to better honor the kinematics of complex geology. Therefore, both the financial and computational cost of today's large surveys is increasing at a rapid pace, but recent technological advances may allow us to reduce both the cost of acquiring and imaging large datasets.

One such technology is acquisition using simultaneous or delayed sources (Womack, 1990; Hampson *et al.* , 2008; Berkhout *et al.* , 2008; Beasley, 2008; Blacquiere *et al.* , 2009). As the name implies, simultaneous sources are multiple sources triggered at the same time but at different spatial locations. By acquiring multiple sources in a shot-gather,

a full-survey may be acquired in much less time than is currently required. The downside to simultaneous source acquisition, is that simultaneous source data creates additional noise in the final seismic image. Past research has circumvented the additional noise by de-blending the simultaneous source shot-gathers to create separate shot-gathers, as would be collected in a conventional survey, for each source prior to imaging (Spitz *et al.* , 2008; Moore *et al.* , 2008; Hampson *et al.* , 2008; Akerberg *et al.* , 2008; Kim *et al.* , 2009; Huo *et al.* , 2009). The separated shot-gathers are imaged using conventional shot-record migration (SRM) to produce the final image. The drawback to separating the shots is two-fold:

- First, the sources must be separable, which typically implies that they are relatively isolated from one another in space which limits the maximum amount of shots that can be used at once during acquisition.

- Second, the imaging process is not sped up by separating the shots because each separated shot must be imaged separately, as in SRM.

Because of these issues, industry has been reluctant to acquire datasets with many simultaneous sources at once.

A related idea, that aims to reduce the cost of imaging large datasets is to use multiple sources at once during imaging, which emulates simultaneous source acquisition. This process, known as blended imaging, combines multiple shot-gathers together prior to migration, which reduces the number of migrations that are needed to produce a final image (Morton, 1999; Liu, 1999; Romero *et al.* , 2000; Soubaras, 2006; Zhang *et al.* , 2007; Perrone & Sava, 2009; Berkhout *et al.* , 2009). Certain forms of blended imaging, most notably plane-wave migration, are used in industry today, but many forms of blending (e.g. random phase) are not used today because they introduce a significant amount of noise to the image. However, the presence of the noise may be justified if the cost advantage over conventional shot-record migration is sufficiently high.

In order to illustrate how blending affects the cost of imaging, I first examine conven-

tional shot-record migration. In shot-record migration, seismic data corresponding to individual shots are backpropagated independently to reconstruct the corresponding receiver-wavefield. Concurrently, the source-wavefield for each shot is constructed by forward propagating the source using a known velocity model. An imaging condition is then applied to the individual reconstructed source and receiver wavefields to produce an image of the subsurface. Because this process is repeated for each shot, the cost of shot-record migration (SRM) is

$$C_{SRM} = N_s C_s, \tag{1.1}$$

where C_{SRM} is the total cost in computation time, N_s is the number of shot records, and C_s is the computational cost for each shot. Since modern seismic surveys often consist of many tens of thousands of shots, and each shot may image a large area (a few km^2), the overall computational cost for imaging a seismic survey is tremendous.

However, equation 1.1 hints at a few possibilities to reduce the overall cost of imaging:

- reduce the cost of migrating each shot (i.e. reduce C_s), while keeping N_s constant,
- reduce the number of shot-records used for migration (i.e. reduce N_s), while keeping C_s constant, or
- reduce both the cost of migrating each shot, C_s, and reduce the number of migrations necessary, N_s, at the same time.

In most scenarios, reducing the cost of migrating each shot C_s is the most practical approach to reducing the cost of imaging because one can choose the algorithm that is used for wavefield reconstruction (e.g. wavefield extrapolation or reverse-time migration). However, much of today's imaging is done in areas with complex geology, so computationally expensive algorithms like reverse-time migration have to be used to produce accurate images. Therefore, most reductions in computational cost typically come from advances in computer hardware. In most cases however, our ability to create more demanding algo-

rithms, e.g. full anisotropic reverse-time migration, and acquire more data greatly exceed the rate of hardware advances.

Furthermore, reducing the number of shots, N_s, that are used during imaging is not typically an option because both the signal-to-noise ratio and the illumination of the seismic image are negatively impacted by removing sources. An alternative to directly reducing the number of sources is to linearly combine multiple, independent shot-records together into blended shot-records prior to migration as in Figure 1.1 (Romero *et al.* , 2000). By doing so, we can effectively reduce the number of migrations that are necessary from N_s to N_e, where N_e is the number of blended shot-gathers or experiments. Therefore, the total cost of imaging using a blended migration scheme becomes:

$$C_{BRM} = N_e C_e. \tag{1.2}$$

In equation 1.2, C_{BRM} is the total cost for a blended source migration, C_e is the cost of an individual blended record migration, which I assume to be the same as the SRM cost C_s. In practice, $C_{BRM} > C_s$ as blended shots must be migrated over a larger migration aperture (Romero *et al.* , 2000). In general, as the number of blended experiments, N_e, decreases the overall cost of blended imaging decreases. In most instances, the reduction in cost by blending greatly outweighs the additional cost of extending the migration aperture as $N_e \ll N_s$, whereas C_e is only *somewhat* greater than C_s. The ultimate goal of blended imaging is to image using all shots in a single migration ($N_e = 1$), but a more realistic goal for blended imaging may be to reduce the computational cost of imaging by at least an order of magnitude.

During the blending process, each shot-record is individually weighted (or phase-shifted) and then all the weighted shot-gathers are summed together to form a blended shot-gather prior to migration (Morton, 1999; Liu, 1999; Romero *et al.* , 2000; Soubaras, 2006; Zhang *et al.* , 2007; Perrone & Sava, 2009; Berkhout *et al.* , 2009). To form a shot-

encoding scheme, a series of blended shot-gathers are produced by changing the weights for each shot-record from blended experiment to blended experiment. The way that the weights are changed from experiment-to-experiment, determines the type of encoding, and the characteristics of that encoding scheme (Liu, 1999; Romero *et al.* , 2000; Perrone & Sava, 2009). After the shot-records are encoded, we can reconstruct the source- and receiver-wavefields for the blended experiments exactly as in conventional shot-record migration. To produce an image using the blended wavefields we can apply the conventional cross-correlation imaging condition to the reconstructed wavefields as follows,

$$R_e(\mathbf{x}) = \sum_e \sum_t \left(\sum_i A_e^i W_s^i(\mathbf{x}, t) \right) \left(\sum_j A_e^j W_r^j(\mathbf{x}, t) \right), \tag{1.3}$$

where i and j are the shot-record indices, $W_s^i(\mathbf{x}, t)$ is the ith reconstructed source wavefield, $W_r^j(\mathbf{x}, t)$ is the jth reconstructed receiver wavefield, e represents the blended shot-gathers index, A_e^i or A_e^j represents the shot-encoding weight for i^{th} or j^{th} shot-record for the, e^{th} blended experiment, t represents time, $\mathbf{x}$ is a vector of locations in space, and N represents the crosstalk artifacts introduced by blending. When $i = j$, equation 1.3 computes the correlation of wavefields related to a single shot gather. Conversely, when $i \neq j$, equation 1.3 computes the value of the source and receiver wavefields that are not physically related to one another, which introduces strong, coherent noise into the image that is virtually indistinguishable from the desired image. This strong noise is referred to as cross-talk noise (Romero *et al.* , 2000). As equation 1.3 shows, the cross-talk terms exist for each blended experiment for each source and every other receiver wavefield which means that there is a significant amount of cross-talk in the final image. Due to the negative impact of cross-talk noise on final image quality, cross-talk noise must be sufficiently attenuated during the imaging procedure.

Previous attempts to remove the cross-talk terms in equation 1.3 have focused on phase-shifting the shot-records in an attempt to shift the wavefields out of phase with one

another. This methodology was first proposed by Rietveld (1994) who used phase-shifts to create wavefields with optimal illumination in complex geology. The prospect of imaging with blended wavefields was then more-or-less neglected until Tieman (1997) introduced plane-wave migration, and Morton (1999) used random phase-shifts to speed up imaging. Some of the more prominent phase-encoding schemes include: plane-wave (Temme, 1984; Tieman, 1997; Stork, 1999; Liu, 1999; Duquet, 1999; Stoffa *et al.* , 2006; Zhang *et al.* , 2005; Liu *et al.* , 2006; Shan & Biondi, 2008), random (Romero *et al.* , 2000), harmonic (Zhang *et al.* , 2007), plane-wave with dithering (Perrone & Sava, 2009), frequency dependent, sign-opposite (Sun, 1999), and modulated-shot migration (Soubaras, 2006). It is important to note that many of these schemes choose to use linear phase-shifts as a function of frequency, and are therefore essentially time-delay encodings. Of these encoding schemes, plane-wave is the most widely used due to its simple implementation and intuitive interpretation.

Regardless of the shot-encoding scheme chosen, the most powerful method for attenuating cross-talk noise is to simply stack the images over many blended experiments (or realizations) that have different cross-talk characteristics because the desired image stacks together, while the incoherent cross-talk noise stacks out (Romero *et al.* , 2000). However, simple stacking is not sufficient to completely remove most of the cross-talk for many shot-encoding schemes at a suitable cost advantage over shot-record migration because cross-talk noise is not completely incoherent from one blended experiment to another (i.e. cross-talk noise stacks together) (Romero *et al.* , 2000). Thus, it is crucial to select encoding schemes that minimize the cross-talk noise during the imaging process. Subsequently, the most important unresolved question regarding blended imaging is: how can one select and implement an *optimal* shot-encoding scheme to produce the best quality image at the lowest possible computational cost?

The main goal of this work is to address the question of how to design an optimal shot-encoding scheme for blended imaging. To do so, I demonstrate in Chapter 2 that a common theory of blended imaging exists that allows us to quantify the effect of blending on the

final image without any prior knowledge of the subsurface and/or the data. Based on the mathematical theory of blending I design amplitude encodings in two ways: first by matrix factorization in Chapter 3 and second by basis projection in Chapter 4. To demonstrate the relative advantages of each type of encoding I perform an one-to-one comparison of amplitude encoding, phase-shift encoding and time-delay encoding on the Sigsbee2A dataset. Next, I demonstrate the equivalence of amplitude and phase-shift encoding (plane-wave migration) in Chapter 5, which shows that amplitude encoding obtains identical results to time-delay encoding but at less computational cost. In Chapter 6, I explore the connection between blended imaging and sampling theory which allows us to remove overly redundant information as a function of frequency, thereby imposing a constraint on the cost advantage of blending over shot-record migration. Overall, this work provides an accurate picture of blended imaging from a holistic point of view which is something thus far missing from the seismic imaging literature.

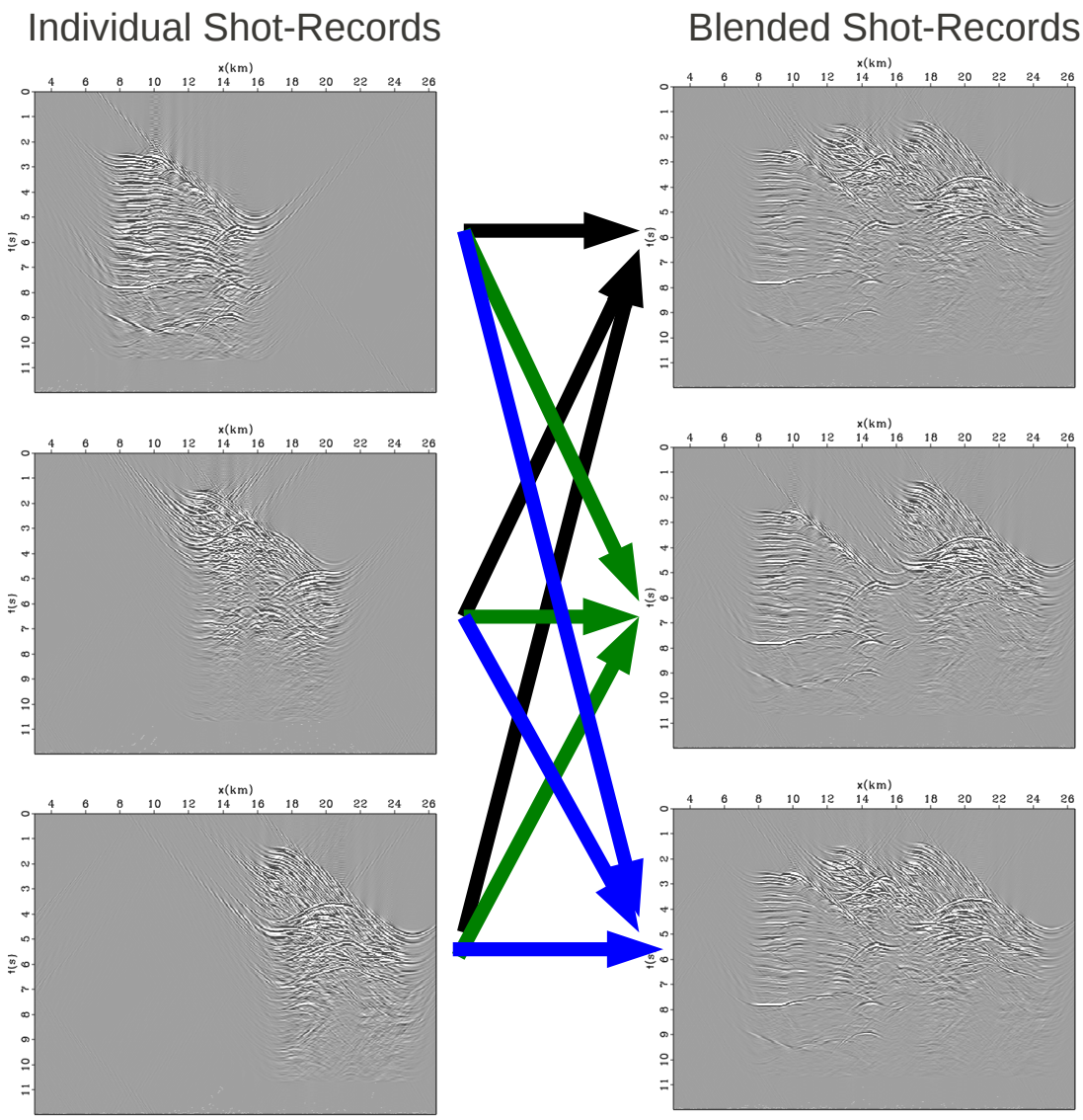
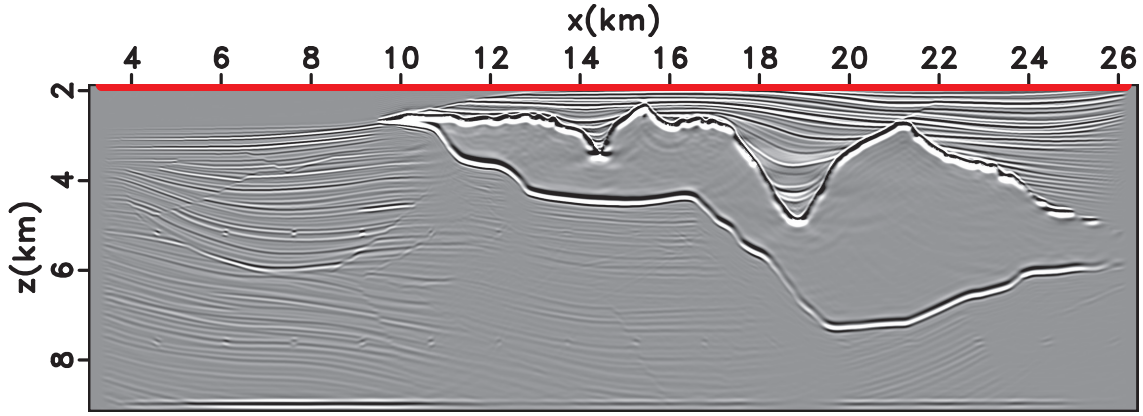
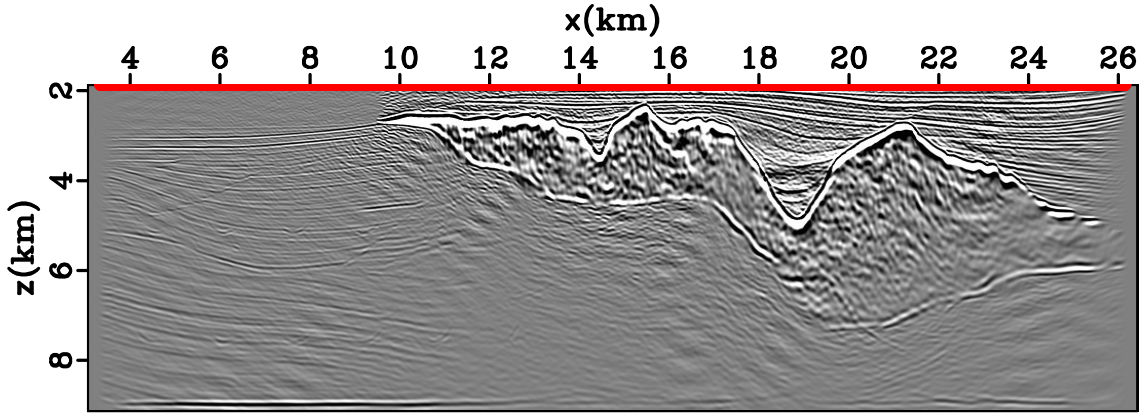


Figure 1.1. A schematic of the blending process. On the left, the individual shot-records, as acquired in the field are combined together by weighting each shot-record and then summing all the shot-records together to produce a single blended record (right). By changing the weighting scheme for each shot-record (green and blue arrows) I can produce different realizations of blended shot-records (right).



(a)



(b)

Figure 1.2. An example of how crosstalk contaminates the final stacked image for blended images. The top image (a) is constructed using conventional shot-record migration. The bottom image (b) is constructed using random time-delay encoding, which introduces crosstalk artifacts that severely degrade the quality of the image.

Chapter 2

A theory of blended source imaging

As previously mentioned, shot-encoding introduces strong cross-talk noise into the final image. Because both the characteristics of the cross-talk noise and the strength of the noise (relative to the desired image) are controlled by the shot-encoding scheme the encoding scheme must be carefully designed (Tieman, 1997; Romero *et al.* , 2000; Perrone & Sava, 2009). Some issues that affect the design of a shot-encoding scheme are:

- 1 - shot-records that are spatially close to one another contain nearly identical information, and combining them together results in lower spatial resolution in the image;
- 2 - shot-records from sources that are spatially distant from one another may interfere at greater depths in the image, thereby producing strong cross-talk noise with geologic-like dips;
- 3 - shot-records that are out of phase with one-another produce strong cross-talk that is not easily attenuated by stacking.

Additionally, an *optimal* shot-encoding scheme requires the fewest number of blended experiments to produce the highest quality image compared to other shot-encoding schemes. Thus, we are faced with a major dilemma when designing shot-encoding schemes: to reduce the effect of cross-talk on our image quality we must use many blended experiments to attenuate the cross-talk noise by stacking, but each additional blended experiment reduces the cost advantage of blending over conventional shot-record migration.

2.1 Matrix representation of seismic imaging

To design blended source encoding schemes, I could start by randomly selecting possible encoding schemes. However, this search spans a space that is infinitely large, and there is no guarantee that one would ever find an optimal encoding. Rather, I find that conventional seismic imaging can be described by a series of matrix operations, which can be expanded to include blended source imaging. The matrix representation allows us to determine the overall suitability of a shot-encoding scheme by determining the amount of cross-talk in the migrated image in advance.

To start, I revisit conventional shot-record migration, which is composed of two steps: wavefield reconstruction and the application of an imaging condition, which produces an image R as follows,

$$R(\mathbf{x}) = \sum_t \sum_i W_s^i(\mathbf{x}, t) W_r^i(\mathbf{x}, t). \tag{2.1}$$

For each shot gather, the source- and receiver-wavefields, W_s^i and W_r^i respectively, are reconstructed separately. Then, the two wavefields are cross-correlated and summed over time or frequency, depending on the domain, to form a partial image. All of the partial images are then stacked together to form the final image. Mathematically, each source and receiver-wavefield can be thought to be an element in a vector that holds all source- or receiver-wavefields respectively,

$$\mathbf{W}_S = \left[W_S^1, W_S^2, ..., W_S^i, ..., W_S^{N_s} \right], \tag{2.2}$$

$$\mathbf{W}_R = \left[W_R^1, W_R^2, ..., W_R^i, ..., W_R^{N_s} \right], \tag{2.3}$$

where $\mathbf{W}_S$ and $\mathbf{W}_R$ are row vectors, N_s is the number of shot-gathers, composed of the back projected wavefields, W_S^i and W_R^i, respectively as in 2.2(a). The ith elements of both $\mathbf{W}_S$ and $\mathbf{W}_R$ correspond to the same physical shot-gather. Thus, conventional seismic imaging

is equivalent to the inner product of the two vectors,

$$R = \mathbf{W}_S \mathbf{W}_R^H, \tag{2.4}$$

where R is the constructed image, H is the complex-conjugate transpose, and the multiplication of two elements of the matrix is actually the application of the imaging condition between those wavefields, 2.2(b). As indicated earlier, the application of the imaging condition implies a summation over either time or frequency depending on which domain I use for wavefield reconstruction. The summation over elements of the vectors implies stacking the partial images together.

2.2 The encoding matrix

In order to expand this notation to blended source imaging, I introduce an additional matrix, which I call the encoding matrix $\mathbf{E}$. The encoding matrix is an $N_s \times N_e$ matrix, where N_e is the number of blended experiments and N_s is the number of shots (wavefields) in the survey. Each column in the encoding matrix corresponds to a single blended experiment (realization), while each row acts as a weight for a particular source- or receiver-wavefield. Thus, each column in the encoding matrix weighs all wavefields to determine how to combine them together prior to imaging. To formally define the encoding matrix, I introduce the matrix notation $E_{m,n}$, where the m index corresponds to the row (or shot index) and n corresponds to the colum, or blended experiment index. The encoding matrix contains weights which may be fractional, positive or negative, and may even be complex-valued. I note that the physical meaning of the weights changes based on their format. For example, real-valued weights correspond to weighting the entire contribution of full wavefields (amplitude encoding), whereas complex-valued weights represent phase-shifting wavefields as in 2.3(a). Additionally, it is possible to combine both amplitude and phase-shifts together,

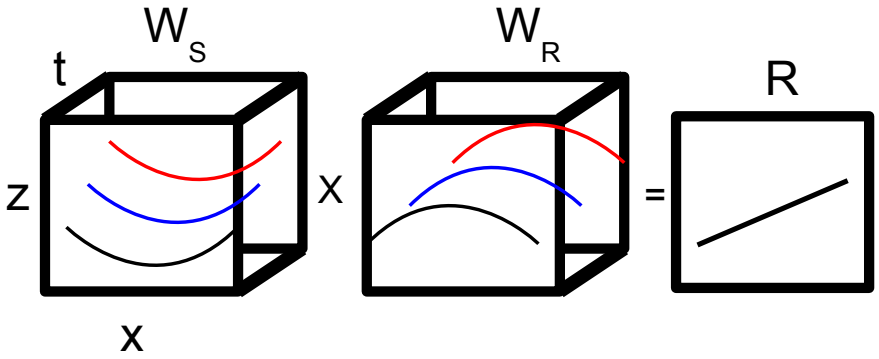
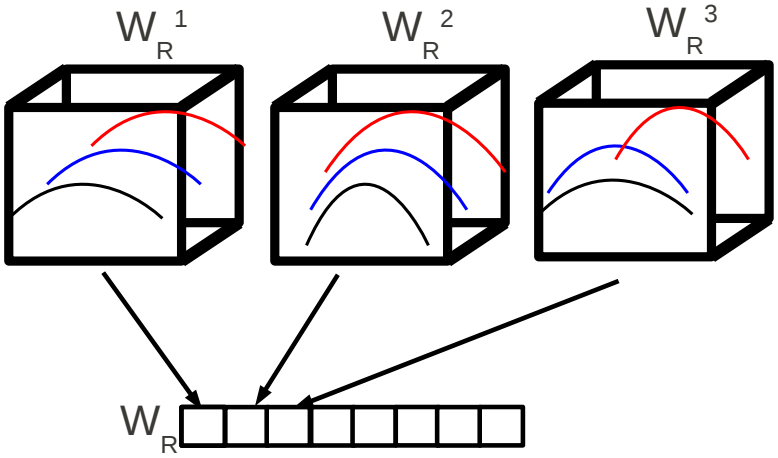
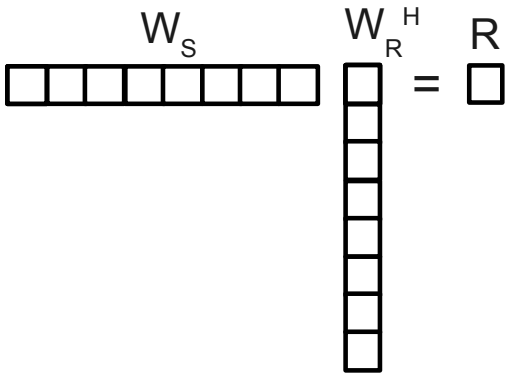


Figure 2.1. Conventional seismic imaging involves two main steps: wavefield reconstruction and the application of the imaging condition. Here the W_S and W_R wavefields are the constructed source- and receiver-wavefields respectively. The multiplication operator represents the application of the conventional cross-correlation imaging condition and the stack over time or frequency depending on which domain the wavefields are in. The image R is produced from a single shot-record. This process is repeated for each shot-record in a seismic survey.



(a)



(b)

Figure 2.2. In the matrix representation scheme the source- and receiver-wavefield vectors $\mathbf{W}_S$ and $\mathbf{W}_R$ (a) and contain full wavefields corresponding to each individual shot-record. In this scheme conventional imaging (b) is represented by the dot product of the source- and receiver-wavefield vectors $\mathbf{W}_S$ and $\mathbf{W}_R$.

which can be represented mathematically as,

$$\mathbf{E}_{m,n} = A_{m,n} \exp(-j\phi_{m,n}), \tag{2.5}$$

where A is the amplitude weight for the mth shot-record for the nth blended experiment, $\phi_{m,n}$ is the corresponding phase-shift for that shot-record, and j is the $\sqrt{-1}$. Amplitude only encodings are represented in equation 2.5 by letting $\phi_{m,n} = 0$ for all m and n. Conversely, phase-encodings are obtained when $A_{m,n} = 1$ for all m and n.

Because I generally assume that I can treat frequencies independently from one another, a different encoding matrix may exist for each frequency simultaneously. Thus, the encoding matrix is actually a three-dimensional matrix whose third dimension is frequency, ω, and $\mathbf{E}$ is a function of ω in the most general case. Mathematically, I can represent the frequency-dependence of an element in the encoding matrix as:

$$\mathbf{E}_{m,n}(\omega) = A_{m,n}(w) \exp(-j\phi_{m,n}(\omega)) \tag{2.6}$$

where $A_{m,n}(\omega)$ is the amplitude weight for the frequency, ω, and $\phi_{m,n}(\omega)$ is the phase-shift for the same frequency. A special case of frequency dependent phase-encodings are time-delay encodings where $\phi_{m,n}(\omega) = 2\pi\omega\tau_{m,n}$, which causes a linear-shift of phase as a function of frequency. Due to the symmetry of the time and frequency domain, it is also possible to consider time-dependent encodings, where a different encoding is used for each time-step, but it is not clear if there is an advantage to doing so.

For simplicity I only consider the frequency independent case where $\mathbf{E}(\omega) = \mathbf{E}$ for all ω, but the framework that I develop is equally valid for both frequency dependent and frequency independent encodings. For reference, Figure 2.3(b) depicts a sample amplitude-only encoding matrix where the weights are real-valued, binary (1 or 0) whose effect is to select only some wavefields per blended experiment.

2.3 Matrix representation of blended imaging

The encoding matrix reduces the effective number of reconstructed wavefields that are used for imaging by projecting the wavefield vector onto the encoding matrix, thereby reducing the dimensionality of the wavefield vector as follows:

$$\mathbf{B} = \mathbf{WE}, then \begin{cases} \mathbf{B}_S &= \mathbf{W}_S\mathbf{E} \\ \mathbf{B}_R &= \mathbf{W}_R\mathbf{E} \end{cases} \tag{2.7}$$

where $\mathbf{WE}$ is the projection of the wavefield vector (i.e. $\mathbf{W}_S$ or $\mathbf{W}_R$) onto the encoding matrix $\mathbf{E}$, and $\mathbf{B}$ is the blended wavefield vector for the source- or receiver-wavefields (i.e. $\mathbf{B}_S$ or $\mathbf{B}_R$). Therefore, $\mathbf{B}_S$ is the $1 \times N_e$ row vector of combined source-wavefields and $\mathbf{B}_R$ is an $N_e \times 1$ column vector of combined receiver-wavefields. Because the migration operator is linear, I can perform the combination of the source- and receiver-data, the product of $\mathbf{WE}$, prior to wavefield reconstruction, thus reducing the necessary number of migrations from N_s to N_e. The final blended source image R_e is represented by

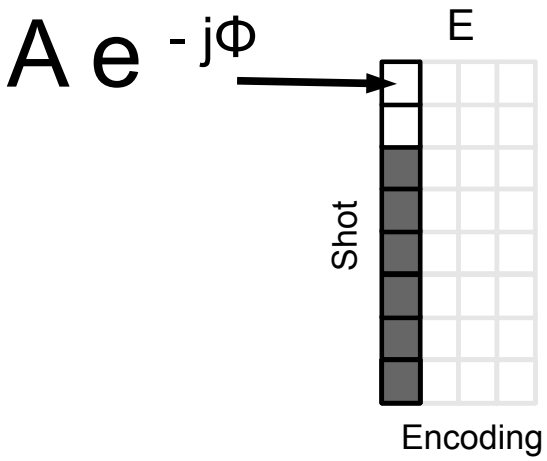
$$R_e = \mathbf{B}_S\mathbf{B}_R^H, \tag{2.8}$$

and is shown in 2.4(a) . By substituting the expressions for $\mathbf{B}_S$ and $\mathbf{B}_R$ from equation 2.7 into equation 2.8, I obtain

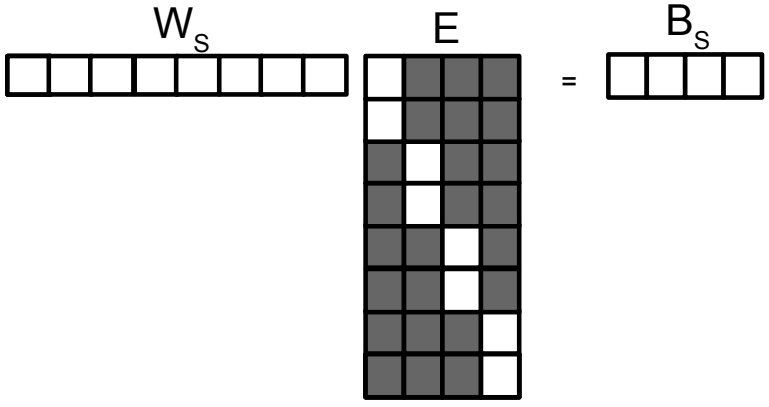
$$R_e = \mathbf{W}_S\mathbf{E}\mathbf{E}^H\mathbf{W}_R^H, \tag{2.9}$$

which is similar to equation 2.4 and is illustrated in Figure 2.4(b). I refer to the product $\mathbf{EE}^H$ as the cross-talk matrix $\mathbf{C}$, which is square and has dimensions of $N_s \times N_s$. Thus, equation 2.9 can be written as

$$R_e = R + X = \mathbf{W}_S\mathbf{C}\mathbf{W}_R^H, \tag{2.10}$$

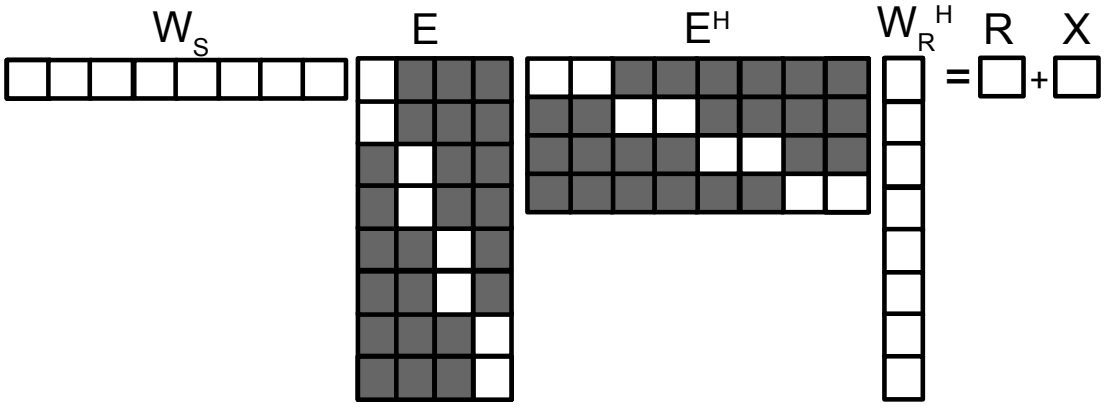


(a)

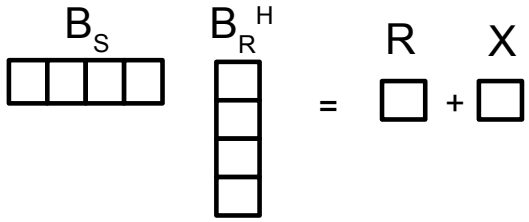


(b)

Figure 2.3. The encoding matrix (a) is composed of weights of the form $A\exp(-j\phi)$ where A is the amplitude weighting term, and ϕ is the phase-shift to be applied. To blend the wavefields, I project the wavefield vector onto the encoding matrix (b) to produced the blended wavefield vector $\mathbf{B}_S$.

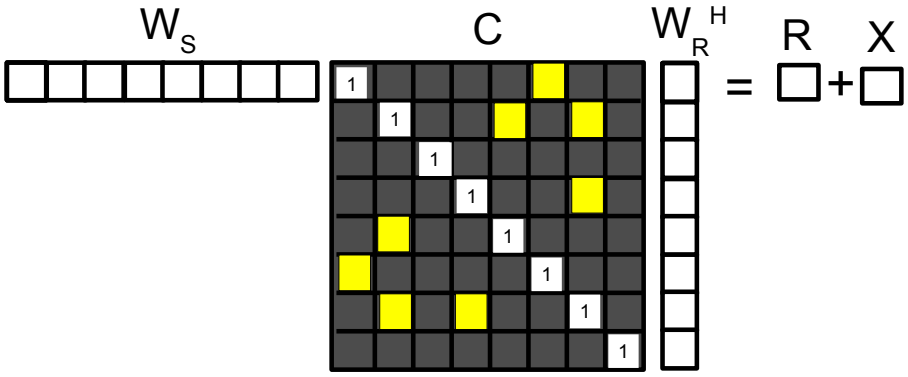


(a)

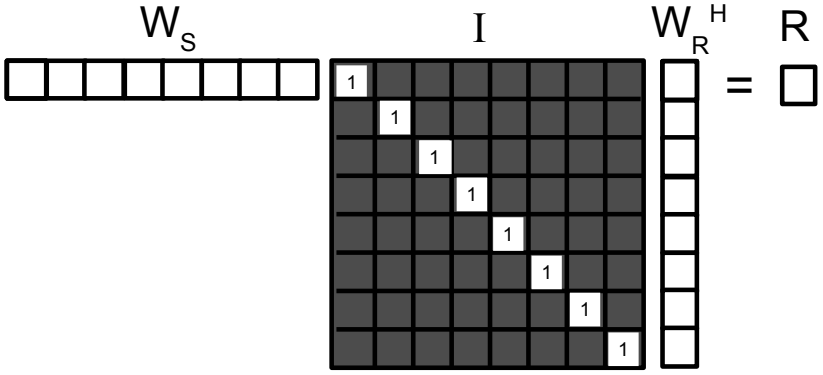


(b)

Figure 2.4. The blended imaging process is expressed as the projection of the wavefields onto the encoding matrix, and then rearranged to obtain the correct dimensionality for the matrix products (a). If I substitute the expressions for the blended source and receiver-wavefields $\mathbf{B}_S$ and $\mathbf{B}_R$ respectively (b), then the blended imaging process is summed up by the expression $\mathbf{B}_S\mathbf{B}_R^H = R + X$.



(a)



(b)

Figure 2.5. The product of the encoding matrix with its conjugate transpose produces the cross-talk matrix **C**. The cross-talk matrix (a) for a blended image is composed of the diagonal component which represents the conventional image, plus the off-diagonal components (yellow squares) that represent the pairing of wavefields not physically related to one another. As (b) shows, conventional shot-record migration is simply a special case of the cross-talk matrix where the cross-talk matrix is unitary, representing only the pairing of wavefields that are physically related to one another.

where X represents the cross-talk noise added to the conventional image. The cross-talk matrix $\mathbf{C}$ is similar to the identity $\mathbf{I}$, but with additional off-diagonal terms as shown in 2.5(a). This is a convenient description because equation 2.4 can be rewritten to include the identity matrix $\mathbf{I}$, 2.5(b), to represent the pairing of each source wavefield with its corresponding receiver wavefield as:

$$R_e = R = \mathbf{W}_S \mathbf{I} \mathbf{W}_R^H. \tag{2.11}$$

Thus, the $\mathbf{C}$ matrix represents the formation of the conventional seismic image (i.e the diagonal terms) plus additional terms in the off-diagonals representing the pairing of wavefields that are not physically related to one another. Subsequently, the off-diagonal components of the $\mathbf{C}$ matrix are the cross-talk terms that I generate by using a certain encoding matrix $\mathbf{E}$. I note that the matrix representation in equation 2.10 is a more generic finding of what Soubaras (2006) and Tieman (1997) previously discussed.

2.4 Critical assumptions in the matrix representation theory

It is critical to note that there are no additional assumptions that are commonly believed to affect shot-encoding schemes. Some of the common assumptions that do not apply to this theory of blending include:

- A fixed acquisition geometry. Since the theory is developed using wavefields, instead of shot-records I do not impose any constraint on source and receiver acquisition geometry. In fact, the source and receiver wavefields can have arbitrary receiver geometries, even completely independent of one-another.

- A requirement on shot-ordering (i.e. nearby shots must be blended in a certain way). As previously mentioned, the framework that I develop is acquisition geometry independent. Furthermore, as long as the product of $\mathbf{E}\mathbf{E}^H = \mathbf{I}$, the blended image will be cross-talk free, regardless of how the shots are ordered in the encoding matrix (even

when randomly ordered).

- I do not make an assumption regarding the velocity model in equation 2.10. Because there is no assumption about the velocity model, I find that the theory works for arbitrarily complex velocity models. Additionally, the theory of blended imaging works even when I use the *incorrect* velocity model for migration, because the cross-talk phenomena is solely dependent on the encoding scheme used provided that the migration operator is linear.

- Lastly, there is no requirement that the shot-records be evenly and uniformly sampled in space as equation 2.10 shows.

2.5 Designing optimal shot-encoding schemes

By examining equations 2.10 and 2.11, I find that the problem of designing optimal shot-encoding schemes becomes the problem of finding a cross-talk matrix $\mathbf{C}$ that is as close to the identity matrix $\mathbf{I}$ as possible in order to minimize the cross-talk in the image R_e. Consequently, the process of choosing shot-encoding scheme becomes one of determining an encoding matrix $\mathbf{E}$ such that $\mathbf{E}\mathbf{E}^H$ has the fewest off-diagonal components, or $\mathbf{E}\mathbf{E}^H \approx \mathbf{I}$. If an encoding exists such that $\mathbf{E}\mathbf{E}^H = \mathbf{I}$, then I can produce the same image as shot-record migration, at a cost proportional to the number of blended source experiments N_e, instead of N_s (Soubaras, 2006).

2.6 The forward problem

The question of how to create an optimal shot-encoding then becomes a question of how to choose a *rectangular* matrix $\mathbf{E}$ such that $\mathbf{E}\mathbf{E}^H \approx \mathbf{I}$. I define the problem of: finding $\mathbf{E}$ such that $\mathbf{E}\mathbf{E}^H \approx \mathbf{I}$, subject to cost constraints (number of columns of) $\mathbf{E}$ as the forward problem. In mathematics, a well-known solution to the ideal forward problem is through the construction of any orthonormal basis (or unitary matrix).

For reference, a matrix $\mathbf{Q}$ is defined to be an orthonormal basis if $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$ and $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$, which implies that the row and column vectors are orthogonal to one another, and that the norm of each vector is one. Thus, I can reconstruct the identity matrix $\mathbf{I}$ from $\mathbf{E}\mathbf{E}^H$, if I use an orthonormal basis for our encoding matrix $\mathbf{E}$. Unfortunately, orthonormal bases are represented by square matrices of the same size as that of the parent matrix or the same size as $\mathbf{C}$ which is $N_s \times N_s$ in our case. Therefore, an orthonormal basis encoding matrix provides no cost advantage compared to standard shot-record imaging because I would have to perform N_s migrations, the same number as conventional shot-record migration.

While it may not be possible to directly use an orthonormal basis for an encoding matrix, one option is to truncate the columns of the orthonormal basis to form an encoding matrix that provides a substantial cost reduction. The missing columns contain information that is lost in the reconstructed image. Thus, the goal in the forward scheme is to truncate the encoding matrix (orthonormal basis) such that the missing information degrades the image quality only slightly. Soubaras (2006) arrived at a similar idea using a discrete Fourier basis for his experiments.

A logical choice for an orthonormal basis to use as an encoding matrix is the identity matrix itself. To do so, I would truncate columns starting from the end of the identity matrix $\mathbf{I}$ to form the encoding matrix $\mathbf{E}$ such that it is $N_s \times N_e$. However, truncating the identity matrix to form the encoding matrix removes entire shot-gathers from the reconstructed $\mathbf{E}\mathbf{E}^H$ matrix which is not optimal because all of the shots should be used for imaging. Therefore only encoding matrices (truncated orthonormal bases) that use all of the shots are viable for consideration as encoding matrices.

However, there are questions that remain as to which orthogonal bases should be used. For example, should I prefer bases that are sparse, or that have specific characteristics in terms of how fast they reconstruct the image? Are there specific bases that optimally rebuild signals in as few columns as possible? Lastly, which orthogonal bases are real-valued only (i.e. amplitude encodings)?

2.7 The inverse problem

An alternative to the forward problem is to start with a desired cross-talk matrix $\mathbf{C}$, and to decompose (or factorize) it into an encoding matrix of particular shape. I refer to this approach to find an encoding matrix as solving the inverse problem.

To solve the inverse problem, I formulate it as an optimization problem where one can minimize the error between the cross-talk matrix for a given encoding with the ideal cross-talk matrix (the identity matrix). Formally, a simple optimization scheme would use an objective function of the form:

$$\Psi = \left|\left|\mathbf{E}\mathbf{E}^H - \mathbf{I}\right|\right|_p . \tag{2.12}$$

In this case, Ψ is the objective function using the ℓ_p-norm of choice. It is important to note, that this optimization problem is not trivial to solve because:

- the problem is highly non-linear,
- the gradient of the functional with respect to the encoding matrix is not known,
- the choice of the norm is arbitrary, and greatly influences the outcome,
- and the system $\mathbf{E}\mathbf{E}^H$ is severely under-determined.

Due to these issues, I have opted not to explore optimization solutions to this problem. Instead, I choose to focus on factorization approaches to indirectly solve the optimization problem. Some factorizations that are popular in the optimization community are: Cholesky factorization, eigendecomposition, and singular value decomposition. The main difference in the factorizations are the additional constraints that certain factorizations require on the matrix to be decomposed.

2.8 Cost comparison

The most crucial component of blended imaging is assessing the computational advantage that blending provides relative to conventional shot-record migration and to other encodings. To do so, I first restate the cost of conventional shot record migration as,

$$C_{SRM} = C_S N_S, \tag{2.13}$$

where C_{SRM} is the total cost of shot-record migration, C_S is the cost of imaging each shot-gather, and N_S is the number of shot gathers. For blended imaging, where the encoding matrix is not frequency dependent, the overall cost is:

$$C_{BRM} = C_B N_B, \tag{2.14}$$

where C_{BRM} is the cost of blended imaging, C_B is the cost of blending and N_B is the number of blended experiments. However it's not clear whether there is a real cost-advantage merely be examining equation 2.13 and equation 2.14 because C_B is usually greater than C_S due to the larger migration volume, while N_B is less than N_S due to blending. To gain more insight, I expand C_B and C_S for the case of downward continuation migration in three dimensions. For downward continuation, the cost of migrating a single shot-record is proportional to:

$$C_S = 2F N_X N_Y N_Z N_\omega = 2F L_S N_\omega \tag{2.15}$$

In equation 2.15, N_X, N_Y and N_Z are the number of points in the migration volume for a single shot-record in their respective dimensions, N_ω is the number of frequencies, F is the number of floating point operations per point in space and frequency, and 2 represents the cost of extrapolating both the source and receiver wavefields. I combine $N_X N_Y N_Z$ into a single term L_S, which is the migration volume for a single shot-record. In the blended case, the migration volume L_B is the number of distinct points from the migration volume of each

shot-gather in the blended shot-gather. For example, if the blended shot-record consists of shot-records that have independent migration volumes, then $L_B = N_S L_S$. Conversely, if the blended shot-gather is composed of shots that have almost identical migration volumes, then L_B is the number of distinct points in sum of the individual shot-record migration volume. Mathematically, L_B is described as the distinct union of all points in the migration volumes for all shot-gathers in the blended shot gathers as follows:

$$L_B = L_S^1 \cup L_S^2 \cup L_S^3 ... L_S^N. \tag{2.16}$$

By substituting equation 2.16 into equation 2.15, I produce the following expression for the cost of migrating a single blended shot-gather,

$$C_B = 2FL_B N_\omega. \tag{2.17}$$

I take the ratio of equation 2.13 and equation 2.14, substituting the appropriate values for C_S and C_B to produce,

$$C = \frac{2FL_S N_S N_\omega}{2FL_B N_B N_\omega} = \frac{L_S N_S}{L_B N_B} \tag{2.18}$$

where C is the cost ratio of shot-record migration to blended record migration. The value of C directly gives us a relative cost-ratio of shot-record migration to blended migration. For example, if C is greater than one, then blended imaging is more efficient than shot-record migration. Conversely, if C is less than one, then shot-record migration is more efficient than blended imaging. For most practical purposes, a blended imaging scheme must at least break-even with SRM ($C = 1$) to be considered a success, while a good blending scheme would significantly reduce the cost of imaging by at least two-fold, $C > 2$. It is important to note that the reduction in the number of experiments needed for blended imaging N_B is countered by the increase in the migration volume L_B (Zhang *et al.* , 2005; Romero *et al.* , 2000; Liu, 1999).

If the encoding matrix is frequency dependent, then the cost of blended imaging may depend on frequency if the number of encodings varies as a function of frequency. Thus, equation 2.14 becomes:

$$C_{BRM} = 2FL_B \sum_{\omega} N_B(\omega). \tag{2.19}$$

Again, taking the ratio of equation 2.19 and equation 2.13, I arrive at the following expression in the most general of terms for the ratio of the cost of blending and shot-record migration:

$$C = \frac{L_S N_S N_\omega}{L_B \sum_\omega N_B(\omega)}. \tag{2.20}$$

As with equation 2.18, when the value of C in equation 2.20 is less than 1, then SRM is less costly than blended imaging. If C is greater than 1, then SRM is more costly than blended imaging.

2.9 Connection to other fields

Blended imaging is remarkably similar to problems studied in other fields such as: data compression, telecommunications, and digital signal processing.

2.10 Data compression

In particular, the concept of projecting the wavefields onto an orthonormal basis and truncating the resulting matrix is directly similar to *lossy* data compression algorithms referred to as transform algorithms (Rabbani, 1991; Salomon, 2007). Some common orthonormal bases used in transform compression include: Fourier, various wavelet bases, and the discrete cosine basis. The key difference between the bases is the level of compression at a certain level of truncation and the quality of the recovered data at a specific level of compression. In all cases, data are *irreversibly* lost once the orthonormal basis is truncated, because the transform is no longer orthogonal, and the signal is subsequently no longer fully reconstructed. In the context of the imaging problem, the loss of data due to

compression corresponds to an irreversible loss of image quality. However, if the degradation of the image is not overly noticeable, then the advantage of compression may be large enough to justify the degradation of the image.

2.11 Telecommunications

Blended imaging is also closely related to the subject of cross-talk elimination in modern telecommunications (Sklar, 2001). In the context of telecommunications, the problem is that we often want to transmit data from multiple sources (e.g. cellphones) simultaneously over a narrow bandwidth of the EM spectrum. In order to avoid eavesdropping on someone else's conversation (or to avoid cross-talk) radio engineers encode (or modulate) EM transmissions in almost the same manner as shot-record encoding. Some of the most effective telecommunication encoding schemes include: frequency modulation, amplitude modulation, phase-shift modulation, orthogonal frequency-division multiplexing and CDMA (Sklar, 2001). However, these encoding schemes are not usually identical to shot-record encoding schemes because we typically do not use multiple realizations in telecommunications, as that would require sending the same signal repeatedly. Therefore, only a few of the telecommunication encodings can converted into a shot-record encoding scheme. In almost all cases, the direct application of telecommunication encoding schemes substantially increase the amount of computational effort required for imaging because they rely on separation in time or frequency, which reduces the cost advantage of blended imaging.

2.12 Digital filter design

The matrix representation theory of blended imaging is most closely related to discrete digital filters because the convolution of any linear, digital filter with a signal is equivalent to product of matrices in the form of:

$$Y = H^T H X \tag{2.21}$$

In equation 2.21, Y represents the output signal vector, X is the input signal vector, and H is the impulse response of the linear filter, which is usually represented as a row-vector, but can be rectangular matrix (Strang, 1997; Oppenheim, 1999). Because equations 2.10 and 2.21 are identical, I can consider the encoding matrix to be a linear filter that is applied to the reconstructed source- and receiver-wavefields.

Furthermore, because the encoding matrix acts over shots, the equivalent filter, equation 2.21, acts as a filter in the space domain, meaning that all encoding matrices are essentially spatial filters. As with all discrete filters, the frequency response (or wave-number response in the case of the encoding matrix) can be obtained by converting the encoding matrix from the space-domain to the wave-number domain using the discrete Fourier transform which may be useful for comparing encodings (Oppenheim, 1999). Since the cross-talk matrix, C, perfectly describes the result of encoding, I can also extract useful information about the result of the encoding (filtering) process directly from the cross-talk matrix. For example, if the cross-talk matrix is Toeplitz, meaning the diagonals are consistent, then the filter is shift-invariant (Strang, 1997). Shift-invariance is important for shot-encoding schemes because shift-invariant encodings produce the same image regardless of which shot the encoding matrix is started on (i.e. the order of the shots in the encoding matrix does not affect the final blended image).

Chapter 3

Amplitude encoding using singular value decomposition

Due to data loss, it is impossible to reconstruct the identity matrix from the product of $\mathbf{E}\mathbf{E}^H$. An alternate way of approaching the problem is to design the cross-talk matrix $\mathbf{C}$ in advance so that it is close to the $\mathbf{I}$ matrix, and then decompose the $\mathbf{C}$ matrix into two matrices $\mathbf{E}$ and $\mathbf{E}^H$ using singular value decomposition (Eckart & Young, 1936). The benefit is that I control how close the approximation $\mathbf{C}$ is to the $\mathbf{I}$ matrix. In this way, the problem can be reformulated as an inverse problem. Overall, I demonstrate that SVD produces viable amplitude encodings, but ultimately the encodings are suboptimal because I do not know how to optimally design the cross-talk matrix for factorization.

3.1 Singular value decomposition

In SVD, the matrix $\mathbf{A}$ is approximated by,

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T, \tag{3.1}$$

where $\mathbf{A}$ is an $(M \times N)$ matrix, $\mathbf{U}$ is $(M \times N)$ matrix of the left singular vectors, $\mathbf{\Sigma}$ is an $N \times N$ diagonal matrix corresponding to the singular values, and $\mathbf{V}^T$ is $(N \times N)$ matrix of the right singular vectors. In the special case where $\mathbf{A}$ is a real-valued, square, and symmetric matrix, which is the case for some cross-talk matrices, then $\mathbf{U}$ =$\mathbf{V}$ and

equation 3.1 become:

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^T. \tag{3.2}$$

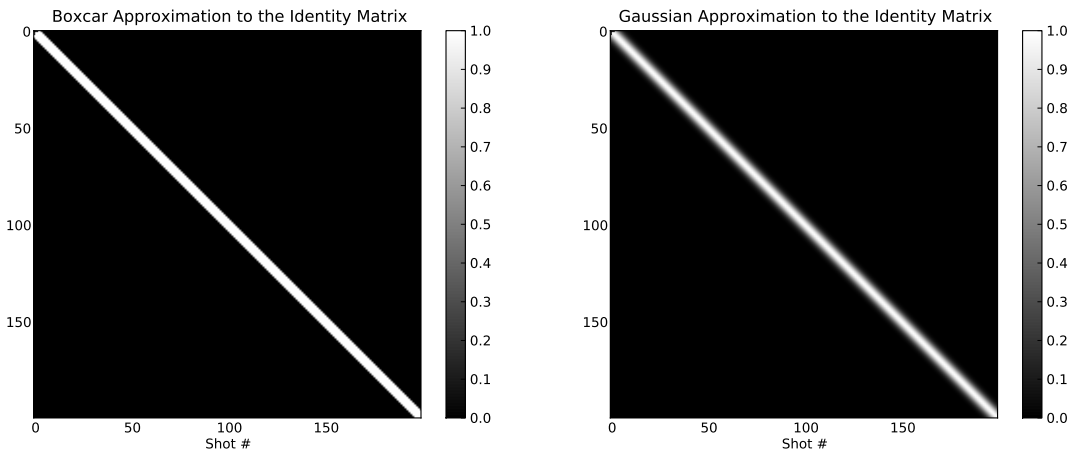
This expression is similar to $\mathbf{C} = \mathbf{E}\mathbf{E}^H$ except that in equation 3.2 $\mathbf{U}$ is a square matrix and that $\mathbf{\Sigma}$ is present. Therefore, I must truncate the matrix $\mathbf{U}$ to form the encoding matrix, i.e. $\mathbf{E} \approx \mathbf{U}_{tc}$ where $\mathbf{U}_{tc}$ is the truncated matrix $\mathbf{U}$. In order to formalize the relationship between the encoding matrix $\mathbf{E}$ and the singular vectors $\mathbf{U}$, I truncate the columns of $\mathbf{U}$ (singular vectors) according to SVD theory, which indicates that I should keep the first N_e columns corresponding to the largest singular values. Large singular values indicate which singular vectors contribute most to the reconstruction of the matrix $\mathbf{C}$. Conversely, small singular values (close to zero) indicate that a singular vector does not have a significant contribution to the reconstruction of $\mathbf{C}$. Additionally, I split the singular value matrix (which is diagonal), σ, by taking the square root of the matrix and multiplying it to U_{tc}. Therefore, $\mathbf{E}$ and $\mathbf{E}^H$ are respectively:

$$\begin{aligned} \mathbf{E} &= \mathbf{U}_{tc}\sqrt{\mathbf{\Sigma}_{tc}}, \\ \mathbf{E}^H &= \sqrt{\mathbf{\Sigma}^{tc}}\mathbf{U}_{tc}^T. \end{aligned} \tag{3.3}$$

I note that in this case, the conjugate-transpose of $\mathbf{E}$ is equivalent to the transpose of $\mathbf{E}$ because the matrix $\mathbf{E}$ is real-valued, i.e.$\mathbf{E}^H = \mathbf{E}^T$. I refer to the constructed encoding matrix $\mathbf{E}$ in equation 3.4 as the Truncated Singular Vector (TSV) encoding. The product of $\mathbf{E}\mathbf{E}^T$ is an *approximation* to $\mathbf{C}$, as a result of truncating U, and C is an *approximation* of the identity matrix I, by design (i.e. $\mathbf{E}\mathbf{E}^T \approx \mathbf{C} \approx \mathbf{I}$).

3.2 Choice of crosstalk matrix

The construction of the $\mathbf{C}$ matrix to approximate $\mathbf{I}$ for the TSV encoding is of paramount importance to the quality of the final image R_e because a $\mathbf{C}$ that is not similar to the identity matrix will introduce strong cross-talk into the final image. Two options for possible $\mathbf{C}$



(a) (b)

Figure 3.1. Example of the boxcar banding (a) and the Gaussian banding (b) of the diagonal of a cross-talk matrix as an approximation to the identity matrix for 250 shots. If I order the shots in the encoding matrix by spatial location, then the banding indicates cross-talk that results in the loss of spatial resolution, because the cross-talk is equivalent to adding nearby shots together.

matrices that approximate **I** are: a banded diagonal, and a banded diagonal using a Gaussian band. Banding the diagonal is equivalent to applying a boxcar filter to the **I** matrix, as in Figure 3.1(a). For the boxcar filter, the full window width is defined as $a = 2\sqrt{3}\sigma$, where a is the window width and σ is the standard deviation. For the Gaussian banded diagonal, I apply a Gaussian filter to the **I** matrix which gradually tapers the values away from the main diagonal in the approximating matrix as in Figure 3.1(b). The Gaussian filter is defined as the normal distribution with a mean μ and a standard deviation σ. In both cases, the additional bands near the main diagonal represent the strong combination of nearby shots in the encoding matrix (provided I order shots in the encoding matrix by proximity in space). Physically, this is identical to constructing small plane-waves, which results in a loss of spatial resolution because the shots constructively interfere together based on the width of the banding in the diagonal (i.e. the wider the diagonal the greater the loss in spatial resolution). However, if the width of the band is sufficiently small then the loss in spatial resolution is negligible.

There are many other options for possible **C** matrices, and this is by no means a comprehensive discussion. Our examples demonstrate how SVD can be used to approximate the Gaussian and boxcar diagonal matrices, and to reconstruct images that are very close to the image obtained by shot-record migration.

3.3 Results

To demonstrate the TSV encoding, I encode all 500 shots from the Sigsbee2A salt-model dataset. Figure 7.4(a) shows the encoding matrix constructed using a boxcar function for banding the diagonal with a half-width of 13 shots. For comparison, Figure 7.7(a) shows the encoding matrix for the Gaussian banded cross-talk matrix. For the Gaussian TSV encoding, I construct an identity matrix with a Gaussian on the main diagonal with $\sigma = 13$ shots. I then decompose the designed **C** matrix into the encoding matrix, Figure 7.7(a). Both encodings are quite similar to one another, until the 30th encoding, at which point

the boxcar encoding matrix must include additional experiments to reconstruct the hard boundary in the cross-talk matrix.

I use a subset of each encoding matrix shown in Figures 7.4(a)-7.7(a) for 25, 50 and 100 encodings by simply truncating the encoding matrix to the adequate number of encodings. Figures 7.2(b)-7.4(b) show the reconstructed cross-talk matrix from using the truncated encoding matrix for the boxcar encoded dataset, and Figures 7.5(b)-7.7(b) show the cross-talk matrices for the Gaussian banded dataset. I note that the cross-talk matrices are similar regardless of the type of banding used in the designed cross-talk matrix. However, the diagonal of the cross-talk matrix for both encodings is not unitary, meaning that some shots are unevenly weighted with respect to their neighbors. Overall this effect is smaller for the Gaussian banded encoding, versus the boxcar banded encoding.

The encoded shot-records are migrated using a downward continuation wavefield extrapolation engine with a conventional cross-correlation imaging condition. The final images for the boxcar banded TSV encoding are shown Figures 7.2(c)-7.4(c). For comparison, Figures 7.5(c)-7.7(c) show the final images for the Gaussian TSV encoding at the same number of encodings as the boxcar encoding scheme.

3.4 Discussion

There are only subtle differences between the images for the boxcar and Gaussian TSV encodings, which is not surprising because the boxcar and Gaussian cross-talk matrices used to construct the $\mathbf{C}$ matrix used for decomposition are quite similar. Because the cross-talk matrices used for factorization are similar, SVD produces nearly identical orthogonal bases for the singular vectors which I use to construct the encoding matrix, which explains why the encodings are quite similar, Figure 7.4(a) and Figure 7.7(a). However, I note that the encoding matrices are quite different after 30 encodings, because the boxcar encoding includes what appear to be sporadic weights in an attempt to reconstruct the hard boundary in the initial cross-talk matrix. The Gaussian encoding by comparison smoothly tapers from

large values to small weights as the number of encodings increases, which explains why the image quality does not really improve as I increase the number of encodings: additional encodings do not add much new information to the image. I see this same phenomena in the boxcar encoding matrix as well.

The image quality of the migrated images is suboptimal, Figures 7.5(c)-7.7(c) and Figures 7.2(c)-7.4(c), especially when compared against the decimated shot-record migration images, Figures 7.61(a)-7.61(c), at comparable cost. The lack of quality is caused by the presence of small weight encodings in the encoding matrix, Figure 7.7(a). In order to rectify this problem I should choose a different cross-talk matrix to decompose for our encoding, but it is not entirely clear how I should design the cross-talk matrix to maximize the contribution of each additional encoding.

Chapter 4

Amplitude encoding by basis projection

As the theory of blended imaging shows, the outcome of blended imaging is fundamentally determined by the choice of the basis that one projects the source- and receiver-wavefields onto. Because different bases reconstruct the cross-talk matrix with different efficiency, and in a different fashion (i.e. produce different cross-talk matrices), the choice of which basis to project onto is critical. To illustrate the effect of choosing a basis, I demonstrate the effect of using different bases for encoding when the number of experiments is fixed on the Sigsbee2A dataset. In each case, the basis that I chose is orthogonal, meaning that if I choose to use the same number of encodings as shots, that the results are identical to that of shot-record migration. When I use a fewer number of encodings (i.e. truncate the basis), the resultant images are different because of how different bases gradually reconstruct the identity matrix.

I start by discussing two previously known encoding schemes: plane-wave migration and modulated-shot migration. Then, I extend the same principles for phase-encodings to real-valued bases (e.g. amplitude-only encodings). For amplitude-only encodings, I demonstrate the usage of: the discrete cosine and sine transforms, the discrete Hartley transform, and a custom pseudo-basis. Additionally, I examine the possibility of using wavelet bases for blended imaging, because they are well-known bases with interesting properties such as sparsity. In all cases, the encodings in this chapter are frequency independent, meaning that I encode every frequency in the dataset for each encoding. I refer to this type of encoding

as "fixed cost" encoding, meaning that I compare the image quality of different encoding schemes at the same number of blended experiments.

4.1 Converting an orthgonal basis to an encoding

As I discuss in Section 2.6, orthogonal bases are natural candidates for encodings for two reasons: 1 - the row and column vectors are orthogonal to one another which means that the information in them is unique, and 2 - the product of the orthogonal basis with its transpose is the identity matrix, which is one of the goals for designing an encoding matrix. However, all orthogonal bases are square matrices, by definition, which means that I do not want to directly convert an orthogonal basis into an encoding matrix because it would require using the same number of encodings as shots in the survey (i.e. $N_e = N_s$) which would not save any cost.

Instead, I convert the orthogonal basis into an encoding matrix by simply truncating the column vectors of the orthogonal basis. For example, if I have an orthogonal basis matrix, $\mathbf{Q}$, which is $(N_s \times N_s)$, then I construct the encoding matrix as follows,

$$\mathbf{E}_{m,n} = \mathbf{Q}_{m,n}, \mathrm{n} = 0, 1, ..., N_e, \tag{4.1}$$

where N_e is the number of blended experiments to be used, m is the shot-record index, and n is the blended experiment index. The main question regarding truncation is which column vectors should be kept in the encoding matrix. For example, should I just take the first N_e columns from the orthogonal basis? Or should I use some other combination. There is little guidance on this subject, but I propose using the first N_e columns of the orthogonal basis and using that to form the encoding matrix since this typically reconstructs a cross-talk matrix that is banded along the main diagonal. However, other choices for columns may be possible, but I would need additional information to make that decision, such as information about the strength of the cross-talk.

Additionally, it is important to note that the ideal situation would be to use a rectangular orthogonal basis as an encoding matrix. As far as I know, no such orthogonal basis (matrix) exists because it would imply that one could fully reconstruct the image using fewer encodings than shot-records, which would violate the "no free-lunch" principle. Even though the optimal basis does not exist, I still examine the effects of truncating already known, square orthogonal bases.

4.2 Fourier-related encodings

A well-known orthonormal basis is the complex-valued Fourier basis (Strang, 1997; Oppenheim, 1999; Soubaras, 2006). Because the Fourier basis is well studied, I might logically start by looking for encodings using the Fourier basis. In fact, two such encodings are already well-known: modulated shot-migration (Soubaras, 2006) and plane-wave migration (Tieman, 1997; Liu, 1999; Zhang *et al.* , 2005) , both of which are phase-shift encodings. Additionally, real-valued encodings (amplitude only) exist that are closely related to the Fourier basis, such as: the discrete cosine transform, the Hartley basis, and the trigonometric pseudo-basis. Since all of these encodings are related to the Fourier basis, they share some common features. For example, all of the aforementioned encodings can be understood in the spatial-wavenumber domain, and are affected by the periodization distance. I start my discussion with these features to explain how they affect Fourier encodings.

4.3 Spatial-sampling and periodization distance

To produce a Fourier-related encoding scheme, I truncate the matrix by keeping the first N_e columns of the orthonormal basis which corresponds to selecting complex or real-valued sinusoids with certain angular frequencies from the basis. In most cases, I keep sinusoids with low frequencies in the encoding matrix, which constructs a portion of the image with a limited spatial frequency band, (Stork, 1999; Zhang *et al.* , 2005; Soubaras, 2006). This relationship is derived directly from the Fourier basis encoding matrix, also

known as modulated-shot migration (Soubaras, 2006). For reference, the discrete Fourier transform encoding matrix is given by,

$$\mathbf{E}_{m,n} = cos\left(2\pi mn/\wp\right) - jsin\left(2\pi mn/\wp\right), \tag{4.2}$$

where m is the shot index, n is the encoding index, and $\wp$ is the periodization distance, which is typically N_s for the unitary DFT matrix. If I substitute into equation 4.2, $m = x_m/\Delta x$ where Δx is the shot-sampling interval, and the periodization distance $x_\wp$ is $\wp\Delta x$, then equation 4.2 becomes:

$$\mathbf{E}_{m,n} = cos\left(2\pi x_m n/(x_\wp)\right) - jsin\left(2\pi x_m n/(x_\wp)\right). \tag{4.3}$$

If I let K represent the spatial wave-number, and $\Delta K = 1/\wp\Delta x$ then equation 4.3 is equivalent to:

$$\mathbf{E}_{m,n} = cos\left(2\pi x_m n\Delta K\right) - jsin\left(2\pi x_m n\Delta K\right). \tag{4.4}$$

From equation 4.3 it is clear that when the encoding matrix is square then the encoding matrix is unitary, which follows from the DFT basis. Additionally, equation 4.4 shows that each additional encoding (specified by the n-index) increases the angular frequency of the argument to the sine and cosine functions. By using the relationship, $K = n\Delta K$, equation 4.4 shows that each encoding adds an additional spatial wave number to the encoding matrix up to the maximum spatial wave-number as specified by the Nyquist sampling criterion (i.e. in the limit DFT encoding is equivalent to SRM). However, when I truncate the DFT basis or truncate the encoding matrix, I do not sample all possible wave-numbers up to the Nyquist criterion, which means that I do not fully sample in the wave-number domain. Since I usually prefer to keep encodings corresponding to low wave-numbers, I wind up constructing an encoding matrix that is equivalent to an ideal low-pass spatial filter, when I truncate Fourier related bases. It's important to note that the ideal

low-pass filter for complex-valued bases (i.e. the Fourier basis) are not zero-phase, which means that the shot-records are phase-shifted with respect to one another. For comparison, real-valued encodings always produce a low-pass spatial filter that is zero-phase, because there is no complex component to the encoding matrix.

The periodization distance is also a critical parameter that greatly affects the outcome of a Fourier-related encoding scheme because if equation 4.2 is changed so that $\wp \neq N_s$, or if I change the periodization distance to be less than $N_s \Delta x$ then I change the period of the sinusoids in the encoding matrix from N_s to $\wp$. By changing the period, I change the spatial sampling rate, ΔK. Typically, ΔK becomes larger, meaning that the same number of encodings obtain more spatial information than before (i.e. the quality of the image improves). At the same time, changing the periodization distance introduces strong, coherent cross-talk between shots that are now a full-period away from one another in the encoding matrix because the weights for those shot-records are the same, (i.e. shots that are $\wp$ away from one another have maximum cross-talk). However, this strong cross-talk, may be acceptable because there are situations in which some shots never contribute cross-talk to other shots. For example, consider the case where two shots are on opposite sides of a large survey. Clearly, those two shots have migration apertures that do not overlap, and thus they can not possibly contribute to cross-terms in $\mathbf{C}$ even though they may exist. For most surveys, the distance at which two shots do not share migration aperture is smaller than the total survey area, and so $x_{\wp}$ and subsequently $\wp$ may be much smaller than N_s depending on shot spacing. Formally this relationship is given as,

$$\wp_{min} = 2x_{aperture}/\Delta x, \tag{4.5}$$

where $x_{aperture}$ is the width in space of the migration aperture for one shot, assuming that all shots have the same migration aperture. As Soubaras (2006) notes, the minimum constraint for the periodization value, substantially affects the cost advantage of blended

imaging, and so is a crucial parameter for Fourier-related blending schemes.

4.4 Modulated-shot migration (DFT)

Equation 4.2 shows the partial encoding matrix for modulated shot-migration, which I refer to as the DFT encoding. As previously mentioned, the DFT encoding matrix is directly related to an optimal low-pass spatial filter, which means that modulated-shot migration only reconstructs the image up to a maximum wave-number, K_{max}. However, modulated-shot migration must include both positive and negative values for K in order to properly illuminate the subsurface. I demonstrate why this is the case mathematically by considering the case of a single encoding with a particular K value, whose encoding is represented as:

$$\mathbf{E}_k = exp(-\mathbf{j}A\rho k) = cos(A\rho k) + \mathbf{j}sin(A\rho k), \tag{4.6}$$

where A represents the collection of all terms (including K) in the numerator and denominator of equation 4.2 and k is the shot index. I construct the cross-talk matrix for this single experiment as follows,

$$\mathbf{C}_{k,l} = \mathbf{E}\mathbf{E}^H = [cos(A\rho k) + \mathbf{j}sin(A\rho k)][cos(A\rho l) - \mathbf{j}sin(A\rho l)], \tag{4.7}$$

which further expands to,

$$\begin{aligned}\mathbf{C}_{k,l} =& cos(A\rho k)cos(A\rho l)\\ &+\mathbf{j}sin(\rho k)cos(A\rho l)\\ &-\mathbf{j}sin(A\rho l)cos(A\rho k)\\ &+sin(A\rho k)sin(A\rho l).\end{aligned}$$

The presence of the complex-valued terms in equation 4.8 represent phase-shifted cross-talk in the final image. I note that these terms vanish when $k = l$, which indicates that the

desired image is not phase-shifted. In most cases, cross-terms from out of phase wavefields produce strong cross-talk that significantly degrades the quality of the image (Romero *et al.* , 2000). However, the cross-terms in equation 4.8 have a regular pattern to them, which instead causes a distortion in the illumination of the final image. The simplest method to remove the cross-terms from equation 4.8 is to concatenate another encoding whose $A = -A$ because the sine function is odd (e.g. $sin(-x) = -sin(x)$). By substituting $A = -A$ into equation 4.8 I arrive at:

$$\begin{aligned}\mathbf{C}_{k,l} =& cos(A\rho k)cos(A\rho l) \\ &-\mathbf{j}sin(A\rho k)cos(A\rho l) \\ &+\mathbf{j}sin(A\rho l)cos(A\rho k) \\ &+sin(A\rho k)sin(A\rho l).\end{aligned} \tag{4.8}$$

I add equation 4.8 and equation 4.8 which eliminates the complex-valued terms from both expressions, and reduces both expressions to:

$$C_{k,l} = cos(A\rho k)cos(A\rho l) + sin(A\rho k)sin(A\rho l). \tag{4.9}$$

Thus, for every encoding in the DFT encoding matrix, I must include another encoding whose argument $A = -A$ for that particular encoding. This effectively doubles the number of encodings that must be used for the DFT encoding matrix which is effectively the same requirement as that for the related amplitude encodings.

Also, when A is equivalent to the arguments passed to the cosine and sine functions in the DCS encoding, equation 4.24, then the cross-talk matrix for modulated-shot migration (DFT encoding) the same as the cross-talk matrix for the DCS amplitude encoding. Which means that the DFT and DCS encodings produce *identical* images at the same computational cost, number of blended experiments. In fact, the only difference between the two encodings, is how they are physically implemented during the blending process (e.g.

phase-shifts versus amplitude weights).

Figures 7.36(c)-7.39(c) are the images produced by the DFT encoding ($\wp = 250$) at the same cost as the DCS encoding. The images are identical to those produced by the DCS encoding, Figures 7.28(c)-7.31(c). The cross-talk matrix for the DFT encoding, Figures 7.36(b)-7.39(b), is the same as that for the DCS encoding, Figures 7.28(b)-7.31(b).

4.5 Plane-wave migration

For plane-wave migration, the encoding matrix is defined by the maximum time-delay and the number of encodings. As mentioned in the literature, the maximum time-delay also determines the maximum wave-number in the blended image, and subsequently the cross-talk present in the final image (Stork, 1999; Liu, 1999; Zhang *et al.* , 2005). To illustrate this consider the expression for the frequency-dependent encoding matrix for plane-wave migration which is given by:

$$\mathbf{E}_{m,n}(\omega) = exp(-j\omega\tau_{m,n}), \tag{4.10}$$

where $\tau_{m,n}$ is the time-delay in seconds, at the mth shot for the nth plane-wave encoding. The time-delay, $\tau_{m,n}$, is related to the shot position and the blended experiment by the relation,

$$\tau_{m,n} = \frac{(t_{max} - n2t_{max}/N_e)m}{N_s} - \frac{t_{max} - n2t_{max}/N_e}{2}, \tag{4.11}$$

where n is the nth encoding, m is the shot index,Δx is the shot-sampling interval, t_{max} is the maximum time-delay in seconds, and N_e is the number of blended experiments. Equation 4.11 requires $2t_{max}$ because plane-wave migration must create plane-waves with both positive and negative time-delays to ensure proper illumination throughout the image.

The cross-talk matrix for plane-wave migration is then given by:

$$\mathbf{C}_{k,l}(\omega) = \sum_{n=0}^{N_e} exp(-j\omega\tau_{k,n} - \tau_{l,n}). \tag{4.12}$$

I expand equation 4.12 into:

$$\mathbf{C}_{k,l}(\omega) = \sum_{n=0}^{N_e} cos(\omega\tau_{k,n} - \tau_{l,n}) + \mathbf{j}sin(\omega\tau_{k,n} - \tau_{l,n}). \tag{4.13}$$

When $k = l$ in equation 4.12, the complex-exponential is 1, which produces the desired image. When $k \neq l$, then there is strong coherent cross-talk with a non-zero phase component (i.e. the sine term in equation 4.13 is non-zero). However, I see that when both positive and negative values of τ are swept, the sign of the phase-terms reverses, thereby removing them. Therefore, equation 4.13 is nearly equivalent (I use the product-to-sum formulae to get them into similar format) to the cross-talk matrices for both the DFT and DCS encoding. However, the frequency-dependence of plane-wave migration is a significant difference between the encodings. As with the DFT encoding, plane-wave migration requires two encodings for each spatial wave-number to properly illuminate the subsurface.

Figures 7.40(c)-7.43(c) show the high quality images produced for the Sigsbee dataset using a nine second maximum time-delay. However,the images have low-amplitude artifacts that are introduced by sampling only some ρ values instead of all ρ values as dictated by sampling theory. I revisit the subject of plane-wave encoding in Chapter 5.

4.6 Discrete cosine basis (DCT-II)

The discrete cosine basis is a well-known transform used in transform coding and data compression that is directly related to the real-valued Fourier transform (Rabbani, 1991; Strang, 1997; Tsitsas, 2010). The DCT exists in a few varities (I-IV), but I focus on the DCT-II because it is widely used in a number of fields. I define the DCT-II encoding matrix

to be the same as the DCT-II matrix given by:

$$\mathbf{E}_{m,n} = b(n)\sqrt{\frac{2}{N_s}}cos\left(\frac{\pi(m+\frac{1}{2})n}{\wp}\right). \tag{4.14}$$

Where m is the shot-index, n is the encoding index, N_s is the number of shots, $\wp$ is the periodization distance (discussed in 4.3) and $b(n)$ is a scaling factor defined as,

$$b(n) = \begin{cases} \frac{1}{\sqrt{2}} & n = 0 \\ 1 & n \neq 0. \end{cases} \tag{4.15}$$

The cross-talk matrix for the DCT is given by,

$$\mathbf{C}_{k,l} = \sum_{n=0}^{N_e} b(n)\frac{2}{N_s}cos\left(\frac{\pi(k+\frac{1}{2})n}{\wp}\right)cos\left(\frac{\pi(l+\frac{1}{2})n}{\wp}\right), \tag{4.16}$$

where k and l are the shot-indices corresponding to the row and column indices of the cross-talk matrix, which refer to individual shot-records. As equation 4.16 shows the diagonal of the cross-talk matrix ($k = l$) is equal to,

$$\mathbf{C}_{k,k} = \sum_{n=0}^{N_e} b(n)\frac{2}{N_s}cos^2\left(\frac{\pi(k+\frac{1}{2})n}{\wp}\right). \tag{4.17}$$

Because the diagonal elements in equation 4.17 are not one unless $N_e = N_s$, there is an amplitude distortion of the desired image caused by the DCT encoding. Also, the DCT is not a shift-invariant encoding scheme, which is apparent from equation 4.16 as the diagonals are not equivalent (i.e. the cross-talk matrix is not Toeplitz). Thus, if I change the shot that I start the encoding on (i.e. shift the encoding matrix), the resultant image is not the same as one created using the DCT encoding with a different shift. The lack of shift-invariance and the amplitude distortion become less important as the number of encodings increases, because the partial sum of cosines from different encodings approaches one and the distortion between shots rapidly decreases.

Its also important to note that changing the periodization distance for the DCT-II encoding substantially changes the cross-talk matrix because the DCT-II is designed to be orthogonal under very strict conditions (e.g. periodic over 4π instead of 2π). Thus, when I change the periodization distance $\wp$ in equation 4.14 I change the orthogonality of the DCT encoding matrix which results in a considerable increase in the amount of cross-talk in the final image.

Additionally, I note that the DCT-II requires twice as many spatial wave-numbers as the DFT encoding because the argument to the cosine function is a function of π instead of 2π, meaning that only the even values of n capture the same spatial information as the encodings in the DFT. This might lead one to think that I could remove the odd values of n from equation 4.14 but doing so is equivalent to changing the periodization distance. Because of this constraint, I find that there is no substantial cost advantage to using the DCT scheme over the DFT encoding scheme (i.e. I must use the same number of encodings either way).

Figure 7.11(a) shows the encoding matrix for the discrete cosine transform when the periodization distance, $\wp$, equals the number of shots, N_s. For comparison, Figure 7.15(a) shows the encoding matrix for the periodized DCT encoding, when the periodization distance, $\wp$, equals $N_s/2$ which is the optimal periodization distance for Sigsbee because shots that are 250 shot-locations away from one another are outside one another's migration aperture. The cross-talk matrices are shown in Figures 7.8(b)-7.11(b) and Figures 7.12(b)-7.15(b) for the regular DCT encoding and the periodized DCT encoding respectively. These figures show clearly that changing the periodization distance introduces additional bands of strong, coherent off-diagonal terms to the cross-talk matrix. Thus, changing the periodization distance is not optimal for the DCT. Figures 7.8(c)-7.11(c) show the images produced for the regular DCT encoding and Figures 7.12(c)-7.15(c) show the images for the periodized DCT encoding. While the spatial resolution of the image for the periodized DCT encoding is clearly superior, the degradation of the image from caused by the additional

cross-talk from periodization substantially degrades the final image. Therefore, I do not recommend periodizing the DCT encoding.

4.7 Discrete sine basis (DST-II)

The discrete sine transform is closely related to the discrete cosine transform as a sine function is simply a cosine function with a phase shift of $\frac{\pi}{2}$ (Strang, 1997; Tsitsas, 2010). As with the DCT, the DST comes in a few varieties, but I focus on the DST-II because of its symmetry to the DCT-II amplitude encoding. The DST-II encoding matrix is defined as:

$$\mathbf{E}_{m,n} = b(n) sin\left(\frac{\pi(m+\frac{1}{2})n}{\wp}\right). \tag{4.18}$$

Where $b(n)$ is a scaling factor defined as,

$$b(n) = \begin{cases} \frac{1}{\sqrt{2}}, & n = N_s - 1 \\ 1 & n, \neq N_s - 1. \end{cases} \tag{4.19}$$

Similarly to the DCT, the cross-talk matrix for the DST is equal to,

$$\mathbf{C}_{k,l} = \sum_{n=0}^{N_e} b(n)\frac{2}{N_s} sin\left(\frac{\pi(k+\frac{1}{2})n}{\wp}\right) sin\left(\frac{\pi(l+\frac{1}{2})n}{\wp}\right). \tag{4.20}$$

As with the DCT, the DST cross-talk matrix is not unitary on the diagonal, and the diagonals off the main diagonal are not equal, meaning that the DST is not a shift-invariant encoding and creates the same amplitude distortion as the DCT. However, there is a unique relationship between the distortion caused by the DCT and the DST in that their respective distortions are $\pi/2$ out of phase with one another, meaning that the amplitude distortion caused by either the DCT or DST is completely cancelled by the other. I exploit this relationship later in the discrete cosine-sine (trigonometric) basis.

Figures 7.19(a) shows the encoding matrix for the DST-II encoding scheme. Notice

that the encoding matrix is the same as the one in Figure 7.11(a) but shifted. Because the DST is equal to the DCT shifted by $\pi/2$ the encodings provide distinct information from one another, as the cross-talk matrices show for the DST encoding, Figures 7.16(b)-7.19(b). Figures 7.16(c)-7.19(c) show the migrated images for the non-periodized, $\wp = N_s$, DCT-II encoding. As with the DCT, changing the periodization distance introduces a large amount of coherent cross-talk into the final image as shown byFigures 7.20(c) -7.23(c), which are for the DST-II encoding, periodized with $\wp = N_s/2$.

4.8 Hartley basis

The Hartley basis is the real-valued analogue of the discrete Fourier transform, which means that it is most closely related to the Fourier basis (Tsitsas, 2010). The Hartley encoding matrix is defined as,

$$\mathbf{E}_{m,n} = cos\left(\frac{2\pi mn}{\wp}\right) + sin\left(\frac{2\pi mn}{\wp}\right) = cas\left(\frac{2\pi mn}{\wp}\right). \tag{4.21}$$

In the literature, $sin(x) + cos(x)$ is referred to as $cas(x)$. The relation to the Fourier transform is directly apparent, but the presence of the addition between the sine and cosine introduces additional cross-terms into the cross-talk matrix as follows:

$$\begin{aligned}\mathbf{C}_{k,l} = \sum_{n=0}^{N_e} & cos\left(\frac{2\pi kn}{\wp}\right) cos\left(\frac{2\pi ln}{\wp}\right) \\ & + cos\left(\frac{2\pi kn}{\wp}\right) sin\left(\frac{2\pi ln}{\wp}\right) \\ & + sin\left(\frac{2\pi kn}{\wp}\right) cos\left(\frac{2\pi ln}{\wp}\right) \\ & + sin\left(\frac{2\pi kn}{\wp}\right) sin\left(\frac{2\pi ln}{\wp}\right).\end{aligned} \tag{4.22}$$

From equation 4.22 it is clear that there are additional off-diagonal components represented by the cross-terms between the $cos(x)sin(y)$ terms present in the expression. I note, that I

could remove these components by using negative frequency values for the same encodings in equation 4.21 and then adding them to equation 4.21, because the sine function is odd (e.g. $sin(-x) = -sin(x)$) and the addition would remove the cross-terms. However, doing so would double the number of encodings required in the transform, fromN_e to $2N_e$. Also important is that the main diagonal of the Hartley cross-talk matrix is almost unitary because of the presence of $cos^2(x) + sin^2(x)$ in equation 4.22.

Figure 7.27(a) shows the encoding matrix for the Hartley basis, periodized with $\wp = 250$, for 250 blended experiments for 500 shot-records. I note that there is a strong similarity with the encoding matrix for the DCT, Figure 7.11(a). Figures 7.24(b)-7.27(b) show the cross-talk matrices for the Hartley encoding. Notice the strong bands in the cross-talk matrix which introduce strong, coherent cross-talk in the final images, Figures 7.24(c)-7.27(c). The images produced by the Hartley encoding are of superior quality at a fewer number of encodings compared to those from the DCT or DST encodings because they obtain more spatial information in fewer encodings than either the DCT or DST. The downside to the Hartley encoding is that it introduces strong cross-talk in the images because of the presence of the cross-terms in equation 4.22.

4.9 Discrete trigonometric basis (DCS)

The trigonometric pseudo-basis (DCS) is similar to the Hartley basis, but removes the cross-terms from equation 4.22 by separating the sine and cosine in the encoding matrix, which prohibits them from "talking" to one another during imaging. There is a major disadvantage to doing so though: each wave-number requires two encodings whereas the corresponding wave-number in the Hartley transform requires only one encoding. However, the Hartley encoding requires another encoding to remove the cross-terms, which means that the DCS and Hartley encoding matrices require the same number of encodings to produce the same quality image. Thus, I prefer the DCS encoding to the Hartley encoding

because of its simplicity. The encoding matrix for the trigonometric pseudo-basis (DCS) is:

$$\mathbf{E}_{m,n} = \begin{cases} 1, & n = 0 \\ cos\left(2\pi m(n/2)/\wp\right), & \text{even n} \\ sin\left(2\pi m(\text{floor}(n/2)+1)/\wp\right), & \text{odd n} \end{cases} \tag{4.23}$$

It's important to note that equation 4.23 is not strictly an orthonormal basis, and indeed the DCS basis requires $2N_s$ encodings to become a strict basis. Since I usually truncate the basis, I am not overly concerned about doing so here, but I note that I must guarantee that I truncate both the sine and cosine term for each wave-number that I choose to keep from equation 4.23. In any case, the cross-talk matrix for the trigonometric basis simplifies to:

$$\mathbf{C}_{k,l} = \sum_{n=0}^{N_e} cos\left(\frac{2\pi kn}{\wp}\right) cos\left(\frac{2\pi ln}{\wp}\right) + sin\left(\frac{2\pi kn}{\wp}\right) sin\left(\frac{2\pi ln}{\wp}\right). \tag{4.24}$$

In comparison to the cross-talk matrices for the DCT, DST and Hartley encodings, the cross-talk matrix for the DCS encoding is shift-invariant and unitary along the diagonal meaning that there is no amplitude distortion. These are important characteristics of the DCS encoding because it ensures that I preserve relative amplitude changes in the image, even when I use far fewer blended experiments than I would use for SRM.

Figure 7.31(a) shows the encoding matrix for the DCS encoding with a periodization distance $\wp = N_s$. The corresponding cross-talk matrices are shown in Figures 7.28(b)-7.31(b). For comparison, the encoding matrix for a DCS encoding periodized with $\wp = N_s/2$ is shown in Figure 7.35(a). The corresponding cross-talk matrix for the periodized encoding is shown in Figures 7.32(b)-7.35(b). The cross-talk matrix clearly shows the bands near shot 250 from changing the periodization distance $\wp$. Figures 7.28(c)-7.31(c) and Figures 7.32(c)-7.35(c) show the images for the DCS encoding and the periodized DCS encoding respectively. The periodized images are of superior quality, because they reconstruct more wave-numbers in the image at the same computational cost. Thus, periodizing the DCS encoding is highly

recommended.

4.10 Wavelet bases

While Fourier-related bases are natural candidates for amplitude encodings, they are not the only real-valued orthonormal bases that exist. Another well-known set of real-valued bases are wavelet bases. Conventionally, wavelets are implemented using filter banks, but almost all wavelet transforms may be written as a discrete valued matrix (Strang, 1997). Additionally, wavelets are usually excellent bases because they are compact in space, they are usually quite sparse when compared to other bases (Strang, 1997). Sparsity and spatial compactness (which are related) are useful in shot-encoding because they allow us to construct blended images using far fewer shots in a blended experiment, meaning that the cost penalty assessed by migrating over a large aperture is greatly reduced.

For wavelet bases, I only show the encodings when the number of blended experiments is a power of two (i.e. $N_e = 2^k$, with k any positive integer). This is because wavelets are well-defined for signals whose length is a power of two. Subsequently, the cross-talk introduced by the wavelet encodings is minimized for N_e at powers of two, which allows us to make the fairest comparison between wavelet encodings and other encodings.

4.11 Haar wavelet basis

The first wavelet basis that I examine is the Haar wavelet basis (Strang, 1997). The Haar wavelet basis is composed of wavelets that whose values alternate between vlaues of $+1, -1$ and 0. For the Haar wavelet, the mother wavelet is defined as,

$$\psi(t) = \begin{cases} 1 & 0 <= t < 0.5 \\ -1 & t >= 0.5 \end{cases} \tag{4.25}$$

The encoding matrix for the Haar wavelet is constructed by shifting and dilating the mother wavelet, equation 4.25, for each power of 2 according to the discrete wavelet transform (Strang, 1997). This process can be written analytically, but it is not useful for examining the cross-talk matrix or the encoding matrix, and so I refer the reader to Strang (1997) for details about how to construct the Haar wavelet encoding matrix.

Figure 7.47(a) shows the encoding matrix for 256 encodings from the Haar wavelet basis. Note that the encoding matrix is very sparse (many shots have zero weights), and that many of the encodings require only a few shots. Figures 7.44(b)-7.47(b) show the corresponding cross-talk matrices. The cross-talk matrix is similar to those obtained by Fourier-related encodings, but is blocky along the main diagonal indicating that the Haar encoding is not shift-invariant. The image quality is comparable to the DCT, DST, and DCS encodings, but at slightly greater cost.

4.12 Daubechies-4

The Daubechies-4 wavelet belongs to a famous family of wavelets that Ingrid Debauchies first discussed following her groundbreaking work on wavelets (Strang, 1997). Of the wavelets that she discussed, the Daubechies-4 wavelet is not particularly special besides its prevalence in signal processing applications. I construct the Daubechies-4 (DAUB-4) wavelet encoding matrix by converting the filter bank coefficients used to define the mother wavelet into an encoding matrx. Strang (1997) details how to carry out this operation.

Figure 7.51(a) shows the encoding matrix for the Daubechies-4 wavelet encoding. The encoding is remarkably similar to the Haar encoding, Figure 7.47(a). Figures 7.48(b)-7.51(b) show the cross-talk matrices for the Daubechies-4 wavelet encoding at various numbers of encodings. I note that there is an interesting behavior in the cross-talk matrix where there is a strong, uneven weighting along the main diagonal and small-groups of cross-talk that appear in a regular pattern. This phenomena is more visible in Figure 7.48(b) and Figure 7.49(b). Figures 7.48(c)-7.51(c) show the images constructed using the Daubechies-4

wavelet encoding scheme. I note that the images are quite similar to those for the Haar wavelet encoding, even though the wavelets are quite different.

4.13 Other orthogonal bases

Two other prominent real-valued orthonormal bases in signal processing are the: Hadamard and Walsh bases. These bases are interesting for blended imaging because they are in a sense the opposite of wavelets, in that they are dense bases and are more closely related to the Fourier basis than any other one. As with wavelets, both the Hadamard and Walsh bases are defined for signals with length of $N = 2^k$. Thus, they work best when I have the number of shot-records as a power of two. Unfortunately, this is not often the case because I do not usually acquire data with the constraint that the number of shots must be a power of two. Therefore, I either have to convert the encoding matrix (using interpolation) or add additional shot-records that are composed of zeros (similar to padding a signal with zeros). In most cases, I prefer the latter because it is easier to construct a shot-record consisting of zeros, which I use to construct additional shot-records until I have $N_s = 2^k$.

4.14 Hadamard basis

The Hadamard basis is a well-known basis that is composed of basis vectors that consist of all $+1$ or -1 values. The Hadamard basis is defined by a recursive relationship with its sub-matrices that defines the basis as a function of the power of 2. For example, the Hadamard basis for a signal with length 2 is given by:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{4.26}$$

The recursive relationship is given by the following relationship,

$$H_k = \begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}, \tag{4.27}$$

where k is the integer exponent of $N = 2^k$. Thus to construct a Hadamard matrix for a signal of length 512, one would recursively construct the Hadamard matrix for each value of k from $k = 1, \ldots, 9$. The encoding matrix is then taken from the equivalent Hadamard matrix by simply truncating the number of columns to be equal to the desired number of encodings N_e. Because of the recursive relationship, I cannot write the expression for the cross-talk matrix analytically for the Hadamard basis.

Figure 7.55(a) shows the encoding matrix for the Hadamard encoding. The encoding matrix has some interesting patterns that are apparent in it, and these patterns cause strong cross-talk to form along diagonals in the cross-talk matrices as shown in Figures 7.52(b)-7.55(b). Figures 7.52(c)-7.55(c) show the final images using the Hadamard encoding for various numbers of encodings. The Hadamard encoding produces images that are somewhat unique compared to the other encodings because they capture all of the spatial information (wave-numbers) immediately, but introduce strong coherent cross-talk at the same time. In some ways, one could consider the Hadamard encoding to be the exact opposite of the Fourier-based encodings which gather spatial resolution slowly but have less cross-talk.

4.15 Walsh basis

The Walsh basis is related to the Hadamard basis by a simple relation: the Walsh basis is a sorted Hadamard basis. To construct the Walsh basis, I construct the Hadamard basis and then sort it by frequency, where frequency is a relative concept relating the number of sign $(+/-)$ changes in each encoding. The Walsh basis is sorted from "low" frequency to "high frequency." Again, because there is not an explicit analytical form for the encoding matrix, I cannot formulate the cross-talk matrix explicitly.

Figure 7.59(a) shows the encoding matrix for the Walsh encoding. The relation to the Hadamard encoding is not immediately obvious by examining Figures 7.55(a) and Figure 7.59(a). However, the sorting substantially affects the cross-talk matrix as shown in Figures 7.56(b)-7.59(b). The images produced by the Walsh encoding, Figures 7.56(c)-7.59(c), are very similar to those produced by the DCT, DST, and DCS encodings.

4.16 Results

I compare the result of the separate encodings against the results of decimated shot-record migration at the same relative computational cost (number of experiments). Figures 7.61(a)-7.61(d) show the images produced by decimated SRM at 25, 50, 100 and 250 migrated shots respectively. Figures 7.61(a)-7.61(d) show the difference between the decimated images and the image from SRM using all the shot-records. Clearly the decimation introduces numerous artifacts from spatially aliasing our data because I do not apply a low-pass filter to the shot-records before decimating. The DCT, DST, Hartley, and DCS encodings produce images that have less spatial information in them than the decimated images, but they do not show aliasing artifacts. As the number of encodings increases all of the Fourier-related encodings reconstruct more spatial information in the image, and reduce the cross-talk apparent in the image. The Haar and Daubechies-4 wavelet encodings also show similar results to the Fourier-related amplitude encodings at slightly greater computational cost. The Hadamard encoding produces images with maximum spatial information because the cross-talk is confined to diagonals far away from the main diagonal, which reduces the spatial filtering by encoding. The Walsh encoding on the other hand adds many shots that are near to one another, which reduces the spatial information at a particular number of encodings similar to the Fourier-related encodings. As expected, all of the encodings show progressive improvement in the image quality as the number of encodings increases, because each encoding's cross-talk matrix approaches the identity matrix. Overall, the decimated images look better than the blended images because the aliasing artifacts are more easily

distinguished from the geology, whereas artifacts introduced by blending (lack of spatial resolution) more visibly degrade the image. Strictly speaking, the information content in the blended images is greater than that in the decimated images because I use more shot-records during blending. I do not see the additional information content though because I lowpass the data, effectively removing most of the information.

Additionally, I show the results for modulated-shot migration, Figures 7.36(c)-7.39(c), and plane-wave migration, Figures 7.40(c)-7.43(c) using the same number of blended experiments as the amplitud encodings. When the DCT, DST and DCS encodings are compared against modulated-shot migration, I see that the results are nearly identical for the DCT, DST, and DCS encodings. The Hartley encoding produces an image that has a substantial amount of cross-talk that contaminates the images which ultimately reduces its viability. However, when I compare the various amplitude and modulated-shot encodings against plane-wave migration I clearly see that plane-wave migration produces a better quality image at the same cost, even though plane-wave migration introduces cross-talk artifacts into the image. When I compare the images from plane-wave migration with the images for decimated SRM, I find that the artifacts are of similar nature, i.e. cross-talk from plane-wave migration is similar to noise caused by spatial aliasing.

To further understand the effect of blending I show the F-K spectra for the various encoding types at the same computational cost,Figures 4.1(a)-4.1(d), which illustrates how the encodings reconstruct the spatial information of the image. For example, Figure 4.1(a) shows the F-K spectrum for SRM which migrates each frequency for each shot in the decimated dataset. By decimating the dataset, I reduce the maximum wave-number that is adequately sampled, K_{nyq} and alias the higher wave-numbers in our data. Figure 4.1(b) and Figure 4.1(c) show the F-K spectrum for the DFT and DCS encodings respectively which only reconstruct the lowest wave-numbers first which effectively spatially low-pass filters the image. For comparison, I also show the F-K spectrum of plane-wave migration, Figure 4.1(d), which shows that plane-wave migration samples the wave-number domain

which introduces weakly coherent aliasing artifacts into the image.

4.17 Discussion

I show that a multitude of amplitude encoding schemes are viable by choosing different orthogonal bases for the encoding matrix. Some obvious choices are: Fourier-related bases, and wavelet bases. In any case, the choice of basis determines the image quality at a fixed cost (number of encodings) and the strength of the cross-talk in the blended image. All of the Fourier related encodings: DCT, DST, Hartley, DCS and modulated-shot migration (DFT), reconstruct the image by progressively adding encodings that capture additional spatial wave-numbers to the image. By using a limited number of encodings, the Fourier-related encodings act as spatial bandpass filters because they only reconstruct a limited range of wave-numbers, starting with the lowest wave-numbers first. As I show with the DCT, and DCS encodings the choice of parameters for the Fourier related encodings is paramount, in particular the periodization distance changes both the spatial sampling and the image quality substantially. Of the Fourier-based encodings, I find that the DFT, and DCS encodings are the best of the group, because they produce identical images with the same cost. It is important to note that the DFT and DCS encoding produce the same result, but are implemented differently: the DFT encoding with phase-shifts, and the DCS encoding with amplitude weights. The DCT, DST encodings are suboptimal because they are not shift-invariant encodings and they do not capture the same spatial information with the same efficiency as the DCS encoding. The Hartley encoding is a possible alternative to the DCS encoding, but it introduces strong cross-talk into the image which can be removed by using additional encodings. Because it requires additional encodings, there is no cost advantage for the Hartley encoding over the DCS encoding. Thus, I conclude that the DCS amplitude encoding is the most promising of the Fourier-related amplitude encodings.

The wavelet and other encodings produce good quality images, but at basically similar quality to the Fourier-related amplitude encodings. However, wavelet encodings are not

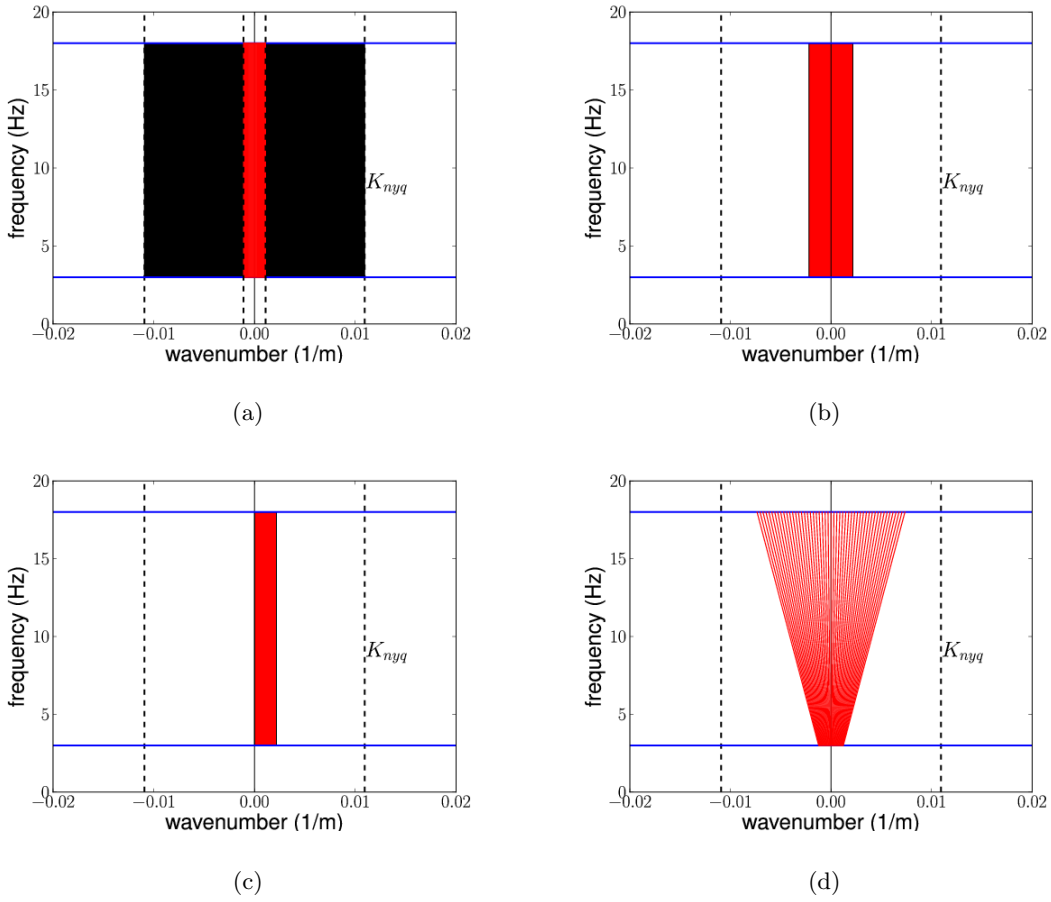


Figure 4.1. Example plots of the F-K spectra for various encoding types when only 50 encodings are used (or only 50 shots in the case of SRM). The F-K spectrum for SRM (a) which includes the entire area up to K^*_{nyq} which is less than K_{nyq} for the original dataset because of decimation. The areas in black indicate the aliased components of the data that are still migrated in the decimated dataset for SRM. For the DFT encoding (b), both positive and negative K values are migrated up to a limited point while aliased wave-numbers are not migrated. The F-K spectra for the DCS encoding (c) shows that only positive wave-numbers are migrated for this amplitude encoding, but because the DCS encoding needs two encodings to represent each wave-number, it only contains the same spatial information as the DFT encoding (b) does. Lastly, the F-K spectrum for plane-wave migration (d) is fan-shaped because of the frequency-dependent encoding matrix which samples K values thereby introducing aliasing like cross-talk artifacts into the image.

Table 4.1. Table comparing the computational cost of various shot-encoding schemes with the cost of conventional SRM for all 500 shot-records. The ratio C indicates the cost-advantage of blending over SRM. When C is greater than one, then blending is more efficient than SRM by that amount (e.g. $C = 2$ means the encoding is twice as efficient). C includes the effect of migrating over a larger aperture. For encoding schemes, such as Haar wavelet encoding, where the migration aperture varies as a function of the number of encodings used, I provide a range of best-case and worst-case scenarios in terms of C.

Shot-encoding scheme	DCT	Hartley	DCS	Haar	Hadamard	DFT	Plane-wave	SRM
Frequency dependent	N	N	N	N	N	N	Y	N
Migration aperture L_B	$2L_S$	$2L_S$	$2L_S$	Varies	$2L_S$	$2L_S$	$2L_S$	L_S
C - 25 experiments	10	10	10	-	-	10	10	20
C - 50 experiments	5	5	5	-	-	5	5	10
C - 100 experiments	2.5	2.5	2.5	-	-	2.5	2.5	5
C - 250 experiments	1	1	1	-	-	1	1	2
C - 32 experiments	-	-	-	16-8	8	-	-	-
C - 64 experiments	-	-	-	8-4	4	-	-	-
C - 128 experiments	-	-	-	4-2	2	-	-	-
C - 256 experiments	-	-	-	2-1	1	-	-	-

shift-invariant, which raises questions about the quality of the image based on the starting point for the encoding (or the shift). Additionally, wavelet encodings do not seem to offer a strong cost-advantage over Fourier-based encodings, thus I find that while scientifically interesting they are not compelling for production use as amplitude encodings. Of the other bases that I studied, the Hadamard basis is the most promising, because it fully captures the spatial information of the image quickly, but has strong cross-talk in the image. The Walsh basis is too similar to Fourier-related encodings to stand out, in terms of its cost-to-image quality.

Another interesting result is that all encodings that produce banded diagonals in the cross-talk matrix appear to introduce amplitude distortions into the final image. For example, consider the results from Hadamard encoding which does not band the diagonal, Figures 7.52(c)-7.55(c), with those from Haar encoding, Figures 7.44(c)-7.47(c), which bands the diagonal. The images produced by Hadamard encoding more closely represent the image from conventional SRM, and the difference images only show the cross-talk artifacts. By comparison, the images from Haar encoding (and most other encoding schemes) apparently have amplitude distortions present throughout most of the image, because the difference images show most of the image. The amplitude distortion is caused by adding together nearby shots that have similar information, which does not happen in conventional SRM. Overall, the amplitude distortion is minor (on an absolute scale), but is still important to note.

Interestingly, shot-record decimation, even with aliasing artifacts, tends to produce better images than blending does. This is probably because aliasing artifacts are weakly coherent across experiments, which means that they often do not stack together where as artifacts introduced by blending tend to be strongly coherent across blended images. This possibly explains why plane-wave migration appears to produce good quality images at only 50 experiments, where no other encoding is able to produce a similar quality image. From Figure 4.1(d), I see that plane-wave migration spatially aliases the data by sampling

wave-numbers thus introducing artifacts into the image. Based on the results of plane-wave migration and decimated SRM, I surmise that it is better to be spatially aliased than it is to use a lowpass filter by blending, because spatial aliasing artifacts are not necessarily prominent in the final image.

However, both plane-wave migration and decimated SRM are highly frequency dependent, as higher frequencies require larger wave-numbers (i.e. must be more finely sampled) in order to reconstruct all the information in the images. Thus, these conclusions may not be valid when higher frequency data is used, and plane-wave migration and decimated SRM may no longer be preferable to the other encodings presented here, because the other encodings are guaranteed to produce an image of a certain quality (i.e. an image with only low wave-numbers). Conversely, for plane-wave migration and decimated SRM, the image quality is unknown apriori, and depends on the frequency and wave-number content of our data. In the worst case scenario, the aliasing artifacts introduced by plane-wave migration and decimated SRM may be much worse than the degradation caused by other encodings.

The final decision of whether or not blended imaging offers any advantage is based on computational cost which I compute following the guidelines in Section 2.8. Table 4.1 details the computational cost comparison for the various encodings and decimated shot-record migration. Overall, if I just compare the number of experiments (encodings or shots) migrated, I find that both blending and decimated SRM produce good quality images at around 100 experiments. If I compare the same number of encodings to the decimated shot-record migration counter-parts, I find that the image quality of the blended images is usually better.

However, if I include the additional computational effort to migrate a blended shot-record over approximately twice the aperture required for a conventional shot-record (e.g. compare blended images using 25 encodings to the decimated SRM images using 50 shots), then I find that the quality of images produced by decimated shot-record migration is on par with those produced by blending at the same computational cost. Therefore, the cost

advantage of blending using these encoding schemes is greatly reduced, or non-existent, which means that the main justification for blended imaging (i.e. cost advantage) may not be completely valid. There are two important caveats to the last statement: 1 - decimated SRM is only preferable if I am willing to accept aliasing artifacts into the image, and 2 - the cost argument is only valid provided that migrating over twice the aperture increases the computational cost by twice as much, which is not necessarily true. I note that wavelet encodings do not incur as great of a cost penalty, because fewer shots are required for most wavelet encodings.

Chapter 5

Plane-wave equivalent amplitude encoding

As Chapter 4 shows plane-wave migration is able to produce good quality images at the lowest possible cost, because it reconstructs all of the spatial information in the image, but permits low-amplitude aliasing artifacts into the image. However, plane-wave migration has a significant downside: it is implemented using time-delays which substantially increase the cost of imaging in the time-domain. For example, the images constructed using plane-wave migration, Figures 7.40(c)-7.43(c), require a maximum time delay of approximately nine seconds. By comparison, the shot-record length for Sigsbee is 12 seconds, meaning that the total time for plane-wave shot-records is 21 seconds (i.e. 75% more). By increasing the time-length of the shot-records, I considerably increase the computational cost of reverse-time migration because the cost of reverse-time migration is proportional to the number of time-steps. Additionally, if I include the wider migration aperture effects into the cost comparison then plane-wave migration quickly becomes more costly than conventional SRM. Therefore, I want to find an amplitude encoding that replicates the results of plane-wave migration, but does so without using time-delays which would make the amplitude encoding more cost effective when using reverse-time migration.

5.1 Theory

As I showed in Section 4.5, the cross-talk matrix for plane-wave migration can be made equivalent to the cross-talk matrix for the DCS encoding (or vice-versa). The main difference

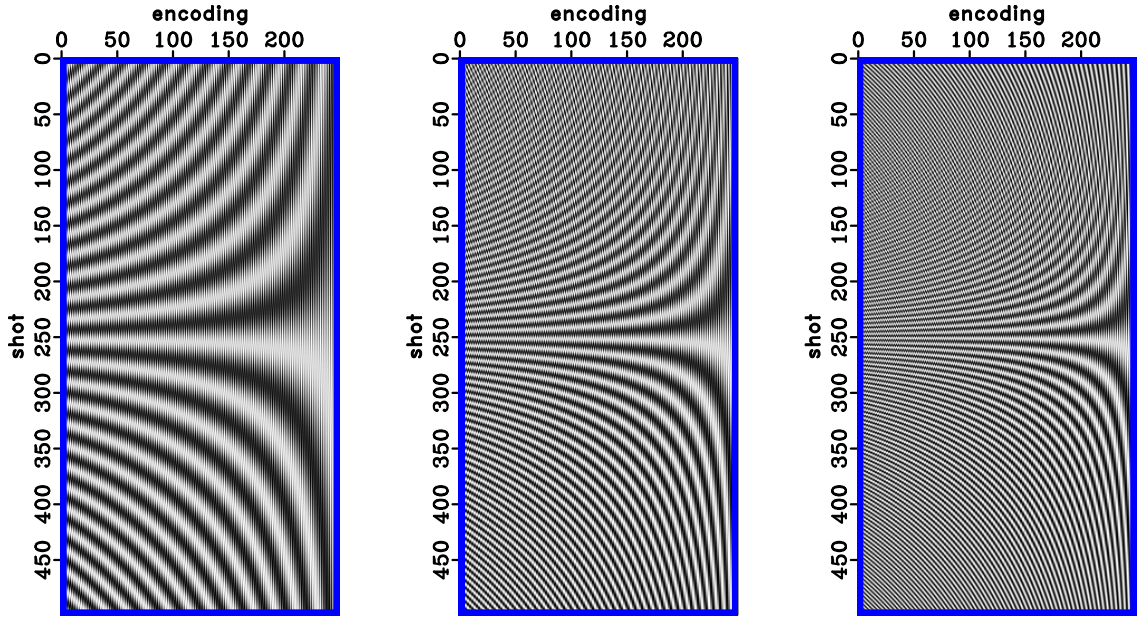
between the two cross-talk matrices, equation 4.24 and equation 4.13, is that the DCS cross-talk matrix is not necessarily frequency dependent. However, any amplitude encoding may be converted into a frequency-dependent amplitude encoding by simply changing the encoding scheme to be a function of frequency. To create a plane-wave equivalent amplitude encoding (PWEAM), I modify the DCS amplitude encoding to have the same arguments as equation 4.13 as a function of frequency, as follows (demonstrated for a single cosine amplitude encoding):

$$\mathbf{E}_m(\omega) = cos(\omega\tau_m). \tag{5.1}$$

Therefore, the DCS encoding produces the same cross-talk matrix as plane-wave migration and subsequently the same image as plane-wave migration but *without* encoding using time-delays. Instead, the encoding is done by frequency dependent amplitude weighting. By not directly using time-delays for encoding, it appears that I am able to save a significant amount of computational effort for reverse time migration.

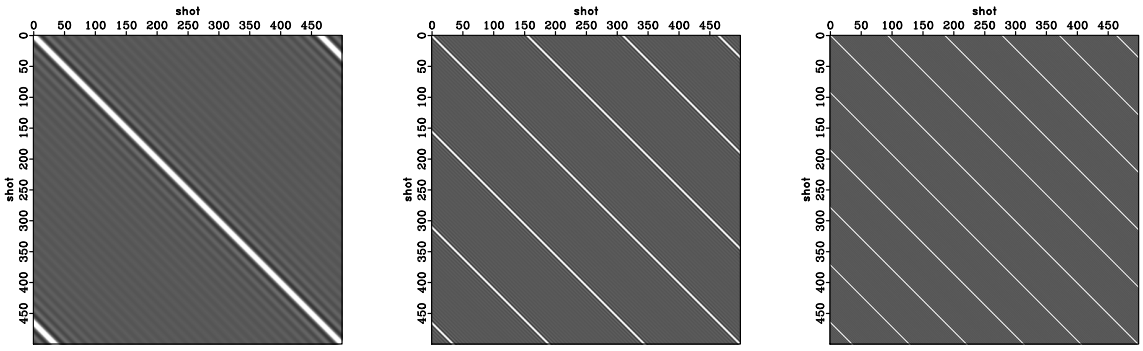
5.2 Results

To demonstrate the equivalence of the frequency-dependent DCS encoding and plane-wave migration, I construct the plane-wave equivalent amplitude encoding using the DCS encoding scheme which produces the frequency-dependent encoding matrices, Figures 5.1(a)-5.1(c). Figures 5.2(a)-5.2(c) show the cross-talk matrices for the plane-wave equivalent amplitude encoding which are the same as those for plane-wave migration. I migrate the plane-wave and DCS encoded datasets using downward continuation instead of reverse-time migration. Figures 7.40(c) -5.4(b) and Figures 7.42(c)-5.4(d) show the images comparing the results for plane-wave migration and the equivalent amplitude encoding. The images for the modified DCS encoding and plane-wave migration at the same cost are identical to within floating point accuracy, meaning that they produce the same result at the same computational cost. As previously mentioned, the cost for the amplitude encoding may be



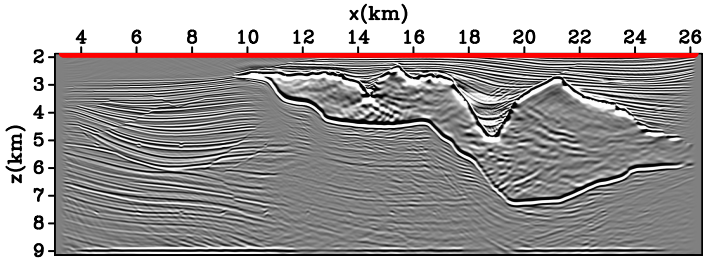
(a) (b) (c)

Figure 5.1. The encoding matrix for the plane-wave equivalent amplitude encoding for 250 encodings for: (a) 3 Hz, (b) 9 Hz and (c) 15 Hz. Notice that the frequency of the sinusoids increases as a function of frequency to mimic the behaviour of plane-wave migration.

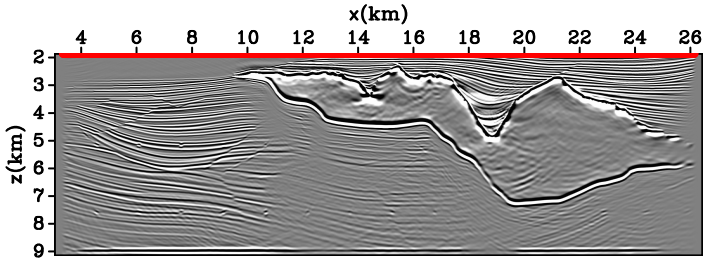


(a) (b) (c)

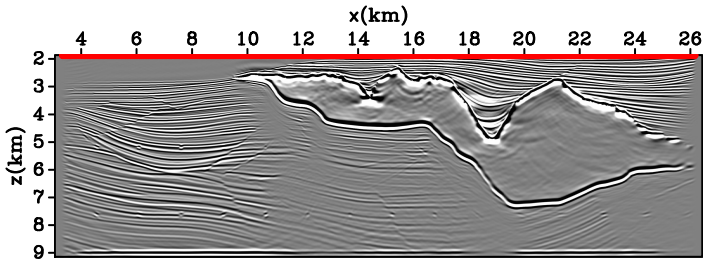
Figure 5.2. The cross-talk matrix for plane-wave migration and the equivalent amplitude encoding for 50 encodings for the following frequencies: (a) 3 Hz, (b) 9 Hz and (c) 15 Hz. Note the presence of more off-diagonal bands of cross-talk as the frequency increases.



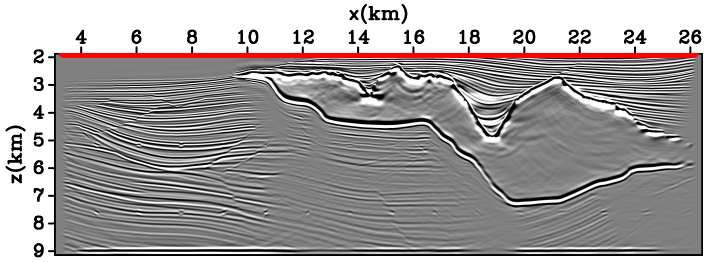
(a)



(b)

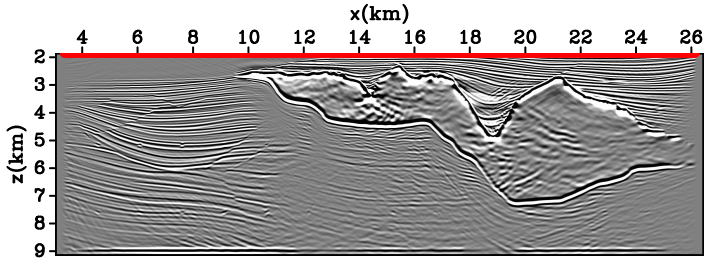


(c)

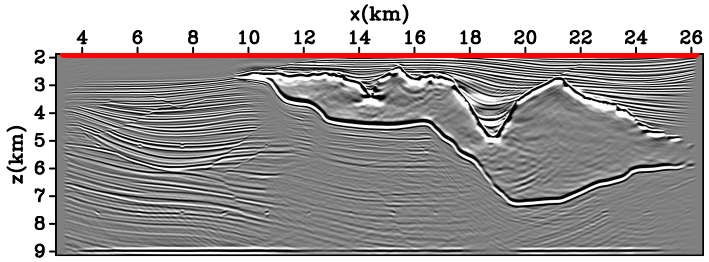


(d)

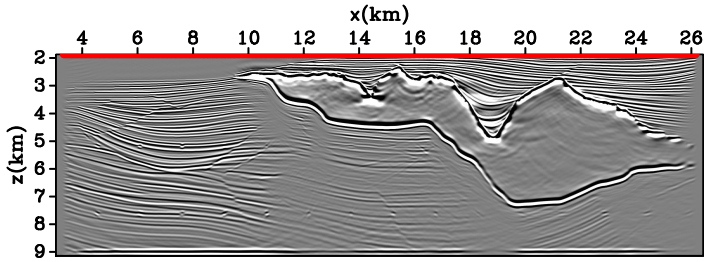
Figure 5.3. The stacked images for plane-wave migration using: (a) 25 encodings, (b) 50 encodings, (c) 100 encodings, and (d) 250 encodings. In all cases, the maximum take-off time-delay is 9 seconds.



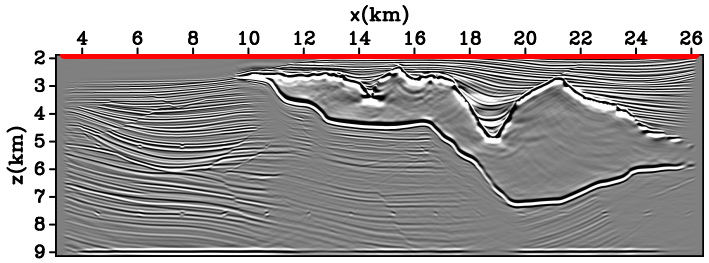
(a)



(b)



(c)



(d)

Figure 5.4. The stacked images for plane-wave equivalent amplitude encoding for: (a) 25 encodings, (b) 50 encodings, (c) 100 encodings, and (d) 250 encodings.

further reduced by changing the periodization distance so that fewer encodings are necessary to capture the same information.

5.3 Discussion

By modifying the DCS amplitude encoding to have the same argument to the sine and cosine function in the encoding matrix I am able to recreate the same crosstalk matrix as plane-wave migration but using amplitude weighting instead of time-delays. Consequently the images constructed from plane-wave migration and plane-wave equivalent amplitude modulation (PWEAM) are identical. In the frequency domain, the difference between the two encodings is insignificant, but in the time-domain the difference between the two encodings appears to be significant because plane-wave migration requires time-delays which introduce additional cost compared to PWEAM.

However, this simple understanding of the differences between the two encoding schemes is not correct, because PWEAM creates time-delayed data inadvertently. To illustrate why this is the case consider the effect of performing the encoding in the frequency domain, which by the convolution theorem is mathematically equivalent to simple multiplication of the encoding matrix with the shot-record, $S(\omega)$:

$$B_{m,n}(\omega) = cos(-j\omega\tau_{m,n})S(\omega), \tag{5.2}$$

where B represents the blended wavefield. By using Euler's formula for the cosine, equation 5.2 becomes:

$$B_{m,n}(\omega) = \frac{e^{-j\omega\tau_{m,n}} + e^{j\omega\tau_{m,n}}}{2} S(\omega). \tag{5.3}$$

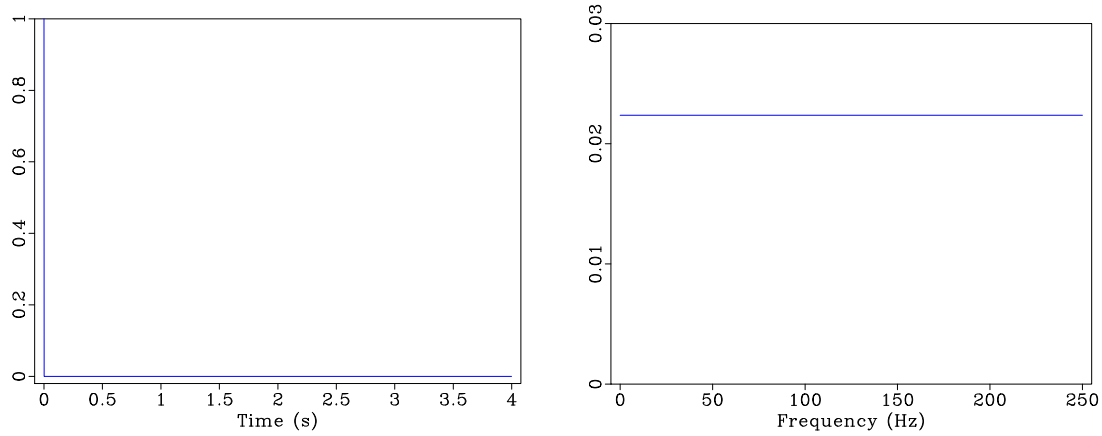
The above equation has additional physical meaning by using the shift theorem, which states that a time-delay is represented by the multiplication by a frequency dependent complex exponential in the frequency domain. Thus, equation 5.2 really represents the signal, $S(\omega)$, shifted by both $+\omega\tau_{m,n}$ and $-\omega\tau_{m,n}$ with half the amplitude distributed at each shift, which

in effect means that PWEAM does in fact create time-delayed data indirectly!

As a demonstration, consider the case of a shot-record (signal), $S(t)$ that consists of a single trace with a unit impulse at the zero sample in the time-domain, Figure 5.5(a), which will provide the impulse response of the PWEAM encoding scheme. The corresponding frequency spectrum, $S(\omega)$, is flat with a value of 1 everywhere, Figure 5.5(b). After the blending process takes place (or convolution) then the frequency spectrum of the blended shot-record for a single blended experiment looks something like Figure 5.5(c). Transforming back to the time-domain I find that the unit impulse has been delayed in time and divided into two separate impulses in Figure 5.5(d), which is quite a surprising result but expected from equation 5.2. Thus, while PWEAM does in fact produce the same result as plane-wave migration, it does so using an encoding scheme that is very different physically from plane-wave migration, but similar in the respect that they both require the use of time-delays, either implicitly or explicitly.

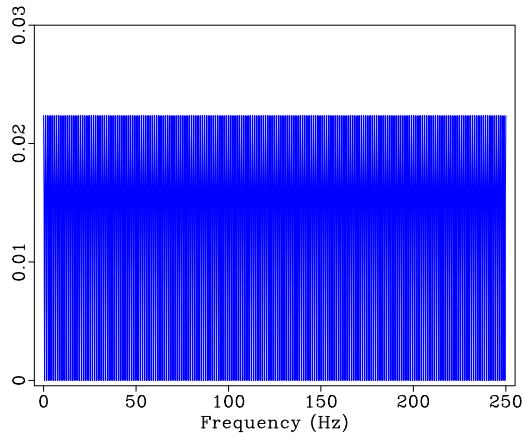
The overall conclusion from this experiment, is that while an equivalent amplitude encoding scheme can be produced for a time-delay encoding, the end result is more-or-less the same as using the direct time delay encoding in terms of the additional computational cost required in the time-domain. While PWEAM is interesting from a purely theoretical approach, it does not appear that it offers a significant advantage over plane-wave migration in terms of computational cost.

Further this result leads me to hypothesize that most (if not all) time-delay and phase-shift encodings can be converted into the equivalent amplitude encoding. In cases where a time-delay encoding does not have an obvious conversion to an amplitude encoding, one may wish to construct the cross-talk matrix for the time-delay encoding and then use SVD on the cross-talk matrix, Chapter 3, to construct the equivalent amplitude encoding matrix without the need for a rigorous mathematical formulation. I note that there is potentially a major issue with time-domain blended imaging: the reverse-time migration operator is not guaranteed to be linear.

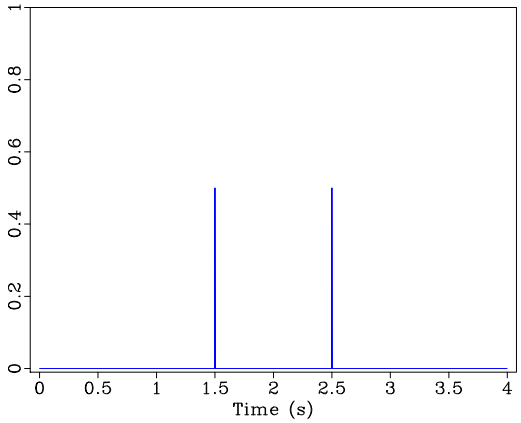


(a)

(b)



(c)



(d)

Figure 5.5. The unit impulse (a), (b) the frequency spectrum of (a), (c) the encoded frequency spectrum of the unit impulse and (d) the encoded unit impulse constructed using PWEAM.

Chapter 6

Frequency-dependent amplitude encoding by basis selection

As I noted in Section 2.2, the encoding matrix E may be frequency dependent as well which may reduce the number of encodings while maintaining image quality because I do not necessarily need all encodings at every frequency due to the presence of overly redundant information at low frequencies. Thus, I discard the overly redundant information at low frequencies by using prior knowledge of our velocity model to estimate the maximum wavenumber in the image which determines the amount of redundant information to discard at each frequency. In essence, this allows us to use blended imaging to produce an identical image to SRM at much lower cost, which I refer to as"fixed quality" encoding.

6.1 Frequency-dependent spatial sampling theory

Previous research by Stork (1999) and Soubaras (2006) indicates that encoding schemes that combine nearby shots are most effective at reducing the impact on blending on the image because they act as spatial bandpass filters. To understand why this is the case, consider blending a few shots that are nearby one another spatially. In most situations, the geology of the Earth changes slowly (relative to our shot sampling), so shots in the immediate vicinity of one another will contain mostly redundant information, which means that our data is quite frequently oversampled in the shot-domain to some extent. If I simply add the nearby shots together with equal weights then I reinforce events that appear

similar in the different shot-gathers, and smear (or destroy) events that appear differently in the shot-records. This process emphasizes features in the shot-records that have very low wave-numbers (almost no dip), or effectively spatially bandpasses our shot-gathers and adds them together to make a blended shot-record. However, if I use a different weighting scheme for the individual shot-records (e.g. Gaussian weighting or exponential weighting) then I can emphasize features with certain wave-numbers during blending. In order to capture information with the highest wave-numbers in our data I must eventually reduce the number of shot-records that I combine together (by changing the width of our bandpass filter).

Since the cross-talk matrix, $\mathbf{C}$, fully describes the result of encoding, I can perform the same spatial bandpassing operation in our cross-talk matrix by simply choosing an encoding scheme that produces a C matrix that is banded on the diagonal. The width of the band on the diagonal determines the number of nearby shots to add together, and the weights for each shot are given in the cross-talk matrix. By doing so, I could construct an identical spatial bandpass filter to apply directly to our shot-gathers, but blending produces the same effect during imaging if I choose the correct encoding scheme.

To blend correctly (i.e. without degrading the quality of the image), I need to establish the maximum spatial wave-number of interest in our image at any point in the image. Using the same argument that Stork (1999) makes, I estimate the maximum wave-number in our image, K_{max}, by first computing the corresponding ρ-value using the expression:

$$\rho_{max} = \frac{sin(\theta_{max})}{v}, \tag{6.1}$$

where θ_{max} is the maximum dip of interest (in degrees or radians) and v is the velocity in the immediate vicinity of the dip of interest. I convert ρ_{max} to the corresponding wave-number at a particular frequency using:

$$\rho\omega = K. \tag{6.2}$$

Equation 6.2 is frequency-dependent, meaning that the maximum K needed for each frequency is different. For comparison, the maximum wave-number sampled by the Nyquist criterion is given by:

$$K_{nyq} = \frac{1}{2\Delta x}, \tag{6.3}$$

where Δx is the shot sampling interval. When $K_{max} < K_{nyq}$ at a given ω, then our dataset is oversampled in the shot-domain, which means that I may want to decimate our shots to reduce the overall number of migrations necessary for imaging. Usually I do not decimate our data, for two reasons:

- the redundant shots improve the fold, or signal-to-noise ratio of the resulting image,
- often optimal decimation requires shot-interpolation.

However, blending allows us to get around the issue of shot-interpolation because blending constructs a low-pass filter over shots which captures all the wave-numbers up to the maximum wave-number of interest from equation 6.2. To demonstrate this, first consider the spatial sampling interval, ΔK, is a function of the periodization distance $\wp$ and given by,

$$\Delta K = \frac{1}{\wp \Delta x}. \tag{6.4}$$

In order to assure proper spatial sampling, I must use each K up to the maximum wave-number that I am interested in, which means that the total number of encodings is frequency dependent and given by,

$$N_e(\omega) = \frac{\min(K_{nyq}, p_{max}\omega)}{\Delta K}. \tag{6.5}$$

The major advantage to blended imaging occurs when fewer wave-numbers are sampled at a given frequency than are done in shot-record migration, equation 6.5, because shot-record migration migrates every wave-number for every frequency for every shot. Thus, the computational advantage of blending over shot-record migration is frequency and shot-sampling dependent.

6.2 Results

I demonstrate the ability of the DFT, DCS and plane-wave migration encodings to produce the same image at different costs by imaging the Sigsbee2A dataset using all 500 shots. I estimate that the maximum dip of interest in the Sigsbee dataset is around 85 degrees at a depth of a few kilometers (near the top of the salt). In this area, the velocity is approximately $2200m/s$ which means that $\rho_{max} = 4e^{-4}$. Using equation 6.5, I compute the number of encodings necessary for both the DFT (modulated-shot migration) and DCS encodings as a function of frequency. For the DFT encoding, the total number of encodings is equal to double the number of encodings because the DFT encoding needs negative K-values to remove cross-talk. Likewise, the DCS encoding requires two encodings for each wave-number. I periodize both the DFT and DCS encodings over 250 shots, because the maximum migration aperture is less than one-half of the total survey (Soubaras, 2006). For plane-wave migration, the number of encodings is equal to the maximum N_e for ω_{max} because plane-wave migration images each frequency for each plane-wave. Figures 6.1(a) -6.1(c) show the result of the separate migrations. The images show some subtle differences between DCS and DFT encodings, but the largest differences are between them and plane-wave migration because the plane-wave migration image appears to be lower frequency, due to the oversampling of low wave-numbers at low frequencies. For comparison, Figure 7.1 shows the result of conventional shot-record migration using all 500 shots. The encodings produce nearly identical images to one another and shot-record migration, but at different costs. Figure 6.2 shows a relative cost comparison of the number of blended experiments necessary for each frequency for the different encoding types, with fewer experiments being better. Clearly, the DFT and DCS encodings have a significant cost-advantage over plane-wave migration and SRM in terms of the number of experiments.

6.3 Discussion

Figures 6.3(a)-6.3(d) show the F-K spectra of the various encoding schemes, which demonstrates that DFT, DCS, and plane-wave encodings all recover the spatial information needed to reconstruct the full image based on the maximum wave-number, K, that I selected. As Figure 6.3(a) shows, I cannot decimate the shot-records for Sigsbee without shot interpolation, because I would alias the highest frequencies and wave-numbers. However, if I decimate the data by a factor of 2 then I only alias the highest wave-numbers at the highest frequencies which results in little corruption of the image as Figure 7.61(d) shows. Even though this image looks identical to the image from undecimated SRM, there are some artifacts caused by the truncation of the highest wave-numbers, but they are hard to distinguish. Nonetheless, I do not consider the decimated image to be equivalent to the SRM image because of the inherent aliasing; although an interpreter might consider the decimated image to be "good enough."

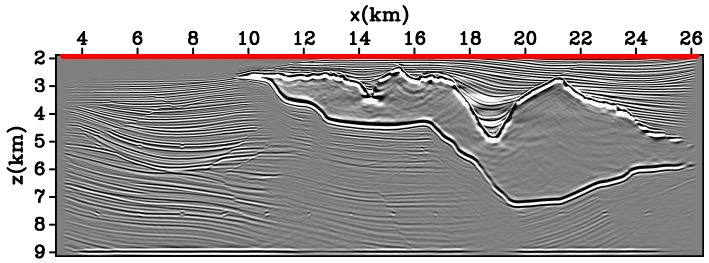
Because the quality between the encodings and SRM is equivalent, I consider the cost of the encodings to be the deciding factor as to whether or not one should use blended imaging. For reference, Section 2.8 describes how I compare the cost of the encoding schemes. Because fixed quality encoding schemes are frequency-dependent for the DCS and DFT encodings, I must use equation 2.20 for a straight comparison of the computational cost. SRM and plane-wave migration use equation 2.18 because their cost is not frequency dependent. For Sigsbee, I estimate that that the blended migration aperture, L_B, is approximately twice the individual shot-record migration aperture, L_S (Stork, 1999; Soubaras, 2006). Table 6.1 shows the results of the cost comparison. In this case, both the DCS and DFT (modulated-shot) encoding require less computational effort than shot-record migration, while plane-wave migration is more expensive than shot-record migration due to its wider migration aperture. Therefore, I conclude that blended migration is preferable to shot-record migration for the Sigsbee dataset when the quality of the image is fixed, because blended imaging is able to produce the same image at less cost than SRM. However, the

Table 6.1. Table comparing the computational cost of various fixed quality encodings.

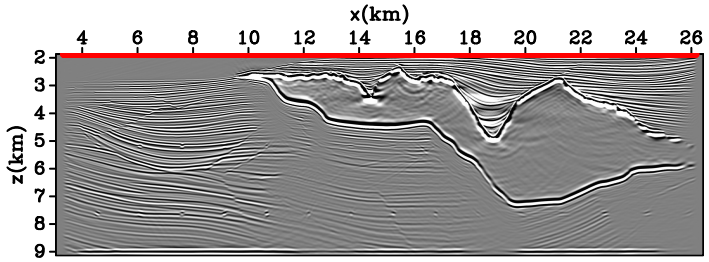
Type	Plane	DFT	DCS	SRM
Frequency dependent	N	Y	Y	N
Migration aperture L_B	$2L_S$	$2L_S$	$2L_S$	L_S
$N_\omega N_e$or $\sum_\omega N_e(\omega)$	80640	47020	47020	120000
C	0.744	1.271	1.271	1.000

story changes if I consider the decimated image, Figure 7.61(d), as being a "good enough" representation of the full image. In this case, the cost of blended imaging for all encodings exceeds the cost of decimated imaging, because the cost of decimated imaging is 1/2 the cost of the full SRM.

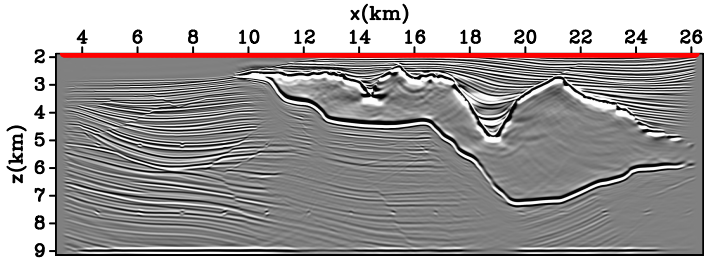
However, these results are dependent on the frequency content of the data that is used for imaging, as higher frequency data requires finer sampling in the F-K domain to ensure no aliasing occurs during imaging. This requirement effectively means that given the quality of data collected it is possible that all encodings may be required in order to produce the same image as SRM, which would completely remove the cost advantage that blending offers.



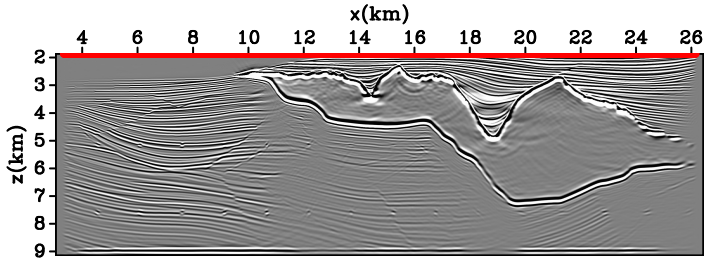
(a)



(b)



(c)



(d)

Figure 6.1. The image produced using (a) the DCS amplitude encoding, (b) DFT phase-shift encoding (modulated-shot migration), (c) plane-wave migration and (d) conventional SRM. In all cases, the encodings produce images that are almost identical to the image for SRM, which indicates the mathematical equivalence of encoding with SRM, and the presence of oversampling in the shot-domain.

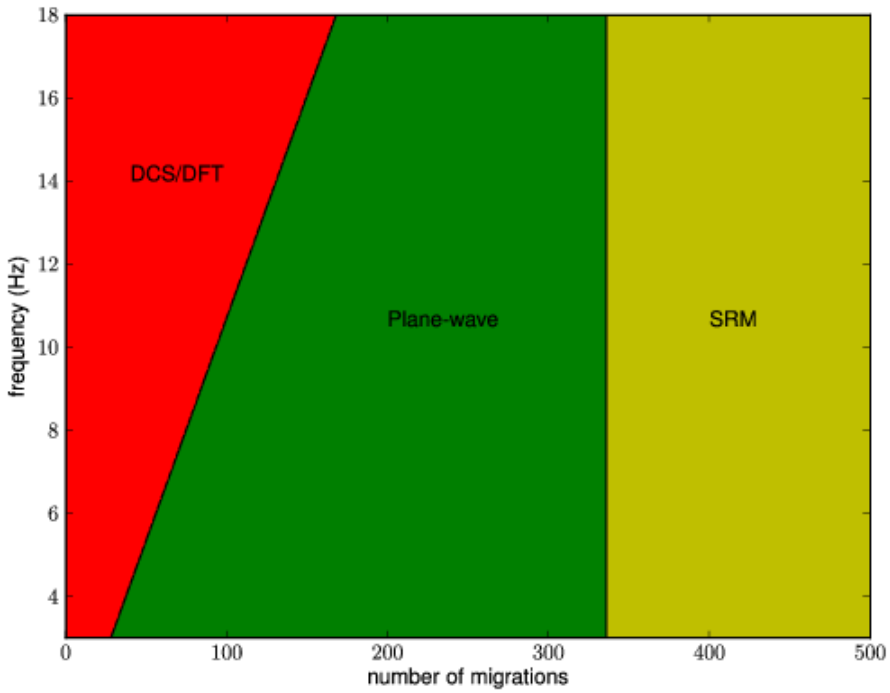


Figure 6.2. A plot comparing the number of encodings necessary for the DCS, DFT, plane-wave and SRM as a function of frequency for a fixed quality encoding (meaning the same quality image amongst all the encodings). Clearly, the DCS and DFT encodings require many fewer encodings than plane-wave migration does. This is because plane-wave migration is implemented using time-delays which require each shot to be migrated at each frequency for every encoding. SRM is the most costly in this diagram, because each frequency is migrated for each shot-record. Overall, this diagram does *not* fully illustrate the cost argument because I do not take into account the effect of migration aperture on the computational cost.

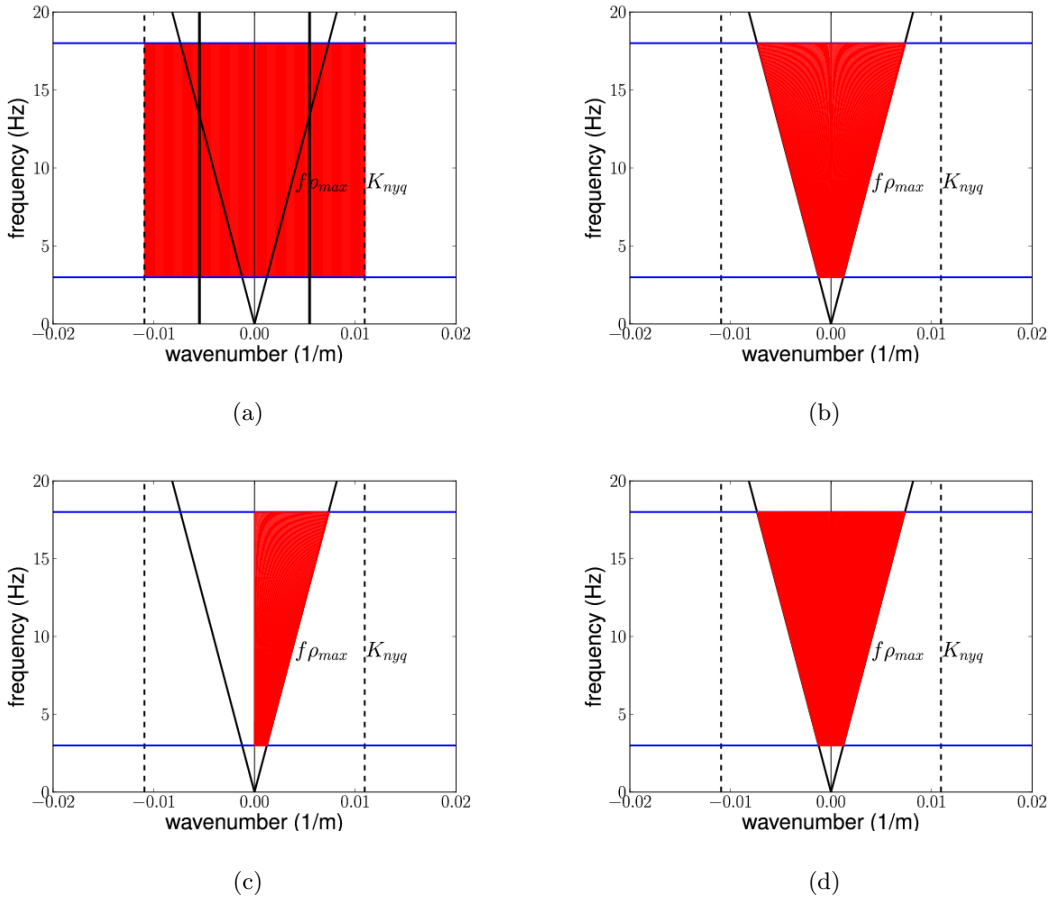


Figure 6.3. Example plots of the F-K spectra for various encoding types when used with optimal values for image reconstruction (i.e. fixed quality encodings). The F-K spectrum for SRM (a) which includes the entire area up to K_{nyq} which indicates that the data is oversampled in the shot-domain. If I were to decimate the data by keeping every other shot-record, I would change K_{nyq} to be the solid black line in (a), which means that I would alias the highest wave-numbers at the highest frequencies. For comparison, (b) shows the F-K spectrum for modulated shot-migration, which includes positive and negative K values. The F-K spectra for the DCS encoding shows that only positive values of K are necessary, and (d) shows the F-K spectra for plane-wave migration which is virtually indistinguishable from (b). All of the encodings only sample the wave-number to the maximum wave-number necessary for that frequency as dictated by $f\rho_{max}$.

Chapter 7

Conclusions

I present a theory of blended imaging that unifies all previous shot-encoding schemes into the same framework. From this framework, I conclude that the process of blending is fundamentally the process of projecting seismic wavefields into another basis (i.e. the quality of the image from an encoding is determined by the basis that I choose for projection). By using that knowledge, I try two approaches to designing encoding schemes using amplitude weights instead of phase-shifts or time-delays: cross-talk factorization and basis selection. I find that basis selection is more productive than cross-talk matrix factorization because of inherent design issues with cross-talk matrix factorization (i.e. uncertainty in choosing the crosstalk matrix for decomposition).

Using basis selection, I design a series of previously unknown amplitude encoding schemes including: Fourier related encodings, wavelet encodings, and other basis encodings. For the amplitude encodings that I explore, the cost-to-image quality is not sufficient to justify the use of blending in most scenarios, especially when compared against plane-wave migration and decimated shot-record migration. However, this conclusion is only true when using low-frequency data. When higher frequency data is available, then the aliasing artifacts that plane-wave migration and decimated SRM introduce may degrade the quality of the image less than spatial bandpassing that other encoding schemes introduce.

Furthermore, I examined an alternative encoding scheme (PWEAM) to plane-wave migration that that produces the same image but using frequency-dependent amplitude

weights instead of time-delays directly. However, I find that the frequency-dependent amplitude weighting scheme effectively creates time-delays in the data which effectively negates the desired advantage of PWEAM over plane-wave migration.

Additionally, I show that blended imaging can be used to reduce the oversampling of our data in the shot-domain using frequency-dependent sampling theory. By doing so, I am able to produce the same image as that from SRM for a series of amplitude and other encodings, but at less cost than SRM. Since most datasets are somewhat oversampled in the shot-domain, this result indicates that blending may be able to save some computational cost relative to full SRM. However, I note that the cost advantage over SRM in even the best case was only 27% for the Sigsbee dataset, using the data available. This result depends on the frequency content of the data. As the frequency content of modern seismic surveys increases, the capability of blending to reduce the available information in the dataset is greatly reduced due to fundamental sampling theory requirements.

Overall, I think that the prospects for blended imaging in the seismic imaging industry are limited, because the initial cost advantage of blending (i.e. reduction in the number of needed migrations) is mostly offset by the cost of migrating over a larger aperture, and the subsequent degradation of the final image. The cost penalty may be somewhat avoided by using an encoding scheme (e.g. wavelet encoding) that does not blend all shots together in each experiment. Consequently, decimated SRM is almost *always* preferable to blended imaging in terms of the quality of the image produced at the overall computational cost. At this time, I do not believe that an encoding scheme exists that is able to produce the best quality image at a significant enough cost advantage to make blended imaging worthwhile. This conclusion is largely drawn from the results of the amplitude encoding schemes that I examine, and additional experiments that I do not show in this work. Moreover, I do not think that any encoding scheme exists that reconstructs the same image as SRM but at far less computational cost, due to mathematical reasons. In particular, such an encoding scheme would be a unitary, rectangular matrix which does not exist as far as I know. Thus,

the prospect of finding an *optimal* blended imaging scheme is quite limited. Therefore, I do not think that blending will become a standard imaging procedure for producing final stacked images.

Recently, there has been a resurgence of interest in blending to reduce the cost of iterative inversion schemes such as: full-waveform inversion, least-squares imaging, and migration velocity analysis (Ayeni *et al.* , 2009; Herrmann *et al.* , 2009; Krebs *et al.* , 2009; Verschuur & Berkhout, 2009; Symes, 2010). In these iterative schemes, the degradation of the image may be considered acceptable relative to the cost advantage of blending, especially considering that the cross-talk can be changed from iteration to iteration. Changing the cross-talk by iteration greatly reduces the impact of the cross-talk on the inversion process, but it remains to be seen if blending has other consequences (e.g. slows down the convergence rate).

References

Akerberg, Peeter, Hampson, Gary, Rickett, James, Martin, Harry, & Cole, Jeff. 2008. Simultaneous source separation by sparse Radon transform. *SEG Technical Program Expanded Abstracts*, **27**(1), 2801.

Ayeni, Gboyega, Tang, Yaxun, & Biondi, Bindo. 2009. Joint preconditioned least-squares inversion of simultaneous source time-lapse seismic data sets. *Pages 3914–3918 of: SEG Technical Program Expanded Abstracts*.

Beasley, Craig J. 2008. A new look at marine simultaneous sources. *The Leading Edge*, **27**(7), 914.

Berkhout, A. J. (Guus), Blacquie?re, Gerrit, & Verschuur, Eric. 2008. From simultaneous shooting to blended acquisition. *SEG Technical Program Expanded Abstracts*, **27**(1), 2831.

Berkhout, A.J., Verschuur, D.J., & Blacquie?re, G. 2009. Seismic imaging with incoherent wavefields. *Pages 2894–2898 of: SEG Technical Program Expanded Abstracts*.

Blacquiere, Gerrit, Berkhout, Guus, & Verschuur, Eric. 2009. Survey Design for Blended Acquisition. *Pages 56–60 of: SEG Technical Program Expanded Abstracts*.

Duquet, Bertrand. 1999. 3D Plane wave migration of streamer data. *SEG Technical Program Expanded Abstracts*, **20**(1), 1033.

Eckart, Carl, & Young, Gale. 1936. The approximation of one matrix by another of lower rank. *Psychometrika*, **1**(3), 211–218.

Hampson, Gary, Stefani, Joe, & Herkenhoff, Fred. 2008. Acquisition using simultaneous sources. *The Leading Edge*, **27**(7), 918.

Herrmann, Felix J., Erlangga, Yogi A., & Lin, Tim T. Y. 2009. Compressive simultaneous full-waveform simulation. *Geophysics*, **74**(4), A35.

Howard, Mike. 2007. Marine seismic surveys with enhanced azimuth coverage: Lessons in survey design and acquisition. *The Leading Edge*, **26**(4), 480.

Huo, Shoudong, Luo, Yi, & Kelamis, Panos. 2009. Simultaneous sources separation via multi-directional vector-median filter. *Pages 31–35 of: SEG Technical Program Expanded Abstracts*.

Kapoor, S. (Jerry), O'Brian, Michael, Desta, Dawit, Atakishiyev, Ilgar, & Tomida, Michiru. 2007. Subsalt imaging — The RAZ/WAZ experience. *SEG Technical Program Expanded Abstracts*, **26**(1), 926.

Kim, Young, Gruzinov, Irina, Guo, Manhong, & Sen, Satyakee. 2009. Source separation of simultaneous source OBC data. *Pages 51–55 of: SEG Technical Program Expanded Abstracts.*

Krebs, Jerome R., Anderson, John E., Hinkley, David, Neelamani, Ramesh, Lee, Sunwoong, Baumstein, Anatoly, & Lacasse, Martin-Daniel. 2009. Fast full-wavefield seismic inversion using encoded sources. *Geophysics*, **74**(6), WCC177.

Liu, Faqi. 1999. Plane wave source composition: An accurate phase encoding scheme for prestack migration. *SEG Technical Program Expanded Abstracts*, **21**(1), 1156.

Liu, Faqi, Hanson, Douglas W., Whitmore, Norman D., Day, Richard S., & Stolt, Robert H. 2006. Toward a unified analysis for source plane-wave migration. *Geophysics*, **71**(4), S129.

Michell, Scott, Shoshitaishvili, Elena, Chergotis, Dean, Sharp, John, & Etgen, John. 2006. Wide azimuth streamer imaging of Mad Dog; Have we solved the Subsalt imaging problem? *SEG Technical Program Expanded Abstracts*, **25**(1), 2905.

Moore, Ian, Dragoset, Bill, Ommundsen, Tor, Wilson, David, Ward, Camille, & Eke, Daniel. 2008. Simultaneous source separation using dithered sources. *SEG Technical Program Expanded Abstracts*, **27**(1), 2806.

Morton, Scott A. 1999. Fastershot-record depth migrations using phase encoding. *SEG Technical Program Expanded Abstracts*, **17**(1), 1131.

Oppenheim, Alan. 1999. *Discrete-time signal processing.* 2nd edn. Upper Saddle River N.J.: Prentice Hall.

Perrone, Francesco, & Sava, Paul. 2009. Comparison of shot encoding functions for reverse-time migration. *Pages 2980–2984 of: SEG Technical Program Expanded Abstracts.*

Rabbani, Majid. 1991. *Digital image compression techniques.* Bellingham Wash. USA: Spie Optical Engineering Press.

Rietveld, W. E. A. 1994. Prestack depth migration by means of controlled illumination. *Geophysics*, **59**(5), 801.

Romero, Louis A., Ghiglia, Dennis C., Ober, Curtis C., & Morton, Scott A. 2000. Phase encoding of shot records in prestack migration. *Geophysics*, **65**(2), 426.

Salomon, D. 2007. *Data compression : the complete reference.* 4th ed. edn. London: Springer.

Shan, Guojian, & Biondi, Biondo. 2008. Plane-wave migration in tilted coordinates. *Geophysics*, **73**(5), S185.

Shoshitaishvili, Elena, Lee, Kyoung-Jin, Michell, Scott, & Etgen, John. 2006. Fast wavefield migration for wide azimuth data.

Sklar, Bernard. 2001. *Digital communications : fundamentals and applications*. 2nd ed. edn. Upper Saddle River N.J.: Prentice-Hall PTR.

Soubaras, Robert. 2006. Modulated-shot migration. *SEG Technical Program Expanded Abstracts*, **25**(1), 2430.

Spitz, Simon, Hampson, Gary, & Pica, Antonio. 2008. Simultaneous source separation: A prediction-subtraction approach. *SEG Technical Program Expanded Abstracts*, **27**(1), 2811.

Stoffa, Paul L., Sen, Mrinal K., Seifoullaev, Roustam K., Pestana, Reynam C., & Fokkema, Jacob T. 2006. Plane-wave depth migration. *Geophysics*, **71**(6), S261.

Stork, Christof. 1999. How many P values do you want to migrate for delayed shot wave equation migration? *SEG Technical Program Expanded Abstracts*, **23**(1), 1041.

Strang, Gilbert. 1997. *Wavelets and filter banks*. Rev. edn. Wellesley MA: Wellesley-Cambridge Press.

Sun, Peiyong. 1999. Prestack migration of areal shot records with phase encoding. *SEG Technical Program Expanded Abstracts*, **21**(1), 1172.

Symes, William. 2010. Source synthesis for waveform inversion. *SEG Expanded Abstracts*, **29**.

Temme, P. 1984. A comparison of common-midpoint, single-shot, and plane-wave depth migration. *Geophysics*, **49**(11), 1896.

Tieman, Hans J. 1997. Improving plane-wave decomposition and migration. *Geophysics*, **62**(1), 195.

Ting, Chu-Ong, & Zhao, Wei. 2009. A simulated wide azimuth simultaneous shooting experiment. *Pages 76–80 of: SEG Technical Program Expanded Abstracts*.

Tsitsas, N. L. 2010. On Block Matrices Associated with Discrete Trigonometric Transforms and Arising in Wave Propagation Theory. *Journal of Computational Mathematics*.

Verschuur, D.J., & Berkhout, A.J. 2009. Target-oriented, least-squares imaging of blended data. *Pages 2889–2893 of: SEG Technical Program Expanded Abstracts*.

Womack, J. E. 1990. Encoding techniques for multiple source point seismic data acquisition. *Geophysics*, **55**(10), 1389.

Zhang, Yu, Sun, James, Notfors, Carl, Gray, Samuel H., Chernis, Leon, & Young, Jerry. 2005. Delayed-shot 3D depth migration. *Geophysics*, **70**(5), E21.

Zhang, Yu, Sun, James, & Gray, Samuel. 2007. Reverse-time migration: Amplitude and implementation issues. *SEG Technical Program Expanded Abstracts*, **26**(1), 2145.

Appendix A

Comparison of various encodings

This appendix contains plots of the various encoding schemes used throughout this paper in the same format. In particular, each page shows:

- (a) the amplitude-weights from the encoding matrix (for phase-encodings the amplitude weights are all one),
- (b) the real-part of the cross-talk matrix for a single frequency (3.0 Hz),
- (c) the final stacked image for the encoding scheme, and
- (d) the amplified difference image between the final stacked image for that encoding and the image from conventional SRM. All difference images are amplified (approximately 10 times) and clipped to the same scale, meaning that one can directly compare the difference images to get an idea of how different two encodings are from one another. Due to the amplification, the differences may appear stronger than they actually are.

In all cases, the images from encoding are compared against Figure 7.1, which shows the image from conventional SRM for the Sigsbee2A dataset.

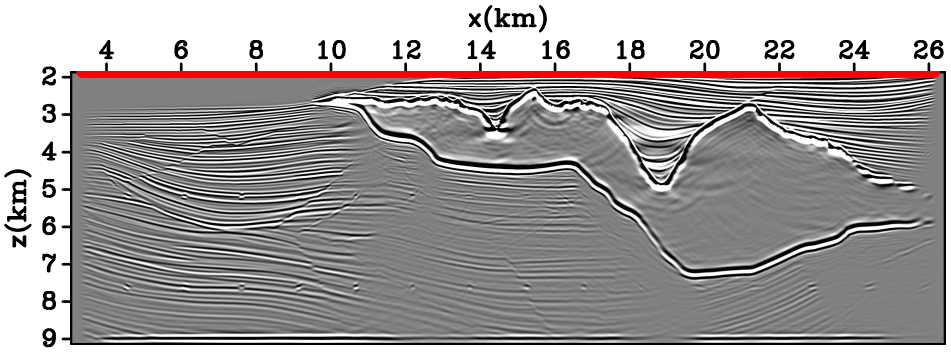
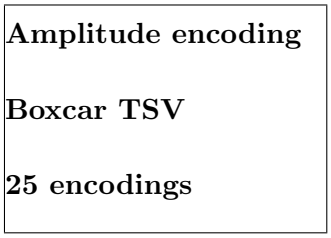
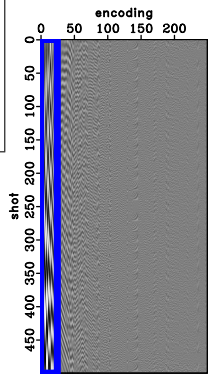
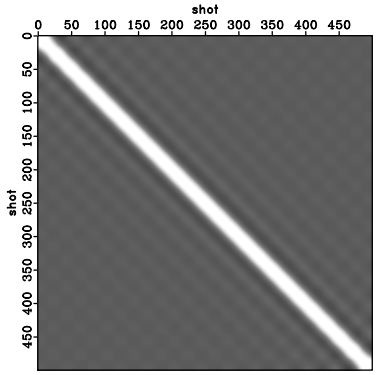


Figure 7.1. The target image, created by using SRM on all 500 shots from the original Sigsbee2A dataset. All encoded images are compared against this image.

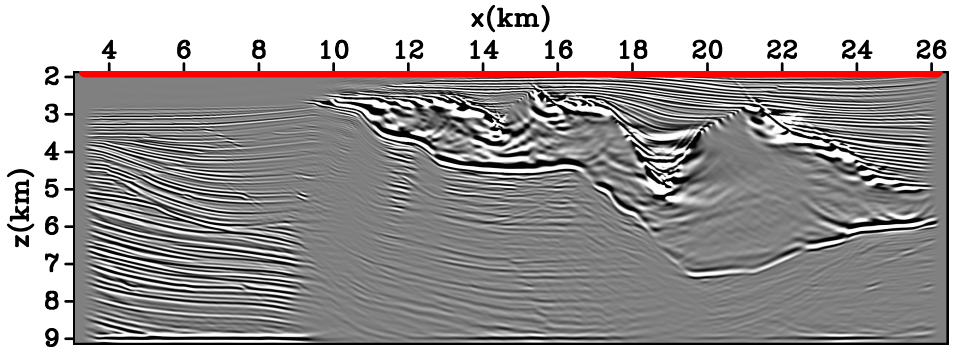




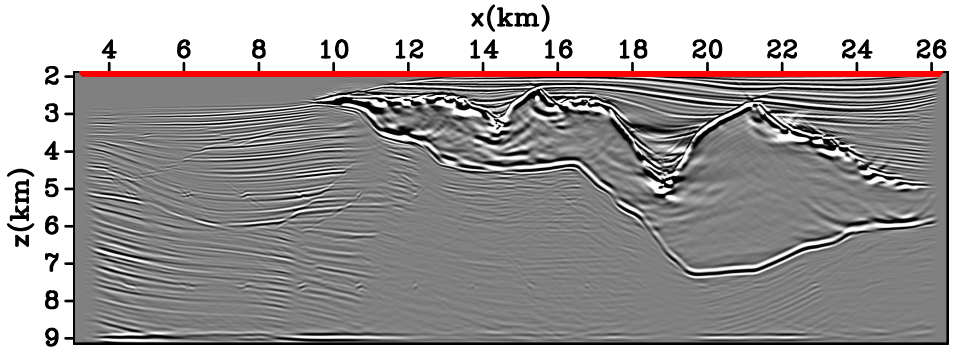
(a)



(b)

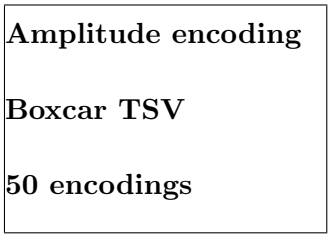


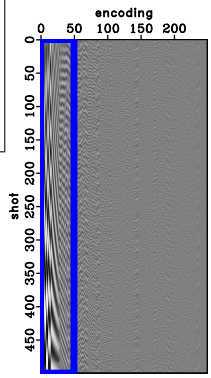
(c)



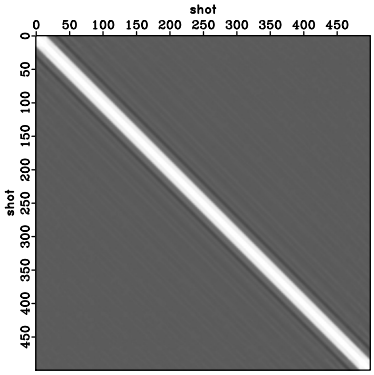
(d)

Figure 7.2. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Boxcar TSV encoding using 25 encodings.

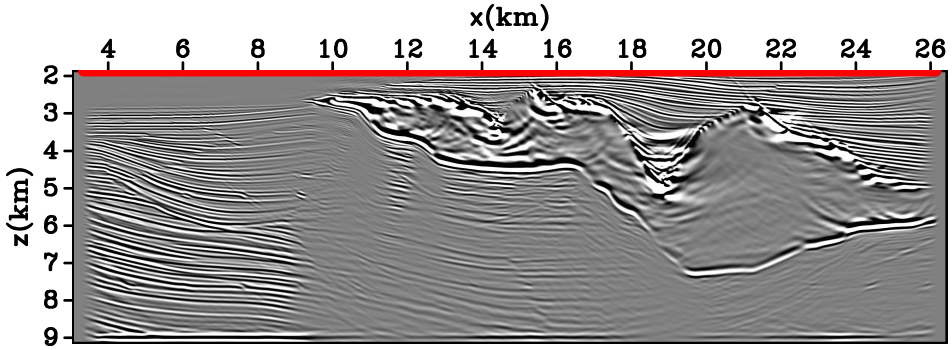




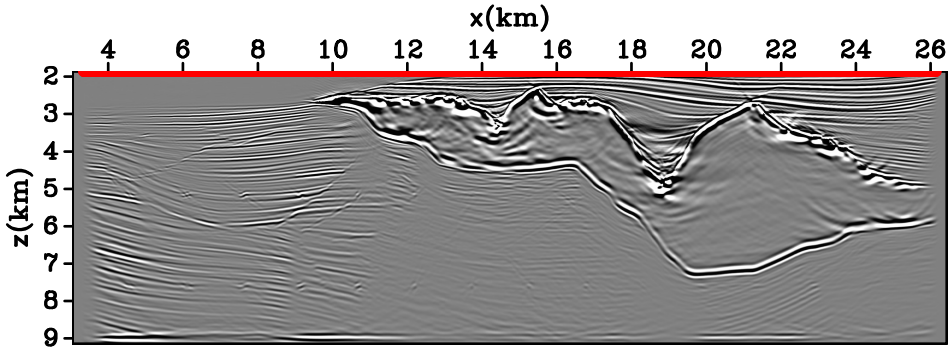
(a)



(b)



(c)



(d)

Figure 7.3. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Boxcar TSV encoding using 50 encodings.

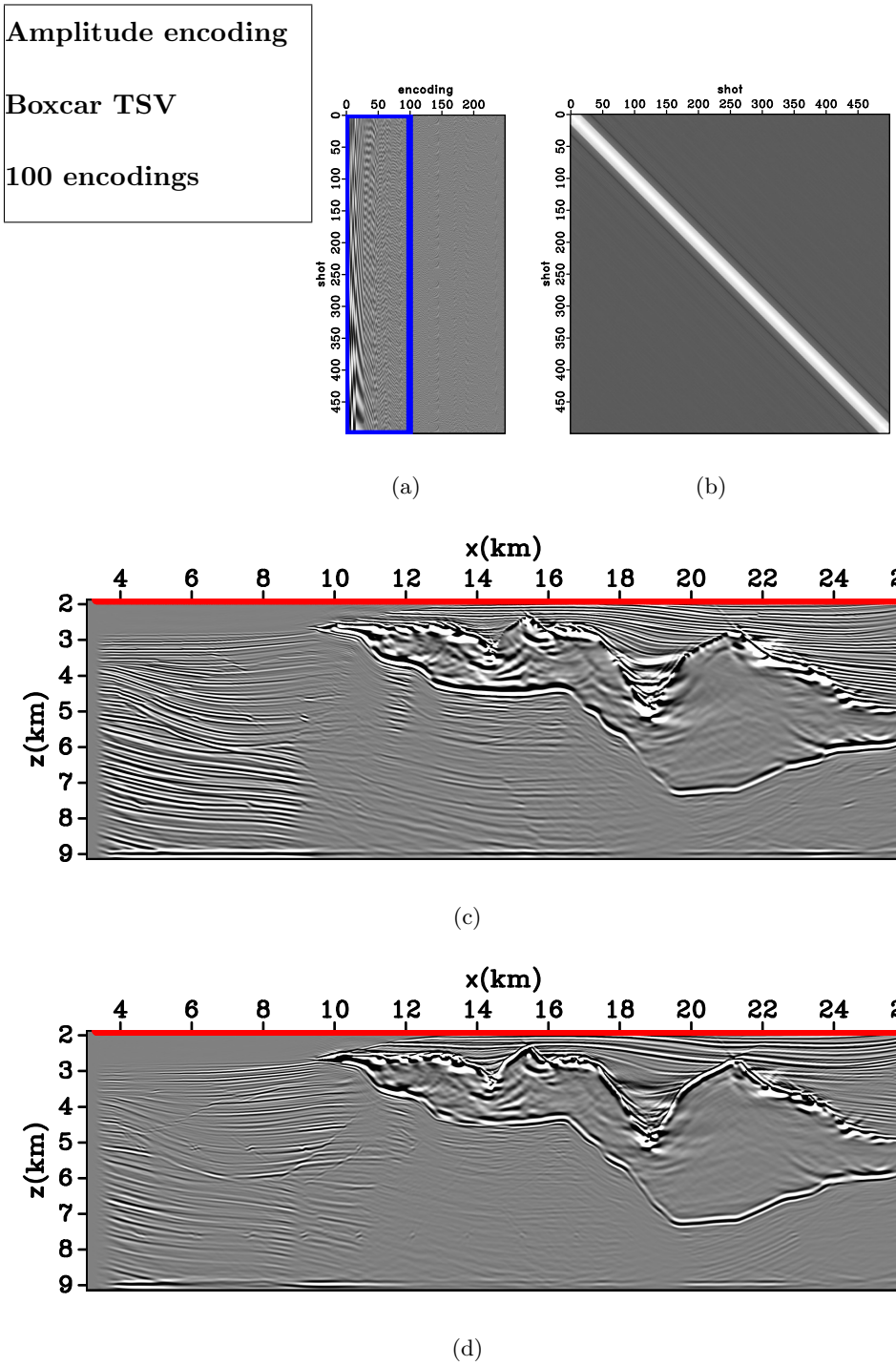
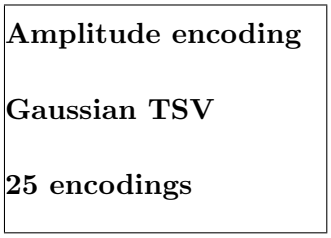
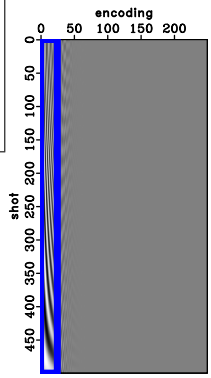
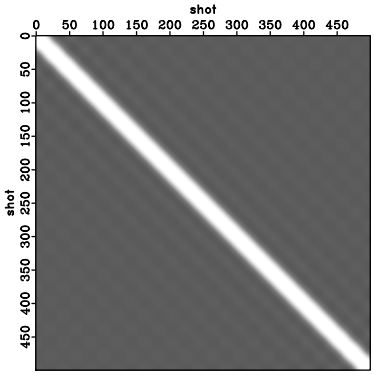


Figure 7.4. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Boxcar TSV encoding using 100 encodings.

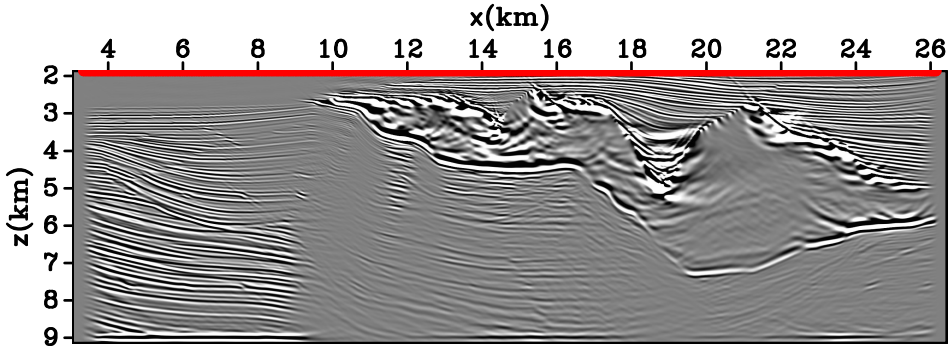




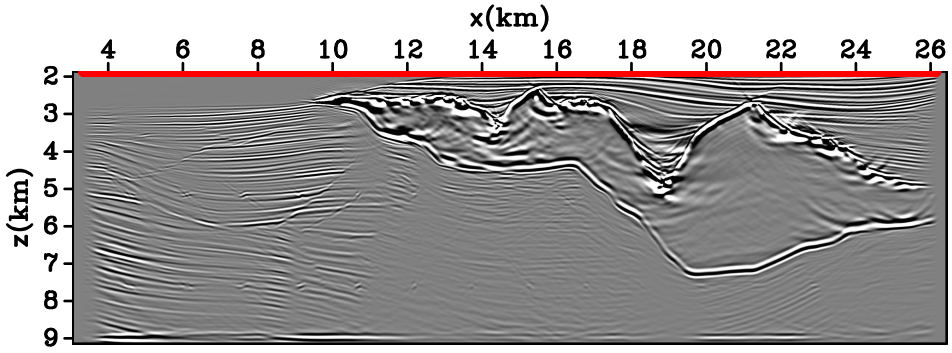
(a)



(b)



(c)



(d)

Figure 7.5. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Gaussian TSV encoding using 25 encodings.

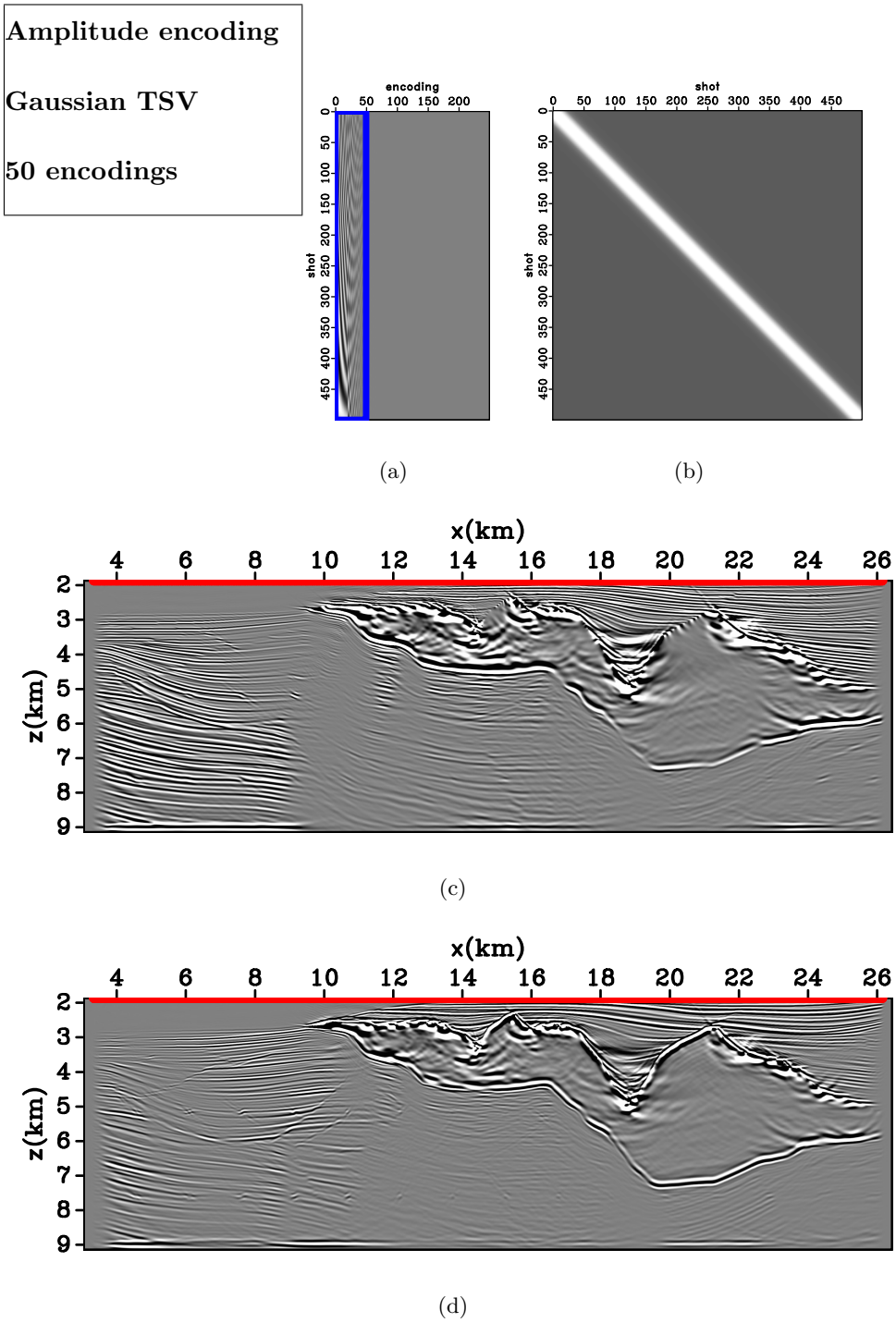
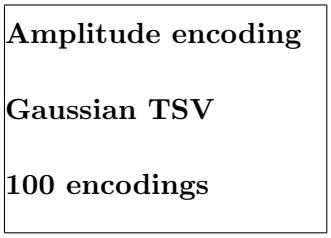
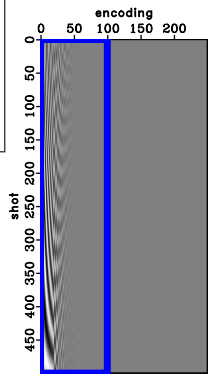
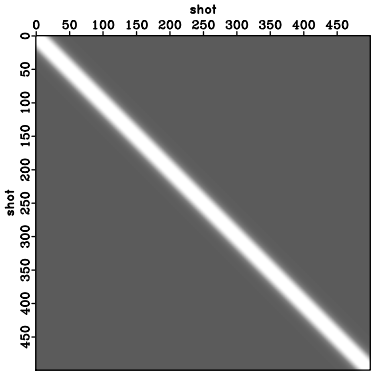


Figure 7.6. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Gaussian TSV encoding using 50 encodings.

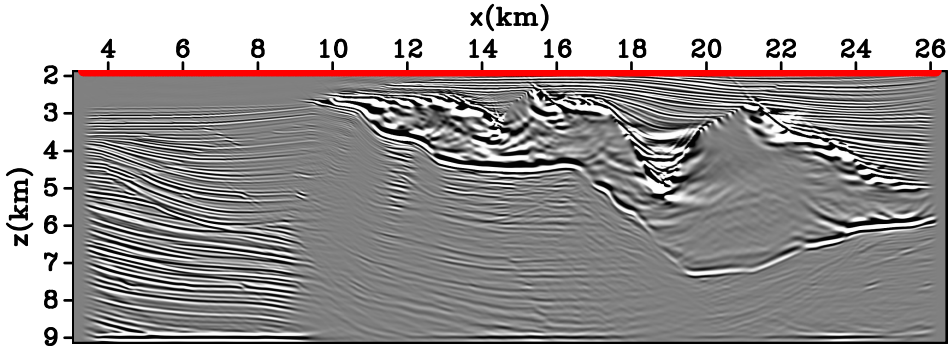




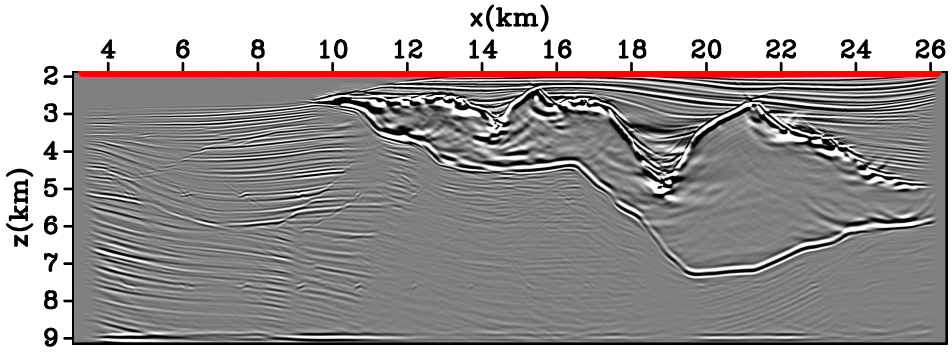
(a)



(b)



(c)



(d)

Figure 7.7. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Gaussian TSV encoding using 100 encodings.

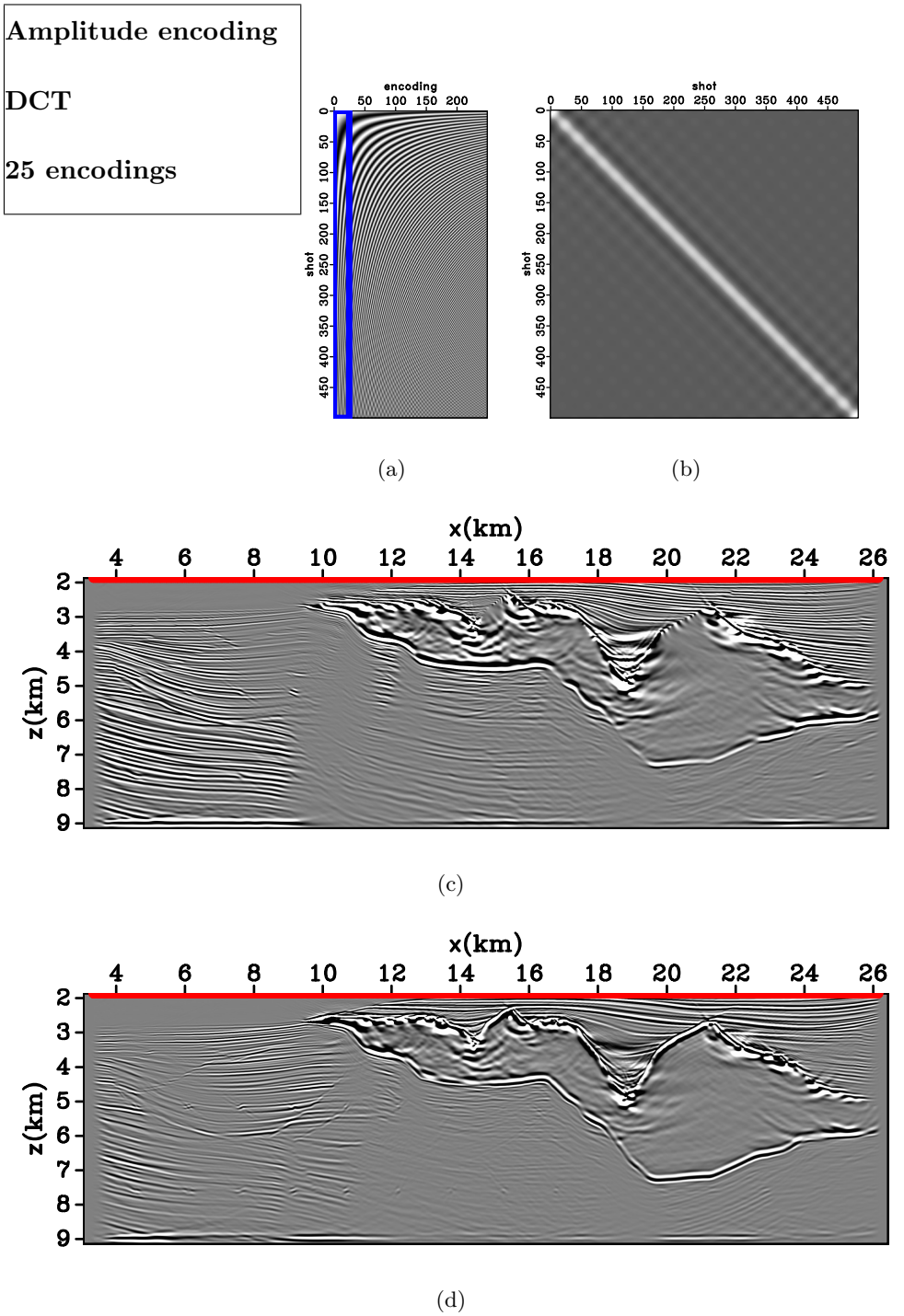
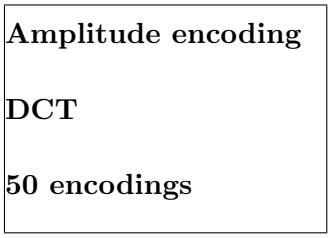
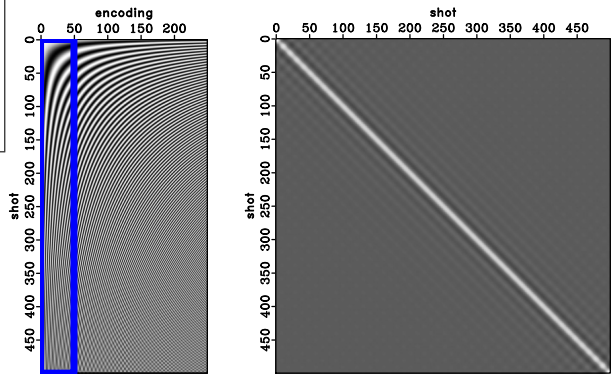
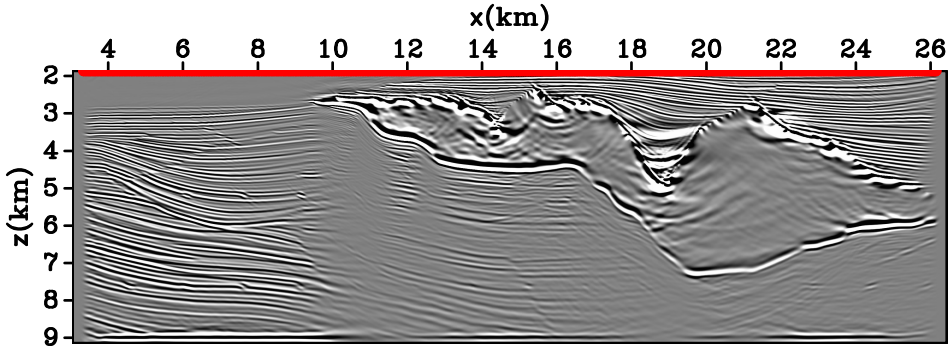


Figure 7.8. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT encoding using 25 encodings.

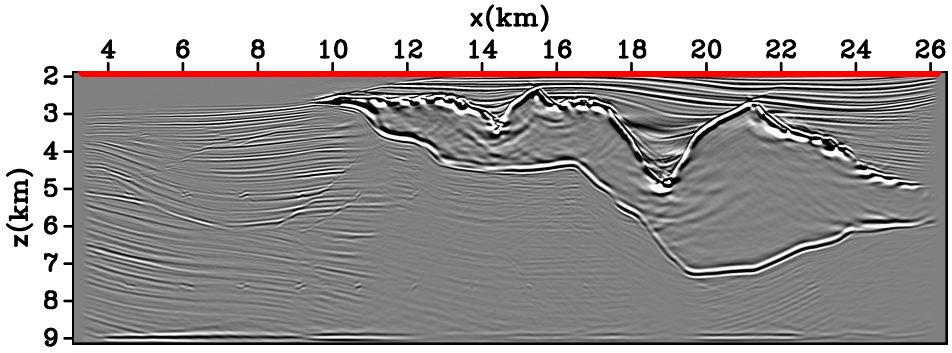




(a) (b)



(c)



(d)

Figure 7.9. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT encoding using 50 encodings.

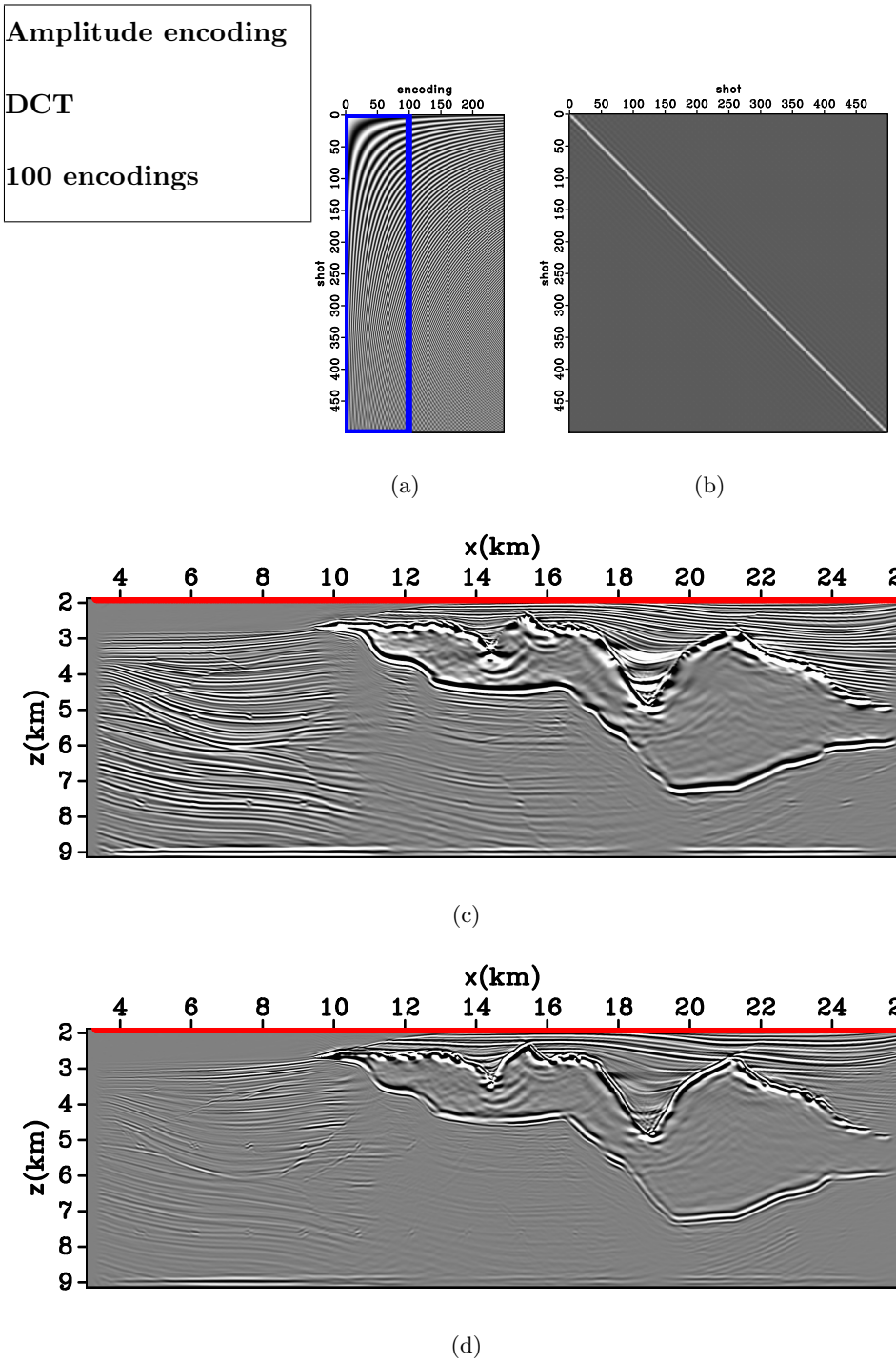
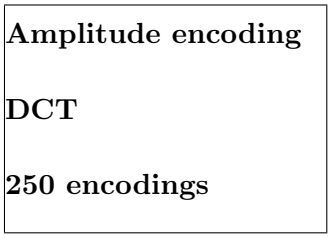
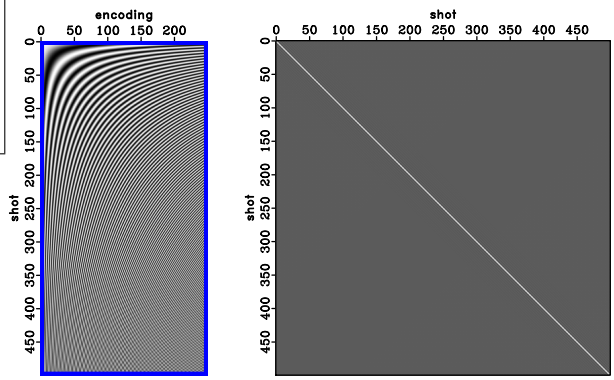
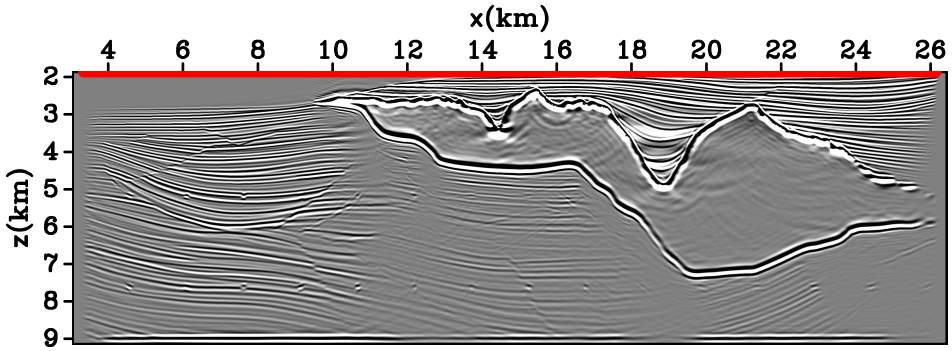


Figure 7.10. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT encoding using 100 encodings.

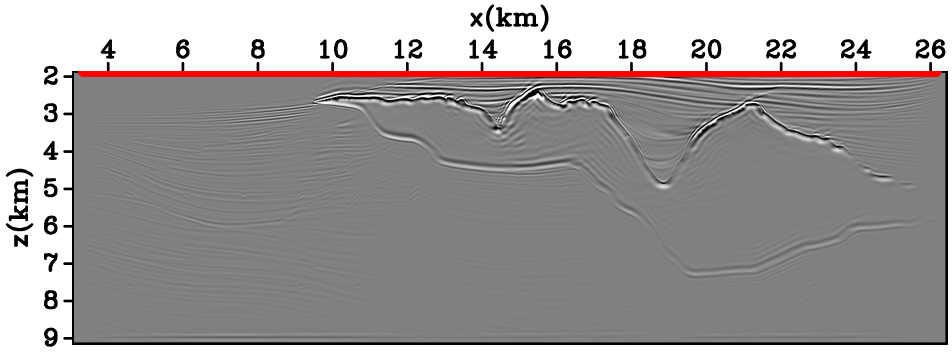




(a) (b)

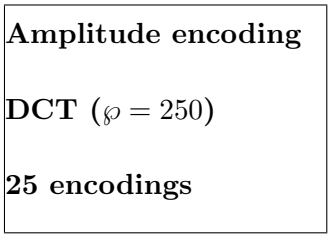


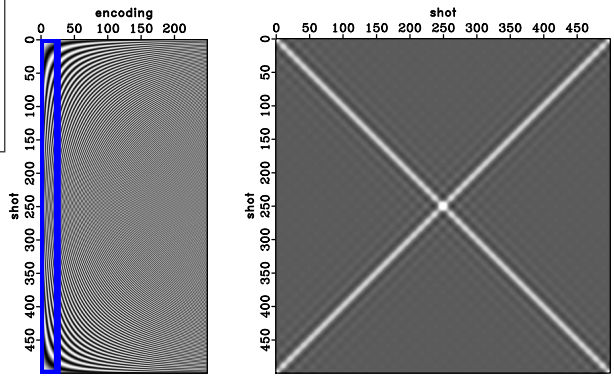
(c)



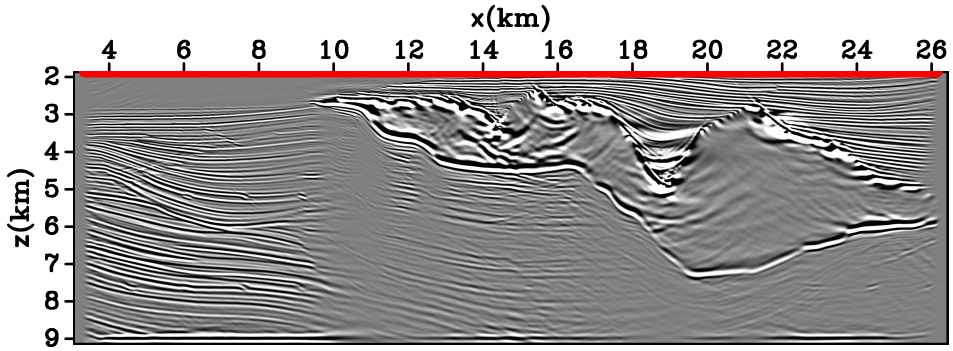
(d)

Figure 7.11. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT encoding using 250 encodings.

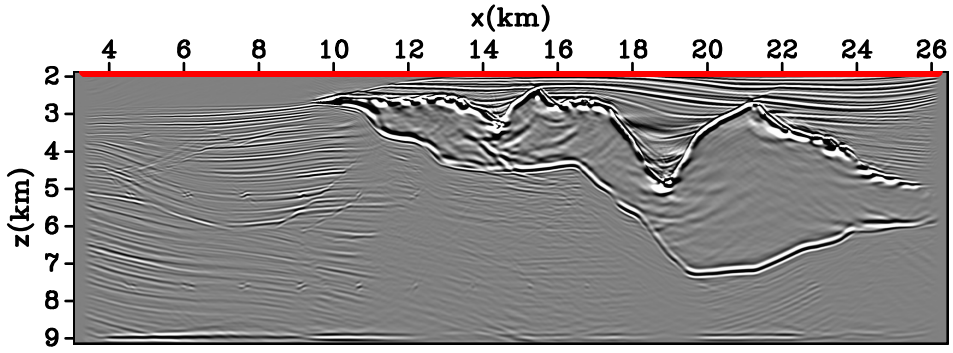




(a) (b)

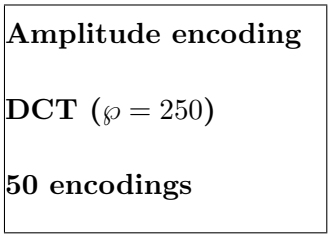


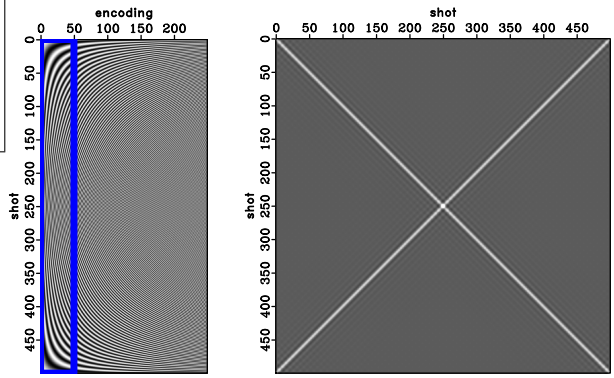
(c)



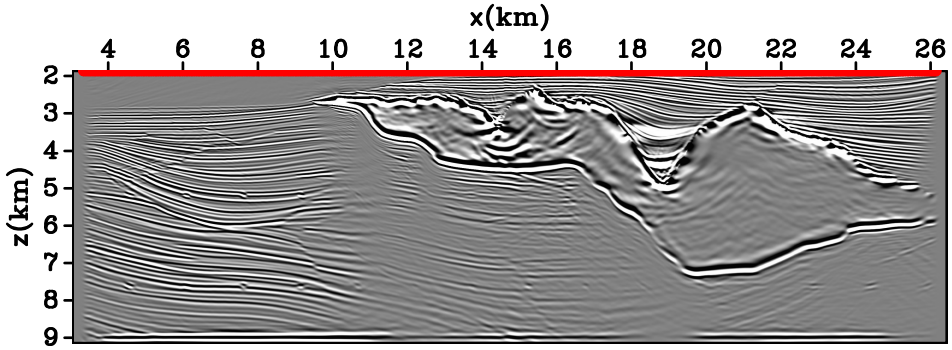
(d)

Figure 7.12. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT ($\wp = 250$) encoding using 25 encodings.

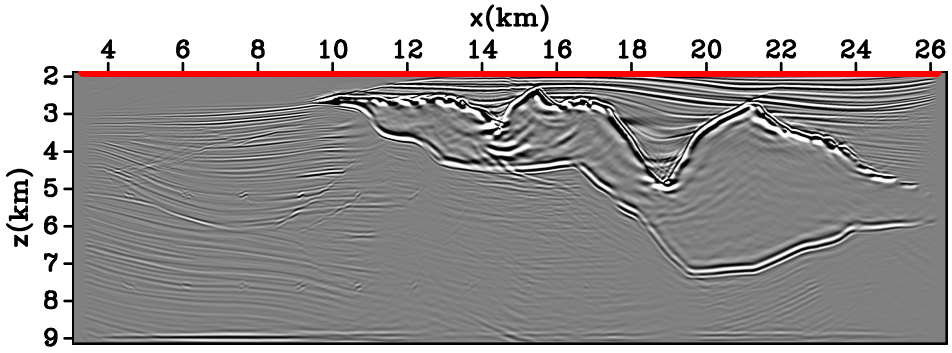




(a) (b)

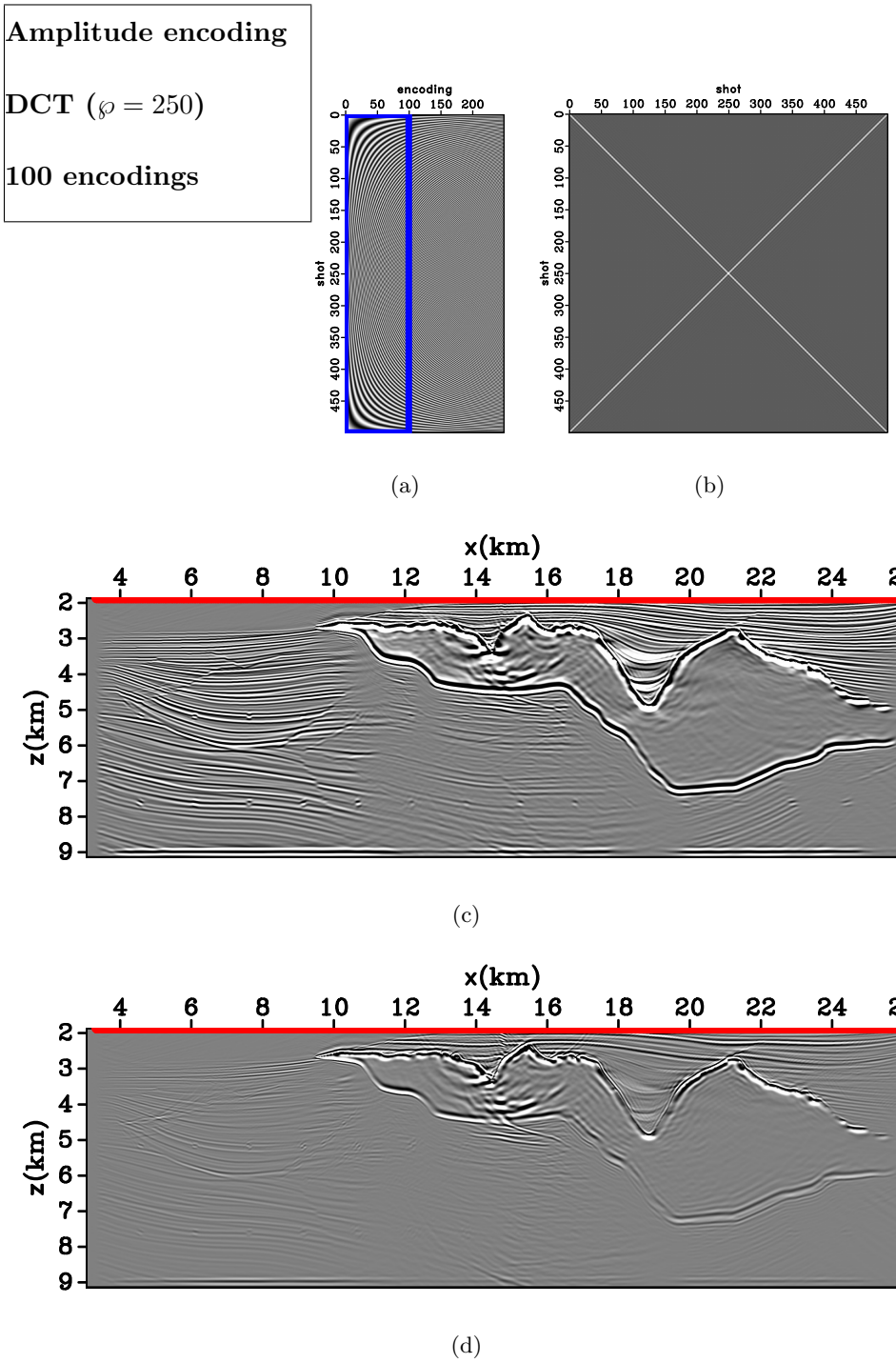


(c)



(d)

Figure 7.13. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT ($\wp = 250$) encoding using 50 encodings.

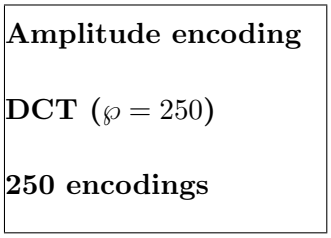


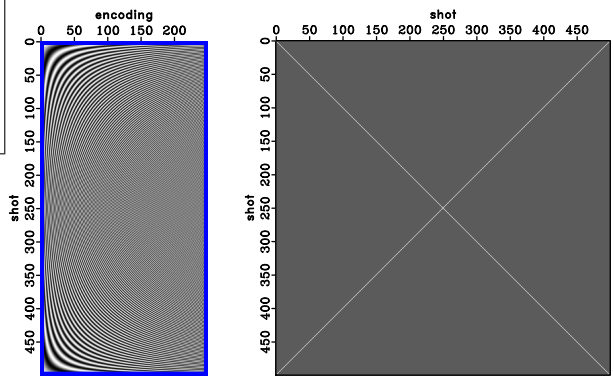
(a) (b)

(c)

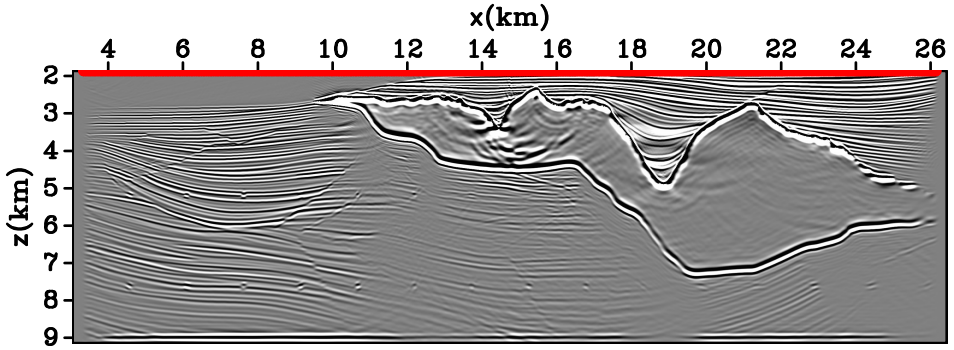
(d)

Figure 7.14. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT ($\wp = 250$) encoding using 100 encodings.

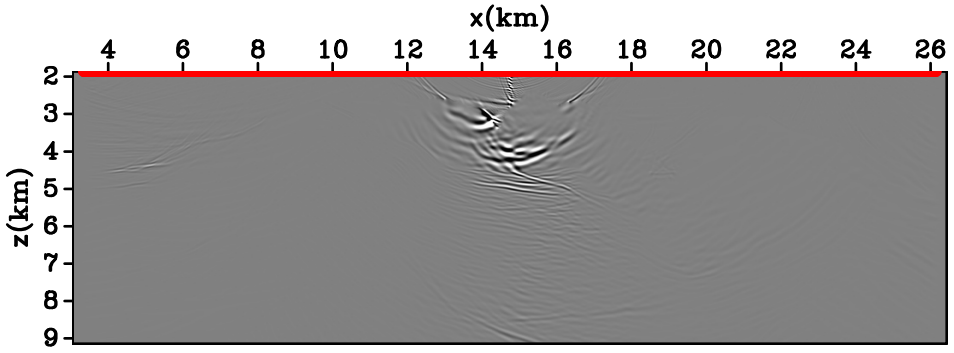




(a) (b)

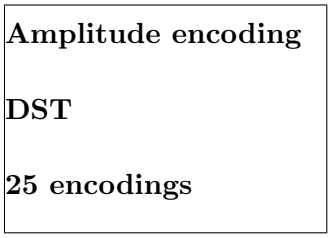


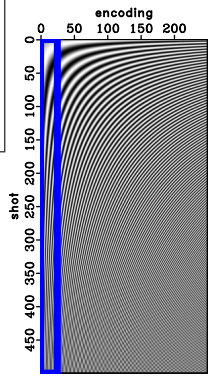
(c)



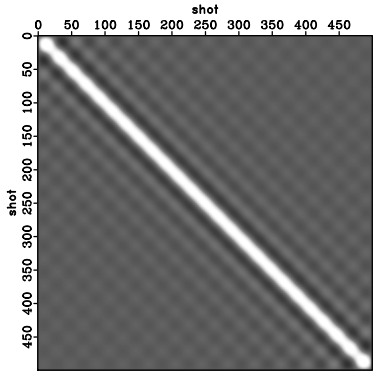
(d)

Figure 7.15. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCT ($\wp = 250$) encoding using 250 encodings.

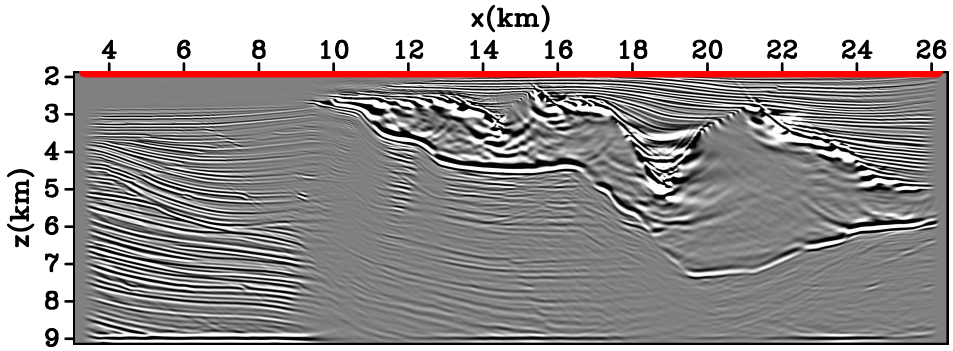




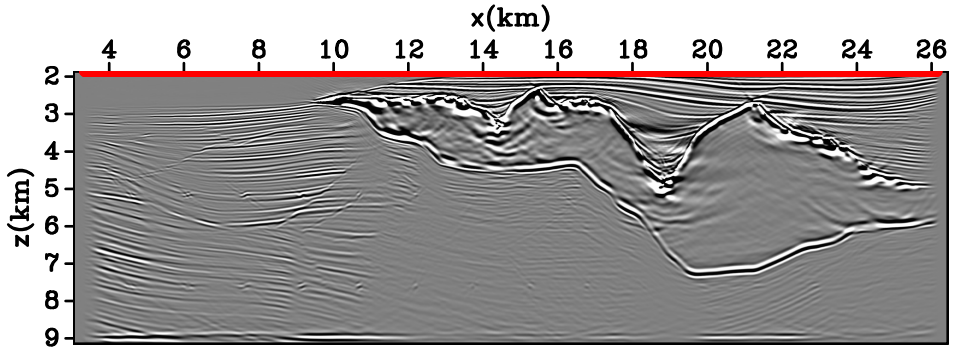
(a)



(b)

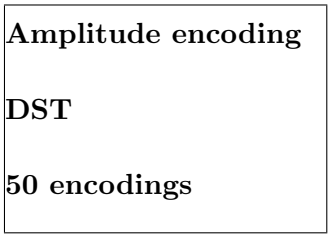


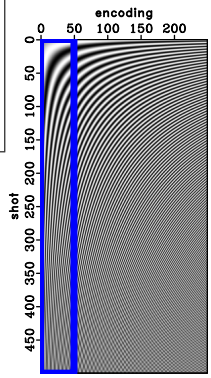
(c)



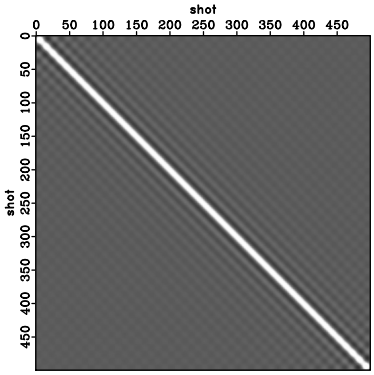
(d)

Figure 7.16. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST encoding using 25 encodings.

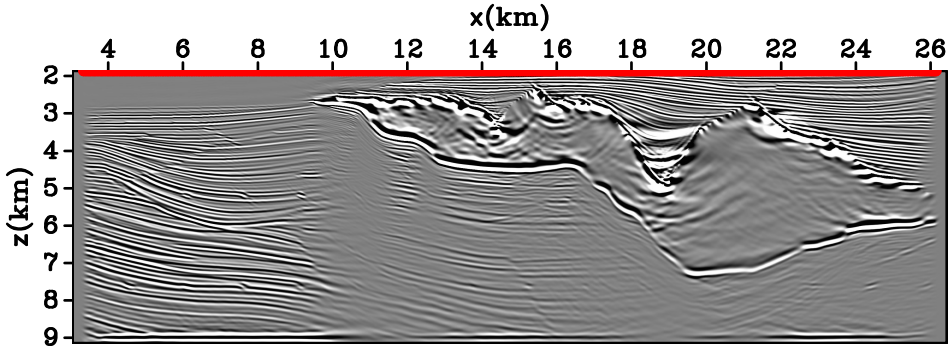




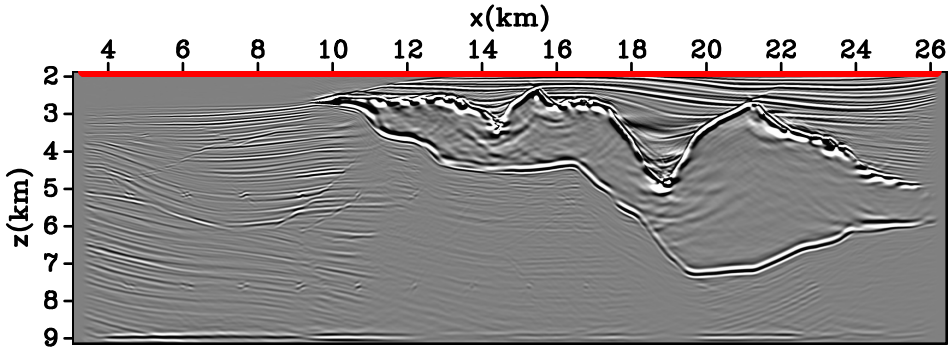
(a)



(b)



(c)



(d)

Figure 7.17. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST encoding using 50 encodings.

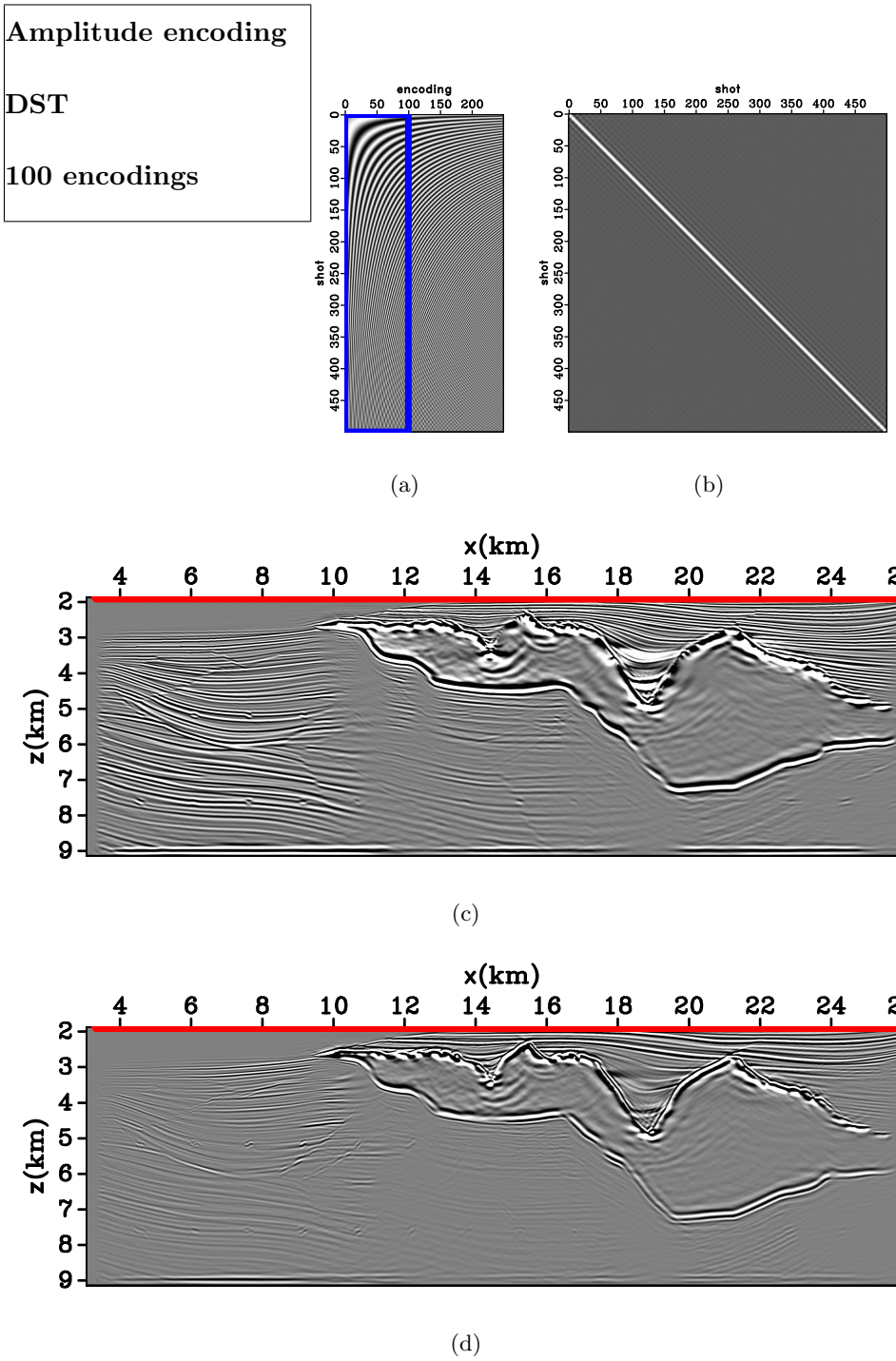


Figure 7.18. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST encoding using 100 encodings.

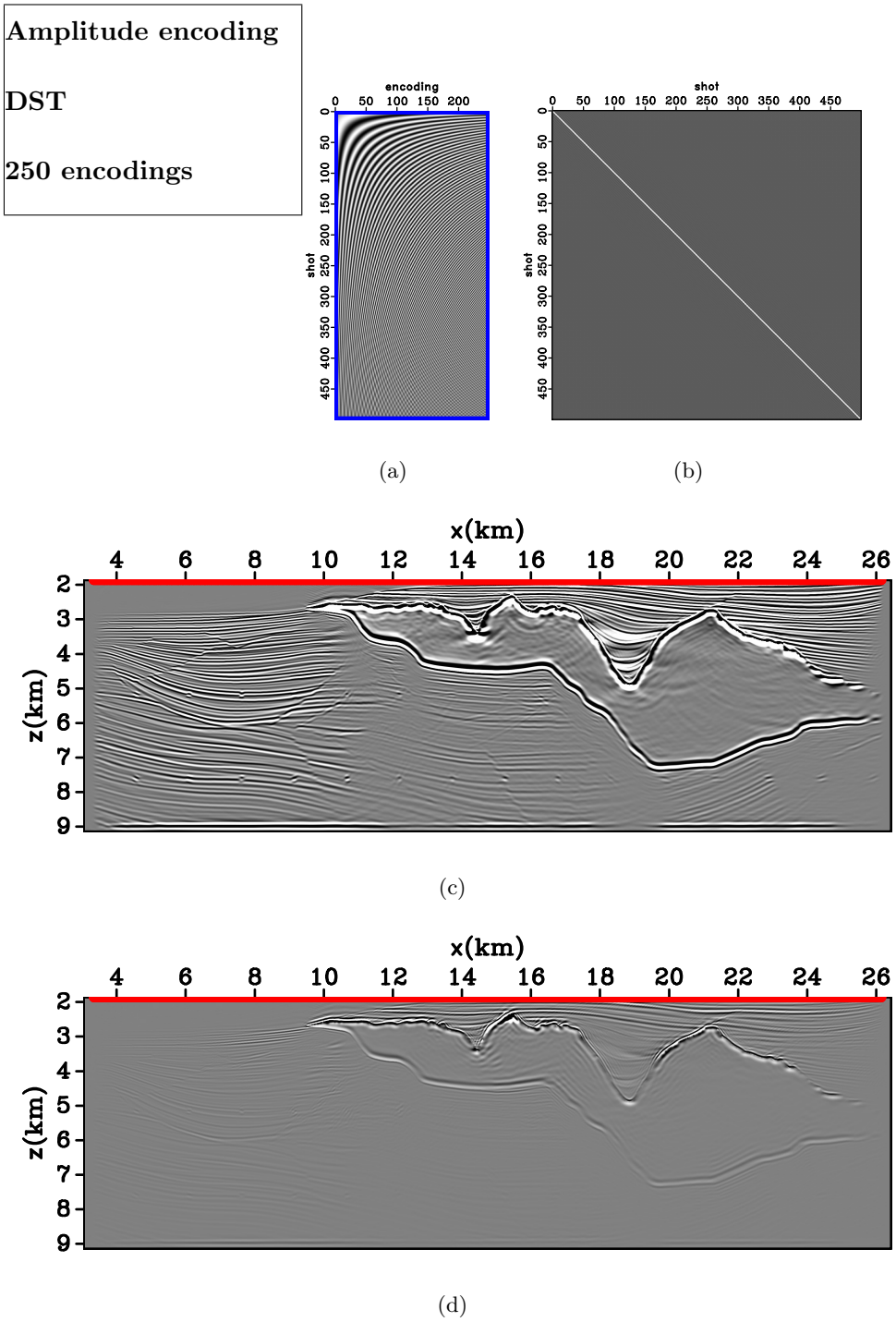


Figure 7.19. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST encoding using 250 encodings.

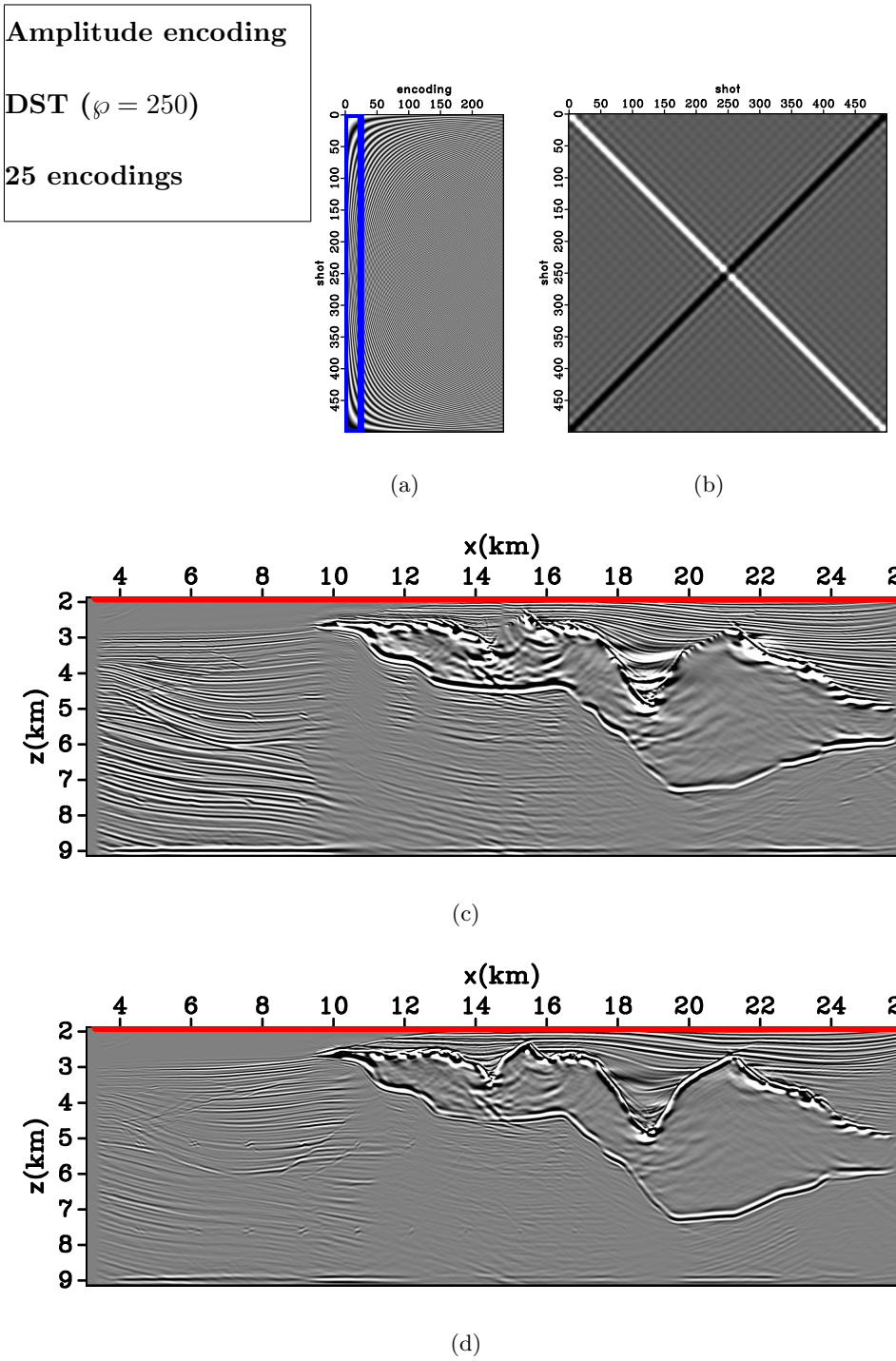
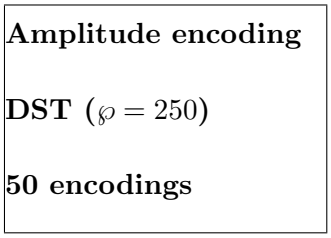
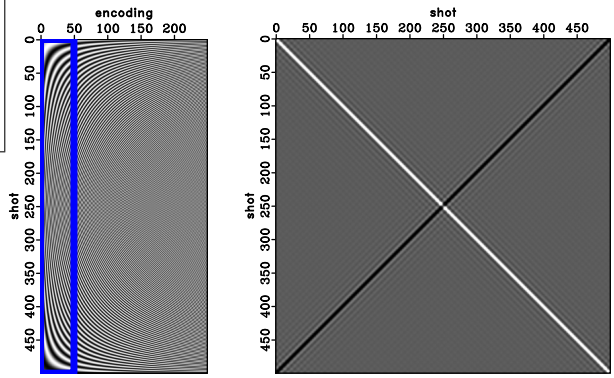
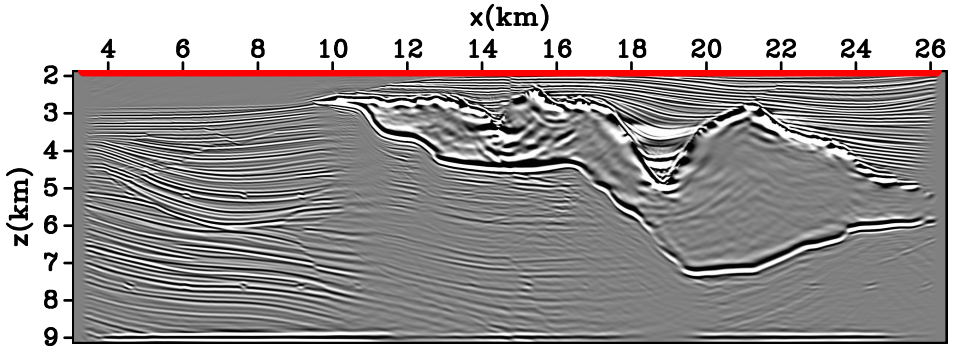


Figure 7.20. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST ($\wp = 250$) encoding using 25 encodings.

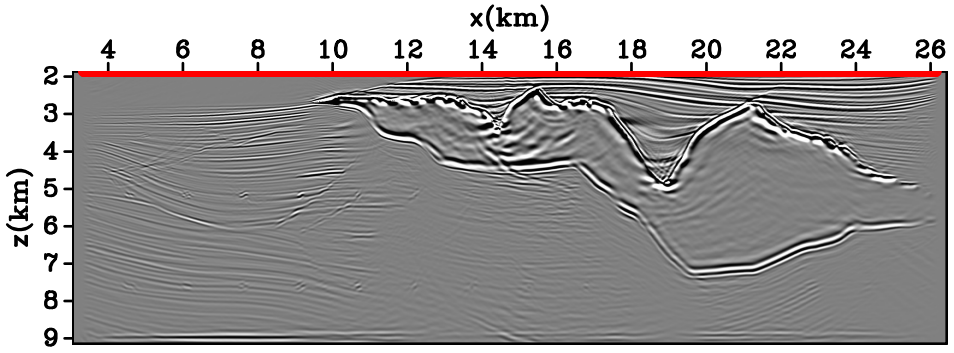




(a) (b)

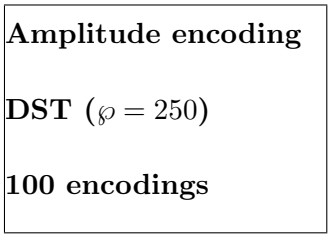


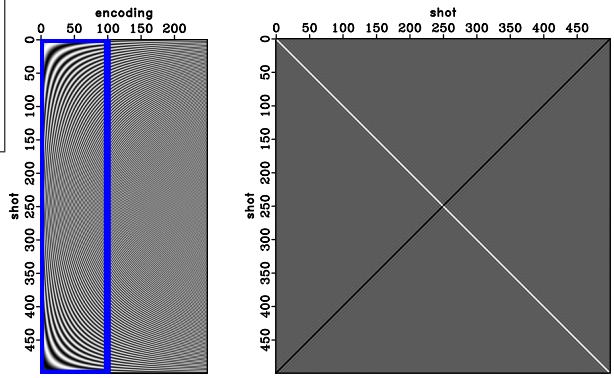
(c)



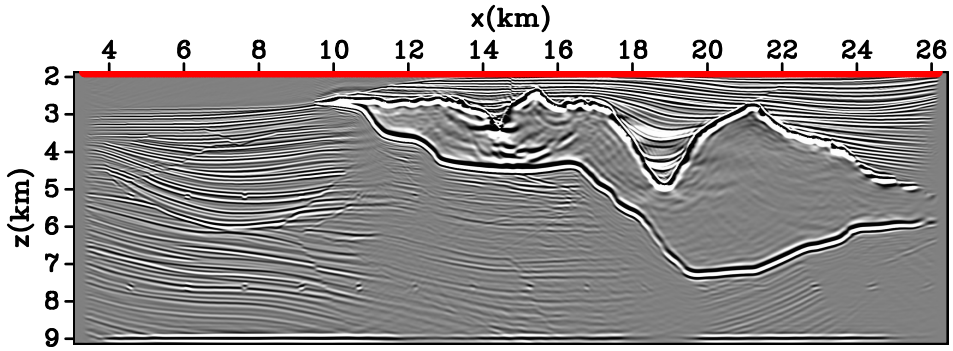
(d)

Figure 7.21. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST ($\wp = 250$) encoding using 50 encodings.

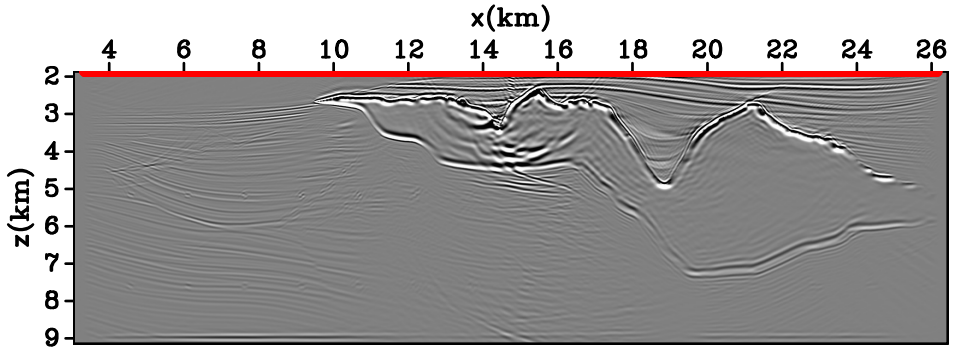




(a) (b)

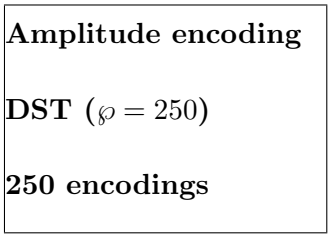


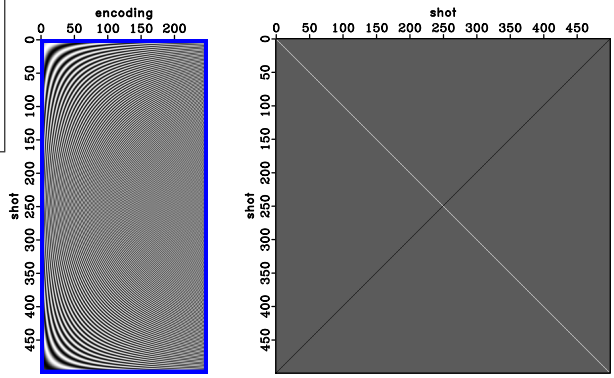
(c)



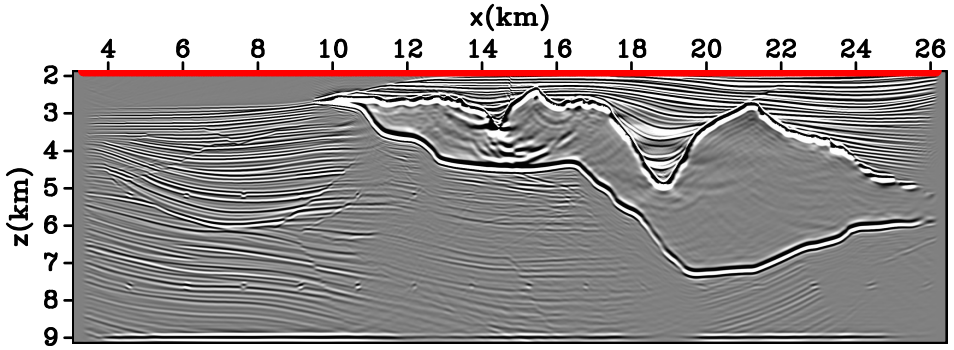
(d)

Figure 7.22. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST ($\wp = 250$) encoding using 100 encodings.

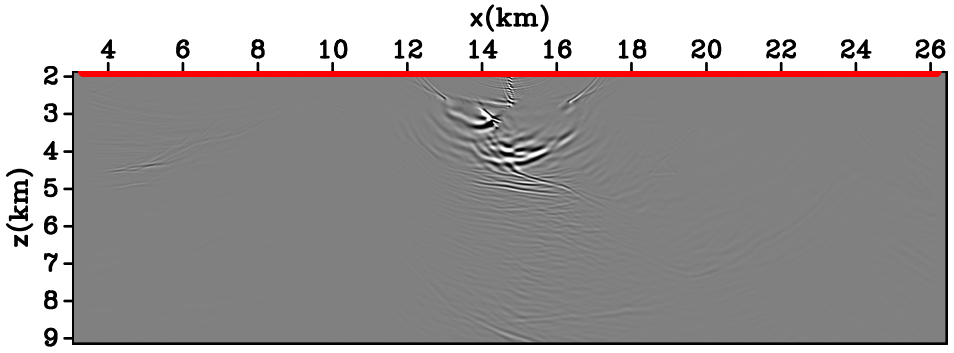




(a) (b)

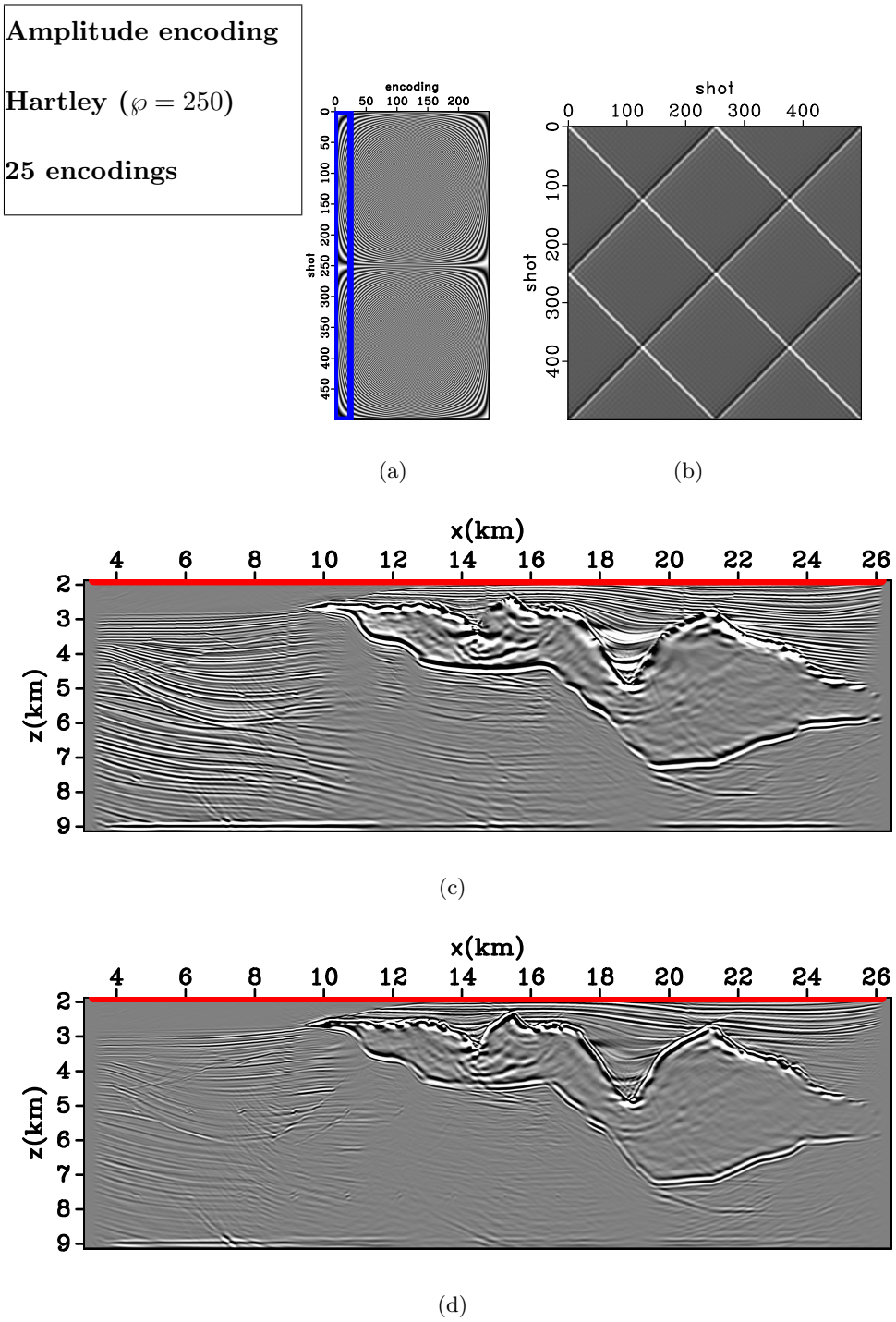


(c)



(d)

Figure 7.23. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DST ($\wp = 250$) encoding using 250 encodings.



(a) (b)

(c)

(d)

Figure 7.24. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hartley ($\wp = 250$) encoding using 25 encodings.

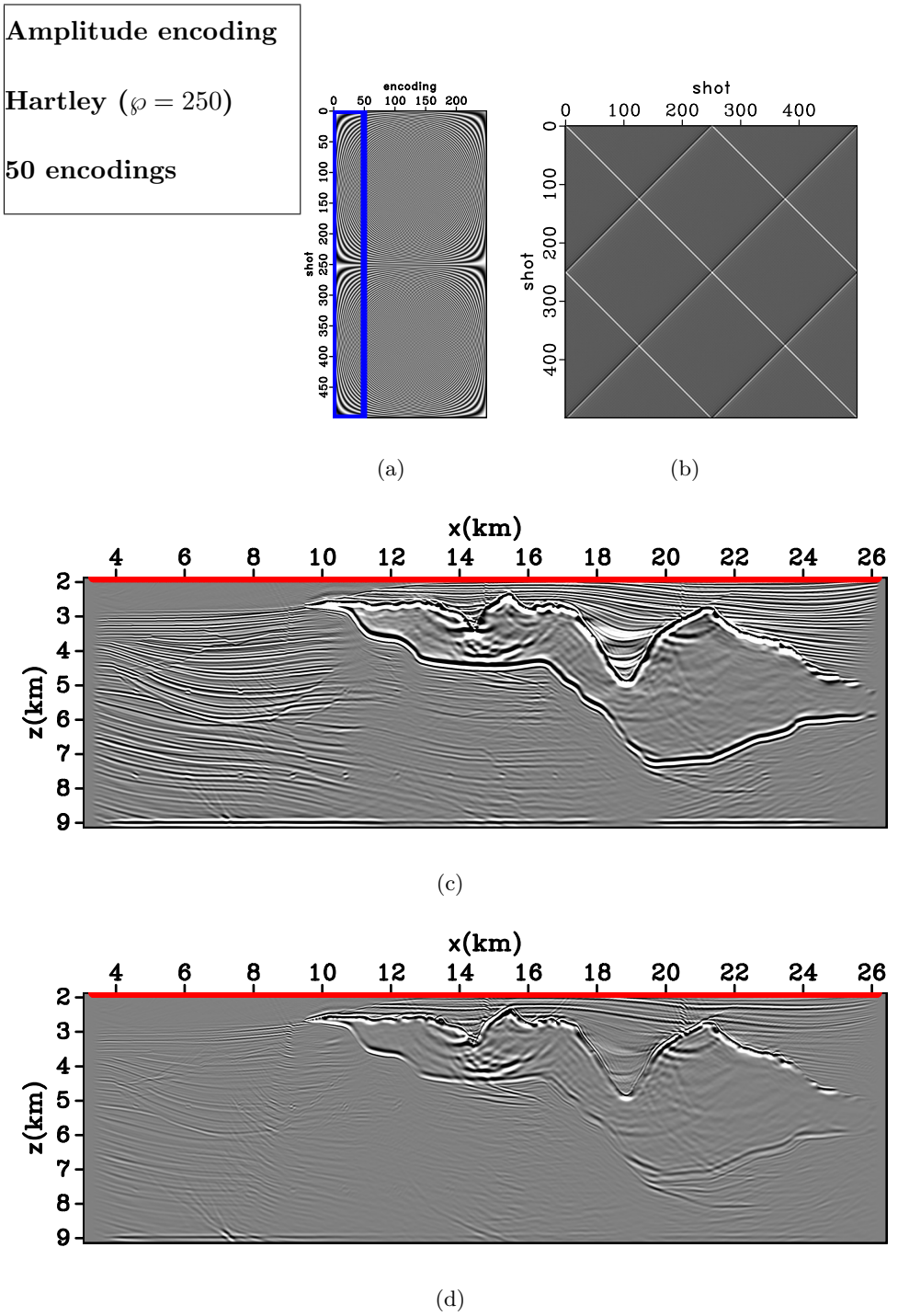


Figure 7.25. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hartley ($\wp = 250$) encoding using 50 encodings.

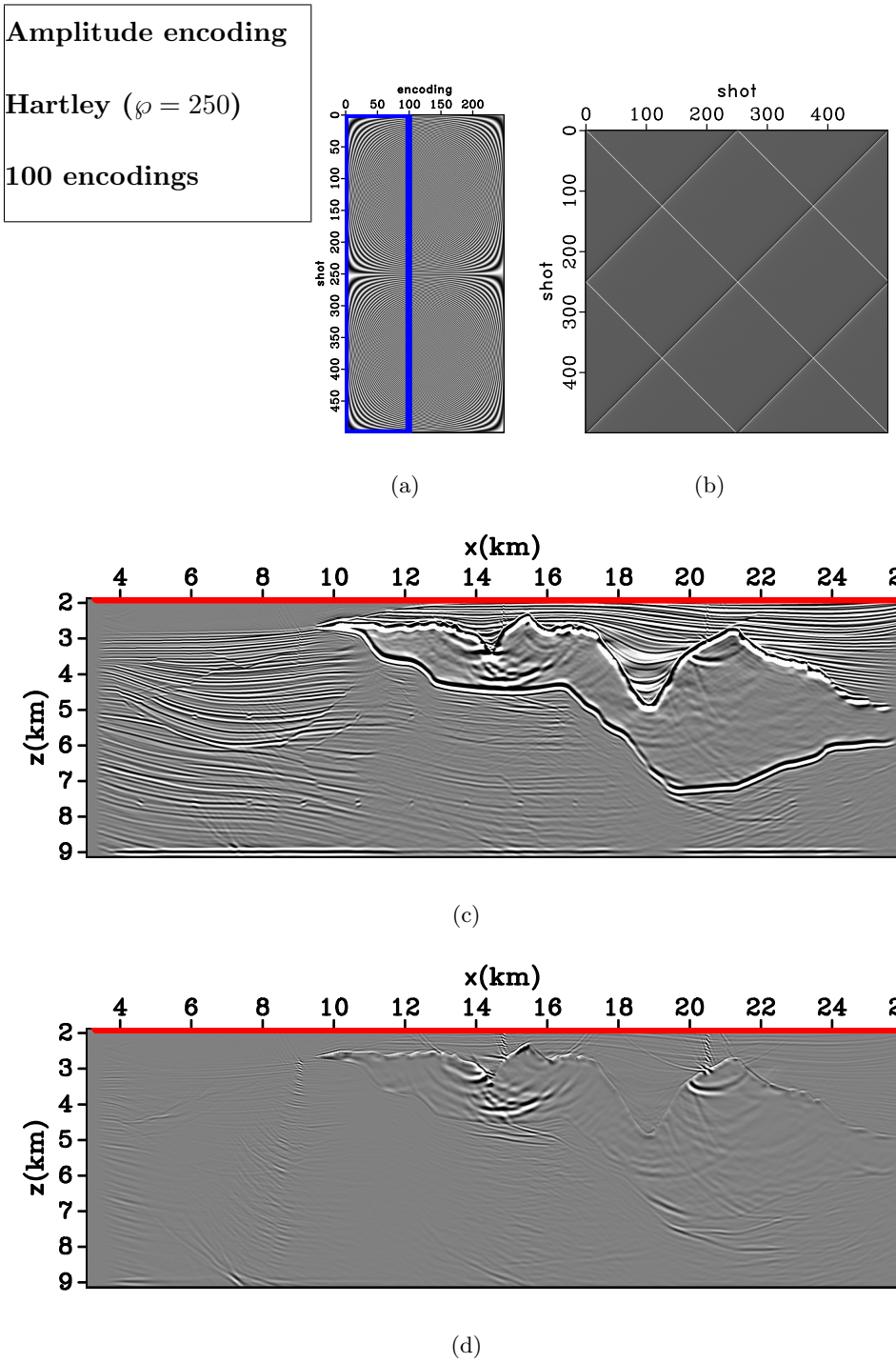
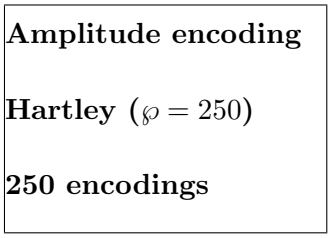
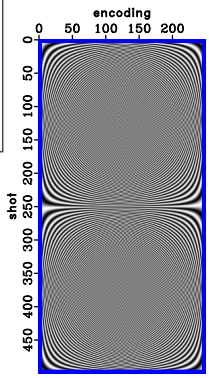
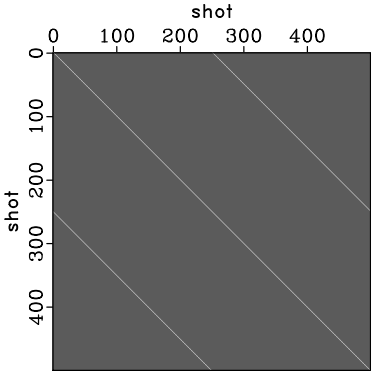


Figure 7.26. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hartley ($\wp = 250$) encoding using 100 encodings.

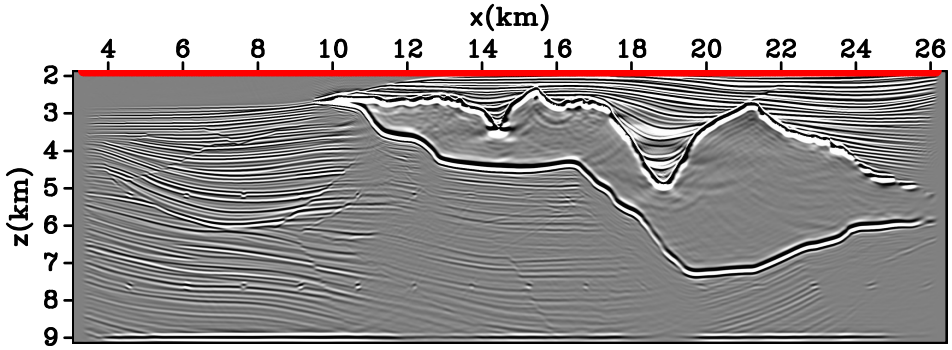




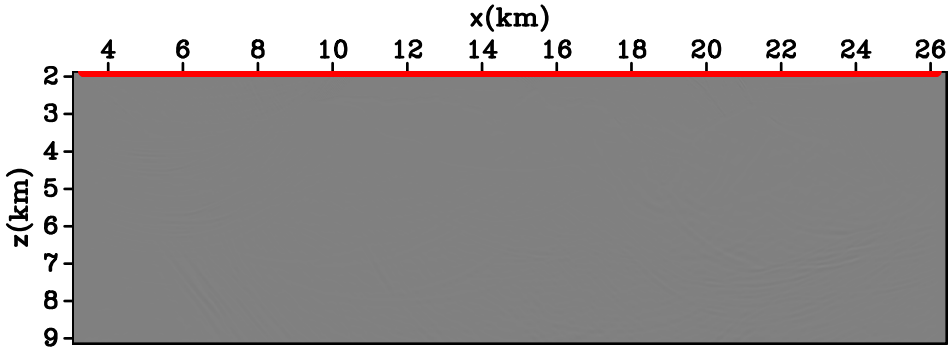
(a)



(b)

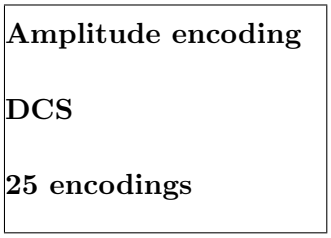


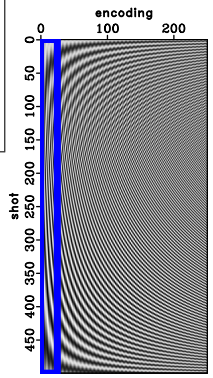
(c)



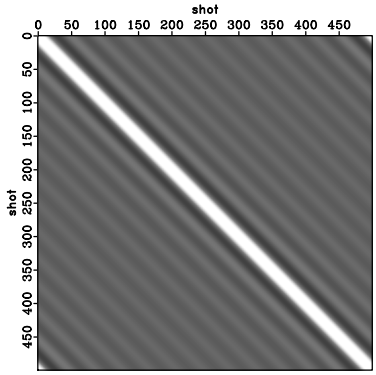
(d)

Figure 7.27. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hartley ($\wp = 250$) encoding using 250 encodings.

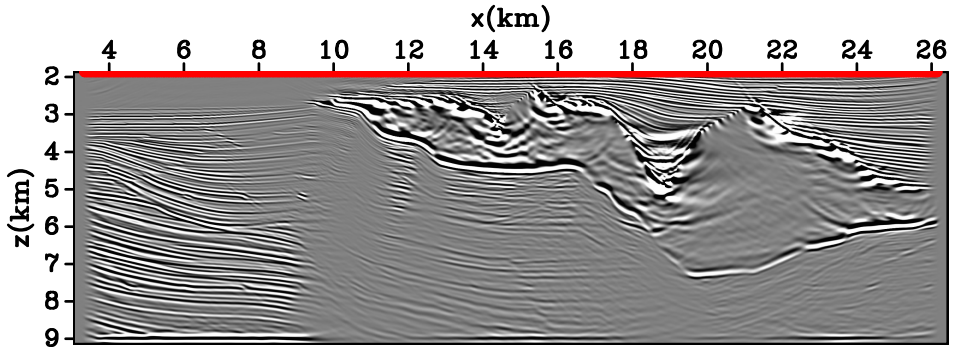




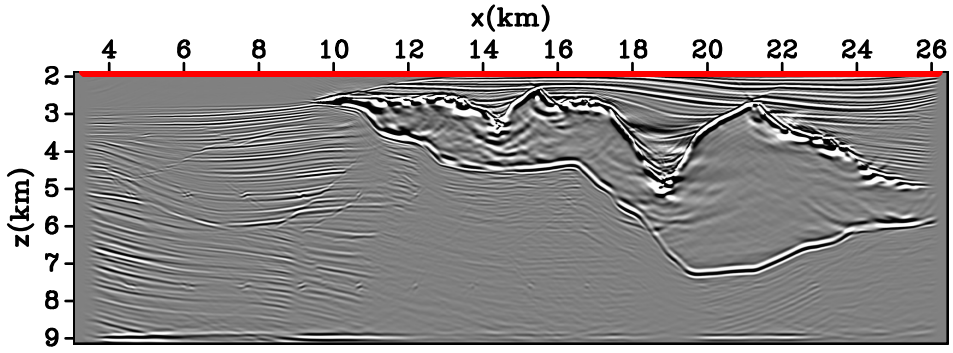
(a)



(b)

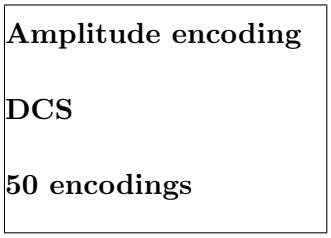


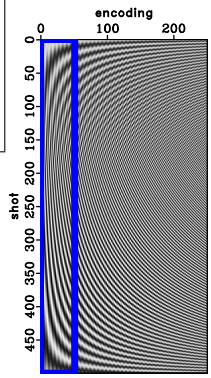
(c)



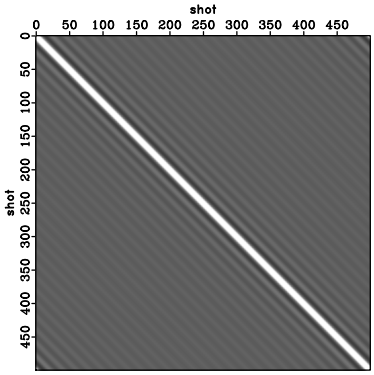
(d)

Figure 7.28. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS encoding using 25 encodings.

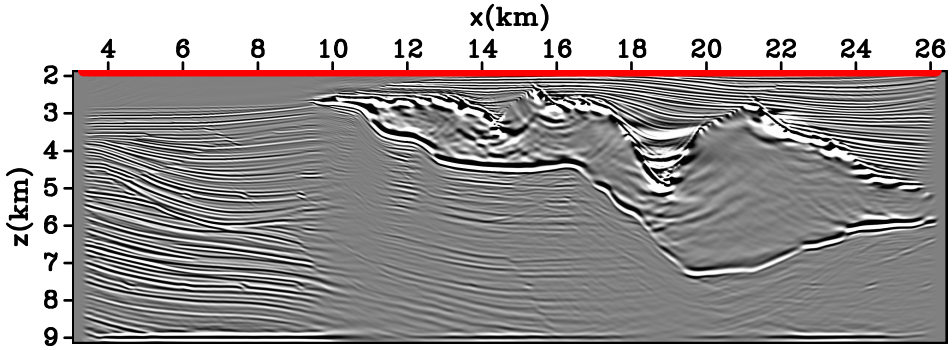




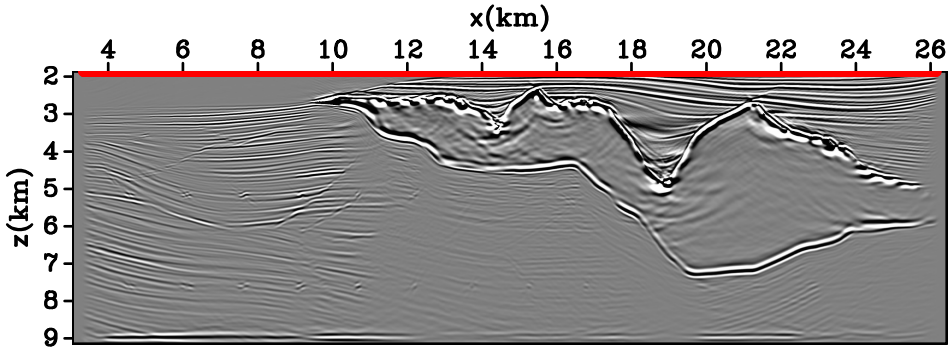
(a)



(b)



(c)



(d)

Figure 7.29. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS encoding using 51 encodings.

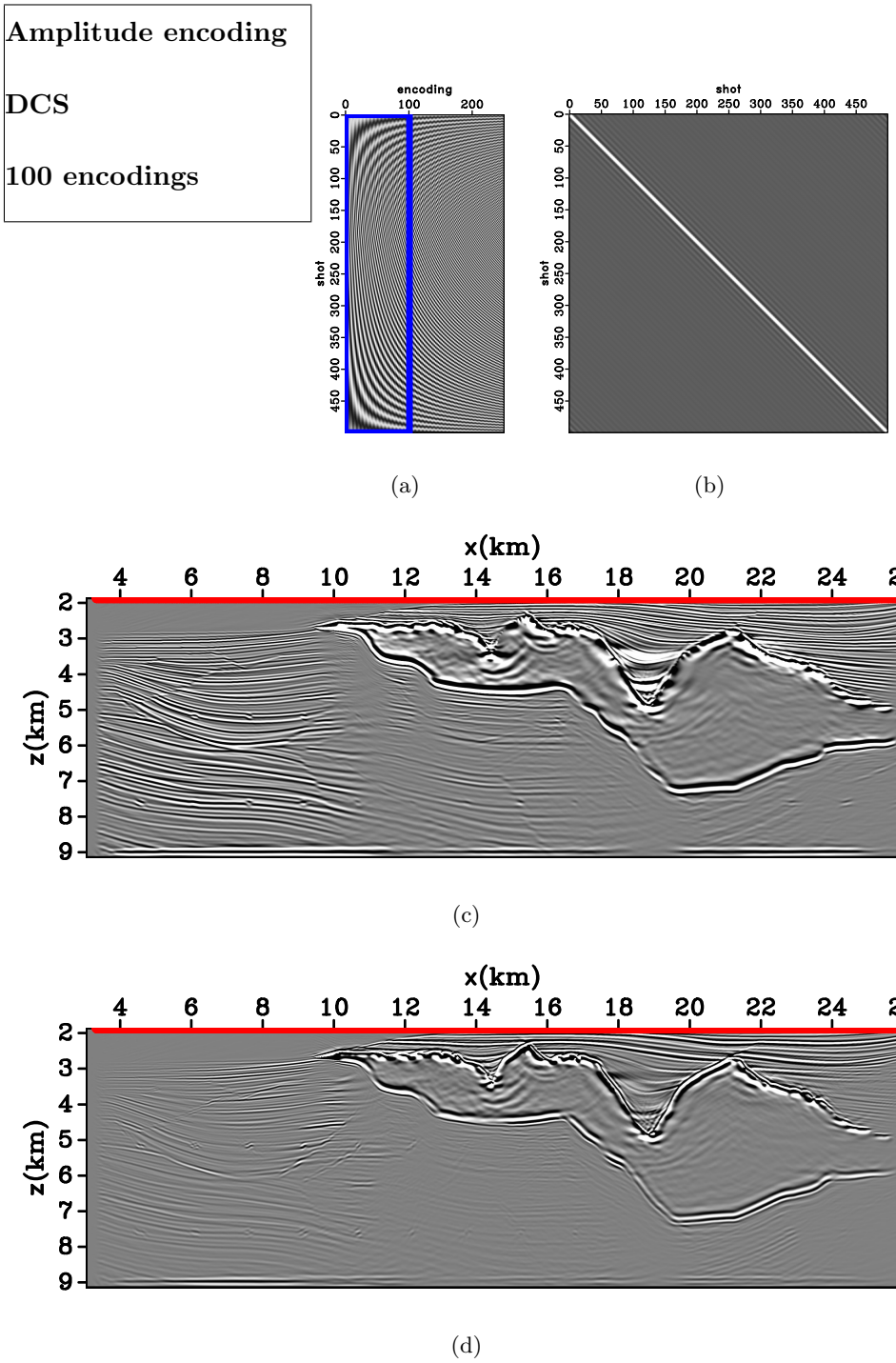
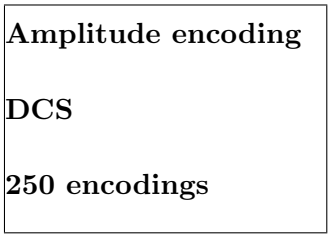
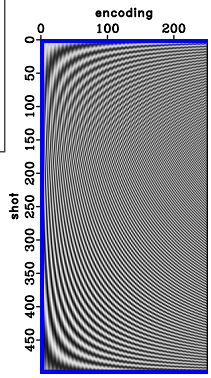
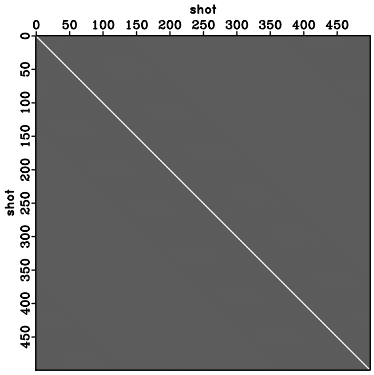


Figure 7.30. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS encoding using 101 encodings.

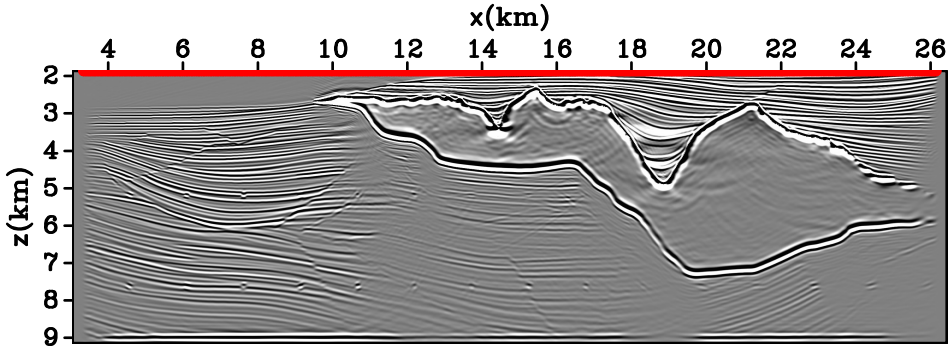




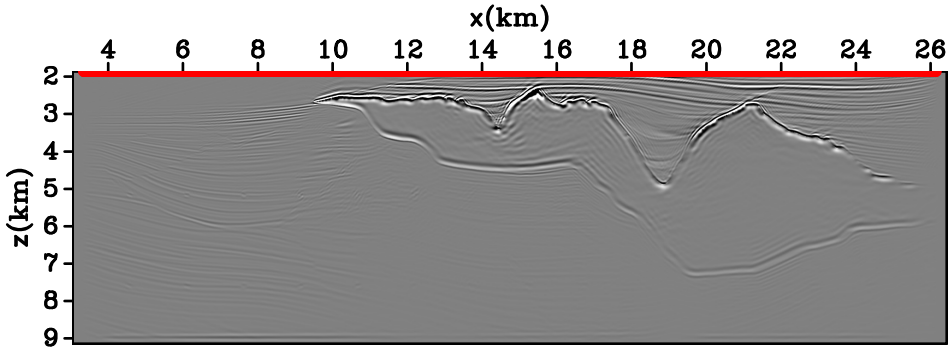
(a)



(b)



(c)



(d)

Figure 7.31. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS encoding using 251 encodings.

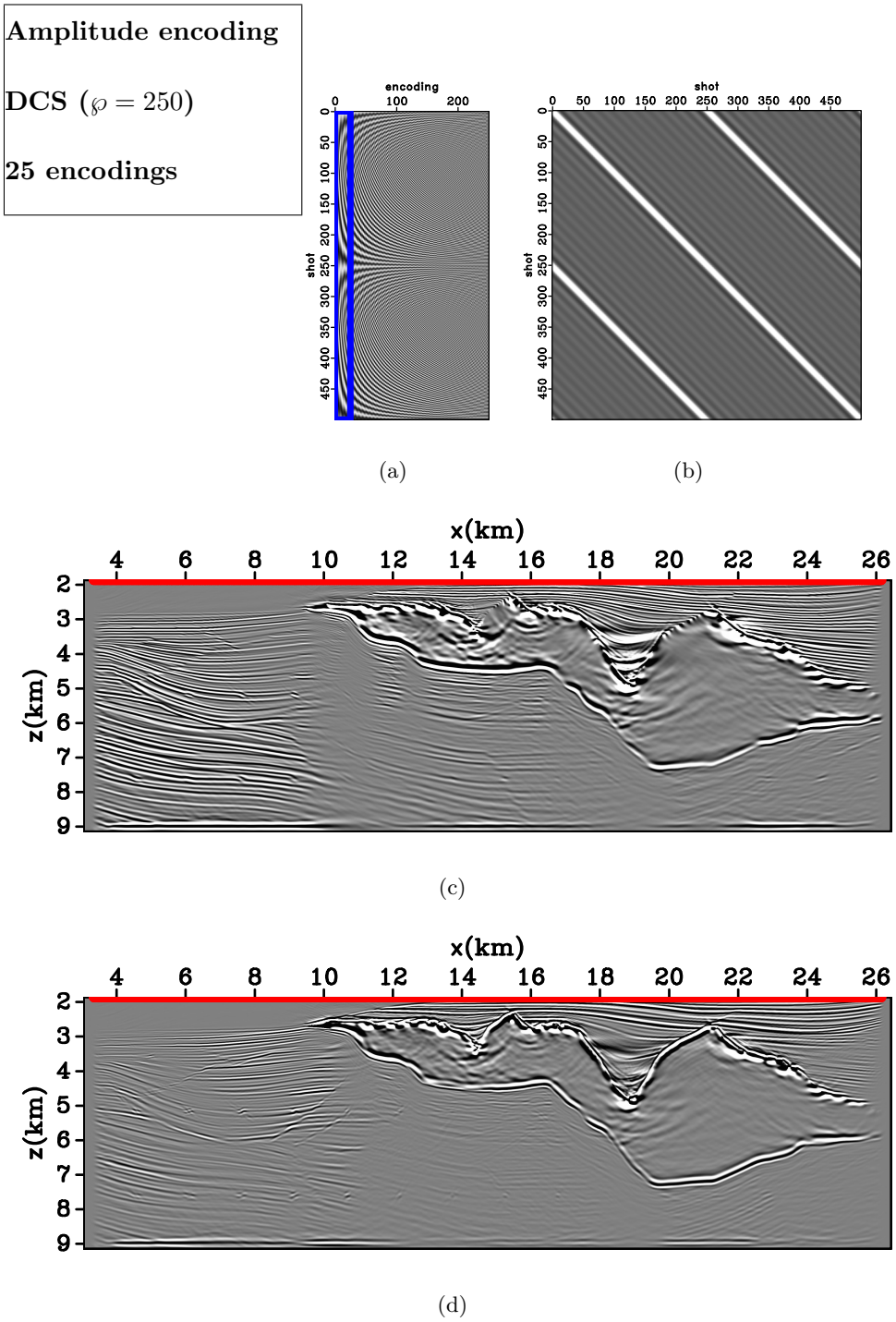
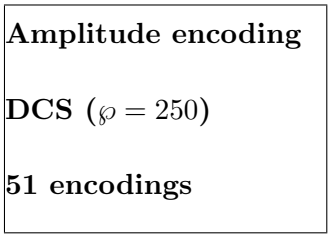
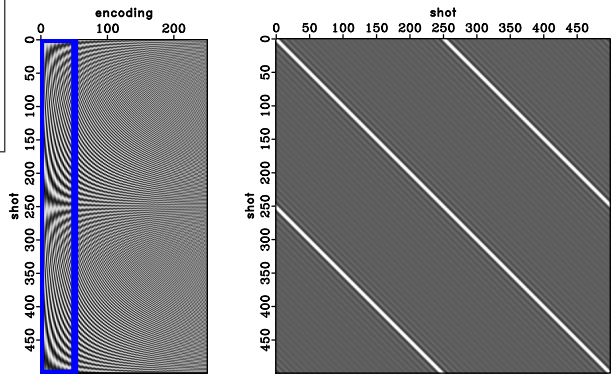
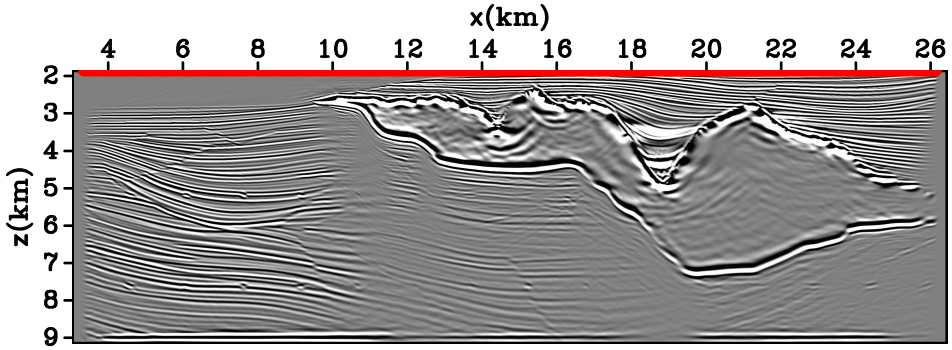


Figure 7.32. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS ($\wp = 250$) encoding using 25 encodings.

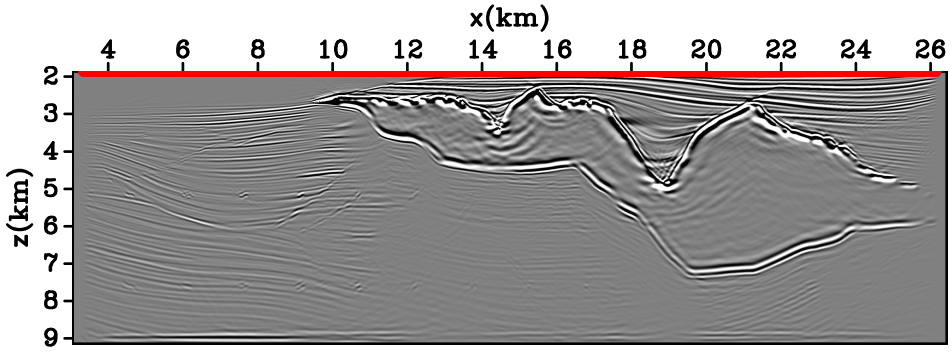




(a) (b)



(c)



(d)

Figure 7.33. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS ($\wp = 250$) encoding using 51 encodings.

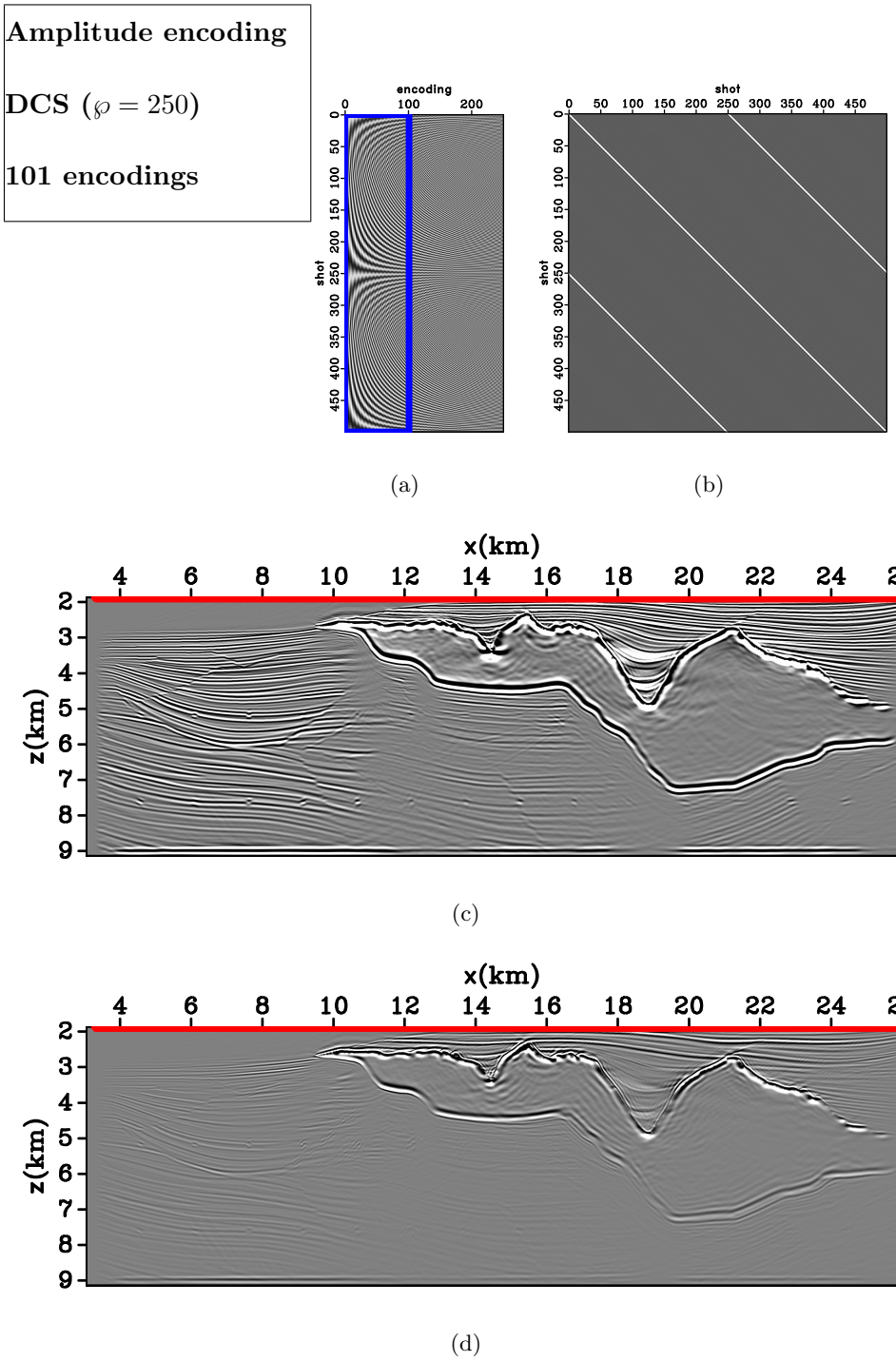
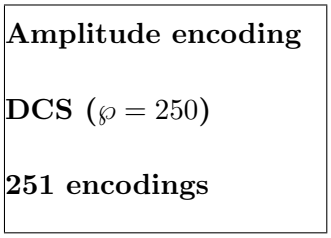
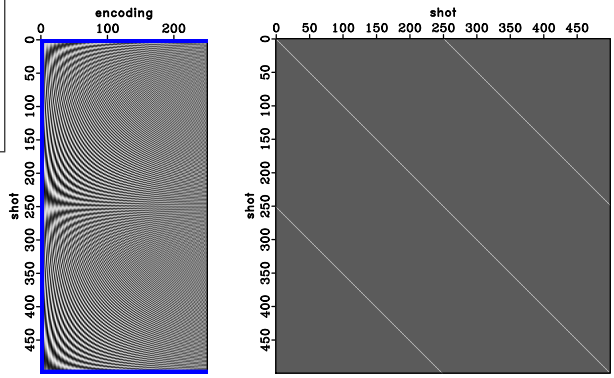
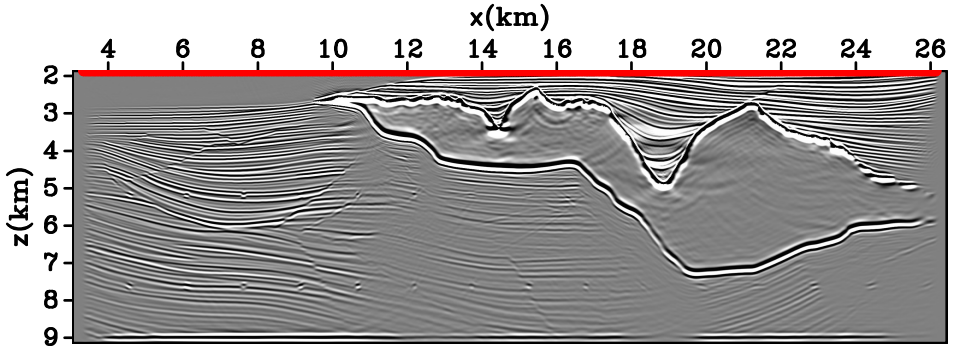


Figure 7.34. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS ($\wp = 250$) encoding using 101 encodings.

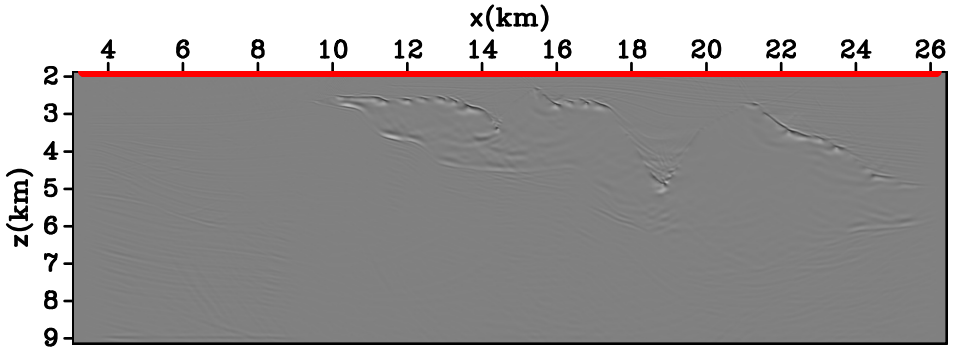




(a) (b)

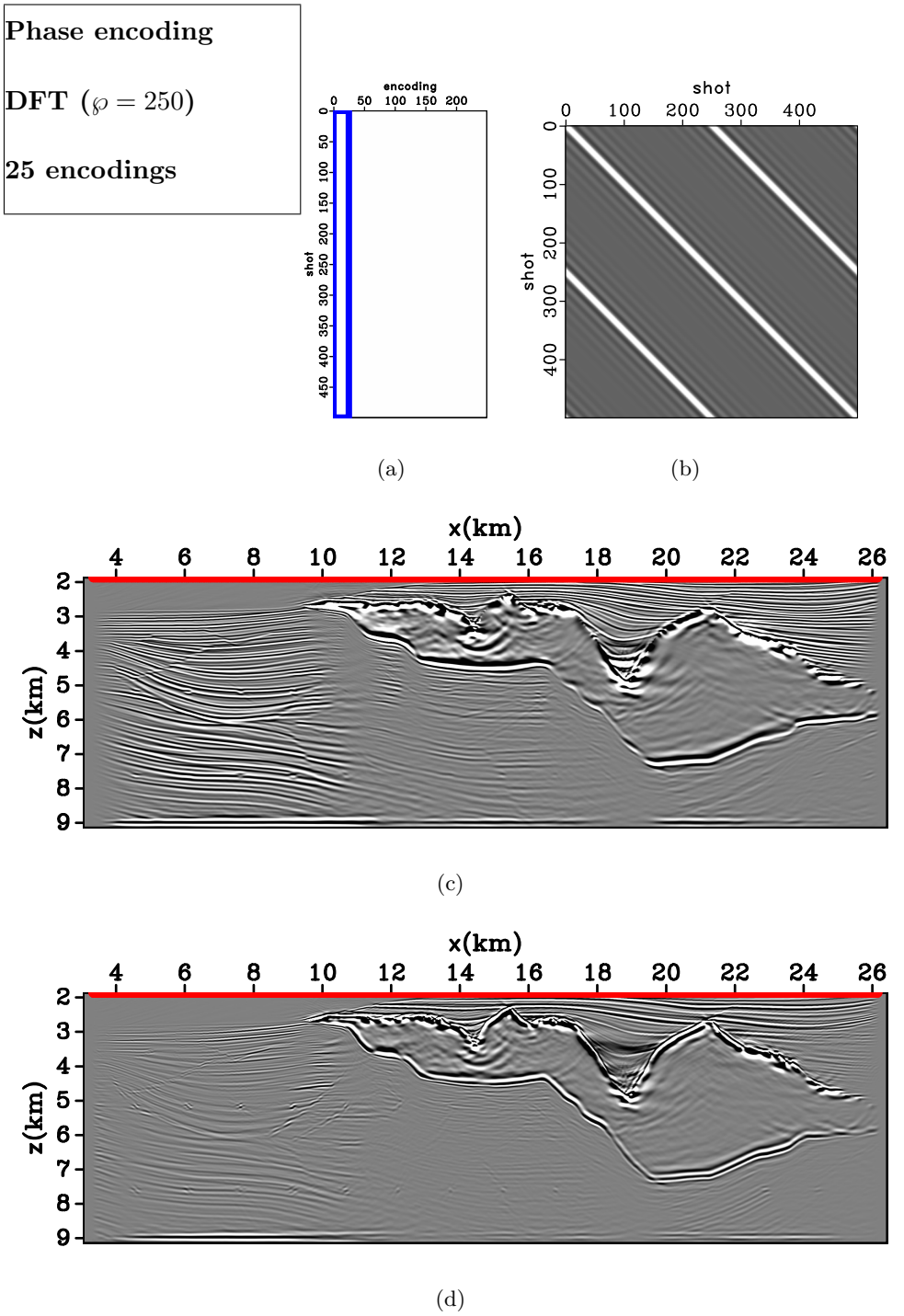


(c)



(d)

Figure 7.35. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DCS ($\wp = 250$) encoding using 251 encodings.

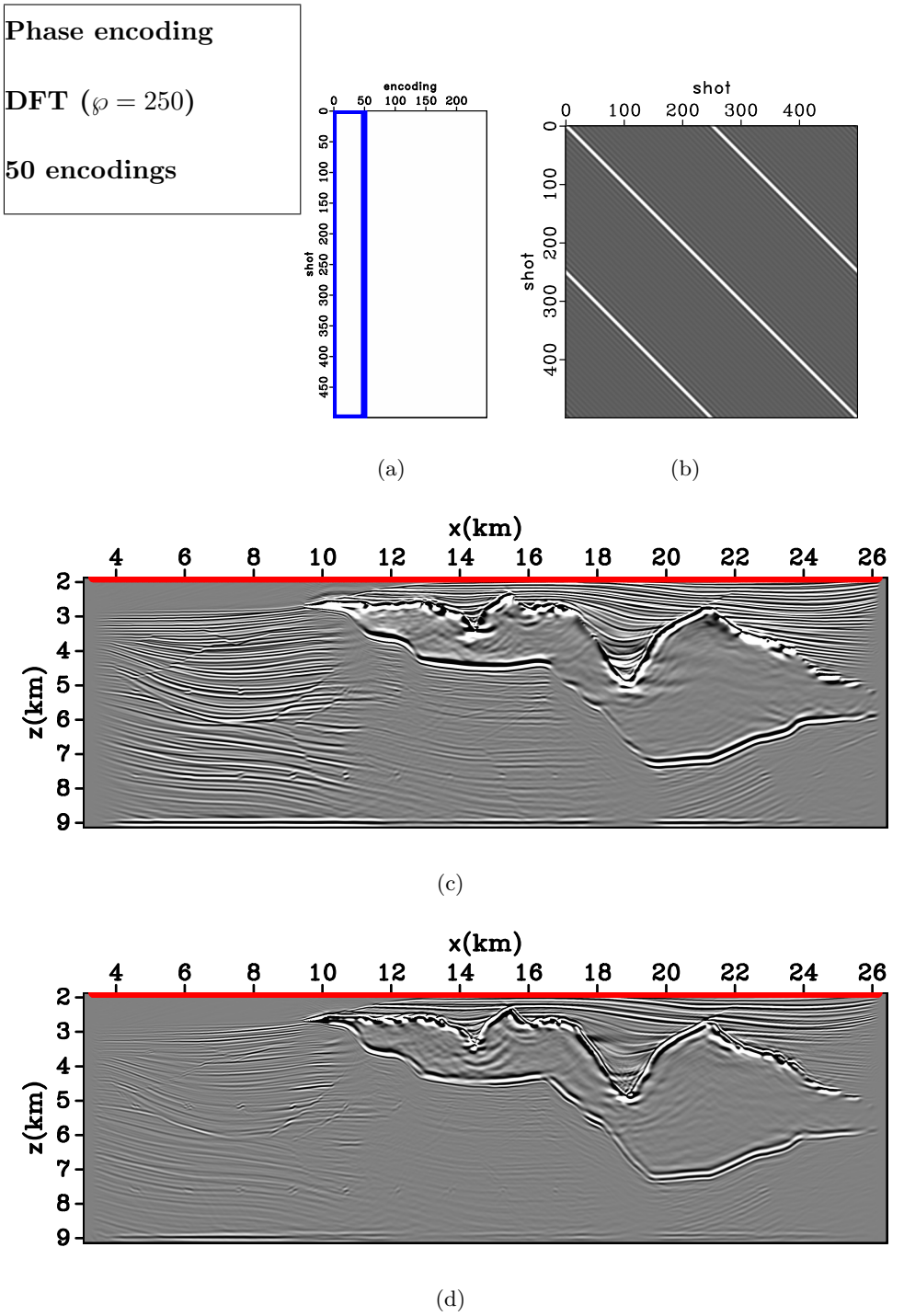


(a) (b)

(c)

(d)

Figure 7.36. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DFT ($\wp = 250$) encoding using 25 encodings.



(a) (b)

(c)

(d)

Figure 7.37. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DFT ($\wp = 250$) encoding using 50 encodings.

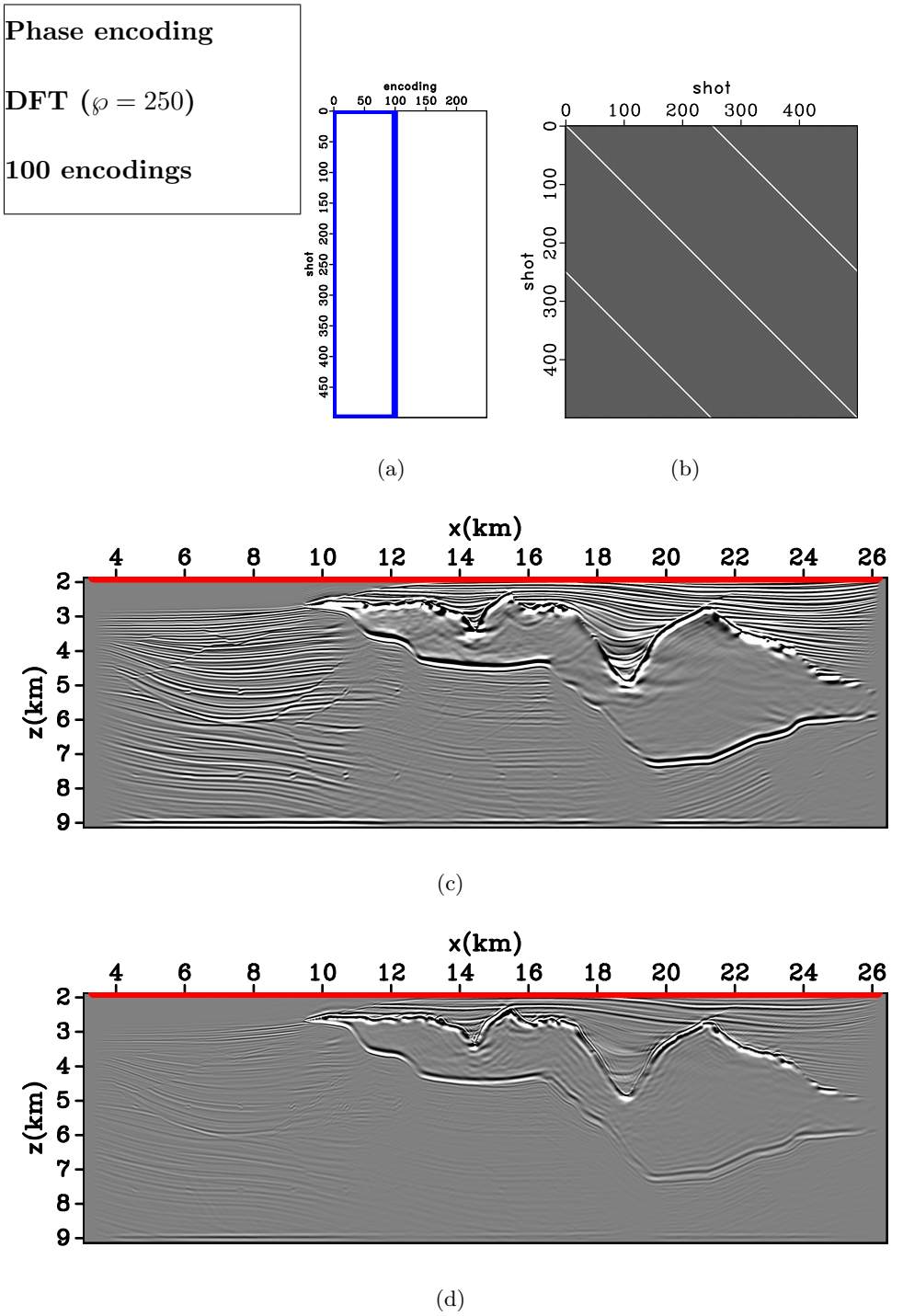


Figure 7.38. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DFT ($\wp = 250$) encoding using 100 encodings.

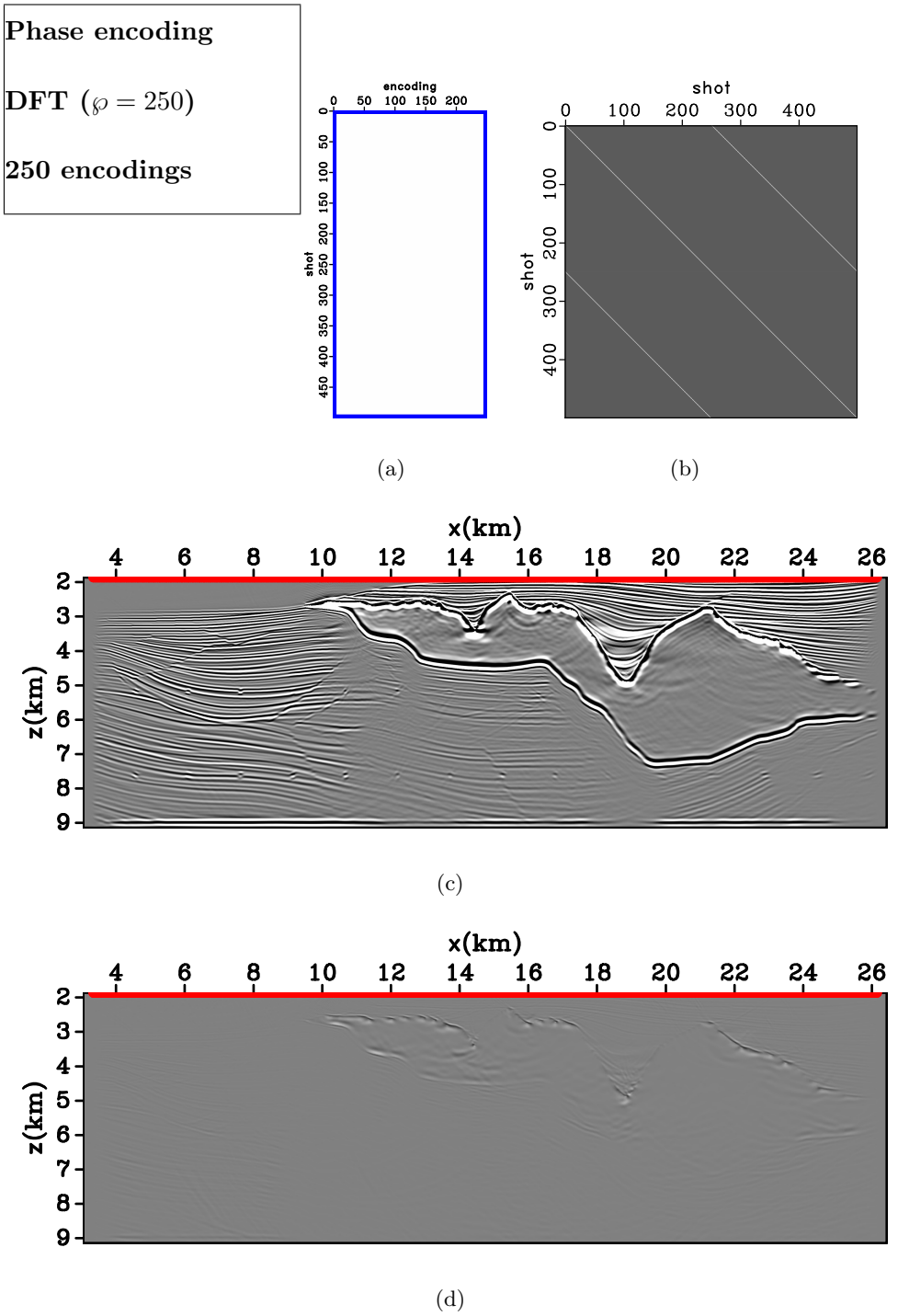
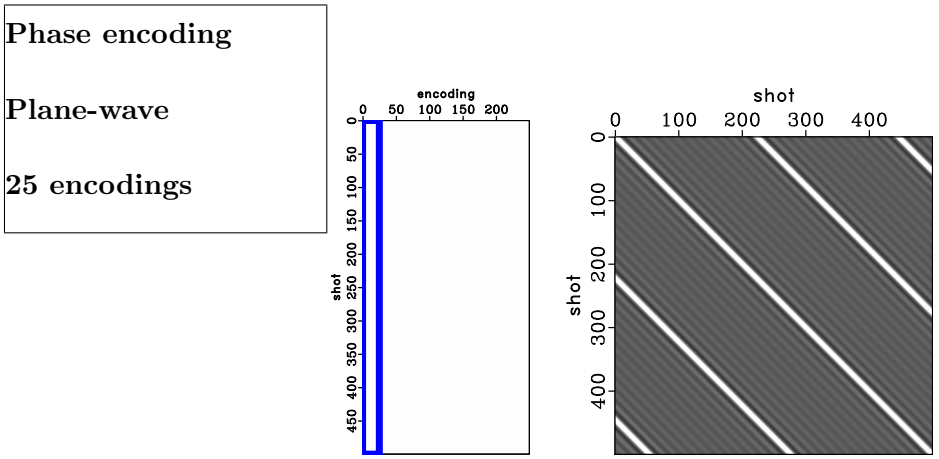
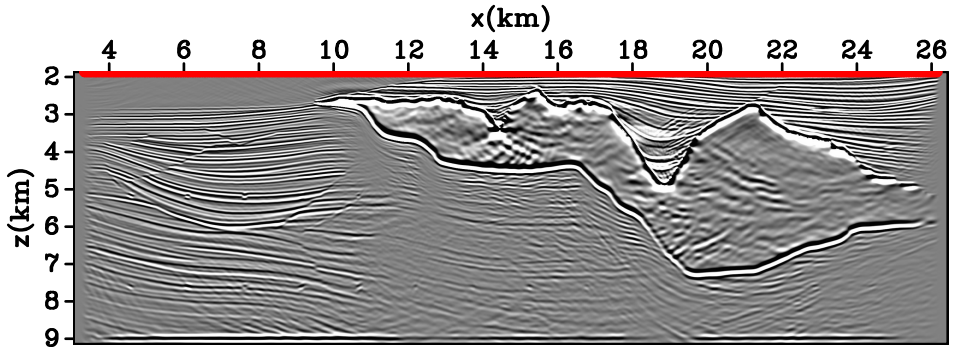


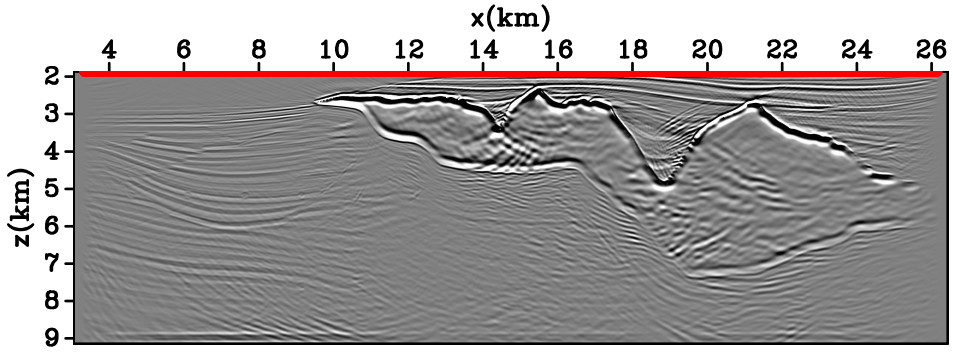
Figure 7.39. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the DFT ($\wp = 250$) encoding using 250 encodings.



(a) (b)



(c)



(d)

Figure 7.40. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Plane-wave encoding using 25 encodings.

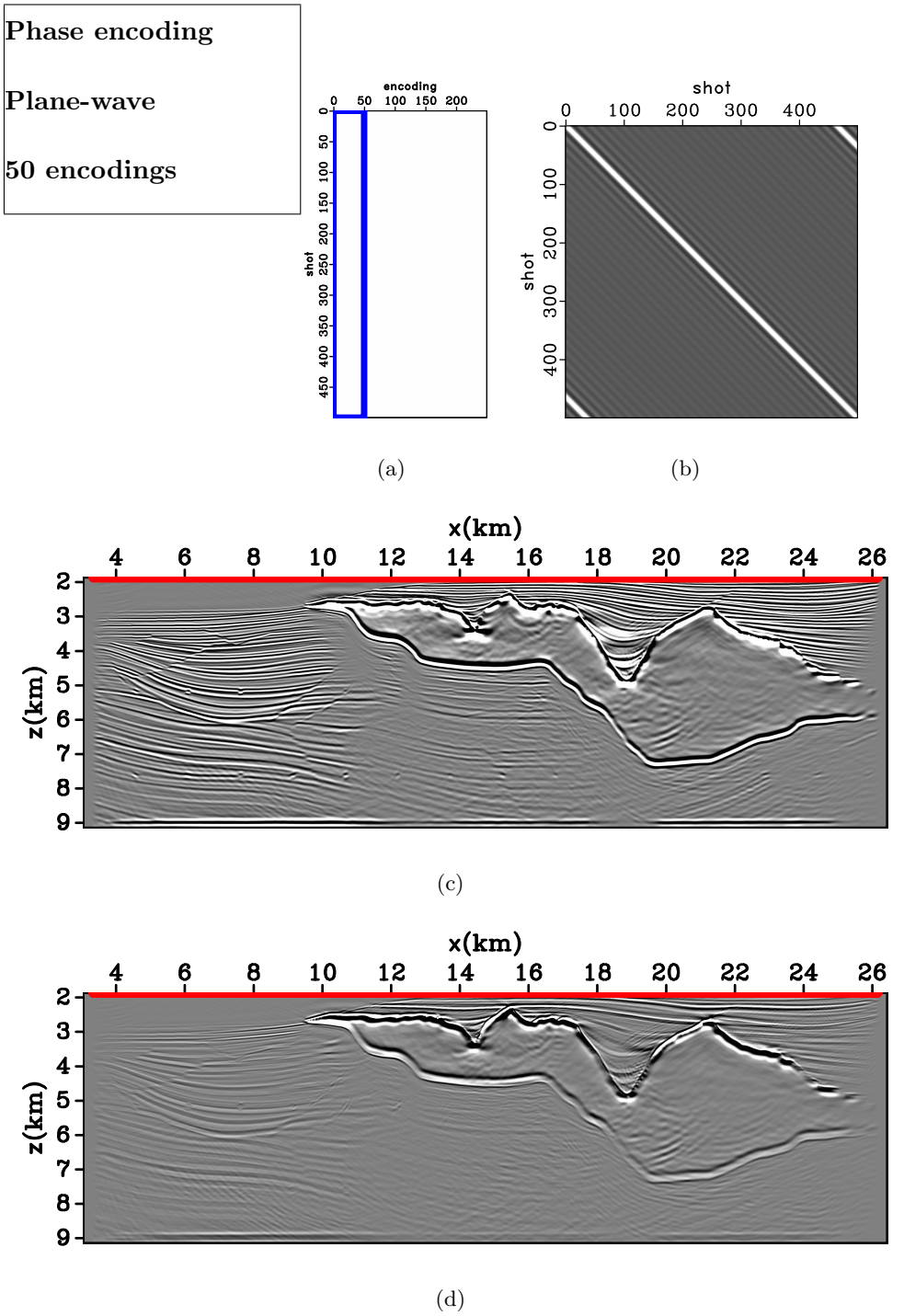
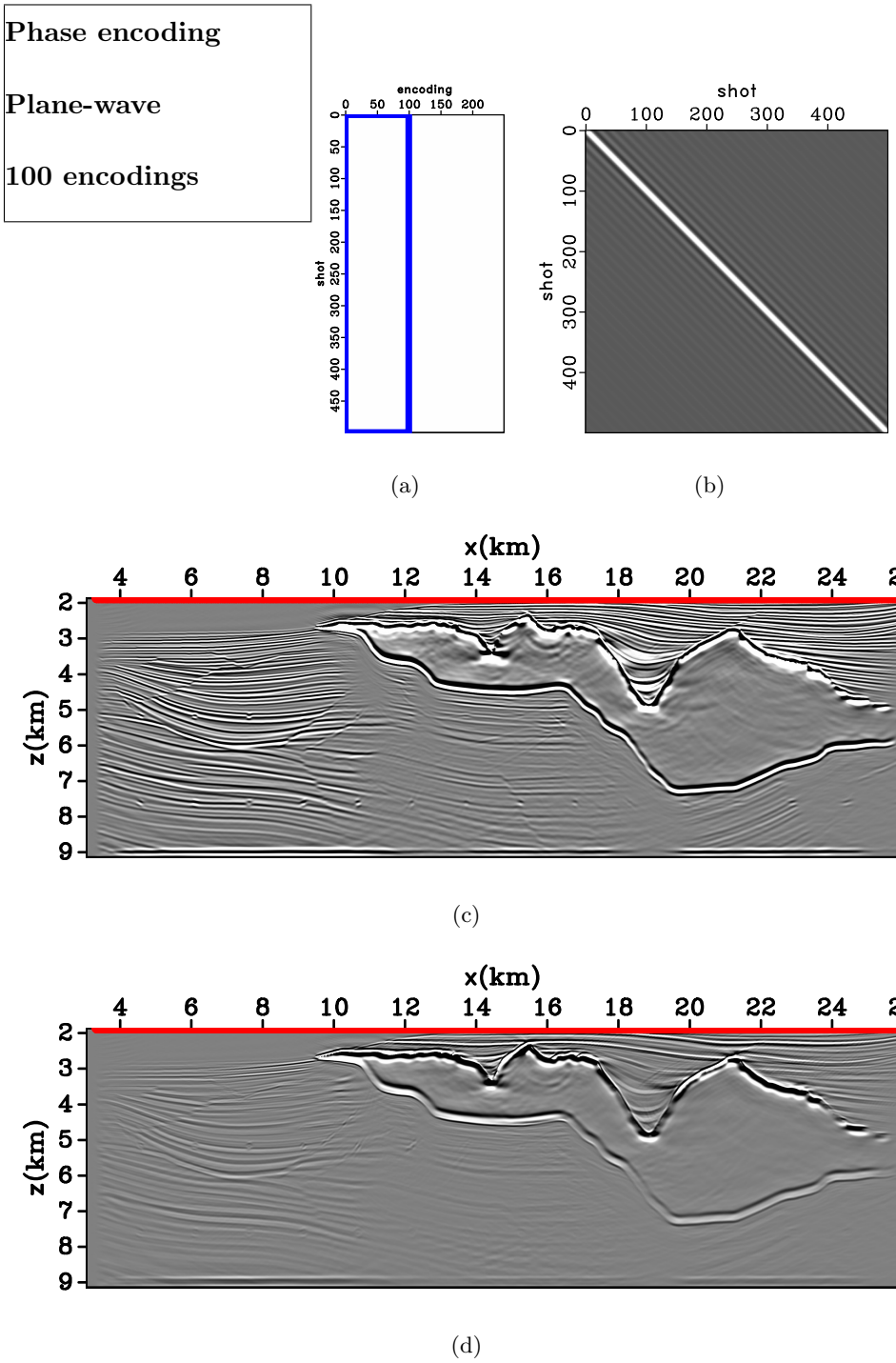


Figure 7.41. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Plane-wave encoding using 50 encodings.



(a) (b)

(c)

(d)

Figure 7.42. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Plane-wave encoding using 100 encodings.

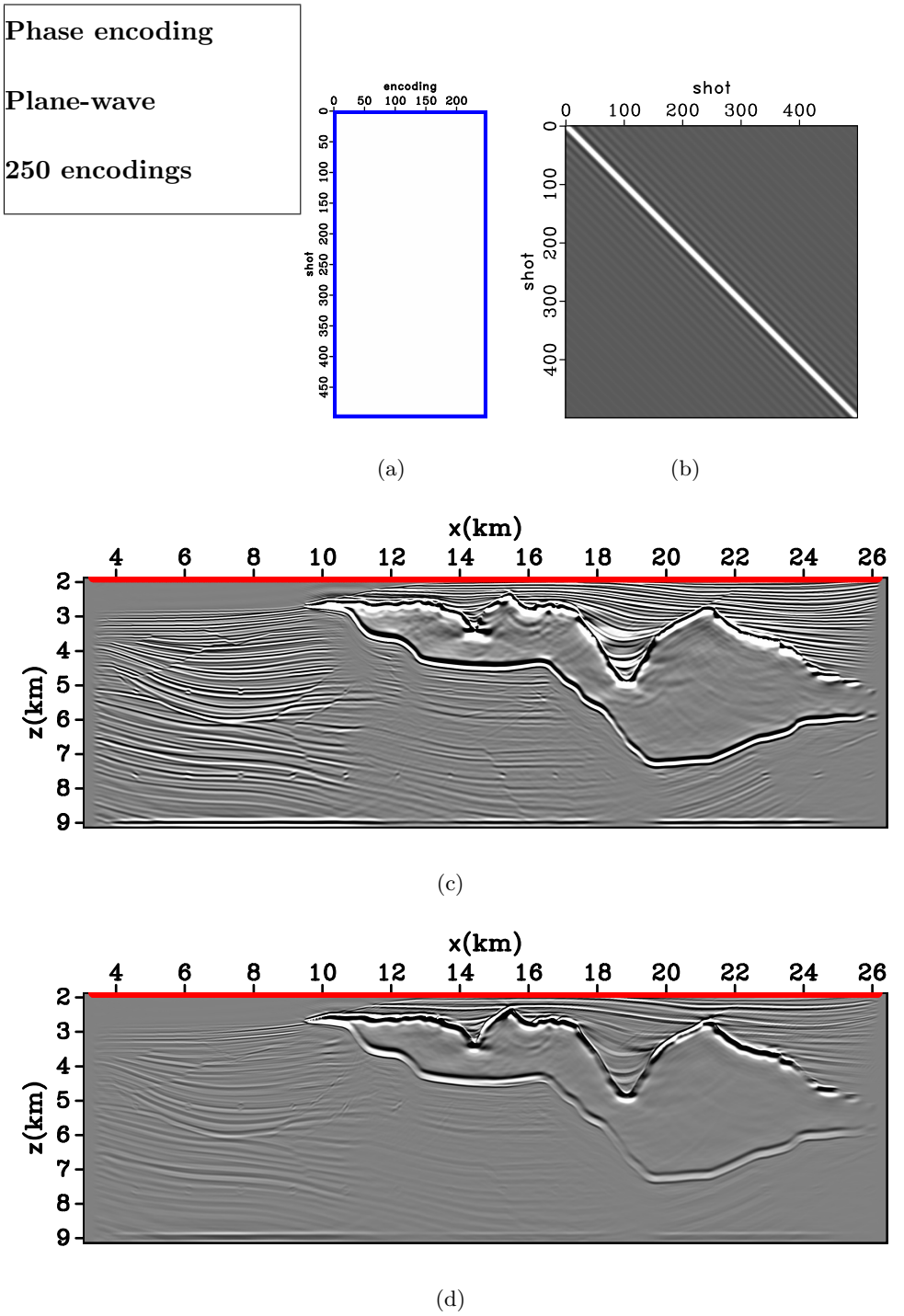
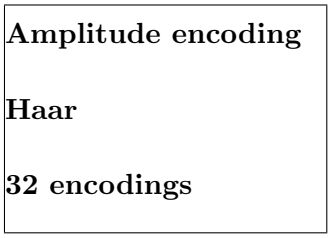
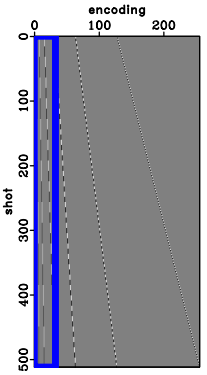
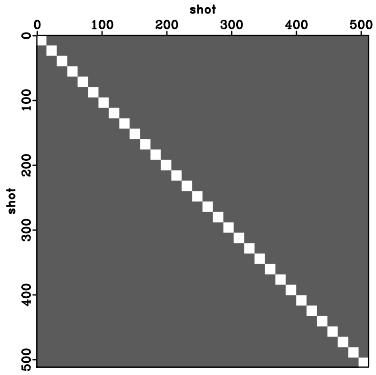


Figure 7.43. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Plane-wave encoding using 250 encodings.

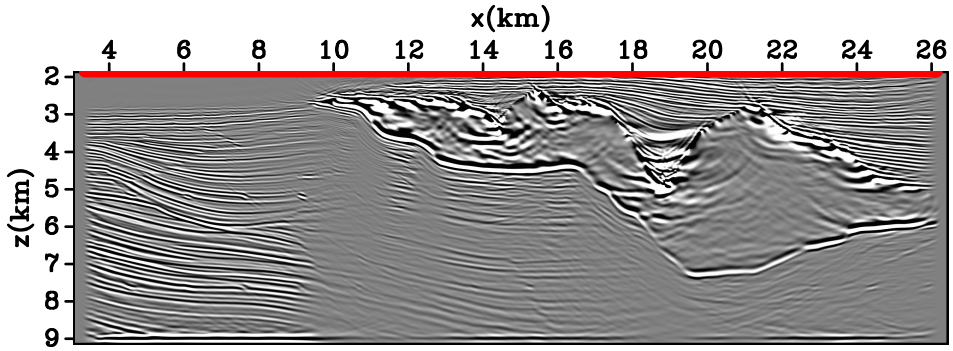




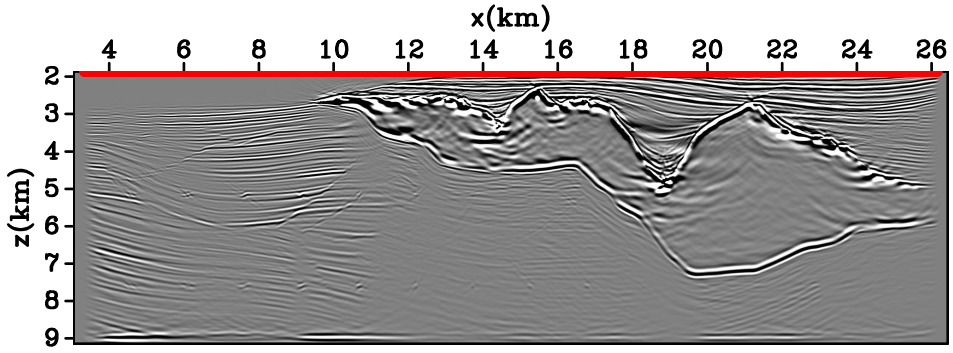
(a)



(b)

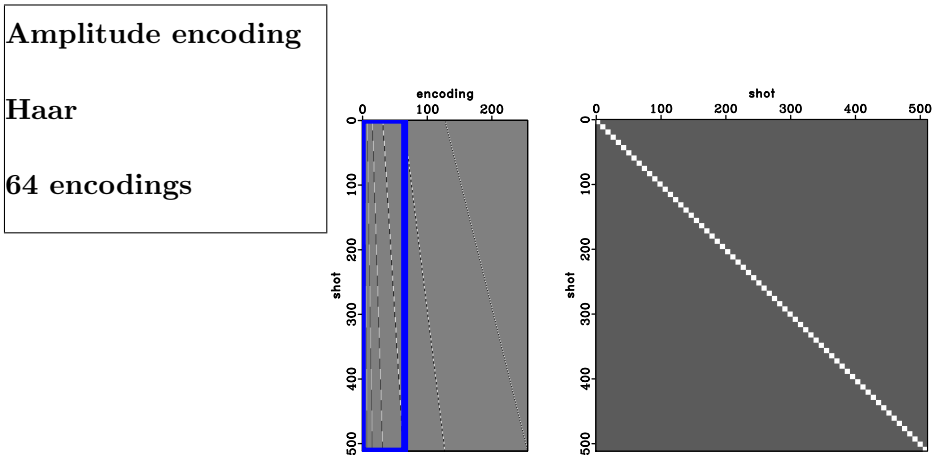


(c)

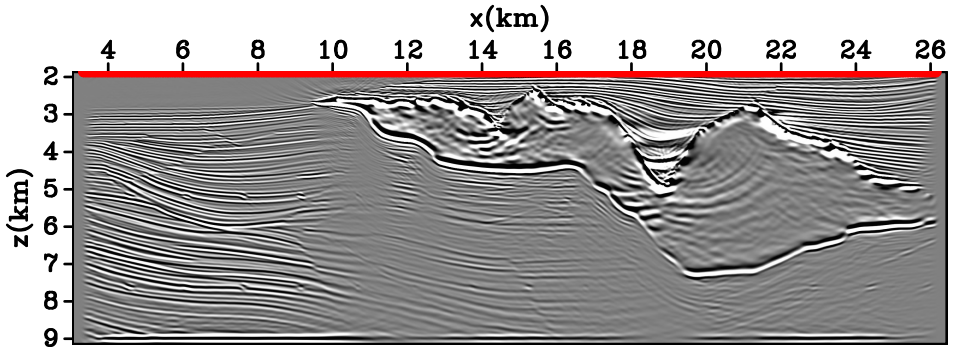


(d)

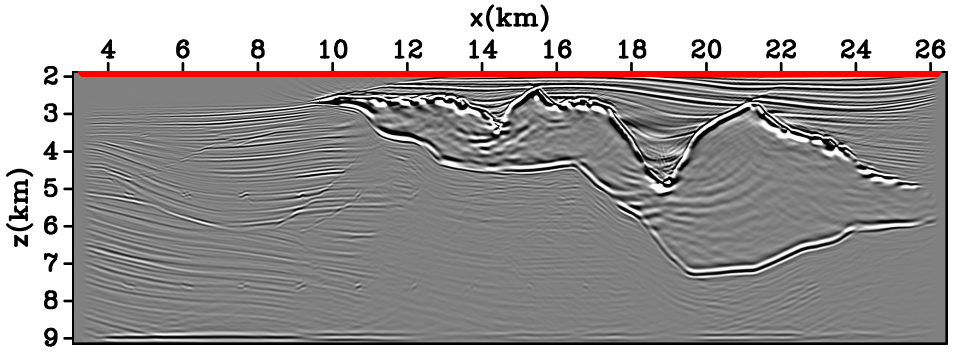
Figure 7.44. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Haar encoding using 32 encodings.



(a) (b)

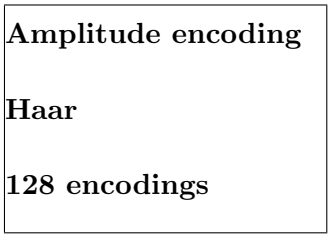


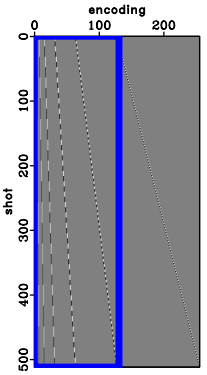
(c)



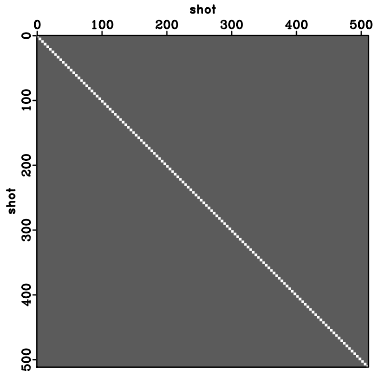
(d)

Figure 7.45. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Haar encoding using 64 encodings.

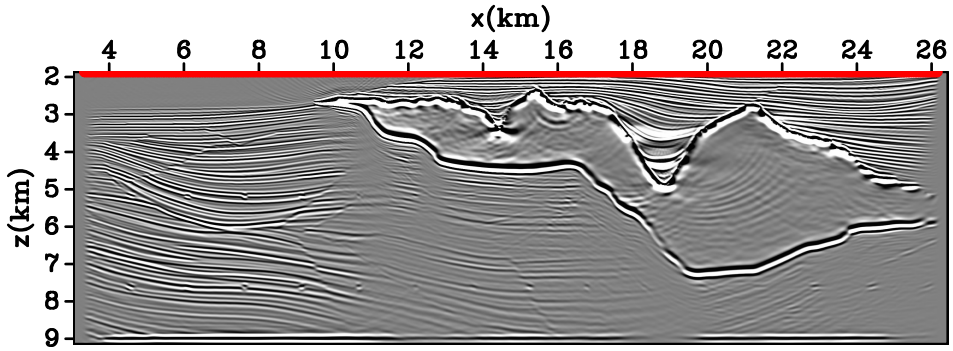




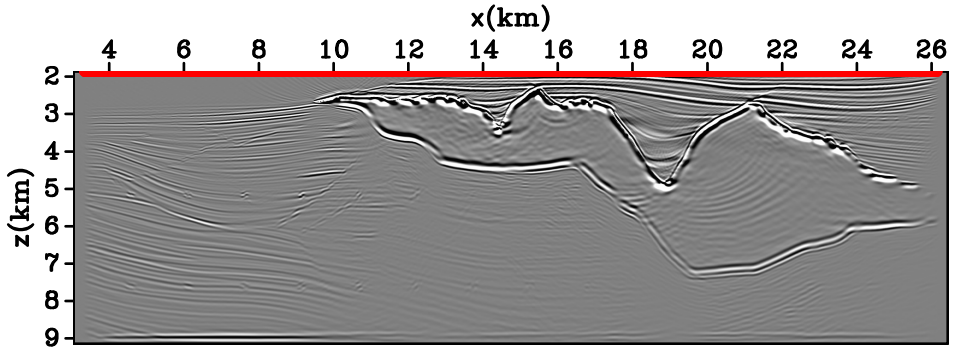
(a)



(b)

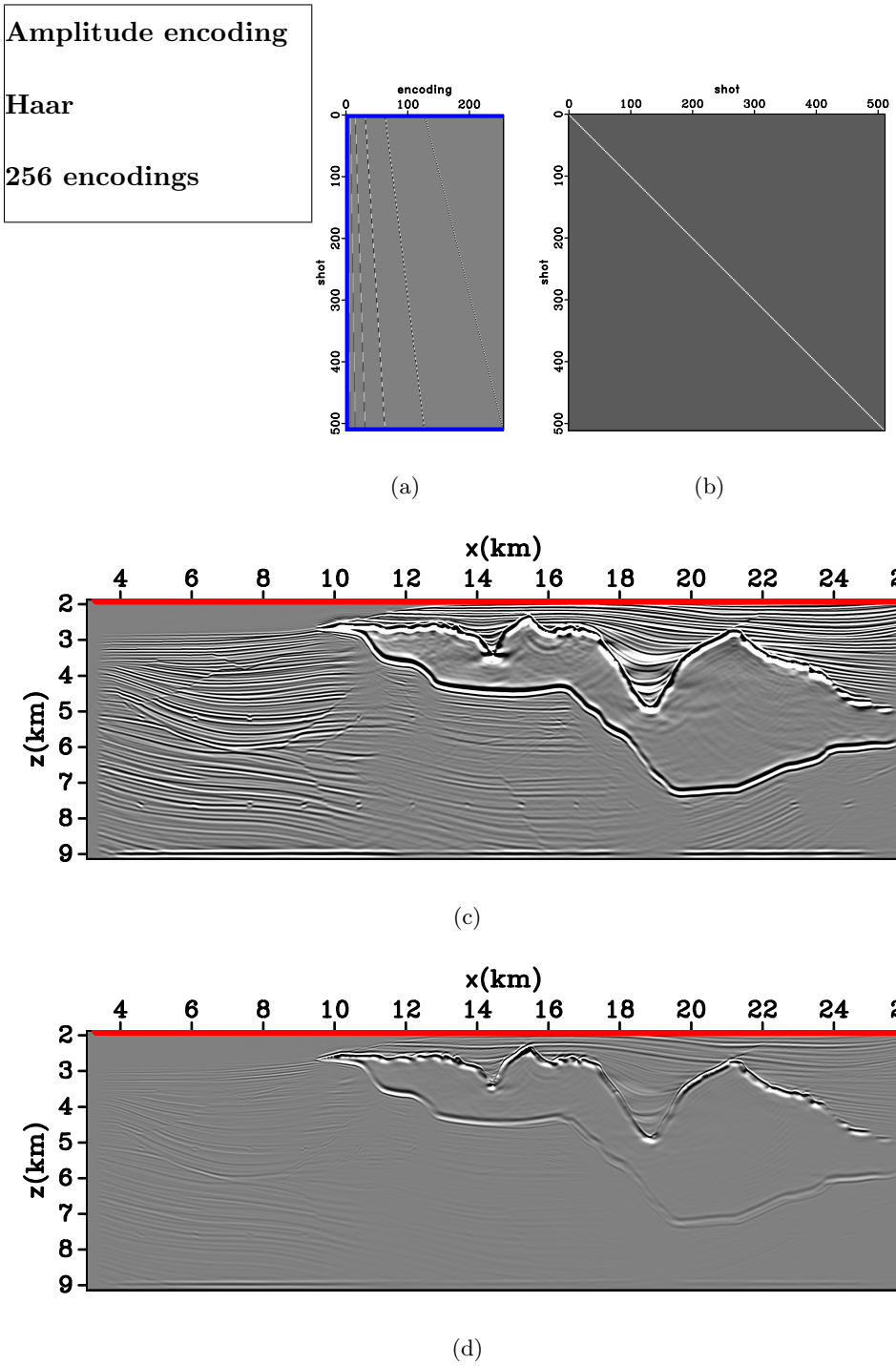


(c)



(d)

Figure 7.46. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Haar encoding using 128 encodings.

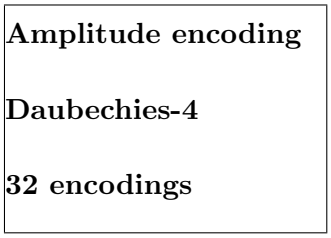


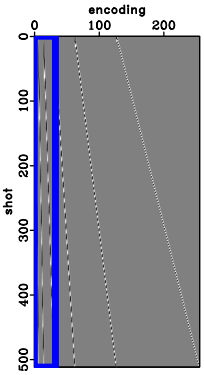
(a) (b)

(c)

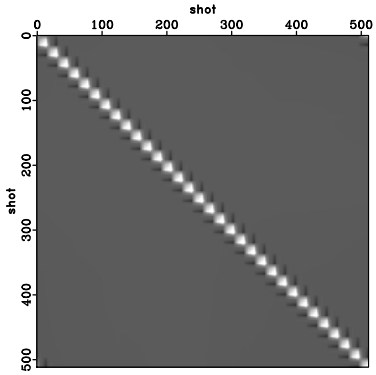
(d)

Figure 7.47. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Haar encoding using 256 encodings.

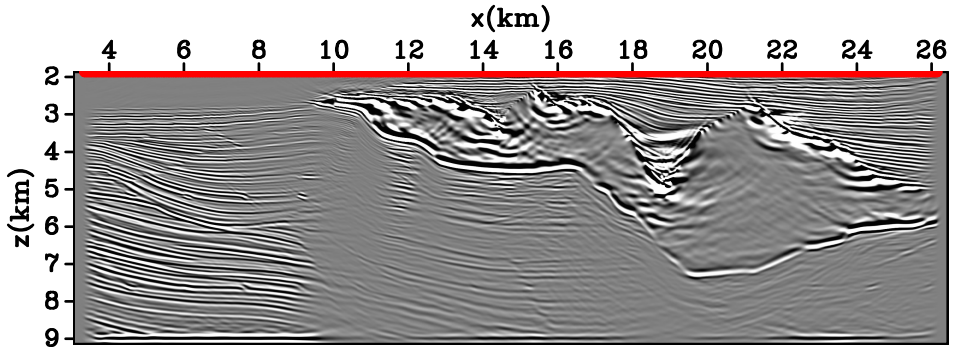




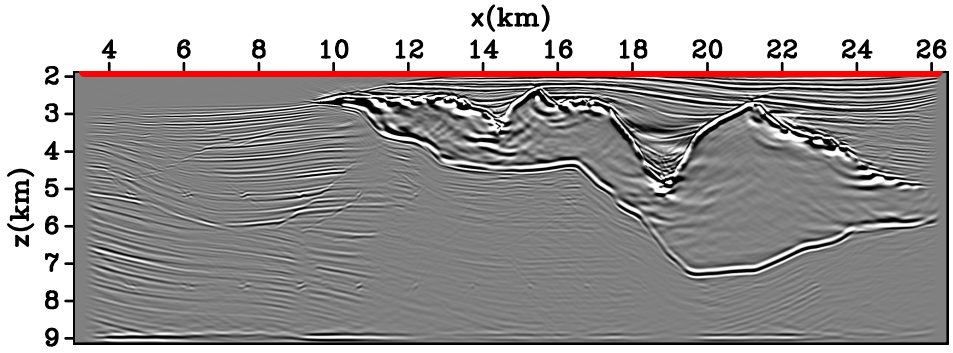
(a)



(b)

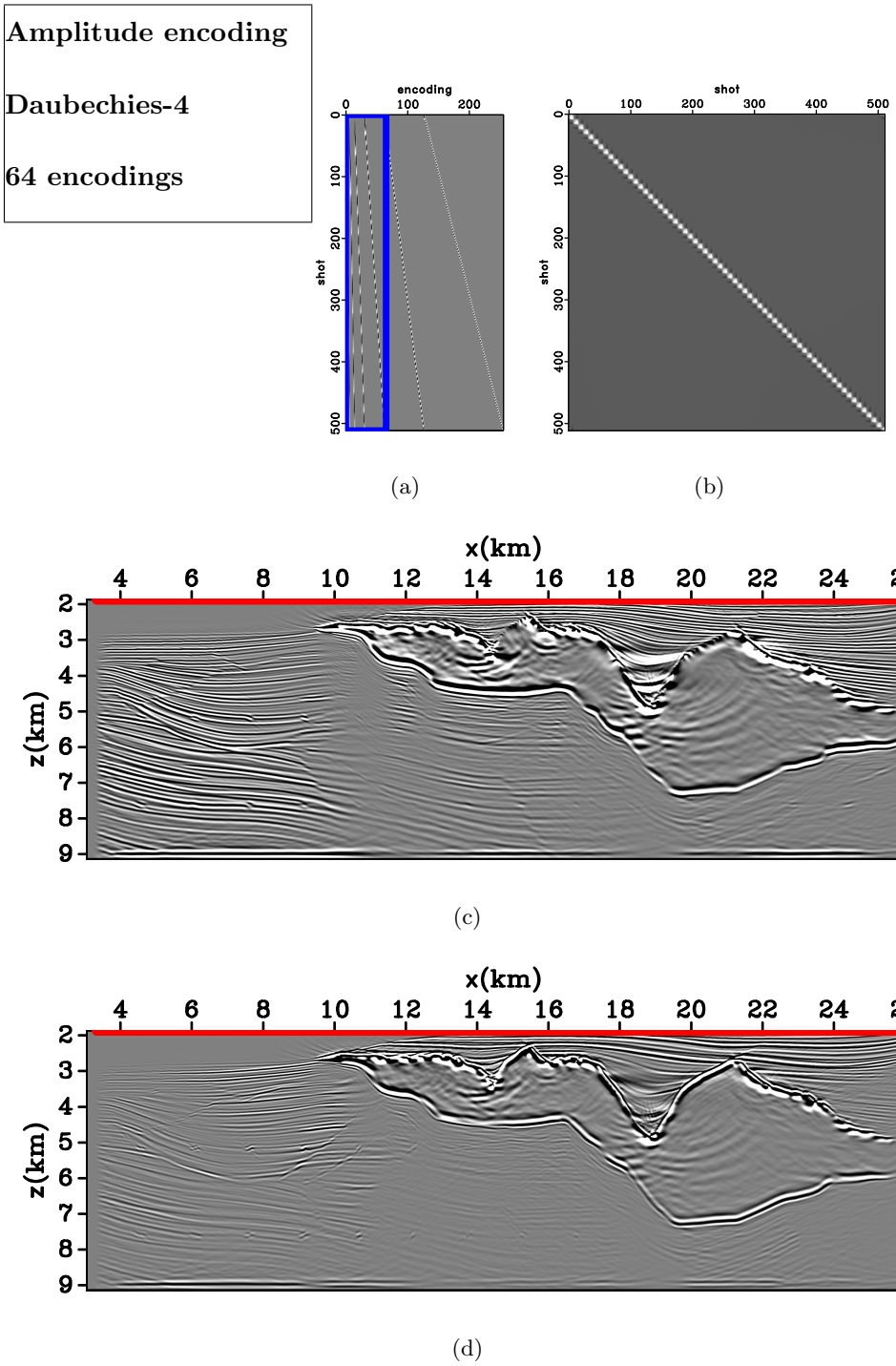


(c)



(d)

Figure 7.48. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Daubechies-4 encoding using 32 encodings.

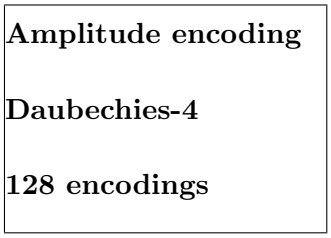


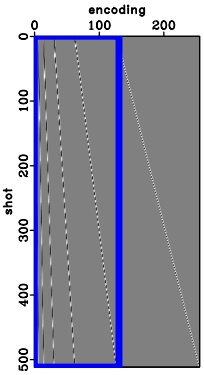
(a) (b)

(c)

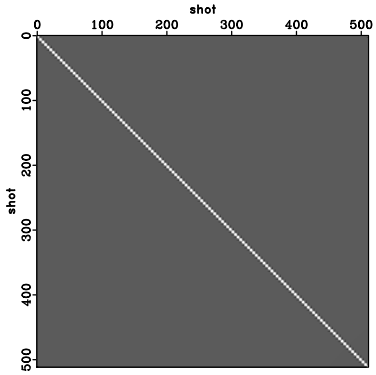
(d)

Figure 7.49. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Daubechies-4 encoding using 64 encodings.

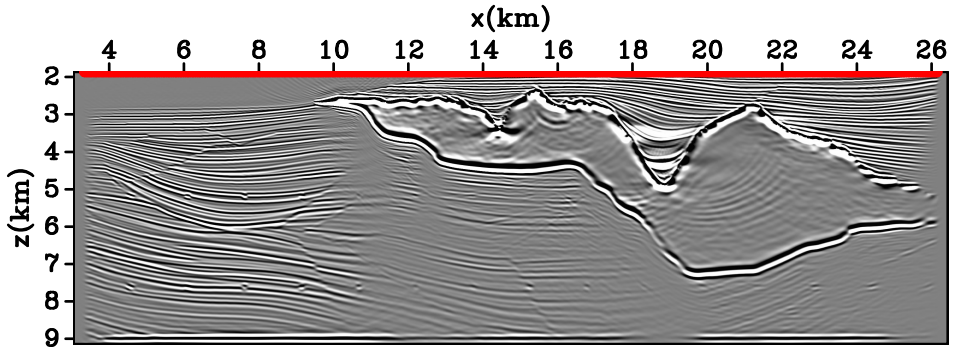




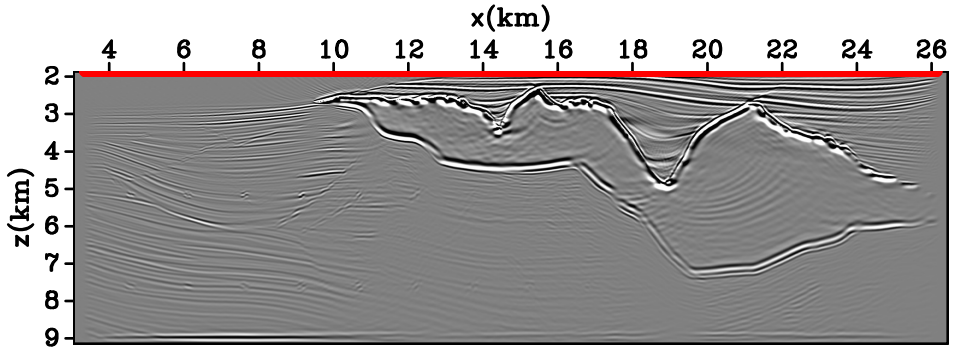
(a)



(b)

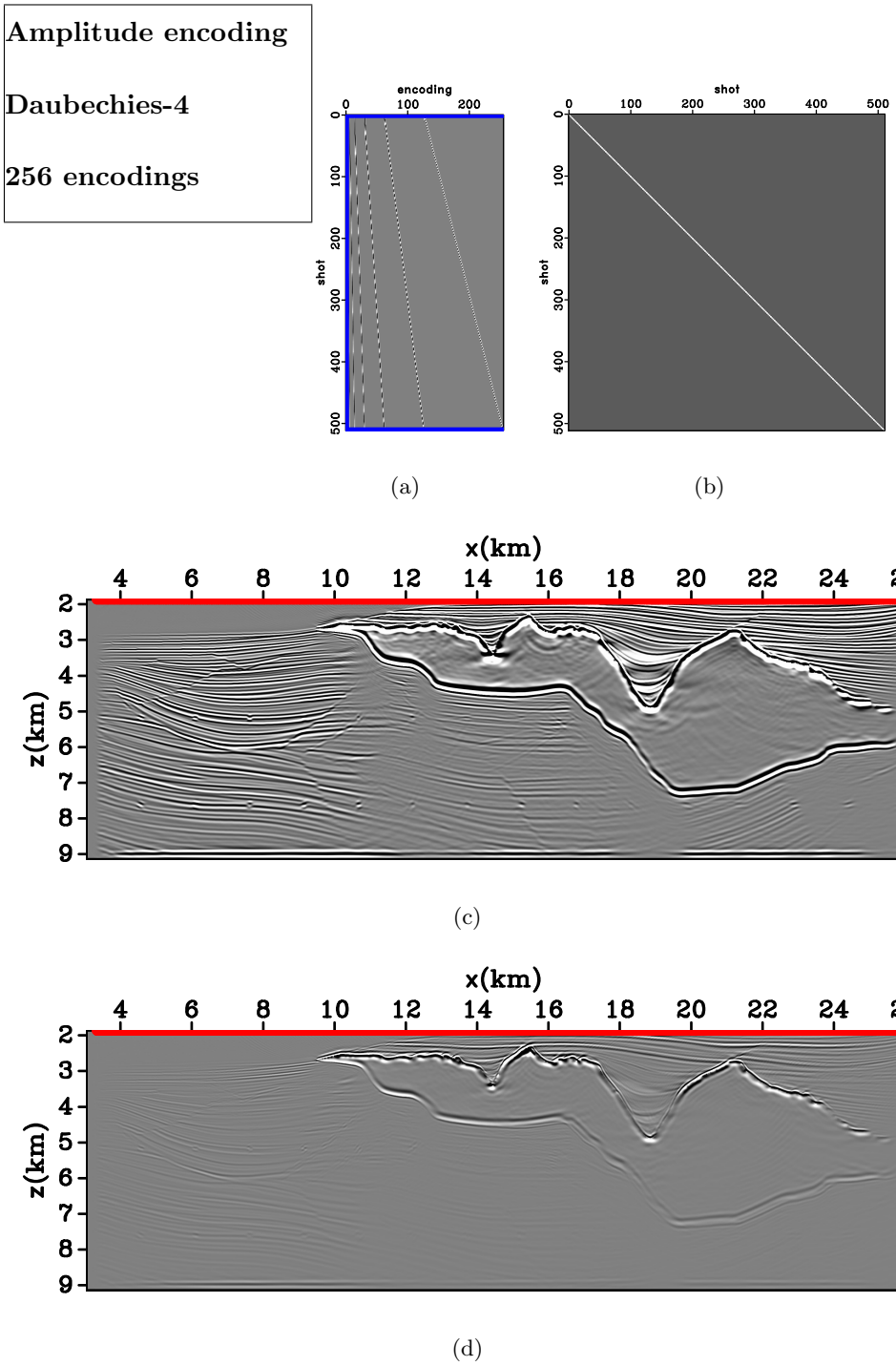


(c)



(d)

Figure 7.50. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Daubechies-4 encoding using 128 encodings.

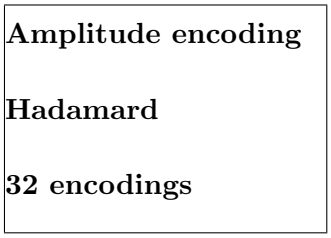


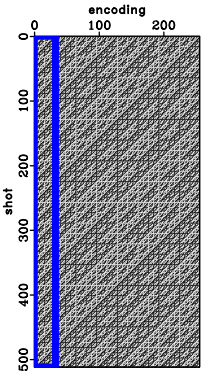
(a) (b)

(c)

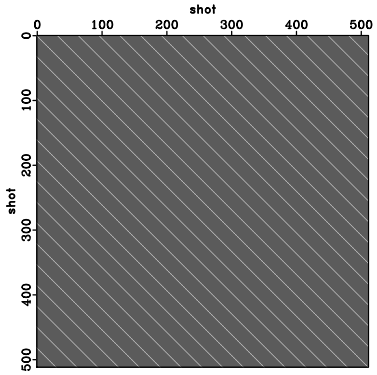
(d)

Figure 7.51. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Daubechies-4 encoding using 256 encodings.

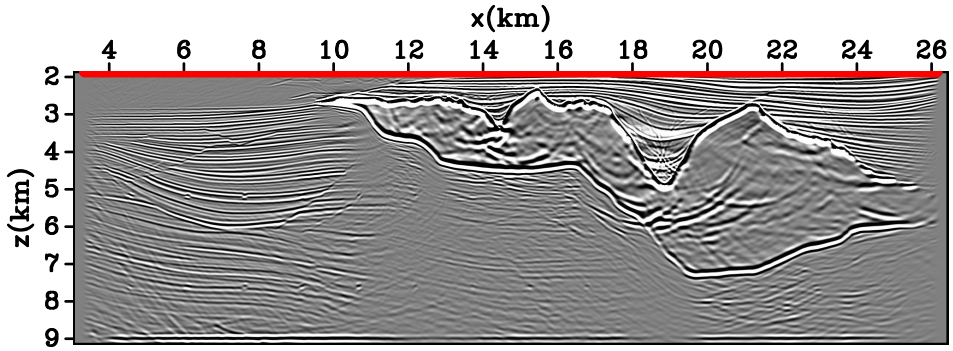




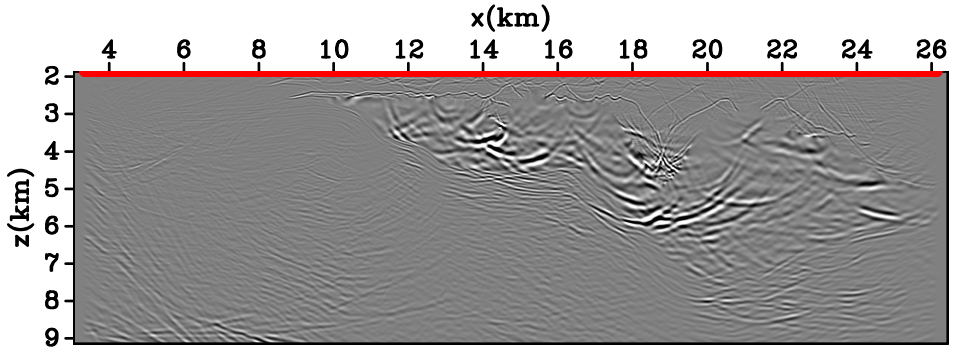
(a)



(b)

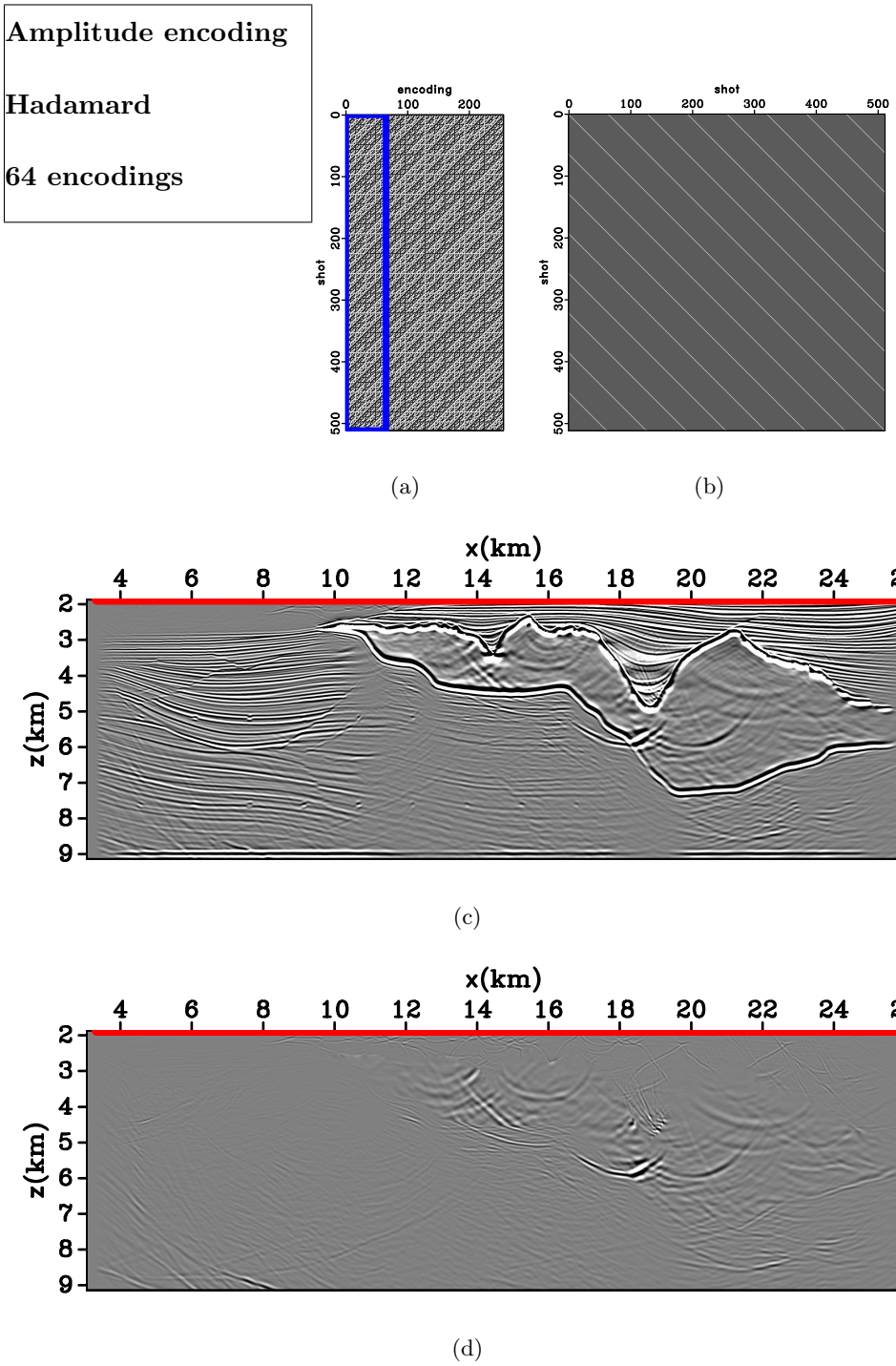


(c)



(d)

Figure 7.52. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hadamard encoding using 32 encodings.

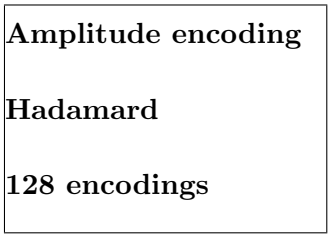


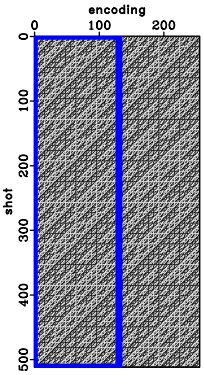
(a) (b)

(c)

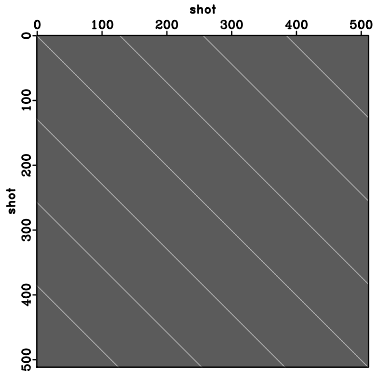
(d)

Figure 7.53. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hadamard encoding using 64 encodings.

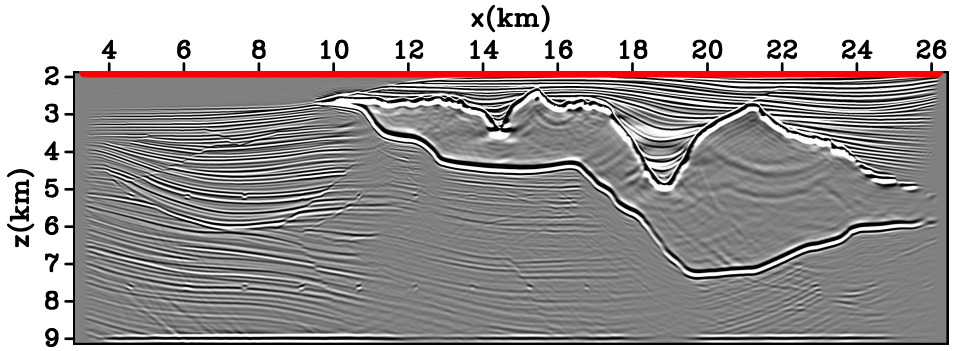




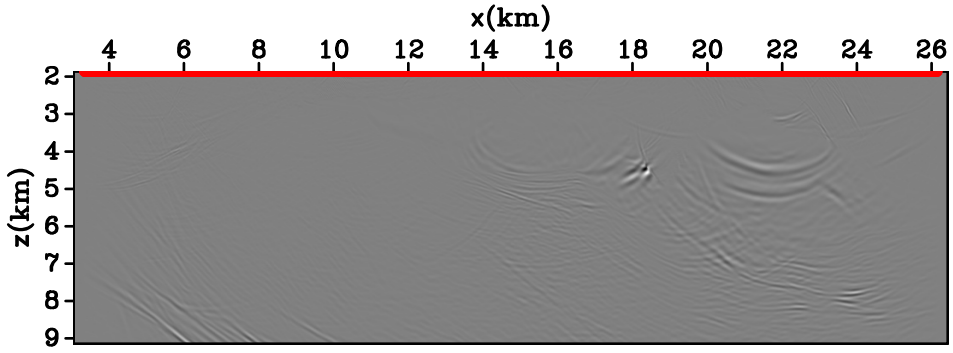
(a)



(b)



(c)



(d)

Figure 7.54. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hadamard encoding using 128 encodings.

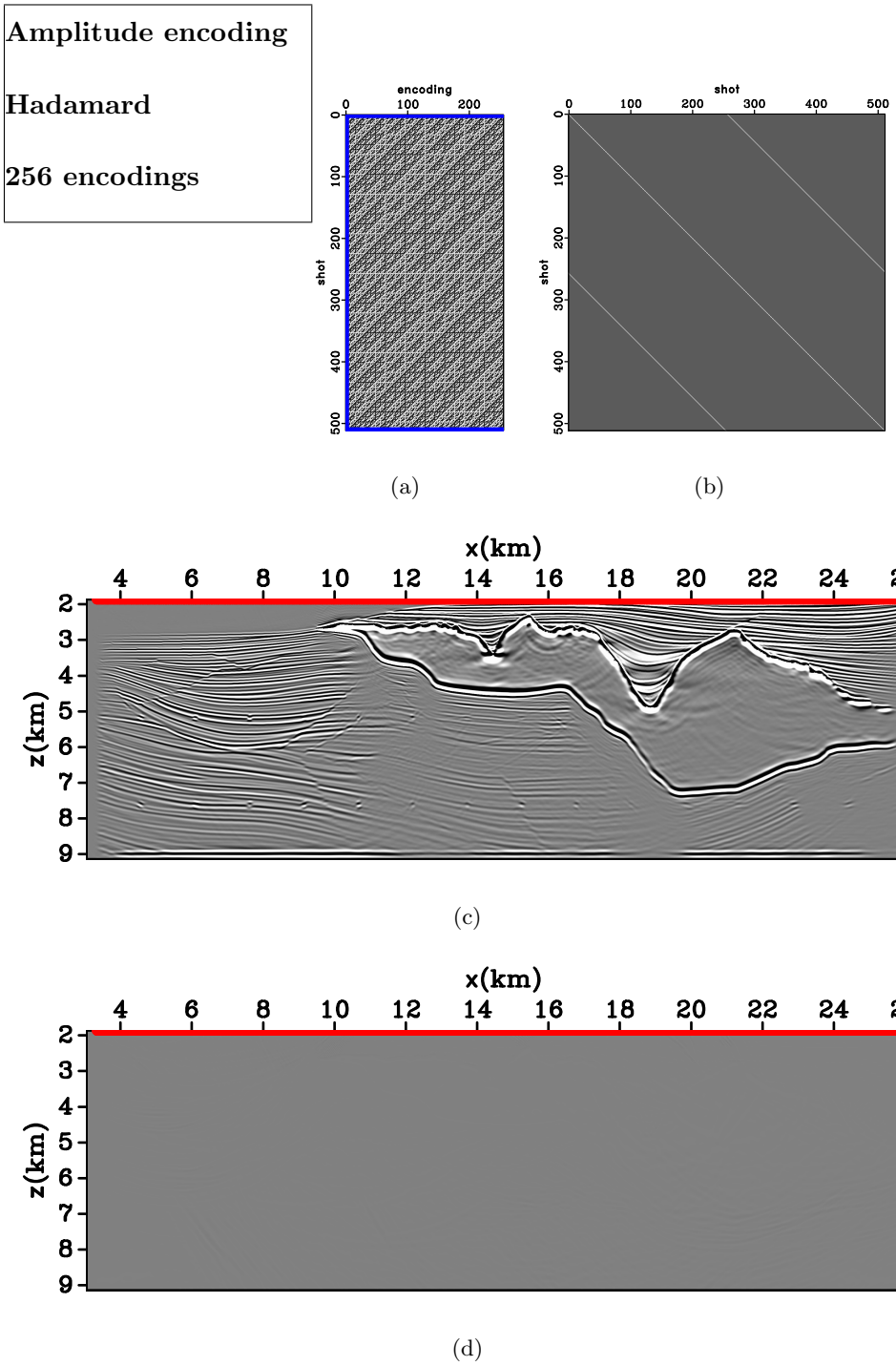


Figure 7.55. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Hadamard encoding using 256 encodings.

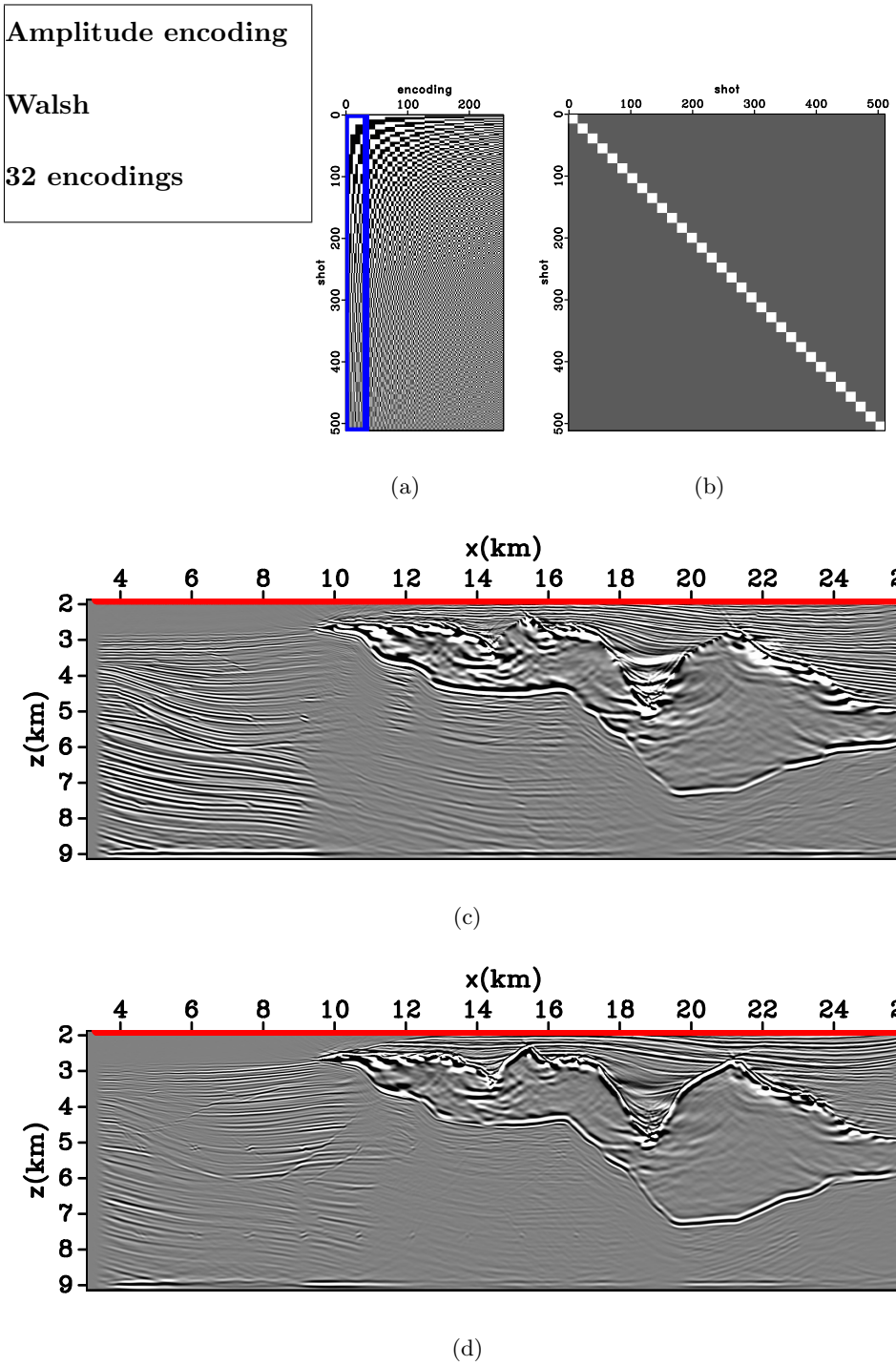
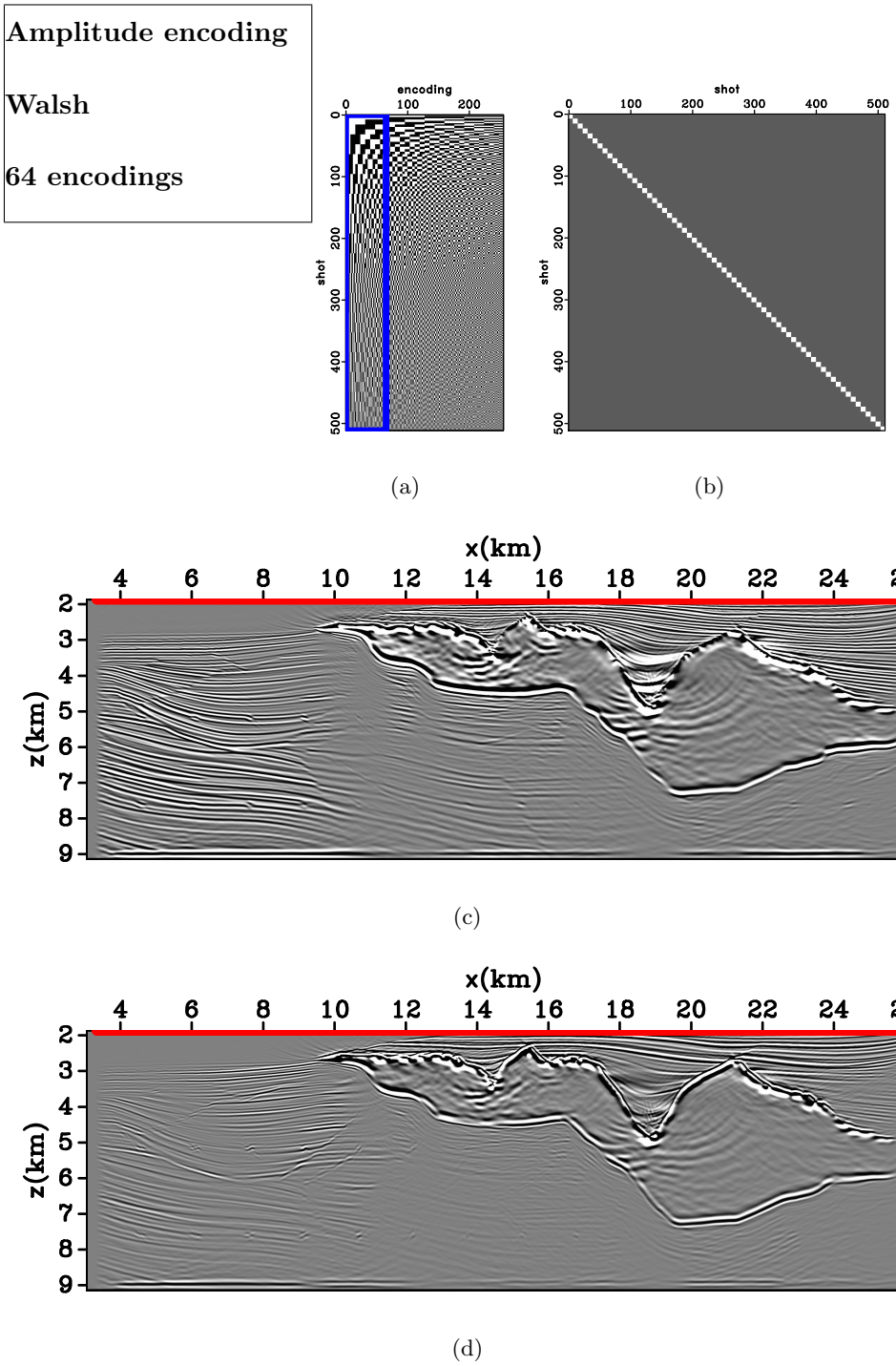


Figure 7.56. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Walsh encoding using 32 encodings.



(a) (b)

(c)

(d)

Figure 7.57. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Walsh encoding using 64 encodings.

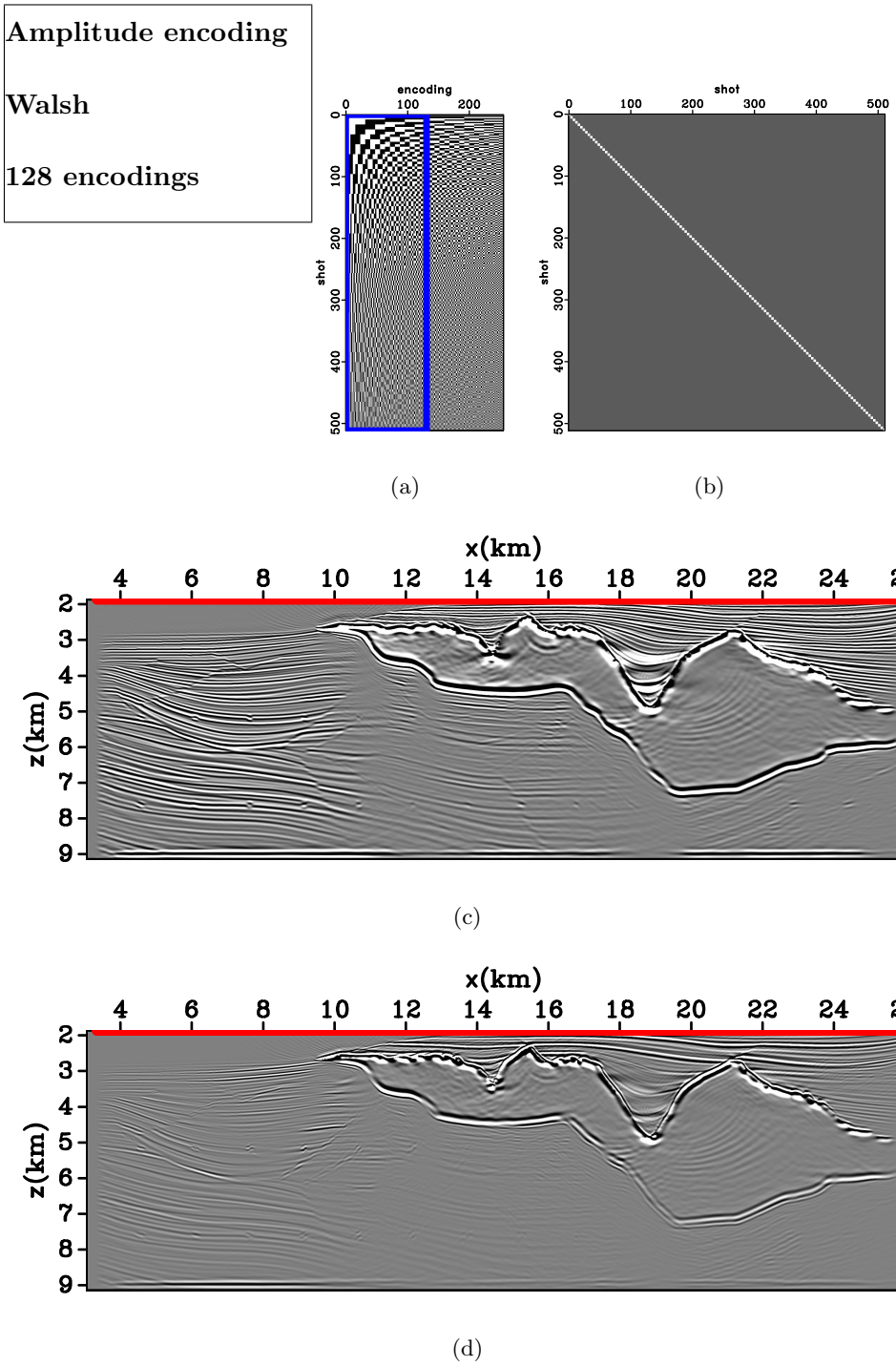
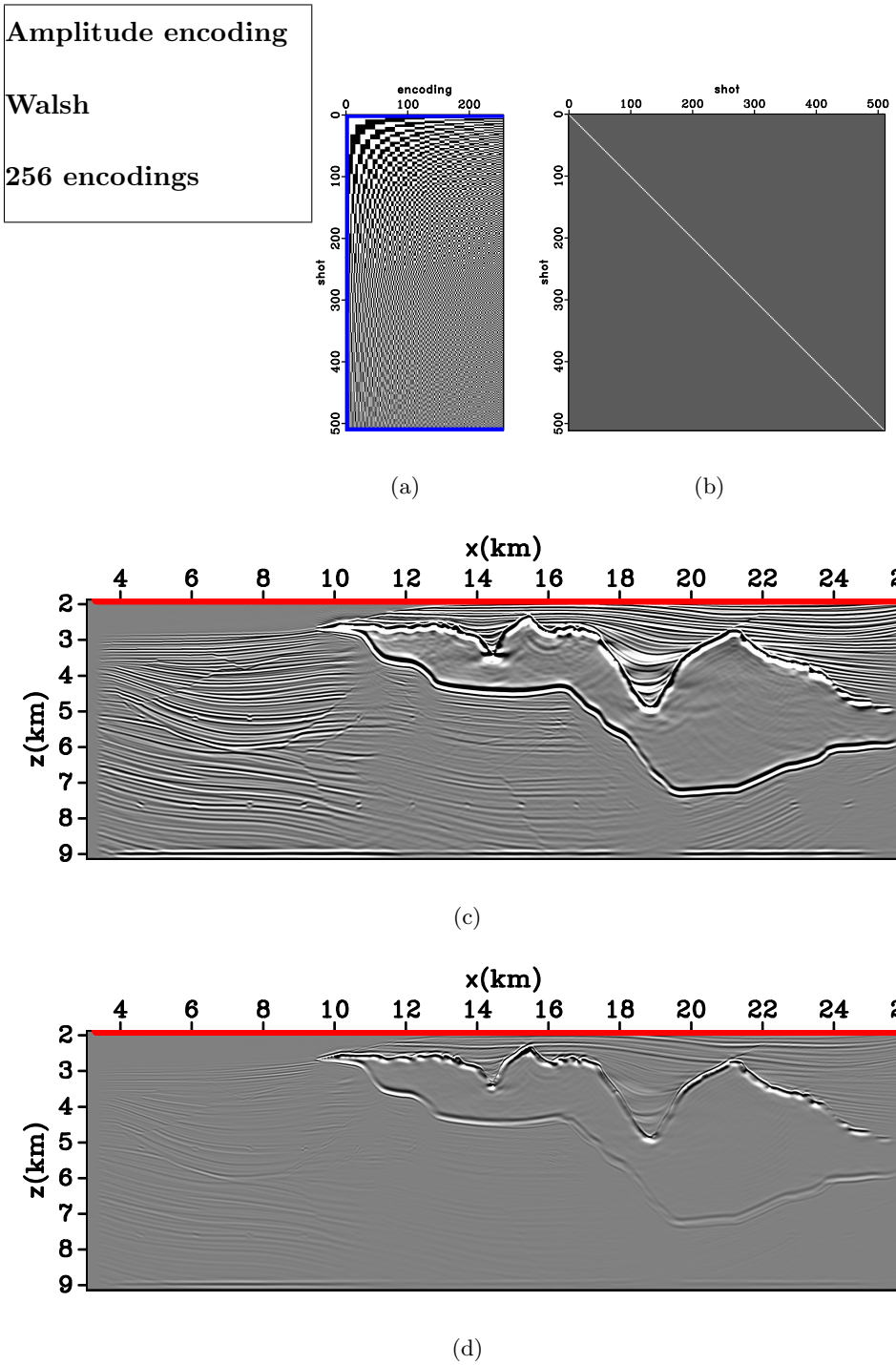


Figure 7.58. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Walsh encoding using 128 encodings.



(a) (b)

(c)

(d)

Figure 7.59. The (a) encoding matrix (highlighted to show used encodings), (b) real-part of the cross-talk matrix (imaginary part is zero), (c) final stacked image, and (d) amplified difference image with SRM for the Walsh encoding using 256 encodings.

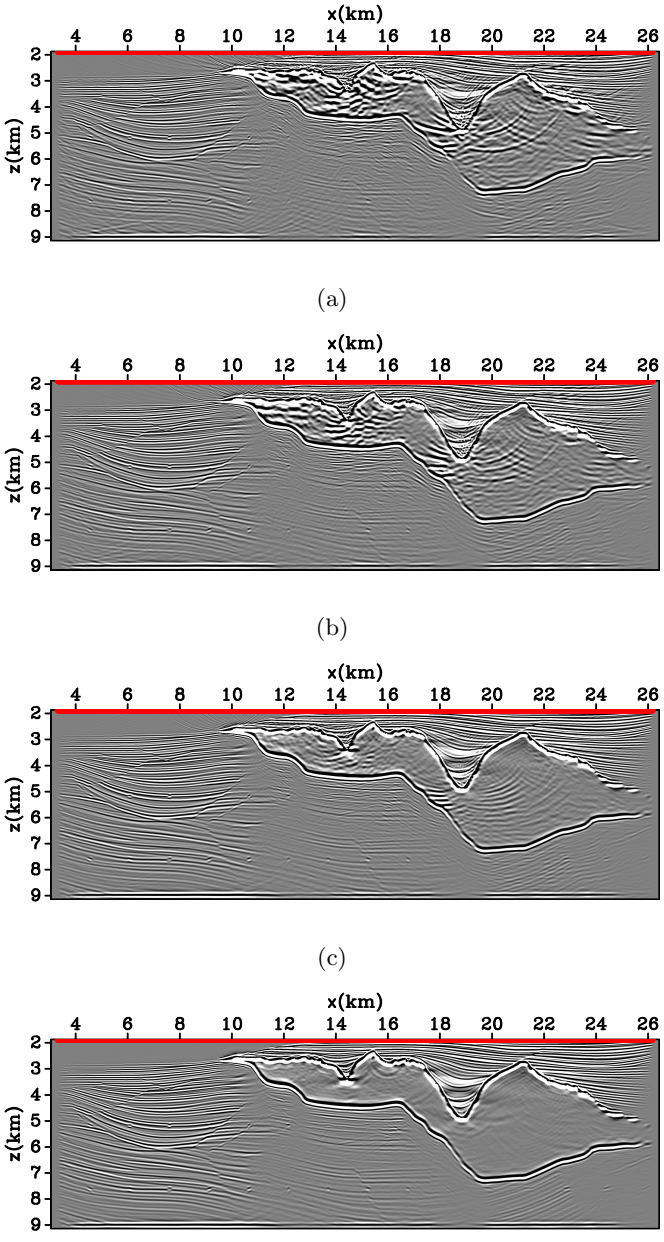


Figure 7.60. Images from conventional SRM decimated using: (a) every 25th shot, (b) every 10th shot, (c) every 5th shot, (d) and every other shot. The quality of the image quickly degrades as the decimation factor increases due to spatial aliasing.

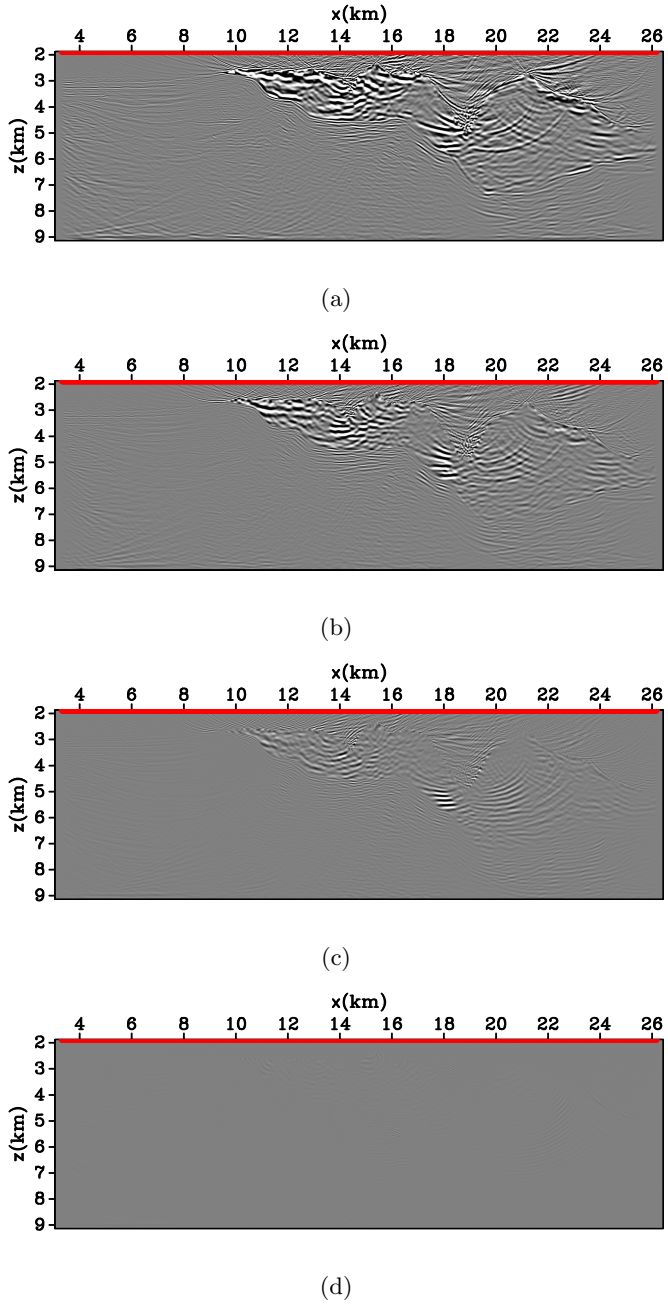


Figure 7.61. Difference images for decimated SRM using (a) every 20th shot-record, (b) every 10th shot, (c) every 5th shot, and (d) every other shot. As the number of shots used increases (i.e. decimation factor decreases) the difference between the decimated image and shot-record migration decreases.