

Projected gradient descent methods for simultaneous-source seismic data
processing

by

Rongzhi Lin

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Geophysics

Department of Physics
University of Alberta

©Rongzhi Lin, 2022

Abstract

Simultaneous-source acquisition is a seismic data acquisition technology that has become quite popular in recent years due to its economic advantages. Contrary to conventional seismic acquisition, where one records the seismic response of only one source at a time, in simultaneous source acquisition, an array of receivers record the response of more than one source. The latter leads to a saving in acquisition time, but it creates new problems in subsequent data processing stages where each seismic record must correspond to the response of one single source. The basic idea for simultaneous source data processing is to separate the sources and thereby estimate the responses one would have acquired via a conventional seismic data acquisition. Then one can adopt a traditional seismic workflow to process and invert the seismic data.

This thesis focuses on developing inversion schemes for separating simultaneous-source data. I pay particular attention to strategies based on the Projected Gradient Descent (PGD) method with a projection synthesized via robust denoising algorithms. First, I propose adopting a robust and sparse Radon transform to define a coherence pass projection operator to guarantee solutions that honour simultaneous source records. I show that a critical improvement in convergence is attainable when the coherence pass projection originates from a robust and sparse Radon transform. The latter is a consequence of having an iterative source separation algorithm that applies intense denoising to erratic blending noise in its initial iterations.

In addition, I also propose an inversion scheme for simultaneous-source data separation based on a robust low-rank approximation algorithm. A robust Multichannel Singular Spectrum

Analysis (MSSA) filtering is adopted as the projection operator to suppress source interferences in the frequency-space domain. The MSSA method is reformulated as a robust optimization problem that includes a low-rank Hankel matrix constraint, written as the product of two matrices of lower dimension obtained by the bifactored gradient descent (BFGD) method.

In the second part of my thesis, I explore an inversion scheme for source separation and source reconstruction that honours actual source coordinates. The proposed method adopts a projected gradient descent optimization with a reduced-rank MSSA projection operator. I propose to adopt an Interpolated-MSSA (I-MSSA) to separate and reconstruct sources in situations where the acquired simultaneous source data correspond to sources with arbitrary irregular-grid coordinates. Additionally, a faster and computational-efficient MSSA (FMSSA) algorithm was applied to speed up the method.

Preface

A version of the work in chapter 3 of this thesis has been published in a journal paper: Lin, R., and M. D. Sacchi, 2020, Separation of simultaneous sources acquired with a high blending factor via coherence pass robust Radon operators: *Geophysics*, **85**, no. 3, V269-V282.

A version of the work in chapter 4 of this thesis has been published in a journal paper: Lin, R., B. Bahia, and M. D. Sacchi, 2021, Iterative deblending of simultaneous-source seismic data via a robust singular spectrum analysis filter: *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 1-10.

A version of the work in chapter 5 of this thesis has been published in a journal paper: Lin R., Y. Guo, F. Carozzi and M. D. Sacchi, 2022, Simultaneous deblending and source reconstruction for compressive 3D simultaneous-source acquisition data via Interpolated MSSA (I-MSSA): *Geophysics*, **87**, no. 6, 1-53.

In these publications, I was responsible for designing and programming the algorithms, preparing data examples and writing the manuscripts. Dr. Mauricio Sacchi was the supervisory author and was involved in concept formulation and manuscript editing.

To my wife, Yi Guo.

Acknowledgements

First and foremost, I would like to thank my supervisor, Prof. Mauricio D. Sacchi, for his patient guidance and supervision. Throughout my Ph.D., he always encourages me to conduct meaningful research and provides useful suggestions for my research. It has been a great fortune to work with such an amazing supervisor. His humour, invaluable expertise, and wisdom are always nurturing and inspiring me during my research and life. I want to thank my examination committee members, Prof. Martyn Unsworth, Prof. Frank Marsiglio, Prof. Jeff Gu and Prof. Daniel Trad, for their support, suggestion and guidance. I am also grateful for my colleagues in the Signal Analysis and Imaging Group (SAIG). The SAIGers are a group of geophysicists full of humour, kindness, and expertise. They make my life at the UofA colourful and wonderful. I want to thank my course instructors, Dr. Jeff Gu, Dr. Aksel Hallin, Dr. Claire Currie, and Dr. Mauricio Sacchi, for their teaching. These courses build a solid foundation for my research. I appreciate my friends Wenlei Gao, Dawei Liu, Hongling Chen, Yunfeng Chen, Jingchuan Wang, Wenhan Sun, Weiheng Geng, Xiaowen Liu, Yiru Zhou, Tianyang Li, Tongzhen Guo, Zhihong Pan, Zhenhua Li, etc. for their accompany and help during my stay at the University of Alberta. Finally, I wish to express my thanks to my family, my relatives, my parents, my parents-in-law, my cousin-in-law, my maternal aunt-in-law, and my uncle-in-law for their support and suggestions, and special thanks to my wife, Yi Guo, for her unconditional trust, understandings, accompany and encouragements throughout my academic pursuits, also my little cute, sweet and “robust” girl for her coming in my life.

Contents

1	Introduction	1
1.1	The seismic exploration method	2
1.2	Simultaneous-source data acquisition: A review	4
1.3	Contribution of this thesis	11
1.4	Thesis overview	13
2	Linear inverse problems	15
2.1	Linear inverse problems	15
2.2	Regularization methods for solving inverse problems	17
2.2.1	Quadratic regularization	18
2.2.2	Non-quadratic regularization and sparsity	19
2.3	Projected Gradient Descent (PGD) method	21
2.3.1	Step-size λ selection	22
2.3.2	A projection operator based on a low-rank approximation	22
3	Coherence pass robust Radon projection operator	27
3.1	Introduction	27
3.2	Theory	28
3.2.1	Deblending via inversion with a coherence pass projection	28
3.2.2	Coherence pass non-robust and robust Radon projection operators	30
3.3	Examples	32

3.3.1	Synthetic example	33
3.3.2	Real data example	39
3.4	Conclusions	43
4	Robust MSSA projection operator	54
4.1	Introduction	54
4.2	Theory	55
4.2.1	Separation of simultaneous-source data via the PGD method	55
4.2.2	Windowed MSSA filter	56
4.2.3	Denoiser based on the classical non-robust MSSA filter	58
4.2.4	Denoiser based on the robust MSSA filter	60
4.3	Examples	62
4.3.1	Synthetic example	63
4.3.2	Real data example	67
4.4	Conclusions	73
5	Compressive blended acquisition	77
5.1	Introduction	77
5.2	Theory	78
5.2.1	Preliminaries	78
5.2.2	3D deblending and source reconstruction via MSSA with an extracting operator	80
5.2.3	3D deblending and source reconstruction via interpolated-MSSA (I-MSSA)	82
5.2.4	The Kaiser window tapered sinc interpolation operator	83
5.2.5	Efficient rank reduction via Randomized SVD	84
5.3	Examples	85
5.3.1	Synthetic example	85
5.3.2	Behaviour of our algorithms versus decimation and rank-reduction solver	91
5.3.3	Real data example	96
5.4	Conclusions	99

6	Conclusions	104
6.1	Conclusions and Summary	104
6.2	Future recommendations	107
	Bibliography	109
	Appendices	
A	Deblending via ADMM and IRLS	118
A.1	Introduction	118
A.2	Theory	119
A.2.1	Review of the Radon transform	119
A.2.2	IRLS method for robust and sparse Radon transform	120
A.2.3	ADMM method for robust and sparse Radon transform	121
A.3	Examples	123
A.3.1	Synthetic example	123
A.3.2	Real data example	130
A.4	Conclusions	130
B	Fast and computational-efficient I-MSSA	136
B.1	Introduction	136
B.2	Method	137
B.2.1	Fast and computational-efficient Multidimensional Singular Spectrum Analysis (FMSSA) filtering	137
B.2.2	Review of compressive simultaneous source data processing via FMSSA and binning	139
B.2.3	Review of compressive simultaneous source data processing via Interpolated- FMSSA (I-FMSSA)	142
B.3	Examples	143
B.4	Conclusion	145

List of Tables

4.1	Computational time and SNR comparison for synthetic and real examples. The computational time listed here is only for one common receiver gather or one common offset gather.	72
5.1	Computational time and SNR comparison for deblending and source reconstruction with different rank-reduction solvers.	95
A.1	Comparison of different algorithms for robust/non-robust Radon transform method for different noise attenuation with synthetic examples. Note that the $IRLS_{\ell_2}$ and FISTA denote the non-robust Radon transform by solving $\ell_2 - \ell_1$ optimization problem, and the $IRLS_{\ell_1}$ and ADMM denote the robust Radon transform by solving $\ell_1 - \ell_1$ optimization problem.	127
A.2	Comparison of different algorithms for robust Radon transform method for different noise attenuation with real examples. Note that the $IRLS_{\ell_1}$ and ADMM denote the robust Radon transform by solving $\ell_1 - \ell_1$ optimization problem.	135
B.1	Computational time and SNR for irregular-grid reconstruction.	145

List of Figures

1.1	(a) Conventional survey. Each source is activated, the data are recorded for a sufficiently large record length, and then the next source is activated. The energy of the source n does not contaminate the response of the source $n + 1$. A common receiver gather (CRG) shows the response from all the sources for one particular receiver. (b) Simultaneous source acquisition with two sources (blending factor is two) where sources' responses overlap. The common receiver gather shows the response for all the sources as in a conventional survey plus incoherent noise caused by source interferences. Source firing times are random to yield incoherent source interferences in the CRG. For illustrative purposes, we consider four sources firing in the sequence $T_1 < T_2 < T_3 < T_4$ where T_i if the firing time of source i	6
1.2	(a) Simultaneous-source marine acquisition. Six sources are fired simultaneously with randomly distributed time delays for the marine acquisition. (b) Simultaneous-source land acquisition. For the land acquisition, twelve Vibroseis vehicles work simultaneously.	7
1.3	Blending operator and pseudo-deblending operator.	9
1.4	Schematic diagram of seismic processing workflow for conventional acquisition data (a) and simultaneous-source acquisition data (b).	11
2.1	Illustration of a toy example for reconstruction based on sparse regularization technique. (a) Ideal signal. (b) Observed signal. (c) Reconstructed signal via sparse inversion of its Fourier coefficients with resulting $SNR = 22.16$ dB. (d) Estimated errors between (c) and (a).	20
2.2	Illustration of a toy example for reconstruction based on projection with constrained low-rank optimization. (a) Ideal signal. (b) Observed signal. (c) Reconstructed signal via PGD method based on constrained low-rank optimization with resulting $SNR = 24.95$ dB. (d) Estimated errors between (c) and (a).	25
3.1	The workflow of the whole deblending process based on coherence pass Radon projection operators.	33

3.2	Deblending of synthetic data in common receiver gather with blending factor of two. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).	35
3.3	Synthetic data example. <i>SNR</i> versus iteration number for blending factor of two.	36
3.4	Deblending of synthetic data in common receiver gather with blending factor of six. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).	37
3.5	Synthetic data example. <i>SNR</i> versus iteration number for blending factor of six.	38
3.6	Deblending of synthetic data in common receiver gather with blending factor of ten. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).	40
3.7	Synthetic data example. <i>SNR</i> versus iteration number for blending factor of ten.	41
3.8	Synthetic data example. <i>SNR</i> versus blending factor.	42
3.9	Deblending example of a real dataset with blending factor of two (Common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a). . . .	44
3.10	Results corresponding to one common shot gather for blending factor of two. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator. (e) Blending interferences in common shot gather given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).	45
3.11	Real data example. <i>SNR</i> versus iteration number for blending factor of two.	46

3.12	Deblending example of a real dataset with blending factor of six (common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a). . . .	47
3.13	Results corresponding to one common shot gather for a blending factor of six. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences in common shot gather given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).	48
3.14	Real data example. <i>SNR</i> versus iteration number for blending factor of six. . . .	49
3.15	Deblending example of a real dataset with blending factor of ten (common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a). . . .	50
3.16	Results corresponding to one common shot gather for a blending factor of ten. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences in the common shot are also the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a). . . .	51
3.17	Real data example. <i>SNR</i> versus iteration number for blending factor of ten. . . .	52
3.18	Real data example. <i>SNR</i> versus blending factor.	53
4.1	The windowing operator and its associated adjoint operator. The windowing operator \mathcal{W}_k extracts small patches of data from a whole gather. The adjoint operator \mathcal{W}_k^* synthesizes all the filtered small patches back into a gather. A linear taper is adopted for overlaps processing. Data \mathbf{D} denotes the whole data gather, and $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \dots, \mathbf{s}_k$ represent small patches of data.	58
4.2	Synthetic data example with small window size 350×15 . (a) Clean unblended data in common-receiver gather. (b) Deblending via the PGD method with robust MSSA filtering (<i>SNR</i> = 19.6 dB). (c) Deblending via the PGD method with non-robust MSSA filtering (<i>SNR</i> = 19.2 dB). (d) Pseudo-deblended data in common-receiver gather. (e) Estimated errors between (b) and (a), corresponding to PGD with robust MSSA. (f) Estimated errors between (c) and (a), corresponding to PGD with non-robust MSSA.	65

4.3	Synthetic data example with large window size 1000×50 . (a) The clean, ideal unblended common-receiver gather data. (b) Deblending via the PGD method with robust MSSA filtering ($SNR = 21.9$ dB). (c) Deblending via the PGD method with non-robust MSSA filtering ($SNR = 12.6$ dB). (d) Pseudo-deblended common-receiver gather data. (e) Estimated errors between (b) and (a), corresponding to PGD with robust MSSA. (f) Estimated errors between (c) and (a), corresponding to PGD with non-robust MSSA.	66
4.4	Zoomed area corresponding to the rectangle in Figure 4.3. (a) The clean, ideal unblended common-receiver gather data. (b) Pseudo-deblended common-receiver gather data. (c) Deblending via the PGD method with robust MSSA filtering. (d) Deblending via the PGD method with non-robust MSSA filtering.	68
4.5	Spectra of singular values extracted from the data in Figure 4.4. The spectrum corresponds to the final singular values of the Hankel matrix extracted at 30 Hz.	69
4.6	Synthetic data example. SNR versus iteration number for varying rank. The rank for each run is fixed for the PDG algorithm with a robust MSSA filter. The PGD algorithm with a non-robust MSSA filter corresponds to the initial rank, which is increased by one every ten iterations.	70
4.7	Synthetic data example. The average SNR versus rank value for 20 realizations of different random source firing times.	71
4.8	Schematic diagram display of ocean bottom cable acquisition of the real data ¹	73
4.9	Common-channel gather of a Gulf of Mexico survey. (a) Clean unblended data in common channel gather. (b) Deblending via PGD with robust MSSA filtering ($SNR = 10.5$ dB). (c) Deblending via PGD with non-robust MSSA filtering ($SNR = 8.1$ dB). (d) Pseudo-deblended data in common channel gather. (e) Estimated error corresponding to (b). (f) Estimated error corresponding to (c).	74
4.10	Real data example. SNR plot versus rank. We vary the rank value from 6 to 20, and we observe that the robust MSSA filter generates higher SNR values and is less sensitive to rank selection than the non-robust MSSA filter	75
5.1	(a) Desired regular grid. This is also the output grid of the proposed reconstruction. (b) The irregular grid is formed by perturbing source coordinates of the desired regular grid in (a). (c) For instance, a subset of the irregular grid can be obtained by random decimation of (b). (d) Observations in (c) after assignment to the nearest-neighbour grid point.	80

5.2	Illustration of different sampling operators used in this chapter. (a) The extraction operator, includes <i>binning</i> and <i>sampling</i> . In red we indicate the given observed true source coordinates. In green we indicate the observed source coordinates after binning. In blue we indicate the desired regular-grid source coordinates. Red \rightarrow Green entails <i>binning</i> . Similarly, Green \rightarrow Blue represents <i>sampling</i> . (b) In this case, a local interpolation operator maps the observed irregular grid points as a weighted summation of desired data on regular grid points. The weights depend on the distances between observed and desired source positions and are computed via truncated sinc interpolation operator.	86
5.3	The slice display with different grids distribution in Figure 5.1. The blue arrows in Figure 5.1 indicate the positions of selected slices. (a) Clean regular data based on coordinates in Figure 5.1a. (b) Initial irregular data based on coordinates in Figure 5.1b. (c) Observed irregular data, obtained by 50% random decimation of (b), based on coordinates in Figure 5.1c. (d) Observations in (c) after the binning process and corresponding coordinates are displayed in Figure 5.1d.	87
5.4	Reconstruction and deblending results for a clean synthetic data example with a randomly 50% decimated volume for one common receiver gather. (a) Clean regular data volume. (b) Pseudo-deblended data volume after binning. (c) Result of the conventional 3D MSSA reconstruction and deblending. (d) Residuals between the clean volume (a) and (c). (e) Deblended and reconstructed data via the I-MSSA algorithm with sinc interpolation. (f) Residuals between the clean volume (a) and (e).	88
5.5	An x-slice showing reconstruction and deblending results for the synthetic example in Figure 5.4. (a) Clean regular data. (b) Pseudo-deblended data after binning. (c) Result of the conventional 3D MSSA reconstruction and deblending. (d) Residuals between (a) and (c). (e) Reconstruction and deblending results were calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between (a) and (e).	90
5.6	Convergence curves showing <i>SNR</i> versus iteration for PGD using MSSA and I-MSSA filters for reconstruction and deblending.	91
5.7	Reconstruction and deblending results for synthetic data example contaminated with noise. The data was decimated randomly at a level of 50%. (a) Clean regular data volume. (b) Pseudo-deblended noisy data volume after binning. (c) MSSA reconstruction and deblending. (d) Residuals between the clean volume (a) and (c). (e) Reconstruction and deblending are calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between the clean volume (a) and (e).	92
5.8	One x-slice display of reconstruction and deblending results for noisy synthetic example in Figure 5.7. (a) Clean regular data. (b) Pseudo-deblended data after binning. (c) Result of the conventional 3D MSSA reconstruction and deblending. (d) Residuals between (a) and (c). (e) Reconstruction and deblending results were calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between (a) and (e).	93

5.9	Convergence curves showing SNR versus PGD iteration with MSSA and I-MSSA filters. These results correspond to the data in Figure 5.7.	94
5.10	The average SNR comparison of MSSA and I-MSSA methods for reconstruction and deblending with different percentages of decimation. For each decimation percentage example, the SNR values are calculated via 50 realizations.	95
5.11	Velocity model used for the synthetic example.	96
5.12	Coordinate systems for synthetic example. (a) Observed irregular-grid coordinates system including 2500 source points. (b) The desired regular-grid coordinate system with $50 \times 100 = 5000$ source points.	97
5.13	Deblending and reconstruction of the synthetic data with the source distribution in Figure 5.12a. (a) Ground truth based on regular-grid distribution. (b) Deblending and reconstruction via the I-MSSA method with a resulting $SNR = 14.32$ dB. (c) Difference between (a) and (b). (d) Observed pseudo-deblended irregular-grid data after binning. (e) Deblending and reconstruction result via the MSSA method with a resulting $SNR = 11.56$ dB. (f) Difference between (a) and (e). (g) Difference between (b) and (e). . . .	98
5.14	Coordinate systems for the real data example. (a) True data coordinates, including 22408 source points. (b) The source coordinate distribution after 20% random decimation contains 17927 source points. (c) The desired output on a regular-grid coordinate system with $398 \times 62 = 24676$ source points. . . .	100
5.15	Deblending and reconstruction of the data with the source distribution in Figure 5.14b. The red arrows in Figure 5.14 indicate the selected slices. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) The result was obtained via the MSSA method. (d) Difference between (b) and (c).	100
5.16	Zoomed area corresponding to the red rectangle (left part) in Figure 5.15. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) Results were obtained via the MSSA method. (d) Difference between (b) and (c).	101
5.17	Zoomed area corresponding to the blue rectangle (center part) in Figure 5.15. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) Results were obtained via the MSSA method. (d) Difference between (b) and (c).	101
5.18	Deblending and reconstruction of field data with source distribution in Figure 5.14c. (a) Reconstruction and deblending result of observed data with I-MSSA method based on source coordinates distribution in Figure 5.14a. (b) Reconstruction and deblending result according to 20% decimated data with I-MSSA method based on coordinates distribution in Figure 5.14b. (c) Difference between (a) and (b).	102

A.1	Comparison of blending noise ($BF = 2$) attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).	125
A.2	Comparison of blending noise ($BF = 8$) attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).	126
A.3	Comparison of blending noise ($BF = 8$) and erratic ambient noise attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).	128
A.4	Comparison of blending noise ($BF = 8$), erratic ambient noise and random noise attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).	129
A.5	Denoising (blending noise $BF = 2 +$ erratic ambient noise) results comparison with different algorithms. (a) Noisy data. (b) Denoising with ADMM method. (c) Denoising with FISTA method. (d) Denoising with IRLS method (Non-robust). (e) Denoising with IRLS method (Robust). (f) Noise section. (g) Residual section by ADMM method. (h) Residual section by FISTA method. (i) Residual section by IRLS method (Non-robust). (j) Residual section by IRLS method (Robust).	131
A.6	Denoising (blending noise $BF = 2 +$ erratic ambient noise + random noise) results comparison with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean data. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).	132
A.7	Denoising (blending noise $BF = 8 +$ erratic ambient noise) results in comparison with different algorithms. (a) Clean data, (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean section. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).	133

A.8	Denoising (blending noise $BF = 8 +$ erratic ambient noise + random noise) results comparison with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean section. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).	134
B.1	Source coordinates distribution considered in the synthetic examples. (a) Coordinates of the irregular grid with 50 % decimation. (b) The desired regular grid of output.	143
B.2	Reconstruction and deblending results. A comparison for one CRG. (a) Clean regular data volume. (b) Pseudo-deblended data volume after binning. (c) Result with MSSA deblending and binning with $SNR = 15.87$ dB. (d) Residuals between (a) and (c). (e) Result with FMSSA deblending and binning with $SNR = 15.25$ dB. (f) Residuals between (a) and (e). (g) Result with I-MSSA deblending with $SNR = 43.66$ dB. (h) Residuals between (a) and (g). (i) Result with I-FMSSA deblending with $SNR = 43.03$ dB. (j) Residuals between (a) and (i).	144
B.3	(a) Observed decimated coordinate distribution containing 17927 source points. (b) The desired output with a regular-grid coordinate system, including $398 \times 62 = 24676$ source points.	146
B.4	Real data example. (a) Observed data after binning. (b) Result with FMSSA deblending and binning. (c) Result with I-FMSSA deblending method. (d) Difference between (b) and (c).	147

List of abbreviations and symbols

MT	Magnetotellurics
EM	Electromagnetic
HSSE	Health, Safety, Security and Environment
BF	Blending Factor
CRG	Common Receiver Gather
$f - x$	Frequency-space domain
SSA	Singular Spectrum Analysis
MSSA	Multichannel Singular Spectrum Analysis
BFGD	Bifactored Gradient Descent
I-MSSA	Interpolated-Multichannel Singular Spectrum Analysis
PGD	Projected Gradient Descent
IRLS	Iterative Reweighed Least-Squares
ADMM	Alternating Direction Method of Multipliers
CGLS	Conjugate Gradient for Least Squares
Dim	Dimension
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
IRR	Iterative Rank Reduction
$t - x$	Time-space domain
DFT	Direct Fourier Transform
IDFT	Inverse Direct Fourier Transform

MAD	Median Absolute Deviation
CS	Compressive Sensing
R-SVD	Randomized Singular Value Decomposition
CNN	Convolutional Neural Network
CCUS	CO ₂ Capture, Utilization and Storage
I-FMSSA	Interpolated Fast Multichannel Singular Spectrum Analysis
FMSSA	Fast and memory-efficient Multichannel Singular Spectrum Analysis
FFT	Fast Fourier transform
LASSO	Least Absolute Shrinkage and Selection Operator
FISTA	Fast Iterative Shrinkage Thresholding Algorithm
OMP	Orthogonal Matching Pursuit
rQRd	randomized QR decomposition
D, $\hat{\mathbf{D}}$	Matrix
d, $\hat{\mathbf{d}}$	Vector
<i>d</i>	Scalar
H, \mathbf{H}_p	Hankel matrix
W, $\mathbf{W}_r, \mathbf{W}_m$	Weighting matrix
L	Explicit matrix (maps the model to observation)
\mathbf{L}^T	Transpose matrix of L
<i>\mathcal{L}</i>	Implicit operator (maps the model to observation)
<i>\mathcal{L}^*</i>	The adjoint operator of <i>\mathcal{L}</i>
<i>\mathcal{B}</i>	Blending operator
<i>\mathcal{B}^*</i>	Pseudo-deblending operator
S	Data set
$\ \cdot\ _2^2$	ℓ_2 norm
$\ \cdot\ _p^p$	ℓ_p norm
$\ \cdot\ _1$	ℓ_1 norm
$\ \cdot\ _F^2$	The Frobenius norm

$ \cdot $	Absolute value
∇J	The gradient
$\mathcal{R}(\mathbf{m})$	The regularization term
\mathcal{P}	The projection operator
\mathcal{H}	The Hankelization operator
\mathcal{A}	The anti-diagonal averaging operator
\mathcal{R}	The Radon transform
\mathcal{R}^*	The adjoint Radon transform
\circ	Element-wise multiplication
\mathcal{I}	Sampling/Extraction operator
\mathcal{W}	The interpolation operator
\mathcal{W}^*	The adjoint interpolation operator
μ, μ_1, μ_2	Trade-off parameter
λ	Step-size parameter
x_i	Elements of a vector
d_{ij}	Elements of a matrix
\mathcal{W}	The windowing operator (Extract local window data)
\mathcal{W}^*	The adjoint operator of \mathcal{W} (Synthesize local window data)
e_{max}	Maximum eigenvalue
$\xi(x_k, y_k)$	Coordinates of irregular-grid point
$\eta(\hat{x}_i, \hat{y}_j)$	Coordinates of regular-grid point
$\mathbf{H}^{(n)}$	Level-n Hankel matrix
\mathbf{H}_p	Rank p Hankel matrix
$\text{prox}_\tau\{y\}$	Proximity operator
\mathcal{F}	Fast Fourier transform
\mathcal{F}^{-1}	Inverse fast Fourier transform
$*$	Vector/Matrix multiplication
Ω	A random matrix

CHAPTER 1

Introduction

Geophysicists aim to develop techniques to quantitatively estimate subsurface geological structures and properties. Moreover, they are also interested in understanding processes occurring in the Earth's interior (Fowler et al., 1990). Mainly, subsurface investigations are carried out via indirect methods. In other words, subsurface properties are inferred from measurements recorded on the surface of the Earth. Geophysicists adopt different strategies for obtaining these properties. These methods could be divided into three main categories. The first of these, the seismic method, uses traveltime and amplitude variations of propagating waves to estimate the elastic properties of the subsurface (Yilmaz, 2001, 2021). Human-made explosive sources or vibratory disturbances are used to emit seismic waves into the subsurface. Geological interfaces reflect these waves; they propagate upwards and are finally recorded by arrays of receivers. After applying computationally intensive signal processing and inversion techniques, the data (seismograms) are mapped to images delineating geological structures.

Then we have the second class of methods that is entirely different in principle. They are often called potential field methods because they depend on the distortion of a scalar or vector potential field caused by some perturbation of a suitable physical property in the subsurface (Telford et al., 1990; Blakely, 1996). Examples of the latter are the gravity prospecting method which measures the distortion of the gravitational potential caused by the variability of subsurface density distribution (Ander et al., 1989). Similarly, electric and magnetic methods measure the distortion of the electric and magnetic fields due to subsurface variations of resistivity and susceptibility (Reynolds, 2011).

Last, a large category of methods includes techniques that can be named electromagnetic geophysical methods in which natural or artificially generated electric or magnetic fields are measured on Earth's surface or in boreholes to estimate electrical or magnetic

properties variations. This group contains some popular techniques: induced polarization, controlled-source electromagnetic, magnetotelluric (MT), and ground penetrating radar surveying (Telford et al., 1990; Reynolds, 2011). In particular, the MT method uses naturally occurring electromagnetic (EM) waves in Earth's atmosphere and magnetosphere. These fields induce currents into the Earth, which are measured at the surface and contain information about subsurface resistivity structures. The MT method can image the electrical resistivity structure of the earth with a depth range from a few 100 metres to several 100 kilometres (Tikhonov, 1950; Cagniard, 1953; Cantwell and Madden, 1960; Price, 1962; Bai et al., 2010). The main applications of MT prospecting include the study of active tectonics and continental dynamics (Arora et al., 2007; Zhao et al., 2012), hydrocarbon exploration (Unsworth, 2005), mine exploration, and geothermal exploration (Munoz, 2014).

My work centers on seismic exploration, particularly processing methods for simultaneous-source seismic data. In the following sections, I briefly describe the seismic reflection method and provide an overview of simultaneous-source acquisition seismic data processing methods.

1.1 The seismic exploration method

The seismic exploration method is essential for discovering, exploiting, and monitoring resources, i.e., hydrocarbon reservoir. It is also adopted for environmental and geotechnical studies of the near-surface (Yilmaz, 2021). The seismic method uses waves caused by a passive or active source to emit elastic waves down into the Earth. As they travel downwards, these waves encounter interfaces separating materials (rocks) of different elastic properties and densities. At these interfaces, part of the energy is transmitted downward and reflected upwards. Waves reflected upwards are recorded by an array of receivers¹. A multi-source experiment permits illuminating a target area in the subsurface with waves and acquiring data that are then numerically processed to obtain images. Customary, four stages are associated with the seismic method:

- **Data acquisition:** In this stage, seismic data are acquired after selecting acquisition parameters related to the depth of investigation and equipment availability. These parameters are also associated with specific sampling requirements for time and space discretization. These requirements are given by the Nyquist-Shannon sampling theorem (Oppenheim et al., 1999)². The seismic acquisition can be carried out

¹Sensors of seismic signals are electromagnetic transducers placed on the surface of the Earth or hydrophones placed just below the ocean surface. Sensors that measure particle velocity and pressure are also placed on the sea-bed.

²Nyquist-Shannon sampling is a strict requirement for regular spatial sampling. Today, one could use Compressive Sensing (Donoho, 2006) for random sub-sampling techniques and sample at less severe rates than the Nyquist-Shannon sampling theorem.

onshore or offshore, it can be a 2D or a 3D acquisition (Vermeer, 1990, 2002), and receivers and sources can be deployed within these classes via different configurations. These acquisition types can be carried out into two distinct modalities that we name conventional and simultaneous-source acquisitions (Abma and Foster, 2020).

- **Data preconditioning (often also called data processing):** Data inaccuracies due to sources and receivers' spatial position are corrected. During this stage, we also apply incoherent and coherent noise removal techniques and source equalization methods (deconvolution) (Yilmaz, 2001). The main idea is to process the data so that each common source approximates the acoustic or elastic Green function corresponding to an experiment in which a point source is propagated in the subsurface. Included in the preconditioning stage are methods for simultaneous-source separation as those that I investigate in my doctorate.
- **Imaging and inversion:** Imaging and inversion are similar processes. In general, imaging refers to methods to determine the boundaries of subsurface structures. Inversion refers to approaches to estimate subsurface parameters such as elastic moduli or propagation speed of compressional and shear waves. Both imaging and inversion are based on solving classical math-physics problems associated with wave propagation phenomena. Historically, imaging techniques are referred to as migration algorithms. Early migration algorithms were based on the Huygens principle and Kirchhoff integral theorem (Gray et al., 2001). Modern imaging and inversion algorithms operate with one-way and two-way acoustic or elastic wave equations, often linearized via the Born approximation (Sava and Hill, 2009).
- **Seismic and geological interpretation:** At this stage, subsurface images are interpreted with the help of regional geological information, core samples, and formation properties estimated by well logging (Brown, 2011). In other words, images of the subsurface provide boundaries, and after inversion, one can also estimate material (rock) properties. However, these images do not tell us how sediments have deposited during a geological time or how structures (faults and folds) have emerged; neither they tell us how fluids in the crust have migrated and accumulated in reservoirs. To discover areas with an accumulation of hydrocarbons, geologists and geophysicists use seismic images in conjunction with geological information for interpretation.

The acquisition stage of the seismic reflection method for the exploration of oil and gas often takes a considerable part of the budget allocated for prospecting an area (Claerbout, 1992). Starting in the late 1980s and with the development of 3D seismology (Biondi, 2006) to prospect for oil and gas in many regions of the world, many efforts to reduce acquisition costs were propelled by large oil companies and geophysical contractors worldwide. As part

of these efforts, the simultaneous-source acquisition has become essential to decrease acquisition costs and obtains high-density seismic data (Rozemond, 1996; Krohn and Johnson, 2006; Krohn and Neelamani, 2008; Howe et al., 2008; Abma and Foster, 2020). The first 3D field trial of the simultaneous-source acquisition technology involving 8 vibrator sources was conducted successfully in 2008 (Howe et al., 2008). Since then, the simultaneous-source acquisition has become increasingly popular in the oil industry as a cost-effective way to record high-density seismic data. Simultaneous sourcing is a revolution in reflection seismology that has changed seismic data acquisition. In spite of its success, developing and improving processing algorithms to cope with simultaneous source data is still an active field of research.

The significant advantage of simultaneous-source acquisition is that seismic exploration in the data acquisition stage using simultaneous-source acquisition methods offers substantial economic advantages over conventional acquisition methods for both land and marine seismic surveys. Compared to traditional seismic surveys, the simultaneous-source acquisition allows more seismic data to be collected in less time. More data can enhance the view of the subsurface, minimizing the survey time to lower survey costs and reduce HSSE (Health, Safety, Security and Environment) exposure (Abma and Foster, 2020), such as lowering exposure time in extreme environments (ice or barren desert). Most simultaneous source surveys involve collecting more data, requiring less time, and gaining HSSE benefits. Many intensive simultaneous source surveys are economically impractical using conventional acquisitions. Acquiring these surveys with simultaneous sources reduces the cost to the point where very dense surveys are now practical.

1.2 Simultaneous-source data acquisition: A review

In the seismic data acquisition stage, conventional seismic data acquisition uses arrays of receivers deployed on the Earth's surface to measure the response caused by one source. The response is collected by an array of receivers in the form of a multichannel time series. The latter is often called a common-source gather. The source is moved to a new position, and the experiment is repeated to generate a new common-source gather. In general, the process is repeated many times with sources at different places.

In a conventional survey, responses of individual sources are recorded separately. For instance, let us consider a survey that consists of N sources activated at times T_1, T_2, \dots, T_N . In general, the attenuation of seismic waves ensures that the energy reflected by subsurface interfaces and recorded by receivers is practically negligible after L_t seconds (record length)³. Therefore if one chooses $T_{n+1} - T_n \geq L_t$, the n th source will not contaminate

³For typical exploration targets in sedimentary basins, L_t is about 8 to 10s.

the record corresponding to the $n + 1$ source. Nowadays, it has become customary to ignore the requirement mentioned above. Instead, one can adopt a simultaneous-source acquisition (also called blending acquisition) modality (Berkhout, 2008; Hampson et al., 2008; Abma and Foster, 2020). In this case, receivers record the response of more than one source simultaneously ($T_{n+1} - T_n < L_t$). One can significantly reduce acquisition time and, consequently, the cost of acquiring dense seismic surveys via a simultaneous-source survey. Unfortunately, seismic data processing and imaging techniques require access to individual source responses. In recent years, significant efforts have been made in developing simultaneous source separation (also called deblending) methods (Akerberg et al., 2008; Moore et al., 2008; Abma et al., 2010; Mahdad et al., 2011; Li et al., 2013; Abma et al., 2015; Li et al., 2019). These methods aim to turn simultaneous source seismic data into the data one would have obtained via a conventional acquisition. Simultaneous-source data processing has become an exciting field for developing new data analysis tools. These tools are derived from the theory of inverse problems, harmonic analysis, sparse representation theory, and constrained-rank optimization.

Figure 1.1 illustrates the difference between conventional and simultaneous-source acquisitions. Sources in the figure are fired in the sequence 1, 2, 3, and 4; being 1 the source that is fired first and 4 the last source fired. In Figure 1.1a, the sources are fired as independent experiments with no contamination from previously fired sources. Figure 1.1b shows the simultaneous-source experiment where one allows the measured responses to produce overlapping in time. Figure 1.2 displays how the simultaneous-source acquisition works for the land and marine acquisition. Six airguns are fired simultaneously with randomly distributed activation time delays for the marine acquisition (Figure 1.2a). The receiver lines keep recording the responses from all six sources. In Figure 1.2b, twelve Vibroseis vehicles work simultaneously for the land acquisition, and the geophones keep listening to all the signal responses from the subsurface.

Many geophysical problems can be expressed as linear inverse problems. For instance, denoising and source equalization problems are often written as linear inverse problems. Non-linear problems associated with the inversion of, for example, elastic properties are often linearized. Examples of the latter are amplitude versus offset inversion, inversion of acoustic impedance, least-squares migration methods, etc. In this dissertation, I also view the source separation problem as an inverse problem. More specifically, the data acquired by a simultaneous source survey can be written by the following expression (Abma and Foster, 2020)

$$b(t, r) = \sum_{s \in S} D(t - \tau(s), r, s) \quad (1.1)$$

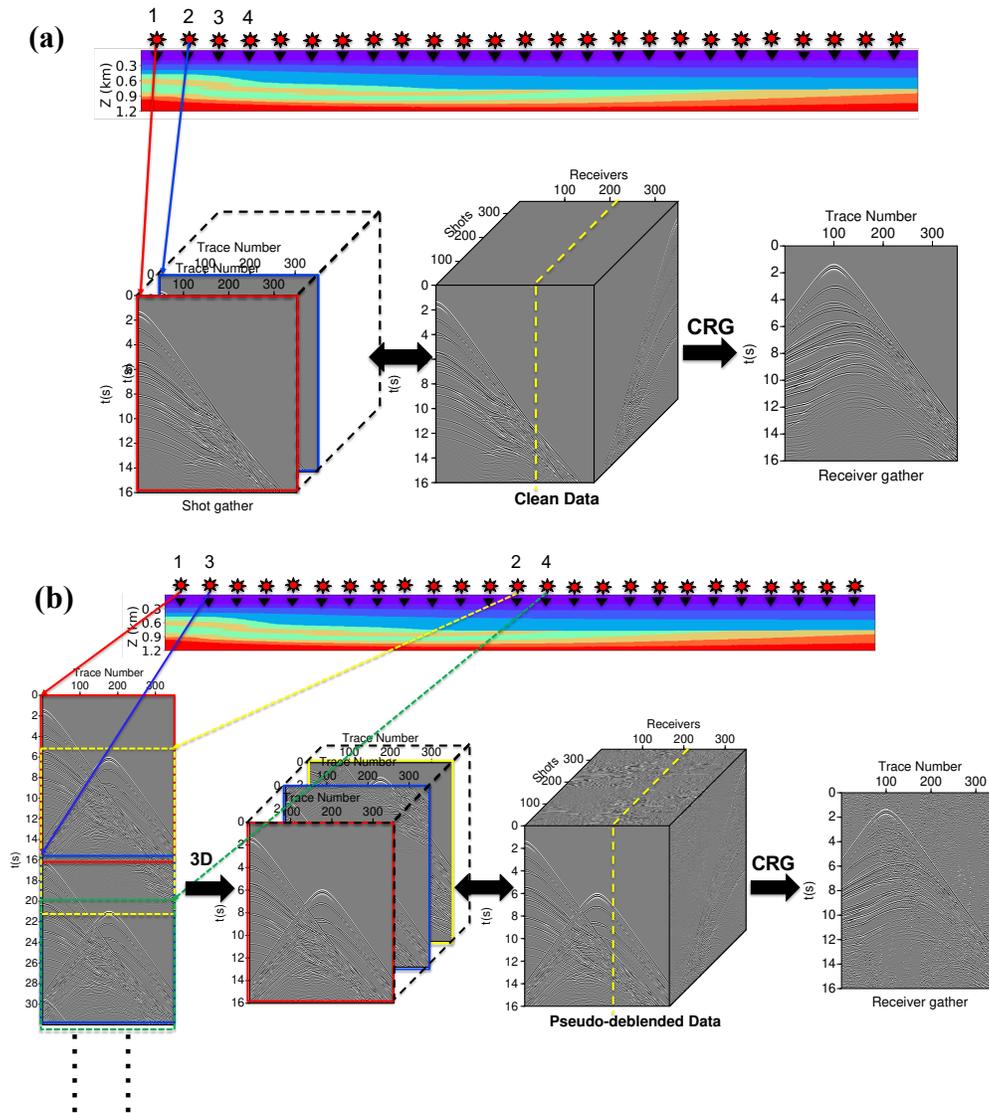


Figure 1.1: (a) Conventional survey. Each source is activated, the data are recorded for a sufficiently large record length, and then the next source is activated. The energy of the source n does not contaminate the response of the source $n + 1$. A common receiver gather (CRG) shows the response from all the sources for one particular receiver. (b) Simultaneous source acquisition with two sources (blending factor is two) where sources' responses overlap. The common receiver gather shows the response for all the sources as in a conventional survey plus incoherent noise caused by source interferences. Source firing times are random to yield incoherent source interferences in the CRG. For illustrative purposes, we consider four sources firing in the sequence $T_1 < T_2 < T_3 < T_4$ where T_i is the firing time of source i .

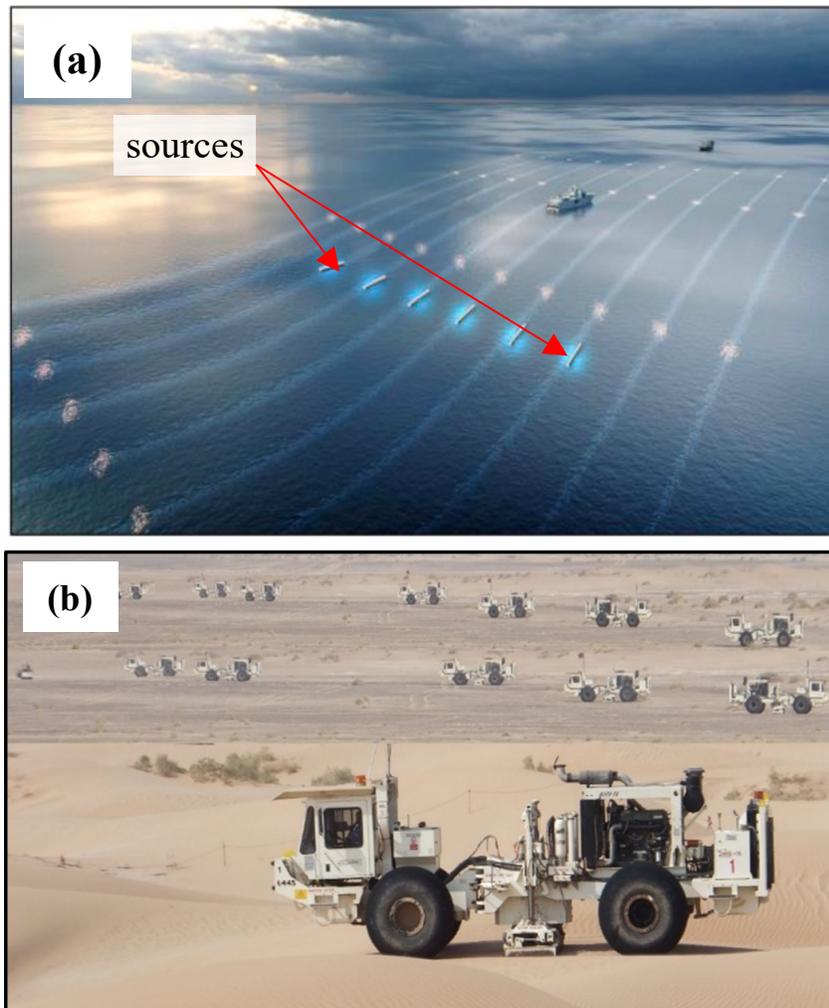


Figure 1.2: (a) Simultaneous-source marine acquisition^a. Six sources are fired simultaneously with randomly distributed time delays for the marine acquisition. (b) Simultaneous-source land acquisition^b. For the land acquisition, twelve Vibroseis vehicles work simultaneously.

^aFigure from Dhelie et al. (2019).

^bFigure from BGP Inc. China National Petroleum Corporation.

where $b(t, r)$ is the simultaneous source data, often called blended data. The variable r denotes receiver or channel, and s denotes source. The random activation time of source s is denoted by $\tau(s)$. Clearly, the goal is to recover $D(t, s, r)$ from its compressed version $b(t, r)$. In a simultaneous, source survey, one directly measures $b(t, r)$ and is tasked with recovering $D(t, r, s)$ from $b(t, s)$. In essence, this is a linear inverse problem. Moreover, it is clear that the problem is non-unique because of the number of observations (size of $b(t, r)$) which is much smaller than the size of the $D(t, r, s)$.

We assume the recorded blended data are represented by $b(t_n, r)$ where $t_n = (n - 1)\Delta t$, $n = 1 \dots N$ denotes discrete time, Δt is the sampling interval, N is the total number of samples of the simultaneous source record and $\Delta t(N - 1)$ denotes the simultaneous source record length. Therefore, for one receiver r , the blended data $b(t_n, r)$ can be expressed as the vector \mathbf{b} of size $N \times 1$. Similarly, the unblended data is denoted by $D(t_n, s_i, r)$, with $t_n = (n - 1)\Delta t$, $n = 1 \dots N_t$ and $s_i = s_0 + (i - 1)\Delta s$, $i = 1 \dots N_s$. The unblended data can be written as the matrix \mathbf{D} of size $N_t \times N_s$. To simplify our approach, we assume N_s equidistant sources with source spatial interval Δs and the first source placed at position s_0 . The length of the desired deblended data is $\Delta t(N_t - 1)$. Evidently, the acquisition turnaround time saving requires $N_s \times N_t \gg N$. For completeness, we also define the blending factor

$$BF = \frac{N_s \Delta t(N_t - 1)}{\Delta t(N - 1)} \quad (1.2)$$

which represents the ratio of the total acquisition time of N_s sources to the total length of the simultaneous source record.

Equation 1.1 can be written in operator form as follows

$$\mathbf{b} = \mathcal{B}\mathbf{D}, \quad (1.3)$$

where \mathcal{B} denotes the blending operator (See Figure 1.3), which contains the random activation time of sources, and \mathbf{b} represents blended data, and \mathbf{D} is the desired data cube that one would have acquired via a conventional seismic survey. The blending operator is equivalent to adding the different shots gathers into a long record also called a ‘‘super shot gather’’ but with random jittering time from different sources. The adjoint operator⁴ of \mathcal{B} , denoted by \mathcal{B}^* (Figure 1.3), represents the so-called pseudo-deblending operator (Berkhout, 2008)

$$\tilde{\mathbf{D}} = \mathcal{B}^*\mathbf{b}. \quad (1.4)$$

Pseudo-deblending is equivalent to applying time shifts and dividing long simultaneous-source records onto the time length of the record one would have obtained via conventional

⁴The adjoint is the operator such that $\langle \mathcal{B}x, y \rangle = \langle \mathcal{B}^*y, x \rangle$. If \mathcal{B} is a matrix \mathbf{B} , then $\mathcal{B}^* \equiv \mathbf{B}^T$.

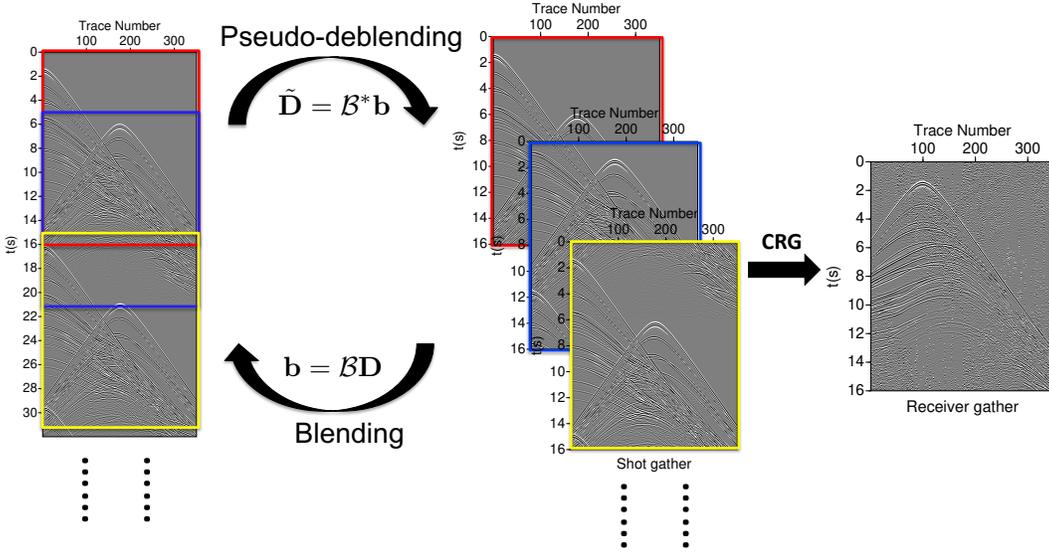


Figure 1.3: Blending operator and pseudo-deblending operator.

seismic acquisition. However, pseudo-deblending does not remove interferences resulting from overlapping sources. This is seen in Figures 1.1b and 1.3 where one Common Receiver Gather (CRG) was extracted from the acquired data cube $\tilde{\mathbf{D}}$ to show the incoherent blending interferences. Usually, for visualization purposes, the pseudo-deblended data in CRG is displayed in order to see the incoherent blending interferences.

The objective of simultaneous-source separation (deblending) is to estimate \mathbf{D} from recorded blended data \mathbf{b} or pseudo-deblended data $\tilde{\mathbf{D}}$ in order to obtain the separated shot gathers. Two categories of methods have been proposed to achieve the goal mentioned above. The first category of methods, often named deblending by denoising, treats deblending as a noise filtering problem that operates on a domain where interferences are incoherent erratic signals (Berkhout, 2008). The second category of methods, deblending via inversion, considers deblending as a linear inverse problem where one seeks to synthesize the desired data \mathbf{D} honouring the simultaneously recorded data \mathbf{b} .

Deblending via denoising operates directly on the pseudo-deblended data $\tilde{\mathbf{D}}$ (Beasley, 2008; Kim et al., 2009) by solving the following problem:

$$\mathbf{D}_{obs} = \mathbf{D} + \mathbf{n} \quad (1.5)$$

where $\mathbf{D}_{obs} = \tilde{\mathbf{D}}$ is the pseudo-deblended data, and \mathbf{n} is called the blending interferences or blending noise. The denoising-based deblending methods mainly rely on the incoherency of blending noise in common receiver, common offset or common midpoint domains (Hampson

et al., 2008; Mahdad et al., 2011). This incoherency in these domains (i.e., in the common receiver gathers of Figure 1.1b) is due to the random activation time of sources in equation 1.1. In general, filtering techniques are adopted to attenuate incoherent noise. Examples of incoherent noise attenuation in blended data include f - k median filters (Huo et al., 2012), f - x filters (Spitz et al., 2008; Peng et al., 2013), and techniques based on the Radon transform (Moore, 2010; Ibrahim and Sacchi, 2013). These methods are commonly used in the early stages of processing simultaneous-source data.

Another category of source separation or deblending comprises inversion-based methods, in which one attempts to pose source separation and deblending as an inverse problem (Abma et al., 2010; Mahdad et al., 2011; Li et al., 2013; Cheng and Sacchi, 2015, 2016). The main goal is to find the solution \mathbf{D} in equation 1.3 that minimizes the residuals $\mathbf{b} - \mathcal{B}\mathbf{D}$ while imposing a coherence pass operator to reduce blending interferences. Normally, sparse inversion methods are adopted to estimate transform domain coefficients that synthesize the ideal deblended data \mathbf{D} . Examples include methods that adopt Fourier domain thresholding algorithms (Abma et al., 2010; Mahdad et al., 2011; Abma et al., 2015; Song et al., 2019; Bahia and Sacchi, 2019), sparse Radon transforms (Akerberg et al., 2008; Lin and Sacchi, 2020a,b; Ibrahim and Trad, 2020), sparse Seislet transforms (Chen et al., 2014), sparse Curvelet transform inversion (Wason et al., 2011; Wang and Geng, 2019), and sparse inversion with migration operators (Cheng et al., 2016; Ibrahim and Trad, 2020). Alternatively, rather than applying sparse inversion methods on auxiliaries bases, one can also adopt rank-constrained optimization to iteratively solve the deblending problem (Maraschini et al., 2012b; Cheng and Sacchi, 2015, 2016; Jeong et al., 2020; Lin et al., 2021). Approaches based on reduced-rank filtering iteratively apply denoising to small data windows extracted from common-receiver gathers (Cheng and Sacchi, 2015). The reduced-rank filter adopted in the projected gradient descent algorithm is the Multichannel Singular Spectrum Analysis (MSSA) filter adapted for seismic data processing (Oropeza and Sacchi, 2011). MSSA filtering, also denominated the Cazdow filter (Cazdow, 1988), has been used extensively for prestack seismic denoising and reconstruction (Trickett, 2008; Trickett et al., 2012).

For conventional acquired seismic data, the seismic acquisition was considered somewhat independent of processing, and processing was considered a separate topic from the seismic acquisition. While for simultaneous-source data processing, compared with the processing workflow for conventional acquisition data, we add another step, deblending, to continue to follow the traditional processing workflow (See Figure 1.4). The connection between acquisition and processing changed from the conventional approaches for simultaneous-source data. Simultaneous sourcing forces a significantly stronger connection to be made between the two. Even though we add one more step, deblending, inside the processing workflow, the economic benefits of the simultaneous source approach are evident. The simultaneous source approach significantly reduces acquisition time, reducing costs and, hence, less like-

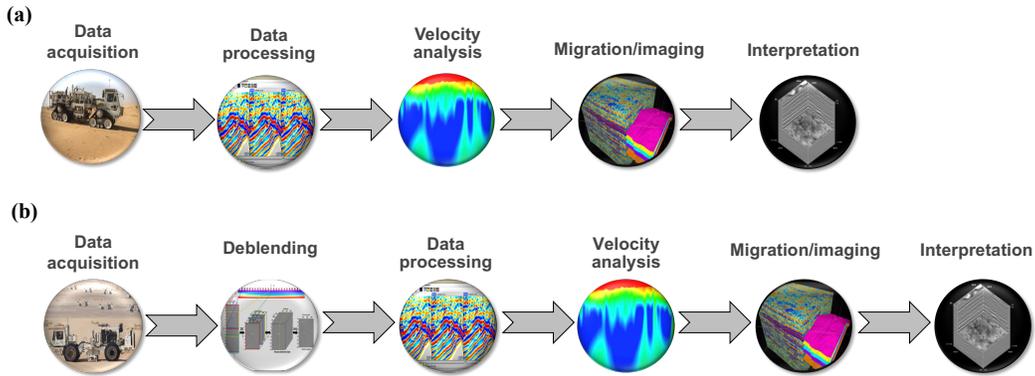


Figure 1.4: Schematic diagram of seismic processing workflow for conventional acquisition data (a) and simultaneous-source acquisition data (b).

likelihood of accidents or exposure to extreme weather conditions. At the same time, it could enable considerably higher source density or a combination of higher source density and faster surveys. For instance, for a field land trial of simultaneous source acquisition, the number of vibrator source points acquired per day by the 8 vibrators was over 500 per unit, 4000 source points per day for the crew, compared with an ongoing Algeria survey production numbers of around 100 to 150 source points per day using two alternating fleets of vibrators (Abma and Foster, 2020). The simultaneous source method can achieve 40 times improvement over conventional practices and in about one-fourth the time of the standard production approach. The main cost of a land survey is related to the number of days it takes to acquire the data. Therefore, the faster a survey can be obtained, the more inexpensive the survey cost and the lower the price per square kilometre for data acquisition. (Abma and Foster, 2020). As I mentioned, safety considerations also make reducing the number of days a seismic crew spends in the field more desirable.

1.3 Contribution of this thesis

The main contributions of this thesis are summarized as follows:

- I propose a new inversion scheme for separating simultaneous-source data. In this scheme, I define a robust and sparse Radon transform that acts as a coherence pass operator. The latter is used as a projection operator in a projected gradient descent framework. The main idea is to estimate deblended data that honours the original blended data and is free of source interferences. The proposed technique shows a

significant improvement in convergence when the coherence pass projection is derived from a robust and sparse Radon transform. This is a consequence of having an iterative deblending algorithm that applies intense denoising to erratic blending noise in its initial iterations. The coherence pass robust Radon operator acts as a data projection operator that preserves coherent signals and annihilates incoherent blending noise right from the start of the iterative process.

- I also propose a new inversion scheme for separating simultaneous-source data from the perspective of a low-rank approximation constraint. I adopt a windowed robust Multichannel Singular Spectrum Analysis (MSSA) filter in the frequency-space domain as the projection operators in a projected gradient descent framework for suppressing simultaneous source interferences. The MSSA filter is reformulated as a robust optimization problem solved via a bifactored gradient descent (BFGD) algorithm, and robustness becomes achievable by adopting Tukey’s biweight loss function for the design of the robust MSSA filter. The proposed robust MSSA filter overcomes the limitations of the non-robust (classical) MSSA filter that requires breaking down common-receiver gathers into small overlapping windows, and the input rank parameter varies from window to window. Therefore, it is less sensitive to the rank selection and makes it appealing for deblending applications that require windowing. Additionally, the robust MSSA projection effectively attenuates blending interferences during the initial iterations of the projected gradient descent method. Compared to the classical non-robust MSSA filter, I also report an acceleration of the projected gradient descent method convergence when we adopt the robust MSSA filter.
- I explicitly address simultaneous deblending and irregular-source reconstruction via inversion using iterative rank-reduction with an irregular-grid geometry. In particular, I adopt a new proposed interpolated-MSSA (I-MSSA) method that combines the projected gradient descent method to deblend and reconstruct sources in situations where the acquired blended data correspond to sources with arbitrary irregular-grid coordinates. The I-MSSA method adopts an interpolation operator and permits to honour true spatial coordinates, overcoming the limitation of conventional MSSA filters requiring data deployed on a regular grid. The proposed I-MSSA method can cope with the real scenario of sources at quasi-regular or irregular coordinates, similar to Compressive Sensing (CS) based surveys, to reduce acquisition time by blending sources and using fewer sources, thereby improving the efficiency of field data acquisition.

Overall my research centers on the seismic exploration method and, in particular, in the subfields of simultaneous-source acquisition and processing with regular-grid and irregular-grid geometry. My thesis contributes to developing algorithms for robust projections to speed up the convergence of the projected gradient descent method, which is used for

simultaneous-source separation. My thesis also contributes to separating and reconstructing simultaneous-source where the acquired blended data correspond to an irregular-grid distribution geometry (honour true spatial coordinates).

1.4 Thesis overview

In **Chapter 2**, I illustrate a brief review of the linear inverse problem and provide a concise tutorial that addresses solving linear inverse problems with the regularization (quadratic and non-quadratic) methods and the Projected Gradient Descent (PGD) method. In addition, a specific denoising technique (low-rank approximation with the one dimensional singular spectrum analysis algorithm) is introduced, which can be interpreted as a projection operator in the PGD method.

In **Chapter 3**, I introduce an iterative method for simultaneous source separation (deblending) which adopts a denoiser based on the Radon transform. The proposed technique adopts the robust and sparse Radon transform to define a coherence pass operator that is used in conjunction with the PGD method to guarantee solutions that honour simultaneous source records. Coherence pass non-robust and robust Radon projection operators are illustrated in this chapter. An important improvement in convergence is attainable when the coherence pass projection is derived from a robust and sparse Radon transform. This is a consequence of having an iterative deblending algorithm that applies intense denoising to erratic blending noise in its initial iterations. The coherence pass robust Radon operator acts as a data projection operator that preserves coherent signals and annihilates incoherent blending noise right from the start of the iterative process. I compare the algorithm with its non-robust version with synthetic and real data examples.

In **Chapter 4**, I introduce a robust MSSA filtering algorithm as the projection filter of the PGD framework to solve the simultaneous source separation problem. Usually, the MSSA filter requires breaking down common-receiver gathers, or common offset gathers into small overlapping windows. This chapter provides two types of windowed projection filters (windowed non-robust and robust MSSA filters). I reformulate the MSSA filter as a robust optimization problem solved via the BFGD algorithm. Robustness becomes achievable by adopting Tukey's biweight loss function to design the robust MSSA filter. The traditional MSSA projection filter method needs the filter rank as an input parameter, which can vary from window to window. The proposed robust MSSA projection filter is less sensitive to the rank selection, making it appealing for deblending applications that require windowing. Additionally, the robust MSSA projection method effectively attenuates random source interferences during the initial iterations of the PGD method, resulting in an acceleration of

the convergence. Finally, I provide synthetic and real data examples and discuss heuristic strategies for parameter selection.

In **Chapter 5**, I propose to adopt an Interpolated-MSSA (I-MSSA) method to simultaneously deblend and reconstruct sources in situations where the acquired blended data correspond to sources with arbitrary irregular-grid coordinates. In essence, I extend the iterative rank-reduction method implemented via MSSA filtering that can honour true source coordinates to overcome the limitation of the traditional method, which requires data deployed on a regular grid. The proposed I-MSSA method adopts Kaiser window tapered sinc interpolation operator and permits to honour true source coordinates and avoiding errors associated with allocating off-the-grid source coordinates to the desired output grid. This work focuses on recovering unblended regular-grid data from irregular-grid compressive simultaneous-source data and reducing acquisition time by blending sources and using fewer sources.

Chapter 6 contains the conclusions of this thesis. I summarize the main contents and identify the contributions and limitations of the algorithms developed in this thesis. I also discuss potential research directions and provide recommendations for future work.

CHAPTER 2

Linear inverse problems: Regularization and Projected Gradient Descent methods

This chapter aims to introduce tools and numerical algorithms I will use to develop methods for the simultaneous source separation presented in Chapters 3, 4 and 5. It intends to be a concise tutorial addressing linear inverse problems, regularization (quadratic and non-quadratic) methods, and Projected Gradient Descent (PGD) algorithm. I will also describe a specific denoising technique (low-rank approximation) that I have used extensively in my work. The latter is the Singular Spectrum Analysis (SSA) method and its multichannel version, the multichannel SSA (MSSA) method. For this chapter, a simple one-dimensional signal reconstruction example (linear inverse problem) is provided by adopting the regularization method and the PGD method.

2.1 Linear inverse problems

Many seismic data processing problems can be written as the solution to an inverse problem. Examples abound in seismic data processing of algorithms that have their roots in the solution of inverse problems. A linear inverse problem arises when we have measurements (data) related to a model of interest (e.g., a signal, an image, subsurface physical parameters, etc.) via a linear system of equations. The latter can be written using standard matrix-times-vector notation as follows

$$\mathbf{d} = \mathbf{L}\mathbf{m} \tag{2.1}$$

where \mathbf{d} is the $N \times 1$ vector of observations. Similarly, \mathbf{L} represents the matrix that maps the model \mathbf{m} into \mathbf{d} . Equation 2.1 is the linear inverse problem in an explicit form where

one has access to the $N \times M$ matrix \mathbf{L} and measurements \mathbf{d} . In many situations, we do not have access to the explicit form of the forward operator \mathbf{L} . In this cases, we can write

$$\mathbf{d} = \mathcal{L}\mathbf{m} \quad (2.2)$$

where \mathcal{L} is now the forward operator given in the implicit form. This formulation is often called a matrix-free approach and is very useful when dealing with linear operators that cannot be written in explicit form. In more practical words, one can think that we have a computer program to which one gives an input vector of parameters \mathbf{m} and the program produces an output data vector \mathbf{d} . The task of the inversion is to find an operator \mathcal{L}^{-1} that can be used to obtain an estimator of the model parameters $\hat{\mathbf{m}} = \mathcal{L}^{-1}\mathbf{d}$.

We also assume the data is contaminated with noise. Therefore, one does not want to fit the observations exactly. The latter can be indicated by rewriting the observations as follows

$$\begin{aligned} \mathbf{d}^{obs} &= \mathbf{d} + \mathbf{n} \\ &= \mathcal{L}\mathbf{m} + \mathbf{n}. \end{aligned} \quad (2.3)$$

The vector \mathbf{n} is used to indicate errors or noise in the measurements \mathbf{d} . The observations are now called \mathbf{d}^{obs} to differentiate them from error-free observations \mathbf{d} . It is interesting to note that when \mathcal{L} is the identity operator; we have a classical denoising problem where one wants to approximate \mathbf{d}^{obs} by $\mathbf{d} = \mathbf{m}$,

$$\begin{aligned} \mathbf{d}^{obs} &= \mathbf{d} + \mathbf{n} \\ &= \mathbf{m} + \mathbf{n}. \end{aligned} \quad (2.4)$$

We interpret the equation 2.4 as finding \mathbf{m} that approximates the ideal data without noise.

When \mathcal{L} denotes the convolution operator such as in many problems in non-destructive testing, astronomy, geophysics, and medical imaging, we have the so-called deconvolution problem (Ulrych and Sacchi, 2005; Bertero et al., 2009). Similarly, when one is tackling a data reconstruction problem (also named an inpainting problem), the operator \mathcal{L} could be the sampling matrix (Liu and Sacchi, 2004). Similarly, for the deblending problem that I will address in Chapters 3, 4 and 5, \mathcal{L} can be an implicit form of blending operator.

Many techniques have been proposed to solve linear inverse problems (Tarantola, 2005; Menke, 2018). This chapter reviews two algorithms for solving the linear inverse problem: regularization techniques (Tikhonov, 1943) and the PGD method (Piana and Bertero, 1997). These are the techniques I have adopted for my research when investigating the deblending problem.

2.2 Regularization methods for solving inverse problems

Usually, inverse problems are ill-posed problems. An ill-posed problem is a mathematical problem where either the solution does not exist, the solution is non-unique, or the solution is unstable. Obviously, geophysical problems do have an answer as we can always obtain a result according to limited observations. However, the solution is non-unique or unstable. The operator \mathcal{L} can have a non-trivial null space; therefore, more than one solution satisfies the observations. More often, the noise in the data causes unsuitability in the solution and makes the problem ill-posed. In other words, a slight amount of noise or errors contaminating the data can produce large perturbations in the estimated model parameters. The latter is a facet ubiquitous to geophysics. The solution of an inverse problem not only involves finding (if it exists) the inverse \mathcal{L}^{-1} , but one also must make assumptions about the regularity of the signal or model parameters \mathbf{m} that one is trying to estimate. The main goal of inverse theory is to design ways to find stable and unique solutions to ill-posed problems (Tarantola, 2005; Menke, 2018). When solving an inverse problem, one generally seeks a solution that fits the data. The latter can be expressed by posing the problem as one where one minimizes the distance between the observed data \mathbf{d}^{obs} and the predicted data computed via the synthesis or forward operator $\mathcal{L}\mathbf{m}$. We can measure the distance as mentioned earlier using, for instance, the ℓ_2 norm

$$\mathcal{M}(\mathbf{m}, \mathbf{d}^{obs}) = \|\mathbf{d}^{obs} - \mathcal{L}\mathbf{m}\|_2^2 \quad (2.5)$$

where the symbol $\|\cdot\|_2$ indicates the ℓ_2 norm of a vector ¹. The cost function in equation 2.5 does not have one minimum due to a limited number of observations available, resulting in an undetermined problem in which there are more unknowns than observations (Tikhonov, 1943; Menke, 1989), or the solution arising from its direct minimization is unstable as the limited observations usually contain the noise. Therefore, regularization methods involve adding an additional term to the error norm, often called the regularization term. In other words, we replace \mathcal{M} by a new cost function given by

$$J = \mathcal{M}(\mathbf{m}, \mathbf{d}) + \mu\mathcal{R}(\mathbf{m}) \quad (2.6)$$

where $\mathcal{R}(\mathbf{m})$ denotes the regularization term. The scalar μ is the trade-off parameter, which controls the relative strength of the fitting term \mathcal{M} and model regularization term \mathcal{R} .

¹If \mathbf{x} is a vector of length M then, $\|\mathbf{x}\|_2^2 = \sum_k |x_k|^2, k = 1, \dots, M$.

2.2.1 Quadratic regularization

Classical regularization theory (Tikhonov, 1943), often named Tikhonov regularization, adopts quadratic regularization terms of the form

$$\mathcal{R}(\mathbf{m}) = \|\mathbf{W}\mathbf{m}\|_2^2, \quad (2.7)$$

where \mathbf{W} is a suitable matrix of weights. The associated cost function is now given by

$$J = \|\mathbf{d}^{obs} - \mathcal{L}\mathbf{m}\|_2^2 + \mu \|\mathbf{W}\mathbf{m}\|_2^2. \quad (2.8)$$

At this point, an important comment is in order. Equation 2.8 states two goals. One is to minimize the error norm. In other words, we would like to find a model \mathbf{m} that fits the observations. The second term represents a second goal; we are trying to minimize a new variable called $\mathbf{v} = \mathbf{W}\mathbf{m}$. One can think that \mathbf{v} is a new variable representing *bad features* (i.e., oscillatory noise) of \mathbf{m} that one would like to minimize². For instance, in tomographic inversion, \mathbf{W} is a first or second-order discrete derivative operator emphasizing high wavenumbers of \mathbf{m} . Given that high wave-numbers (oscillatory features) are more pronounced on $\mathbf{v} = \mathbf{W}\mathbf{m}$ than on \mathbf{m} , the minimization of $\|\mathbf{v}\|_2$ leads to spatially smooth solutions. Discrete derivatives operators are often adopted as a regularization strategy to estimate smooth slowness model in seismic tomography (Lizarralde and Swift, 1999).

When $\mathbf{W} = \mathbf{I}$ (identity), the cost function reduces to

$$J = \|\mathbf{d}^{obs} - \mathcal{L}\mathbf{m}\|_2^2 + \mu \|\mathbf{m}\|_2^2. \quad (2.9)$$

A closed-form solution to the minimum of equations 2.8 and 2.9 exists when the operator \mathcal{L} can be written in explicit form matrix $\mathcal{L} = \mathbf{L}$. The model-weighted least-squares solution is given by

$$\hat{\mathbf{m}} = (\mathbf{L}^T\mathbf{L} + \mu\mathbf{W}^T\mathbf{W})^{-1}\mathbf{L}^T\mathbf{d}^{obs}. \quad (2.10)$$

Similarly, when $\mathbf{W} = \mathbf{I}$ the minimum of J is given by the damped least-squares solution

$$\hat{\mathbf{m}} = (\mathbf{L}^T\mathbf{L} + \mu\mathbf{I})^{-1}\mathbf{L}^T\mathbf{d}^{obs}. \quad (2.11)$$

In general, when the forward operator is given in the implicit form, one cannot form explicit terms of the form $\mathcal{L}^T\mathcal{L}$ and their associated regularized inverses (equation 2.10 and 2.11). In this case, the cost function J is a sum of two quadratic cost functions, and it can be

²Intuitively \mathbf{W} amplifies *bad features* that one would like to eliminate.

minimized by iterative methods (Golub and Van Loan, 1996) such as steepest descent or conjugate gradients (Hestenes et al., 1952; Scales, 1987). In these methods, one does not need to know operators in their explicit form. They only require the forward operator \mathcal{L} and its adjoint \mathcal{L}^* . More precisely, both the steepest descent and conjugate gradient method can find the minimum of the cost function J iteratively, where in each iteration, one needs to know the action of the forward operator on a vector, for instance, \mathbf{v} of size $M \times 1$, $\mathcal{L} \mathbf{v}$ and the action of the adjoint on a vector \mathbf{v} of size $N \times 1$, $\mathcal{L}^* \mathbf{v}$. Clearly, one needs to ensure that numerically the operators \mathcal{L} and \mathcal{L}^* behave in such a way that one is the adjoint of the other. This seems trivial at first glance, but in reality, as I already mentioned, the operators \mathcal{L} and \mathcal{L}^* are numerical codes. Hence, one must ensure that they behave as a forward-adjoint pair. To guarantee that \mathcal{L}^* is the adjoint of \mathcal{L} , we can use the dot product test (Claerbout, 1992).

2.2.2 Non-quadratic regularization and sparsity

The quadratic regularization ℓ_2 term can be replaced by a non-quadratic term such as the ℓ_1 norm in order to achieve a sparse solution. The problem is often called an $\ell_2 - \ell_1$ optimization problem, and it has received tremendous attention in recent years since the inception of Compressive Sensing (Donoho, 2006) in signal processing.

A typical $\ell_2 - \ell_1$ linear inverse problem entails finding models \mathbf{m} that are sparse. This is achieved by minimizing the following cost (Daubechies et al., 2004, 2009; Beck and Teboulle, 2009)

$$J = \|\mathbf{d}^{obs} - \mathcal{L}\mathbf{m}\|_2^2 + \mu\|\mathbf{m}\|_1 \quad (2.12)$$

where the regularization term is given by an ℓ_1 norm $\|\mathbf{m}\|_1 = \sum_i |m_i|$. It can be shown that the minimization of equation 2.12 leads to sparse solutions. This problem has been used numerous times to solve, for instance, the sparse deconvolution problem (Oldenburg et al., 1983; Sacchi, 1997) to obtain broad-band estimators of the seismic reflectivity. It has also become the cornerstone of methods for seismic data processing, including reconstruction methods for estimating missing observations (Sacchi et al., 1998; Zwartjes and Gisolf, 2007; Herrmann, 2010; Mosher et al., 2014).

A simple example of a reconstruction problem is shown in Figure 2.1, where I used sparse inversion to estimate the Fourier coefficients that synthesize data. For a simple reconstruction problem, the cost function based on equation 2.3 can be rewritten as

$$J = \|\mathbf{d}^{obs} - \mathcal{T}\mathcal{F}\mathbf{m}\|_2^2 + \mu\|\mathbf{m}\|_1, \quad (2.13)$$

where \mathcal{T} is the sampling operator and \mathcal{F} denotes the inverse discrete Fourier transform.

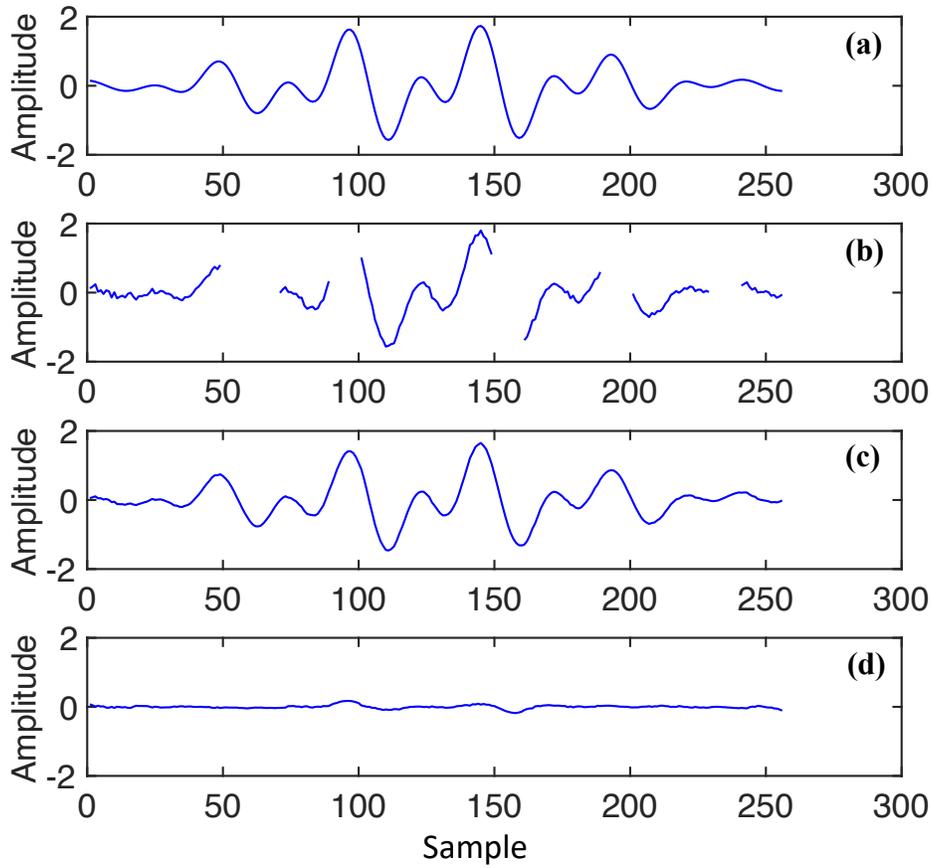


Figure 2.1: Illustration of a toy example for reconstruction based on sparse regularization technique. (a) Ideal signal. (b) Observed signal. (c) Reconstructed signal via sparse inversion of its Fourier coefficients with resulting $SNR = 22.16$ dB. (d) Estimated errors between (c) and (a).

The observed data \mathbf{d}^{obs} has missing observations. In other words, one can say that the ideal idea \mathbf{d} is given by $\mathbf{d}^{obs} = \mathcal{T}\mathbf{d}$. Similarly one also can assume that the ideal data \mathbf{d} can be synthesized via the inverse Fournier transform via $\mathbf{d} = \mathcal{F}\mathbf{m}$. Clearly, the model parameters for this example \mathbf{m} now represent unknown Fourier coefficients that model the ideal data. In essence, one assumes that the distribution of Fourier coefficients that model the data is sparse. Evidently, we have two operators that could be expressed as one $\mathcal{L} = \mathcal{T}\mathcal{F}$. The associated adjoint operator is given by $\mathcal{L}^* = \mathcal{F}^*\mathcal{T}^*$ where \mathcal{F}^* and \mathcal{T}^* are the adjoint of the Fourier synthesis operator and sampling operator, respectively.

Figure 2.1a shows the ideal data before decimation. Similarly, Figure 2.1b shows the decimated data $\mathbf{d}^{obs} = \mathcal{T}\mathbf{d}$. Finally, Figure 2.1c shows the reconstructed data obtained via the Fourier synthesis $\mathcal{F}\hat{\mathbf{m}}$ where $\hat{\mathbf{m}}$ are the Fourier coefficients estimated by minimizing expression 2.13. The missing samples have been fully reconstructed with the non-quadratic sparse inversion. Figure 2.1d is the estimated error between the ideal data and the reconstructed data. The signal-to-noise ratio of the reconstruction quality is measured by the following equation:

$$SNR = 10 \log \frac{\|\mathbf{d}^{true}\|_F^2}{\|\mathbf{d}^{true} - \mathbf{d}^{rec}\|_F^2}, \quad (2.14)$$

where \mathbf{d}^{true} is the ideal data without decimation and \mathbf{d}^{rec} is the reconstructed data. The symbol $\|\cdot\|_F$ denotes the Frobenius norm.

2.3 Projected Gradient Descent (PGD) method

Alternatively, the projected gradient method can also be used for solving linear inverse problems described in preceding sections (Piana and Bertero, 1997). To incorporate such constraints, one can reformulate equation 2.3 as a constrained least-squares problem

$$\begin{aligned} &\text{minimize} && J = \|\mathbf{d}^{obs} - \mathcal{L}\mathbf{m}\|_2^2 \\ &\text{subject to} && \mathbf{m} \in \mathbb{S}. \end{aligned} \quad (2.15)$$

where the solution \mathbf{m} must belong to a class of solutions that belong to the set \mathbb{S} .

For this constrained least-squares problem, the PGD method is a simple algorithm to solve equation 2.15 and, in the next chapters, I will use similar ideas to solve the simultaneous-source separation problems.

In our context of this chapter, we will declare that there is a projection operator \mathcal{P} such that for any arbitrary signal \mathbf{x} , there is a new signal $\mathbf{y} = \mathcal{P}[\mathbf{x}]$ such that $\mathbf{y} \in \mathbb{S}$. I describe the PGD method in the following paragraphs. First, one performs an update via a step in the negative direction of the gradient without considering the projection. Then we apply

the projection. The following iteration gives the solution to equation 2.15.

$$\begin{aligned}\mathbf{m}^\nu &= \mathcal{P} [\mathbf{m}^{\nu-1} - \lambda \nabla J (\mathbf{m}^{\nu-1})] \\ &= \mathcal{P} [\mathbf{m}^{\nu-1} - \lambda \mathcal{L}^* (\mathcal{L} \mathbf{m}^{\nu-1} - \mathbf{d}^{obs})]\end{aligned}\tag{2.16}$$

where ∇J is the gradient of the cost function J , and ν denotes the iteration number. Parameter λ is the length of step-size which must be chosen to guarantee the convergence of the PGD method (Cheng and Sacchi, 2015; Bolduc et al., 2017). It is clear from the above expression that PGD entails one step of the steepest descent method followed by the projection operator \mathcal{P} .

2.3.1 Step-size λ selection

Classical convergence proofs of PGD rely on assuming a convex projection, but, in general, convexity can be ignored for practical applications, and numerical experiments can be used instead of analytical methods for the analysis of convergence. As I mentioned, convexity helps to obtain convergence proofs (Cheng, 2017). However, it is important to point out that, in general, one can apply the PGD method even in situations where the projection is non-convex (Peters et al., 2019). In this case, the step-size λ needs to be selected carefully to guarantee the convergence of the PGD method (Cheng and Sacchi, 2015; Bolduc et al., 2017). In equation 2.16, the initial step-size can be properly selected (Fazel, 2002) to ensure that the iterative algorithm converges. For a convex projection, the convergence can be guaranteed when the initial step-size is given by

$$0 < \lambda_0 < \frac{2}{e_{max}},\tag{2.17}$$

where e_{max} is the maximum eigenvalue of the operator $\mathcal{L}^* \mathcal{L}$. This eigenvalue can be computed by the power iteration method by starting with an initial random guess for the eigenvector associated with the maximum eigenvalue (Ma et al., 2011; Cheng and Sacchi, 2015). We use the same starting value for non-convex projections, but then we exponentially decreased the step length with iteration to converge to the solution. Another strategy is to optimally find each iteration's step length via a backtracking line search (Bolduc et al., 2017). In the following chapters, more details about step-length selection are discussed in the particular examples for the deblending problem.

2.3.2 A projection operator based on a low-rank approximation

My thesis uses the Multichannel Singular Spectrum Analysis (MSSA) filter as a projection operator. In other words, \mathcal{P} in equation 2.16 is given by an MSSA filter. The one-

dimensional version, SSA filter, was proposed to analyze one-dimensional time series analysis and extended to a multichannel case (MSSA) for signals that depend on time and space, such as the case of seismic signals. Here I briefly review the simple SSA filter in the way it is applied for time series or one-dimensional signals (Vautard and Ghil, 1989; Vautard et al., 1992). We can use an arbitrary signal \mathbf{x} to indicate the input signal to the SSA filter. Similarly, the projected or filtered signal is indicated by $\mathbf{y} = \mathcal{P}[\mathbf{x}]$. The SSA filter entails the following steps;

1. *Hankelization*

First, the Hankel matrix is formed. The data \mathbf{x} , of length N , $\mathbf{x} = [x(1), x(2), \dots, x(N)]^T$ is embedded into a Hankel matrix

$$\begin{aligned} \mathbf{H} &= \mathcal{H}[\mathbf{x}] \\ &= \begin{pmatrix} x(1) & x(2) & \dots & x(N_c) \\ x(2) & x(3) & \dots & x(N_c + 1) \\ x(3) & x(4) & \dots & x(N_c + 2) \\ \vdots & \vdots & \vdots & \vdots \\ x(N_r) & x(N_r + 1) & \dots & x(N) \end{pmatrix} \end{aligned} \quad (2.18)$$

with $N_r = \lfloor N/2 \rfloor + 1$ and $N_c = N - N_r$. The Hankel \mathbf{H} is square for N odd or almost square for N even. The symbol \mathcal{H} denotes the Hankelization operator.

2. *Rank-reduction*

Then, we apply rank reduction via the Singular Value Decomposition (SVD) to estimate the rank p matrix \mathbf{H}_p that minimizes the Frobenius norm $\|\mathbf{H} - \mathbf{H}_p\|_F$ (Eckart and Young, 1936). The SVD decomposition yields the orthonormal matrices \mathbf{U}_p and \mathbf{V}_p of size $N_r \times p$ and $N_c \times p$, respectively, and the $p \times p$ diagonal matrix of singular values $\mathbf{\Sigma}_p$. The reduced-rank approximation is given by $\mathbf{H}_p = \mathbf{U}_p \mathbf{\Sigma}_p \mathbf{V}_p^H$, and this process is called rank-reduction, which can be denoted as \mathcal{R} . Clearly, p is the desired rank of the approximation supplied by the user.

3. *Anti-diagonal averaging*

The filtered signal is recovered via anti-diagonal averaging of the elements of the matrix \mathbf{H}_p

$$\mathbf{y} = \mathcal{A}[\mathbf{H}_p] \quad (2.19)$$

where \mathcal{A} is the anti-diagonal averaging operator, which can be expressed as the fol-

lowing equation:

$$y_i = \begin{cases} \frac{1}{i} \sum_{j=1}^i \hat{H}_{j,i-j+1} & 1 \leq i \leq N_c \\ \frac{1}{N_c} \sum_{j=1}^{N_c} \hat{H}_{j,i-j+1}, & N_c \leq i \leq N_r, \\ \frac{1}{N-i+1} \sum_{j=i-N_r+1}^{N_c} \hat{H}_{j,i-j+1}, & N_r \leq i \leq N, \end{cases} \quad (2.20)$$

Therefore, the operator \mathcal{P} contains three steps, *Hankelization*, *Rank-reduction*, *Anti-diagonal averaging*, which can be summarized as a single operator

$$\mathbf{y} = \mathcal{P}[\mathbf{x}] \equiv \mathcal{A}\mathcal{R}\mathcal{H}[\mathbf{x}]. \quad (2.21)$$

In Figure 2.2, I adopted PGD in conjunction with the SSA filter to reconstruct a series. As in comparing Figure 2.1, we have a signal that has been decimated, and the task is to estimate the whole signal. The underlying assumption is that the ideal complete signal can be represented via the rank p Hankel matrix. It is possible to show that signals consisting of p complex sinusoids lead to Hankel matrices of rank p (Oropeza, 2010). In essence, our problem is given by

$$\begin{aligned} & \text{minimize} && J = \|\mathbf{d}^{obs} - \mathcal{T}\mathbf{m}\|_2^2 \\ & \text{subject to} && \mathbf{m} \in \mathbb{S}. \end{aligned} \quad (2.22)$$

where the solution \mathbf{m} is now the desired signal and, as before, \mathcal{T} is the sampling operator. If an arbitrary signal \mathbf{x} is filtered by the SSA filter, $\mathbf{y} = \mathcal{P}[\mathbf{x}]$, then we say that \mathbf{y} belongs to the set \mathbb{S} . The problem is solved via the following iterative algorithm

$$\begin{aligned} \mathbf{m}^\nu &= \mathcal{P} [\mathbf{m}^{\nu-1} - \lambda \nabla J (\mathbf{m}^{\nu-1})] \\ &= \mathcal{P} [\mathbf{m}^{\nu-1} - \lambda \mathcal{T}^* (\mathcal{T}\mathbf{m}^{\nu-1} - \mathbf{d}^{obs})] . \end{aligned} \quad (2.23)$$

Figure 2.2a shows the ideal signal, Figure 2.2b shows the signal with missing data, and Figure 2.2c shows the reconstructed data via PGD with the SSA filter. Figure 2.2d shows the estimated errors between the ideal signal and the reconstructed signal. The results are equivalent to those we obtained with sparse inversion (Figure 2.1). However, it is essential to stress the difference between adopting sparse inversion (discussed in section 2.2.2) and the PGD method. We adopted the regularization method in the sparse inversion approach to finding sparse Fourier coefficients that synthesize the ideal observed data. In the PGD approach, we use the steepest descent method followed by a projection that directly acts on the data. In other words, in the PGD method, we do not require to operate on an auxiliary domain as in the case of sparse inversion. Another compelling feature of PGD

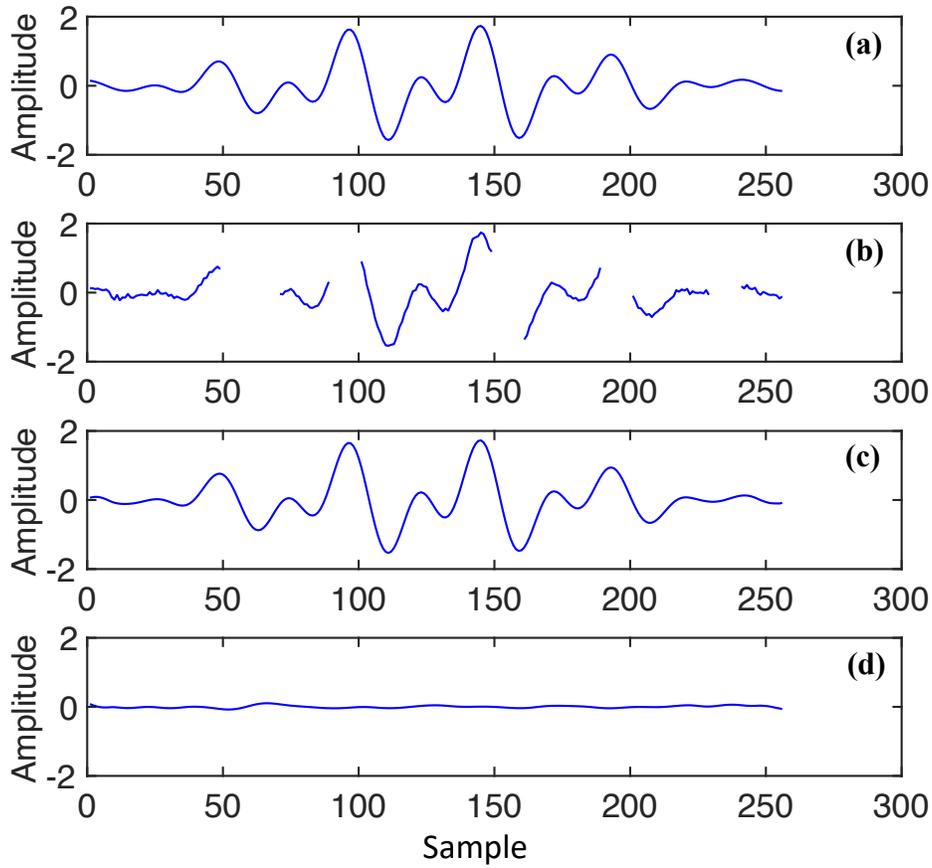


Figure 2.2: Illustration of a toy example for reconstruction based on projection with constrained low-rank optimization. (a) Ideal signal. (b) Observed signal. (c) Reconstructed signal via PGD method based on constrained low-rank optimization with resulting $SNR = 24.95$ dB. (d) Estimated errors between (c) and (a).

is that the projection can be any filters or denoisers capable of forcing desired features on the signal to recover, and eliminate the bad features. The approach is not restricted to the SSA filter; other denoisers such as median or prediction error filters could be used as projections. In the following chapters, I will show how I adopt different denoisers for separating simultaneous-source data.

CHAPTER 3

Deblending via a coherence pass robust Radon projection operator ¹

3.1 Introduction

In simultaneous source separation, in the category of inversion-based methods, sparse, non-robust inversion methods are generally adopted for iteratively deblending (Abma et al., 2010; Song et al., 2019; Lin and Sacchi, 2020a; Chen et al., 2014; Wang and Geng, 2019). The deblending algorithm is developed by minimizing a cost function with a data fidelity ℓ_2 error function and a sparse model ℓ_1 norm regularization. I expand these ideas by developing a new sparsity-driven and robust inversion algorithm acting as the coherence pass projection operator for deblending.

This chapter proposes an iterative method for simultaneous source separation suitable for data acquired with a high blending factor. The proposed technique adopts the robust and sparse Radon transform to define a coherence pass operator used in conjunction with the steepest descent method to guarantee solutions that honour simultaneous source records. We show that an essential improvement in convergence is attainable when the coherence pass projection is derived from a robust sparse Radon transform. This is a consequence of having an iterative deblending algorithm that applies intense denoising to erratic blending noise in its initial iterations. The coherence pass robust Radon operator acts as a data projection operator that preserves coherent signals and annihilates incoherent blending noise right from the start of the iterative process. We compare the algorithm with its non-robust version

¹A version of this chapter is published in Lin, R., and M. D. Sacchi, 2020, Separation of simultaneous sources acquired with a high blending factor via coherence pass robust Radon operators: *Geophysics*, **85**, no. 3, V269-V282.

and show that a coherence-pass non-robust Radon operator will only achieve high-quality results for acquisitions with a moderate blending factor.

This chapter follows the subsequent structure. First, I introduce the notation adopted and provide preliminary definitions of deblending via inversion with a coherence pass projection. I continue with a description of a comparison of the coherence pass non-robust and robust Radon projection operators. Finally, I test the proposed method via synthetic and field data with a high blending factor.

3.2 Theory

3.2.1 Deblending via inversion with a coherence pass projection

We minimize two objective functions simultaneously, one associated with data fidelity, and the other with a coherence pass operator that permits eliminating blending noise in common receiver gathers or in common channel gathers (Peng et al., 2013; Abma et al., 2015). We first consider the data fidelity constraint, which is given by the following cost function

$$J_d = \|\mathbf{b} - \mathcal{B}\mathbf{D}\|_2^2 + \mu_1 \|\mathbf{D}\|_2^2. \quad (3.1)$$

The expression above simply states that the unknown deblended data must be able to reproduce simultaneous source records. Notice we have also added a quadratic regularization term controlled by a trade-off parameter μ_1 . The second cost function ensures that the deblended common receiver gathers can be represented in terms of Radon domain coefficients

$$J_i = \|\mathbf{D}_i - \mathcal{R}\mathbf{m}_i\|_p^p + \mu_2 \|\mathbf{m}_i\|_1, \quad i = 1 \dots N_r, \quad (3.2)$$

where \mathbf{m}_i denotes the Radon coefficients representing the common receiver i . Similarly, \mathcal{R} represents the forward Radon transform which in our case is a time-domain linear Radon transform (Trad et al., 2001). Equation 3.2 is also equipped with a regularization term that promotes sparse Radon coefficients. The parameter μ_2 controls the level of the sparsity of the solution and the fit between synthesized data $\mathcal{R}\mathbf{m}_i$ and \mathbf{D}_i .

When $p = 1$, equation 3.2 denotes the robust Radon transform with a constraint to encourage sparse Radon coefficients. The case corresponding to $p = 2$ in equation 3.2 denotes the non-robust sparse Radon transform. Robust and non-robust Radon transforms can be used as coherence pass operators. The coherence pass operator cannot be written in a closed-form solution and, therefore, it will be indicated via the data projection operator \mathcal{P}_c that results from estimating the Radon coefficients and synthesizing data for each common receiver

gather

$$\hat{\mathbf{m}}_i = \underset{\mathbf{m}_i}{\operatorname{argmin}} \|\mathbf{D}_i - \mathcal{R}\mathbf{m}_i\|_p^p + \mu_2 \|\mathbf{m}_i\|_1^1 \quad (3.3)$$

$$\begin{aligned} \hat{\mathbf{D}}_i &= \mathcal{R} \hat{\mathbf{m}}_i \\ &= \mathcal{P}_c \{\mathbf{D}_i\} . \end{aligned} \quad (3.4)$$

Without losing generality, we also define the coherence pass operator $\hat{\mathbf{D}} = \mathcal{P}_c \{\mathbf{D}\}$ to indicate the process of computing Radon coefficients and synthesizing data for all common receiver gathers $i = 1 \dots N_r$.

The coherence pass operator can be computed via the sparse non-robust Radon transform ($p = 2$) or the sparse robust Radon transform ($p = 1$). We will demonstrate that the sparse robust Radon transform ($p = 1$) yields an algorithm that can ensure a high-quality source separation for acquisitions with a high blending factor. Both operators produce similar results for a moderate blending factor (e.g., a blending factor of two). Interestingly, the solution via the coherence pass non-robust sparse Radon operator is very similar to the solution one can estimate using deblending via sparse inversion (Abma et al., 2010; Chen et al., 2014).

The problem stated by equations 3.1 and 3.3 can be solved iteratively

$$\mathbf{Z} = \mathbf{D}^k - \lambda [\mathcal{B}^* (\mathcal{B}\mathbf{D}^k - \mathbf{b}) + \mu_1 \mathbf{D}^k] \quad (3.5)$$

$$\mathbf{D}^{k+1} = \mathcal{P}_c \{\mathbf{Z}\} , \quad (3.6)$$

where the supra-index k denotes iteration. Equation 3.5 corresponds to a steepest descent update with step size λ (Bertsekas, 1997). The parameter λ is chosen to ensure convergence via the expression $\lambda < 2/e_{max}$ where e_{max} is the maximum eigenvalue of the operator $\mathcal{B}^* \mathcal{B}$ (Ma et al., 2011). The power method is used to iteratively estimate e_{max} (Golub and Van Loan, 1996). As already stated, the coherence pass operator \mathcal{P}_c entails computing Radon coefficients and synthesizing data for each common receiver gather. Algorithm 1 summarizes the proposed deblending procedure. For all our numerical tests, we set a maximum number of iterations `Max_Outer_Iter`= 40. Stopping criteria could have been added to limit the number of iterations. We have preferred, however, to evaluate results for a fixed number of iterations to study convergence differences between the coherence pass non-robust and robust Radon operators.

Algorithm 1 Deblending via inversion

```

1: Inputs:
     $\mathbf{b}, \mu_1, \lambda.$ 
2: Initialization:
     $k = 0, \mathbf{D}^k = \mathcal{B}^* \mathbf{b}.$ 
3: for  $k = 1 : \text{Max\_Outer\_Iter}$  do
4:    $J_d^k = \|\mathcal{B}\mathbf{D}^k - \mathbf{b}\|_2^2 + \mu_1 \|\mathbf{D}^k\|_2^2$            Calculate cost function to check convergence
5:    $\mathcal{Z} = \mathbf{D}^k - \lambda[\mathcal{B}^*(\mathcal{B}\mathbf{D}^k - \mathbf{b}) + \mu_1 \mathbf{D}^k]$        Steepest descent update
6:    $\mathbf{D}^{k+1} = \mathcal{P}_c \mathcal{Z}$                                        Coherence pass operator (Algorithm 2)
7: end for
    
```

3.2.2 Coherence pass non-robust and robust Radon projection operators

This section describes the methodology to compute the Radon transform coefficients used to define coherence pass operators. For simplicity, we consider one common receiver gather \mathbf{D}_i and ignore the subindex i with the understanding that the procedure must be applied to all common receiver gathers. We use the method described by Ibrahim and Sacchi (2013) who adopted the iterative reweighed least-squares (IRLS) method. (A more efficient framework named the Alternating Direction Method of Multipliers (ADMM) is described in Appendix A. Also, a comparative study of IRLS and ADMM algorithms is described in Appendix A.)

We again state the problem of estimating sparse Radon coefficients under a ℓ_p measure of error. The Radon coefficients are estimated by minimizing the cost function

$$J = \|\mathbf{D} - \mathcal{R}\mathbf{m}\|_p^p + \mu_2 \|\mathbf{m}\|_1. \quad (3.7)$$

The minimum of equation 3.7 can be computed by sequential minimization of the following quadratic cost function (Trad et al., 2003)

$$J_q = \|\mathbf{W}_r \circ (\mathbf{D} - \mathcal{R}\mathbf{m})\|_2^2 + \mu_2 \|\mathbf{W}_m \circ \mathbf{m}\|_2^2, \quad (3.8)$$

where the symbol \circ indicates element-wise multiplication. The matrix \mathbf{W}_r is given by

$$[\mathbf{W}_r]_{ij} = \begin{cases} \frac{1}{|r_{ij}|^{2-p+\epsilon_r}}, & \text{if } p < 2 & \text{Robust} \\ 1, & \text{if } p = 2 & \text{Non-robust} \end{cases} \quad (3.9)$$

where r_{ij} denotes the element ij of the residual $\mathbf{r} = \mathbf{D} - \mathcal{R}\mathbf{m}$.

Significant errors are down-weighted when $p < 2$, leading to solutions that emphasize sparse residuals consistent with erratic blending noise. The synthesis of the common receiver

gather from sparse Radon coefficients will produce coherent events. Hence, the error at each iteration should contain blending noise. When adopting a norm with $p = 2$, the method will incorporate the blending noise into the Radon solution, and therefore, the algorithm will be less efficient in reducing blending noise. On the other hand, with $p = 1$ the blending noise is modelled correctly; therefore, the algorithm requires less effort (iterations) to reach an adequately deblended solution. The matrix \mathbf{W}_m is given by

$$[\mathbf{W}_m]_{ij} = \frac{1}{|m_{ij}| + \epsilon_m} \quad (3.10)$$

where m_{ij} denotes the element ij of \mathbf{m} . The elements of the matrix \mathbf{W}_m represent weights that, during the iterative inversion, are responsible for producing sparse Radon coefficients (Trad et al., 2003). We clarify that small numbers ϵ_r and ϵ_m are needed to avoid division by zero. For our tests we have adopted $\epsilon_r = \epsilon_m = 10^{-6}$. We stress that the robust case with $p = 1$ corresponds to assuming that the residuals $\mathbf{r} = \mathbf{D} - \mathcal{R}\mathbf{m}$ are sparse.

To further simplify the problem, we reduce the cost function to its standard form by the following change of variables $\mathbf{u} = \mathbf{W}_m \circ \mathbf{m}$

$$J_q = \|\mathbf{W}_r \circ (\mathbf{D} - \mathcal{R}\mathbf{P}_m \circ \mathbf{u})\|_2^2 + \mu_2 \|\mathbf{u}\|_2^2, \quad (3.11)$$

where $[\mathbf{P}_m]_{ij} = 1/[\mathbf{W}_m]_{ij}$ and $\mathbf{m} = \mathbf{P}_m \circ \mathbf{u}$. The cost function given by equation 3.11 is minimized via the method of Conjugate Gradient for Least Squares (CGLS) (Scales, 1987). We remind the reader that the time-domain Radon operator \mathcal{R} and its adjoint \mathcal{R}^* are provided via numerical algorithms (implicit operators) rather than explicit matrices. Therefore, one can only evaluate their action on Radon coefficients ($\mathcal{R}\mathbf{x}$) or data ($\mathcal{R}^*\mathbf{y}$) where \mathbf{x} and \mathbf{y} symbolize arrays of dimensions $Dim(\mathbf{m})$ and $Dim(\mathbf{D})$, respectively. Fortunately, the CGLS algorithm does not require matrices in explicit form, and it directly works with implicit operators \mathcal{R} and \mathcal{R}^* (Claerbout, 1992). The solution \mathbf{u} is used to compute $\mathbf{m} = \mathbf{P}_m \circ \mathbf{u}$ and the residuals \mathbf{r} which then update the weights \mathbf{W}_r and \mathbf{P}_m (Ibrahim and Sacchi, 2013). Algorithm 2 shows the coherence-pass Radon operator. Similarly, we also provide the IRLS solver in Algorithm 3. The whole workflow can be found in Figure 3.1.

Trad et al. (2003) pointed out that one can ignore the trade-off parameter μ_2 in equation 3.11 and utilize the number of iterations of the conjugate gradient method to avoid overfitting the data. Our experience with various numerical tests indicates that regularization by iteration is not always advisable. Regularization by iteration can lead to an unstable update of the weights, preventing the IRLS algorithm's convergence. Therefore, we have preferred to run the conjugate gradient algorithm with a predefined value of μ_2 until a convergence criterion is satisfied. The norm of the gradient of J_q at a given iteration relative to its initial value ($|\nabla J_q^k|/|\nabla J_q^0| < \eta_{cglS}$) is adopted as a stopping criterion for CGLS. The external iteration

Algorithm 2 Coherence pass Radon Operator \mathcal{P}_c

```

1: Inputs:
    $\mathbf{D}, \mu_2, p.$ 
2: Initialization:
    $k = 0, \mathbf{D}^k = \mathcal{B}^* \mathbf{b}.$ 
3: for  $i = 1 : N_r$  do
4:    $\mathbf{D}_i = \mathbf{D}[:, i, :]$  Extract one CRG
5:    $\mathbf{m} = \text{irls}(\mathbf{D}_i, \mathcal{R}, \mathcal{R}^*, p, \mu_2)$  Radon coefficients (Algorithm 3)
6:    $\hat{\mathbf{D}}_i = \mathcal{R} \mathbf{m}$  Synthesize the CRG
7:    $\hat{\mathbf{D}}[:, i, :] = \hat{\mathbf{D}}_i$  Reinsert CRG in data volume
8: end for
    
```

Algorithm 3 Sparse robust and non-robust Radon via IRLS

```

1: Inputs:
    $\mathbf{D}, \eta, \mu_2.$ 
2: Initialization:
    $k = 0, \mathbf{W}_m = \text{ones}, \mathbf{W}_r = \text{ones}.$ 
3: while  $|J^{k+1} - J^k|/|J^k| \geq \eta_{irls}$  &  $k \leq \text{Max\_IRLS\_Iter}$  do
4:    $\mathbf{u} = \text{cgls}(\mathbf{D}, \mathcal{R}, \mathcal{R}^*, \mathbf{W}_r, \mathbf{P}_m, \mu_2)$  Solve equation 3.11
5:    $\mathbf{m} = \mathbf{P}_m \circ \mathbf{u}$ 
6:    $\mathbf{r} = \mathbf{D} - \mathcal{R} \mathbf{m}$  Compute residuals
7:    $\mathbf{P}_m = \text{Model\_weights}(\mathbf{m})$  Weights for Radon coefficients
8:    $\mathbf{W}_r = \text{Residual\_weights}(\mathbf{r}, p)$  Weights for residuals
9:    $J^k = \|\mathbf{r}\|_p^p + \mu_2 \|\mathbf{m}\|_1$  Cost function
10: end while
    
```

of the IRLS algorithm (iteration to update the weights) stops when a maximum number of updates is reached (typically five or six iterations) or when $|J^k - J^{k-1}|/|J^k| \leq \eta_{irls}$.

3.3 Examples

In this section, we test the proposed algorithm with one numerically blended synthetic example with varying blending factors and a real seismic data example. We will use the following expression to measure the signal-to-noise ratio of the reconstruction

$$SNR = 10 \log \frac{\|\mathbf{D}^{true}\|_F^2}{\|\mathbf{D}^{true} - \mathbf{D}^{rec}\|_F^2}, \quad (3.12)$$

where \mathbf{D}^{true} is the true data prior to blending and \mathbf{D}^{rec} is the recovered data after deblending. The symbol $\|\cdot\|_F$ denotes the Frobenius norm.

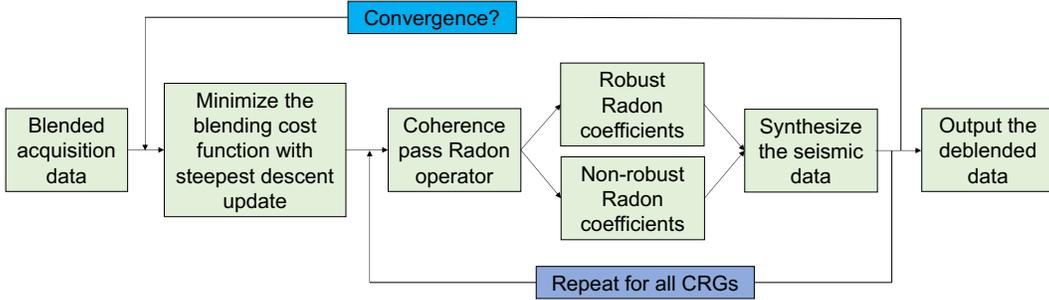


Figure 3.1: The workflow of the whole deblending process based on coherence pass Radon projection operators.

3.3.1 Synthetic example

Our first example uses synthetic data modelled to test the proposed deblending method. We first synthesize an example containing five linear events to mimic a small patch of a noise-free common receiver gather. In this example, we committed the so-called “inverse problem crime” because we have synthesized the linear events in the common receiver gathers by forward modelling Radon domain coefficients. Then, the synthesized common receiver gathers were organized in shot gathers and blended. Firing times correspond to uniformly distributed random numbers scaled to produce a desired blending factor. The numerical example consists of $N_r = 60$ receivers and $N_s = 80$ shots. For each common receiver gather, we extract every second shot from the 80 shots for display purposes. The source wavelet was synthesized with a Ricker wavelet of central frequency 30 Hz. We also point out that we have adopted a time-domain linear Radon transform that operates on the full aperture of each common receiver gather. In our next example, we adopt a local time-domain linear Radon transform. This type of unusual synthetic has as its main goal testing the algorithm and tuning trade-off parameters. Bear in mind that similar synthetic tests have been proposed by several authors (Akerberg et al., 2008; Ibrahim and Sacchi, 2013; Chen et al., 2014; Cheng and Sacchi, 2015). The trade-off parameters of the problem were set to $\mu_1 = 0.1$ and $\mu_2 = 1$. The maximum number of iterations of IRLS was set to 5 and $\eta_{irls} = 1.0^{-6}$. Similarly, the maximum number of iterations for the CGLS solver was set to 30 and the parameter $\eta_{cglS} = 1.0^{-6}$.

Figure 3.2a shows the original unblended common receiver gather. The pseudo-deblended data with a blending factor (defined with equation 1.2) of two is shown in Figure 3.2b. The deblended result obtained via iterative inversion with the non-robust sparse Radon transform as coherence pass operator can be found in Figure 3.2c. The deblended result

with the coherence-pass robust sparse Radon operator can be found in Figure 3.2d. Figure 3.2e shows the level of blending noise in common receiver gather. Figure 3.2f and 3.2g shows the difference between 3.2c and 3.2a and 3.2d and 3.2a, respectively. In this case, Figure 3.2e represents the blending noise for a blending factor of two. In Figure 3.2f and 3.2g, we observe that the blending noise is properly suppressed, and the error panel contains almost no signal, which is almost the same as the unblended data. This is true for the proposed deblending algorithm for both the coherence pass robust and non-robust Radon operators. In other words, both methods produce similar results for moderate levels of blending noise. Figure 3.3 shows the SNR of the deblending process versus the outer iteration of the algorithm for the coherence pass non-robust and robust Radon operators, and the 0-th iteration denotes the initial level of blending noise. It is clear that both algorithms achieve identical results for a blending factor of two. One can point out that the coherence-pass robust Radon operator eliminates more blending noise early in the iterative process and achieves convergence to $SNR = 45$ dB earlier than the iterative inversion that uses the coherence pass non-robust Radon operator. The difference is insignificant for this case but yet relevant to be mentioned.

We also tested the proposed algorithm with a higher blending factor of six, which consequently yields higher contamination with blending noise. Figure 3.4a presents one clean unblended common receiver gather, and Figure 3.4b shows the pseudo-deblended common receiver gather. Figure 3.4c and 3.4d shows the deblended results by adopting coherence pass non-robust and robust Radon operators, respectively. The estimation error sections can be found in Figure 3.4e-3.4g. In Figure 3.4e, we notice that the blending noise is more energetic than that in Figure 3.2e. Figure 3.4f and 3.4g shows the coherence-pass robust Radon operator outperforms its non-robust counterpart. The error in Figure 3.4g shows no signal, while in Figure 3.4f, a moderate residual error is visible. In Figure 3.5, we can observe that the coherence pass robust Radon operator achieves an $SNR = 40$ dB after 40 iterations. The non-robust operator has achieved an $SNR = 20$ dB for the same number of iterations. Again, it is evident that the robust operator is more efficient at removing blending noise at early iterations than the non-robust operator.

Our last test considers an unusual level of blending of a blending factor of ten (Figure 3.6). This is a challenging problem, as one can observe from the pseudo-deblended common receiver gather in Figure 3.6b. In Figure 3.6f and 3.6g, we observe that the coherence pass robust Radon operator generates a more accurate deblending than the coherence pass non-robust Radon operator. Figure 3.7 shows values of the SNR versus the outer iteration of our algorithm for coherence-pass non-robust and robust Radon operators. Evidently, the robust operator can eliminate blending noise during initial iterations and almost reaches an $SNR = 40$ dB after 40 iterations.

Our last figure for this example (Figure 3.8) displays the SNR versus blending factor for

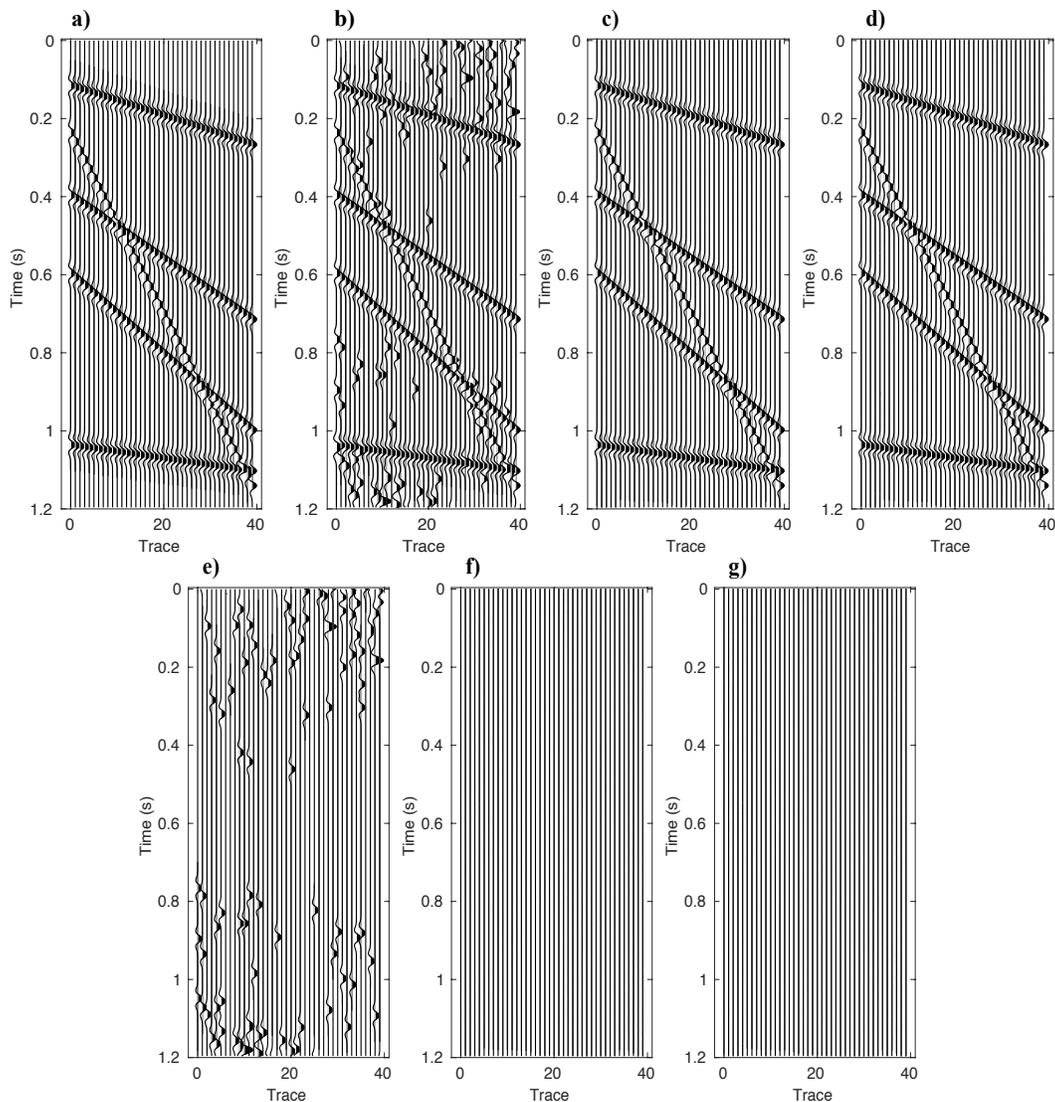


Figure 3.2: Deblending of synthetic data in common receiver gather with blending factor of two. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

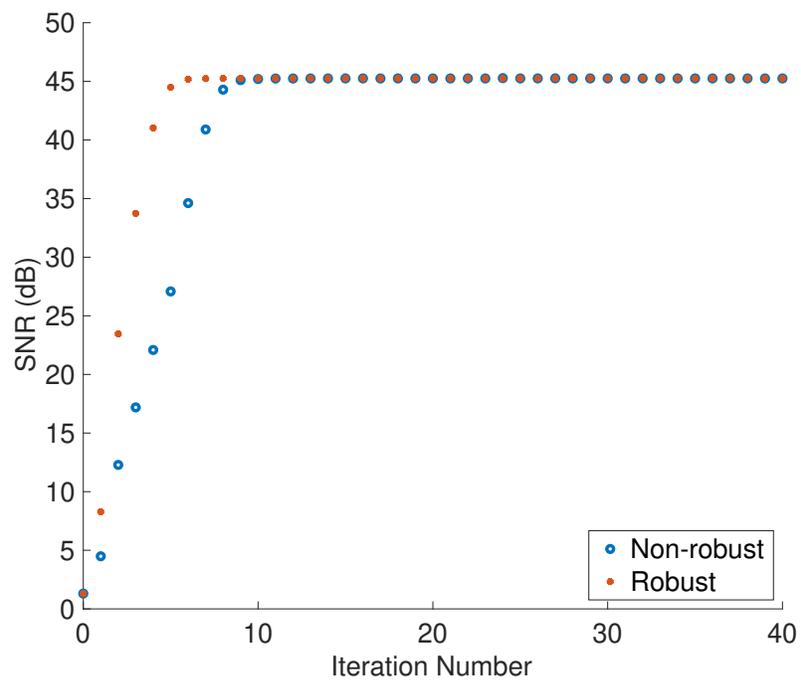


Figure 3.3: Synthetic data example. SNR versus iteration number for blending factor of two.

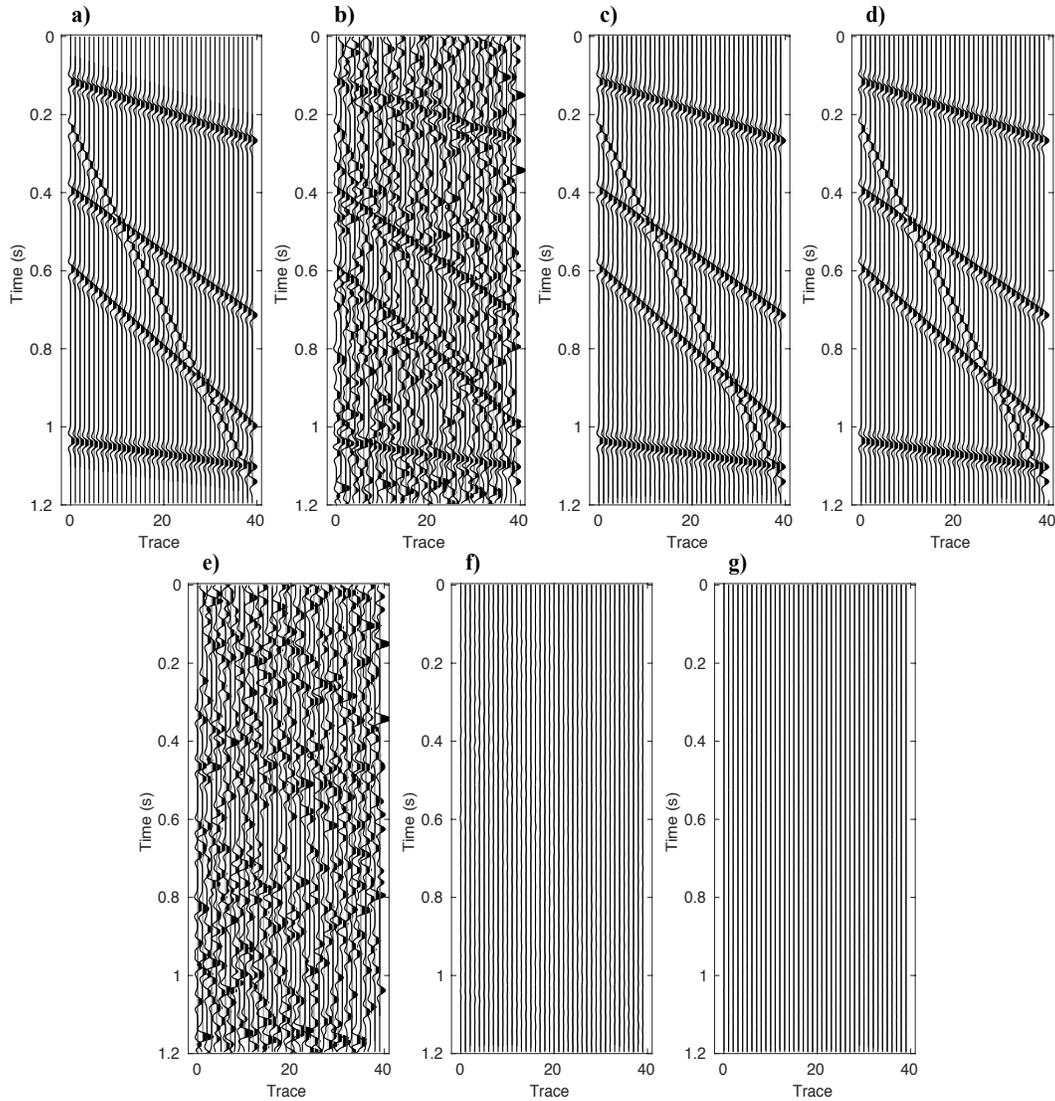


Figure 3.4: Deblending of synthetic data in common receiver gather with blending factor of six. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

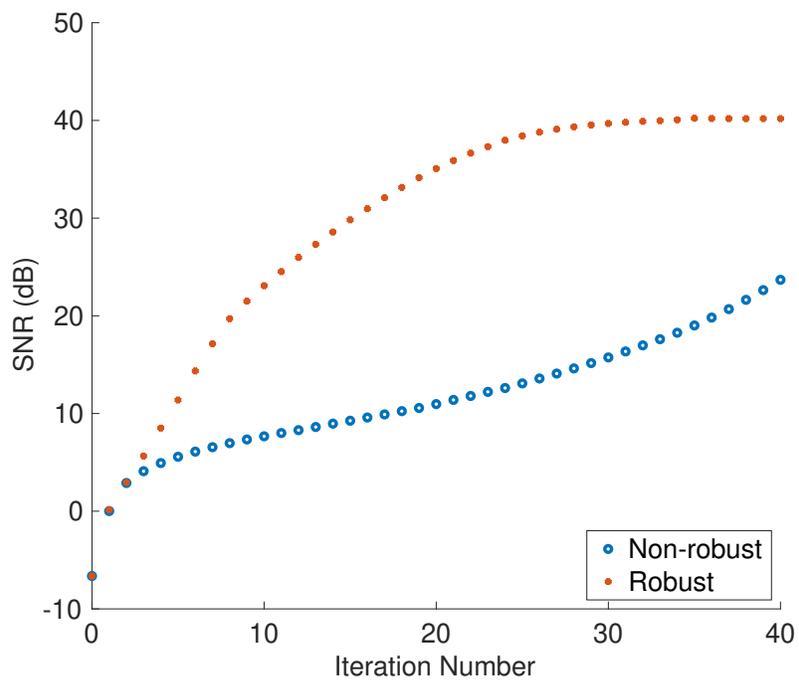


Figure 3.5: Synthetic data example. SNR versus iteration number for blending factor of six.

the coherence pass robust and non-robust Radon operators. All our results correspond to a fixed final number of iterations `Max_Outer_Iter` = 40 (Algorithm 1). We emphasize that we could have improved the results corresponding to the coherence-pass non-robust operator by increasing the number of iterations at the cost of augmenting computational time. Our examples, including convergence curves, confirm that the coherence pass robust Radon operator has an early action on the deblending process leading to faster convergence to solutions with a high *SNR* figure of merit.

3.3.2 Real data example

We also test our method with a marine dataset from the Gulf of Mexico (Mississippi Canyon data). The data were numerically blended with varying blending factors = 2, 4, 6, 8, 10 to extract *SNR* figures of quality.

We selected $N_s = 33$ shot gathers to simulate streamer data acquired via a simultaneous source acquisition with $N_r = 92$ receivers extracted from every second trace of gathers with 183 receivers. In this case, the coherence pass operator is applied in common channel gathers (Peng et al., 2013). Contrary to our previous example, where we adopted a linear Radon transform operating on the full aperture of the gather, we use a local linear Radon transform instead. Common channel gathers have events that cannot be modelled with a linear Radon transform that operates on the whole common channel gather. Therefore, we adopted a local linear Radon transform that uses 11 consecutive traces with an overlap of three traces. The results are patched back with proper tapering in the areas of overlap to avoid artifacts. In other words, the coherence pass Radon operator \mathcal{P}_c corresponds to a sparse local linear Radon transform with non-robust or robust data fidelity term followed by data synthesis with proper superposition of spatially overlapping windows. For the field data example, we set $\mu_2 = 0.5$ for the coherence pass robust Radon operator and $\mu_2 = 0.01$ for the non-robust one. The other parameters were kept unchanged. Parameter selection was carried out heuristically by selecting values of μ_2 that lead to a minimum amount of signal leakage in the final error panel. We assigned $\mu_1 = 0.1$ to both non-robust and robust cases, and we found that it does not appear to play a significant role in the final result.

The unblended and pseudo-deblended data for blending factor of two are shown in Figure 3.9a and 3.9b. The latter correspond to a common channel gather. Figure 3.9c and 3.9d shows the deblending results computed via the coherence pass non-robust and robust Radon operators, respectively. Figure 3.9e, 3.9f and 3.9g is the estimated error sections obtained via the difference between 3.9b and 3.9a, 3.9c and 3.9a, 3.9d and 3.9a, respectively. Comparing Figure 3.9c and 3.9d, we can observe that the coherence pass robust Radon method is more effective for suppressing blending noise and produces less signal leakage. Moreover,

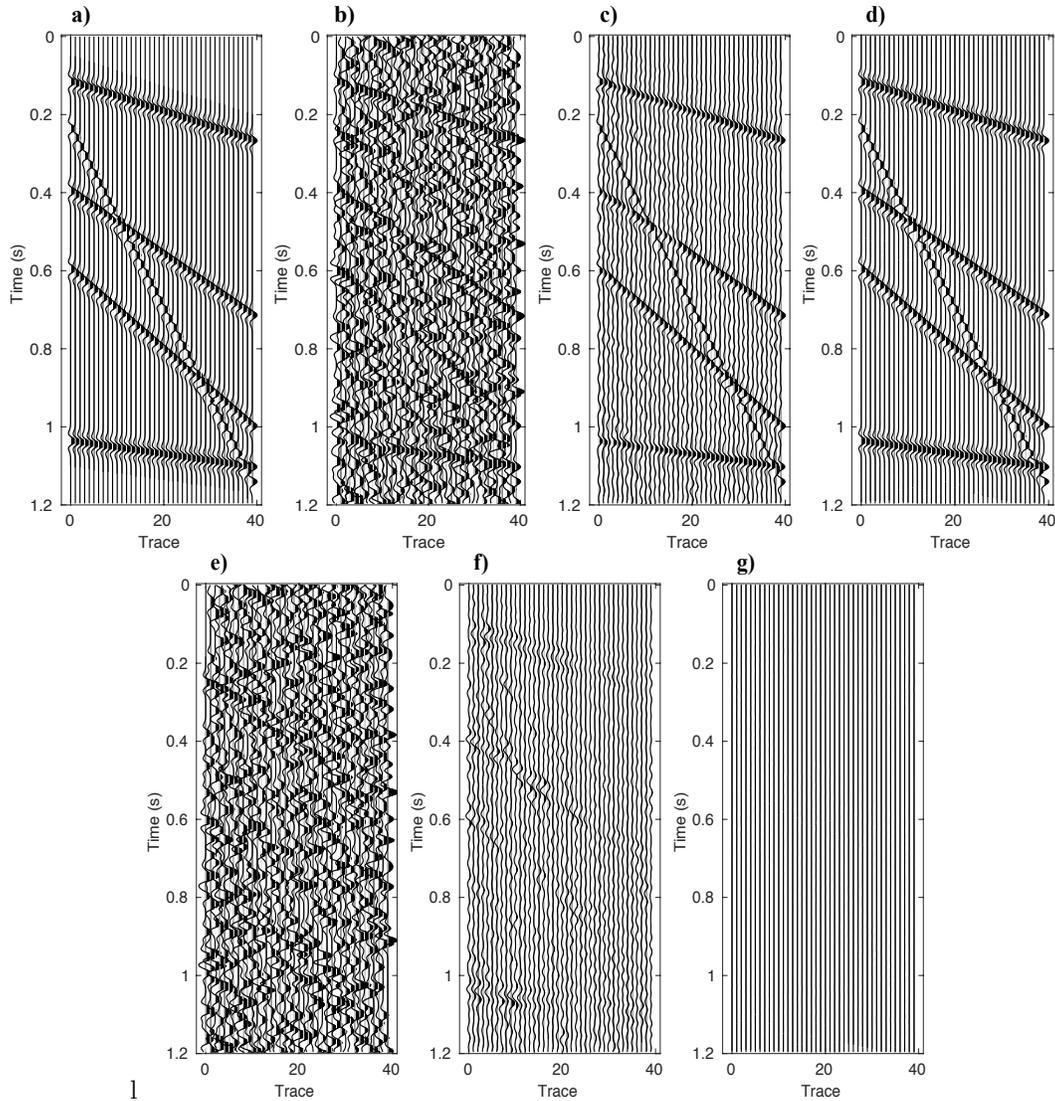


Figure 3.6: Deblending of synthetic data in common receiver gather with blending factor of ten. (a) Unblended data. (b) Pseudo-deblended data. (c) The deblending result with the coherence-pass non-robust sparse Radon operator method. (d) The deblending result with the coherence-pass robust sparse Radon operator method. (e) Blending noise and the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

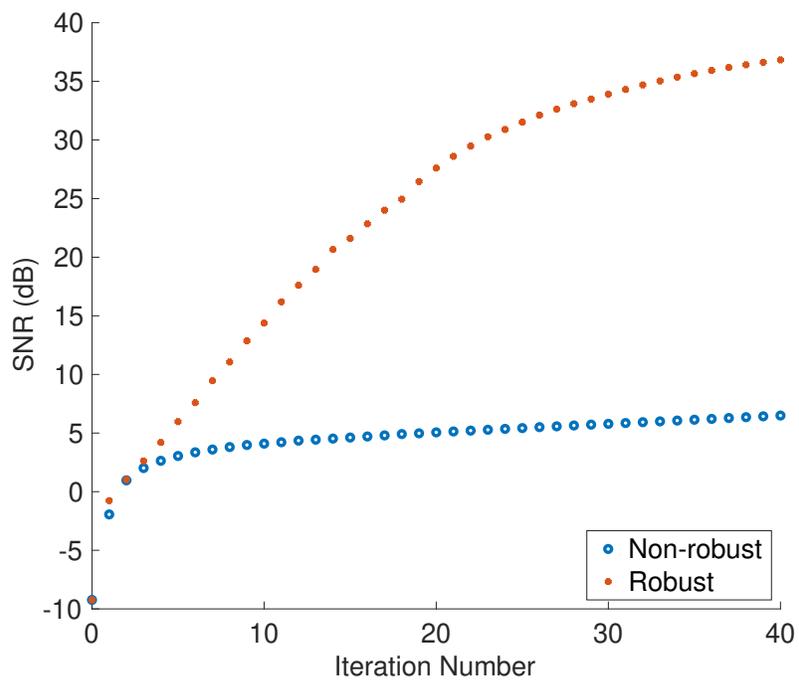


Figure 3.7: Synthetic data example. SNR versus iteration number for blending factor of ten.

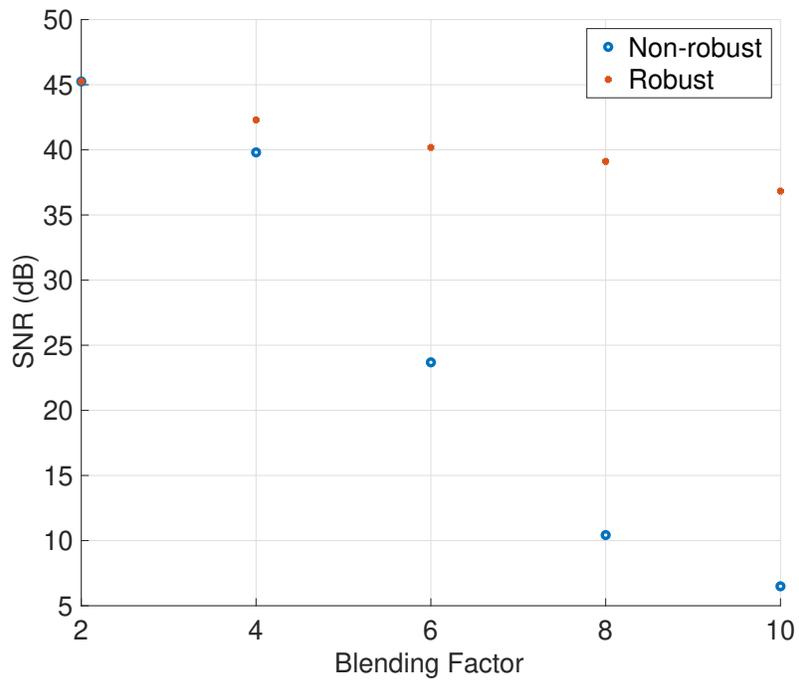


Figure 3.8: Synthetic data example. SNR versus blending factor.

the error estimation section of the robust coherence pass operator (Figure 3.9g) is almost negligible. Comparisons for one common shot gather can be found in Figure 3.10. We notice that adopting a coherence pass robust Radon operator can generate cleaner deblended data (Figure 3.10d). In Figure 3.10c, we still can observe some blending noise left. By comparing Figure 3.10f and 3.10g, we observe signal leakage and residual blending noise in the non-robust solution. The error panel of the solution obtained via the robust coherence pass operator (Figure 3.10g) has a negligible amount of coherent signal. Figure 3.11 shows the diagram portraying SNR versus iteration.

We repeated tests for blending factor of six (Figures 3.12, 3.13 and 3.14) and for blending factor of ten (Figure 3.15, 3.16 and 3.17). Again, similar to the previous example, we observe less signal leakage and a higher SNR value when a coherence-pass robust Radon operator is adopted (Figures 13 and 16).

Finally, Figure 3.18 shows the SNR versus the blending factor. As expected, the SNR decreases for both methods as more blending noise is included. We also conclude that the coherence pass robust Radon operator outperforms the non-robust Radon operator, with differences becoming more noticeable for increasing blending factor.

3.4 Conclusions

This chapter illustrates an inversion scheme for separating simultaneous source data. The proposed method adopts a robust Radon transform as a coherence pass operator used to clean common receiver (or channel) gathers in a typical iterative deblending by inversion process. We also compare the inversion with the classical coherence pass non-robust Radon operator. Our main conclusion is that deblending by inversion methods requires many iterations to obtain high-quality results if one does not aggressively remove incoherent noise during initial iterations. The latter can be achieved systematically by developing robust coherence pass operators like the one described in this chapter. It is important to stress that the coherence pass robust and non-robust Radon operators have similar computational costs when implemented via the IRLS algorithm. Given that in both cases, we have considered a sparsity constraint to estimate the Radon coefficients, the iterative updates of the IRLS algorithm are roughly cost-wise equivalent for coherence pass non-robust and robust operators.

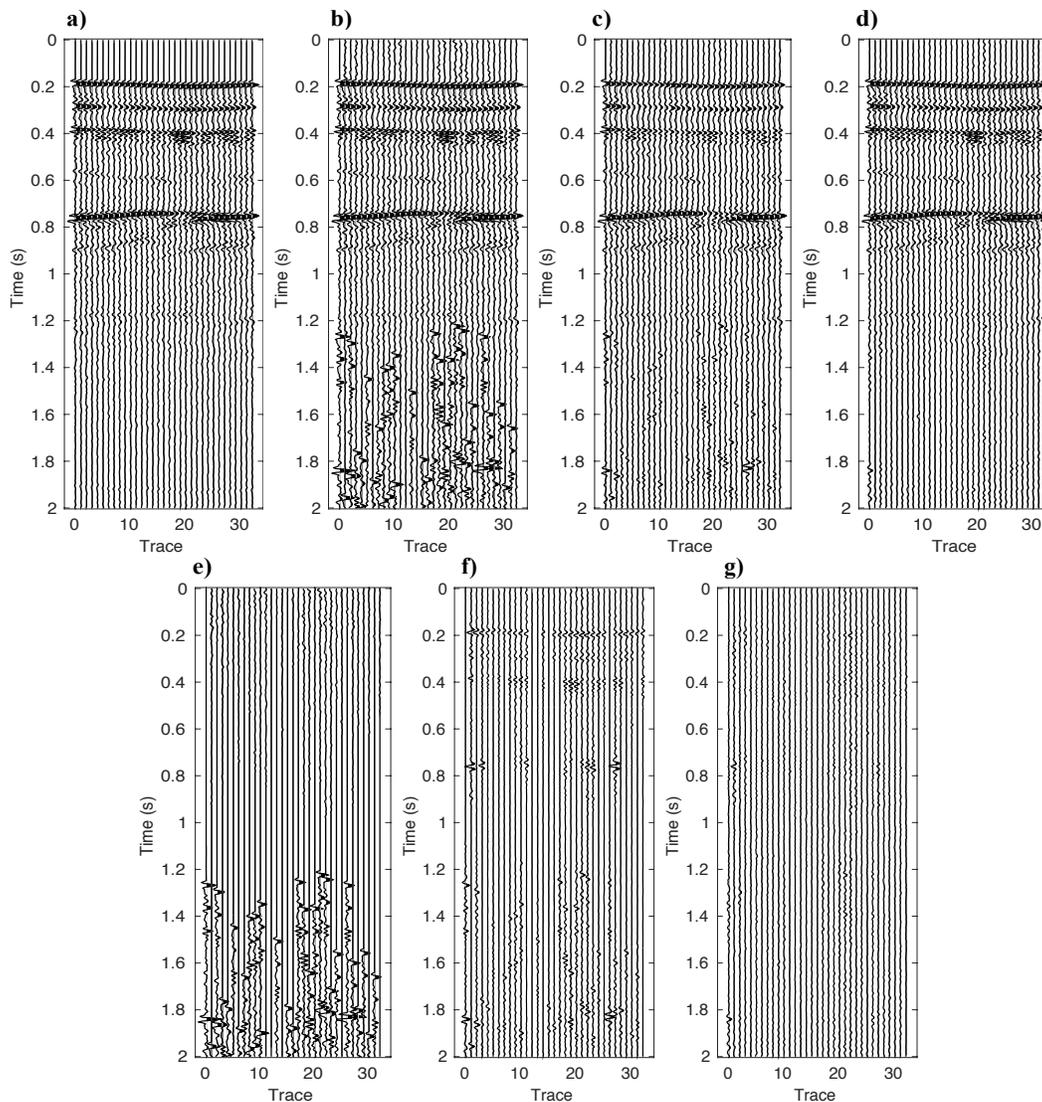


Figure 3.9: Deblending example of a real dataset with blending factor of two (Common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

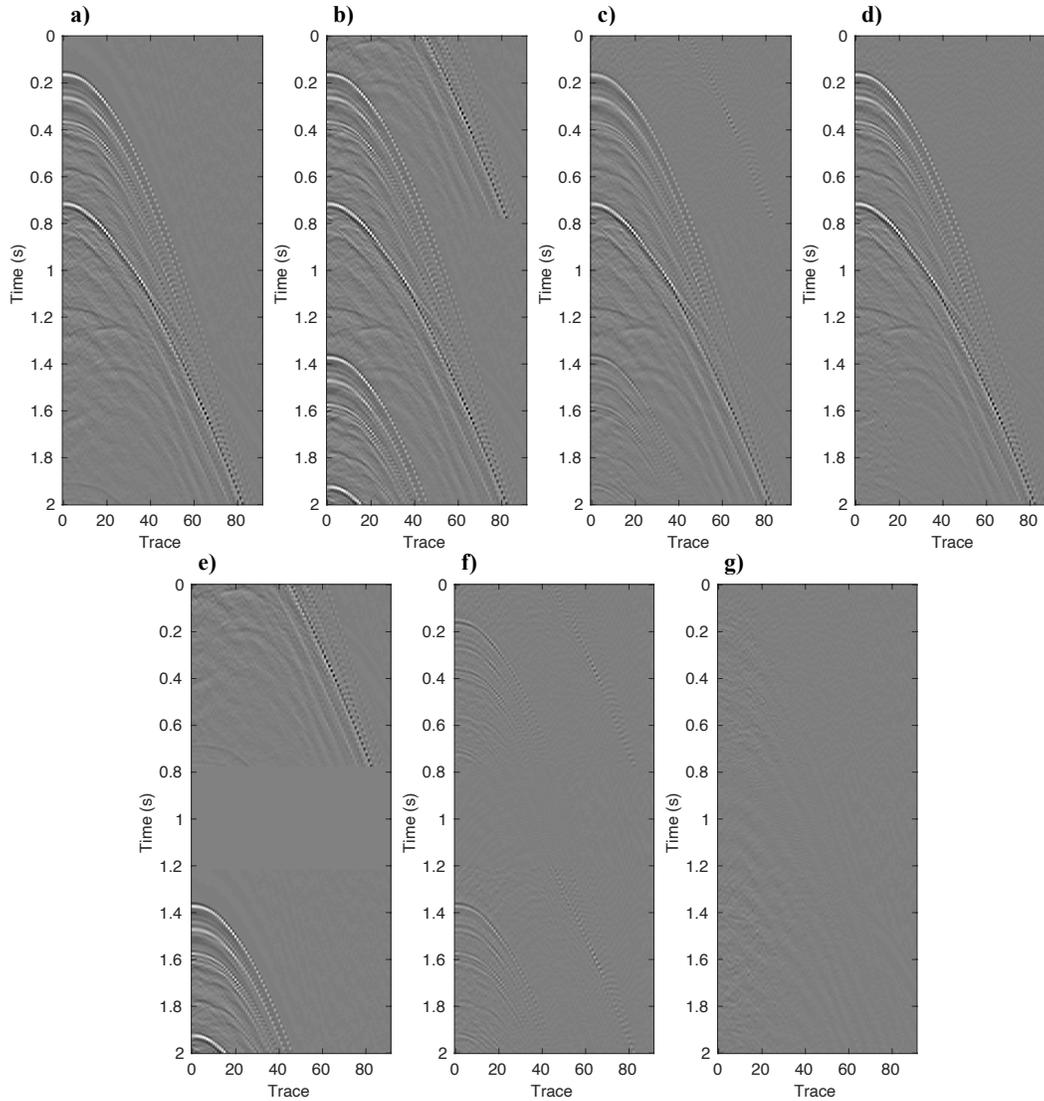


Figure 3.10: Results corresponding to one common shot gather for blending factor of two. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator. (e) Blending interferences in common shot gather given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

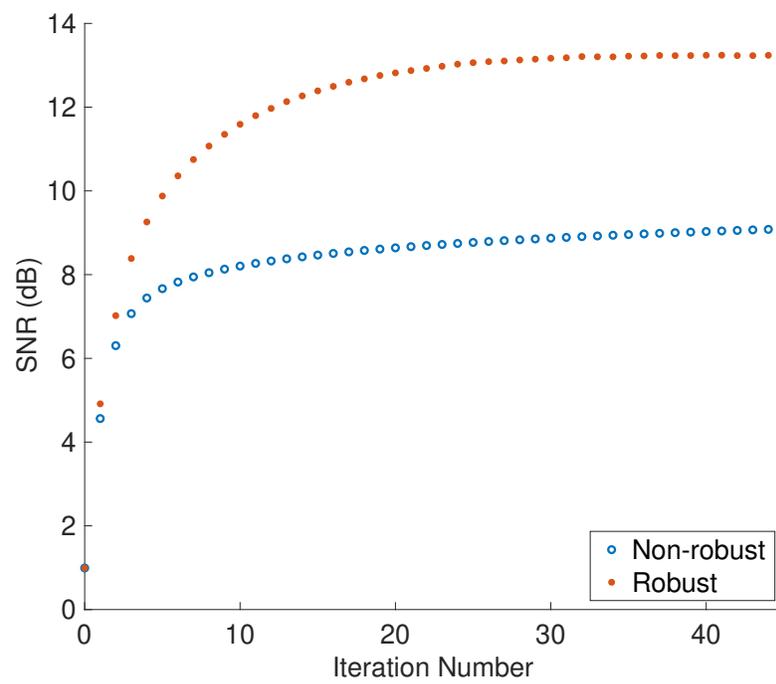


Figure 3.11: Real data example. SNR versus iteration number for blending factor of two.

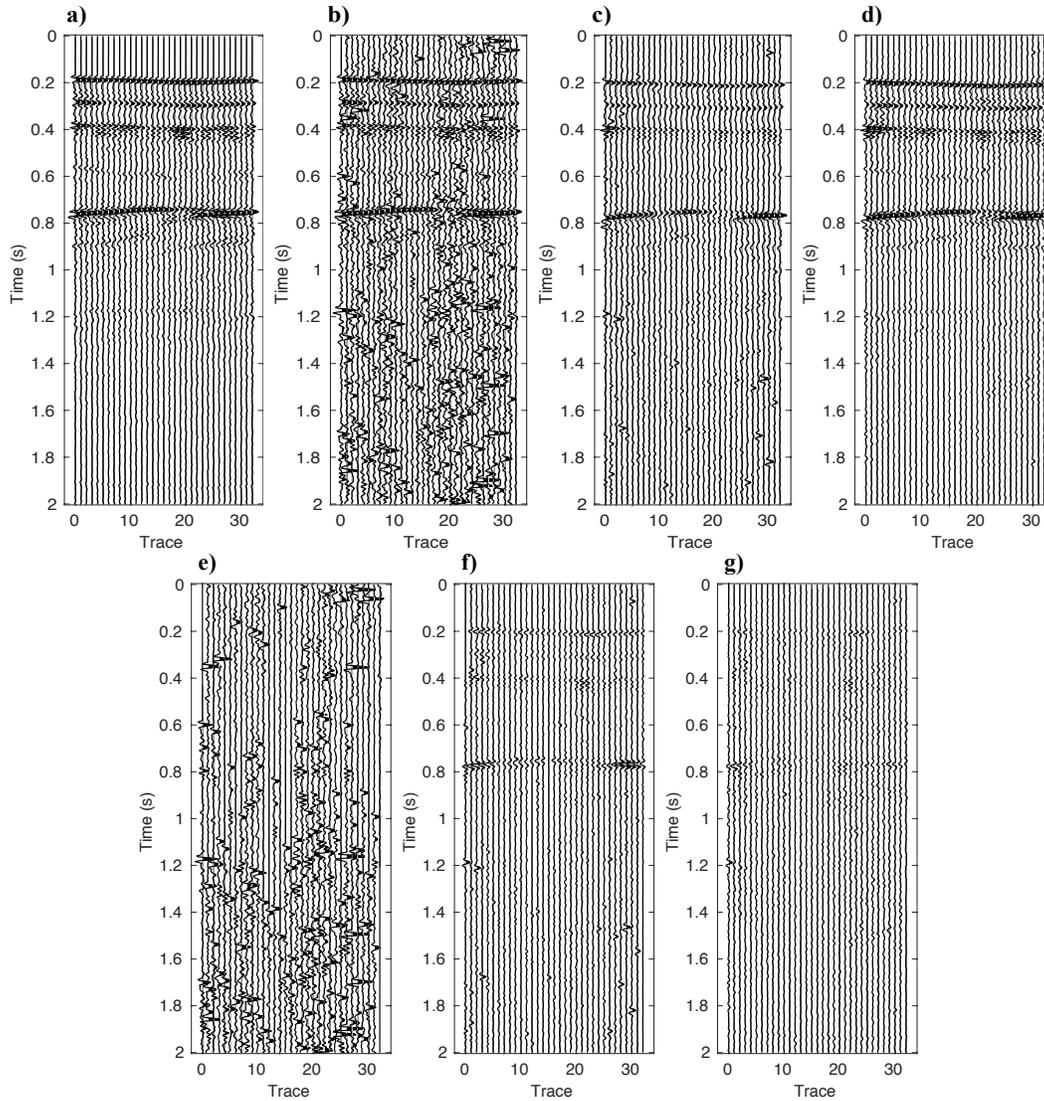


Figure 3.12: Deblending example of a real dataset with blending factor of six (common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

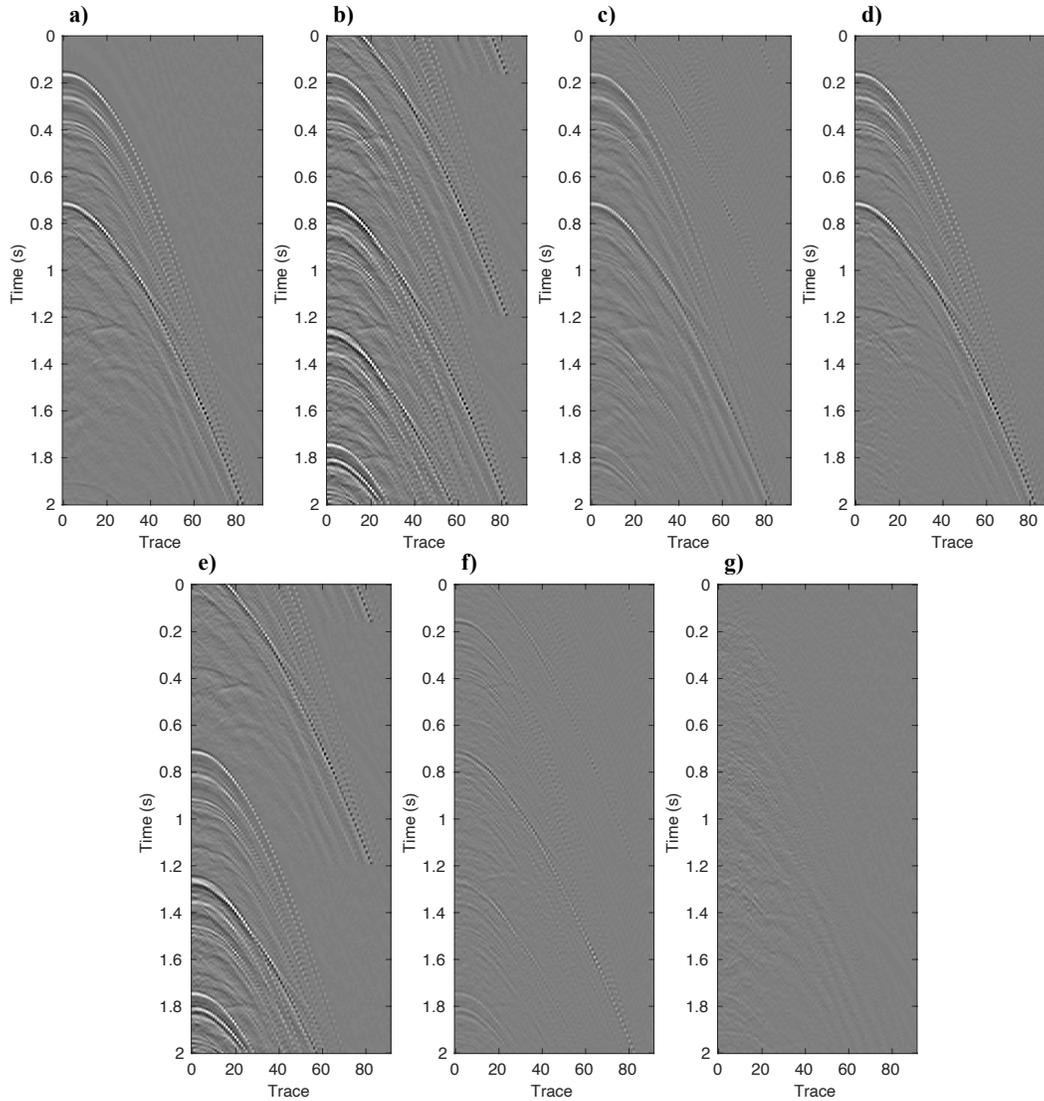


Figure 3.13: Results corresponding to one common shot gather for a blending factor of six. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences in common shot gather given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

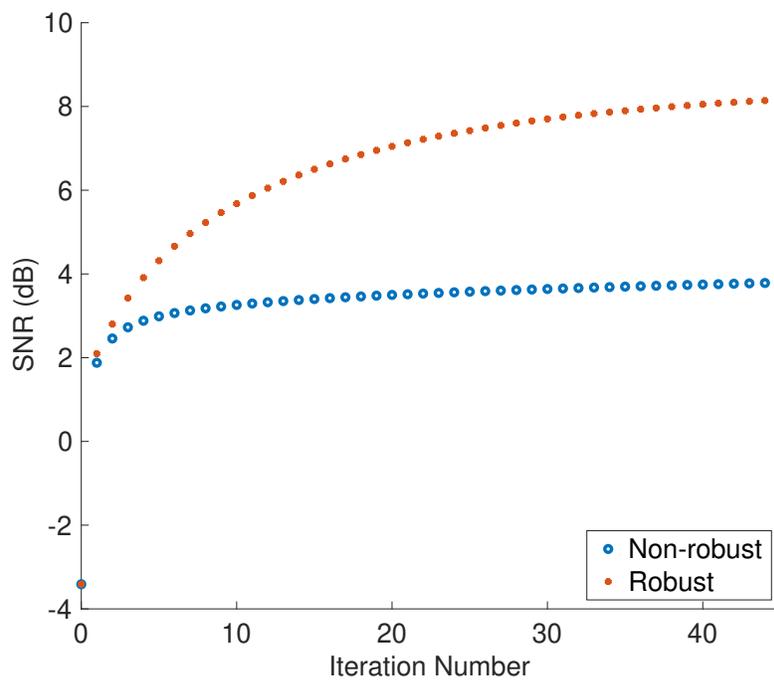


Figure 3.14: Real data example. SNR versus iteration number for blending factor of six.

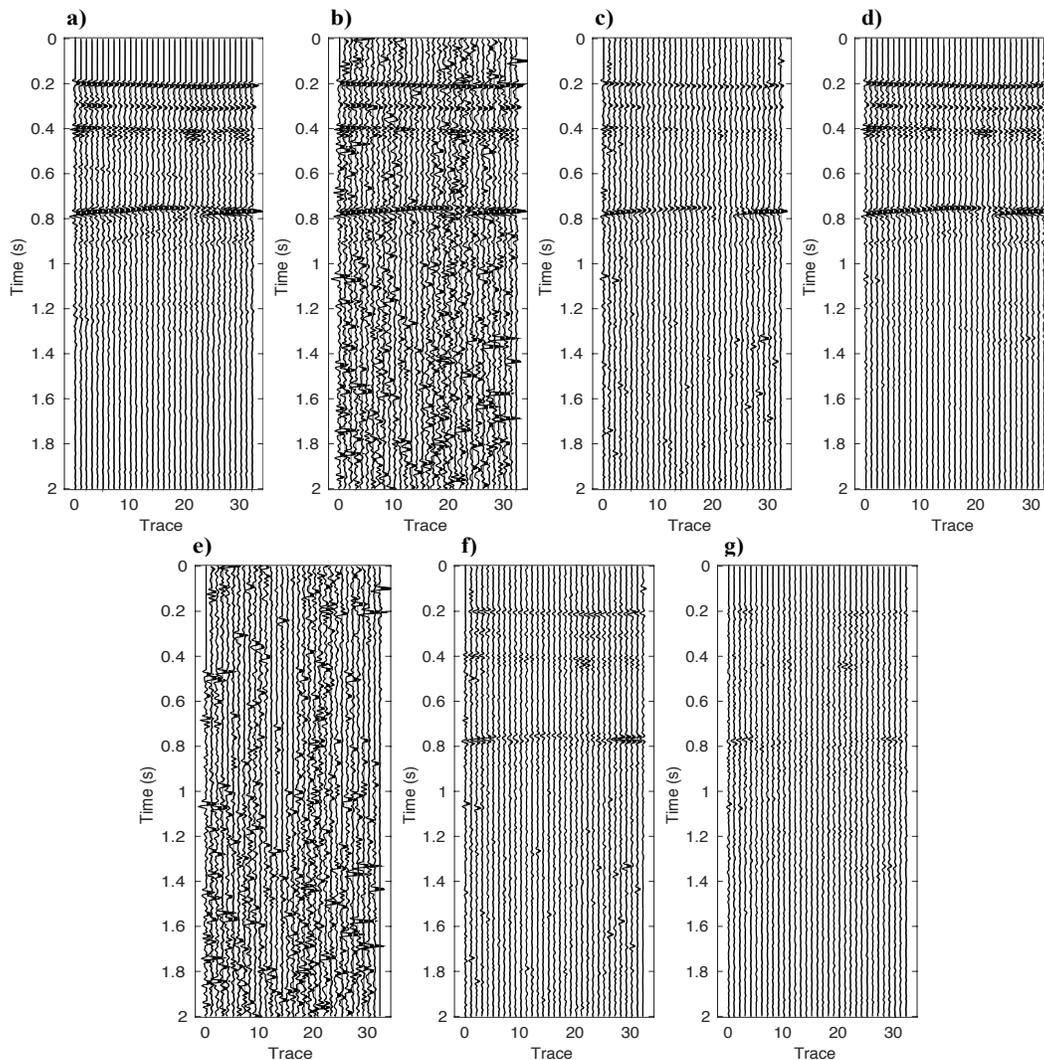


Figure 3.15: Deblending example of a real dataset with blending factor of ten (common channel gather). (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences given by the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

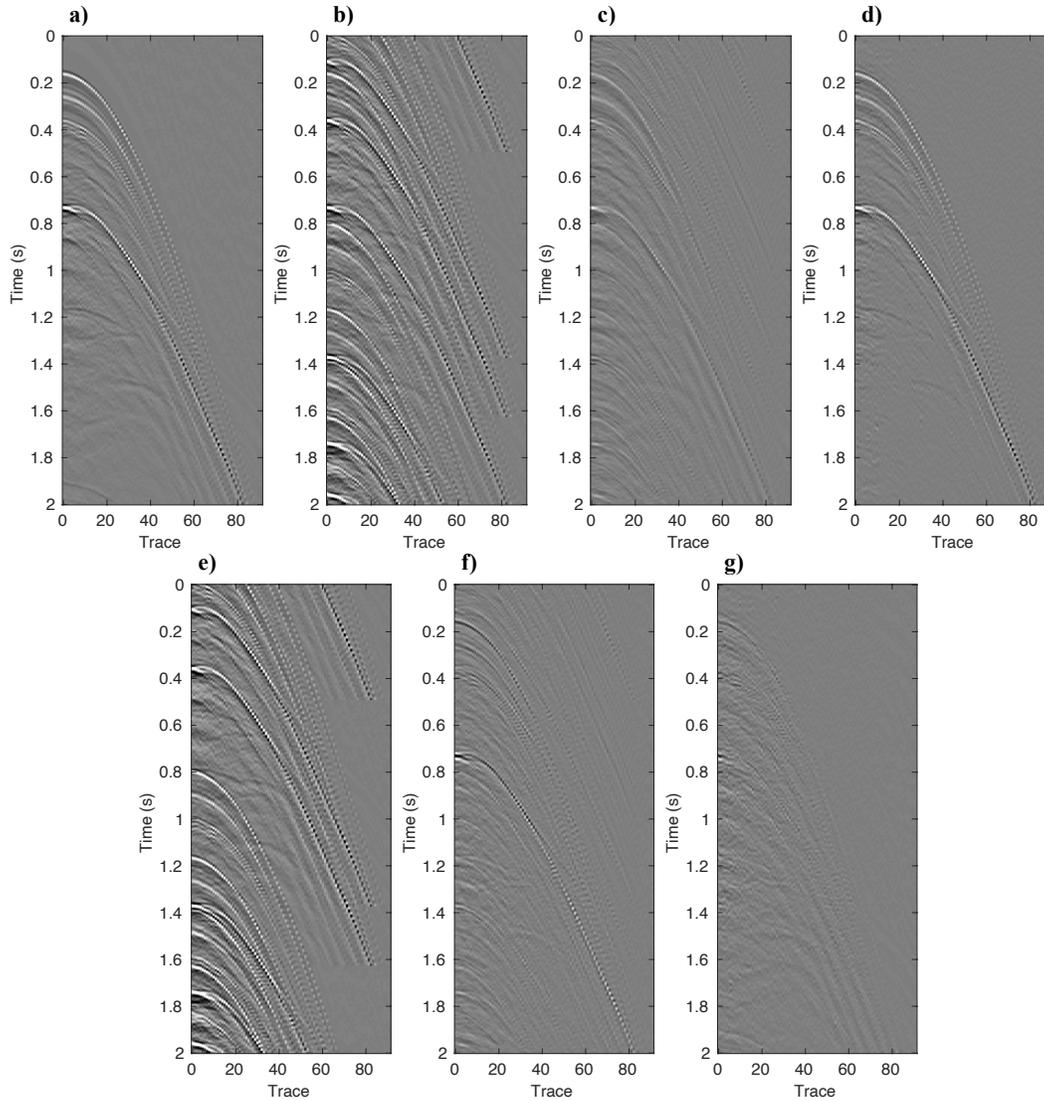


Figure 3.16: Results corresponding to one common shot gather for a blending factor of ten. (a) Unblended data. (b) Pseudo-deblended data. (c) Deblended data via the coherence pass non-robust sparse Radon operator method. (d) Deblended data via the coherence pass robust sparse Radon operator method. (e) Blending interferences in the common shot are also the difference between (b) and (a). (f) The difference between (c) and (a) gives the estimation error. (g) The estimation error is given by the difference between (d) and (a).

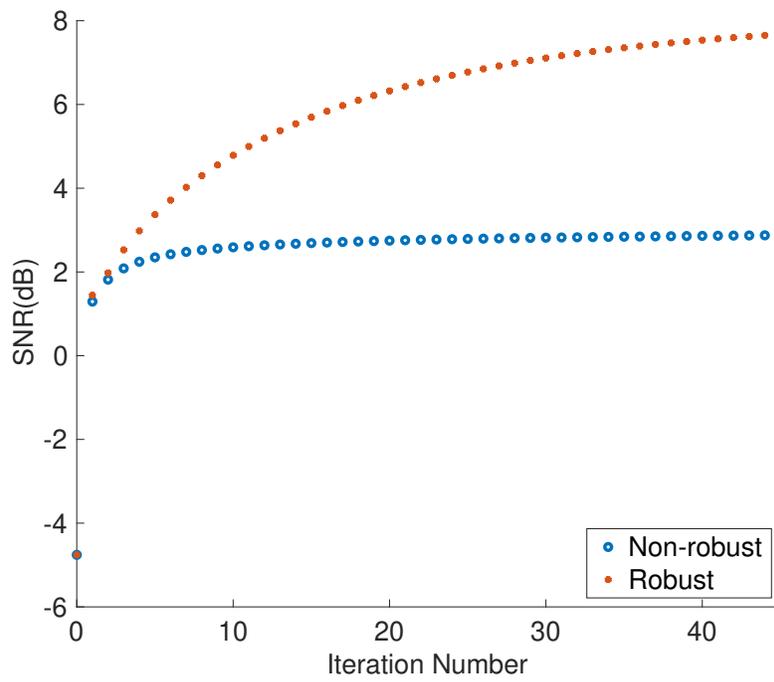


Figure 3.17: Real data example. SNR versus iteration number for blending factor of ten.

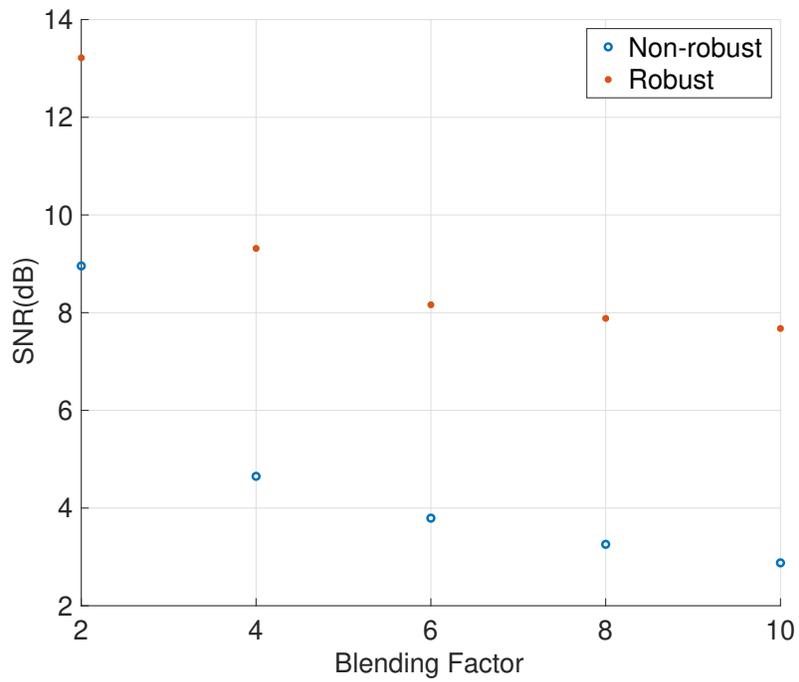


Figure 3.18: Real data example. SNR versus blending factor.

CHAPTER 4

Deblending via a robust Multichannel Singular Spectrum Analysis filter ¹

4.1 Introduction

For simultaneous source separation, in the category of inversion-based methods, rather than applying sparse inversion methods on auxiliaries bases, one can also adopt rank-constrained optimization to iteratively solve the deblending problem (Maraschini et al., 2012b; Cheng and Sacchi, 2015, 2016; Jeong et al., 2020).

Approaches based on reduced-rank filtering iteratively apply denoising to small data windows extracted from common-receiver gathers or common channel gathers (Cheng and Sacchi, 2015). The latter could be problematic because, in general, reduced-rank filtering techniques require rank as a user-supplied parameter. The optimal rank could vary from window to window. The reduced-rank filters adopted in the projected gradient descent algorithm could be the one-dimensional Singular Spectrum Analysis (SSA) filter or its multichannel version (MSSA) filter (Ghil et al., 2002) adapted for seismic data processing (Oropeza and Sacchi, 2011). The SSA/MSSA filtering, also denominated the Cazdow filter (Cazdow, 1988), has been used extensively for prestack seismic denoising and reconstruction (Trickett, 2008; Trickett et al., 2012). The MSSA/Cazdow filter is non-robust and generally yields suboptimal results when bursts of erratic noise contaminate observations. Attempts to improve the robustness of the MSSA filter include methods that replace the Singular Value Decomposition with a robust matrix factorization (Cheng and Sacchi, 2015; Bahia and Sacchi, 2019), or with a robust iterative reweighting scheme (Trickett et al., 2012).

¹A version of this chapter is published in Lin, R., B. Bahia, and M. D. Sacchi, 2021, Iterative deblending of simultaneous-source seismic data via a robust singular spectrum analysis filter: *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 1-10.

In this chapter, we solve the simultaneous source separation problem by adopting the projected gradient descent (PGD) method to iteratively estimate the data one would acquire via a conventional seismic acquisition. The projection operator is a windowed robust MSSA filter that suppresses source interferences in the frequency-space domain. We reformulate the MSSA filter as a robust optimization problem solved via a bifactored gradient descent (BFGD) algorithm. The MSSA filter requires breaking down common-receiver gathers, common channel gathers or common offset gathers into small overlapping windows. The traditional MSSA method needs the filter rank as an input parameter, which can vary from window to window. The latter has been a shortcoming for applying classical MSSA filtering to complex seismic data processing. The proposed robust MSSA filter is less sensitive to the rank selection, making it appealing for deblending applications that require windowing. Additionally, the robust MSSA projection effectively attenuates random source interferences during the initial iterations of the projected gradient descent method. Comparing classical and robust MSSA filters, we also report an acceleration of the projected gradient descent method convergence when we adopt the robust MSSA filter. Finally, we provide synthetic and real data examples and discuss heuristic strategies for parameter selection.

This chapter follows the subsequent structure. First, I briefly introduce the projected gradient method for simultaneous-source separation with low-rank constraints. I continue with a description of windowed MSSA to guarantee the linear-event assumption with the MSSA filtering. Next, I compare the classical non-robust MSSA filter with the robust MSSA filter as the projection operators. Finally, I test the two projection operators via synthetic and field data.

4.2 Theory

4.2.1 Separation of simultaneous-source data via the PGD method

For simultaneous-source separation based on rank-constrained optimization, one can estimate \mathbf{D} by solving an inverse problem with the following cost function

$$\begin{aligned} \min \quad & J = \|\mathbf{b} - \mathcal{B}\mathbf{D}\|_2^2 \\ \text{s.t.} \quad & \text{rank}(\mathbf{D}) \leq k. \end{aligned} \tag{4.1}$$

With the PGD method (Cheng and Sacchi, 2016; Bolduc et al., 2017; Peters et al., 2019; Lin et al., 2021), one can solve equation 4.1 by defining the gradient-descent step followed by a projection:

$$\begin{aligned} \mathbf{D}^\nu &= \mathcal{P} [\mathbf{D}^{\nu-1} - \lambda \nabla J (\mathbf{D}^{\nu-1})] \\ &= \mathcal{P} [\mathbf{D}^{\nu-1} - \lambda \mathcal{B}^* (\mathcal{B} \mathbf{D}^{\nu-1} - \mathbf{b})] , \end{aligned} \quad (4.2)$$

where ∇J is the gradient of the error function J . The scalar λ is a step-size, and \mathcal{P} indicates the projection operator. In our work, \mathcal{P} is a denoising algorithm based on the $f - x$ MSSA filter (Oropeza and Sacchi, 2011; Cheng and Sacchi, 2015). The MSSA filter works on small overlapping spatio-temporal windows to iteratively remove blending erratic noise. The latter is the basis of the Iterative Rank Reduction (IRR) debrending method (Cheng and Sacchi, 2015). The IRR process and its robust version are part of our discussion in subsequent sections. The filtering is applied to one common receiver/channel gather simultaneously, but we stress that the process repeats for all receivers/channels. Then, filtered common-receiver/channel gathers are organized in common source gathers to continue with the standard seismic data processing flow. Algorithm 4 provides the PGD method for debrending with low-rank constraint optimization. Notice that the step length needs to be estimated to guarantee the convergence of the PGD method (Iusem, 2003; Bolduc et al., 2017). We describe a strategy for prescribing the step length when discussing our examples.

Algorithm 4 Debrending by the Projected Gradient Descent Method

```

1: Inputs:
   Blended data  $\mathbf{b}$ , step size  $\lambda$ , and tolerance  $\epsilon$ .
2: Outputs:
   Debrended data  $\mathbf{D}$ .
3: Initialization:
    $k = 0$ ,  $\mathbf{D} = \mathcal{B}^* \mathbf{b}$ ,  $\mathbf{e} = \mathcal{B} \mathbf{D} - \mathbf{b}$ ,  $J = \|\mathbf{e}\|_2^2$ .
4: while  $k \leq k_{max} \mid J \geq \epsilon$  do
5:    $\hat{\mathbf{D}} \leftarrow \mathbf{D}^k - \lambda \mathcal{B}^* \mathbf{e}$ 
6:    $\mathbf{D}^{k+1} \leftarrow \mathcal{P}[\hat{\mathbf{D}}]$    Algorithm 5
7:    $\mathbf{e} = \mathcal{B} \mathbf{D}^{k+1} - \mathbf{b}$ 
8:    $J = \|\mathbf{e}\|_2^2$ 
9:    $k \leftarrow k + 1$ 
10: end while

```

4.2.2 Windowed MSSA filter

For simplicity, we designate the projection operator in equation 4.2 as follows

$$\hat{\mathbf{D}} = \mathcal{P}[\mathbf{D}] \quad (4.3)$$

where \mathbf{D} and $\hat{\mathbf{D}}$ are input and filtered data after the projection \mathcal{P} , respectively. The projector operator entails several operators described below. First, we describe the windowing

operator we have adopted to validate an important assumption made by MSSA filtering. In general, the MSSA filter is optimal for waveforms represented by linear events in the $t - x$ domains. These events correspond to complex exponentials in the $f - x$ domain. It is easy to show that P linear events in the $t - x$ domain correspond to the superposition of P complex exponentials in the $f - x$ domain where the exponentials are a function of space x (Oropeza and Sacchi, 2011). The assumption mentioned above is the essence of prediction error filtering methods adopted by exploration seismologists for denoising and the main driving principle of the $f - x$ MSSA filter for seismic signal enhancement and reconstruction (Ulrych and Sacchi, 2005). A linear moveout is a good approximation for seismic reflections in small spatio-temporal windows. Therefore, it is crucial to emphasize that MSSA filtering acts on small overlapping windows where one can model reflections as a superposition of events with linear moveout. The proposed windowing approach is summarized as follows

1. First step of the windowed MSSA filter is to break the data into small overlapping data patches to limit the number of events (dips)

$$\mathbf{S}_k = \mathbf{W}_k[\mathbf{D}], \quad k = 1 \dots N_w \quad (4.4)$$

where N_w is the number of windows, and \mathbf{W} represents the windowing operator with the action of extracting a spatio-temporal window with proper tapering (Figure 4.1). We call \mathbf{W}_k the k -th analysis window. The extracted k -th data \mathbf{S}_k is a matrix of size $L_t \times L_x$ where L_t is the number of time samples and L_x the number of seismic traces in the window.

2. We then apply the MSSA filter or the robust MSSA filter to each window. Both filters are described in the next subsections

$$\hat{\mathbf{S}}_k = \mathcal{S}_{\mathcal{F}}[\mathbf{S}_k] \quad k = 1 \dots N_w, \quad (4.5)$$

where $\mathcal{S}_{\mathcal{F}}$ symbolizes the action of applying MSSA or robust MSSA to a data window.

3. Then, the filtered data patches are properly moved back to their position and summed up to yield the filtered data

$$\hat{\mathbf{D}} = \sum_{k=1}^{N_w} \mathbf{W}_k^*[\hat{\mathbf{S}}_k] \quad (4.6)$$

where \mathbf{W}^* is the synthesis window operator for data patch k (See Figure 4.1).

Combining equations 4.4, 4.5 and 4.6 leads to

$$\hat{\mathbf{D}} = \sum_k \mathbf{W}_k^*[\mathcal{S}_{\mathcal{F}}[\mathbf{W}_k[\mathbf{D}]]]. \quad (4.7)$$

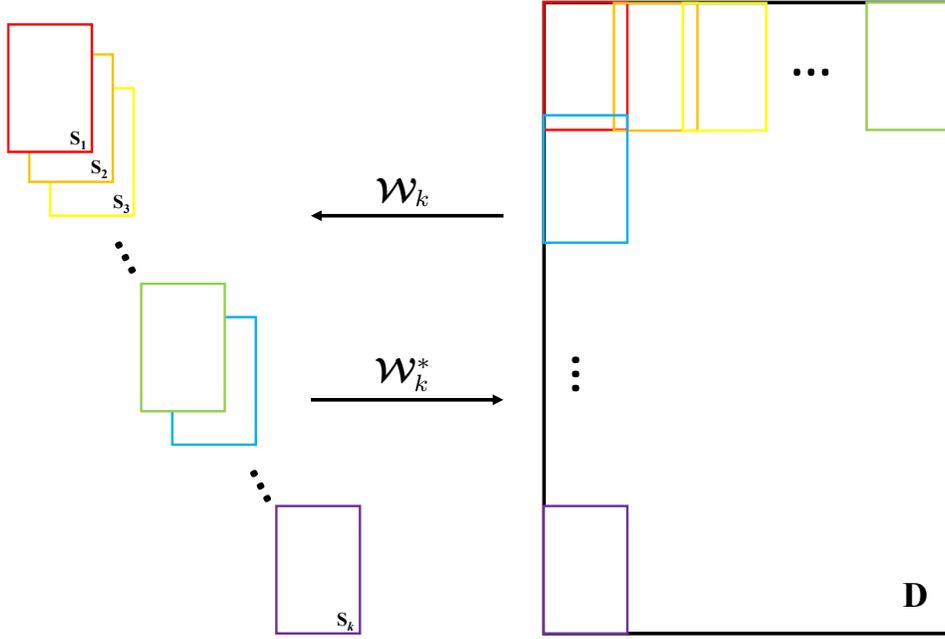


Figure 4.1: The windowing operator and its associated adjoint operator. The windowing operator \mathcal{W}_k extracts small patches of data from a whole gather. The adjoint operator \mathcal{W}_k^* synthesizes all the filtered small patches back into a gather. A linear taper is adopted for overlaps processing. Data \mathbf{D} denotes the whole data gather, and $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \dots, \mathbf{s}_k$ represent small patches of data.

The operators for analysis and synthesis are chosen such that if $\mathcal{S}_{\mathcal{F}} = \mathcal{I}$ (no filtering is applied), $\sum_k \mathcal{W}_k^* \mathcal{W}_k = 1$. The latter guarantees that in the absence of filtering $\hat{\mathbf{D}} = \mathbf{D}$. In other words, the windowing process does not introduce any distortion. See also Algorithm 5.

4.2.3 Denoiser based on the classical non-robust MSSA filter

We now concentrate on applying the MSSA filter to one data patch \mathbf{S}_k by equation 4.5. We can safely ignore the window subindex k and understand that the MSSA filter is applied to all data patches. The MSSA filter is applied in the frequency domain and constitutes a multi-level block-Hankel matrix for each frequency. We first transform the data patch to the frequency-space domain which accounts to $s(t, x) \leftrightarrow s(f, x)$. We use the vector $\mathbf{s}(f)$ to also designate the spatial data for frequency f , in other words, $\mathbf{s}(f) = [s(f, 1), s(f, 2), \dots, s(f, L_x)]^T$. The classical MSSA filter can be summarized by the following steps:

Algorithm 5 Windowed MSSA: $\hat{\mathbf{D}} = \mathcal{P}[\mathbf{D}]$ 1: **Inputs:**

Number of windows N_w and parameters prescribing analysis and synthesis windows.

\mathbf{D} is $N_t \times N_x$ common-receiver gather.

2: **Outputs:**

$\hat{\mathbf{D}}$ is filtered $N_t \times N_x$ common-receiver gather.

3: **for** $k = 1 : N_w$ **do**

4: $\mathbf{S}_k = \mathcal{W}_k[\mathbf{D}]$ *Extract a t - x data patch*

5: $\hat{\mathbf{S}}_k = \mathcal{S}_{\mathcal{F}}[\mathbf{S}_k]$ *MSSA filter (Algorithm 6)*

6: $\hat{\mathbf{D}} = \hat{\mathbf{D}} + \mathcal{W}_k^*[\hat{\mathbf{S}}_k]$ *Synthesize data from filtered data patches*

7: **end for**

1. For each frequency f , the spatial data $\mathbf{s}(f)$ is embedded into a Hankel matrix

$$\begin{aligned} \mathbf{H}(f) &= \mathcal{H}[\mathbf{s}(f)] \\ &= \begin{pmatrix} s(f, 1) & s(f, 2) & \dots & s(f, N_c) \\ s(f, 2) & s(f, 3) & \dots & s(f, N_c + 1) \\ s(f, 3) & s(f, 4) & \dots & s(f, N_c + 2) \\ \vdots & \vdots & \vdots & \vdots \\ s(f, N_r) & s(f, N_r + 1) & \dots & s(f, L_x) \end{pmatrix} \end{aligned} \quad (4.8)$$

with $N_r = \lfloor L_x/2 \rfloor + 1$ and $N_c = L_x - N_r$. The Hankel $\mathbf{H}(f)$ is square for L_x odd or almost square for L_x even. The symbol \mathcal{H} is the Hankelization operator.

2. Then, we apply rank reduction via the Singular Value Decomposition (SVD) to estimate the rank p matrix $\mathbf{H}_p(f)$ that minimizes the Frobenius norm $\|\mathbf{H}(f) - \mathbf{H}_p(f)\|_F$ (Eckart and Young, 1936). The SVD decomposition yields the orthonormal matrices $\mathbf{U}_p(f)$ and $\mathbf{V}_p(f)$ of size $N_r \times p$ and $N_c \times p$, respectively, and the $p \times p$ diagonal matrix of singular values $\mathbf{\Sigma}_p$. The reduced-rank approximation is given by $\mathbf{H}_p(f) = \mathbf{U}_p(f)\mathbf{\Sigma}_p\mathbf{V}_p(f)^H$. Clearly, p is the desired rank of the approximation supplied by the user.
3. The filtered signal is recovered via antidiagonal averaging the elements of the matrix $\mathbf{H}_p(f)$

$$\hat{\mathbf{s}}(f) = \mathcal{A}[\mathbf{H}_p(f)] \quad (4.9)$$

where \mathcal{A} is the antidiagonal averaging operator.

4. Finally, once $\hat{\mathbf{s}}(f)$ is estimated for all frequencies, one can use the inverse DFT to transform back the signal to the $t - x$ to obtain the filtered data $\hat{\mathbf{s}}(t, x)$.

Algorithm 6 shows the classical MSSA filter in its non-robust option.

Algorithm 6 Classical MSSA filter via SVD and robust MSSA via BFGD, $\hat{\mathbf{S}} = \mathcal{S}_{\mathcal{F}}[\mathbf{S}]$

```

1: Inputs:
   Small patch of data  $\mathbf{S}$  of size  $L_t \times L_x$  and frequency
   band  $B$  where denoising is carried out.
2: Outputs:
   Filtered patch of data  $\hat{\mathbf{S}}$  of size  $L_t \times L_x$ .
3:  $\mathbf{S} \leftarrow \text{fft}(\mathbf{S})$  1D column-wise DFT ( $t-x \rightarrow f-x$ )
4: for all  $f \in B$  do
5:    $\mathbf{s}(f) = \mathbf{S}(f, :)$  Extract spatial data at frequency  $f$ 
6:   if robust==true then
7:      $[\mathbf{U}_p, \mathbf{V}_p] = \text{BFGD}(\mathbf{s}(f), p)$  Algorithm 7
8:      $\mathbf{H}_p = \mathbf{U}_p \mathbf{V}_p^H$ 
9:   else
10:     $\mathbf{H} = \mathcal{H}[\mathbf{s}(f)]$  Hankelization
11:     $[\mathbf{U}_p, \mathbf{\Sigma}_p, \mathbf{V}_p] = \text{SVD}(\mathbf{H}, p)$  Rank-reduction
12:     $\mathbf{H}_p = \mathbf{U}_p \mathbf{\Sigma}_p \mathbf{V}_p^H$ 
13:   end if
14:    $\hat{\mathbf{s}}(f) = \mathcal{A}[\mathbf{H}_p]$  Antidiagonal average
15:    $\hat{\mathbf{S}}(f, :) = \hat{\mathbf{s}}(f)$ 
16: end for
17:  $\hat{\mathbf{S}} \leftarrow \text{ifft}(\hat{\mathbf{S}})$  1D column-wise IDFT ( $f-x \rightarrow t-x$ )

```

4.2.4 Denoiser based on the robust MSSA filter

The SVD provides the least-squares solution to the matrix low-rank approximation problem (Eckart and Young, 1936). Deblending algorithms must cope with intense erratic noise, usually responsible for their slow convergence (Lin and Sacchi, 2020a). The proposition is that an MSSA filter designed via a robust procedure, rather than the SVD, will act as a more effective means to eliminate source interferences in early iterations of the PGD algorithm.

We make the MSSA filter robust by reformulating it as an optimization problem. We represent again the observed spatial data at a given frequency f by $\mathbf{s}(f)$ and assume that the observed data can be approximated by the antidiagonal averaging operator applied to an unknown matrix $\mathbf{H}_p(f)$ of rank p . Therefore, the proposed cost function to estimate the robust MSSA filter is given by

$$J(f) = \|\mathbf{s}(f) - \mathcal{A}[\mathbf{H}_p(f)]\|_{\rho} \quad (4.10)$$

where $\|\cdot\|_{\rho}$ is used to indicate Tukey's biweight loss function (Ji, 2012; Belagiannis et al.,

2015). If the scaled error is given by $\mathbf{r}(f) = (\mathbf{s}(f) - \mathcal{A}[\mathbf{H}(f)])/\sigma$ then

$$J(f) = \|\mathbf{r}(f)\|_\rho \quad (4.11)$$

$$= \sum_i \rho(r_i(f)), \quad (4.12)$$

where

$$\rho(x) = \begin{cases} \frac{\beta^2}{6} [1 - (1 - (\frac{|x|}{\beta})^2)^3], & \text{for } |x| \leq \beta \\ \frac{\beta^2}{6}, & \text{for } |x| > \beta. \end{cases} \quad (4.13)$$

We can ignore the symbol f from our notation and understand that the process is carried out for all frequencies as in the classical $f - x$ MSSA filter. Furthermore, we include the low-rank constraint by expressing the unknown matrix of desired rank p via $\mathbf{H}_p = \mathbf{U}\mathbf{V}^H$ where \mathbf{U} and \mathbf{V} are matrices of size $N_r \times p$ and $N_c \times p$, respectively. Rather than optimizing the loss function with respect to the unknown matrix \mathbf{H}_p , we adopt the BFGD algorithm (Park et al., 2018) and solve

$$\hat{\mathbf{U}}, \hat{\mathbf{V}} = \underset{\mathbf{U}, \mathbf{V}}{\operatorname{argmin}} \|\mathbf{s} - \mathcal{A}[\mathbf{U}\mathbf{V}^H]\|_\rho. \quad (4.14)$$

The BFGD algorithm is a first-order optimization method that operates directly with factors \mathbf{U} and \mathbf{V} . The gradients of equation 4.14 with respect to \mathbf{U} and \mathbf{V} lead to the classical steepest descent update

$$\mathbf{U}^{i+1} = \mathbf{U}^i - \eta \nabla_{\mathbf{U}} J^i \quad (4.15)$$

$$\mathbf{V}^{i+1} = \mathbf{V}^i - \eta \nabla_{\mathbf{V}} J^i \quad (4.16)$$

The index i indicates iteration, and η denotes suitable step lengths. The gradients are given by

$$\nabla_{\mathbf{U}} J = (\nabla_{\mathbf{H}} J) \mathbf{V} \quad (4.17)$$

$$\nabla_{\mathbf{V}} J = (\nabla_{\mathbf{H}} J)^H \mathbf{U} \quad (4.18)$$

where

$$\nabla_{\mathbf{H}} J = -\mathcal{A}^* \mathbf{W} (\mathbf{s} - \mathcal{A}[\mathbf{H}]). \quad (4.19)$$

The elements of the diagonal matrix of weights \mathbf{W} for Tukey's biweight loss function are given by the following expression:

$$W_{ii} = \begin{cases} \left[1 - \left(\frac{|r_i|}{\beta} \right)^2 \right]^2, & |r_i| \leq \beta \\ 0, & |r_i| > \beta. \end{cases} \quad (4.20)$$

where r_i indicates an element of the vector of scaled residuals \mathbf{r} and β is a user-defined tuneable parameter that controls the level of robustness of the denoiser. A common way to obtain the scale parameter σ is by adopting normalized median absolute deviation (MAD) (Holland and Welsch, 1977)

$$\sigma = 1.4826 \text{ MAD}, \quad (4.21)$$

where $\text{MAD} = \text{median}(|\mathbf{e} - \text{median}(|\mathbf{e}|)|)$ and \mathbf{e} denotes the unscaled vector of residuals $\mathbf{e} = \mathbf{s} - \mathcal{A}[\mathbf{H}_p]$. Because the solution is not known a priori, we cannot estimate the scale parameter σ . We prefer not to iterate the scale parameter proposed by Chen and Sacchi (2014). We simply start with a non-robust factorization computed by the SVD to determine the initial residuals from where we estimate σ and keep this value constant for the remaining iterations of Algorithm 7.

The reader can refer to Park et al. (2018) for the details of the selection of appropriate step-size η and initial factor matrices that lead to linear convergence. Last, it is important to mention that the operator \mathcal{A}^* is the adjoint of the antidiagonal averaging operator \mathcal{A} . It can be shown that \mathcal{A}^* is a scaled version of the Hankelization operator \mathcal{H} (Wang et al., 2021), and therefore, we safely replace \mathcal{A}^* by \mathcal{H} which was defined in equation 4.8. Replacing \mathcal{A}^* by \mathcal{H} is equivalent to using the pseudo-inverse of \mathcal{A} rather than its adjoint operator. The latter provides an improved descent direction in equations 4.15 and 4.16.

A point worth mentioning is that the final resulting factors computed by BFGD lead to a low-rank matrix $\widehat{\mathbf{H}}_p = \widehat{\mathbf{U}}\widehat{\mathbf{V}}^H$. However, the resulting $\widehat{\mathbf{H}}_p$ matrix is not guaranteed to be a Hankel matrix. The latter is not a problem because the final denoised signal is $\mathcal{A}[\widehat{\mathbf{H}}_p]$ and the associated Hankel matrix could be re-estimated by applying the Hankelization operator \mathcal{H} to the final denoised signal $\widehat{\mathbf{s}}$. One could iteratively refine the estimator of the signal by repeating BFGD until the final factors form a Hankel matrix. However, our numerical experiments show that such a procedure does not significantly improve the deblending performance.

4.3 Examples

This section tests the proposed algorithm with synthetic and numerically blended field data. We will use the following expression to measure the signal-to-noise ratio of the recovered data after deblending

$$SNR = 10 \log \frac{\|\mathbf{D}^{true}\|_F^2}{\|\mathbf{D}^{true} - \mathbf{D}^r\|_F^2}, \quad (4.22)$$

where \mathbf{D}^{true} is the true data before blending, and \mathbf{D}^r is the recovered data after deblending.

Algorithm 7 Robust MSSA filter via BFGD

```

1: Inputs:
   Data vector  $\mathbf{s}$  of size  $L_x \times 1$ , step length  $\eta$ , and tolerance  $\nu$ ;
   Initial factors  $\mathbf{U}$  of size  $N_r \times p$  and  $\mathbf{V}$  of size  $N_c \times p$ ;
   Initial diagonal matrix of weights  $\mathbf{W} = \mathbf{I}$  of size  $L_x \times L_x$ .
2: while  $k \leq k_{max}$  and  $\|\mathbf{r}\|_2 > \nu$  do
3:    $\nabla_{\mathbf{H}} J = -\mathcal{A}^*(\mathbf{W}(\mathbf{s} - \mathcal{A}[\mathbf{H}]))$ 
4:    $\nabla_{\mathbf{U}} J = (\nabla_{\mathbf{H}} J)\mathbf{V}$ 
5:    $\nabla_{\mathbf{V}} J = (\nabla_{\mathbf{H}} J)^H \mathbf{U}$ 
6:    $\mathbf{U} \leftarrow \mathbf{U} - \eta \nabla_{\mathbf{U}} J$ 
7:    $\mathbf{V} \leftarrow \mathbf{V} - \eta \nabla_{\mathbf{V}} J$ 
8:    $\mathbf{H} = \mathbf{U}\mathbf{V}^H$ 
9:    $\mathbf{r} = (\mathbf{s} - \mathcal{A}[\mathbf{H}])/\sigma$ 
10:  update  $\mathbf{W}$ 
11:   $k \leftarrow k + 1$ 
12: end while
13: Return  $\hat{\mathbf{s}} = \mathcal{A}[\mathbf{U}\mathbf{V}^H]$ 

```

4.3.1 Synthetic example

We synthesize our first example using a finite-difference modelling algorithm that simulates prestack seismic data with a blending factor of two ($BF = 2$). The simulation corresponds to two vessels firing about two sources per pseudo-deblended record. The numerical example consists of 200 receivers and 120 shots. The source is modelled with a Ricker wavelet of a central frequency of 30 Hz. The maximum iteration number is 30 and the stopping parameter $\epsilon = 1 \times 10^{-4}$ (Algorithm 4). We adopted an exponential schedule to decrease the select step length λ of Algorithm 4. The initial step length is computed via $2/(e_{max})$, where e_{max} is the maximum eigenvalue of the operator $\mathcal{B}^* \mathcal{B}$ (Cheng and Sacchi, 2015).

Deblending experiments via the projected gradient descent method were tested with the projection given by the classical MSSA (non-robust) (Cheng and Sacchi, 2015) and the projection introduced in this chapter via the robust MSSA filter. We first run numerical tests with windowing operators \mathbf{W} of different sizes. In our first example, we adopted windows of size $L_t = 350$ time samples and $L_x = 15$ traces. The overlap comprises 70 samples in time and 5 traces in space with linear tapering in time and space. We use the same window size to conduct tests using non-robust and robust MSSA filter projection methods. For the non-robust MSSA filter, heuristically, we also decided to increase the rank of the MSSA filter by one at every five iterations with an initial rank value of two. For the robust MSSA filter, we set the rank constant to a value of $p = 2$ and allow the parameter β of Tukey's biweight loss function to vary with the iteration number. At initial iterations, we adopt a small β value to apply harsh robust denoising of strong interferences and then gradually increase the value of β to allow retrieving weak signals. A strategy that

works for us entails using an initial value $\beta = 2.0$ and then increasing it by 0.1 in each iteration. Deblending results can be found in Figure 4.2. Figure 4.2a and 4.2d shows the initially clean data and the pseudo-deblended data for one common-receiver gather. Many incoherent blending interferences can be observed from the pseudo-deblended data (Figure 4.2d). Figure 4.2b and 4.2c shows the deblending results with the robust and non-robust MSSA filter used as the projection method, respectively. Figure 4.2e and 4.2f corresponds to the error estimation sections. There is one strong event at the top of the seismic gather. Also, the red box displays the actual window size in the seismic gather. We expect to capture a few strong blending interferences in such a small window. The SNR values for PGD with robust and non-robust MSSA filter projections are very similar. For instance, in Figure 4.2c the SNR is 19.6 dB while in Figure 4.2d is 19.2 dB.

We also increase the size of the window to $L_t = 1000$ time samples and $L_x = 50$ traces with an overlap of 200 samples in time and 10 traces in space. This setup shows significant differences in Figure 4.3. Figure 4.3a and 4.3d shows the initially clean data and the pseudo-deblended data for one common-receiver gather. Figure 4.3b and 4.3c shows the deblending results with the robust and non-robust MSSA filter used as the projection method, respectively. Figure 4.3e and 4.3f corresponds to the error estimation sections. For the non-robust MSSA filter, we still increase the rank of the MSSA filter by one at every ten iterations to a maximum where the rank is never larger than $0.5 \times \max(N_r, N_c)$ where N_r and N_c define the size of the Hankel matrix in Algorithm 6. For the robust MSSA filter, we set the rank constant to $p = 10$ and increase parameter β by 0.4 in each iteration with an initial value of $\beta = 2.0$. Now, SNR for PGD with robust MSSA filtering projecting is 21.9 dB, while the classical non-robust MSSA projection yields a lower SNR of 12.6 dB. More blending interferences and signals are included in such a large red rectangular window. The blending noise behaves in this rectangular window like high-amplitude erratic noise. The robust algorithm is more effective for erratic noise attenuation (Chen and Sacchi, 2014). Moreover, in the rectangular window, the events are slightly curved. Therefore, the data violates the linear event assumption made by MSSA. In Figure 4.3d, we observe significant blending interferences unseparated from the data. This phenomenon also can be found in other articles (Maraschini et al., 2012a; Cheng and Sacchi, 2015). On the contrary, the robust MSSA filter projection method yields a cleaner deblending output. The strong blending interferences are attenuated effectively with a large window. Comparing Figure 4.3e and 4.3f, we also can conclude that adopting a robust MSSA denoising projection outperforms the non-robust MSSA filter. Also, when comparing Figure 4.2b with Figure 4.3b, we conclude that using a large window size to contain more information can achieve better deblending results via the robust MSSA projection method. The computational time and SNR comparison of Figure 4.2 and 4.3 can be found in Table 4.1.

Figure 4.4 shows deblending outputs for a zoomed rectangular zone of the data in Figure

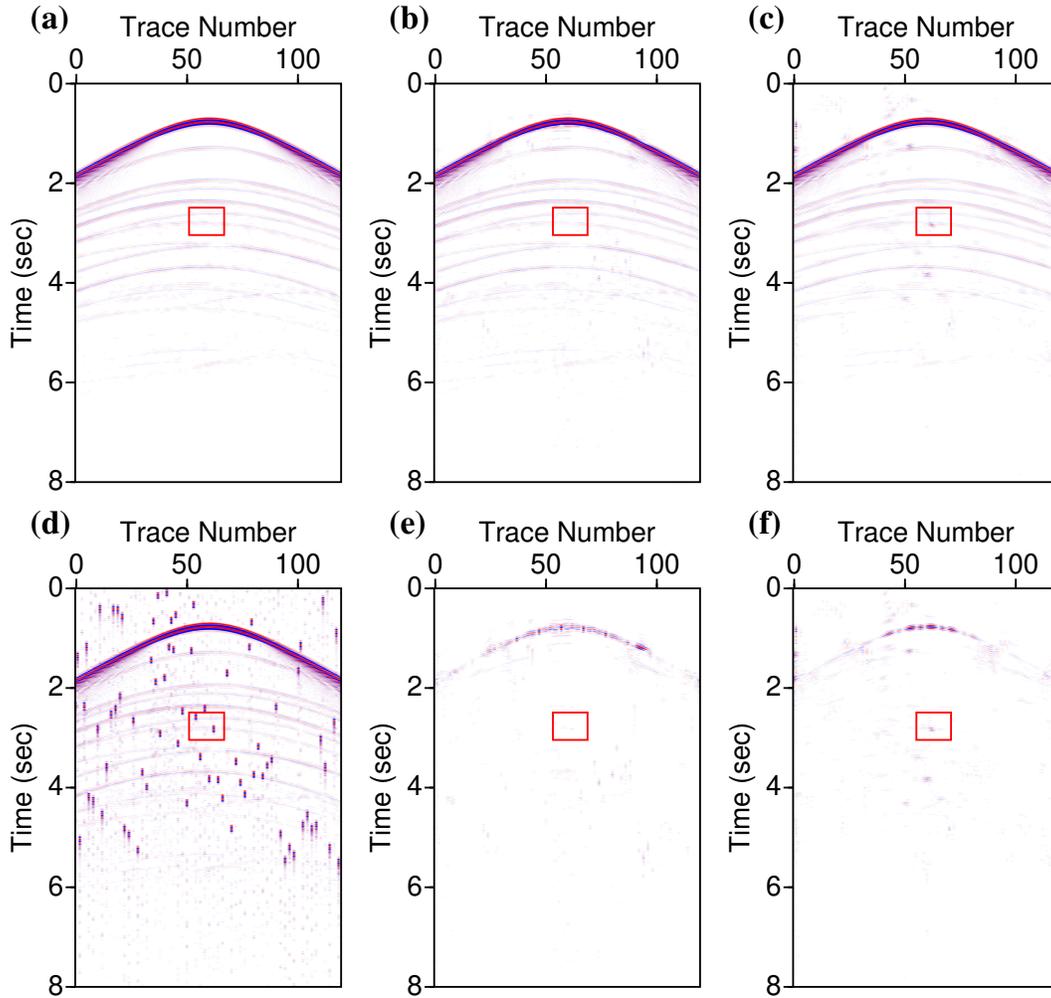


Figure 4.2: Synthetic data example with small window size 350×15 . (a) Clean unblended data in common-receiver gather. (b) Deblending via the PGD method with robust MSSA filtering ($SNR = 19.6$ dB). (c) Deblending via the PGD method with non-robust MSSA filtering ($SNR = 19.2$ dB). (d) Pseudo-deblended data in common-receiver gather. (e) Estimated errors between (b) and (a), corresponding to PGD with robust MSSA. (f) Estimated errors between (c) and (a), corresponding to PGD with non-robust MSSA.

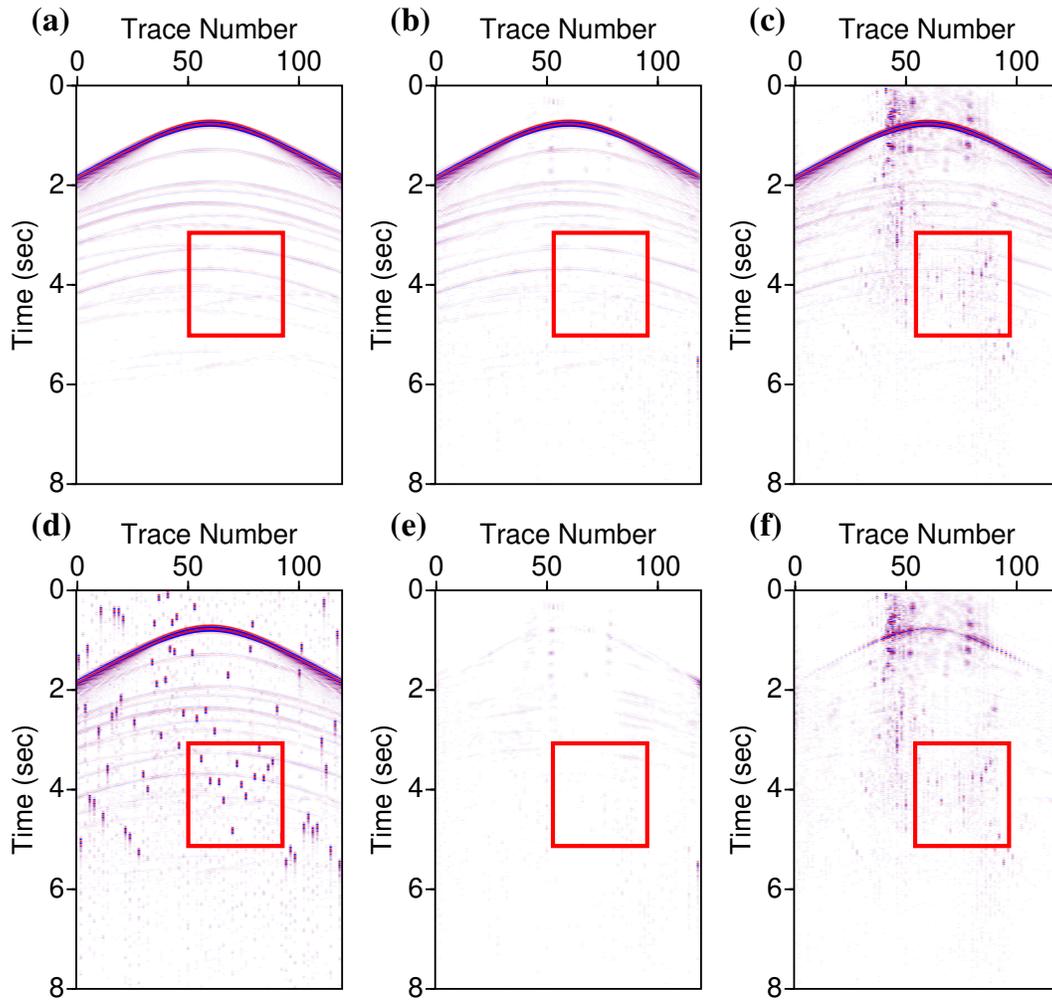


Figure 4.3: Synthetic data example with large window size 1000×50 . (a) The clean, ideal unblended common-receiver gather data. (b) Deblending via the PGD method with robust MSSA filtering ($SNR = 21.9$ dB). (c) Deblending via the PGD method with non-robust MSSA filtering ($SNR = 12.6$ dB). (d) Pseudo-deblended common-receiver gather data. (e) Estimated errors between (b) and (a), corresponding to PGD with robust MSSA. (f) Estimated errors between (c) and (a), corresponding to PGD with non-robust MSSA.

4.3. In Figure 4.4d, we observe significant blending interferences resulting from applying the non-robust MSSA filter in the PGD algorithm. We find fewer interferences for the PGD algorithm with robust MSSA filtering in Figure 4.4c. The final singular value distribution of the Hankel matrix at a specific 30 Hz frequency slice can be found in Figure 4.5. We observe highly amplified singular values for the pseudo-deblended data. The singular values are much larger than those of the clean one. It is clear that the spectra of singular values for this particular window after deblending is similar to the spectrum of the clean data.

We also tested the influence of rank selection in our results; we conducted experiments where we varied the rank value. The rank for the PGD algorithm with robust MSSA filtering is fixed for each run, going from $p = 8$ to 15. For the PGD algorithm with non-robust MSSA filtering, the rank reported in Figure 4.6 corresponds to the initial rank. In other words, we increase the rank with the iteration number of PGD as described above. We find that the PGD algorithm with the robust MSSA denoising projection method is not as sensitive to rank selection as PGD with the non-robust filter. The insensitivity of rank selection for the robust MSSA projection makes the technique suitable for window-based processing, where variations of optimal rank from window to window are highly likely to occur.

Figure 4.7 also verifies our conclusion. We generated 20 different random firing time source schedules for each rank value. We numerically blended the synthetic data for each random schedule and used robust and non-robust MSSA projections in the PGD algorithm to separate the blended data. Figure 4.7 shows the average SNR value of the 20 different realizations versus rank.

For the synthetic example, we find that adopting the PGD algorithm with robust MSSA filtering yields accurate and stable deblending results with less sensitivity of rank selection for relatively large window sizes. Moreover, the convergence of PGD is improved when we adopt the robust MSSA filter. The latter has a simple explanation. Denoising is more aggressive in early iterations of PGD when one adopts the robust MSSA filter. We also observed that the PGD algorithm with robust MSSA filtering requires more oversized windows than its non-robust counterpart. More specifically, if we choose a small window, the robust algorithm cannot decide whether erratic noise or signal is present. Therefore, we advocate adopting relatively large window sizes when applying PGD with a robust MSSA projection.

4.3.2 Real data example

This section applies the proposed method to real seismic data from the Gulf of Mexico (Mississippi Canyon data) with ocean bottom cable acquisition design (Figure 4.8). The data were numerically blended with a blending factor of two ($BF = 2$) to simulate a simultaneous source marine acquisition. We select a part of the data containing the complex

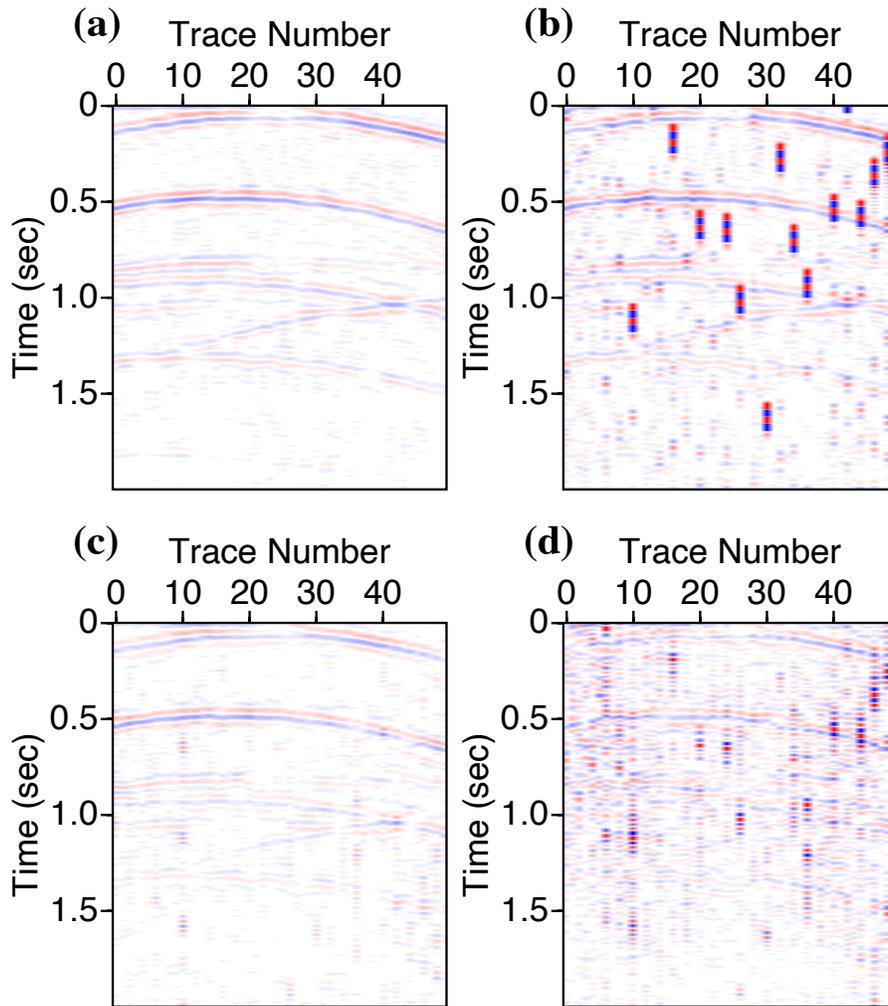


Figure 4.4: Zoomed area corresponding to the rectangle in Figure 4.3. (a) The clean, ideal unblended common-receiver gather data. (b) Pseudo-deblended common-receiver gather data. (c) Deblending via the PGD method with robust MSSA filtering. (d) Deblending via the PGD method with non-robust MSSA filtering.

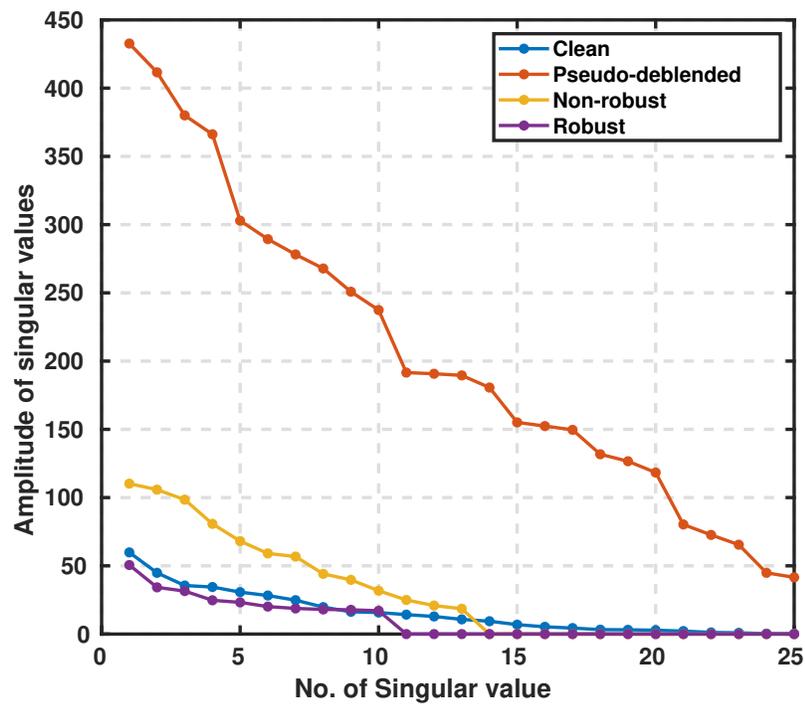


Figure 4.5: Spectra of singular values extracted from the data in Figure 4.4. The spectrum corresponds to the final singular values of the Hankel matrix extracted at 30 Hz.

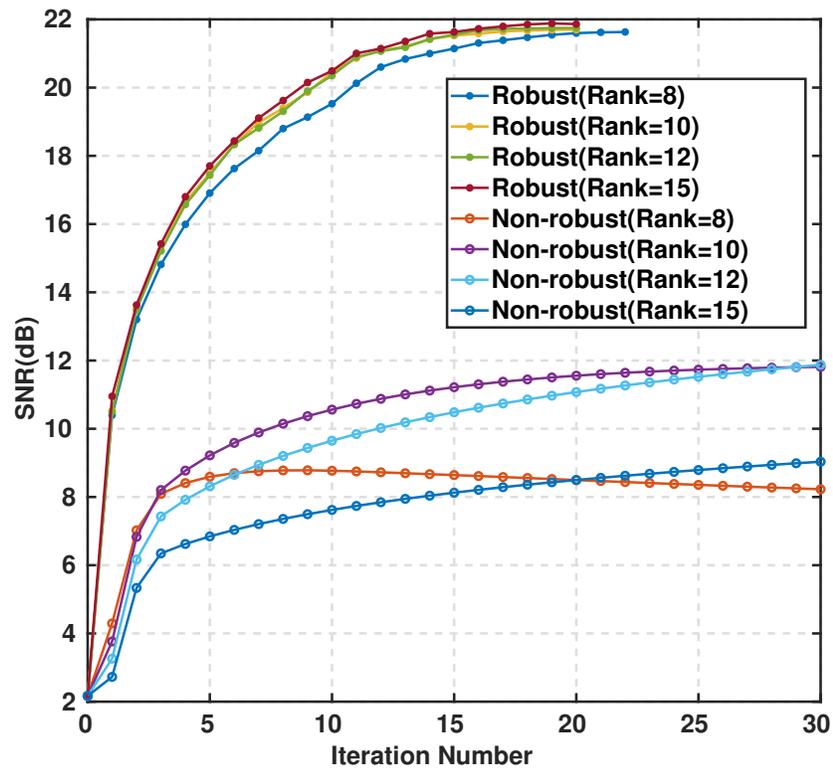


Figure 4.6: Synthetic data example. SNR versus iteration number for varying rank. The rank for each run is fixed for the PDG algorithm with a robust MSSA filter. The PDG algorithm with a non-robust MSSA filter corresponds to the initial rank, which is increased by one every ten iterations.

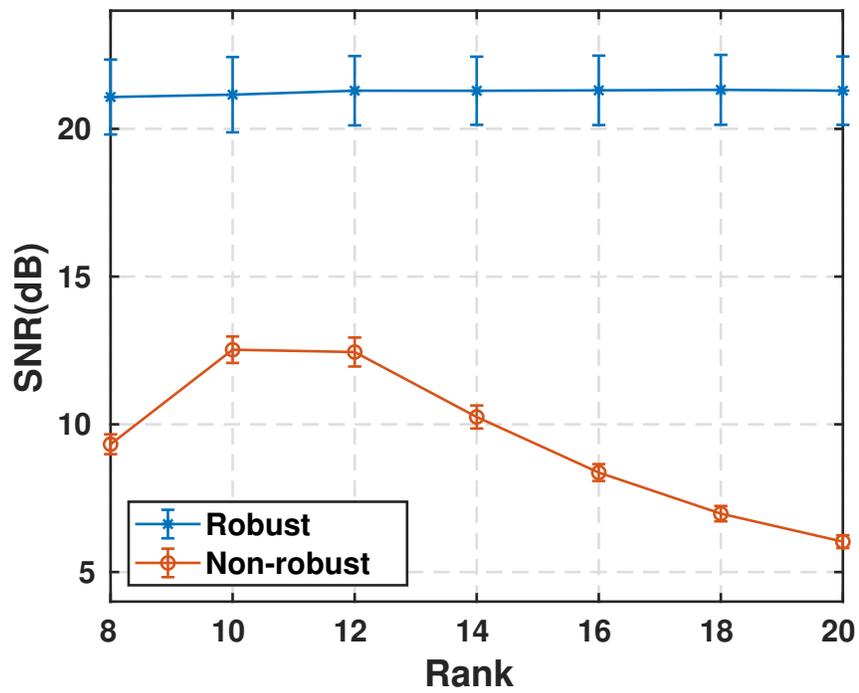


Figure 4.7: Synthetic data example. The average SNR versus rank value for 20 realizations of different random source firing times.

Cases	Methods	Time(sec)	SNR(dB)
Synthetic example (\mathcal{W} : 350×15)	RMSSA	846.31	19.62
	MSSA	393.38	19.23
Synthetic example (\mathcal{W} : 1000×50)	RMSSA	811.77	21.69
	MSSA	164.97	11.81
Real example (\mathcal{W} : 100×50)	RMSSA	272.44	10.54
	MSSA	152.43	8.06

Table 4.1: Computational time and SNR comparison for synthetic and real examples. The computational time listed here is only for one common receiver gather or one common offset gather.

subsurface structures to compare the effectiveness of our algorithms. The data includes 250 shot gathers with 183 receivers each. We use this data to simulate streamer data acquired via a simultaneous-source acquisition; therefore, the robust and non-robust algorithms are applied in common-channel gathers (Peng et al., 2013), which is equivalent to common offset gather as the streamers (receivers) and sources move together. The maximum iteration number was set to 30 and $\epsilon = 1 \times 10^{-3}$ (Algorithm 4). Again, We adopt the strategy used for the previous example to step lengths and rank for the non-robust MSSA filter. For the robust MSSA filter, we also increase the tuneable parameter β by 0.4 in each iteration with an initial value of 2.5. The operator \mathcal{W} extracts relatively large windows of size 100 samples in time and 50 traces. The overlap is of 20 samples in time and 10 traces in space with a linear tapering in time and space.

The original unblended data and pseudo-deblended data for a common-channel gather are shown in Figure 4.9a and 4.9d, respectively. Different from the common shot gather, there exists incoherent blending interferences in the common channel gather. In Figure 4.9d, it is easy to identify that windows A, B, and C should be processed with MSSA filters, likely with a different rank parameter. Figure 4.9b and 4.9c shows the deblending results with a moderate rank value of $p = 10$ for the PGD algorithm with robust and non-robust MSSA filters, respectively. Figure 4.3e and 4.3f portrays the corresponding error estimation sections. We observe signal leakage for PGD with the non-robust projection (Figure 4.9e and 4.9f); this is particularly evident in windows B and C. The non-robust filter has left significant interferences in the deblended data. It is noticeable in window A. In contrast, the robust MSSA filter generates a more accurate deblending result. For instance, if one observes window A, the result is almost equal to the original unblended data. Table 4.1 shows the detailed computational time and SNR comparison of Figure 4.9.

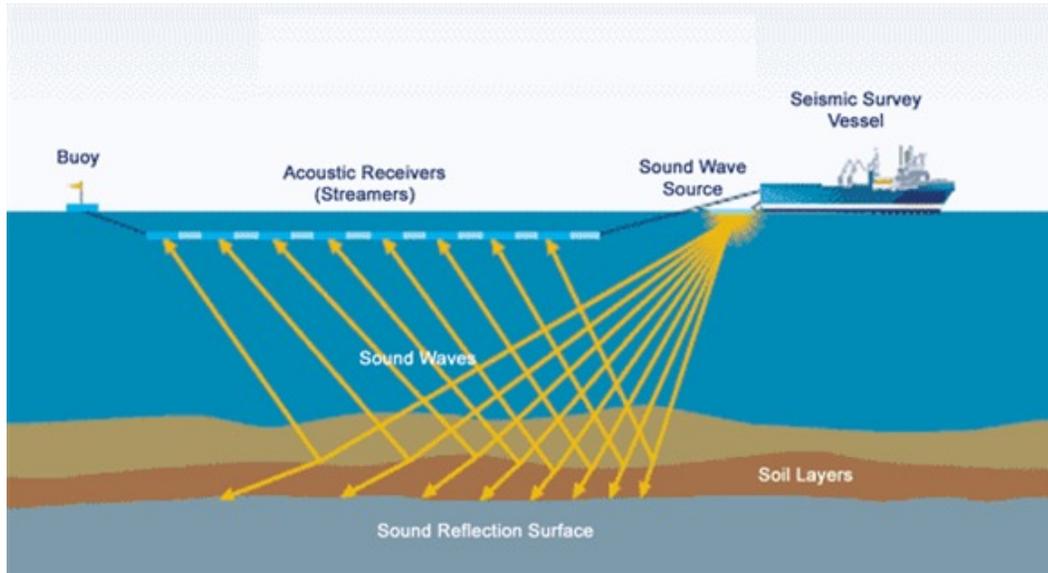


Figure 4.8: Schematic diagram display of ocean bottom cable acquisition of the real data².

To test the influence of rank selection in our results, we vary the rank value from 8 to 20. Again, for the non-robust MSSA filter, the rank 8 to 20 corresponds to initial ranks. In Figure 4.10, we observe that the robust MSSA filter generates higher SNR values and, as discussed before, is less sensitive to rank selection than the non-robust MSSA filter.

4.4 Conclusions

This chapter illustrates an inversion scheme for separating simultaneous source data. We adopted the PGD algorithm with projection given by two flavours of the MSSA filter that we named non-robust (classical) MSSA and robust MSSA. We show the benefit of adopting the robust MSSA filter with a relatively large window size to formulate the window-based projection required by the PGD algorithm. Investigation with numerical experiments shows improved deblending results when one adopts the robust MSSA filter.

The robust MSSA filter described in the chapter is computed via the BFGD approach. We adopted Tukey's biweight loss function to robustize the error in the MSSA filter. It is interesting to clarify that other robust measures of error could have been used. But in general, results using Cauchy or Huber norms are similar to those we can obtain with

²From FishSAFE. Source: <https://fishsafe.org/en/offshore-structures/seismic-surveys/>

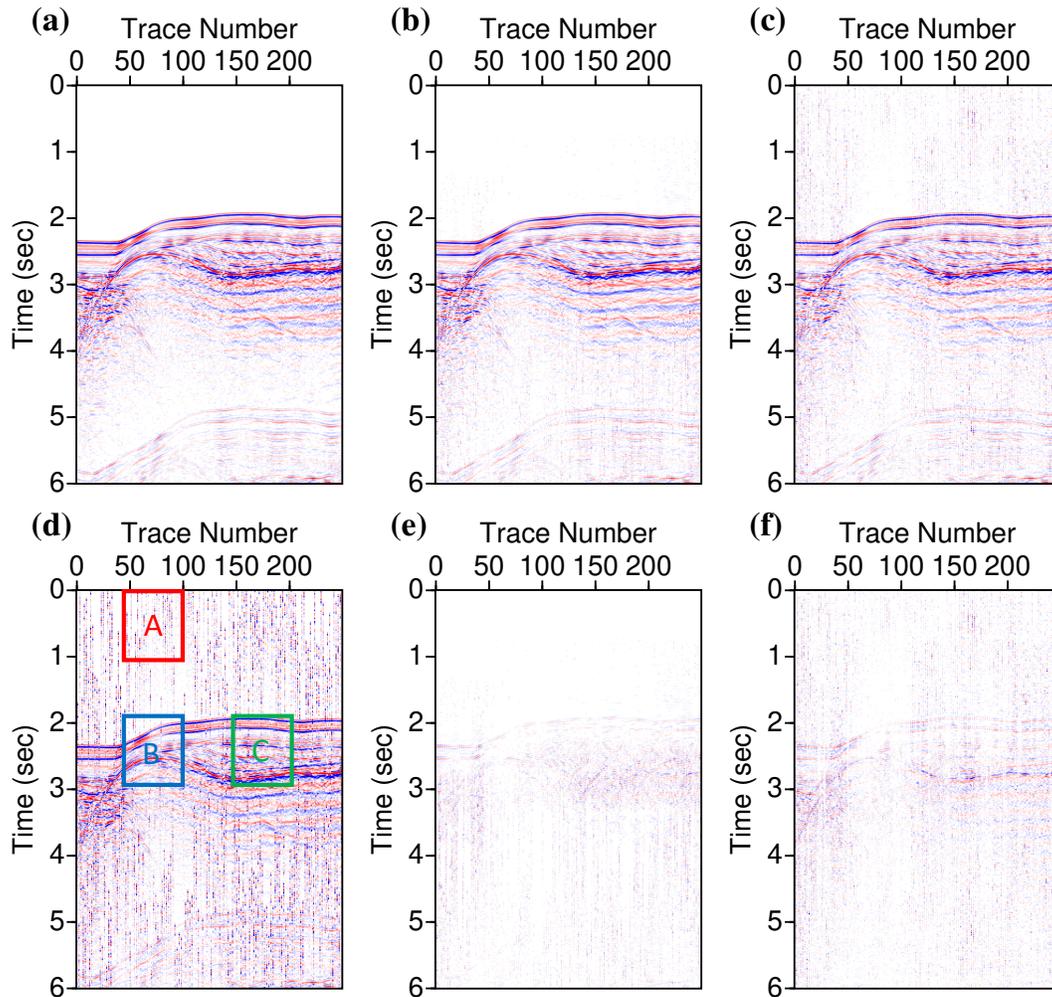


Figure 4.9: Common-channel gather of a Gulf of Mexico survey. (a) Clean unblended data in common channel gather. (b) Deblending via PGD with robust MSSA filtering ($SNR = 10.5$ dB). (c) Deblending via PGD with non-robust MSSA filtering ($SNR = 8.1$ dB). (d) Pseudo-deblended data in common channel gather. (e) Estimated error corresponding to (b). (f) Estimated error corresponding to (c).

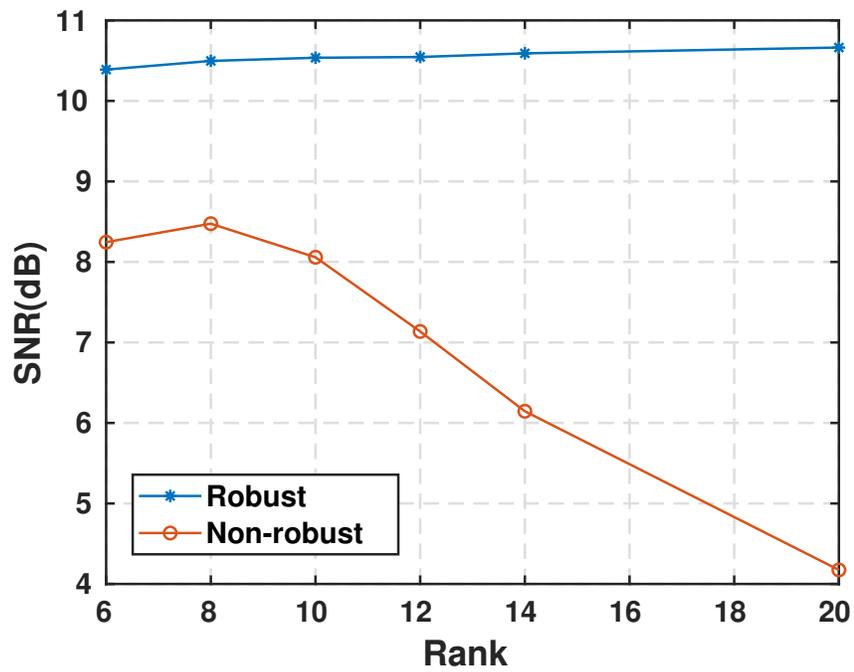


Figure 4.10: Real data example. SNR plot versus rank. We vary the rank value from 6 to 20, and we observe that the robust MSSA filter generates higher SNR values and is less sensitive to rank selection than the non-robust MSSA filter

Tukey's biweight loss. We also discussed a few heuristic considerations that add practicality to our research, such as keeping rank value constant and varying Tukey's biweight parameter β with iteration to avoid erasing weak reflections during the deblending iterative process.

The PGD method can be an expensive algorithm; deblending via PGD is generally time-consuming. The robust version of our algorithm is substantially more costly than the non-robust version. The algorithm with the non-robust MSSA filter requires one SVD per window and per frequency for each PDG iteration. On the other hand, the robust MSSA algorithm is iterative and adds extra cost to the optimization problem. Typical processing times for medium size data sets such as the one from the Gulf of Mexico double when one adopts the robust MSSA projection. This negative facet is counterbalanced by the gain in the denoising power of the robust MSSA filter.

CHAPTER 5

Compressive blended acquisition with irregular-grid geometry ¹

5.1 Introduction

In the category of inversion-based methods for simultaneous source separation, iterative rank-reduction implemented via Multichannel Singular Spectrum Analysis (MSSA) filtering has been proposed for data deblending (Cheng and Sacchi, 2015, 2016; Lin et al., 2021). However, the original algorithm based on the MSSA filter is suitable for solving deblending problems assuming that sources and receivers are deployed on a regular-grid coordinate system. Due to topography or obstacles effects, i.e., mountains, rivers, highways or buildings, and logistic considerations, seismic surveys are generally not perfectly regular. This unavoidable irregularity occurs naturally and is exploited by classical prestack interpolation methods (Liu and Sacchi, 2004; Trad, 2009). Recent research in the field of compressive sensing (CS) relies on purposely constructed irregular grids to enable accurate seismic data reconstruction (Li et al., 2012; Mosher et al., 2012). In other words, CS-based methods rely on designed random sampling schemes that permit reconstruction of the irregular-grid seismic data onto a regular and dense grid (Hennenfent and Herrmann, 2008).

In this chapter, we propose to adopt a recently proposed Interpolated-MSSA (I-MSSA) method to deblend and reconstruct sources in situations where the acquired blended data correspond to sources with arbitrary irregular-grid coordinates. The I-MSSA method permits applying Hankel-based rank-reduction filtering to data consisting of traces with arbitrary spatial coordinates. The presented method solves the problem of deblending and

¹A version of the work in chapter 5 of this thesis has been published in a journal paper: Lin R., Y. Guo, F. Carozzi and M. D. Sacchi, 2022, Simultaneous deblending and source reconstruction for compressive 3D simultaneous-source acquisition data via Interpolated MSSA (I-MSSA): *Geophysics*, **87**, no. 6, 1-53.

source reconstruction/interpolation by one single algorithm. In essence, we propose an iterative rank-reduction deblending method that can honour true source coordinates. In addition, we show how the technique can also be used for source regularization and interpolation. This work focuses on recovering unblended regular-grid data from irregular-grid compressive simultaneous-source data. This work applies to the case where survey irregularities happen due to obstacles and mispositioning of source and when the survey is randomly designed deliberately. Like CS-based surveys, we attempt to reduce acquisition time further by blending sources and using fewer sources, thereby improving the efficiency of field data acquisition (Mosher et al., 2017).

The contribution of this chapter is twofold. First, we study deblending via the projected gradient method with the I-MSSA filter (projection) to cope with irregularly deployed sources. Secondly, we investigate the possibility of conducting compressive source acquisition by analyzing how our algorithm can simultaneously perform deblending and source reconstruction as we decrease the number of simultaneous sources.

This chapter follows the subsequent structure. First, I introduce different grids used in this chapter. Next, I describe the conventional 3D deblending and source reconstruction via MSSA with an extracting operator. I continue with a description of the new proposed I-MSSA method for 3d deblending and source reconstruction. The Kaiser window tapered sinc interpolation operator is introduced in the following section. Finally, I test the two methods (MSSA and I-MSSA) via synthetic and field data.

5.2 Theory

5.2.1 Preliminaries

Our deblending method operates with a denoiser (MSSA) originally devised for data deployed on a regular grid. The MSSA was adopted with an iterative data imputation algorithm where traces are allocated to a regular grid for reconstruction purposes. Not all grid points contain traces during the allocation process, and therefore, the imputation algorithm is used to denoise and reconstruct unobserved traces (Oropeza and Sacchi, 2011). The denoiser was modified to cope with cases where we would like to honour the actual data positions (Carozzi and Sacchi, 2021) via the so-called I-MSSA method.

Our first task, to avoid confusion, is to define the different grids clearly we will use in this chapter:

- The desired regular grid (Figure 5.1a) is the grid with constant spacing where we

assign the reconstructed and deblended seismic data. In other words, this is also the output grid.

- The initial irregular grid (Figure 5.1b) is generated by adding a perturbation to the regular grid. Notice that the number of grid points in the initial irregular grid and desired regular grid are the same. We use this intermediate grid to generate synthetic data in general.
- The observed data (irregular) grid (Figure 5.1c) which is also the grid where the observations are deployed, considered a subset of the initial irregular grid or decimated initial irregular grid (Figure 5.1b).
- Finally, we have the grid associated with data undergoing binning (also called nearest-neighbour assignment). In other words, sources are assigned to grid points from their true spatial location to their nearest grid point of the desired regular grid. The data in binned coordinates correspond to the grid portrayed in Figure 5.1d.

Now that we have defined what we meant by the different grids, we stress that many proposed reduced-rank filtering methods operated on binned regular-grid data. The recently proposed I-MSSA works directly on the observation grid, such as the one portrayed in Figure 5.1c.

Our main goal is to study if iterative reduced-rank deblending based on I-MSSA can simultaneously deblend and reconstruct seismic data acquired on a configuration similar to the one portrayed in Figure 5.1c. More specifically, we would like to record blended data with sources in the format given by Figure 5.1c and develop an algorithm to simultaneously deblend and reconstruct new sources into the grid portrayed in Figure 5.1a. The latter allows algorithms to compress the seismic acquisition by blending and shot decimation.

In a simultaneous-source acquisition, receivers continuously record the response of more than one source at a time (Beasley et al., 1998). For instance, for a conventional seismic acquisition, the acquired survey can be denoted by $u(t, s, r)$ where t , r , and s indicate time, receiver, channel, and source, respectively. For a simultaneous-source acquisition, the trace recorded by one receiver r can be expressed via the following equation

$$b(t, r) = \sum_{i=1}^{N_s} u(t - \tau_i, s_i, r), \quad (5.1)$$

where $b(t, r)$ represents the recorded blended trace acquired by receiver r . Similarly, s_i is the i th source, and τ_i is the random firing time of the i th source.

A compact form of equation 5.1 can be written as follows:

$$\mathbf{b} = \mathcal{B} \mathbf{U} \quad (5.2)$$

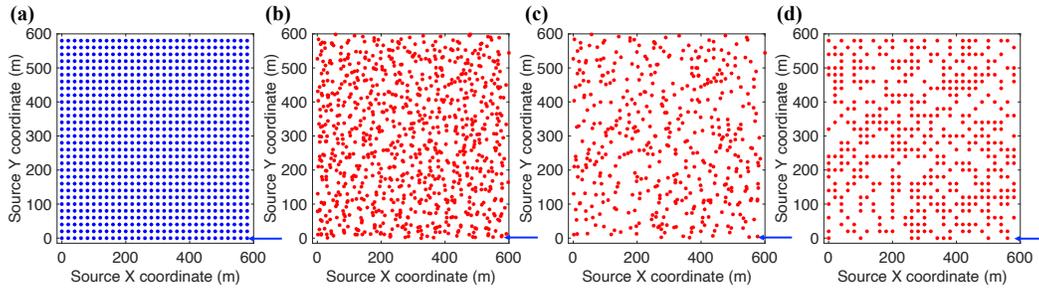


Figure 5.1: (a) Desired regular grid. This is also the output grid of the proposed reconstruction. (b) The irregular grid is formed by perturbing source coordinates of the desired regular grid in (a). (c) For instance, a subset of the irregular grid can be obtained by random decimation of (b). (d) Observations in (c) after assignment to the nearest-neighbour grid point.

where \mathcal{B} denotes the blending operator, including the random jittering time for all sources, and \mathbf{U} represents the unblended individual source responses in true spatial coordinates, and \mathbf{b} is the continuous recorded blended data. The adjoint operator \mathcal{B}^* represents the pseudo-deblended operator (Berkhout, 2008)

$$\tilde{\mathbf{U}} = \mathcal{B}^* \mathbf{b}. \quad (5.3)$$

Pseudo-deblending is equivalent to applying time shifts and separating the blended data records into records of the same time length N_t that correspond to conventional seismic records. The data created via pseudo-deblending, also called combed data (Abma and Foster, 2020), is designated by $\tilde{\mathbf{U}}$. Clearly, pseudo-deblending does not eliminate interferences resulting from responses of different sources (Abma and Foster, 2020). Pseudo-deblended records contain many interferences, frequently called blending noise or blending interferences.

5.2.2 3D deblending and source reconstruction via MSSA with an extracting operator

We consider the case where sources are irregularly or quasi-irregularly distributed on the earth's surface during simultaneous source acquisition. First, we consider a deblending and reconstruction scenario where one adopts a simple sampling operator based on data binning, which can be expressed with the following equation

$$\mathbf{U} = \mathcal{D} \mathbf{D} \quad (5.4)$$

where \mathbf{U} denotes the unblended data (in equation 5.2) honoring true observed irregular-grid coordinates (Figure 5.1c), and \mathbf{D} represents unblended data with desired regular-grid coordinates (Figure 5.1a). We define \mathcal{T} as the extraction operator, which symbolizes two processes: *binning* and *sampling*. The *binning* process means assigning arbitrary irregular-grid source coordinates to nearest-neighbour regular-grid points. The *sampling* process means multiplying by 1 all regular grid points with data and by 0 all regular grid points that are empty (Liu and Sacchi, 2004). Figure 5.2a illustrates how the extraction operator maps data from observed arbitrary irregular-grid points to the desired regular-grid points. Specifically, the *binning* process in Figure 5.2a shows assignment regular grid points back to their correct arbitrary irregular-grid coordinates.

After combining equations 5.2 and 5.4, we get

$$\mathbf{b} = \mathcal{B}\mathcal{T}\mathbf{D}. \quad (5.5)$$

We now pose the deblending and reconstruction as the solution that minimizes the following cost function

$$J = \|\mathbf{b} - \mathcal{B}\mathcal{T}\mathbf{D}\|_2^2. \quad (5.6)$$

Equation 5.6 is an underdetermined problem and does not have a unique solution. Typically, adding a regularization term can guarantee a unique and stable solution. Alternatively, the projected gradient-descent (PGD) method (Cheng and Sacchi, 2016; Lin et al., 2021) can also be adopted to minimize equation 5.6 subject to the application of a projection or filtering operator. The PGD entails a step toward the steepest descent direction followed by a projection. In our work, the projection operator is a denoiser. To summarize, the iterative PGD algorithm (Cheng and Sacchi, 2016; Bolduc et al., 2017; Peters et al., 2019; Lin et al., 2021) is given by

$$\begin{aligned} \mathbf{Y} &= \mathbf{D}^{\nu-1} - \lambda \mathcal{T}^* \mathcal{B}^* (\mathcal{B}\mathcal{T}\mathbf{D}^{\nu-1} - \mathbf{b}) \\ \mathbf{D}^\nu &= \mathcal{P}[\mathbf{Y}]. \end{aligned} \quad (5.7)$$

Notice that the step length λ needs to be estimated to guarantee the convergence of the PGD method (Cheng and Sacchi, 2015; Bolduc et al., 2017). It is also worth mentioning that the PGD method can converge to the global optimal if the projection operator is convex. Rank-reduction is not a convex projection; hence, convergence to the global minimum is not guaranteed. However, as pointed out by many researchers (Cheng and Sacchi, 2015; Peters et al., 2019), PGD still offers a practical means to carry out constrained optimization even when the projection is non-convex. Our algorithm adopts a strategy where the step length is decreased exponentially. This strategy is also discussed in Cheng and Sacchi (2015). Moreover, the initial step size also needs to be properly selected (Fazel, 2002) to ensure that the iterative algorithm will not be trapped in the local minima. The convergence can be

guaranteed when the initial step size

$$\lambda_0 < 2/e_{max}, \quad (5.8)$$

For this specific problem, e_{max} is the maximum eigenvalue of the operator $(\mathcal{B}\mathcal{T})^*(\mathcal{B}\mathcal{T})$ (Ma et al., 2011; Cheng and Sacchi, 2015).

As we mentioned early, \mathcal{P} indicates the projection operator, which in our case is a denoising algorithm based on the MSSA filter (Trickett, 2008; Oropeza and Sacchi, 2011; Cheng and Sacchi, 2015; Lin et al., 2021). In general, the MSSA filter optimally removes noise when the waveforms in the data are represented by linear events in the $t-x$ domains, and it is easy to prove that P linear events in $t-x$ space which correspond to the superposition of P complex exponentials in $f-x$ domain (Oropeza and Sacchi, 2011). Seismic gathers consist of a superposition of waveforms with spatially varying dips. Hence, it is crucial to emphasize that the MSSA filter must be applied on small overlapping windows where one can approximately model reflections as a superposition of events with linear moveout (Cheng and Sacchi, 2015; Lin et al., 2021). Hence, the projection \mathcal{P} indicates applying MSSA filtering on overlapping windows.

We call the whole process conventional MSSA deblending and reconstruction method or simply the MSSA method. In essence, inside the extraction operator, the binning step aims to deploy the observed off-the-grid data into a regular grid where one can easily apply the MSSA projection operator as shown by equation 5.7. This binning strategy introduces errors in the amplitude and phase of the traces, which, as we will show, could deteriorate the reconstruction.

5.2.3 3D deblending and source reconstruction via interpolated-MSSA (I-MSSA)

We now consider the scenario where one would like to deblend and reconstruct sources, but we also require that true source positions be honoured. In other words, the goal is to avoid binning errors, yet we want to apply the MSSA filter. Again, we use the grids illustrated in Figure 5.1 to describe the problem. We consider \mathbf{U} represents unblended data at irregular source positions (Figure 5.1c), and \mathbf{D} denotes the desired unblended data located on the regular-grid coordinates (Figure 5.1a). This time, we connect \mathbf{U} and \mathbf{D} via a local interpolation operator of the form

$$\mathbf{U} = \mathcal{I}\mathbf{D}, \quad (5.9)$$

where \mathcal{W} describes, in operator form, a 2D Kaiser window tapered sinc interpolation operator (Jiang et al., 2017; Carozzi and Sacchi, 2021; Wang et al., 2022) that maps the sought data on their regular coordinates to the data in irregular coordinates (Figure 5.2b). See the next section for a detailed description. Algorithm 8 provides the pseudocode defining the forward \mathcal{W} interpolation operator and its adjoint counterpart \mathcal{W}^* .

Following the rationale used in the previous section, we can recover the positions of the unblended and reconstructed shots by minimizing

$$J = \|\mathbf{b} - \mathcal{B}\mathcal{W}\mathbf{D}\|_2^2. \quad (5.10)$$

Adopting the PGD method with the MSSA projection leads to

$$\begin{aligned} \mathbf{Z} &= \mathbf{D}^{\nu-1} - \lambda \mathcal{W}^* \mathcal{B}^* (\mathcal{B}\mathcal{W}\mathbf{D}^{\nu-1} - \mathbf{b}) \\ \mathbf{D}^\nu &= \mathcal{P}[\mathbf{Z}] \end{aligned} \quad (5.11)$$

The term $\mathcal{B}\mathcal{W}\mathbf{D}^{\nu-1} - \mathbf{b}$ defines a perturbation in the direction of steepest descent. Next, the adjoint interpolator \mathcal{B}^* maps the error into data in the irregular observation grid. Finally, the operator \mathcal{W}^* distributes the data onto the desired regular grid. The fitting goal guarantees data fidelity. In other words, the algorithm honours the actual coordinates of the sources without introducing binning errors. The parameter λ is also decreased exponentially with initial value determined via equation 5.8 with the maximum eigenvalue e_{max} of the operator $\mathcal{W}^* \mathcal{B}^* \mathcal{B}\mathcal{W}$. Again, the projection operator in equation 5.11 is the MSSA filter. However, we call the whole filtering process I-MSSA to stress the inclusion of the operator \mathcal{W} that permits the adaptation of the MSSA method on data that is not deployed on a regular grid (Carozzi and Sacchi, 2021). In our examples, we refer to this method as the I-MSSA method.

5.2.4 The Kaiser window tapered sinc interpolation operator

For the description of the Kaiser window tapered sinc interpolation operator in equation 5.9, we first define the coordinates for the observed irregular-grid source position (red points in Figure 5.2) as

$$\xi_k = \xi(x_k, y_k) \quad (5.12)$$

where ξ_k corresponds to the spatial coordinate of the k th source with $k = 1, \dots, N_s$, where N_s is the total number of acquired sources, and $\xi_k \in \mathbf{U}$.

Similarly, we define the coordinate of the desired regular-grid source positions (blue points in Figure 5.2) as

$$\eta_{(i,j)} = \eta(\hat{x}_i, \hat{y}_j) \quad (5.13)$$

where $\hat{x}_i = \hat{x}_o + i\Delta x$, $\hat{y}_j = \hat{y}_o + j\Delta y$, and (\hat{x}_o, \hat{y}_o) represents the coordinates of the first regular-grid point and Δx and Δy are the x and y grid intervals, $i = 1, \dots, N_{sx}$, $j = 1, \dots, N_{sy}$, and the total number of reconstructed sources is $N_{sx} \times N_{sy}$, and $\eta_{(i,j)} \in \mathbf{D}$.

Algorithm 8 provides pseudocodes for forward and adjoint interpolation operators that are required by the PGD solver. In algorithm 8, $\mathcal{W}(t)$ are the coefficients of the Kaiser window tapered sinc interpolator (Fomel, 2001; Carozzi and Sacchi, 2021), which can be expressed as

$$\mathcal{W}(t) = \text{sinc}(\pi t) \frac{I_0\left(a\sqrt{1 - (t/(N+1))^2}\right)}{I_0(a)} \quad (5.14)$$

where t is the distance from the irregular grid point to the regular grid point, with either argument t_x or t_y given in algorithm 8. The length of the interpolator is $2N + 1$.

Algorithm 8 The Kaiser window tapered sinc interpolator operator

```

1: Inputs:
    $\eta_{i,j} = \eta(\hat{x}_i, \hat{y}_j), \xi_k = \xi(x_k, y_k)$ .
2: for  $k = 1 : N_s$  do
3:    $i = \lfloor (x_k - \hat{x}_o) / \Delta x \rfloor + 1$ 
4:    $j = \lfloor (y_k - \hat{y}_o) / \Delta y \rfloor + 1$ 
5:   for  $i = i - N : i + N$  do
6:     for  $j = j - N : j + N$  do
7:        $t_x = (x_k - \hat{x}_i) / \Delta x$ 
8:        $t_y = (y_k - \hat{y}_j) / \Delta y$ 
9:       if  $\text{adj} = \text{true}$  then
10:        Input =  $\mathbf{U}$ , output =  $\tilde{\mathbf{D}}$ 
11:         $\tilde{\mathbf{D}}(\eta_{i,j})_+ = \mathcal{W}(t_x)\mathcal{W}(t_y)U(\xi_k)$ 
12:       else
13:        Input =  $\mathbf{D}$ , output =  $\mathbf{U}$ 
14:         $U(\xi_k)_+ = \mathcal{W}(t_x)\mathcal{W}(t_y)D(\eta_{i,j})$ 
15:       end if
16:     end for
17:   end for
18: end for

```

5.2.5 Efficient rank reduction via Randomized SVD

The MSSA and I-MSSA filters typically adopt the SVD for its rank reduction step. A faster algorithm is attainable when the SVD is replaced by the Randomized SVD (R-SVD) (Oropeza, 2010; Halko et al., 2011). Algorithm 9 provides the summary of the MSSA algorithm with the option of either using SVD or R-SVD. We assume that operator \mathcal{P} acts on a 3D cube (a CRG) as presented in the PGD algorithm in equations 5.7 and 5.11. (Appendix B introduces a faster and more computational-efficient Multidimensional Singular

Spectrum Analysis (FMSSA) algorithm. Also, a comparative study between I-MSSA and I-FMSSA is provided in Appendix B.)

Algorithm 9 MSSA filter: $\hat{\mathbf{d}} = \mathcal{P}[\mathbf{d}]$

```

1: Inputs:
   CRG:  $\mathbf{d}(t, x, y)$ , rank:  $p$ .
2: Outputs:
   Filtered CRG:  $\hat{\mathbf{d}}(t, x, y)$ .
3: Initialization:
    $\mathbf{D}(\omega, x, y) \leftarrow \text{fft}[\mathbf{d}(t, x, y)]$ 
4: for  $\omega = \omega_{min} : \omega_{max}$  do
5:    $\mathbf{H} = \mathcal{H}[\mathbf{D}(\omega, :, :)]$  Hankelization
6:   if Randomized SVD==true then
7:      $\mathbf{Y} = (\mathbf{H}\mathbf{H}^*)^2\mathbf{H}\mathbf{\Omega}$  Shrink columns of  $\mathbf{H}$  by multiplication
8:     with random matrix  $\mathbf{\Omega}$ 
9:      $[\mathbf{Q}, \mathbf{R}] = \text{qr}(\mathbf{Y})$  qr decomposition
10:     $\mathbf{B} = \mathbf{Q}^*\mathbf{H}$ 
11:     $[\mathbf{U}_p, \mathbf{\Sigma}_p, \mathbf{V}_p] = \text{SVD}(\mathbf{B}, p)$ 
12:     $\mathbf{B}_p = \mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^*$ 
13:     $\mathbf{H}_p = \mathbf{Q}\mathbf{B}_p$ 
14:   else
15:     $[\mathbf{U}_p, \mathbf{\Sigma}_p, \mathbf{V}_p] = \text{SVD}(\mathbf{H}, p)$ 
16:     $\mathbf{H}_p = \mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^*$ 
17:   end if
18:    $\hat{\mathbf{D}}(\omega, :, :) = \mathcal{A}[\mathbf{H}_p]$  Anti-diagonal averaging
19: end for
20:  $\hat{\mathbf{d}}(t, x, y) \leftarrow \text{ifft}[\hat{\mathbf{D}}(\omega, x, y)]$ 

```

5.3 Examples

This section tests the proposed algorithm with synthetic and field data. We will use the following expression to measure the signal-to-noise ratio of the recovered data

$$SNR = 10 \log_{10} \frac{\|\mathbf{D}^{true}\|_F^2}{\|\mathbf{D}^{true} - \mathbf{D}^r\|_F^2}, \quad (5.15)$$

where \mathbf{D}^{true} is the true data before sampling and blending. Similarly, \mathbf{D}^r is the recovered data after debanding and reconstruction. The symbol $\|\cdot\|_F$ denotes the Frobenius norm.

5.3.1 Synthetic example

In this chapter, our first example uses modelled synthetic data to test the proposed de-

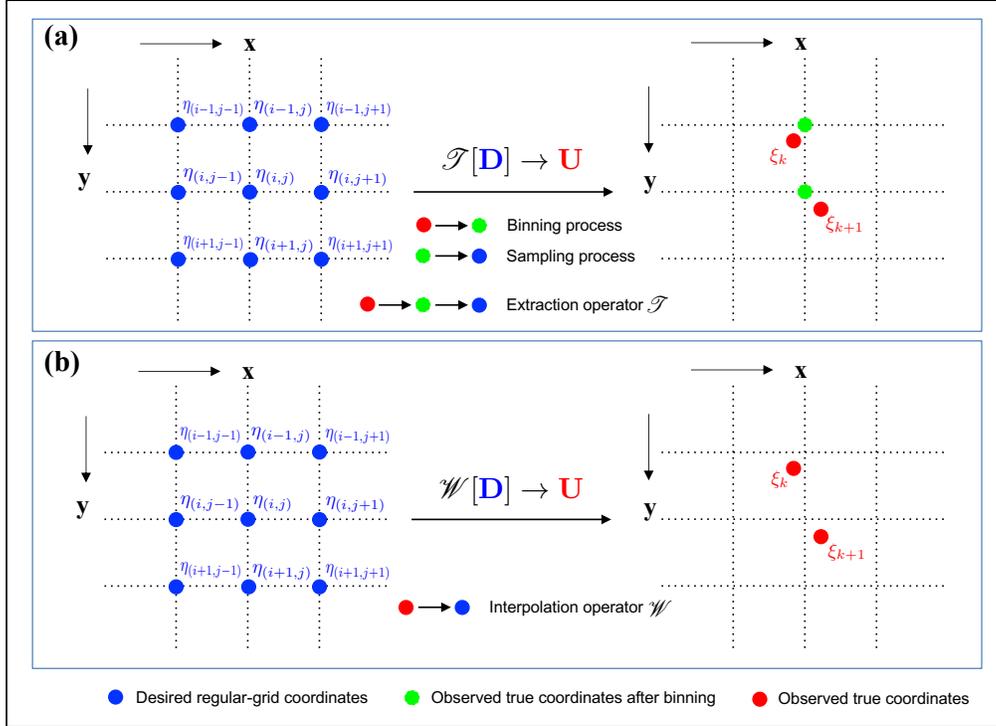


Figure 5.2: Illustration of different sampling operators used in this chapter. (a) The extraction operator, includes *binning* and *sampling*. In red we indicate the given observed true source coordinates. In green we indicate the observed source coordinates after binning. In blue we indicate the desired regular-grid source coordinates. Red \rightarrow Green entails *binning*. Similarly, Green \rightarrow Blue represents *sampling*. (b) In this case, a local interpolation operator maps the observed irregular grid points as a weighted summation of desired data on regular grid points. The weights depend on the distances between observed and desired source positions and are computed via truncated sinc interpolation operator.

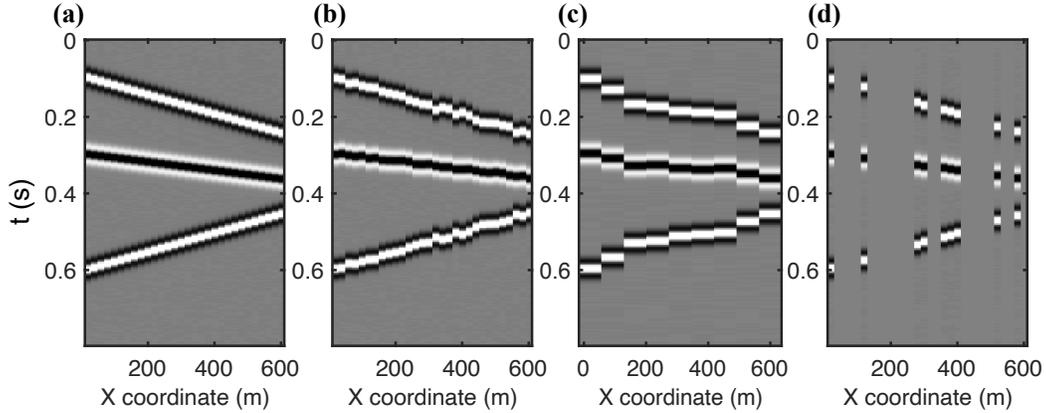


Figure 5.3: The slice display with different grids distribution in Figure 5.1. The blue arrows in Figure 5.1 indicate the positions of selected slices. (a) Clean regular data based on coordinates in Figure 5.1a. (b) Initial irregular data based on coordinates in Figure 5.1b. (c) Observed irregular data, obtained by 50% random decimation of (b), based on coordinates in Figure 5.1c. (d) Observations in (c) after the binning process and corresponding coordinates are displayed in Figure 5.1d.

blending and source reconstruction method. We first synthesize an example containing three dipping linear events to mimic a small 3D patch of a common receiver gather. This assumes that only one receiver keeps recording during the simultaneous-source acquisition experiment with a blending factor $BF = 2$. The regular grid consists of 30×30 source points with interval $\Delta x = \Delta y = 20\text{m}$ in the x - and y - directions. Then, the synthesized common receiver gather was organized in shot gathers and numerically blended. A Ricker wavelet of central frequency 20 Hz was adopted for this synthetic example.

We added a perturbation to the regular grid to obtain an irregular source distribution. Specifically, to avoid the generation of significant gaps, source x and y coordinates were perturbed via random deviates drawn from a uniform distribution in the range $[-\Delta x, \Delta x]$ and $[-\Delta y, \Delta y]$. The geometry of the observed source coordinates is displayed in Figure 5.1c. Similarly, Figure 5.1a is our desired regular-grid source distribution. For completeness, we also show the different stages used for generating the observed irregular data in Figure 5.3 (Blue arrows in Figure 5.1 indicate the selected positions). We first generate the regular-grid data (Figure 5.3a). Next, we add random perturbations to the regular-grid points to account for the initial irregular data (Figure 5.3b). Then, we randomly decimate the initial irregular data to generate the decimated observed irregular-grid data (Figure 5.3c). Finally, we assign the irregular-grid points to the nearest-neighbour regular grid points to obtain the binned decimated irregular data for data display only (Figure 5.3d).

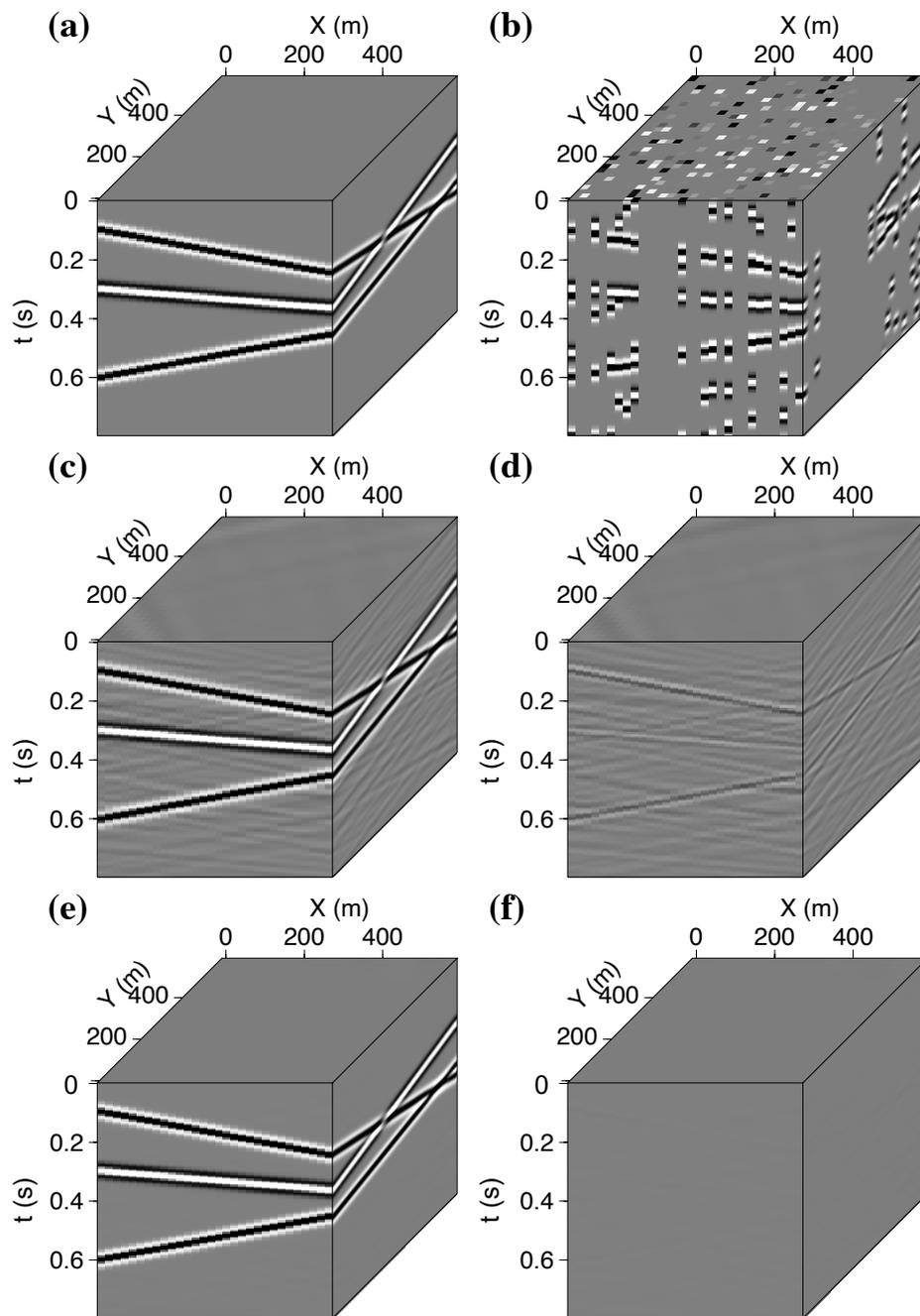


Figure 5.4: Reconstruction and debrending results for a clean synthetic data example with a randomly 50% decimated volume for one common receiver gather. (a) Clean regular data volume. (b) Pseudo-debanded data volume after binning. (c) Result of the conventional 3D MSSA reconstruction and debrending. (d) Residuals between the clean volume (a) and (c). (e) Debanded and reconstructed data via the I-MSSA algorithm with sinc interpolation. (f) Residuals between the clean volume (a) and (e).

We first consider noise-free 3D data. For this synthetic example, the selected rank for the MSSA filter is $p = 3$ (which equals the number of linear events in $t - x$ domains). The step size is exponentially decreased to guarantee the algorithm convergence, and the power method (Golub and Van Loan, 1996) is used to estimate e_{max} iteratively in equation 5.8. Figure 5.4 is the 3D volume corresponding to the first test. Figure 5.4a shows the noise-free regular-grid data used as a reference. Figure 5.4b shows the observed irregular-grid pseudo-deblended data after binning. Figure 5.4c and 5.4d show deblending and reconstruction results via the MSSA method and its corresponding error, respectively. Figure 5.4e and 5.4f illustrates deblending and reconstruction results by the proposed I-MSSA method and the corresponding error, respectively. Comparing Figure 5.4d with 5.4f, we observe significant signal leakage when one adopts the MSSA method ($SNR = 15.90$ dB). Conversely, deblending and reconstruction via the I-MSSA method produce negligible signal leakage ($SNR = 43.56$ dB). Also, the blending interferences are effectively eliminated, and the missing seismic events are fully recovered. The difference in quality results from the introduction of small-time shifts introduced by binning, which could compromise the lateral coherence of signals when adopting the MSSA algorithm.

Figure 5.5 shows 2D slices of the data in Figure 5.4. Figure 5.5b displays the observed irregular-grid pseudo-deblended data after binning. We can observe the random decimation of the traces and the blending interferences in the pseudo-deblended common receiver gather. Figure 5.5c-5.5f presents a slice of the data after deblending and reconstruction. The I-MSSA method can effectively eliminate the blending interferences and fully recover the seismic events with negligible leakage. We conclude that the I-MSSA method outperforms the MSSA method when processing data with irregular source positions.

Figure 5.6 shows the quality of the reconstruction in terms of SNR versus the iteration number of PGD. In our numerical experiments, we set a maximum number of iterations as a large fixed number to verify the convergence performance of PGD methods. The step length of PGD decreased exponentially, starting with an initial step length that was sophisticatedly estimated according to equation 5.8. To be specific, the following examples are tested with appropriate following stopping criterion, either $\|\nabla J^\nu\| / \|\nabla J^1\| < \eta$ ($\eta = 10^{-2}$) or the maximum iteration number is reached.

To complete our tests, we also add Gaussian noise with $SNR_i = 1$ dB to the clean 3D data in Figure 5.4b. The input signal-to-noise is given by $SNR_i = 10 \log_{10} (\|\mathbf{D}^c\|_F^2 / \|\mathbf{N}\|_F^2)$ where \mathbf{N} represents Gaussian additive noise and \mathbf{D}^c is the noise-free data. The deblending and reconstruction results can be found in Figure 5.7 and the corresponding slices are displayed in Figure 5.8. Similarly, Figure 5.9 shows the SNR versus iteration for this example. As expected, for data contaminated with noise, the achievable SNR is lower than in the noise-free case (Figure 5.6).

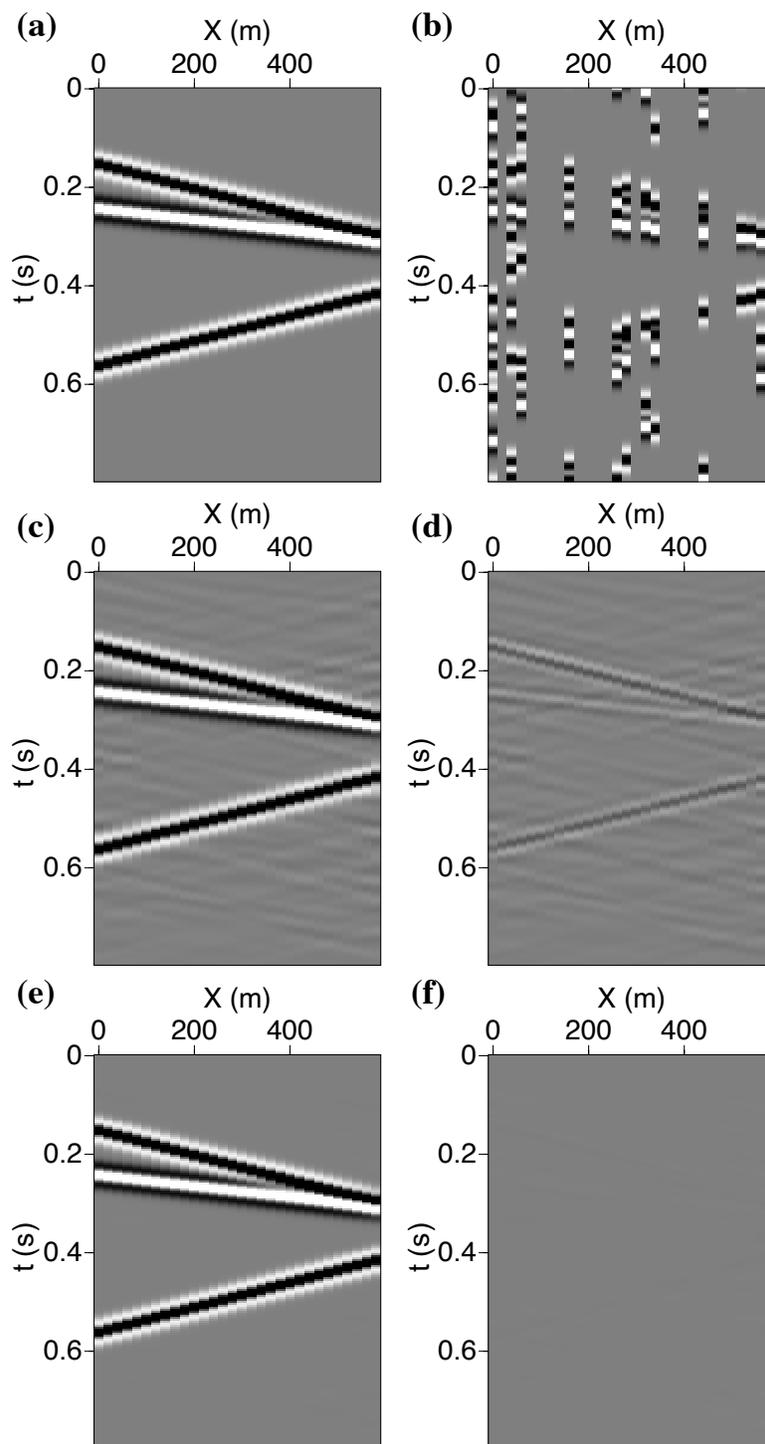


Figure 5.5: An x-slice showing reconstruction and debrending results for the synthetic example in Figure 5.4. (a) Clean regular data. (b) Pseudo-debanded data after binning. (c) Result of the conventional 3D MSSA reconstruction and debrending. (d) Residuals between (a) and (c). (e) Reconstruction and debrending results were calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between (a) and (e).

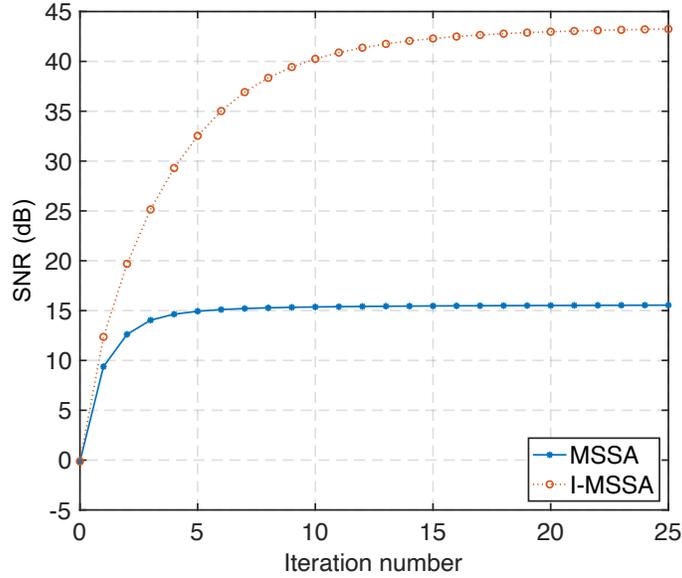


Figure 5.6: Convergence curves showing SNR versus iteration for PGD using MSSA and I-MSSA filters for reconstruction and debrending.

5.3.2 Behaviour of our algorithms versus decimation and rank-reduction solver

In Figure 5.10, we display the average SNR versus decimation for debrending and source reconstruction via MSSA and I-MSSA. The error bars were obtained by running the algorithm for 50 realization associated with different simple synthetics composed of linear events (similar to those in Figure 5.4). At this point, we also would like to mention that the rank-reduction step of the MSSA filter \mathcal{P} in equations 5.7 and 5.11 (PGD solver) can be implemented via different means. The original MSSA reconstruction and denoising article by Oropeza and Sacchi (2011) adopted the Singular Value Decomposition (SVD) and the Randomized Singular Value Decomposition (R-SVD) (Liberty et al., 2007). Results obtained via the SVD and R-SVD are incredibly similar. However, the computational cost decreases when one adopts the R-SVD to perform rank-reduction inside the MSSA filter \mathcal{P} . Table 5.1 shows a comparison of computational time and the SNR figure of merit for MSSA (SVD) and MSSA (R-SVD) algorithms for debrending and reconstruction. In this case, we used CRG of size 200 time samples with a 30×30 source pattern perturbed and decimated as in the previous examples. We observed no significant differences in the reconstruction quality between these two rank-reduction solvers. However, MSSA and I-MSSA are faster when the rank-reduction step is implemented via the R-SVD algorithm. Algorithm 9 provides

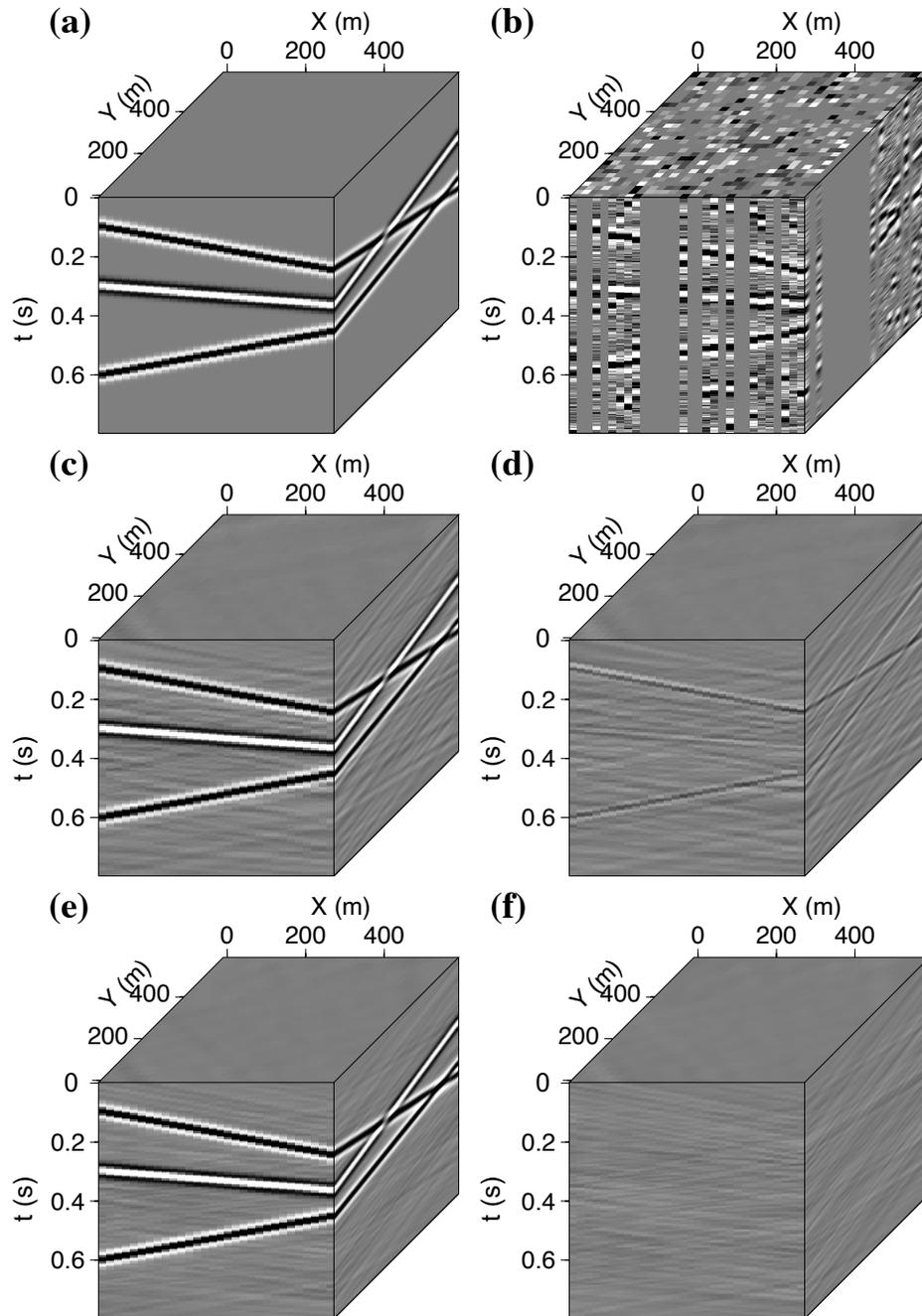


Figure 5.7: Reconstruction and debrending results for synthetic data example contaminated with noise. The data was decimated randomly at a level of 50%. (a) Clean regular data volume. (b) Pseudo-debanded noisy data volume after binning. (c) MSSA reconstruction and debrending. (d) Residuals between the clean volume (a) and (c). (e) Reconstruction and debrending are calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between the clean volume (a) and (e).

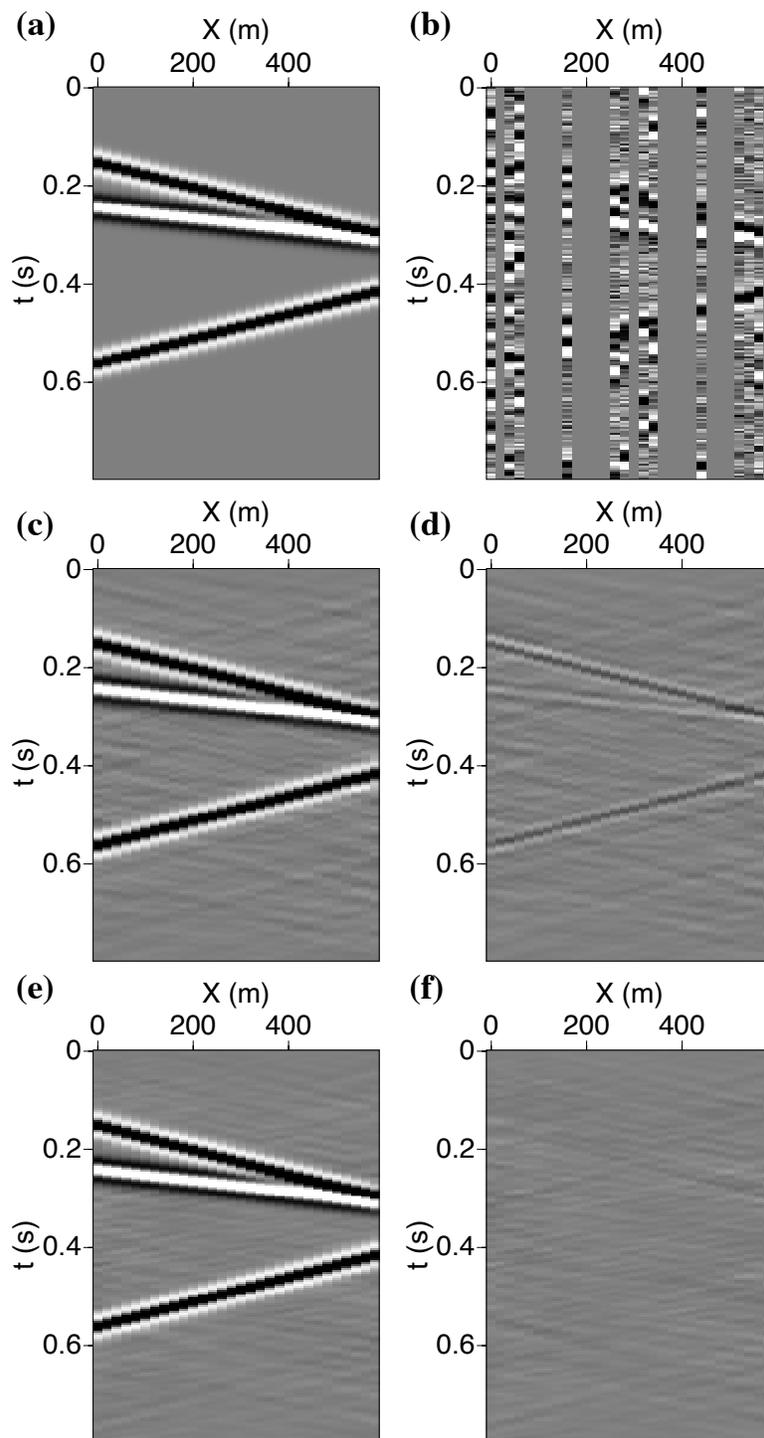


Figure 5.8: One x-slice display of reconstruction and debrending results for noisy synthetic example in Figure 5.7. (a) Clean regular data. (b) Pseudo-debanded data after binning. (c) Result of the conventional 3D MSSA reconstruction and debrending. (d) Residuals between (a) and (c). (e) Reconstruction and debrending results were calculated via the I-MSSA algorithm with sinc interpolation. (f) Residuals between (a) and (e).

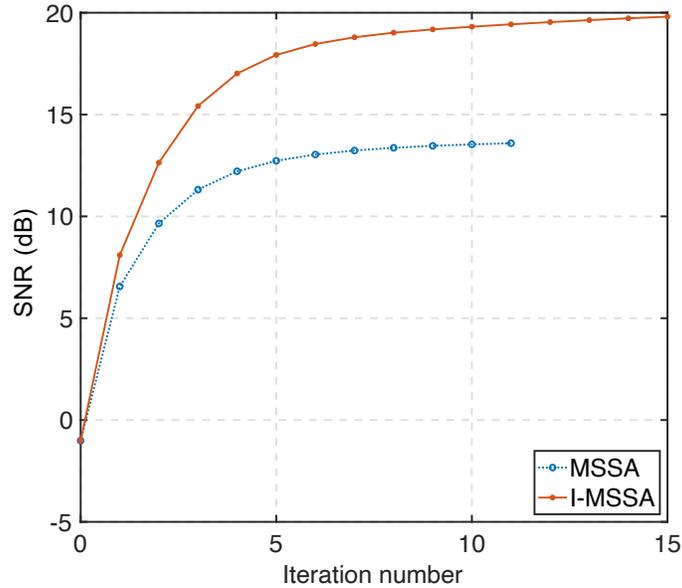


Figure 5.9: Convergence curves showing SNR versus PGD iteration with MSSA and I-MSSA filters. These results correspond to the data in Figure 5.7.

pseudocodes for the MSSA (SVD) and MSSA (R-SVD) algorithms.

The following tests are based on the MSSA (R-SVD) algorithm as a projection operator to avoid ambiguity. To test the stability and robustness of the methods, we generated 50 different random decimation schedules for each percentage of decimation. Figure 5.10 shows the average SNR value of the 50 realizations versus the decimation level. Through this figure, up to 60% decimation, the I-MSSA algorithm can obtain a much higher SNR figure of quality (above 40 dB). When the percentage of decimation increases to 80%, both MSSA and I-MSSA methods result in a deterioration of SNR . This can be explained by insufficient input signal information to MSSA for fully recovering the data.

Our last synthetic test adopts a complex 3D synthetic data example with a blending factor $BF = 2$. Figure 5.11 shows the 3D velocity model we adopt for generating the synthetic data. Figure 5.12 delineates the observed irregular-grid coordinates system, including 2500 source points. Figure 5.12b shows the desired output in a regular-grid coordinate system, which includes $50 \times 100 = 5000$ source points. We adopt a local window strategy (Cheng and Sacchi, 2015; Lin et al., 2021) to satisfy the linear event assumption made by the MSSA filter. The size of the local windows is 200 time samples and $L_x = 20$ traces in the x direction and $L_y = 20$ traces in the y direction. The overlap is of 50 samples in time and 8 traces in x and y directions with linear tapering in time and space. The step side of the PGD

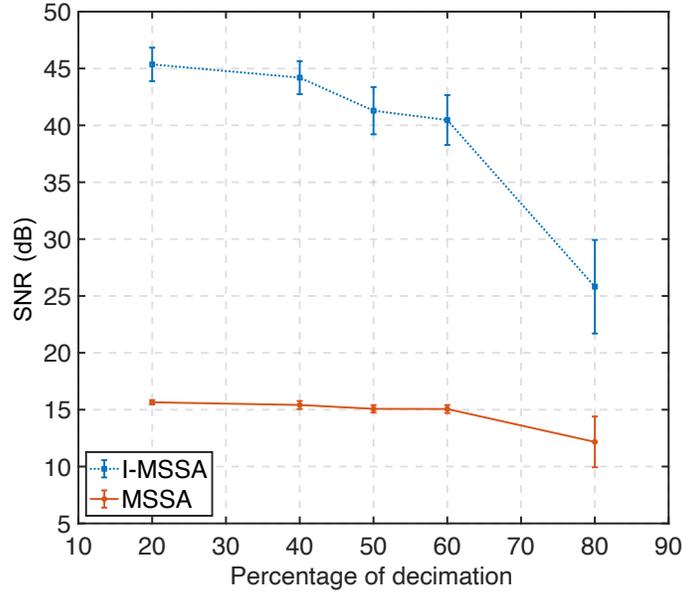


Figure 5.10: The average SNR comparison of MSSA and I-MSSA methods for reconstruction and deblending with different percentages of decimation. For each decimation percentage example, the SNR values are calculated via 50 realizations.

Cases	Methods	Time (sec)	SNR (dB)
Original method	MSSA(SVD)	38.04	15.90
	MSSA(R-SVD)	17.18	15.89
Proposed method	I-MSSA(SVD)	40.56	43.61
	I-MSSA(R-SVD)	22.58	43.56

Table 5.1: Computational time and SNR comparison for deblending and source reconstruction with different rank-reduction solvers.

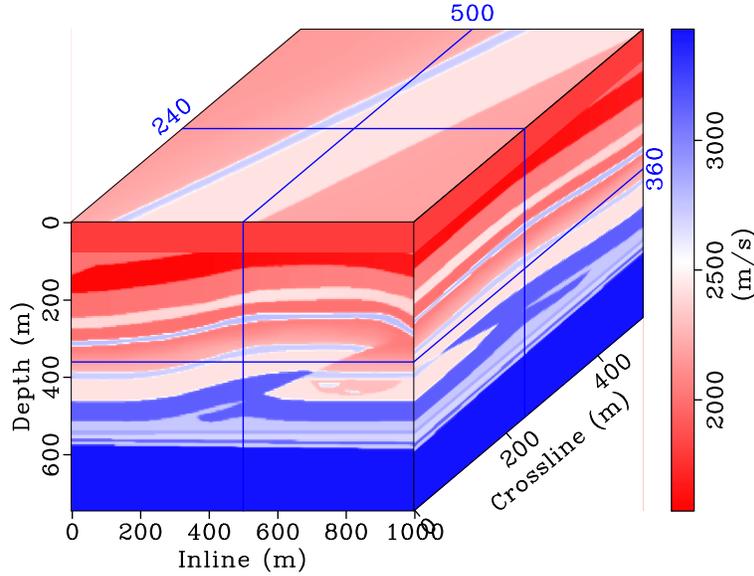


Figure 5.11: Velocity model used for the synthetic example.

method was exponentially decreased, and the power method is also used to estimate e_{max} in equation 5.8. We set the rank of the MSSA filter to $p = 5$ for each window. The deblending and reconstruction results can be found in Figure 5.13. Figure 5.13d shows the observed irregular-grid pseudo-deblended data after binning. Figure 5.13b shows deblending and reconstruction result via the proposed I-MSSA method ($SNR = 14.32$ dB). Figure 5.13e displays deblending and reconstruction result by the MSSA method ($SNR = 11.56$ dB). The I-MSSA method can eliminate the blending interferences and fully recover the seismic events. Comparing the difference plots in Figure 5.13c with 5.13f, we observe additional coherent signal leakage when adopting the MSSA method. We also plot the difference between the two methods (Figure 5.13g), which means that even without access to ground-truth data (Figure 5.13a) as a reference, which is a common problem for field data, we can still conclude that the I-MSSA method outperforms the MSSA method as I-MSSA shows less signal leakage than MSSA.

5.3.3 Real data example

We also test our methods with a field 3D blended-acquisition data from Oman (Song et al., 2019). Again, we also only use one receiver to test our algorithm. The survey area contains

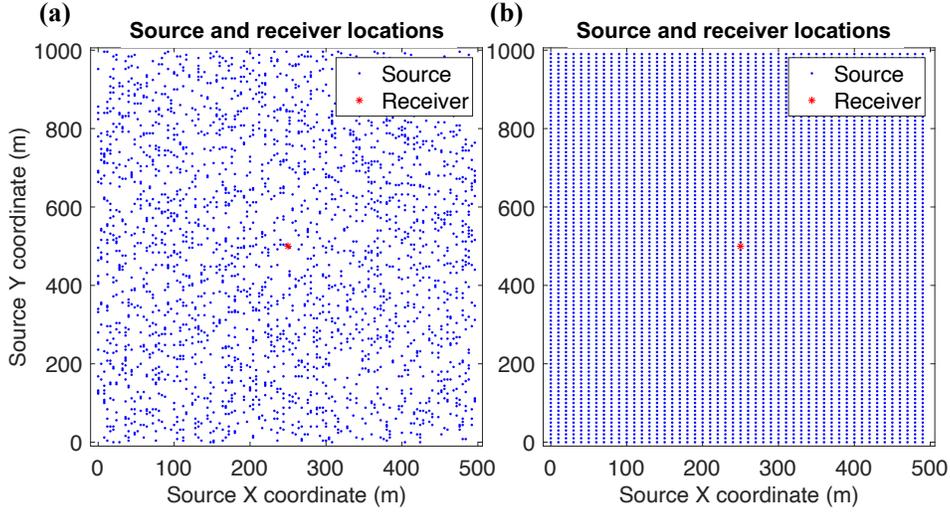


Figure 5.12: Coordinate systems for synthetic example. (a) Observed irregular-grid coordinates system including 2500 source points. (b) The desired regular-grid coordinate system with $50 \times 100 = 5000$ source points.

22408 source points operated by 12 Vibroseis vehicles simultaneously. Figure 5.14 delineates the coordinates of sources. The mean interval between sources and source lines is 25 m. Figure 5.14a shows the field source coordinate distribution. Figure 5.14b shows the source coordinate distribution after 20% random decimation, resulting in 17927 source points. Figure 5.14c displays the desired output with a regular-grid coordinate system, which includes $398 \times 62 = 24676$ source points. For this real data example, again, windowing is adopted when applying the MSSA filter. The size of the windows is 200 time samples and $L_x = 45$ traces in the x direction and $L_y = 25$ traces in the y direction. The overlap comprises 80 samples in time, 20 traces in the x direction, and 10 traces in the y direction. The step size of the PGD method, as described in previous examples, is decreased exponentially with iteration. We also consider an initial rank value $p = 60$ for the MSSA filter. Heuristically, we found that increasing the rank of the MSSA filter with iteration leads to solutions with reduced signal leakage (Cheng and Sacchi, 2014). Specifically, we increase the rank every five iterations by one. The process uses about 10 – 15 iterations to converge. Hence, the rank is always less than the minimum size of the block Hankel matrix. The spatial window size in this example leads to block Hankel matrices of size 299×299 .

Our test aims to compare the I-MSSA method with the MSSA method for deblending and reconstruction from the observed decimated irregular-grid data (Figure 5.14b). The results can be found in Figure 5.15. Figure 5.15a is the observed decimated data based on

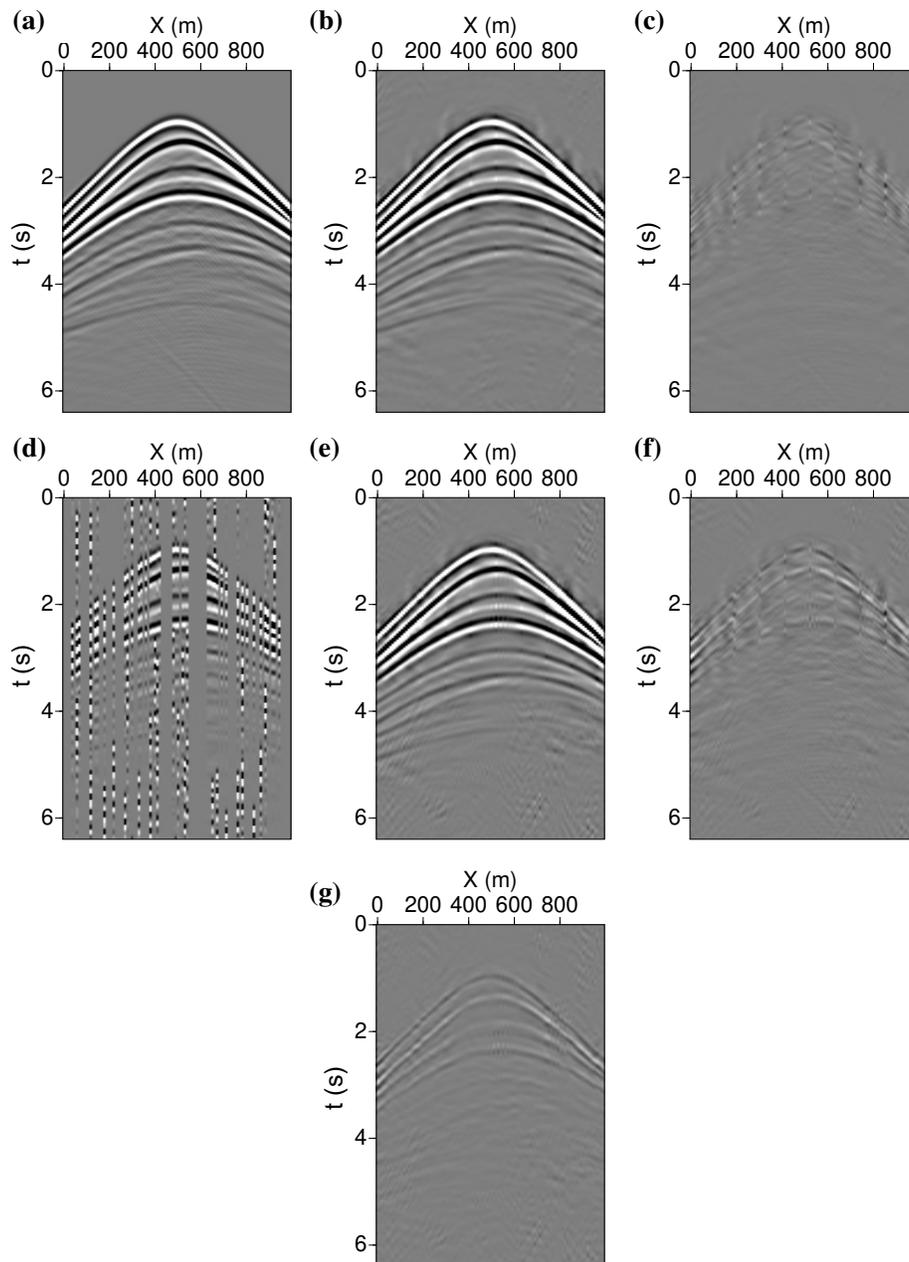


Figure 5.13: Deblending and reconstruction of the synthetic data with the source distribution in Figure 5.12a. (a) Ground truth based on regular-grid distribution. (b) Deblending and reconstruction via the I-MSSA method with a resulting $SNR = 14.32$ dB. (c) Difference between (a) and (b). (d) Observed pseudo-deblended irregular-grid data after binning. (e) Deblending and reconstruction result via the MSSA method with a resulting $SNR = 11.56$ dB. (f) Difference between (a) and (e). (g) Difference between (b) and (e).

the source coordinates distribution in Figure 5.14b. Figure 5.15b presents the deblending and reconstruction results via the I-MSSA method. Similarly, Figure 5.15c displays the deblending and reconstruction results via the MSSA method. Figures 5.16 and 5.17 show zoomed sections of Figure 5.15. We observe that the blending noise has been effectively eliminated, and the decimated sources have been fully reconstructed.

Due to the lack of ground truth data as a reference, it is hard to evaluate signal leakage for I-MSSA precisely (Figure 5.15b) and MSSA (Figure 5.15c). For this dataset, the difference between true source coordinates and desired regular coordinates is not significant, which causes the evaluation to be much more difficult to be carried out. In these circumstances, we calculate the difference between I-MSSA and MSSA methods (Figure 5.15d) and evaluate their difference and quality visually, as what we did for the synthetic example in Figure 5.13g. In Figure 5.15d, we observe coherent signals (same as Figure 5.13g), and we conclude that these events are a consequence of the errors introduced by data binning when one adopts the MSSA method. Given that binning assigns coordinates to grid points via a crude nearest-point interpolation, one expects some amplitude distortion. When sources are not deployed on a perfectly regular grid, one should adopt I-MSSA rather than MSSA despite the minor differences we might obtain.

We also compare the deblending and reconstruction results obtained via the I-MSSA method when the algorithm is run with the full data (from Figure 5.14a to 5.14c) and decimated data (from Figure 5.14b to 5.14c). The results are shown in Figure 5.18 and, clearly, not knowing the ground truth makes it difficult to guarantee that we can safely decimate source positions. However, the results are pretty similar, providing encouraging confidence in them.

Figure 5.18a displays the deblending and reconstruction results obtained via the I-MSSA method when the full-field data (Figure 5.14a) is used. Figure 5.18b shows the deblending and reconstruction results with the I-MSSA method when the field data is decimated (Figure 5.14b). The corresponding difference can be found in Figure 5.18c. Even though we manually decimate 4481 source points from our survey area, we still can fully recover the unblended data associated with the desired regular grid.

5.4 Conclusions

This chapter illustrates an inversion scheme for deblending and source reconstruction that honours actual source coordinates. For this purpose, I have adopted Projected Gradient Descent optimization with a projection operator given by a reduced-rank MSSA filter. I have examined two cases. In one case, we use an extraction operator that assigns true source position to grid points via nearest-neighbour assignment (*binning process*). This process

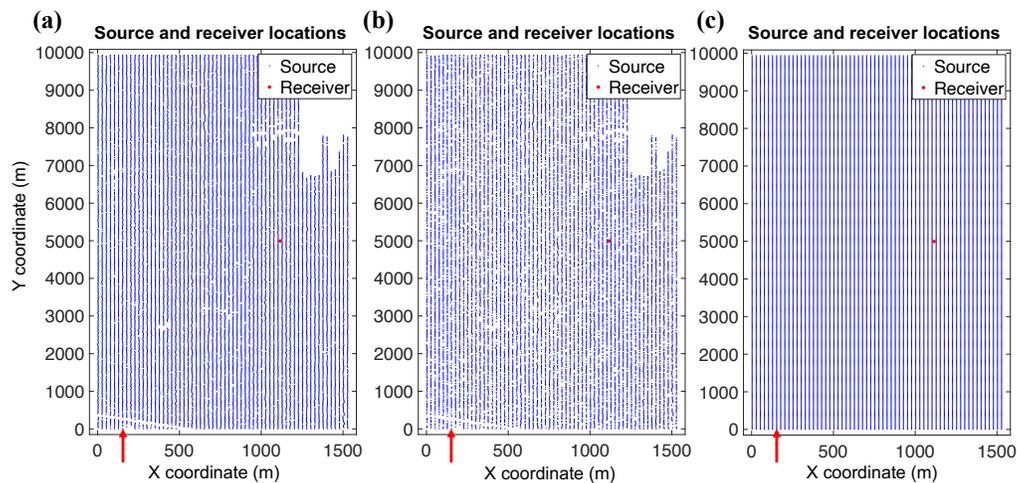


Figure 5.14: Coordinate systems for the real data example. (a) True data coordinates, including 22408 source points. (b) The source coordinate distribution after 20% random decimation contains 17927 source points. (c) The desired output on a regular-grid coordinate system with $398 \times 62 = 24676$ source points.

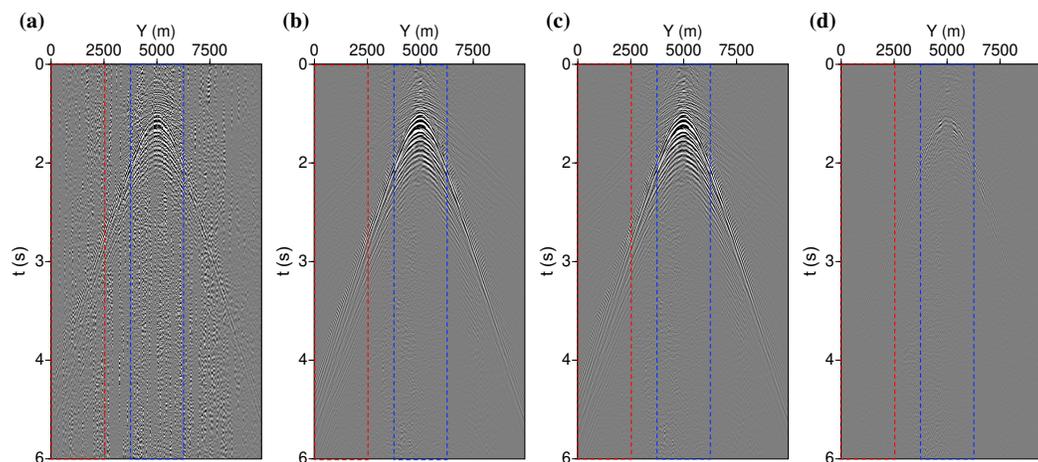


Figure 5.15: Deblending and reconstruction of the data with the source distribution in Figure 5.14b. The red arrows in Figure 5.14 indicate the selected slices. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) The result was obtained via the MSSA method. (d) Difference between (b) and (c).

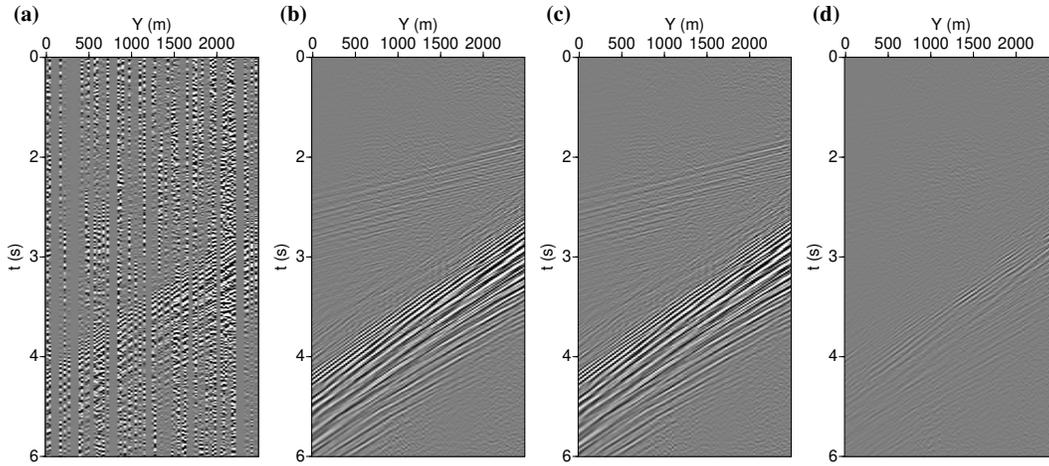


Figure 5.16: Zoomed area corresponding to the red rectangle (left part) in Figure 5.15. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) Results were obtained via the MSSA method. (d) Difference between (b) and (c).

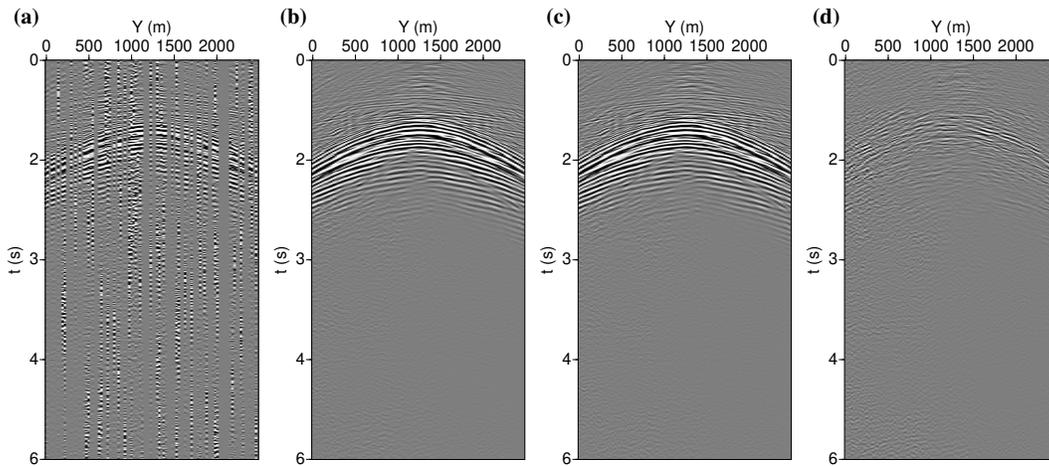


Figure 5.17: Zoomed area corresponding to the blue rectangle (center part) in Figure 5.15. (a) Observed decimated irregular-grid data. (b) Deblending and reconstruction via the I-MSSA method. (c) Results were obtained via the MSSA method. (d) Difference between (b) and (c).

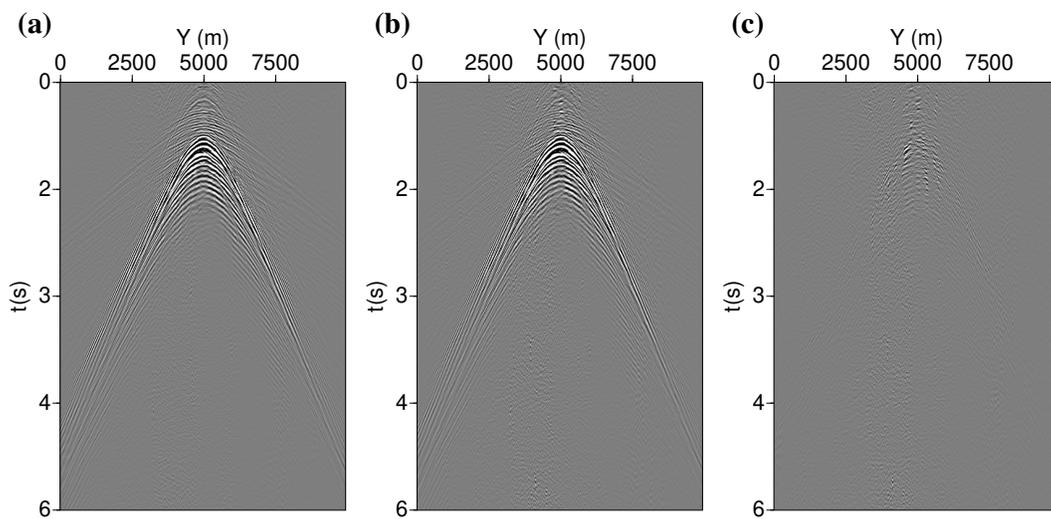


Figure 5.18: Deblending and reconstruction of field data with source distribution in Figure 5.14c. (a) Reconstruction and deblending result of observed data with I-MSSA method based on source coordinates distribution in Figure 5.14a. (b) Reconstruction and deblending result according to 20% decimated data with I-MSSA method based on coordinates distribution in Figure 5.14b. (c) Difference between (a) and (b).

introduces time shifts that account for significant reconstruction errors in our synthetic examples. The second case is inspired by the recently proposed reconstruction method called I-MSSA, which, in this chapter, was adapted to carry out source deblending and reconstruction simultaneously.

The I-MSSA method adopts Kaiser window tapered sinc interpolation operator and permits to honour true source coordinates. Computationally, there is no high extra cost in running deblending and reconstruction via MSSA or I-MSSA; hence, it is always advisable to adopt I-MSSA, which can cope with the real scenario of sources at quasi-regular or irregular coordinates.

CHAPTER 6

Conclusions

6.1 Conclusions and Summary

Simultaneous-source acquisition has become a standard method for collecting seismic data. Its popularity relies on its ability to decrease seismic acquisition turnaround time in high-density surveys. The simultaneous source approach allows a huge reduction in the time needed to acquire data which reduces costs as well as reduces exposure to work-related accidents. At the same time, the acquired higher density data can improve pre- and post-stack seismic data quality and increase the resolution of seismic images. Contrary to conventional seismic acquisitions, where only one single source is recorded and stored for each common source gather, simultaneous-source acquisition entails firing more than one source at a close interval and requires random firing times. Therefore, simultaneous-source data sorted in common-receiver gathers, common-channel gathers, common offset gathers, or common midpoint gathers, contain source interferences that manifest as erratic noise. This feature leads to various separation/deblending algorithms based on denoising and inversion methods. Meanwhile, for the simultaneous-source data processing, only adding another step, deblending, into the conventional seismic data processing workflow, no extra efforts are required in the stage of seismic data processing, which makes it attractive and increasingly popular.

This dissertation aims to introduce an inversion scheme with the projected gradient descent (PGD) method for simultaneous-source data separation. The PGD method is an optimization technique for solving constrained inverse problems. The denoising algorithms are implemented as the projection operator to iteratively minimize the objective function.

Chapter 2 provides a concise review that addresses solving linear inverse problems with regularization (quadratic and non-quadratic) methods and the PGD methods.

Chapter 3 defines the separation of simultaneous-source data as a coherence-pass constrained inverse problem. A robust sparse Radon transform is adopted as a coherence pass projection operator that cleans common receiver (or channel) gathers in a typical iterative deblending by inversion process. I also compare the inversion with the classical coherence pass non-robust Radon operator. The main conclusion of this chapter is that deblending by inversion methods requires a large number of iterations to obtain high-quality results if one does not aggressively remove incoherent noise during initial iterations. The latter can be achieved systematically by developing robust coherence pass operators like the one described in this chapter. In addition, it is also important to stress that the coherence pass robust and non-robust Radon operators have similar computational costs when implemented via the IRLS algorithm. Given that in both cases, I have considered a sparsity constraint to focus on the Radon coefficients, the iterative updates of the IRLS algorithm are roughly cost-wise equivalent for coherence pass non-robust and robust operators. I compared these two methods using numerically blended synthetic and real data examples with different blending factors and showed that a coherence pass non-robust Radon operator would only achieve high-quality results for acquisitions with a moderate blending factor.

Chapter 4 illustrates another inversion scheme with a rank-constrained optimization to iteratively solve the deblending problem. I adopted the PGD algorithm with the projection given by two flavours of the MSSA filter named non-robust (classical) MSSA and robust MSSA, operated at small overlapping windows by breaking down common-receiver or common offset gathers. The robust MSSA filter described in the chapter is reformulated as a robust optimization problem solved via a bifactored gradient descent (BFGD) algorithm. A Tukey's biweight loss function is adopted to robustize the error in the MSSA filter. For small overlapping windows, the traditional MSSA method needs the filter rank as an input parameter, which can vary from window to window. The latter has been a shortcoming for applying classical MSSA filtering to complex seismic data processing. The proposed robust MSSA filter is less sensitive to the rank selection, making it appealing for deblending applications that require windowing. Additionally, the robust MSSA projection effectively attenuates random source interferences during the initial iterations of the projected gradient descent method. We also show the benefit of adopting the robust MSSA filter with a relatively large window size to formulate the window-based projection required by the PGD algorithm. Comparing classical and robust MSSA filters, we also report an acceleration of the projected gradient descent method convergence when we adopt the robust MSSA filter. We use synthetic and real data examples to test our algorithms.

Chapter 5 proposed an inversion scheme for simultaneous deblending and source reconstruction for compressive simultaneous-source data by honouring actual source coordinates. I adopt PGD optimization with a projection operator given by a reduced-rank MSSA filter. In this chapter, I have examined two cases. In one case, I use an extraction operator that as-

signs true source position to grid points via nearest-neighbour assignment (*binning process*). This process introduces time shifts that account for significant reconstruction errors in our synthetic examples. The second case is inspired by the recently proposed reconstruction method called I-MSSA, which, in this chapter, was adapted to carry out source deblending and reconstruction simultaneously. In essence, the proposed I-MSSA method can handle simultaneous-source acquisition with an irregular-grid geometry based on compressive sensing design. With the I-MSSA algorithm, we can reduce acquisition time further by blending sources and using fewer sources, thereby improving the efficiency of field data acquisition. The I-MSSA method adopts Kaiser window tapered sinc interpolation operator and permits to honour of true source coordinates. Computationally, there is no high extra cost in running deblending and reconstruction via MSSA or I-MSSA; hence, it is always advisable to adopt I-MSSA, which can cope with the real scenario of sources at quasi-regular or irregular coordinates. We compare the proposed algorithm with traditional iterative rank reduction that adopts a regular source grid and ignores errors associated with allocating off-the-grid source coordinates to the desired output grid. Synthetic and field data examples show how the proposed method can deblend and reconstruct sources simultaneously.

Appendix A provides a comparative study between the IRLS algorithm (described in chapter 3) and the ADMM algorithm to solve robust sparse Radon transform for handling different kinds of noise, including blending noise, high-amplitude erratic ambient noise, and random noise. The IRLS is a classical and common-used method for solving nonquadratic ($\ell_1 - \ell_1$) optimization problems. However, its computational cost has always been a concern due to two loops requirement. The ADMM is a simple but powerful framework that is flexible in solving many high-dimensional optimization (quadratic or nonquadratic) problems. The procedure of ADMM normally decouples one original problem into several subproblems, which can be solved in an alternating minimization manner. Synthetic and field data examples show that the robust (ADMM and robust IRLS) methods outperform the non-robust (FISTA and non-robust IRLS) methods when existing erratic ambient noise. Also, the ADMM method shows a remarkable superiority in computational time-consuming, which could be a suitable replacement for the IRLS method when dealing with the $\ell_1 - \ell_1$ optimization problem.

In Appendix B, a faster and more computational-efficient algorithm, FMSSA, is proposed for low-rank estimation, acting as the projection operator in chapter 5 for separating and reconstructing compressive simultaneous-source data deployed on an irregular acquisition grid. The FMSSA method adopts a fast Fourier transform for Hankel matrix-vector products to avoid building explicit form Hankel matrices and using randomized QR decomposition to replace the SVD procedure. Also, the final anti-diagonal averaging of the Hankel matrix is computed efficiently via a convolution algorithm. Therefore, it is a suitable substitution for conventional MSSA when dealing with low-rank optimization problems. The synthetic and field data examples display a comparative study of the I-MSSA and the I-FMSSA methods

for simultaneous deblending and source reconstruction.

6.2 Future recommendations

The content of my dissertation relies on developing new algorithms/methods for processing simultaneous-source acquisition data. During my research, I also found some problems/limitations that can be considered in future research directions. First, to solve the Radon transform, whether adopting the IRLS algorithm or ADMM algorithm, I have only considered Radon operators in the time domain. Solving the robust sparse Radon transform in a mixed time-frequency domain could be an alternative option. The Radon operator can be written in an explicit matrix format in the frequency domain, and the optimization process can be solved in the time domain. In the mixed time-frequency domain, the products of adjoint Radon matrix (\mathcal{R}^*) and forward Radon matrix (\mathcal{R}) times a vector can be computed efficiently by exploiting the Toeplitz structure of frequency domain Radon operators (Beylkin, 1987; Kostov, 1990; Sacchi and Porsani, 1999). This could lead to more efficient Radon coherence-pass operators for the PGD deblending method.

Second, for the rank-constrained optimization problem, due to the projection being non-convex, even though I have tested some strategies for selecting the step length based on Bolduc et al. (2017), I believe interesting work can be conducted to improve the PGD algorithm. During my tests, the strategy of exponentially decreasing the step length is an effective way to achieve algorithm convergence. However, I believe a more sophisticated approach to selecting the step length λ can be tested to obtain a more efficient PGD method when the projection is non-convex.

Third, with the recent development of machine learning, deblending based on machine learning methods could be exploited. We have obtained encouraging deblending results via a Convolutional Neural Network (CNN) approach (Matharu et al., 2020). However, access to sufficient and diverse data for training the network seems to be still a significant obstacle. I suggest incorporating machine learning methods in conjunction with traditional signal processing methods, i.e., adopting a trained CNN as a projection operator in conjunction with a PGD method to avoid a purely example-based (end-to-end) machine learning approach. The approach can follow the steps initiated by Torres and Sacchi (2022) for seismic imaging problems.

Fourth, with the increasing debate focusing on reducing the environmental impact of fossil resources and transitioning away from a hydrocarbon-based economy, the simultaneous-source acquisition method could be beneficial for CO₂ Capture Utilization and Storage (CCUS) projects (Metz et al., 2005; Liu et al., 2017). In particular, geophysical monitoring

of CO₂ injected back into the earth is essential to secure its correct storage and containment in reservoirs. 3D reflection seismology in the time-lapse modality (4D seismic method) is necessary for CCUS projects. In this case, simultaneous-source methods could be used to obtain high-density surveys in a repeated fashion in an economical manner. Similarly, a low-cost 4D seismic method assisted by a simultaneous-source acquisition method could benefit the development of programs to monitor geothermal fields (Weemstra et al., 2016), which could be a possible energy substitute for the future economic transition to energy sources with lower greenhouse gas emission.

Bibliography

- Abma, R. and M. S. Foster, 2020, Simultaneous source seismic acquisition: Society of Exploration Geophysicists.
- Abma, R., D. Howe, M. Foster, I. Ahmed, M. Tanis, Q. Zhang, A. Arogunmati, and G. Alexander, 2015, Independent simultaneous source acquisition and processing: *Geophysics*, **80**, no. 6, WD37–WD44.
- Abma, R., T. Manning, M. Tanis, J. Yu, and M. Foster, 2010, High quality separation of simultaneous sources by sparse inversion: 72nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, B003.
- Akerberg, P., G. Hampson, J. Rickett, H. Martin, and J. Cole, 2008, Simultaneous source separation by sparse Radon transform: 78th Annual International Meeting, SEG, Expanded Abstracts, 2801–2805.
- Ander, M. E., M. A. Zumberge, T. Lautzenhiser, R. L. Parker, C. L. Aiken, M. R. Gorman, M. M. Nieto, A. P. R. Cooper, J. F. Ferguson, E. Fisher, et al., 1989, Test of Newton’s inverse-square law in the Greenland ice cap: *Physical Review Letters*, **62**, no. 9, 985.
- Arora, B., M. J. Unsworth, and G. Rawat, 2007, Deep resistivity structure of the northwest Indian Himalaya and its tectonic implications: *Geophysical Research Letters*, **34**, no. 4, L04307.
- Bahia, B. and M. Sacchi, 2019, Robust singular spectrum analysis via the bifactored gradient descent algorithm: 89th Annual International Meeting, SEG, Expanded Abstracts, 4640–4644.
- Bai, D., M. J. Unsworth, M. A. Meju, X. Ma, J. Teng, X. Kong, Y. Sun, J. Sun, L. Wang, C. Jiang, et al., 2010, Crustal deformation of the eastern Tibetan plateau revealed by magnetotelluric imaging: *Nature geoscience*, **3**, no. 5, 358–362.
- Beasley, C. J., 2008, A new look at marine simultaneous sources: *The Leading Edge*, **27**, no. 7, 914–917.
- Beasley, C. J., R. E. Chambers, and Z. Jiang, 1998, A new look at simultaneous sources: 68th Annual International Meeting, SEG, Extended Abstracts, 133–135.
- Beck, A. and M. Teboulle, 2009, A fast iterative shrinkage-thresholding algorithm for linear

- inverse problems: *SIAM journal on imaging sciences*, **2**, no. 1, 183–202.
- Belagiannis, V., C. Rupprecht, G. Carneiro, and N. Navab, 2015, Robust optimization for deep regression: *Proceedings of the IEEE international conference on computer vision*, 2830–2838.
- Berkhout, A. J., 2008, Changing the mindset in seismic data acquisition: *The Leading Edge*, **27**, no. 7, 924–938.
- Bertero, M., P. Boccacci, G. Desidera, and G. Vicidomini, 2009, Image deblurring with Poisson data: from cells to galaxies: *Inverse Problems*, **25**, no. 12, 123006.
- Bertsekas, D. P., 1997, *Nonlinear programming*: *Journal of the Operational Research Society*, **48**, no. 3, 334–334.
- Beylkin, G., 1987, Discrete Radon transform: *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **35**, no. 2, 162–172.
- Biondi, B. L., 2006, *3D seismic imaging*: *Society of Exploration Geophysicists*.
- Blakely, R. J., 1996, *Potential theory in gravity and magnetic applications*: *Cambridge university press*.
- Blumensath, T. and M. E. Davies, 2008, Iterative thresholding for sparse approximations: *Journal of Fourier analysis and Applications*, **14**, no. 5, 629–654.
- Bolduc, E., G. C. Knee, E. M. Gauger, and J. Leach, 2017, Projected gradient descent algorithms for quantum state tomography: *npj Quantum Information*, **3**, no. 44, 1–9.
- Boyd, S., N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., 2011, Distributed optimization and statistical learning via the alternating direction method of multipliers: *Foundations and Trends® in Machine learning*, **3**, no. 1, 1–122.
- Brown, A. R., 2011, *Interpretation of Three-Dimensional Seismic Data*: *American Association of Petroleum Geologists*.
- Cadzow, J. A., 1988, Signal enhancement - a composite property mapping algorithm: *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **36**, no. 1, 49–62.
- Cagniard, L., 1953, Basic theory of the magneto-telluric method of geophysical prospecting: *Geophysics*, **18**, no. 3, 605–635.
- Cai, T. T. and L. Wang, 2011, Orthogonal matching pursuit for sparse signal recovery with noise: *IEEE Transactions on Information Theory*, **57**, no. 7, 4680–4688.
- Cantwell, T. and T. Madden, 1960, Preliminary report on crustal magnetotelluric measurements: *Journal of Geophysical Research*, **65**, no. 12, 4202–4205.
- Carozzi, F. and M. D. Sacchi, 2021, Interpolated multichannel singular spectrum analysis: A reconstruction method that honors true trace coordinates: *Geophysics*, **86**, no. 1, V55–V70.
- Chen, K. and M. D. Sacchi, 2014, Robust reduced-rank filtering for erratic seismic noise attenuation: *Geophysics*, **80**, no. 1, V1–V11.
- Chen, Y., S. Fomel, and J. Hu, 2014, Iterative deblending of simultaneous-source seismic

- data using seislet-domain shaping regularization: *Geophysics*, **79**, no. 5, V179–V189.
- Cheng, J., 2017, Gradient projection methods with applications to simultaneous source seismic data processing: Ph.D. thesis, University of Alberta.
- Cheng, J., N. Kazemi, and M. Sacchi, 2016, Least-squares migration via a gradient projection method - application to seismic data deblending: 78th EAGE Conference and Exhibition, 1–5.
- Cheng, J., M. Sacchi, and J. Gao, 2019, Computational efficient multidimensional singular spectrum analysis for prestack seismic data reconstruction: *Geophysics*, **84**, no. 2, V111–V119.
- Cheng, J. and M. D. Sacchi, 2014, An analysis of the distribution of time delays on simultaneous source separation: 84th Annual International Meeting, SEG, Expanded Abstracts, 275–279.
- 2015, Separation and reconstruction of simultaneous source data via iterative rank reduction: *Geophysics*, **80**, no. 4, V57–V66.
- 2016, Fast dual-domain reduced-rank algorithm for 3D deblending via randomized QR decomposition: *Geophysics*, **81**, no. 1, V89–V101.
- Claerbout, J., 1992, *Earth soundings analysis: Processing versus inversion*: Blackwell Scientific Publications, London.
- Daubechies, I., M. Defrise, and C. De Mol, 2004, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, **57**, no. 11, 1413–1457.
- Daubechies, I., R. DeVore, M. Fornasier, and C. S. Gunturk, 2009, Iteratively reweighted least squares minimization for sparse recovery: *Communications on Pure and Applied Mathematics*, **63**, 1–38.
- Dhelie, P. E., V. Danielsen, J. E. Lie, E. Davenne, F. Andersson, K. Eggenberger, and J. Robertsson, 2019, Source-over-cable marine acquisition with six sources in simultaneous mode: 89th Annual International Meeting, SEG, Expanded Abstracts, 92–96.
- Donoho, D. L., 2006, Compressed sensing: *IEEE Transactions on information theory*, **52**, no. 4, 1289–1306.
- Eckart, C. and G. Young, 1936, The approximation of one matrix by another of lower rank: *Psychometrika*, **1**, no. 3, 211–218.
- Fazel, M., 2002, *Matrix rank minimization with applications*: Ph.D. thesis, Stanford University.
- Fomel, S., 2001, *Three-dimensional seismic data regularization*: Ph.D. thesis, Stanford University.
- Fowler, C. M. R., C. M. R. Fowler, and M. Fowler, 1990, *The solid Earth: an introduction to global geophysics*: Cambridge University Press.

- Ghil, M., M. Allen, M. Dettinger, K. Ide, D. Kondrashov, M. Mann, A. Robertson, A. Saunders, Y. Tian, F. Varadi, and P. Yiou, 2002, Advance spectral methods for climatic time series: *Reviews of Geophysics*, **40**, no. 1, 1–41.
- Golub, G. H. and C. F. Van Loan, 1996, *Matrix computations: The Johns Hopkins University Press*, third edition.
- Gray, S. H., J. Etgen, J. Dellinger, and D. Whitmore, 2001, Seismic migration problems and solutions: *Geophysics*, **66**, no. 5, 1622–1640.
- Halko, N., P.-G. Martinsson, and J. A. Tropp, 2011, Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions: *SIAM review*, **53**, no. 2, 217–288.
- Hampson, G., J. Stefani, and F. Herkenhoff, 2008, Acquisition using simultaneous sources: *The Leading Edge*, **27**, no. 7, 918–923.
- Hennenfent, G. and F. J. Herrmann, 2008, Simply denoise: Wavefield reconstruction via jittered undersampling: *Geophysics*, **73**, no. 3, V19–V28.
- Herrmann, F. J., 2010, Randomized sampling and sparsity: Getting more information from fewer samples: *Geophysics*, **75**, no. 6, WB173–WB187.
- Hestenes, M. R., E. Stiefel, et al., 1952, Methods of conjugate gradients for solving linear systems: *Journal of Research of the National Bureau of Standards*, **49**, no. 1.
- Holland, P. W. and R. E. Welsch, 1977, Robust regression using iteratively reweighted least-squares: *Communications in Statistics - Theory and Methods*, **6**, no. 9, 813–827.
- Howe, D., M. Foster, T. Allen, B. Taylor, and I. Jack, 2008, Independent simultaneous sweeping -a method to increase the productivity of land seismic crews: 78th Annual International Meeting, SEG, Expanded Abstracts, 2826–2830.
- Huo, S., Y. Luo, and P. G. Kelamis, 2012, Simultaneous sources separation via multidirectional vector-median filtering: *Geophysics*, **77**, no. 4, V123–V131.
- Ibrahim, A. and M. D. Sacchi, 2013, Simultaneous source separation using a robust Radon transform: *Geophysics*, **79**, no. 1, V1–V11.
- Ibrahim, A. and D. Trad, 2020, Inversion-based deblending using migration operators: *Geophysical Prospecting*, **68**, no. 8, 2459–2470.
- Iusem, A. N., 2003, On the convergence properties of the projected gradient method for convex optimization: *Computational and Applied Mathematics*, **22**, no. 1, 35–52.
- Jeong, W., C. Tsingas, and M. S. Almubarak, 2020, A numerical study on deblending of land simultaneous shooting acquisition data via rank-reduction filtering and signal enhancement applications: *Geophysical Prospecting*, **68**, no. 8, 1742–1757.
- Ji, J., 2012, Robust inversion using biweight norm and its application to seismic inversion: *Exploration Geophysics*, **43**, no. 2, 70–76.
- Jiang, T., B. Gong, F. Qiao, Y. Jiang, A. Chen, D. Hren, and Z. Meng, 2017, Compressive seismic reconstruction with extended POCS for arbitrary irregular acquisition: 87th

- Annual International Meeting, SEG, Extended Abstracts, 4272–4277.
- Kim, Y., I. Gruzinov, M. Guo, and S. Sen, 2009, Source separation of simultaneous source OBC data: 79th Annual International Meeting, SEG, Expanded Abstracts, 51–55.
- Kostov, C., 1990, Toeplitz structure in slant-stack inversion: 60th Annual International Meeting, SEG, Expanded Abstracts, 1618–1621.
- Krohn, C. and R. Neelamani, 2008, Simultaneous sourcing without compromise: 70th EAGE Conference and Exhibition incorporating SPE EUROPEC 2008, cp–40.
- Krohn, C. E. and M. L. Johnson, 2006, HFVSTM: Enhanced data quality through technology integration: *Geophysics*, **71**, no. 2, E13–E23.
- Li, C., C. C. Mosher, and S. T. Kaplan, 2012, Interpolated compressive sensing for seismic data reconstruction: 82nd Annual International Meeting, SEG, Expanded Abstracts, 1–6.
- Li, C., C. C. Mosher, L. C. Morley, Y. Ji, and J. D. Brewer, 2013, Joint source deblending and reconstruction for seismic data: 83rd Annual International Meeting, SEG, Expanded Abstracts, 82–87.
- Li, P., J. Song, S. Zhang, W. Wang, P. Sun, and Y. Ma, 2019, Deblending by sparse inversion: Case study on land data from Oman: 89th Annual International Meeting, SEG, Expanded Abstracts, 97–101.
- Liberty, E., F. Woolfe, P.-G. Martinsson, V. Rokhlin, and M. Tygert, 2007, Randomized algorithms for the low-rank approximation of matrices: *Proceedings of the National Academy of Sciences*, **104**, no. 51, 20167–20172.
- Lin, R., B. Bahia, and M. D. Sacchi, 2021, Iterative deblending of simultaneous-source seismic data via a robust singular spectrum analysis filter: *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 1–10.
- Lin, R. and M. D. Sacchi, 2020a, Separation of simultaneous sources acquired with a high blending factor via coherence pass robust Radon operators: *Geophysics*, **85**, no. 3, V269–V282.
- 2020b, Separation of simultaneous sources via coherence pass robust Radon operators: 90th Annual International Meeting, SEG, Extended Abstracts, 2805–2809.
- Liu, B. and M. D. Sacchi, 2004, Minimum weighted norm interpolation of seismic records: *Geophysics*, **69**, no. 6, 1560–1568.
- Liu, H., P. Were, Q. Li, Y. Gou, and Z. Hou, 2017, Worldwide status of CCUS technologies and their development and challenges in China: *Geofluids*, **2017**.
- Lizarralde, D. and S. Swift, 1999, Smooth inversion of VSP travelttime data: *Geophysics*, **64**, no. 3, 659–661.
- Ma, S., D. Goldfarb, and L. Chen, 2011, Fixed point and Bregman iterative methods for matrix rank minimization: *Mathematical Programming*, **128**, no. 1-2, 321–353.
- Mahdad, A., P. Doulgeris, and G. Blacchiere, 2011, Separation of blended data by iterative estimation and subtraction of blending interference noise: *Geophysics*, **76**, no. 3, Q9–Q17.

- Maraschini, M., R. Dyer, K. Stevens, and D. Bird, 2012a, Source separation by iterative rank reduction-theory and applications: 74th EAGE Conference and Exhibition incorporating EUROPEC, cp-293.
- Maraschini, M., R. Dyer, K. Stevens, D. Bird, and S. King, 2012b, An iterative SVD method for deblending: theory and examples: 82nd Annual International Meeting, SEG, Expanded Abstracts, 1-5.
- Matharu, G., W. Gao, R. Lin, Y. Guo, M. Park, and M. D. Sacchi, 2020, Simultaneous source deblending using a deep residual network: SEG 2019 Workshop: Mathematical Geophysics: Traditional vs Learning, Beijing, China, 13-16.
- Menke, W., 1989, Geophysical data analysis: Discrete inverse theory: Academic Press.
- 2018, Geophysical data analysis: Discrete inverse theory: Academic press.
- Metz, B., O. Davidson, H. De Coninck, M. Loos, and L. Meyer, 2005, IPCC special report on carbon dioxide capture and storage: Cambridge: Cambridge University Press.
- Moore, I., 2010, Simultaneous sources - Processing and applications: 72nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, B001.
- Moore, I., B. Dragoset, T. Ommundsen, D. Wilson, C. Ward, and D. Eke, 2008, Simultaneous source separation using dithered sources: 78th Annual International Meeting, SEG, Expanded Abstracts, 2806-2810.
- Mosher, C., S. Kaplan, and F. Janiszewski, 2012, Non-uniform optimal sampling for seismic survey design: 74th EAGE Conference and Exhibition incorporating EUROPEC, cp-293.
- Mosher, C., C. Li, L. Morley, Y. Ji, F. Janiszewski, R. Olson, and J. Brewer, 2014, Increasing the efficiency of seismic data acquisition via compressive sensing: *The Leading Edge*, **33**, no. 4, 386-391.
- Mosher, C. C., C. Li, F. D. Janiszewski, L. S. Williams, T. C. Carey, and Y. Ji, 2017, Operational deployment of compressive sensing systems for seismic data acquisition: *The Leading Edge*, **36**, no. 8, 661-669.
- Munoz, G., 2014, Exploring for geothermal resources with electromagnetic methods: *Surveys in geophysics*, **35**, no. 1, 101-122.
- Naghizadeh, M. and M. D. Sacchi, 2010, On sampling functions and Fourier reconstruction methods: *Geophysics*, **75**, no. 6, WB137-WB151.
- Oldenburg, D. W., T. Scheuer, and S. Levy, 1983, Recovery of the acoustic impedance from reflection seismograms: *Geophysics*, **48**, no. 10, 1318-1337.
- Oppenheim, A. V., R. W. Schaffer, and J. R. Buck, 1999, *Discrete-time signal processing*: Prentice-hall Englewood Cliffs, second edition.
- Oropeza, V., 2010, The singular spectrum analysis method and its application to seismic data denoising and reconstruction: M.Sc. thesis, University of Alberta.
- Oropeza, V. and M. Sacchi, 2011, Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis: *Geophysics*, **76**, no. 3, V25-V32.

- Park, D., A. Kyrillidis, C. Caramanis, and S. Sanghavi, 2018, Finding low-rank solutions via nonconvex matrix factorization, efficiently and provably: *SIAM Journal on Imaging Sciences*, **11**, no. 4, 2165–2204.
- Peng, C., B. Liu, A. Khalil, and G. Poole, 2013, Deblending of simulated simultaneous sources using an iterative approach - An experiment with variable-depth streamer data: 83rd Annual International Meeting, SEG, Expanded Abstracts, 4278–4282.
- Peters, B., B. R. Smithyman, and F. J. Herrmann, 2019, Projection methods and applications for seismic nonlinear inverse problems with multiple constraints: *Geophysics*, **84**, no. 2, R251–R269.
- Piana, M. and M. Bertero, 1997, Projected landweber method and preconditioning: *Inverse problems*, **13**, no. 2, 441.
- Price, A., 1962, The theory of magnetotelluric methods when the source field is considered: *Journal of Geophysical Research*, **67**, no. 5, 1907–1918.
- Reynolds, J. M., 2011, *An introduction to applied and environmental geophysics*: John Wiley & Sons.
- Rozemond, H. J., 1996, Slip-sweep acquisition: 66th Annual International Meeting, SEG, Expanded Abstracts, 64–67.
- Sacchi, M. D., 1997, Reweighting strategies in seismic deconvolution: *Geophysical Journal International*, **129**, no. 3, 651–656.
- Sacchi, M. D. and M. Porsani, 1999, Fast high resolution parabolic Radon transform: 69th Annual International Meeting, SEG, Expanded Abstracts, 1477–1480.
- Sacchi, M. D., T. J. Ulrych, and C. J. Walker, 1998, Interpolation and extrapolation using a high-resolution discrete Fourier transform: *IEEE Transactions on Signal Processing*, **46**, no. 1, 31–38.
- Sava, P. and S. J. Hill, 2009, Overview and classification of wavefield seismic imaging methods: *The Leading Edge*, **28**, no. 2, 170–183.
- Scales, J. A., 1987, Tomographic inversion via the conjugate gradient method: *Geophysics*, **52**, no. 2, 179–185.
- Song, J., P. Li, Z. Qian, M. Zhang, P. Sun, W. Wang, and Y. Ma, 2019, Simultaneous vibroseis data separation through sparse inversion: *The Leading Edge*, **38**, no. 8, 625–629.
- Spitz, S., G. Hampson, and A. Pica, 2008, Simultaneous source separation: A prediction-subtraction approach: 78th Annual International Meeting, SEG, Expanded Abstracts, 2811–2815.
- Tarantola, A., 2005, *Inverse problem theory and methods for model parameter estimation*: Society for industrial and applied mathematics.
- Telford, W. M., W. Telford, L. Geldart, and R. E. Sheriff, 1990, *Applied geophysics*: Cambridge university press.

- Thorson, J. R. and J. F. Claerbout, 1985, Velocity-stack and slant-stack stochastic inversion: *Geophysics*, **50**, no. 12, 2727–2741.
- Tibshirani, R., 1996, Regression shrinkage and selection via the Lasso: *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**, no. 1, 267–288.
- Tikhonov, A., 1950, On determining electric characteristics of the deep layers of the Earth's crust: *Proceedings of Academy of Sciences (USSR) Doklady*, **83**, no. 2, 295–297.
- Tikhonov, A. N., 1943, On the stability of inverse problems: *Dokl. Akad. Nauk SSSR*, 195–198.
- Torres, K. and M. Sacchi, 2022, Least-squares reverse time migration via deep learning-based updating operators: *Geophysics*, **87**, no. 6, 1–80.
- Trad, D., 2009, Five-dimensional interpolation: Recovering from acquisition constraints: *Geophysics*, **74**, no. 6, V123–V132.
- Trad, D., M. D. Sacchi, and T. J. Ulrych, 2001, A hybrid linear-hyperbolic Radon transform: *Journal of Seismic Exploration*, **9**, no. 4, 303–318.
- Trad, D., T. Ulrych, and M. Sacchi, 2003, Latest views of the sparse Radon transform: *Geophysics*, **68**, no. 1, 386–399.
- Trickett, S., 2008, F-xy Cadzow noise suppression: 78th Annual International Meeting, SEG, Expanded Abstracts, 2586–2590.
- Trickett, S., L. Burroughs, and A. Milton, 2012, Robust rank-reduction filtering for erratic noise: 82nd Annual International Meeting, SEG, Expanded Abstracts, 1–5.
- Ulrych, T. J. and M. D. Sacchi, 2005, *Information-based inversion and processing with applications*: Elsevier Science, 1st edition.
- Unsworth, M., 2005, New developments in conventional hydrocarbon exploration with electromagnetic methods: *CSEG Recorder*, **30**, no. 4, 34–38.
- Vautard, R. and M. Ghil, 1989, Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series: *Physica D: Nonlinear Phenomena*, **35**, no. 3, 395 – 424.
- Vautard, R., P. Yiou, and M. Ghil, 1992, Singular-spectrum analysis: A toolkit for short, noisy chaotic signals: *Physica D*, **58**, 95–126.
- Vermeer, G., 1990, Seismic wavefield sampling: Society of Exploration Geophysicists.
- Vermeer, G. J., 2002, 3-D seismic survey design: Society of Exploration Geophysicists.
- Wang, B. and J. Geng, 2019, Efficient deblending in the PFK domain based on compressive sensing: *IEEE Transactions on Geoscience and Remote Sensing*, **58**, no. 2, 995–1003.
- Wang, B., J. Geng, and J. Song, 2022, Off-the-grid blended data deblending via a sinc-based interpolator: SEG 2021 Workshop: 4th International Workshop on Mathematical Geophysics: Traditional & Learning, 5–8.
- Wang, B., Y. Zhang, W. Lu, and J. Geng, 2019, A robust and efficient sparse time-invariant Radon transform in the mixed time–frequency domain: *IEEE Transactions on Geoscience*

- and Remote Sensing, **57**, no. 10, 7558–7566.
- Wang, H., J.-F. Cai, T. Wang, and K. Wei, 2021, Fast Cadzow’s algorithm and a gradient variant: *Journal of Scientific Computing*, **88**, no. 2, 1–21.
- Wason, H., F. J. Herrmann, and T. T. Lin, 2011, Sparsity-promoting recovery from simultaneous data: A compressive sensing approach: 81st Annual International Meeting, SEG, Expanded Abstracts, 6–10.
- Weemstra, C., A. Obermann, A. Verdel, B. Paap, H. Blanck, E. Gunason, G. Hersir, P. Jousset, and A. Sigursson, 2016, Time-lapse seismic imaging of the reykjanes geothermal reservoir: Presented at the Proc. European Geothermal Congress, Strasbourg, France.
- Wen, F., L. Pei, Y. Yang, W. Yu, and P. Liu, 2017, Efficient and robust recovery of sparse signal and image using generalized nonconvex regularization: *IEEE Transactions on Computational Imaging*, **3**, no. 4, 566–579.
- Wen, F., R. Ying, P. Liu, and R. C. Qiu, 2019, Robust PCA using generalized nonconvex regularization: *IEEE Transactions on Circuits and Systems for Video Technology*, **30**, no. 6, 1497–1510.
- Yilmaz, Ö., 2001, *Seismic data analysis: Processing, inversion, and interpretation of seismic data*: Society of exploration geophysicists.
- 2021, *Land seismic case studies for near-surface modeling and subsurface imaging*: Society of Exploration Geophysicists.
- Zhao, G., M. J. Unsworth, Y. Zhan, L. Wang, X. Chen, A. G. Jones, J. Tang, Q. Xiao, J. Wang, J. Cai, et al., 2012, Crustal structure and rheology of the Longmenshan and Wenchuan Mw 7.9 earthquake epicentral area from magnetotelluric data: *Geology*, **40**, no. 12, 1139–1142.
- Zwartjes, P. and A. Gisolf, 2007, Fourier reconstruction with sparse inversion: *Geophysical Prospecting*, **55**, no. 2, 199–221.

APPENDIX A

Deblending via ADMM and IRLS ¹

A.1 Introduction

In the category of inversion-based methods for simultaneous-source separation, Chapter 3 provides a new deblending strategy by adopting a robust sparse Radon transform to define a coherence pass projection operator in conjunction with the steepest descent method to separate the simultaneous-source data iteratively. The robust sparse Radon transform is computed with the IRLS algorithm by transforming the nonquadratic $(\ell_1 - \ell_1)$ problem into a sequence of quadratic $(\ell_2 - \ell_2)$ minimization problems by introducing weighting matrices for data misfit term and model term. However, even though the IRLS is a classical and common-used method for solving nonquadratic $(\ell_1 - \ell_1)$ optimization problems, its computational cost has always been a concern due to two loops requirement. In this appendix, a more efficient and flexible algorithm named Alternating Direction Method of Multipliers method (ADMM) (Boyd et al., 2011) is introduced for efficient computation of the nonquadratic robust sparse Radon optimization problem. The ADMM is a flexible and powerful framework for solving many optimization problems by decoupling one original problem into several subproblems, translated in an alternating minimization manner (Wen et al., 2017), to achieve an efficient and faster algorithm.

This appendix solves the robust and sparse Radon transform with two different algorithms and is an extended part of Chapter 3 and follows the subsequent structure. First, I briefly review the definition of Radon transform. I continue with a description of the IRLS and ADMM methods for solving nonquadratic $(\ell_1 - \ell_1)$ optimization problems. Finally, I com-

¹A version of this appendix is published in Lin, R., and M. D. Sacchi, 2020, Deblending via ADMM and IRLS: A comparative study, GeoConvention 2020.

pare the IRLS method with the ADMM method for blending interferences and random and high-amplitude erratic noise attenuation via synthetic and field data examples.

A.2 Theory

A.2.1 Review of the Radon transform

We denote the common receiver gather data as $d(t, h)$ or in matrix form \mathbf{d} , and the variable h and t represent time and offset, respectively. The forward and its adjoint Radon transform can be expressed in the following equations for the discrete form:

$$\tilde{m}(q, \xi) = \sum_h d(t = \tilde{\phi}(\tau, h, \xi), h), \quad (\text{A.1})$$

$$d(t, h) = \sum_{\xi} m(\tau = \phi(t, h, \xi), \xi), \quad (\text{A.2})$$

where $\tilde{m}(q, \xi)$ are Radon coefficients that one can obtain by adjoint Radon operator. In operator form, the adjoint Radon transform and forward Radon transform can be expressed as the following

$$\tilde{\mathbf{m}} = \mathcal{R}^* \mathbf{d}, \quad (\text{A.3})$$

$$\mathbf{d} = \mathcal{R} \mathbf{m}. \quad (\text{A.4})$$

Unlike the Fourier or wavelet transform, the Radon transform is not orthogonal. We cannot obtain the Radon coefficients from the adjoint Radon transform directly. Normally we need to adopt an inversion method (Thorson and Claerbout, 1985) to recover the Radon coefficient by minimizing the following cost function:

$$J = \|\mathbf{d} - \mathcal{R} \mathbf{m}\|_2^2 + \mu \|\mathbf{m}\|_2^2. \quad (\text{A.5})$$

The first term on the right side is the misfit term, and the second is the model (regularization) term to achieve a stable and unique solution. Equation A.5 can be easily solved by the damped least square method or conjugate gradient method (Scales, 1987), and μ denotes the trade-off parameter. While solved via the $\ell_2 - \ell_2$ problem typically obtain low-resolution Radon coefficients, which cannot recover the data exactly (Trad et al., 2003). We can adopt the sparse Radon transform by using ℓ_1 norm for the model term to obtain the high-resolution Radon coefficients with the following cost function:

$$J = \|\mathbf{d} - \mathcal{R} \mathbf{m}\|_2^2 + \mu \|\mathbf{m}\|_1. \quad (\text{A.6})$$

The cost function in equation A.6 ($\ell_2 - \ell_1$ problem) can be solved by IRLS method (Ibrahim and Sacchi, 2013) (see equation A.9 for details) and FISTA method (Beck and Teboulle, 2009) to obtain the non-robust sparse Radon transform for denoising.

A.2.2 IRLS method for robust and sparse Radon transform

Adopting sparse Radon transform is useful for random noise attenuation. The sparse Radon transform can be solved by Iteratively Reweighted Least-Squares (IRLS) method by adding a weighting matrix into the model term. The weighting matrix for the model term can be expressed as:

$$\|\mathbf{m}\|_1^1 = \sum_i |m_i| |m_i|^{-1} |m_i| = \|\mathbf{W}_m \mathbf{m}\|_2^2, \quad (\text{A.7})$$

where \mathbf{W}_m is a diagonal matrix with the diagonal elements $[\mathbf{W}_m]_{ii} = |m_i|^{-1/2}$. By adding a weighting matrix, the ℓ_1 norm for the model term can be transformed into the ℓ_2 norm; we still can adopt the conjugate gradient or least square method to solve it. Then, the minimization of the problem A.6 produces the following equation:

$$J = \|\mathbf{d} - \mathcal{R}\mathbf{m}\|_2^2 + \mu \|\mathbf{W}_m \mathbf{m}\|_2^2. \quad (\text{A.8})$$

A simple preconditioning modification by setting $\mathbf{u} = \mathbf{W}_m \mathbf{m}$ is used to obtain its standard form

$$J = \|\mathbf{d} - \mathcal{R}(\mathbf{W}_m)^{-1} \mathbf{u}\|_2^2 + \mu \|\mathbf{u}\|_2^2. \quad (\text{A.9})$$

When the data are contaminated by erratic noise, i.e., blending noise and high amplitude erratic noise, the sparse Radon transform will not work effectively. In this case, we can utilize the robust and sparse Radon transform for noise attenuation. Besides setting ℓ_1 norm for the model term, we also set ℓ_1 norm instead of ℓ_2 norm for the misfit term with the following equation

$$J = \|\mathbf{d} - \mathcal{R}\mathbf{m}\|_1^1 + \mu \|\mathbf{m}\|_1^1. \quad (\text{A.10})$$

To solve the robust and sparse Radon transform, I first adopt the IRLS method to solve the $\ell_1 - \ell_1$ problem by adding two weighting matrices to transform the original problem into an $\ell_2 - \ell_2$ problem (Ibrahim and Sacchi, 2013). The weighting matrix for the misfit term is similar to the associated weighting matrix for model norm

$$\|\mathbf{d} - \mathcal{R}\mathbf{m}\|_1^1 = \|\mathbf{r}\|_1^1 = \sum_i |r_i| |r_i|^{-1} |r_i| = \|\mathbf{W}_r \mathbf{r}\|_2^2, \quad (\text{A.11})$$

where $\mathbf{r} = \mathbf{d} - \mathcal{R}\mathbf{m}$ is the residual vector, and \mathbf{W}_r is a diagonal matrix and $[\mathbf{W}_r]_{ii} = |r_i|^{-1/2}$. The weighting matrix cannot be computed for $r_i = 0$ for ℓ_1 norm. Thus, we normally adopt

$[\mathbf{W}_r]_{ii} = |r_i|^{-1/2} + \epsilon$, where ϵ is small number that avoids dividing $r_i = 0$, which is the same for \mathbf{W}_m .

Therefore, for the robust and sparse Radon transform, we can turn the nonquadratic ($\ell_1 - \ell_1$) problem into a sequence of quadratic minimization problems ($\ell_2 - \ell_2$) by adding weighting matrices \mathbf{W}_r and \mathbf{W}_m to obtain the following equation:

$$\begin{aligned} J &= \|\mathbf{W}_r \mathbf{r}\|_2^2 + \mu \|\mathbf{W}_m \mathbf{m}\|_2^2, \\ &= \|\mathbf{W}_r (\mathbf{d} - \mathcal{R} \mathbf{m})\|_2^2 + \mu \|\mathbf{W}_m \mathbf{m}\|_2^2. \end{aligned} \quad (\text{A.12})$$

Similarly, a simple preconditioning modification is used to express J in its standard form

$$J = \left\| \mathbf{W}_r \left(\mathbf{d} - \mathcal{R} (\mathbf{W}_m)^{-1} \mathbf{u} \right) \right\|_2^2 + \mu \|\mathbf{u}\|_2^2. \quad (\text{A.13})$$

Equation A.13 can be minimized by the conjugate gradients method (Scales, 1987) followed by updates of the weighting matrices \mathbf{W}_r and \mathbf{W}_m (Ibrahim and Sacchi, 2013).

We adopt a method similar to the one described by Trad et al. (2003); however, we do not ignore the trade-off parameter μ . Our experience with various numerical tests indicates that we can obtain a more accurate solution by adding the trade-off parameter. The solution solved by the conjugate gradient method is the same as the damped least squares solution, even though it takes more time to run the IRLS method. In essence, I have an internal iteration to minimize the cost function via the method of conjugate gradients and an external iteration to update the weighting matrices (IRLS algorithm). The algorithm is stopped when the misfit change between iterations is less than a defined tolerance value (e.g., tolerance = 10^{-6}) or when it reaches a maximum number of iterations (Ibrahim and Sacchi, 2013). Last, I mention that the IRLS (non-robust) algorithm denotes adopting equation A.9 for solving non-robust sparse Radon transform, and the IRLS (robust) algorithm denotes adopting equation A.13 for solving robust sparse Radon transform.

A.2.3 ADMM method for robust and sparse Radon transform

The ADMM is a simple but powerful framework (Boyd et al., 2011), which is flexible in solving many high-dimensional optimization problems by separating coupled components in the cost function by including auxiliary constraint variables. The ADMM method normally decouples one original problem into several subproblems, which can be solved in an alternating minimization manner (Wen et al., 2017).

The generalized scaled form of ADMM can be summarized as the following:

$$\begin{aligned} & \text{minimize} && f(x) + g(z) \\ & \text{subject to} && Ax + Bz = c \end{aligned} \quad (\text{A.14})$$

The original problem A.14 can be separated into the following iterations

$$\begin{aligned} x^{k+1} &:= \underset{x}{\operatorname{argmin}} \left(f(x) + (\rho/2) \|Ax + Bz^k - c + u^k\|_2^2 \right) \\ z^{k+1} &:= \underset{z}{\operatorname{argmin}} \left(g(z) + (\rho/2) \|Ax^{k+1} + Bz - c + u^k\|_2^2 \right) \\ u^{k+1} &:= u^k + Ax^{k+1} + Bz^{k+1} - c \end{aligned} \quad (\text{A.15})$$

where u is the Lagrangian multiplier, $\rho > 0$ is a penalty parameter or a balancing parameter (Wang et al., 2019).

Similarly, for the robust and sparse Radon transform, we can minimize the following cost function

$$\text{minimize } J = \|\mathcal{R}\mathbf{m} - \mathbf{d}\|_1 + \mu\|\mathbf{m}\|_1. \quad (\text{A.16})$$

Then, the scaled form of ADMM for the robust and sparse Radon transform can be written as

$$\begin{aligned} & \text{minimize} && \|\mathbf{r}\|_1 + \mu\|\mathbf{m}\|_1 \\ & \text{subject to} && \mathbf{r} = \mathcal{R}\mathbf{m} - \mathbf{d} \end{aligned} \quad (\text{A.17})$$

The ADMM consists of the following three steps

$$\mathbf{r}^{k+1} = \underset{\mathbf{r}}{\operatorname{argmin}} \left\{ \|\mathbf{r}\|_1 + \frac{\rho_1}{2} \|\mathbf{r} - \mathcal{R}\mathbf{m}^k + \mathbf{d} + \mathbf{u}_1^k\|_2^2 \right\} \quad (\text{A.18})$$

$$\mathbf{m}^{k+1} = \underset{\mathbf{m}}{\operatorname{argmin}} \left\{ \mu\|\mathbf{m}\|_1 + \frac{\rho_1}{2} \|\mathbf{r}^{k+1} - \mathcal{R}\mathbf{m} + \mathbf{d} + \mathbf{u}_1^k\|_2^2 \right\} \quad (\text{A.19})$$

$$\mathbf{u}_1^{k+1} = \mathbf{u}_1^k + [\mathbf{r}^{k+1} - \mathcal{R}\mathbf{m}^{k+1} + \mathbf{d}] \quad (\text{A.20})$$

where \mathbf{u} is the vector of Lagrange multipliers, μ is the trade-off parameter for balancing the misfit and model term, and $\rho_1 > 0$ is a penalty parameter. Equation A.18 can be solved by the proximity operator

$$\mathbf{r}^{k+1} = \operatorname{prox}_{1/\rho_1} \{ \mathbf{d} - \mathcal{R}\mathbf{m}^k + \mathbf{u}_1^k \}, \quad (\text{A.21})$$

where

$$\operatorname{prox}_\tau \{y\} = \underset{x}{\operatorname{argmin}} \left\{ \frac{1}{2}(x - y)^2 + \tau|x| \right\}. \quad (\text{A.22})$$

represents the proximity operator, which is applied component-wise. The analytical solution

of proximity operator A.22 is

$$\text{prox}_\tau\{y\} = \text{sign}(y) \max(|y| - \tau, 0) \quad (\text{A.23})$$

which is equivalent to the soft-thresholding operator (Blumensath and Davies, 2008).

The \mathbf{m} -update step A.19 can be reformatted into

$$\mathbf{m}^{k+1} = \arg \min_{\mathbf{m}} \left\{ \|\mathbf{b}^k - \mathcal{R}\mathbf{m}\|_2^2 + \frac{2\mu}{\rho_1} \|\mathbf{m}\|_1 \right\}, \quad (\text{A.24})$$

where $\mathbf{b}^k = \mathbf{r}^{k+1} + \mathbf{d} + \mathbf{u}_1^k$. Equation A.24 is the classical $\ell_2 - \ell_1$ problem, which can be easily solved by the least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996), fast iterative shrinkage thresholding algorithm (FISTA) (Beck and Teboulle, 2009), or orthogonal matching pursuit (OMP) (Cai and Wang, 2011), etc. In my case, \mathbf{m} -update step A.24 can be solved by the FISTA algorithm

$$\mathbf{m}^{k+1} = \text{prox}_{\lambda/\rho_1} \left\{ \mathbf{m}^k - t_1 \mathcal{R}^* (\mathcal{R}\mathbf{m}^k - \mathbf{b}^k) \right\}, \quad (\text{A.25})$$

where $t_1 > 0$ is a suitable stepsize, which can be selected to be a Lipschitz constant, i.e. $1/t_1 > \lambda_{\max}(\mathcal{R}^*\mathcal{R})$ to make sure the augmented Lagrangian function does not increase when the \mathbf{m} -update step is approximately solved by equation A.25.

The standard scaled form of the ADMM method utilizes the update steps A.20, A.21 and A.25 to guarantee to converge to the global minimum of the equation A.16 for the robust sparse Radon transform. In our examples, I refer to ADMM as the robust and sparse Radon transform by solving $\ell_1 - \ell_1$ optimization problem, and FISTA as the non-robust sparse Radon transform by solving $\ell_2 - \ell_1$ optimization problem.

A.3 Examples

In my experiments, the trade-off parameters μ for the regularization term in each method are chosen by providing the best performance in terms of relative error of recovery.

A.3.1 Synthetic example

To compare the IRLS and ADMM methods, we first synthesize an example containing five parabolic plane-wave events with different curvatures to mimic a common receiver gather in the conventional seismic acquisition. The numerical example consists of 60 receivers and 80

shots. The source wavelet is synthesized with a Ricker wavelet of central frequency 30 Hz. We apply the robust sparse Radon transform in the common receiver gathers for denoising, including blending noise, erratic ambient noise and random noise.

First, we synthesize the data only containing the blending noise ($BF = 2$) to compare the different algorithms for less blending noise attenuation (Figure A.1). Figure A.1a shows the clean data, and the noisy data can be found in Figure A.1b. Figure A.1c - A.1f shows the denoising results with ADMM, FISTA, IRLS (non-robust) and IRLS (robust) methods, respectively. We can observe that all algorithms work well for denoising when less blending noise exists. The error estimation sections can be found in Figure A.1i - A.1l. These figures show that the ADMM and FISTA methods have insignificant signal leakage compared with IRLS methods. This is because ADMM and FISTA methods adopt proximity (soft-thresholding) operator for solving ℓ_1 problem, which is a biased estimator (Wen et al., 2019), and IRLS methods adopt a weighting factor to down-scaling the outliers when the data contains more signals. In Table A.1, we also observe that IRLS method can achieve higher SNR values than the ADMM and FISTA methods. The IRLS method (robust) gains an extremely high SNR value of $SNR = 127.83$ dB.

Our next example is to test including more blending noise ($BF = 8$). The denoising results can be observed in Figure A.2. Again, Figure A.2a and A.2b shows the clean and noisy data. Figure A.2c - A.2f shows the denoising results by ADMM, FISTA, IRLS (non-robust) and IRLS (robust) methods, respectively. In this case, we still can find that all methods can obtain high-quality denoising results. In the error estimation sections (Figure A.2i - A.2l), IRLS methods start to show signal leakage. In Table A.1, for the high blending noise ($BF = 8$) case, we can find that the SNR value of the IRLS method (robust) decreases dramatically (from 127.83 dB to 28.49 dB), which is almost the same SNR value obtained by the ADMM method. In contrast, regarding the time-consuming part, the ADMM method spends less time than the IRLS method (robust).

The third example entails testing the condition of heavy blending noise ($BF = 8$) and erratic ambient noise simultaneously. The clean and noisy data can be found in Figure A.3a and A.3b. In Figure A.3b and A.3h, we can observe that except for blending noise, the noisy data also includes three high-amplitude erratic ambient noise. This is to mimic the erratic noise generated by the ambient environment. The denoising results can be found in Figure A.3c - A.3f. In this case, we find that by adding the erratic noise, the FISTA and IRLS (non-robust) methods do not work effectively for denoising. The robust methods (ADMM and IRLS) still obtain acceptable denoising results. When comparing Figure A.3c and A.3f, we can find that the ADMM method outperforms the IRLS method. In Figure A.3f, we can observe the denoising result includes some artefacts, and this is also verified in the error estimation sections (Figure A.3i - A.3l). The reason for the IRLS method (robust)

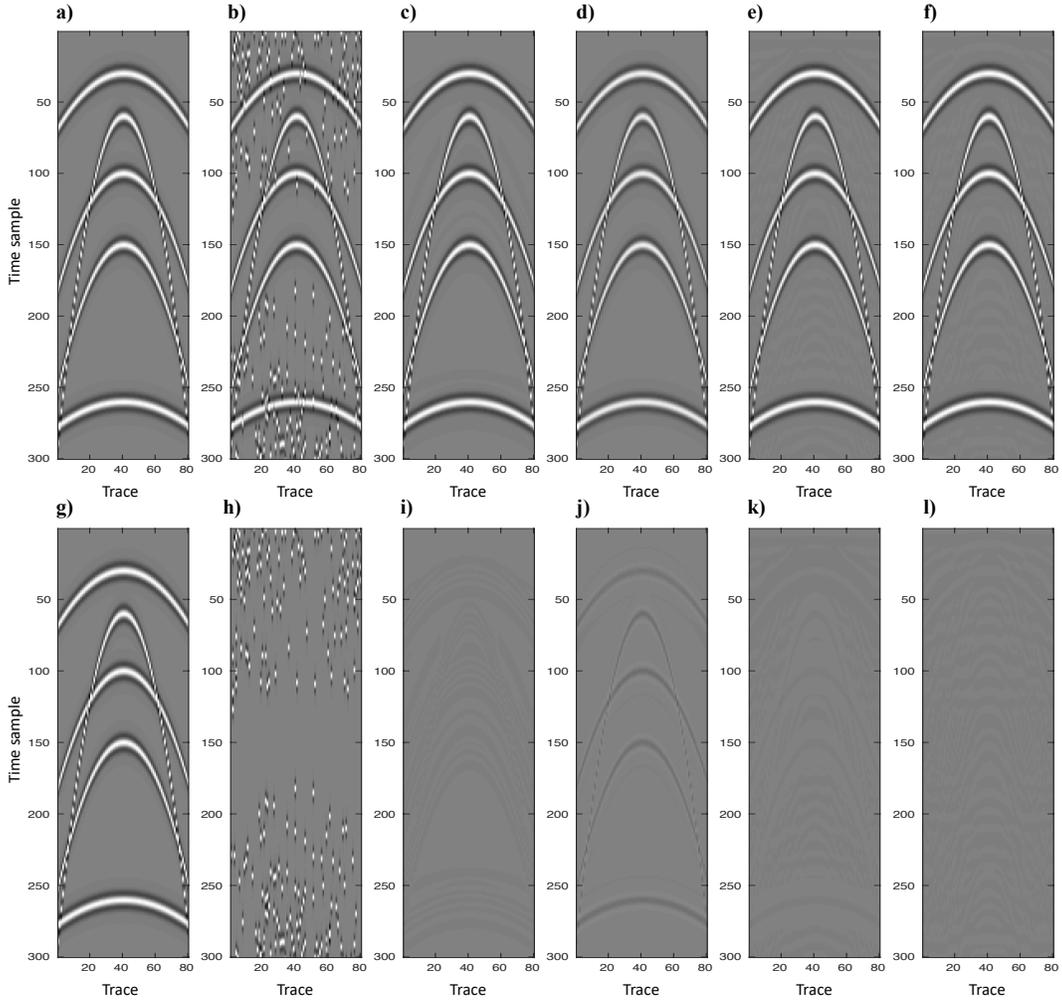


Figure A.1: Comparison of blending noise ($BF = 2$) attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).

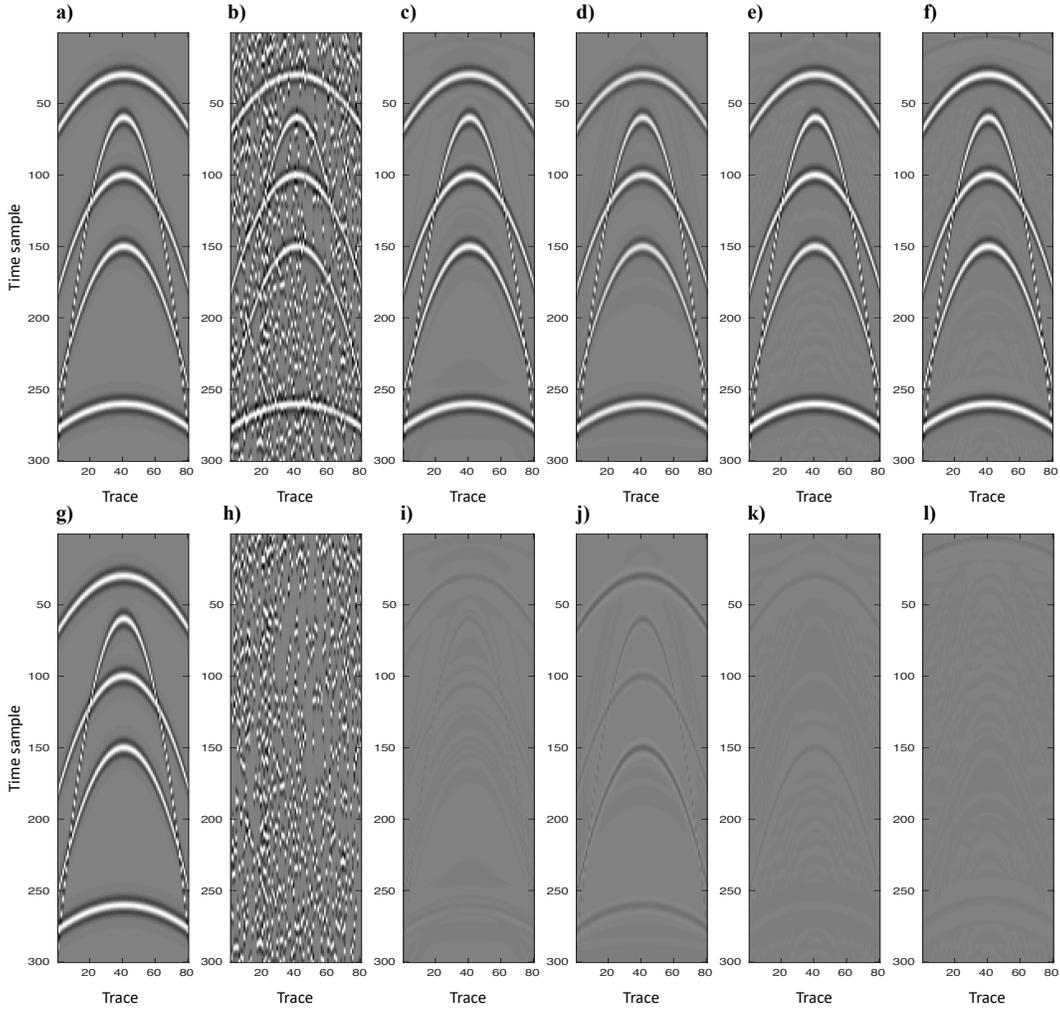


Figure A.2: Comparison of blending noise ($BF = 8$) attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).

to include artifacts is because when the noise is dominant, the weighting factors start to down-scaling the signals, and some noises will be treated as signals. In Table A.1, we also can find that the SNR value for the ADMM method still keeps “robust” compared with the IRLS method (robust), as the SNR value for the IRLS method (robust) decreases a lot. At the same time, the ADMM method uses less computational time than the IRLS (robust) method.

Our last example is to test an extremely noisy condition, which includes heavy blending noise ($BF = 8$), erratic ambient noise and random noise. The clean data and the noisy data can be found in Figure A.4a and A.4b and the denoising results can be observed in Figure A.4c - A.4f. In this extremely noisy situation, we can observe that the ADMM still achieves a reasonable denoising result compared to the other three methods. Due to the high-amplitude erratic ambient noise, the non-robust methods (FISTA and IRLS) fail to denoise. At the same time, the IRLS method (robust) also obtains an unacceptable result because more noises are treated as signals. The error sections (Figure A.4i - A.4l) also confirm our observations. In the extreme noisy test, the ADMM also shows some signal leakage. In Table A.1, we can observe that the ADMM method gains a high SNR value and consumes less time than the other three methods.

Synthetic Case	Algorithm	Time (sec)	SNR (dB)
Blending Noise ($BF = 2$)	ADMM	10.69	32.66
	FISTA	7.25	18.28
	IRLS- ℓ_2	4.63	32.81
	IRLS- ℓ_1	10.72	127.83
Blending Noise ($BF = 8$)	ADMM	4.19	27.79
	FISTA	7.22	16.94
	IRLS- ℓ_2	5.72	26.98
	IRLS- ℓ_1	17.28	28.49
Blending+Erratic Noise ($BF = 8$)	ADMM	7.10	24.09
	FISTA	7.26	-7.08
	IRLS- ℓ_2	21.47	-8.84
	IRLS- ℓ_1	22.46	8.97
Blending+Erratic +Random Noise ($BF = 8$)	ADMM	5.12	12.47
	FISTA	7.28	-6.68
	IRLS- ℓ_2	21.83	-8.87
	IRLS- ℓ_1	22.44	2.88

Table A.1: Comparison of different algorithms for robust/non-robust Radon transform method for different noise attenuation with synthetic examples. Note that the IRLS- ℓ_2 and FISTA denote the non-robust Radon transform by solving $\ell_2 - \ell_1$ optimization problem, and the IRLS- ℓ_1 and ADMM denote the robust Radon transform by solving $\ell_1 - \ell_1$ optimization problem.

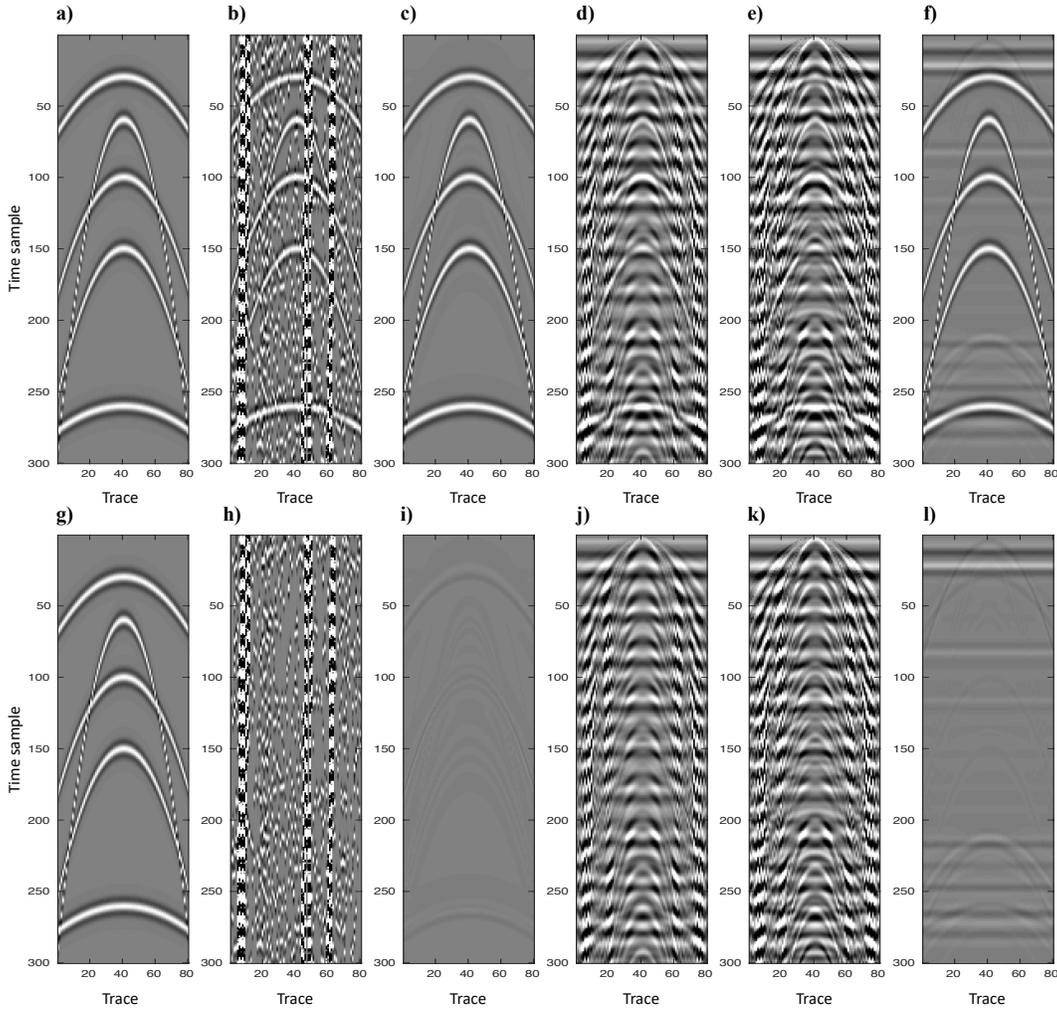


Figure A.3: Comparison of blending noise ($BF = 8$) and erratic ambient noise attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).

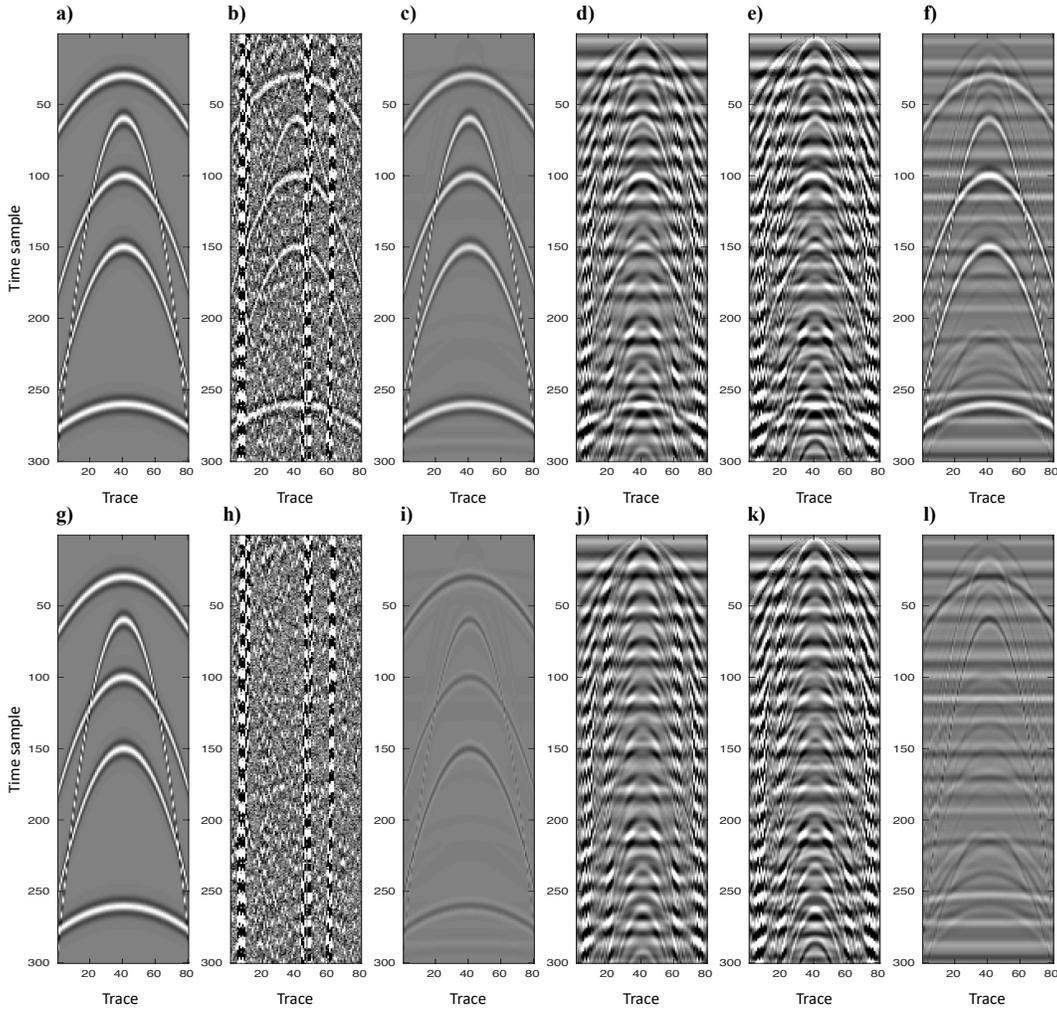


Figure A.4: Comparison of blending noise ($BF = 8$), erratic ambient noise and random noise attenuation with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method (Robust). (d) Denoising with FISTA method (Non-robust). (e) Denoising with IRLS method (Non-robust). (f) Denoising with IRLS method (Robust). (g) Clean data. (h) Noise section. (i) Difference between (c) and (a). (j) Difference between (d) and (a). (k) Difference between (e) and (a). (l) Difference between (f) and (a).

A.3.2 Real data example

I also test our method for different types of noise denoising with actual seismic data from the Gulf of Mexico. The blending noise is included by numerically blending, and the traces of erratic ambient noise are randomly added to 5 out of 100 traces in each unblended common shot gather.

Our first example is to test denoising with blending noise $BF = 2$ and erratic ambient noise. The noisy data and denoising results can be found in Figure A.5a - A.5e. Due to the erratic ambient noise, the non-robust methods fail to denoise the erratic ambient noise (Figure A.5c and A.5e). Both ADMM and IRLS (robust) methods effectively suppress the blending noise and erratic noise (Figure A.5b and A.5d). In the error sections (Figure A.5g - A.5j), we can observe that the ADMM methods still have small signal leakage due to the biased proximity operator estimation. In Table A.2, I maintain that both ADMM and IRLS methods obtain almost the same SNR value, which means for this case, the denoising effects for both robust techniques are similar. However, regarding computational time, the ADMM method has a significant advantage to the robust IRLS method.

The next example is to test the denoising effect by adding random noise based on the first example (Figure A.5). As we know that non-robust methods (FISTA and non-robust IRLS method) do not work for erratic ambient noise attention. We only compare the ADMM and IRLS (robust) methods in the following examples. The clean and noisy data are shown in Figure A.6a and A.6b. The denoising results are displayed in Figure A.6c and A.6d. Compare Figure A.6c and A.6d, we notice some noise left for the robust IRLS method. This is also verified by the residual sections (Figure A.6g and A.6h). In Table A.2, we notice that the SNR values for both robust methods are similar, while ADMM outperforms IRLS in computational time cost.

We repeat the same test with heavier blending noise ($BF = 8$) in Figure A.7 and Figure A.8. We still notice that the ADMM method typically has signal leakage drawback for the denoising problem, while the robust IRLS method has noise left drawback, even though the SNR value for both methods are similar. However, the ADMM method outperforms the robust IRLS method in the time-consuming part (See Table A.2).

A.4 Conclusions

This appendix is an extension of Chapter 3 of the IRLS and ADMM methods for solving the robust and sparse Radon transform when tackling different denoising scenarios, including blending noise, high-amplitude erratic ambient noise and random noise. Via examples, I

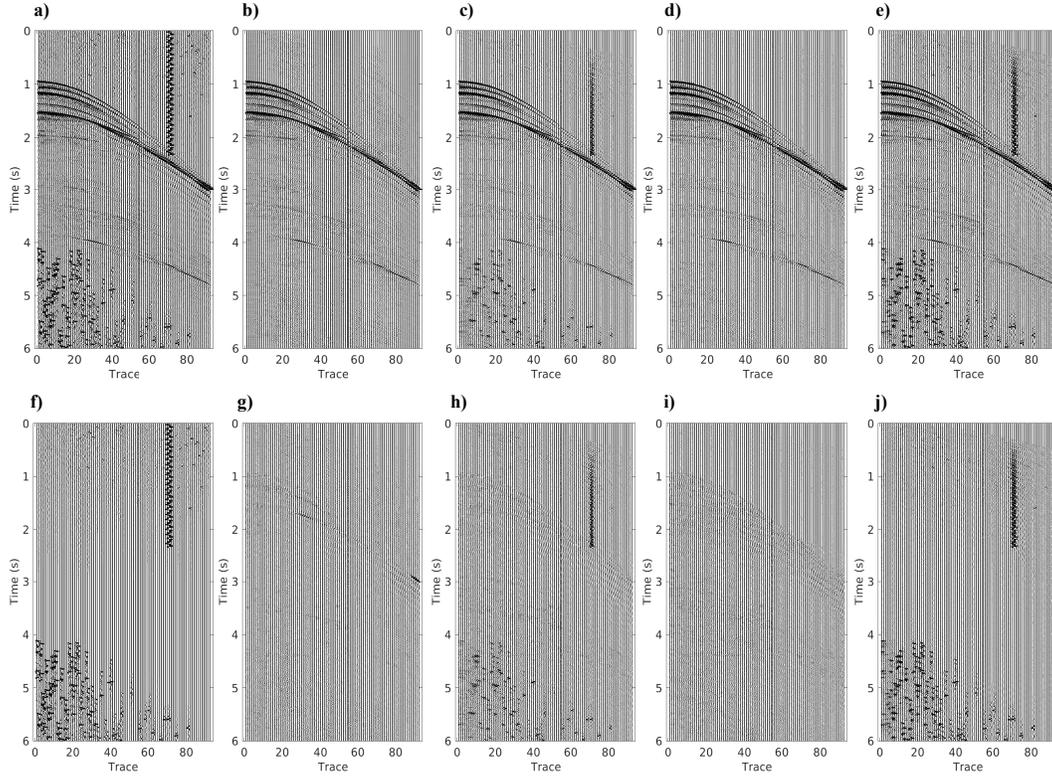


Figure A.5: Denoising (blending noise $BF = 2 +$ erratic ambient noise) results comparison with different algorithms. (a) Noisy data. (b) Denoising with ADMM method. (c) Denoising with FISTA method. (d) Denoising with IRLS method (Non-robust). (e) Denoising with IRLS method (Robust). (f) Noise section. (g) Residual section by ADMM method. (h) Residual section by FISTA method. (i) Residual section by IRLS method (Non-robust). (j) Residual section by IRLS method (Robust).

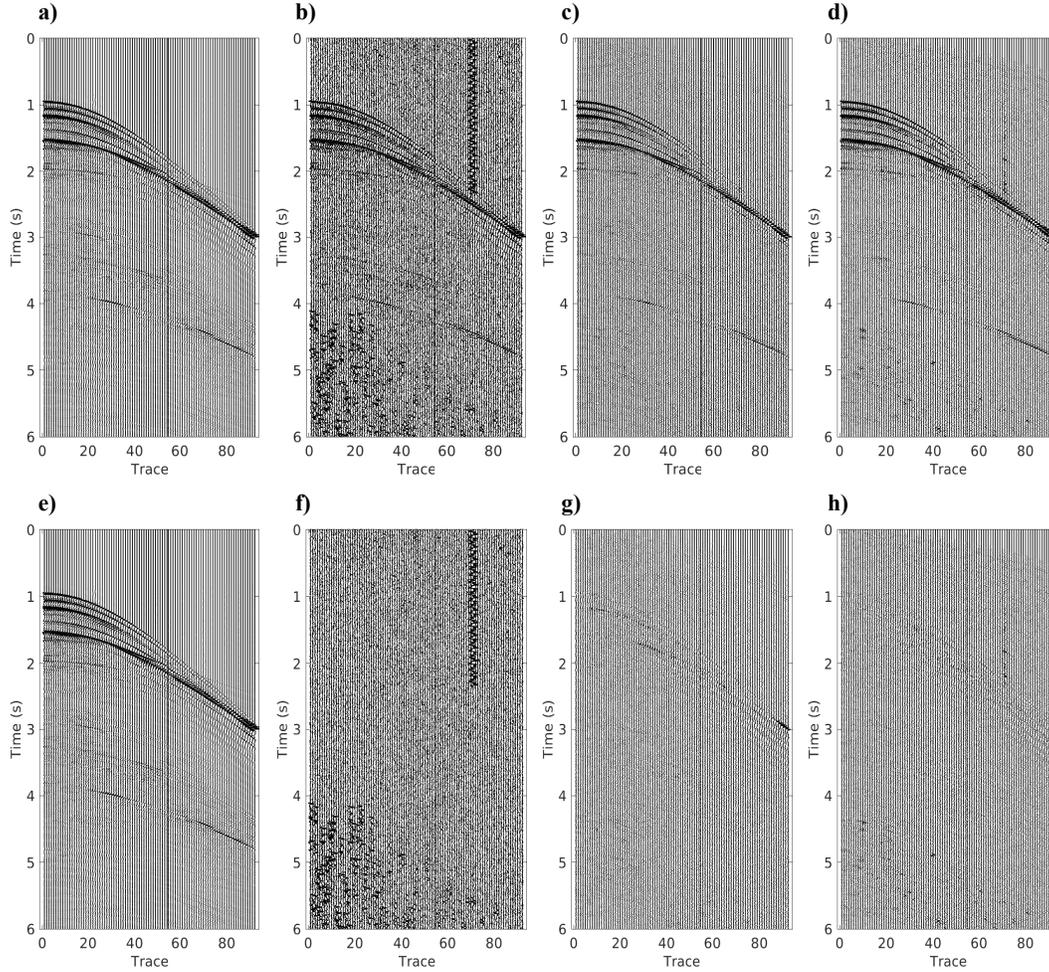


Figure A.6: Denoising (blending noise $BF = 2$ + erratic ambient noise + random noise) results comparison with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean data. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).

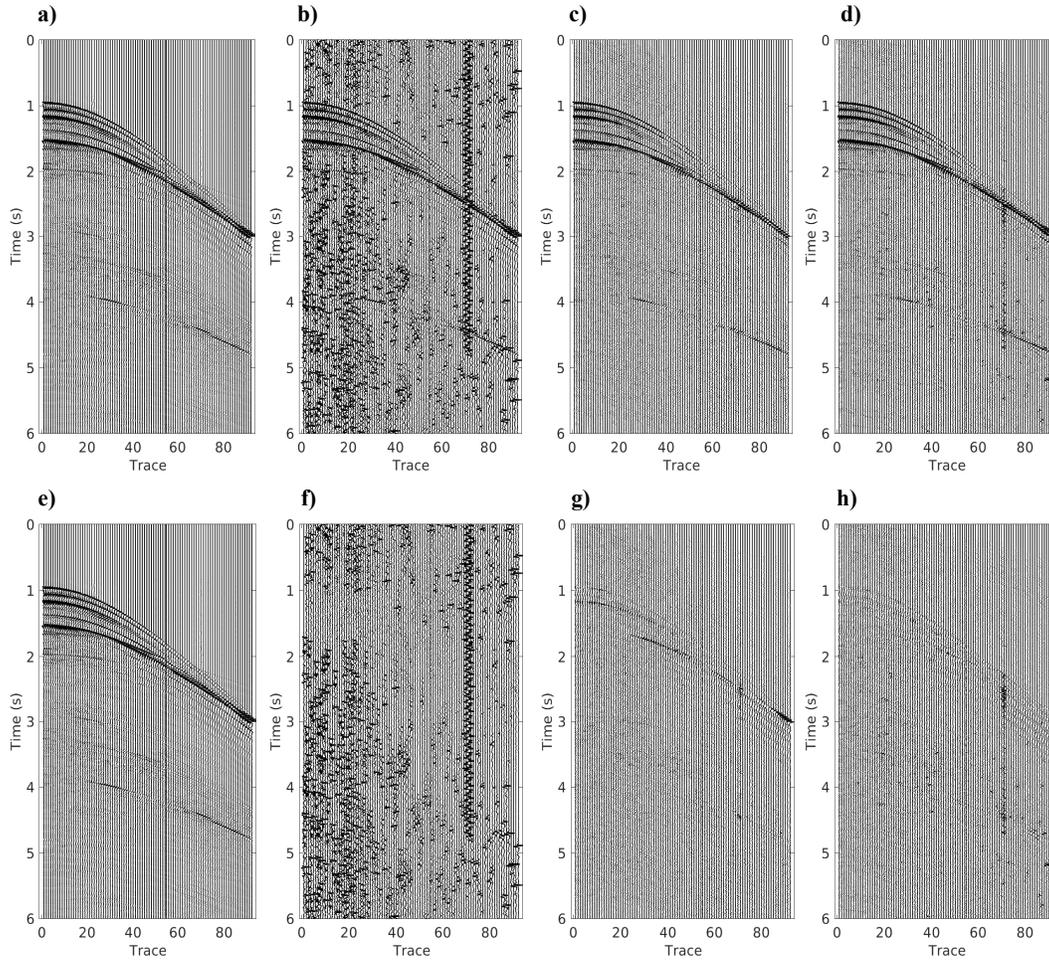


Figure A.7: Denoising (blending noise $BF = 8$ + erratic ambient noise) results in comparison with different algorithms. (a) Clean data, (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean section. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).

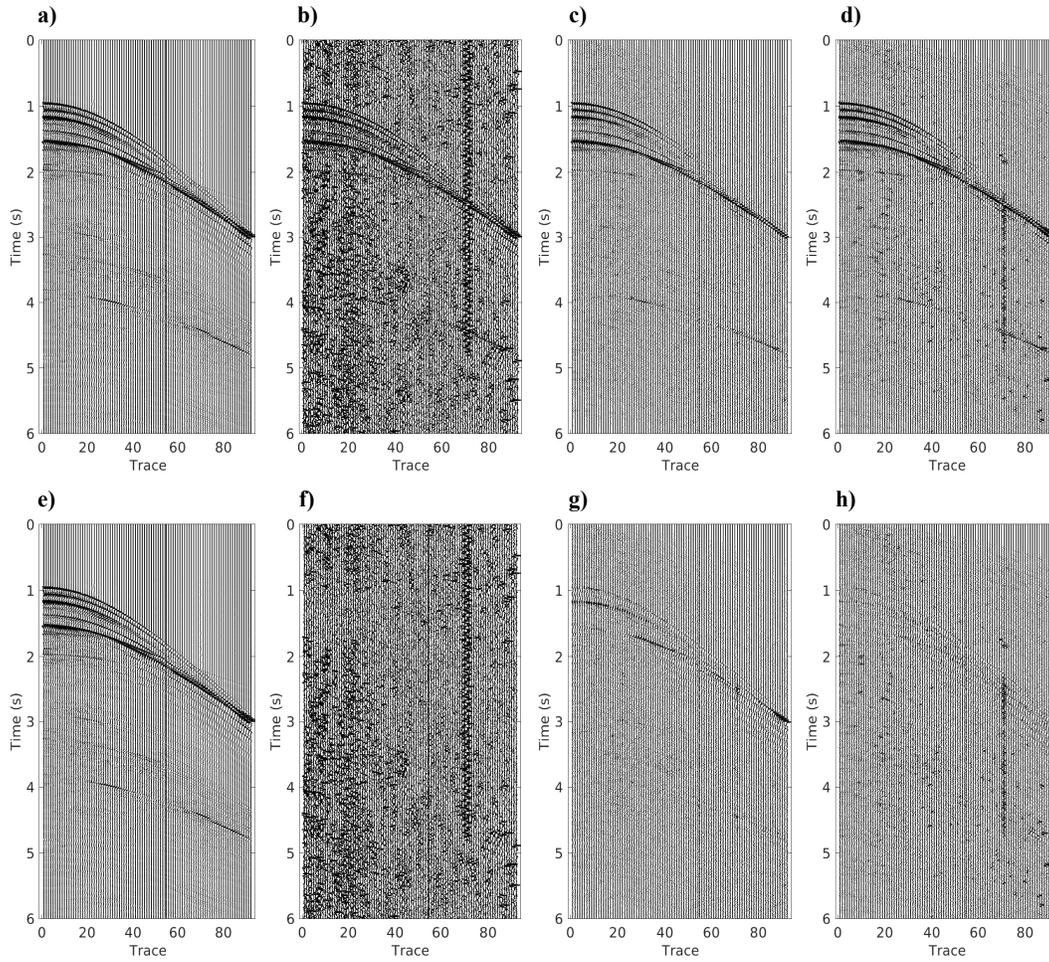


Figure A.8: Denoising (blending noise $BF = 8$ + erratic ambient noise + random noise) results comparison with different algorithms. (a) Clean data. (b) Noisy data. (c) Denoising with ADMM method. (d) Denoising with IRLS method (Robust). (e) Clean section. (f) Noise section. (g) Residual between (c) and (a). (h) Residual between (d) and (a).

Real Case	Algorithm	Time (sec)	SNR (dB)
Blending($BF = 2$)+Erratic Noise	ADMM	236.80	9.02
	IRLS- ℓ_1	986.57	8.79
Blending($BF = 2$)+Erratic+Random Noise	ADMM	229.70	4.73
	IRLS- ℓ_1	1058.6	3.92
Blending($BF = 8$)+Erratic Noise	ADMM	331.29	4.05
	IRLS- ℓ_1	1004.8	3.75
Blending($BF = 8$)+Erratic+Random Noise	ADMM	311.06	2.85
	IRLS- ℓ_1	1050.5	1.72

Table A.2: Comparison of different algorithms for robust Radon transform method for different noise attenuation with real examples. Note that the IRLS- ℓ_1 and ADMM denote the robust Radon transform by solving $\ell_1 - \ell_1$ optimization problem.

conclude that the robust (ADMM and robust IRLS) methods outperform the non-robust (FISTA and non-robust IRLS) methods when data are contaminated with erratic ambient noise. Also, the ADMM method shows a great superiority in computational time, which could be a suitable replacement for the IRLS method when dealing with the $\ell_1 - \ell_1$ optimization problem. Beyond that, due to the application of the proximity operator, the ADMM method typically produces solutions with signal leakage, significantly when the data are contaminated by a considerable amount of noise. The latter is a problem that could be solved by including a debiasing final stage in the ADMM algorithm.

APPENDIX B

Fast and computational-efficient I-MSSA ¹

B.1 Introduction

For separation of compressive simultaneous-source data deployed on an irregular acquisition grid in Chapter 5, despite the accuracy of SSA for low-rank optimization, its computational cost has always been a concern due to the Singular Value Decomposition (SVD) implementation. The Randomized SVD (R-SVD) has been proposed to replace conventional SVD to achieve a more efficient algorithm. Even though the R-SVD shrinks the size of the Hankel matrix, it still needs to form the Hankel matrix and apply the SVD for rank reduction. For multidimensional data, building block-Hankel matrices requires a high computational cost, which could be a drawback for industrial applications. In this appendix, I provide a more efficient and faster algorithm for low-rank estimation named Fast and computational-efficient Multidimensional Singular Spectrum Analysis (FMSSA). The FMSSA method avoids building block-Hankel matrices and uses randomized QR decomposition to substitute the SVD procedure. Also, the Hankel matrix's final anti-diagonal averaging is computed efficiently via a convolution algorithm. The main idea is borrowed from Cheng et al. (2019) and is first implemented for compressive simultaneous-source data deployed on an irregular grid. This FMSSA algorithm is developed in Julia language and constitutes a deblending part in SeismicJulia (<https://github.com/SeismicJulia>) for seismic data processing.

This appendix is an extended part of Chapter 5 and follows the subsequent structure. First, I describe the FMSSA algorithm and provide the pseudocode of it. Then, I offer a brief

¹A version of this appendix is published in Lin, R., Y. Guo, F. Carozzi and M. D. Sacchi, 2022, Interpolated fast and computational-efficient multidimensional Singular Spectrum Analysis (I-FMSSA) for compressive simultaneous-source data processing, SEG/AAPG/SEPM International Meeting for Applied Geoscience & Energy 2022.

review of compressive simultaneous source data processing via FMSSA with binning and the I-FMSSA method. Finally, I provide a comparative study of the I-MSSA method with the I-FMMSA method via synthetic and field data examples.

B.2 Method

B.2.1 Fast and computational-efficient Multidimensional Singular Spectrum Analysis (FMSSA) filtering

The MSSA algorithm (Oropeza and Sacchi, 2011) is an accurate method for $f - x - y$ low-rank denoising, and it contains three main steps that are applied in frequency slides: 1) *Hankelization*. 2) *Rank-reduction*. 3) *Anti-diagonal averaging*.

The main drawbacks of the MSSA algorithm are:

- Forming Hankel matrices introduce a high computational cost and increases memory requirement.
- Classical MSSA applies SVD on large matrices, which could be expensive for multidimensional problems.

The fast and computational-efficient MSSA (FMSSA) algorithm (Cheng et al., 2019) is an appropriate substitution for MSSA when one desires to avoid adopting the SVD for its rank-reduction step. Moreover, the FMSSA avoids forming explicit form of Hankel matrices. In essence, the main features of FMSSA can be summarized as follows:

- *Hankel matrix-vector products are computed via FFTs to avoid building explicit form Hankel matrices.*

The Hankel matrix can be embedded into a circulant matrix so that Hankel matrix-vector multiplication can be computed via the fast Fourier transform (FFT):

$$\mathbf{H}\mathbf{x} = \mathbf{T}\hat{\mathbf{x}} = \mathbf{C}\check{\mathbf{x}} \quad (\text{B.1})$$

$$\begin{bmatrix} D_1 & D_2 \\ D_2 & D_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} D_2 & D_1 \\ D_3 & D_2 \end{bmatrix} \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} D_1 & D_3 & D_2 \\ D_2 & D_1 & D_3 \\ D_3 & D_2 & D_1 \end{bmatrix} \begin{bmatrix} x_2 \\ x_1 \\ 0 \end{bmatrix}$$

and

$$\mathbf{C}\check{\mathbf{x}} = \begin{bmatrix} D_1 & D_3 & D_2 \\ D_2 & D_1 & D_3 \\ D_3 & D_2 & D_1 \end{bmatrix} \begin{bmatrix} x_2 \\ x_1 \\ 0 \end{bmatrix} = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{c}) \circ \mathcal{F}(\hat{\mathbf{x}})) \quad (\text{B.2})$$

where $\mathbf{H} = \begin{bmatrix} D_1 & D_2 \\ D_2 & D_3 \end{bmatrix}$ denotes the Hankel matrix, and $\mathbf{T} = \begin{bmatrix} D_2 & D_1 \\ D_3 & D_2 \end{bmatrix}$ represents the Toeplitz matrix, and $\hat{\mathbf{x}} = \text{reverse}(\mathbf{x})$ means reverse the elements order of vector \mathbf{x} . Matrix $\mathbf{C} = \begin{bmatrix} D_1 & D_3 & D_2 \\ D_2 & D_1 & D_3 \\ D_3 & D_2 & D_1 \end{bmatrix}$ denotes the circulant matrix, which can be computed via Fast Fourier transform (FFT), and $\mathbf{c} = [D_1 \ D_2 \ D_3]^T$, $\tilde{\mathbf{x}} = [x_2 \ x_1 \ 0]^T$, and \circ means element-wise multiplication. Algorithm 10 shows a detailed summary of Hankel matrix-vector product calculation via fast Fourier transform.

- *Instead of applying SVD to the Hankel matrix, a randomized QR decomposition (rQRd) is adopted as an alternative for fast reduced-rank approximation.*

A random projection is performed to reduce the size of the Hankel matrix:

$$\mathbf{M} = \mathbf{H}\Omega \quad (\text{B.3})$$

where \mathbf{H} is the Hankel matrix with size of $N_r \times N_c$, and Ω denotes a random matrix with size of $N_c \times p$ that composes of p independent vectors ($p \ll N_c$), and matrix \mathbf{M} is a much smaller matrix with size of $N_r \times p$. Then, an economic-size QR decomposition is applied to matrix \mathbf{M} :

$$[\mathbf{Q}, \mathbf{R}] = \text{qr}(\mathbf{M}) \quad (\text{B.4})$$

Finally, the low-rank estimation of Hankel matrix \mathbf{H} can be computed through the orthonormal basis \mathbf{Q} :

$$\hat{\mathbf{H}} = \mathbf{Q}\mathbf{Q}^H\mathbf{H} \quad (\text{B.5})$$

- *The convolution operator is adopted to accelerate the anti-diagonal averaging process.*
- For simplicity, we assume that \mathbf{H} is a rank of $p = 1$ matrix, and we let $\mathbf{q}_1 = \mathbf{Q}$ and $\mathbf{t}_1 = \mathbf{Q}^H\mathbf{H}$, then equation B.5 can be rewritten as:

$$\hat{\mathbf{H}} = \mathbf{q}_1\mathbf{t}_1 \quad (\text{B.6})$$

In addition, $\mathbf{t}_1 = \mathbf{Q}^H\mathbf{H}$ can be computed with fast Hankel matrix-vector product

(equation B.1). Therefore, the anti-diagonal averaging can be expressed as :

$$\hat{\mathbf{D}} = \begin{cases} \frac{1}{i} \sum_{j=1}^i q_{1_j} t_{1_{i-j+1}}, & 1 \leq i \leq N_c, \\ \frac{1}{N_c} \sum_{j=1}^{N_c} q_{1_j} t_{1_{i-j+1}}, & N_c \leq i \leq N_r, \\ \frac{1}{L-i+1} \sum_{j=i-N_r+1}^{N_c} q_{1_j} t_{1_{i-j+1}}, & N_r \leq i \leq L, \end{cases} \quad (\text{B.7})$$

$$= w \sum_{j=1}^L q_{1_j} t_{1_{i-j+1}}.$$

Equation B.7 can be written in a matrix form:

$$\hat{\mathbf{D}} = \mathbf{w} \circ (\mathbf{q}_1 * \mathbf{t}_1) = \mathbf{w} \circ \mathcal{F}^{-1} \left(\mathcal{F}(\mathbf{q}_1) \circ \mathcal{F}(\mathbf{t}_1) \right) \quad (\text{B.8})$$

where \circ means element-wise multiplication and $*$ denotes convolution operation. \mathcal{F} and \mathcal{F}^{-1} represent fast Fourier transform and inverse fast Fourier transform, respectively. Therefore, for rank = p , equation B.8 can be generalized as:

$$\begin{aligned} \hat{\mathbf{D}} &= \mathbf{w} \circ [(\mathbf{q}_1 * \mathbf{t}_1) + (\mathbf{q}_2 * \mathbf{t}_2) + \cdots + (\mathbf{q}_p * \mathbf{t}_p)] \\ &= \mathbf{w} \circ \sum_{i=1}^p \mathbf{q}_i * \mathbf{t}_i = \mathbf{w} \circ \sum_{i=1}^p \mathcal{F}^{-1} \left(\mathcal{F}(\mathbf{q}_i) \circ \mathcal{F}(\mathbf{t}_i) \right) \end{aligned} \quad (\text{B.9})$$

Algorithm 11 shows the detailed summary of the FMSSA algorithm based on the strategies mentioned above when tackling a 3D data cube ($t - x - y$).

B.2.2 Review of compressive simultaneous source data processing via FMSSA and binning

Conventional processing for compressive simultaneous source data often adopt binning to represent the desired data on a regular grid. In this vein, we write the separation and reconstruction as the solution that minimizes the following cost function

$$J(\mathbf{D}) = \|\mathbf{b} - \mathcal{B}\mathcal{T}\mathbf{D}\|_2^2 \quad (\text{B.10})$$

where \mathbf{b} denotes the blended data, and \mathcal{B} is the blending operator. The operator \mathcal{T} is the sampling operator (Liu and Sacchi, 2004; Naghizadeh and Sacchi, 2010; Cheng and Sacchi, 2015), and \mathbf{D} denotes data deployed in regular-grid coordinates.

Again, the projected gradient-descent (PGD) method (Cheng and Sacchi, 2016; Lin et al., 2021) is adopted to solve the problem B.10 by defining the gradient-descent step followed

Algorithm 10 Fast Level-2 Hankel matrix-vector product \mathcal{P}_{FHP}

```

1: function  $\mathbf{y} = \mathcal{P}_{FHP}(\mathbf{c}, \mathbf{v}, \text{flag} = \text{"forward" or "adjoint"})$ 
2:   Initialization:
3:     Size of  $\mathbf{c}$ :  $(L_x, L_y) = \text{size}(\mathbf{c})$ .
4:     Size of level-2 block Hankel matrix:
5:        $N_r = \lfloor L_x/2 \rfloor + 1$  and  $N_c = L_x + 1 - N_r$ .
6:        $M_r = \lfloor L_y/2 \rfloor + 1$  and  $M_c = L_y + 1 - M_r$ .
7:     Reverse the elements order of  $\mathbf{v}$  for each dimension:
8:        $\hat{\mathbf{v}} = \text{reverse}(\text{reverse}(\mathbf{v}, \text{dims} = 1), \text{dims} = 2)$ .
9:     Fourier transform:  $\mathbf{C} = \mathcal{F}(\mathbf{c})$ .
10:  if forward then
11:    Padding zeros:  $\hat{\mathbf{v}} = [\hat{\mathbf{v}}; \text{zeros}(N_c - 1, M_c - 1)]$ 
12:    Fourier transform:  $\mathbf{V} = \mathcal{F}(\hat{\mathbf{v}})$ 
13:    Element-wise multiplication and inverse Fourier transform:  $\mathbf{r} = \mathcal{F}^{-1}(\mathbf{C} \circ \mathbf{V})$ 
14:    Truncate:  $\mathbf{y} = \mathbf{r}[N_r : L_x, M_r : L_y]$ 
15:    Output a vector:  $\mathbf{y} = \text{reshape}(\mathbf{y}, [], 1)$ 
16:  else
17:    Padding zeros:  $\hat{\mathbf{v}} = [\hat{\mathbf{v}}; \text{zeros}(N_r - 1, M_r - 1)]$ 
18:    Fourier transform:  $\mathbf{V} = \mathcal{F}(\hat{\mathbf{v}})$ 
19:    Conjugate property of Fourier transform:  $\mathbf{C}^* = \text{conj}(\mathbf{C})$ 
20:    Element-wise multiplication and inverse Fourier transform:  $\mathbf{r} = \mathcal{F}^{-1}(\mathbf{C}^* \circ \mathbf{V})$ 
21:    Truncate:  $\mathbf{y} = \mathbf{r}[N_c : L_x, M_c : L_y]$ 
22:    Output a vector:  $\mathbf{y} = \text{reshape}(\mathbf{y}, [], 1)$ 
23:  end if
24: end function

```

Algorithm 11 Fast and computational-efficient MSSA (*FMSSA*)

```

1: Inputs:
2:   Seismic data:  $\mathbf{d}$ ; rank:  $p$ .
3: Output:
4:   FSSA filtered data:  $\hat{\mathbf{d}}$ .
5: Initialization:
6:    $\mathbf{D}(\omega, x, y) \leftarrow \mathbf{d}(t, x, y)$  (1D FFT)
7:    $(N_t, N_1, N_2) = \text{size}(\mathbf{D})$ 
8:    $M_1 = \lfloor N_1/2 \rfloor + 1$  and  $L_1 = N_1 + 1 - M_1$ 
9:    $M_2 = \lfloor N_2/2 \rfloor + 1$  and  $L_2 = N_2 + 1 - M_2$ 
10: for  $\omega = \omega_{min} : \omega_{max}$  do
11:    $\mathbf{D} = D[\omega, :, :]$ 
12:    $\Omega = \text{rand}(M_1, M_2, p)$ 
13:   for  $i = 1 : p$  do
14:      $\mathbf{Y}[:, i] = \mathcal{P}_{FHP}(\mathbf{D}, \Omega[:, :, i], \text{flag} = \text{"forward"})$ 
15:   end for
16:    $[\mathbf{Q}, \mathbf{R}] = \text{qr}(\mathbf{Y})$ 
17:   for  $i = 1 : p$  do
18:      $\mathbf{q} = \text{reshape}(\mathbf{Q}[:, i], L_1, L_2)$ 
19:      $\mathbf{z} = \mathcal{P}_{FHP}(\mathbf{D}, \mathbf{q}, \text{flag} = \text{"adjoint"})$ 
20:      $\tilde{\mathbf{D}} = \tilde{\mathbf{D}} + \mathcal{F}^{-1}(\mathcal{F}(\mathbf{q}) \circ \mathcal{F}(\mathbf{z}))$ 
21:   end for
22:    $\tilde{D}[\omega, :, :] = \tilde{\mathbf{D}}$ 
23: end for
24:  $\mathbf{d}(t, x, y) \leftarrow \tilde{\mathbf{D}}(\omega, x, y)$  (1D IFFT)

```

by a projection of the form

$$\begin{aligned}\mathbf{D}^\nu &= \mathcal{P} [\mathbf{D}^{\nu-1} - \lambda \nabla J (\mathbf{D}^{\nu-1})] \\ &= \mathcal{P} [\mathbf{D}^{\nu-1} - \lambda \mathcal{T}^* \mathcal{B}^* (\mathcal{B} \mathcal{T} \mathbf{D}^{\nu-1} - \mathbf{b})]\end{aligned}\tag{B.11}$$

where ∇J is the gradient of the cost function J , the scalar λ is the length of step-size, and \mathcal{P} indicates the projection operator. The operator \mathcal{P} could be a denoising algorithm based on MSSA filter (Oropeza and Sacchi, 2011) or FMSSA filter (Cheng et al., 2019).

The conventional rank-reduction deblending and reconstruction methods assume a regular grid distribution of traces. In this case, the original source coordinates are allocated to binned coordinates via simple nearest-neighbour interpolation. However, when more than one trace falls into the same bin, the binning strategy introduces errors in the amplitude and phase of the traces, leading to distortion of the recovered deblended and reconstructed signals. We refer to the conventional method as MSSA deblending (projection operator = MSSA) or FMSSA deblending (projection operator = FMSSA).

B.2.3 Review of compressive simultaneous source data processing via Interpolated-FMSSA (I-FMSSA)

For a blending acquisition with an irregular-grid coordinate distribution (especially for shot irregular-grid distribution), the blended data can be written as

$$\mathbf{b} = \mathcal{B} \mathcal{W} \mathbf{D}\tag{B.12}$$

where \mathbf{D} denotes the desired unblended data on the regular grid, and \mathbf{b} is the observed irregular-grid blended acquisition data. The operator \mathcal{W} denotes the window tapered sinc Kaiser interpolation operator that links the traces from the regular desired grid to irregular observation grid (Jiang et al., 2017; Carozzi and Sacchi, 2021). Therefore, the deblending and shot reconstruction can be written as the solution \mathbf{D} that minimizes the following cost function

$$J = \|\mathbf{b} - \mathcal{B} \mathcal{W} \mathbf{D}\|_2^2.\tag{B.13}$$

Equation B.13 can be solved by projected gradient-descent (PGD) method

$$\mathbf{D}^\nu = \mathcal{P} [\mathbf{D}^{\nu-1} - \lambda \mathcal{W}^* \mathcal{B}^* (\mathcal{B} \mathcal{W} \mathbf{D}^{\nu-1} - \mathbf{b})]\tag{B.14}$$

The term $\mathcal{B} \mathcal{W} \mathbf{D}^{\nu-1} - \mathbf{b}$ defines the error in the irregular grid. Next, the adjoint interpolator \mathcal{W}^* and \mathcal{B}^* maps the error back to the pseudo-deblended regular grid. The fitting goal guarantees data fidelity by honouring the true spatial coordinates of the seismic traces. In other

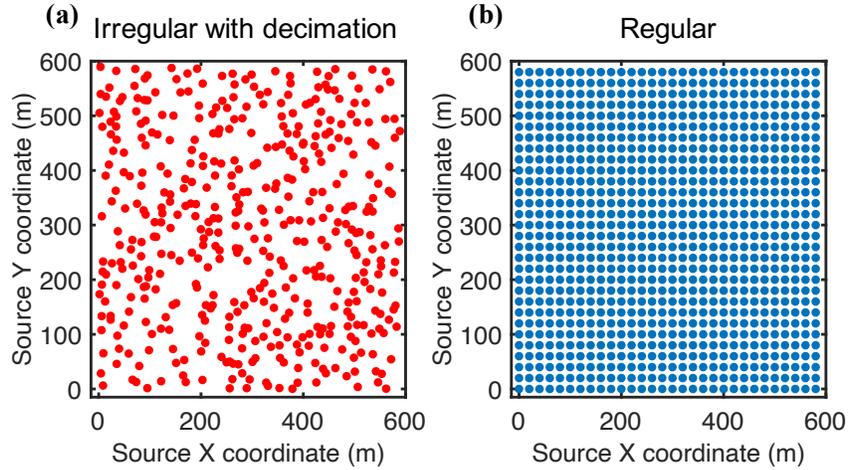


Figure B.1: Source coordinates distribution considered in the synthetic examples. (a) Coordinates of the irregular grid with 50 % decimation. (b) The desired regular grid of output.

words, the algorithm does not introduce time-shift errors arising from the nearest neighbour interpolation (binning). We refer to this method as I-MSSA debrending (projection operator = MSSA) or I-FMSSA debrending (projection operator = FMSSA).

B.3 Examples

We first consider noise-free 3D synthetic data to compare different algorithms. The synthetic example contains three dipping linear events to mimic a small 3D patch of common receiver gather. The regular grid consists of 30×30 source points with interval $\Delta x = \Delta y = 20$ m in the x - and y - directions, and a Ricker wavelet of central frequency 20 Hz was adopted. We add a perturbation to the regular grid to generate the irregular distribution. The perturbation was generated with uniform random numbers in the range $[-\Delta x, \Delta x]$ in x -direction and in the range $[-\Delta y, \Delta y]$ in y -direction. Then 50% source points are randomly decimated to produce the compressive irregular-grid distribution. The geometry of the source coordinate of irregular-grid and desired regular-grid distribution is displayed in Figure B.1. Figure B.1a shows irregular distribution after 50% decimation, and Figure B.1b represents our desired regular-grid output.

We first compare the computational performance of the MSSA projection operator and FMSSA projection operator for debrending and reconstruction. The debrending and reconstruction results for other strategies can be found in Figure B.2. Through Figure B.2d and

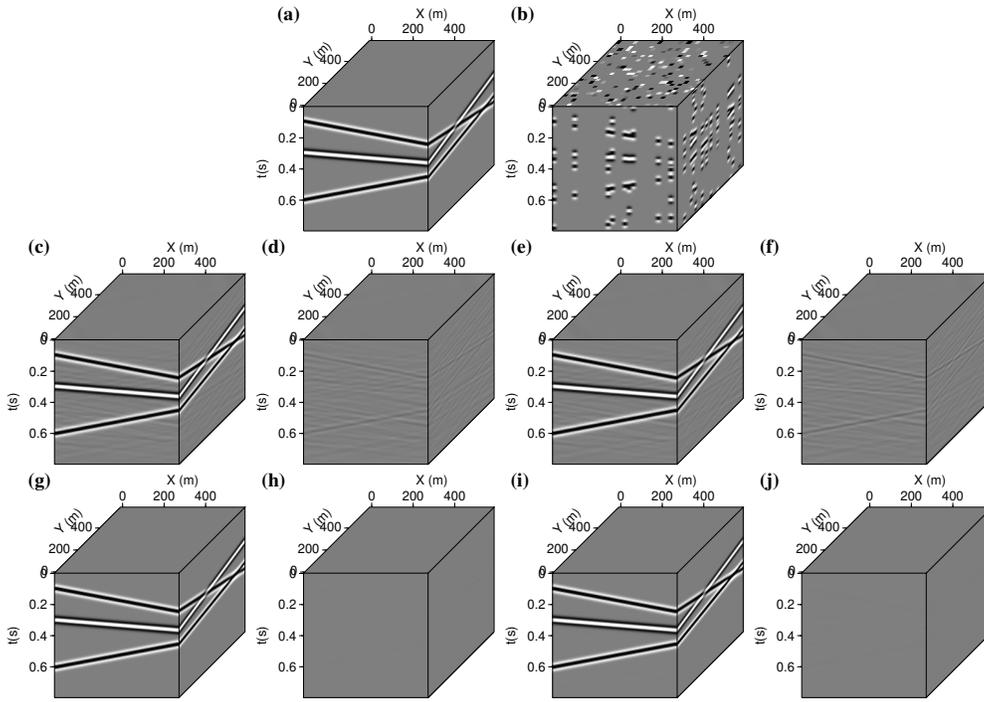


Figure B.2: Reconstruction and debrending results. A comparison for one CRG. (a) Clean regular data volume. (b) Pseudo-debrended data volume after binning. (c) Result with MSSA debrending and binning with $SNR = 15.87$ dB. (d) Residuals between (a) and (c). (e) Result with FMSSA debrending and binning with $SNR = 15.25$ dB. (f) Residuals between (a) and (e). (g) Result with I-MSSA debrending with $SNR = 43.66$ dB. (h) Residuals between (a) and (g). (i) Result with I-FMSSA debrending with $SNR = 43.03$ dB. (j) Residuals between (a) and (i).

B.2f or Figure B.2h and B.2j, we get incredibly the same debrending and reconstruction quality for both MSSA and FMSSA projection operators. Table B.1 shows the detailed comparison of computational time and SNR value. Adopting the FMSSA projection operator shows the superior performance of the computational time without compromising the debrending results. Comparing Figure B.2d, B.2f with Figure B.2h, B.2j, we observe significant signal leakage when one adopts the conventional MSSA or FMSSA debrending method ($SNR \approx 15$ dB). Conversely, with a sinc Kaiser interpolation operator, the proposed debrending and reconstruction methods produce negligible signal leakage ($SNR \approx 43$ dB). The difference in quality arises from small time shifts due to data binning. These time shifts could compromise the lateral coherence of signals.

For the field example, we consider a field 3D blended-acquisition data from Oman (Song

Methods	Time(sec)	SNR(dB)
MSSA	97.68	15.87
FMSSA	23.14	15.25
I-MSSA	108.28	43.66
I-FMSSA	39.28	43.03

Table B.1: Computational time and SNR for irregular-grid reconstruction.

et al., 2019). The survey area is acquired by 12 vibroseis simultaneously. As the FMSSA projection operator shows a significant advantage in the computational time, we only compare FMSSA deblending algorithm with I-FMSSA deblending algorithm for this real example without considering the MSSA projection operator. Figure B.3 delineates the sources' geometry and the receiver's location. The mean interval between sources and source lines is 25 m. Figure B.4 shows the results for an inline slice. Figure B.4a is the observed decimated pseudo-deblended data after binning. Figure B.4b shows the deblending and reconstruction result with the FMSSA deblending method. Figure B.4c displays deblending and reconstruction results with the I-FMSSA deblending method. We observe that the blending interferences have been effectively eliminated, and the missing traces have been reconstructed.

We have the same problem discussed in Chapter 5 for the real data example. Due to the lack of ground truth data as a reference, it is hard to evaluate the performances of the I-FMSSA deblending method (Figure B.4b) and the FMSSA deblending method (Figure B.4c). Again, we calculate the difference (Figure B.4d) between the I-FMSSA deblending method and the FMSSA deblending method and evaluate their difference and quality heuristically. In Figure B.4d, there exist non-negligible differences containing signals. These events likely result from the errors introduced by data binning when one adopts the FMSSA deblending method. Given that binning assigns coordinates to grid points via a crude nearest-point interpolation, one could expect amplitude distortion. The vibroseis source points were not ideally deployed on a regular grid. Hence, one should adopt the I-FMSSA deblending method rather than FMSSA deblending despite the slight differences we might obtain.

B.4 Conclusion

This appendix shows an extension of Chapter 5 by illustrating an inversion scheme for separating and reconstructing irregular-grid compressive simultaneous-source data with a faster

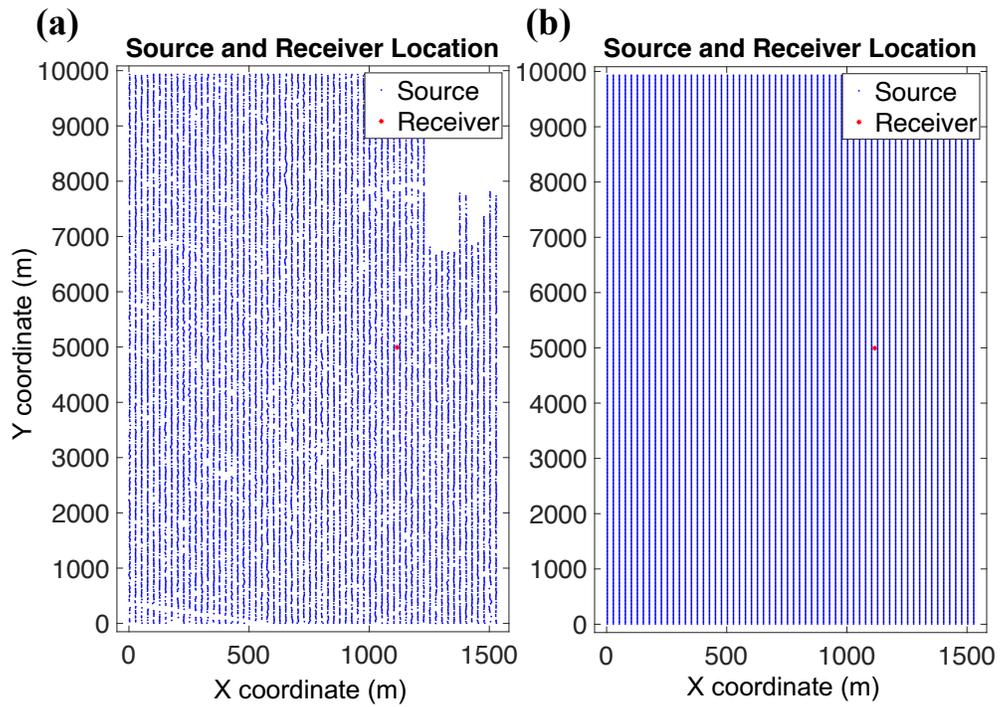


Figure B.3: (a) Observed decimated coordinate distribution containing 17927 source points. (b) The desired output with a regular-grid coordinate system, including $398 \times 62 = 24676$ source points.

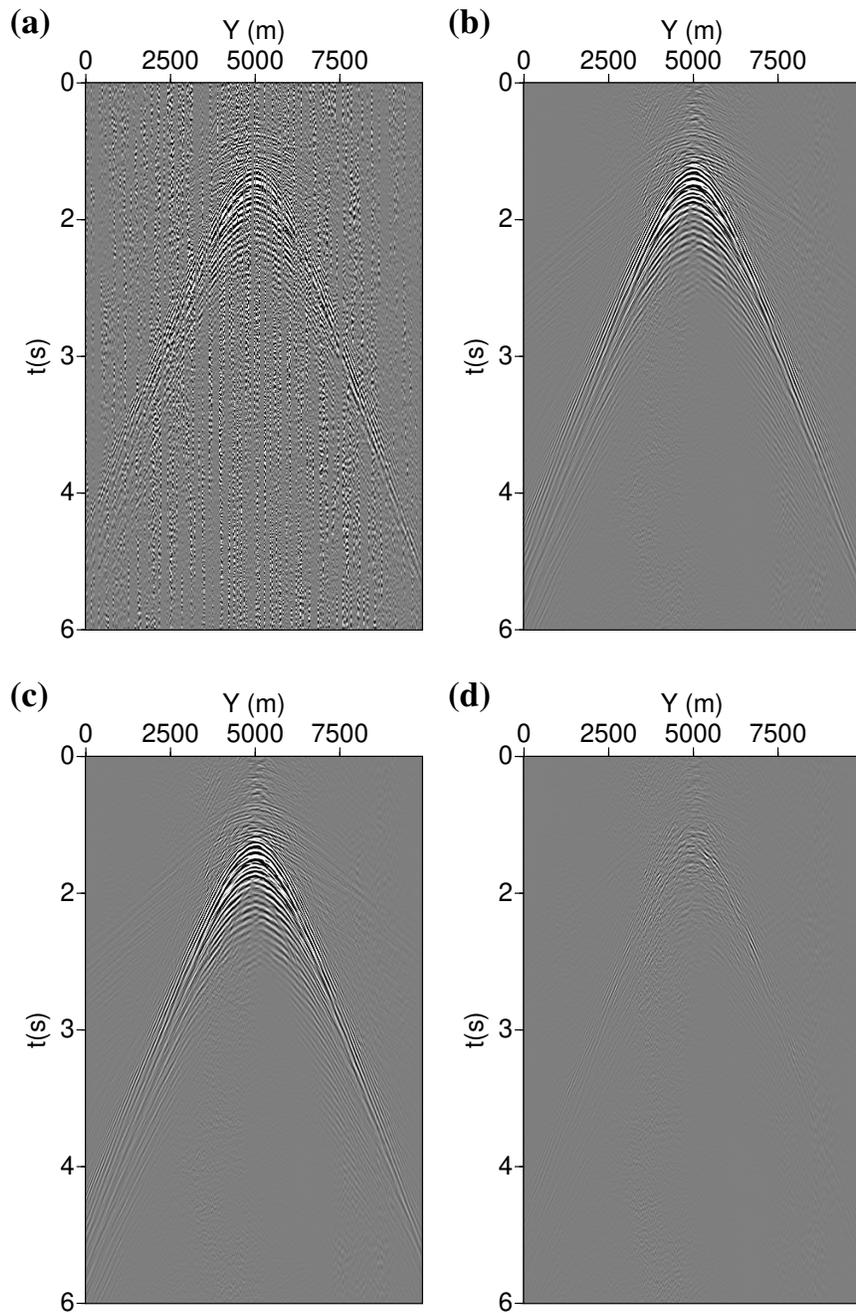


Figure B.4: Real data example. (a) Observed data after binning. (b) Result with FMSSA deblending and binning. (c) Result with I-FMSSA deblending method. (d) Difference between (b) and (c).

and more computational-efficient projection operator (FMSSA). The projected gradient descent method is employed with the I-FMSSA algorithm to iteratively solve this problem and to guarantee the solution honours the true irregular grid observations. The FMSSA algorithm appropriately replaces the MSSA algorithm to reduce the computational burden by avoiding building Hankel-structure matrices and speeding up anti-diagonal averaging with convolution. Synthetic examples show significant improvement in deblending and reconstruction with the I-FMSSA deblending method when adopted by the sinc Kaiser interpolation operator. In addition, the FMSSA projection operator offers remarkable advantages in computational time. For the field data example, we can fully recover the unblended regular-grid data from the observed data. Again, we cannot precisely evaluate signal leakage performance due to the lack of the ground truth as a reference. This is a common problem for all algorithms for deblending when applied to the field data.