Dimensionality-reduced estimation of primaries by sparse inversion

Bander Jumah and Felix J. Herrmann
Department of Earth and Ocean sciences, University of British Columbia,
Vancouver, Canada

ABSTRACT

Data-driven methods—such as the estimation of primaries by sparse inversion—suffer from the 'curse of dimensionality' that leads to disproportional growth in computational and storage demands when moving to realistic 3D field data. To remove this fundamental impediment, we propose a dimensionality-reduction technique where the 'data matrix' is approximated adaptively by a randomized low-rank factorization. Compared to conventional methods, which need for each iteration passage through all data possibly requiring on-the-fly interpolation, our randomized approach has the advantage that the total number of passes is reduced to only one to three. In addition, the low-rank matrix factorization leads to considerable reductions in storage and computational costs of the matrix multiplies required by the sparse inversion. Application of the proposed method to synthetic and real data shows that significant performance improvements in speed and memory use are achievable at a low computational up-fron cost required by the low-rank factorization.

INTRODUCTION

Demand for oil and gas is increasing rapidly while new discoveries are more difficult to make because most of the relatively easy to find reservoirs are being depleted. This development combined with high oil prices and the decline of conventional oil reserves are the driving forces behind continued efforts of the oil and gas industry towards unconventional and more difficult to find and produce reservoirs. In order to achieve this ambition, state-of-the-art technologies, such as high-density, wide-azimuth seismic recording and imaging, are now being utilized to generate higher resolution images of the earth's subsurface.

Unfortunately, these new technologies generate enormous volumes of 3D data that require massive amounts of computational resources to store, manage, and processes. For example, it is nowadays not unusual to conduct seismic surveys that gather one million traces per square mile, which is a significant increase compared the 10,000 traces that were collected traditionally for this area. This development not only makes the acquisition costly but it also makes processing these massive data volumes extremely challenging. These challenges become particularly apparent in the case of data-driven seismic methods, e.g. Surface-Related Multiple Elimination (SRME, Berkhout and Verschuur, 1997) and Estimation of Primaries by Sparse Inversion (EPSI, van Groenestijn and Verschuur, 2009; Lin and Herrmann, 2011, 2012). These methods are known to be compute intensive because they rely on multi-dimensional convolutions that translate into massive dense matrix-vector multiplies, and therefore, suffer from the "curse of dimensionality", where the size of the data volume increases exponentially with the and survey area and desired resolution. For example, in 3D a data set with of 1000 sources and receivers results in multiplications for each frequency of dense matrices of size $1000^6 \times 1000^6$. Needless to say these matrices can not be stored, are expensive to apply, and often require on-the-fly interpolation because acquired data is nearly always incomplete.

Working with massive data volumes also creates communication bottlenecks that form a major impediment for iterative methods that require multiple passes through all data. For this reason, current-day seismic data processing centers spend approximately the same amount of time on running SRME as on migration and this explains the slow adaption of EPSI by industry this technique requires more passes through the data.

We present a dimensionality-reduction technique that is aimed at limiting the number of passes over the data (including on-the-fly-interpolation) to one to three while reducing the memory imprint, accelerating the matrix multiplications, and leveraging parallel capabilities of modern computer architectures. A low-rank matrix factorization with the randomized singular-value decomposition (SVD, Halko et al., 2011) lies at the heart of our method. We use this factorization to approximate the action of monochromatic 'data matrices', whose columns are given by seismic shot records. For early application of SVDs to data matrices, we refer to Minato et al. (2011). By selecting the rank of each monochromatic data matrix adaptively, we are able to approximate matrix multiplies with a controllable error. Because the randomized SVD only requires the action of the data matrix on some set of random vectors, our algorithm can work with existing code bases for SRME.

A key step in the randomized SVD is formed by matrix probing (Halko et al., 2011; Chiu and Demanet, 2012; Demanet et al., 2012), where information on the range of a matrix is obtained by applying the matrix on a small set of random test vectors. The number of these vectors depends on the rank of the matrix, which in turn determines the degree of dimensional reduction and speedup yielded by the low-rank approximation. While standard SVDs are of limited value to large-scale problems, the proposed randomized SVD is well suited to computer architectures that can do fast matrix multiplies but that have difficulties moving large amounts of data in and out of memory.

The paper is organized as follows. First, we briefly introduce EPSI by recasting Berkhout's data matrix into a vectorized form, which makes it conducive to curvelet-domain sparsity promotion. Next, we identify that the matrix multiplies are the dominant cost and we show that these costs can be reduced by replacing the data matrix for each frequency by a low-rank factorization with SVDs. To get better accuracy, we propose an adaptive scheme that selects the appropriate ranks for each data matrix depending on their spectral norm. Next, we introduce the randomized SVD based on matrix probing. This technique allows us to carry out the SVD on large systems. Because data often has missing shots, we also discuss how matrix probing can be extended so that it no longer relies on full sampling or on-the-fly interpolation but instead can work with data with missing shots directly. We show for either case that matrix probing leads to a significant reduction in the number passes through data not only for the calculation of the factorizations but also for the iterative solution of EPSI. To address the increase in rank with frequency, we introduce Hierarchically Semi-Separable Matrix Representation (HSS, Lin et al., 2011) matrices. Finally, we conclude by performing tests of our method on synthetic and real seismic lines.

THEORY

Estimation of Primaries by Sparse Inversion (EPSI) proposed by van Groenestijn and Verschuur (2009) is an important new development in the mitigation of surface-related multiples. As opposed to conventional multiple removal, where multiples are predicted and subtracted after matching, the surface-related multiples are mapped to the surface-free Green's function by carrying out a sparse inversion. During this inversion, the upgoing wavefield is inverted with respect to the downgoing wavefield. In describing EPSI, we make use of Berkhout's (Berkhout and Pao, 1982) detail-hiding monochromatic matrix notation (see Figure 1), where each monochromatic wavefield is arranged into a matrix with columns and rows representing common-shot/common-receiver gathers, respectively. Throughout the paper, we reserve the hat symbol to represent monochromatic quantities, and upper-case variables denote matrices or linear operators. In this notation, multiplication of hatted quantities corresponds to convolution in the physical domain.

Data-matrix formulation and its vectorized form

EPSI describes the relation between the total up-going wavefield $\widehat{\mathbf{P}}$, the surface-free Green's function $\widehat{\mathbf{G}}$, and the downgoing wavefield $\left(\widehat{\mathbf{Q}} - \widehat{\mathbf{P}}\right)$. The latter depends on the source signature $\widehat{\mathbf{Q}}$ and assumes perfect reflection at the surface. Mathematically, the EPSI formulation derives from the following expression van Groenestijn and Verschuur (2009):

$$\widehat{\mathbf{P}} = \widehat{\mathbf{G}} \left(\widehat{\mathbf{Q}} - \widehat{\mathbf{P}}\right). \tag{1}$$

As in Lin and Herrmann (2011), we cast the above relationship into vectorized form—i.e., $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{b}$ represents the upgoing wavefield and $\mathbf{x}$ the unknown surface-free Green's function in some transformed domain. In this formulation, the matrix $\mathbf{A}$ represents the modeling operator that maps primaries to surface-related multiples given the source function, which we assume to be known.

To arrive at this formulation, which includes Fourier and sparsiying transforms, we use the relation $\text{vec}\,(\mathbf{AXB}) = \big(\mathbf{B}^T \otimes \mathbf{A}\big)\,\text{vec}\,(\mathbf{X})$, which holds for arbitrary matrices – of compatible sizes – $\mathbf{A}, \mathbf{X}$, and $\mathbf{B}$. In this expression, the symbol $\otimes$ refers to the Kronecker product and vec stacks the columns of a matrix into a long concatenated vector (matlab's colon operator). Equation 1, can now be rewritten as

$$\Big((\widehat{\mathbf{Q}} - \widehat{\mathbf{P}})_i^T \otimes \mathbf{I}\Big)\,\text{vec}\,\Big(\widehat{\mathbf{G}}_i\Big) = \text{vec}\,\Big(\widehat{\mathbf{P}}_i\Big)\,,\ i = 1 \cdots n_f, \tag{2}$$

where $\mathbf{I}$ is the identity matrix. After inclusion of the temporal Fourier transform ($\mathbf{F}_t = (\mathbf{I} \otimes \mathbf{I} \otimes \mathcal{F}_t)$ with $\mathcal{F}_t$ the temporal Fourier transform), we arrive at the following block-diagonal system:

$$\underbrace{\mathbf{F}_t^* \begin{bmatrix} \Big((\widehat{\mathbf{Q}} - \widehat{\mathbf{P}})_1^T \otimes \mathbf{I}\Big) & & \\ & \ddots & \\ & & \Big((\widehat{\mathbf{Q}} - \widehat{\mathbf{P}})_{n_f}^T \otimes \mathbf{I}\Big) \end{bmatrix} \mathbf{F}_t}_{\mathbf{U}} \underbrace{\begin{bmatrix} \text{vec}\,(\mathbf{G}_1) \\ \vdots \\ \text{vec}\,(\mathbf{G}_{n_t}) \end{bmatrix}}_{\mathbf{g}} = \underbrace{\begin{bmatrix} \text{vec}\,(\mathbf{P}_1) \\ \vdots \\ \text{vec}\,(\mathbf{P}_{n_t}) \end{bmatrix}}_{\mathbf{p}}. \tag{3}$$

In this expression, we use the symbol * to denote the Hermitian transpose or adjoint.

The above vectorized equation is amenable to transform-domain sparsity promotion by defining $\mathbf{A} := \mathbf{US}^*$, where $\mathbf{g} = \mathbf{S}^*\mathbf{x}$ is the transform-domain representation of $\mathbf{g}$ and $\mathbf{S}$ the sparsifying transform. We use a combination of the 2D curvelet, along the source-receiver coordinates, and the 1D wavelet transform along the time coordinates. Using the Kronecker product again, we define $\mathbf{S} = \mathbf{C} \otimes \mathbf{W}$. With these definitions, we relate the sparsifying representation for the surface-free Green's function to the vectorized upgoing wavefield $\mathbf{b} = \text{vec}\,(\mathbf{P})$ via $\mathbf{Ax} = \mathbf{b}$. This relationship forms the basis for our inversion.

Sparse inversion

Solving for the transform-domain representation of the surface-free Green's function $g(t, x_s, x_r)$ with t time, x_s the source and x_r the receiver coordinates, corresponds after discretization to inverting a linear system of equations where the monochromatic wavefields $\{(\widehat{\mathbf{Q}} - \widehat{\mathbf{P}})_i\}_{i=1\cdots n_f}$ and temporal wavefields $\{\mathbf{P}_i\}_{i=1\cdots n_t}$ are related —through the temporal Fourier transform—to the curvelet-wavelet representation of the discretized wavefield vector $\mathbf{g}$ in the physical domain (cf. Equation 3). To control issues related to the null-space of $\mathbf{A}$, we solve this system by promoting sparsity—i.e, we solve

$$\tilde{\mathbf{x}} = \arg\ \min_{\mathbf{x}} \|\mathbf{x}\|_\mathbf{1} \text{ subject to } ||\mathbf{Ax} - \mathbf{b}||_\mathbf{2} \leq \sigma, \tag{4}$$

where σ is the error between the predicted and recorded data.

Solving these optimization problems requires multiple iterations. Each of these iterations are challenging for real applications in 3D because (i) the matrices are dense and extremely large, e.g. for each frequency the data matrix becomes easily $10^6 \times 10^6$ for $n_s = n_r = 1000$ (with n_s the number of sources and n_r the number of receivers). This means that these optimizations require lots of storage and computational resources to carry-out the multiplications; (ii) The collected data volumes are incomplete, which makes it necessary to carry out 'on-the-fly' interpolations that are costly but that have the advantage that the data matrix does not need to be stored and formed explicitly; and (iii) the solvers require multiple evaluations of $\mathbf{A}$, $\mathbf{A}^*$, and possibly $\mathbf{A}^*\mathbf{A}$. Remember, each of these matrix-vector operations include a Fourier, curvelet, and wavelet transforms, and the application of the data matrix. While the transforms can be carried out relatively quickly, the application of the data matrix requires a complete pass over all data during which data needs to be transferred in and out of main memory. In practice, this leads to processing times that are of the same order of migration for a single iteration. To reduce these storage and multiplication costs, we replace the data matrix by a low-rank approximation using Singular-Value Decomposition (SVD).

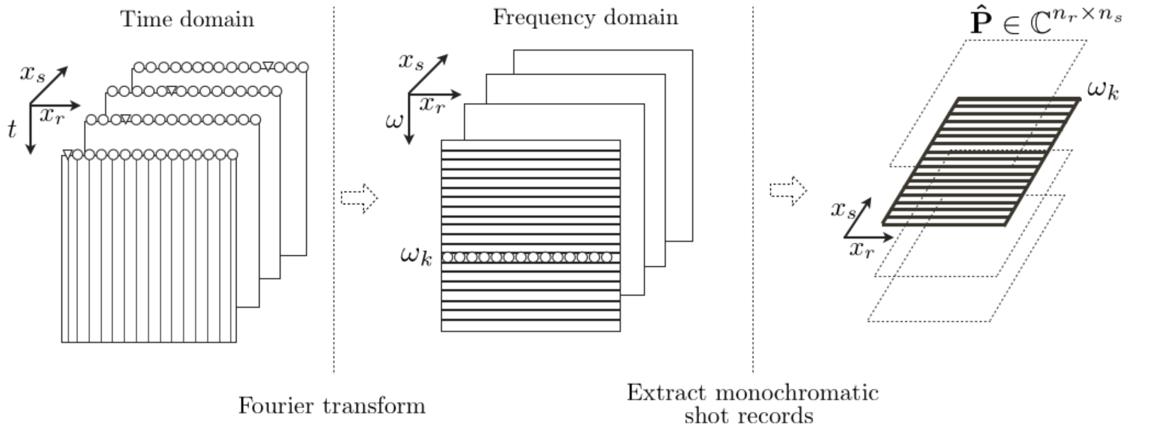

Time domain
Frequency domain
$\hat{\mathbf{P}} \in \mathbb{C}^{n_r \times n_s}$
x_s
x_r
t
ω
ω_k
Fourier transform
Extract monochromatic shot records

Figure 1: Extraction of monochromatic wavefield matrices, by transforming the data from the time domain into the frequency domain via the Fourier transform. Then, for each frequency component a data matrix $\hat{\mathbf{P}} \in \mathbb{C}^{n_r \times n_s}$ is extracted (adapted from Dedem (2002)).

Dimensionality reduction via singular-value decomposition

To limit the storage and multiplication costs, we approximately factorize the data matrix into

$$\widehat{\mathbf{P}}_{n_r \times n_s} \approx \widehat{\mathbf{L}}_{n_r \times k} \widehat{\mathbf{R}}^*_{k \times n_s}, \tag{5}$$

where the symbol $\approx$ refers to an approximation with a controllable error. In this formulation, a data matrix with n_r receivers and n_s sources is factorized into much smaller tall and fat matrices that reduce the computational and memory costs of applying the data matrix. The approximation is low rank if the error is small for $k << \min(n_r, n_s)$.

A special case of a low-rank matrix factorization is the Singular Value Decomposition (SVD) where the data matrix $\widehat{\mathbf{P}}$ is decomposed into three factors, namely

$$\widehat{\mathbf{P}}_{n_r \times n_s} \approx \widehat{\mathbf{U}}_{n_r \times k} \widehat{\mathbf{S}}_{k \times k} \widehat{\mathbf{V}}^*_{k \times n_s}, \tag{6}$$

with the matrices $\widehat{\mathbf{U}} \in \mathbb{C}^{n_r \times k}$ and $\widehat{\mathbf{V}} \in \mathbb{C}^{k \times n_s}$ representing the left and right singular vectors of $\widehat{\mathbf{P}}$. The diagonal matrix $\widehat{\mathbf{S}} \in \mathbb{C}^{k \times k}$ carries the non-negative singular values on its diagonal. Refer to Figure 2 for an example using of this factorization to a monochromatic data matrix with at 40 Hz and with $n_s = n_r = 150$ and $k = 20$. Applying this type of dimensionality reduction to EPSI requires low-rank approximations for data matrices for all frequencies. Since we can not expect the ranks of these matrices to be same for each frequency, we introduce an adaptive scheme to select the proper rank. In Table 1, we summarize the advantages of SVD in relation to multiplication and storage cost for data matrices with $n = n_s = n_r$. If we ignore the cost of computing the SVD factorizations, we see that we can achieve significant reductions in these costs when k is relatively small.

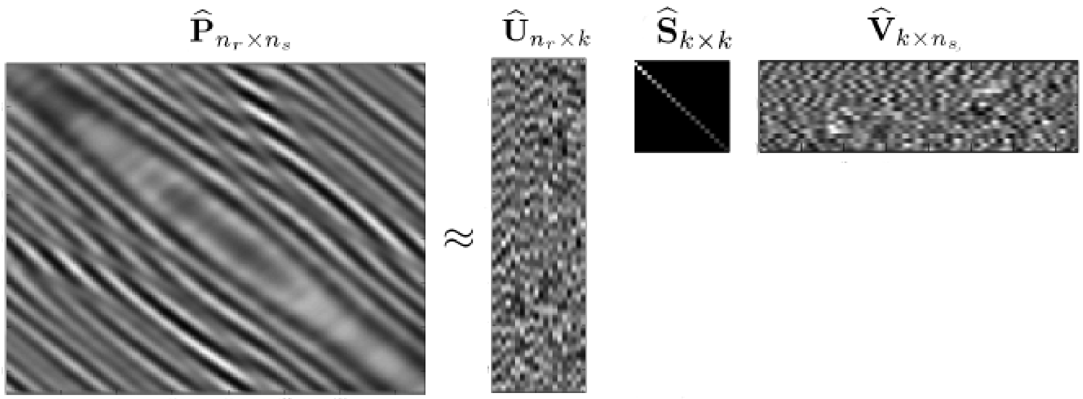

$\widehat{\mathbf{P}}_{n_r \times n_s}$
$\approx$
$\widehat{\mathbf{U}}_{n_r \times k}$
$\widehat{\mathbf{S}}_{k \times k}$
$\widehat{\mathbf{V}}_{k \times n_s}$

Figure 2: Matrix factorization by SVD for a frequency slice $\widehat{\mathbf{P}} \in \mathbb{C}^{n_r \times n_s}$ on the left, and its SVD decompositions on the right. The matrix $\widehat{\mathbf{U}}_{n_r \times k}$ and $\widehat{\mathbf{V}}^*_{k \times n_s}$ contain the left and right singular vectors of $\widehat{\mathbf{P}}$, respectively. The diagonal matrix $\widehat{\mathbf{S}}_{k \times k}$ contains the singular values of $\widehat{\mathbf{P}}$ on its diagonal.

	SVD approximation	Regular method
Matrix-matrix multiplication	$\mathcal{O}(kn(2n+k))$	$\mathcal{O}(n^3)$
Storage	$\mathcal{O}(k(2n+1))$	$\mathcal{O}(n^2)$

Table 1: Advantages of using low-rank approximations via SVD in terms of matrix-matrix multiplications and storage requirements.

Adaptive low-rank approximation

To successfully apply our dimensionally-reduction technique, we need to know the behavior of the singular values as a function of the frequency so we can strike a balance between the rank for the approximation and the quality of the approximation. For data matrices, with singular values that do not vanish the quality of approximation improves for increasing k. Moreover, we would expect singular values concentrate in the seismic band and to be relatively small outside this band. As we can see from Figure 3 this is indeed the case and we also observe that the singular values decay relatively quickly facilitating an accurate approximation as long as we select the rank for the approximation adaptively.

Since the monochromatic wavefields in the data matrices are applied as linear operators (cf. Equation 3) in the EPSI modeling operator, it is natural to use the spectral norm to quantify the error. This norm relates the energy of a vector that is the result of a matrix-vector product to the energy of an arbitrary input vector $\mathbf{x}$—i.e., we define spectral norm for an arbitrary matrix $\mathbf{A}$ as

$$\|\mathbf{A}\|_S = \max_{x \neq 0} \frac{\|\mathbf{Ax}\|_2}{\|\mathbf{x}\|_2}. \tag{7}$$

In this expression, $\|\mathbf{x}\|_2 = \sqrt{\sum_{k=1}^{n} |x_k|^2}$ defines the ℓ_2-norm. The spectral norm corresponds to the largest (the first) singular value of $\mathbf{A}$.

To distribute a given total rank budget—i.e., the sum of selected ranks for each frequency—across all data matrices we compute the spectral norms as a function of the frequency. Given a user defined subsampling ratio $\delta = K'/K$ with $K = n_s \times n_f$ and $K' \ll K$ the total rank budget, we select the ranks for each frequency proportional to the spectral norms. We included Figure 4 to illustrate the importance of this adaptive strategy by comparing the action of the EPSI prediction operator on the surface-free Green's function for the compete data matrix, and low-rank factorizations with and without adaptation for $\delta = 1/12$. These results clearly show that selecting the ranks adaptively leads to visually improved results (juxtapose Figures 4(b) and 4(c)). This improvement can be attributed to the fact that the energy of the upgoing wavefield in concentrated in the seismic frequency band. Table 2 confirms the relation between the spectral norms and the SNR for the predicted multiples as a function of the subsampling ratio δ for a synthetic dataset. (See the results section at the end of the paper for a description of this data set.) The relative spectral norms in this table are given the sum of the relative spectral norms defined by $\frac{\|\mathbf{A}-\mathbf{A}_k\|_S}{\|\mathbf{A}\|_S}$ with $\mathbf{A}_k$ the k-rank approximation of $\mathbf{A}$.

With the low-rank SVD approximation, we are able to approximate the action of the multiple prediction operator with reasonable accuracy for substantial subsampling ratios. This leads to a corresponding reduction in computational and storage costs, which becomes important when scaling EPSI to 3D. However, our approach comes with an additional pre-processing step to perform the SVD. While efficient implementations of the SVD exist, they typically require at least k passes through al data,

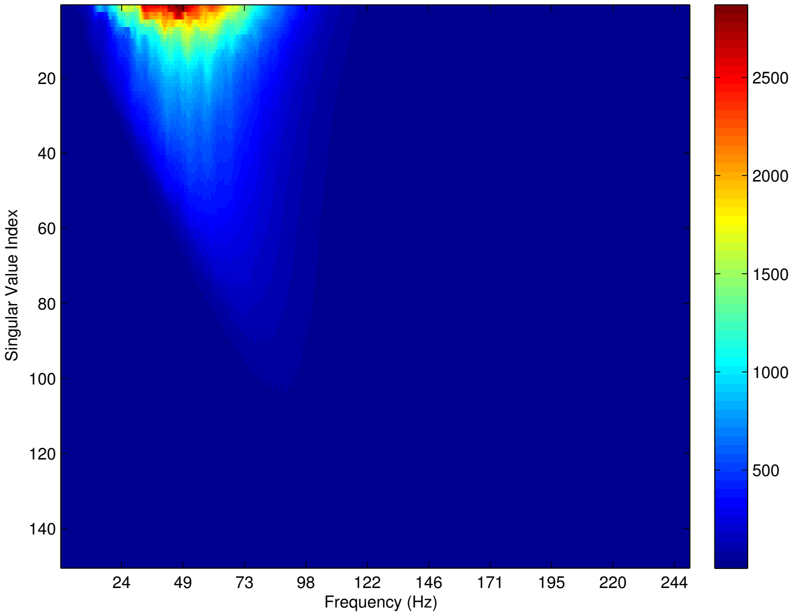


Figure 3: Singular values for data matrices as a function of the frequency. As expected, the largest singular values are concentrated within the seismic band.

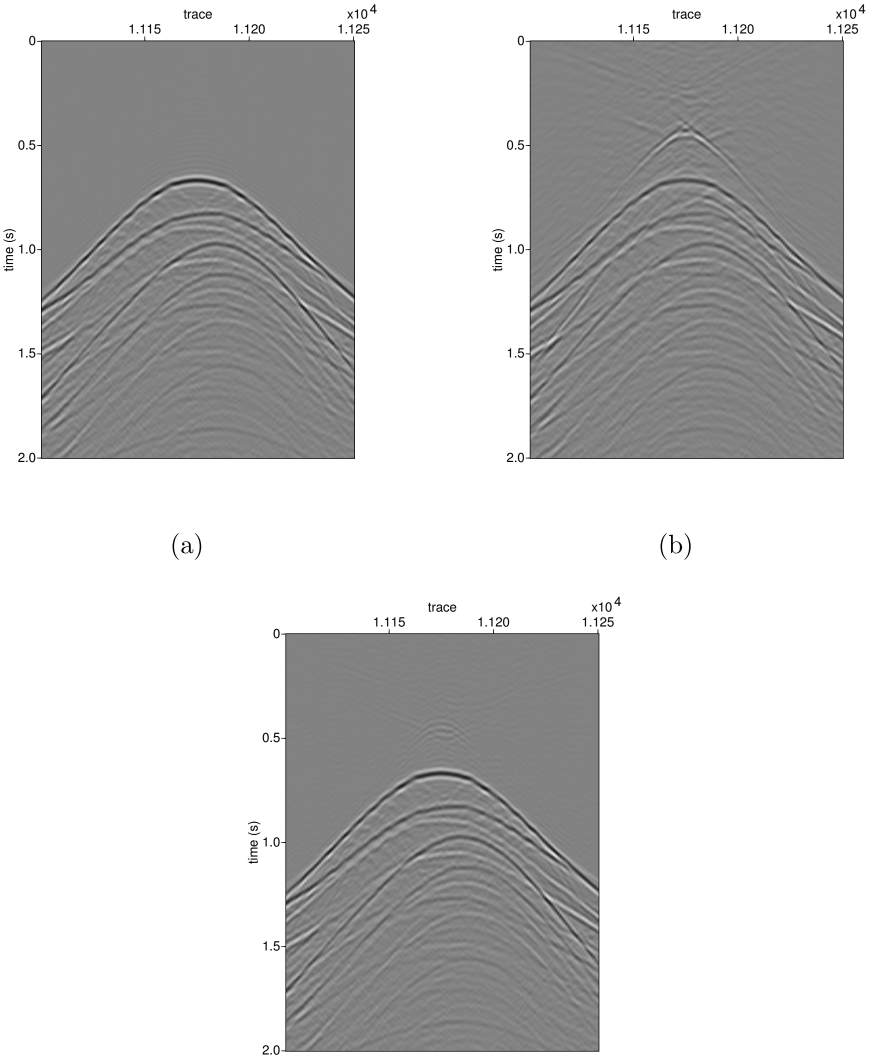
which becomes prohibitively expensive for large data sets (Halko et al. (2011)). To meet this challenge, we adapt randomization techniques that are specifically designed to leverage current computer architectures that are good in applying multiplications in parallel but bad in moving data in and out of memory.

RANDOMIZED SINGULAR-VALUE DECOMPOSITION

To overcome shortcomings related to memory handling and parallelism of modern computing architectures, we employ recent developments in randomized dimensionality reduction as proposed by Halko et al. (2011). These approaches have the advantage that they require fewer passes over all the data while also being suitable for parallel implementation. First, we briefly discuss the main steps that are part of this new procedure, followed by what to do when the singular values decay slowly and how to handle cases where sources are missing.

General framework

The randomized SVD algorithm (Halko et al., 2011) is composed of two stages, namely matrix probing capturing the matrix action on a small number of random vectors, followed by an orthogonalization, and a SVD on the reduced system.



(a)

(b)

(c)

Figure 4: Adaptive versus non-adaptive rank selection. (a) prediction of multiples with the full data matrix,(b) prediction with the low-rank factorization for $\delta = 1/12$, and (c) the prediction with adaptive rank selection.

Sampling ratio δ	1/2	1/5	1/8	1/12
Multiplication speed-up factor	1.4	2.2	4.7	6.5
Memory-reduction factor	1/3	1/2	3/4	6/7
SNR dB	29	26	19	11
Relative spectral norm ($\times 10^{-4}$)	4	12	25	63

Table 2: Quality of multiple prediction as a function of the subsampling ratio for a synthetic data set with $n_s = n_r = 150$, and $n_t = 512$. The results confirm the relationship between the spectral norm and SNR for the multiple prediction.

Stage A: matrix probing. During this stage, we randomly sample the action of the data matrix by applying this matrix, possibly in parallel, to a small number of random vectors—i.e.,

$$\mathbf{Y} = \widehat{\mathbf{P}}\widehat{\mathbf{W}}. \tag{8}$$

Here, $\widehat{\mathbf{W}}_{n_r\times(p+k)}$ is a tall random zero-mean Gaussian matrix. The number of columns of this matrix is given by the rank k of the input data matrix $\widehat{\mathbf{P}}$ plus an oversampling by p, typically set to $5-10$. This operation is illustrated in Figure 5 and corresponds to forming $p+k$ amplitude encoded supershots by summing over all sequential shots with random Gaussian weights (see e.g. van Leeuwen et al., 2011).

As long as the data matrix has a rank k or smaller this procedure obtains sufficient information to compute the SVD within a prescribed accuracy. To be more specific, the action on the random vectors gives us access to the range of the data matrix, which is spanned by the left vectors $\mathbf{Q} \in \mathbb{C}^{n_s\times(k+p)}$ of the QR factorization of $\mathbf{Y}$. With these vectors, we form

$$\mathbf{B}_{n_s\times(k+p)} = \mathbf{Q}_{n_s\times(k+p)}\widehat{\mathbf{P}}_{n_r\times n_s}, \tag{9}$$

which is input to stage B, where we carry out the SVD on this reduced system. The operations that form stage A are summarized in Algorithm 1.

```
Algorithm 1 Randomized SVD
Input: $\widehat{\mathbf{P}}_{n_r\times n_s}$, a target rank $k$ and the over-sampling parameter $p$
Output Orthonormal matrix $\mathbf{Q}$ whose range approximates the range of $\widehat{\mathbf{P}}$
  1. Draw a random Gaussian matrix $\widehat{\mathbf{W}}_{n_s\times(k+p)}$.
  2. Form $\mathbf{Y}_{n_r\times(k+p)} = \widehat{\mathbf{P}}\widehat{\mathbf{W}}$.
  3. Construct the orthonormal matrix $\mathbf{Q}_{m\times(k+p)}$ by computing QR factorization of $\mathbf{Y}$.
```

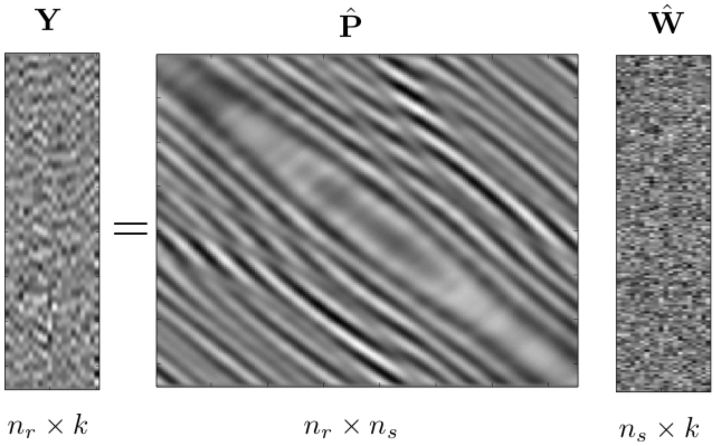


Figure 5: Supershots created by randomized superposition of sequential shots.

Stage B: computation of the low-rank factorization. The output of stage A corresponds to a randomized dimensionality reduction capturing the action of the data matrix from which we can now calculate the SVD with the advantage that we are working with a much smaller system. Following Halko et al. (2011), we compute

$$\mathbf{B}_{n_r \times k} = \widetilde{\mathbf{U}}_{k \times k} \widehat{\mathbf{S}}_{k \times k} \widehat{\mathbf{V}}^*_{k \times n_s}, \tag{10}$$

from which we subsequently calculate the left singular vectors using the following expression

$$\widehat{\mathbf{U}}_{n_r \times k} = \mathbf{Q}_{n_r \times k} \widetilde{\mathbf{U}}_{k \times k}. \tag{11}$$

Figure 6, describes how the left singular vectors $\widehat{\mathbf{U}}$ of a data matrix $\widehat{\mathbf{P}}$ are computed using the orthonormal matrix $\mathbf{Q}$.

During this second stage, we factored the data matrix by carrying out the SVD on the dimensionality reduced system. The advantage in this approach is that the full data matrix only needs to be accessed two times namely once for the action on the random vectors to compute $\mathbf{Y}$ and once to form $\mathbf{B}$. This is a significant improvement compared to the k passes over the data required by the conventional SVD (see Halko et al., 2011). However, the proposed Algorithm 1 is only appropriate for matrices that have low rank or rapidly decaying singular values. Unfortunately, the algorithm will perform poorly when approximating matrices that exhibit slow decay for their singular values (Halko et al., 2011). In that case, singular vectors associated with the small singular values are known to interfere with the approximation and this leads to poor quality of the approximation. To reduce these interferences, the decay for the singular singular values can be improved by using the power method (Halko et al., 2011), which replaces Equation 8 by

$$\mathbf{B} = (\widehat{\mathbf{P}}\widehat{\mathbf{P}}^*)^q \widehat{\mathbf{P}}\widehat{\mathbf{W}}, \tag{12}$$

with q the order of the power. As a result of raising the matrix to the q^{th} power, the singular values decay faster as illustrated in Figure 7. While this procedure reduces the interference, it comes at a cost of $(q+1)$ additional passes over the data. Fortunately, experience has shown that setting $q = 1$, 2 is usually sufficient. Because the cost of probing with Gaussian vectors may become expensive, we replace the Gaussian matrix $\widehat{\mathbf{W}}$ by fast Fourier-based phase-encoding. For more detail on the power method and phase encoding, we refer to Halko et al. (2011); Herrmann et al. (2009).

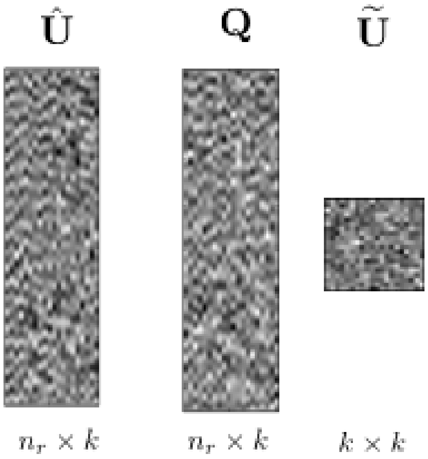


Figure 6: Approximation of the left singular vectors $\widehat{\mathbf{U}}$ of a data matrix $\widehat{\mathbf{P}}$ using the orthonormal matrix $\mathbf{Q}$.

Incomplete data

Whilst the proposed method entails a significant reduction of the problem size and reliance on prohibitively large computer resources, our randomized approach requires full sampling, which excludes acquisition with missing sources. Since data is nearly always sampled incompletely this may not be an issue because current 3D implementations of surface-related multiple elimination (SRME) interpolate data on the fly. However, these interpolations are expensive and require extensive storage.

To overcome this problem, we propose to change the matrix probing with Gaussian vectors (cf. Equation 8) by a probing with randomly selected columns from the Dirac basis. This corresponds to applying the randomized SVD on data with randomly selected shots and negates the need to interpolate the data during the matrix probing. Remember, that the compute $\mathbf{B}$ (cf. Equation 9) requires one full pass over the data including interpolation.

Matrix probing relies on the fact that Gaussian matrices capture the range of a matrix. This can be understood because Gaussian vectors are incoherent with vectors of any arbitrary orthogonal basis, including bases spanned by singular vectors. Following the theory of compressive sensing (Candès et al., 2006; Donoho, 2006), we

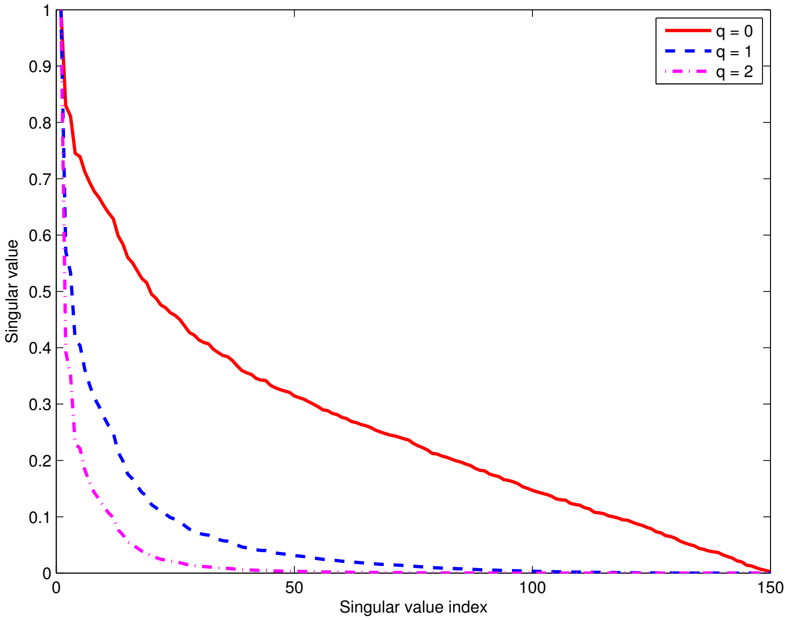


Figure 7: Effect of the power iteration on the decay of singular values for a frequency slice $\mathbf{P}_{150\times150}$. Remember, Algorithm 1 is equivalent to $q = 0$.

introduce the mutual coherence between the columns of bases $\mathbf{U}$ and $\mathbf{R}$ as

$$\mu(\mathbf{U},\mathbf{R}) = \max_{1<i,j<n} \frac{<\mathbf{u}_i, \mathbf{r}_j>}{\|\mathbf{u}_i\|_2 \|\mathbf{r}_j\|_2}. \tag{13}$$

In this expression, $\mathbf{u}_i$, $\mathbf{r}_j$ correspond to the j^{th} and i^{th} columns of $\mathbf{U}$ and $\mathbf{R}$, respectively. As we have seen in related work on full-waveform inversion (Li et al., 2011), Gaussian test vectors can be replaced by other vectors as long as the mutual coherence remains low. For this purpose, we compute the mutual coherence as a function of frequency between the singular vectors and randomly selected columns of the Fourier and Dirac bases. The results of this exercise are summarized in Figure 8 and show that the Gaussian vectors have as expected the lowest mutual coherence followed by the Dirac and Fourier bases. From these results, we conclude that we can work with randomly missing shots instead of randomly superimposed simultaneous shots that require full acquisition. Working with Fourier is not an option because the coherence is too high.

Examples

To test the performance of our randomized SVD, we consider two examples, namely a data matrices with fast and slow decaying singular values. Physically, this corresponds to matrices with a low frequency (5 Hz) or a high frequency (100 Hz). Following, Halko

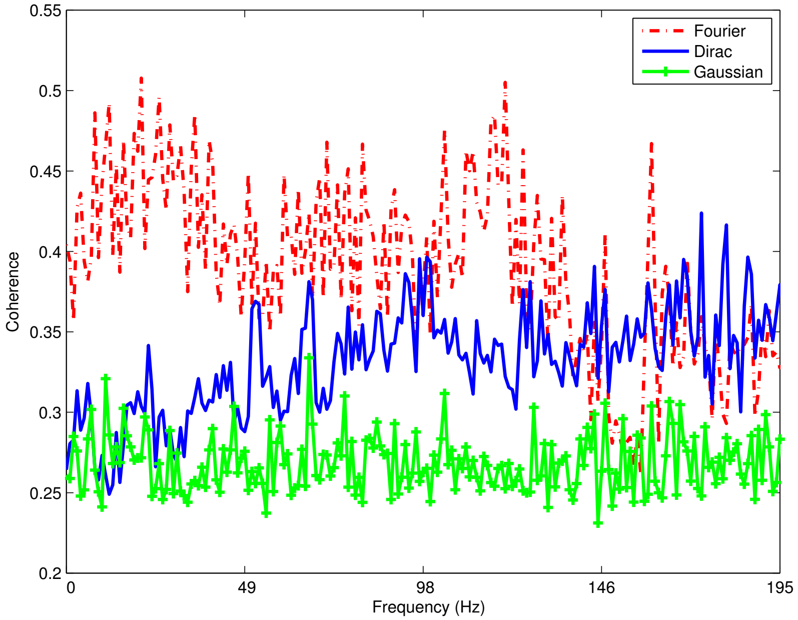


Figure 8: Coherence of the singular vectors of the data matrix with Fourier, Dirac, and Gaussian bases for all frequencies.

et al. (2011) we compute the error for the k-rank approximation via

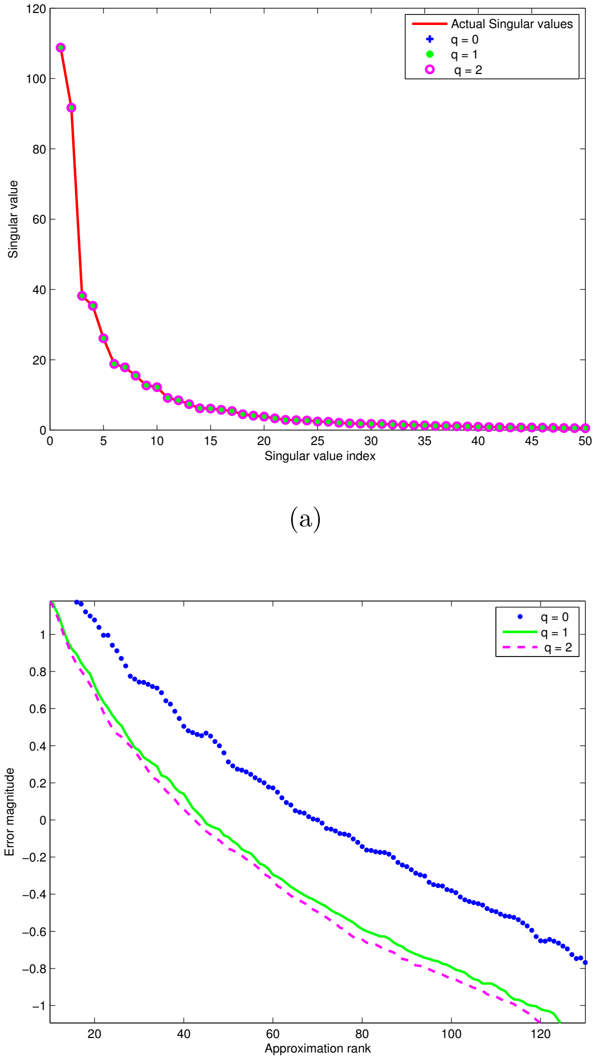
$$e_k = \|(\mathbf{I} - \mathbf{Q}\mathbf{Q}^*)\widehat{\mathbf{P}}\|_S. \tag{14}$$

Fast decay: In this example, we apply the randomized SVD on a frequency slice at 5 Hz from our synthetic dataset. Figure 9(a) contains the estimated singular values by matrix probing with Gaussian vectors for $q = 0$, 1, 2, and with $k = 50$. From this plot, we observe that the singular values are estimated correctly for each q. We also observe from Figure 9(b) that the error given by the above equation is reduced significantly if we make at least one additional pass over the data—i.e., $q > 0$.

Slow decay: In this example, we apply the randomized SVD on the same data set but at a high frequency of 100 Hz. In this case, we need an additional pass over the data to get the accurate estimates for the singular values (see Figure 10(a). Unfortunately, Figure 10(b)) shows that the errors given by Equation 14 continue to decay slowly for increasing q. However, there is significant improvement compared to standard probing ($q = 0$).

Different samplings: Finally, we compare the errors given by Equation 14 for matrix probing with Dirac, encoding with the subsampled Fourier transform (SRFT, Halko et al., 2011)), and Gaussian. The results are summarized in Figures 11(a)–11(b), and indicate that changing the sampling method between Gaussian, fast encoding, and Dirac has limited effect especially for $q = 2$.

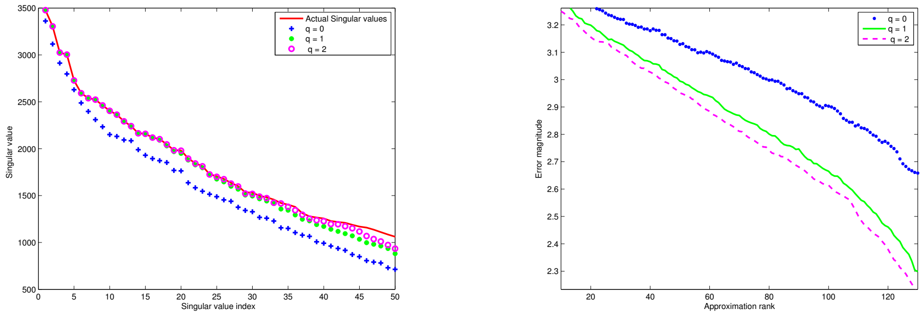
Unfortunately, in most cases we are only able to get satisfactory approximations when we allow ourselves to make additional passes over the data. This is a consequence of the fact that the underlying matrices are essentially not low rank in particular for higher frequencies where the energy tend to be located along the diagonal of the matrix. By a multiscale decomposition, we try to overcome this shortcoming.



(a)

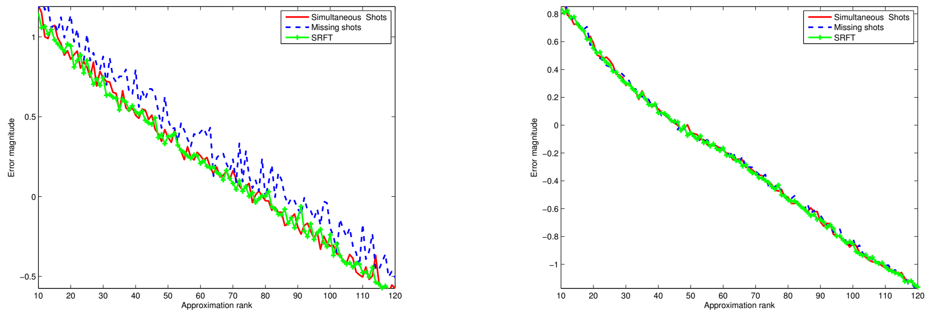
(b)

Figure 9: Decay of the singular values for a single frequency slice at 5 Hz for $n_r = n_s = 150$. (a) Approximation of the singular values and (b) the behavior of the spectral approximation error for different q.



(a) (b)

Figure 10: Decay of the singular values for a single frequency at 100 Hz for $n_r = n_s = 150$. (a) Approximation of the singular values, and (b) the behavior of the spectral approximation error.



(a) (b)

Figure 11: Errors of the randomized SVD for different sampling methods (Gaussian, phase-encoded Fourier (SRFT), and Dirac (randomly-selected sequential shots), (a) approximation error using $q = 0$, and (b) approximation error for $q = 2$.

HIERARCHICALLY SEMI-SEPARABLE MATRIX REPRESENTATION

As we have seen in the previous section, data matrices become higher rank at higher frequencies because the number of oscillations increase while the energy tends to focus more around the diagonal. The latter property has to do with the increased curvature of seismic events as the frequency increases. While some recent theoretical work has been done to address this issue by including directionality in the formulation (Engquist and Ying, 2010), we rely on the Hierarchically Semi-Separable Matrix Representation (HSS, Chandrasekaran et al., 2006) in combination with the randomized SVD along the lines of recent developments by Lin et al. (2011).

HSS matrices provide a way to represent high-rank structured dense matrices in terms of low-rank submatrices with the aim of reducing the cost of linear algebraic operations, e.g. reducing the cost of matrix-vector products from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$. As part of the HSS representation, matrices are recursively repartitioned into high-rank diagonal and low-rank off-diagonal matrixes. SVDs are carried on the off diagonal submatrices while the high-rank diagonal submatrices are recursively partitioned to some user defined level, which depends on the desirable compression and accuracy. With this decomposition, HSS is able to carry out fast matrix operations (Chandrasekaran et al., 2006).

A 2×2 block partitioning of an input matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{1;1,1} & \mathbf{A}_{1;1,2} \\ \mathbf{A}_{1;2,1} & \mathbf{A}_{1;2,2} \end{pmatrix},$$

where the subscripts represent: the partition level, the row number, and the column number, respectively. For each recursion, the-off diagonal submatrices are decomposed into their low-rank approximations by using the randomized SVD with power iterations. After the first iteration, we can write,

$$\mathbf{A} = \begin{pmatrix} \mathbf{D}_{1;1,1} & (\mathbf{USV}^*)_{1;1,2} \\ (\mathbf{USV}^*)_{1;2,1} & \mathbf{D}_{1;2,2} \end{pmatrix},$$

with the first subscript denoting the subdivision level and the second pair of subscripts indicating the subblock. The matrices $\mathbf{D}$ are the diagonal submatrices and the factorization $\mathbf{USV}^*$ correspond to the singular value decompositions of the off-diagonal submatrices. At the next iteration, this nested partitioning again divides the two high-rank diagonals into 2×2 blocks yielding

$$\mathbf{A} = \begin{pmatrix} \begin{pmatrix} \mathbf{D}_{2;1,1} & (\mathbf{USV}^*)_{2;1,2} \\ (\mathbf{USV}^*)_{2;2,1} & \mathbf{D}_{2;2,2} \end{pmatrix} & (\mathbf{USV}^*)_{1;1,2} \\ (\mathbf{USV}^*)_{1;2,1} & \begin{pmatrix} \mathbf{D}_{2;1,1} & (\mathbf{USV}^*)_{2;1,2} \\ (\mathbf{USV}^*)_{2;2,1} & \mathbf{D}_{2;2,2} \end{pmatrix} \end{pmatrix}.$$

For further details on HSS, we refer the reader to Chandrasekaran et al. (2006). Because HSS handles the high-rank parts of the matrix by the recursive partitioning, we end up with an algorithm that only requires a few passes over the data. We achieve this by carrying out the matrix probing on each low-rank submatrix separately while leaving the high-rank diagonal matrices at the fines level alone. This approach has the advantage of increasing the decay of the singular for the low-rank submatrices, which is beneficial to the matrix probing. In addition, the algorithm does not need to store the singular vectors for the coarse low-rank decompositions. Instead, the algorithm computes the singular vectors at the lower level of the decomposition recursively from the singular vectors at the finer level Chandrasekaran et al. (2006). To determine the proper rank for the matrix probing, we use Algorithm 4.2 from Halko et al. (2011).

Example

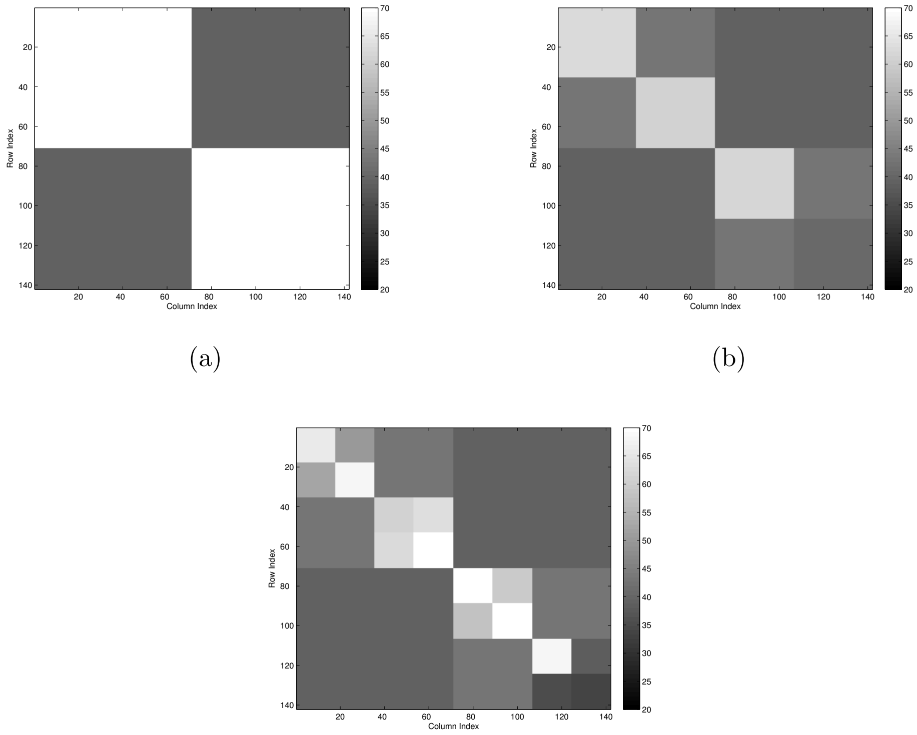
To demonstrate the effectiveness of the HSS representation, we consider a monochromatic frequency slice at 100 Hz, which we approximate with a three-level HSS representation in combination with our randomized SVD with adaptive rank selection. Before applying this algorithm, we first verify the anticipated behavior of the HSS blocks in Figure 12, which shows that the rank of the off-diagonal subblocks is indeed lower than the rank of the diagonal subblocks. This justifies the use of HSS on high-frequency data matrices. As we can see from Figure 13, the HSS-based factorization attains a better approximation, which is reflected in the spectral norms shown in the grey-scale plots.

APPLICATION TO SYNTHETIC AND REAL DATA

To establish the performance of our method, we compare the output of EPSI (Lin and Herrmann, 2011) with the output of EPSI using low-rank approximations for the data matrix. Depending on the ratio between the largest singular value of a particular data matrix and the largest singular amongst all data matrices, we either proceed by applying the randomized SVD or we first apply the HSS partitioning prior to the computation of the randomized SVDs on the off diagonals. We conduct two experiments and we fix the subsampling ratios to $\delta = [1/2, 1/5, 1/8, 1/12]$.

Synthetic data

In this example, we use data modeled from a velocity model that consists of a high-velocity layer, which represents salt, surrounded by sedimentary layers and a water bottom that is not completely flat (see Herrmann et al., 2007). Using an acoustic finite-difference modeling algorithm, 150 shots with 150 receivers are simulated on a fixed receiver spread. A shot record for the first 2 s with surface-related multiples is plotted in Figure 14. For a plot of the singular values of this synthetic data sets, refer to Figure 3.

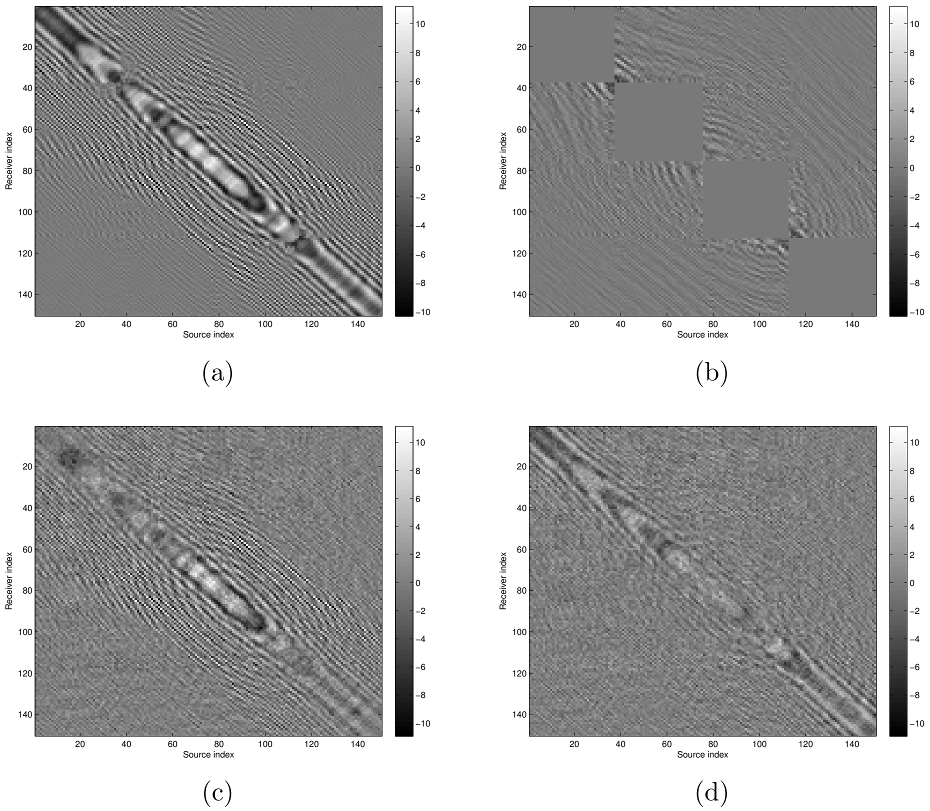


(a)

(b)

(c)

Figure 12: HSS partitioning of a frequency slice at 100 Hz. The grey scales correspond to the rank of the sub matrix (white is high-rank and black is low-rank) (a) First level of HSS partitioning; (b) second level; (c) third level.



(a)

(b)

(c)

(d)

Figure 13: Low-rank factorization of a frequency slice at frequency of 100 Hz and $n_s = n_r = 150$. (a) HSS approximation (relative spectral norm = 0.22), (b) the difference between the HSS approximation and the original data matrix, (c) SVD approximation of the same frequency slice (relative spectral norm = 0.43) and (d) the difference of the SVD approximation with the original data matrix.

Results of our adaptive rank selection on this data set are plotted in Figure 15 and show that most of the available rank budget is assigned to data matrices with frequencies that lie in the seismic band. For the low subsampling ratios, we distributed part of the rank budget amongst all frequencies equally. With these ranks, we compute the low-rank factorizations with randomized SVD for $q = 2$ and $\delta = 1/12$. We use HSS representations for the matrices that have the highest spectral norm. Given the low-rank factorization, we solve the EPSI problem (cf. Equation 4) and the results are plotted in Figure 16. Comparison between the EPSI result with the full data matrix and its low-rank approximation shows that our method is capable of estimating the surface-free Green's function. Despite the fact that we reduced the size of the system significantly, we are to get a satisfactory result, which is confirmed by the difference plot in Figure. 16(c) that contains relatively little energy. Also remember that we only need three passes over all data. The EPSI itself no longer involves passes through the data by virtue of the low-rank approximation.

For completeness, we also include table 3 with SNRs for dimensionality-reduced EPSI experiments carried out for $\delta = [1/2, 1/5, 1/8, 1/12]$. As expected, the SNRs computed with respect to the output of EPSI carried out with the full data matrix, show a decrease in SNR for decreasing subsampling ratios.

Gulf of Suez data

To test the viability of our method on real data, we carried out a series of similar experiments for $q = 2$ on a Gulf of Suez dataset for the same subsampling ratio. A shot record for the first 2 s of this data set is plotted in Figure 17. The singular values and assigned ranks are included in Figure 18. Because this dataset is more broad band, the ranks are assigned over a wider range of frequencies. As before, we carry out EPSI for the complete and low-rank approximated data matrices. The results are shown in Figure 19. While both results are comparable, the difference plots contain some energy loss for regions in the shot record that have high curvature. This is expected because high-curvature events lead to high ranks. Because real data is more complex, the SNRs listed in table 4 are not as high as for the synthetic example.

Subsample ratio δ	1/2	1/5	1/8	1/12
Recovery error (dB)	44	30	18	13
Speed up (×)	2	5	8	12

Table 3: Results of estimating the surface-free Green's function using the different subsampling ratios (synthetic dataset).

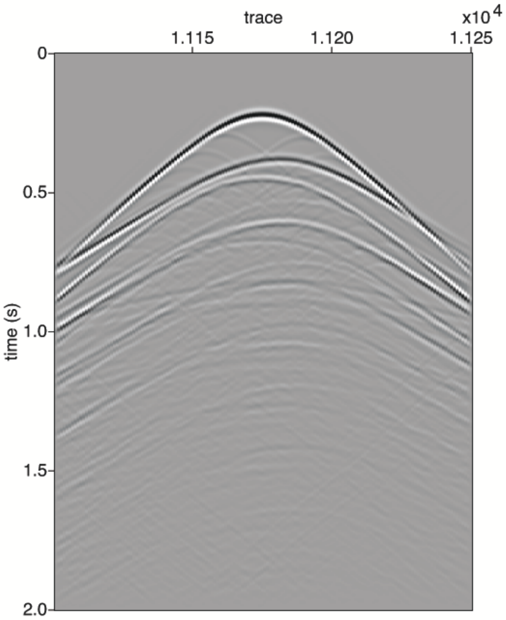


Figure 14: Shot gather from synthetic dataset including surface-related multiples.

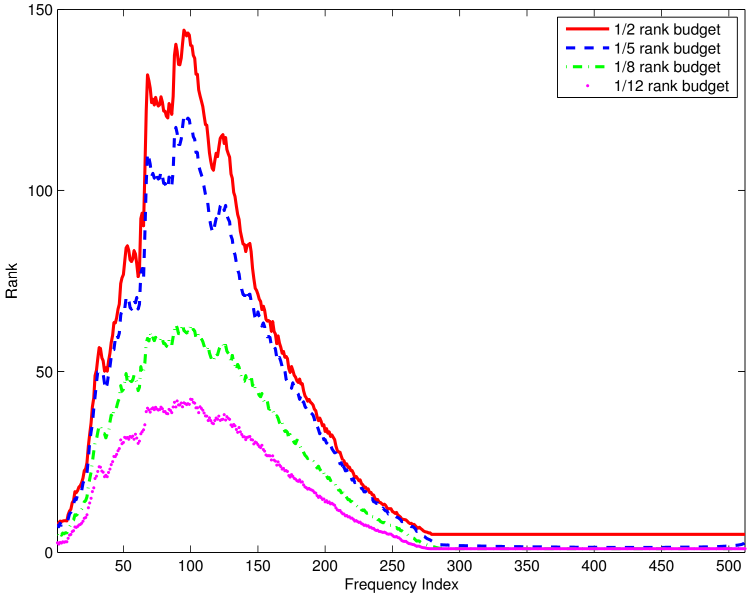
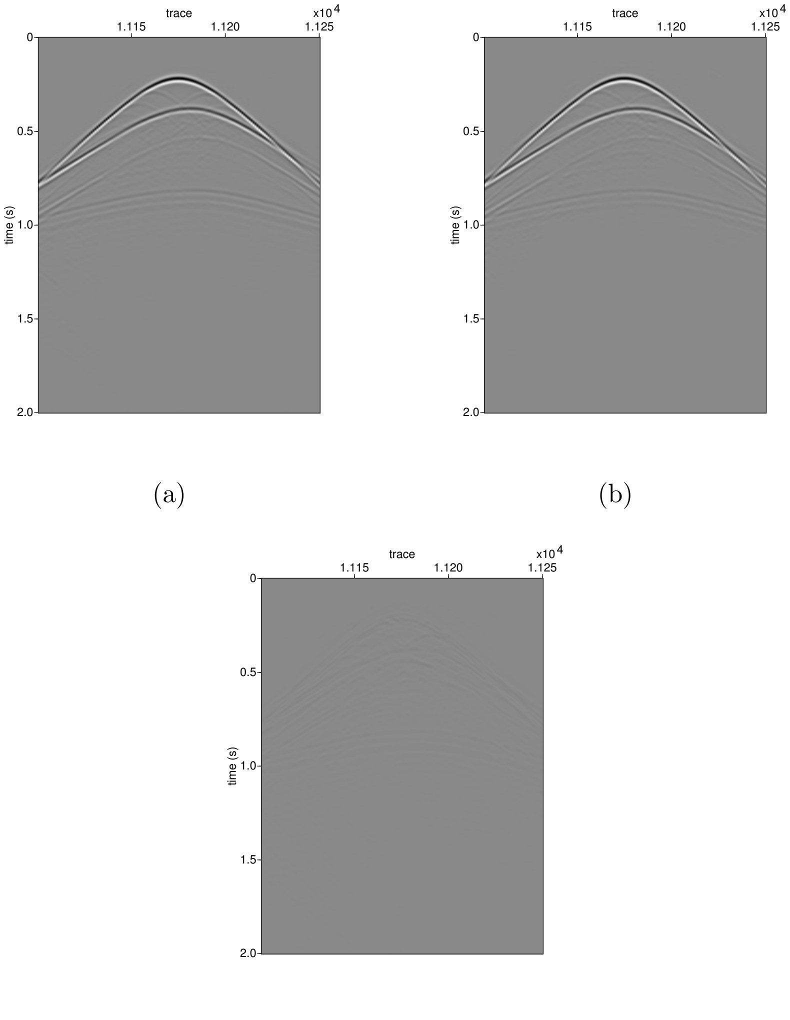


Figure 15: Approximation rank for varying rank budgets in the synthetic case. The minimum rank is user defined.



(a)

(b)

(c)

Figure 16: EPSI results from the synthetic dataset. (a) Estimated Green's function using the full dataset, (b) estimated Green's function with low-rank approximation for $\delta = 1/12$, and (c) difference plot.

Subsample ratio δ	1/2	1/5	1/8	1/12
Recovery error (dB)	30	27	17	12
Speed up (×)	1.6	2	3.5	5.7

Table 4: Summarizing results of estimating the surface-free Green's function using the different sub-sampling ratios.

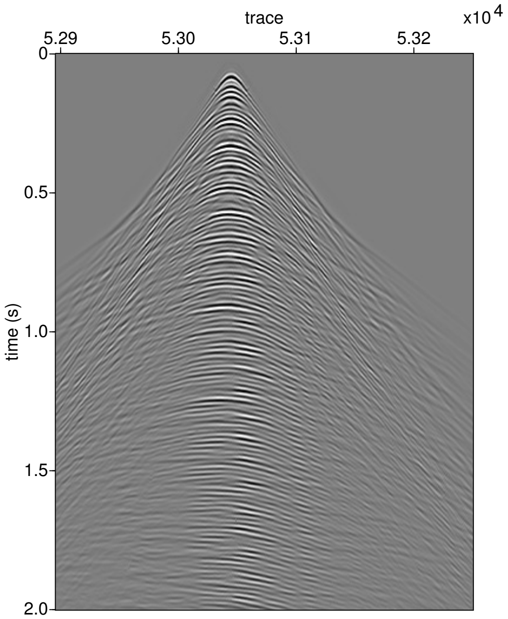
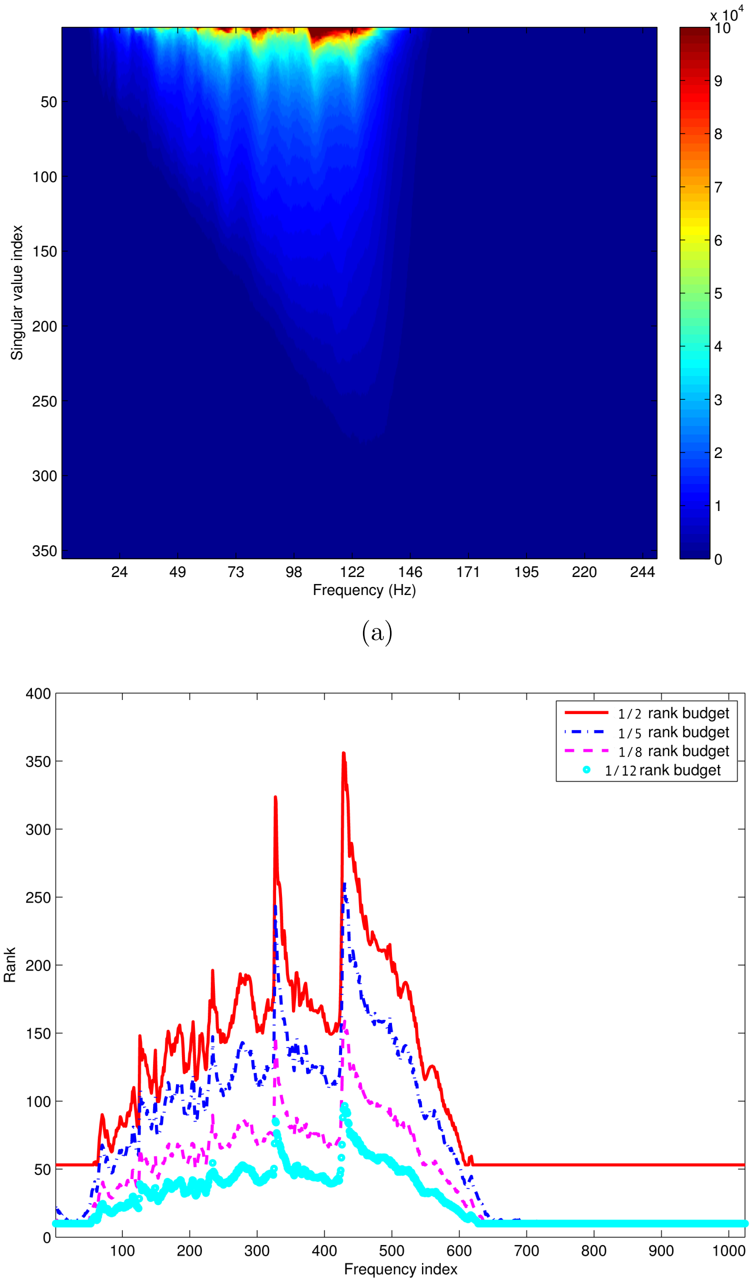


Figure 17: Shot gather from the Gulf of Suez dataset including surface-related multiples.

DISCUSSION AND OUTLOOK

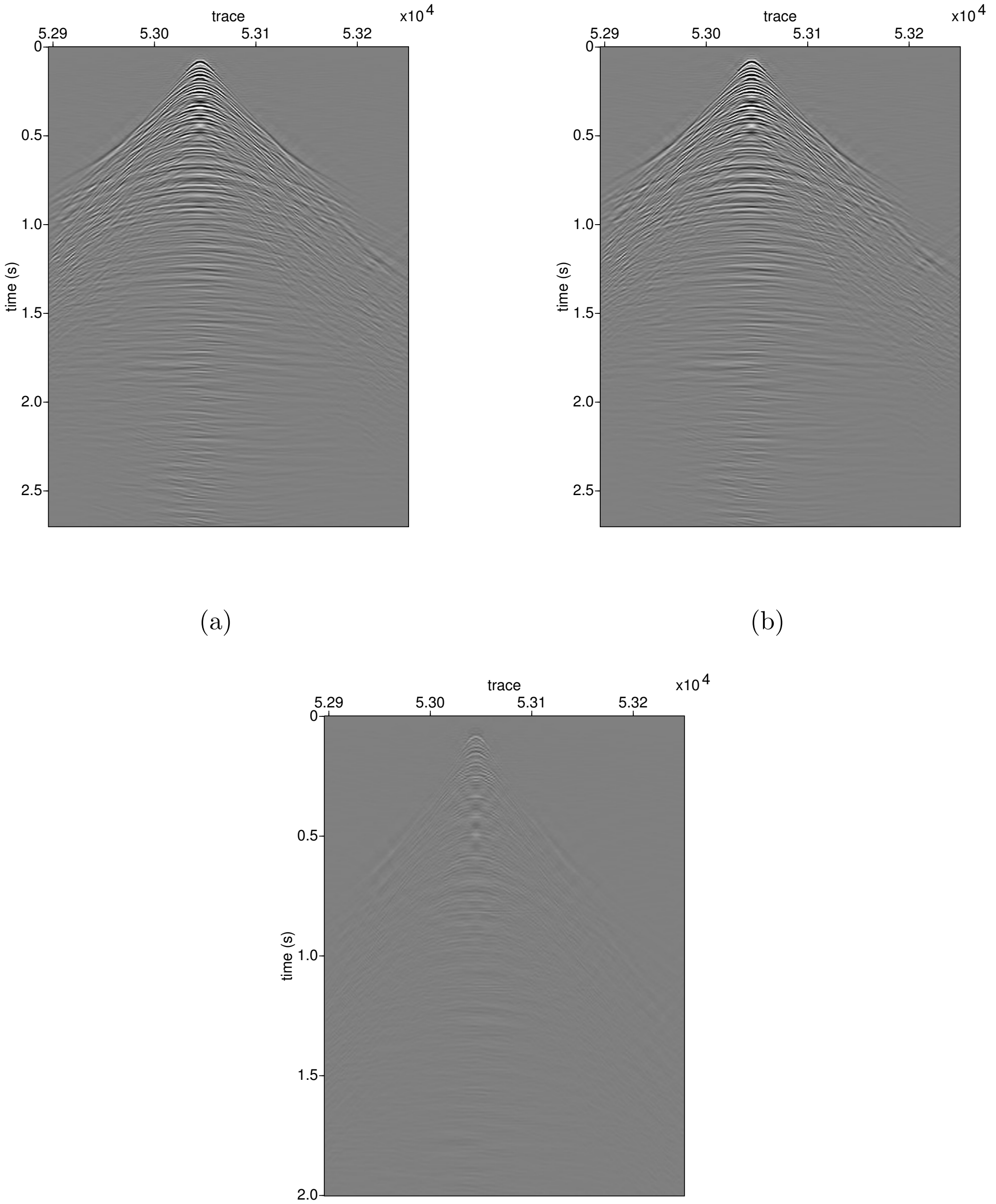
The presented method is exciting for the following reasons. First, we reduce the storage and multiplication costs by roughly a factor of δ at a small up-front cost consisting of $2-3$ passes through all data, a QR factorization, and a singular-value decomposition on the dimensionality reduced system, all of which can be carried out in parallel. The resulting matrix factorization has low memory imprint, leads to fast matrix multiplies, and removes the requirement of passing through all data for each application of the data matrix during the inversion. This feature potentially removes one of the major challenges of extending estimation of primaries by sparse inversion to 3D. Note, however,that this extension needs a low-rank factorization of tensors representing wavefields in more than two dimensions (see e.g. Kolda and Bader, 2009; Caiafa and Cichocki, 2010, for recent developments on this topic). Second, there are



(a)

(b)

Figure 18: EPSI result for Gulf of Suez data. (a) Singular values of the gulf of Suez dataset, (b) approximation rank budgets for varying sub-sampling ratios δ. The minimum rank is user defined.



(a)

(b)

(c)

Figure 19: EPSI result for the gulf of Suez dataset. (a) Estimated Green's function with the full data matrix, (b) estimated Green's function for low-rank approximation for $\delta = 1/12$, and (c) difference plot.

connections between matrix probing and simultaneous sourcing during acquisition of marine data (Wason et al., 2011) or during imaging and full-waveform inversion (Herrmann et al., 2009). This opens the possibility to further speed up our algorithms (see e.g. Herrmann, 2010; van Leeuwen et al., 2011) or to work with simultaneously acquired data directly. Third, because the singular vectors of the data matrix are incoherent with the Dirac basis, we can limit the need of interpolating the data to only once as part of the second stage during which the singular-value decomposition is conducted. In case the locations of the missing shots are sufficiently randomly distributed, we showed that it is no longer necessary to interpolate the data as part of the matrix probing. Instead, we can consider data with randomly missing shots as the result of the matrix probing. Needless to say, this could lead to significant cost savings. Fourth, the proposed algorithm is relatively simple, requires matrix-free application (read 'black-box' implementations of SRME (Verschuur et al., 1992; Berkhout and Verschuur, 1997; Weglein et al., 1997))) of the data matrix only, and limits the number of passes through the data. This may lead to significant speedup because the method only requires a few on-the-fly interpolations. Fifth, our low-rank approximations of the data matrix allows us to leverage recent extensions of compressive sensing to matrix-completion problems (Candes and Recht, 2009; Gandy et al., 2011) where matrices or tensors are reconstructed from incomplete data (read data with missing traces). In these formulations, data is regularized solving the following optimization problem

$$\widetilde{\mathbf{X}} = \arg\min_{\mathbf{X}} \|\mathbf{X}\|_* \quad \text{subject to} \quad \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|_2 \leq \sigma, \tag{15}$$

with $\|\cdot\|_* = \sum |\lambda_i|$ the nuclear norm summing the magnitudes of the singular values (λ) of the matrix $\mathbf{X}$. Here, $\mathcal{A}(\cdot)$ a linear operator that samples the data matrix. It can be shown that this program is a convex relaxation of finding the matrix $\mathbf{X}$ with the smallest rank given incomplete data. Finally, low-rank approximations for tensors were recently proposed by Oropeza and Sacchi (2010) for seismic denoising. Sixth, the singular vectors of our low-rank approximation can also be used in imaging or full-waveform inversion Habashy et al. (2010).

CONCLUSIONS

Data-driven methods—such as the estimation of primaries by sparse inversion—suffer from the 'curse of dimensionality' because these methods require repeated applications of the data matrix whose size grows exponentially with the dimension. In this paper, we leverage recent developments in dimensionality reduction that allow us to approximate the action of the data matrix via a low-rank matrix factorization based on the randomized singular-value decomposition. Combination of this method with hierarchical semi-separable matrix representations enabled us to efficiently factor high-frequency data matrices that have relative high ranks. The resulting low-rank factorizations of the data matrices lead to significant reductions in storage and matrix multiplication costs. The reduction in costs for the low-rank approximations themselves are, by virtue of the randomization, cheap and only require a limited number of applications of the full data matrix to random vectors. This operation can easily be carried out in parallel using existing code bases for surface-related multiple prediction and can lead to significant speedups and reductions in memory use. Because the singular vectors of the data matrices are incoherent with the Dirac basis, matrix probing by Gaussian vectors that require on-the-fly interpolations can be replaced by matrix probings consisting of data with missing shots. As a consequence, the number of interpolations is reduced to only one and this could give rise to a significant improvement in the performance of the inversion, which typically requires several applications of the data matrix.

ACKNOWLEDGMENTS

We would like to thank the authors of SPOT (`http://www.cs.ubc.ca/labs/scl/spot/`), SPGℓ_1 (`http://www.cs.ubc.ca/labs/scl/spgl1/`), CurveLab (`curvelet.org`), and the MSN toolbox for the HSS matrices (`http://scg.ece.ucsb.edu/msn.html`). We also would like to thank Tim Lin for providing us with the code for EPSI. This paper was prepared with Madagascar (`rsf.sf.net`) and was in part financially supported by by the CRD Grant DNOISE 334810-05. The industrial sponsors of the Seismic Laboratory for Imaging and Modelling (SLIM) BG Group, BP, BGP, BP, Chevron, ConocoPhillips, Petrobras, PGS, Total SA., and WesternGeco are also gratefully acknowledged.

REFERENCES

Berkhout, A. J., and Y.-H. Pao, 1982, Seismic migration—imaging of acoustic energy by wave field extrapolation: Journal of Applied Mechanics, **49**, 682–683.

Berkhout, A. J., and D. J. Verschuur, 1997, Estimation of multiple scattering by iterative inversion, part I: theoretical considerations: Geophysics, **62**, 1586–1595.

Caiafa, C. F., and A. Cichocki, 2010, Generalizing the column-row matrix decomposition to multi-way arrays: Linear Algebra and its Applications, **433**, 557 – 573.

Candes, E., and B. Recht, 2009, Exact matrix completion via convex optimization: Foundations of Computational Mathematics, **9**, 717–772.

Candès, E. J., J. Romberg, and T. Tao, 2006, Stable signal recovery from incomplete and inaccurate measurements: **59**, 1207–1223.

Chandrasekaran, S., P. Dewilde, M. Gu, W. Lyons, and T. Pals, 2006, A fast solver for hss representations via sparse matrices.: SIAM J. Matrix Analysis Applications, **29**, 67–81.

Chiu, J., and L. Demanet, 2012, Sublinear randomized algorithms for skeleton decompositions.

Dedem, E., 2002, 3D surface-related multiple prediction: PhD thesis. (Delft University of Technology).

Demanet, L., P.-D. Letourneau, N. Boumal, H. Calandra, J. Chiu, and S. Snelson, 2012, Matrix probing: a randomized preconditioner for the wave-equation Hessian: Journal of Apllied Computational Harmonic Analysis, **32**, 155–168.

Donoho, D. L., 2006, Compressed sensing: IEEE Trans. Inform. Theory, **52**, 1289–1306.

Engquist, B., and L. Ying, 2010, Fast directional algorithms for the Helmholtz kernel: Journal of Computational and Applied Mathematics, **234**, 1851 – 1859.

Gandy, S., B. Recht, and I. Yamada, 2011, Tensor completion and low-n-rank tensor recovery via convex optimization: Inverse Problems, **27**, 025010.

Habashy, T. M., A. Abubakar, G. Pan, and A. Belani, 2010, Full-waveform seismic inversion using the source-receiver compression approach: SEG Technical Program Expanded Abstracts, SEG, 1023–1028.

Halko, N., P. G. Martinsson, and J. A. Tropp, 2011, Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions: SIAM Rev., **53**, no. 2, 217–288.

Herrmann, F. J., 2010, Randomized sampling and sparsity: Getting more information from fewer samples: Geophysics, **75**, WB173–WB187.

Herrmann, F. J., U. Boeniger, and D. J. Verschuur, 2007, Non-linear primary-multiple separation with directional curvelet frames: Geophysical Journal International, **170**, 781–799.

Herrmann, F. J., Y. A. Erlangga, and T. Lin, 2009, Compressive simultaneous full-waveform simulation: Geophysics, **74**, A35.

Kolda, T. G., and B. W. Bader, 2009, Tensor decompositions and applications: SIAM Review, **51**, 455–500.

Li, X., A. Y. Aravkin, T. van Leeuwen, and F. J. Herrmann, 2011, Fast randomized full-waveform inversion with compressive sensing. to appear in Geophysics.

Lin, L., J. Lu, and L. Ying, 2011, Fast construction of hierarchical matrix representation from matrix-vector multiplication: Journal of Computational Physics, **230**, 4071 – 4087.

Lin, T. T., and F. J. Herrmann, 2011, Estimating primaries by sparse inversion in a curvelet-like representation domain: Presented at the 73th Ann. Internat. Mtg., EAGE, Eur. Ass. of Geosc. and Eng., Expanded abstracts.

Lin, T. T. Y., and F. J. Herrmann, 2012, Robust estimation of primaries by sparse inversion via ℓ_1 minimization. (in preparation).

Minato, S., T. Matsuoka, T. Tsuji, D. Draganov, J. Hunziker, and K. Wapenaar, 2011, Seismic interferometry using multidimensional deconvolution and crosscorrelation for crosswell seismic reflection data without borehole sources: Geophysics, **76**, SA19–SA34.

Oropeza, V. E., and M. D. Sacchi, 2010, A randomized SVD for multichannel singular spectrum analysis (MSSA) noise attenuation: SEG, Expanded Abstracts, **29**, 3539–3544.

van Groenestijn, G. J. A., and D. J. Verschuur, 2009, Estimating primaries by sparse inversion and application to near-offset data reconstruction: Geophysics, **74**, A23–A28.

van Leeuwen, T., A. Aravkin, and F. J. Herrmann, 2011, Seismic waveform inversion by stochastic optimization: International Journal of Geophysics, **689041**.

Verschuur, D. J., A. J. Berkhout, and C. P. A. Wapenaar, 1992, Adaptive surface-related multiple elimination: Geophysics, **57**, 1166–1177.

Wason, H., F. J. Herrmann, and T. T. Lin, 2011, Sparsity-promoting recovery from simultaneous data: a compressive sensing approach: SEG Technical Program Expanded Abstracts, SEG, 6–10.

Weglein, A. B., F. A. Carvalho, and P. M. Stolt, 1997, An iverse scattering series method for attenuating multiples in seismic reflection data: Geophysics, **62**, 1975–1989.