

Seismic Data Reconstruction via Matrix Completion

Yi Yang, Jianwei Ma and Stanley Osher

Abstract

In seismic processing, one goal is to recover missing traces when the data is sparsely and incompletely sampled. We present a method which treats this reconstruction problem from a novel perspective. By utilizing its connection with the general matrix completion (MC) problem, we build an approximately low-rank matrix, which can be reconstructed through solving a proper nuclear norm minimization problem. Two efficient algorithms, accelerated proximal gradient method (APG) and low-rank matrix fitting (LMaFit) are discussed in this paper. The seismic data can then be recovered by the conversion of the completed matrix into the original signal space. Numerical experiments show the efficiency and high performance of data recovery for our model compared with other algorithms.

I. INTRODUCTION

Seismic data is often irregularly sampled along spatial coordinates, which is mostly caused by dead or severely corrupted traces, surface obstacles, acquisition aperture and economic limit. Seismic data regularization spatially transforms irregularly acquired data to regularly sampled data in order to subsequently display, process and interpret it. This is a long-standing and important problem in seismic processing. Seismic data interpolation and reconstruction is one particular case of data regularization. Nowadays, the recovery of missing traces has become a main issue in seismic research, where input data are already measured or observed in a regular grid and one needs to reconstruct the value in missing traces (i.e., empty bins).

Different approaches have been proposed to solve this problem. Some of these techniques require converting the data into different domains with certain transform dictionary, such as

Y. Yang and S. Osher are with the Department of Mathematics, University of California, Los Angeles, CA 90095, USA ({yyang, sjo}@math.ucla.edu). J. Ma is with the Institute of Applied Mathematics, Harbin Institute of Technology, Harbin, China (jma@hit.edu.cn).

Fourier transform ([1], [2], [3], [4]), curvelet transform ([5], [6], [7]) and shearlet transform ([8]). The idea of L_1 -norm minimization was applied in ([5], [6], [8]) to reconstruct the missing traces of seismic data. Another popular technique is the use of prediction filters ([9], [10], [11]), where low-frequency non-aliased components or observed data are used to build antialiasing error-prediction filters and then the filters are applied to interpolate high-frequency components or missing traces. The idea of prediction filters has been extended by many researchers in the past few years. For instance, Liu and Fomel ([12]) proposed a regularized nonstationary autoregression for adaptive error-prediction filtering; Naghizadeh and Sacchi ([13]) used multidimensional prediction filters for seismic data reconstruction and in ([7]) they proposed prediction filters using curvelet transform instead of Fourier transform.

Recently, rank-reducing methods are used for seismic data reconstruction. Trickett et al. ([14]) presented a truncated singular value decomposition based on rank reduction of constant-frequency slices for trace interpolation. In ([15]), Oropeza and Sacchi reorganized the seismic data into a Hankel matrix, and then used multichannel singular spectrum analysis (MSSA) to solve the rank-reduction problem of the Hankel matrix. The MSSA comes from the generalization of singular spectrum analysis (SSA) ([16]). Basically, SSA is used for 2D data and MSSA is applied to 3D data recovery.

Mathematically speaking, this reconstruction problem can be understood in the following way: we are given a matrix with some missing columns, and we want to complete the matrix by filling in the missing data. This interpretation has some similarities with the matrix completion (MC) problem, in which we are given some components of a matrix X and we want to reconstruct X with its rank as low as possible. In this paper, we will explore the connection in a novel way such that a lot of efficient MC algorithms can be applied to seismic data recovery. The technology of rank reduction has been used for data reconstruction in MSSA based methods in [15], but here we start from different theory and algorithms.

From the mathematical theory of MC, however, one can not simply define the given signal matrix as an input in MC framework. The main reason is that based on the justification in [17], it is impossible to reconstruct a matrix when the sampling set avoids any column or row of X . That is to say, we need at least one nonzero value in each column vector. Unlike the applications of MC in image inpainting where pixels are randomly missed, we have several columns missing in seismic data. In order to apply the MC idea, we should first design a pre-

transformation which transforms the given seismic data with column/trace missing to a new matrix with pixel missing, i.e., each column and row of the new matrix has at least one nonzero entry. Besides, sometimes the original signal may have complicated structure and the matrix may not necessarily be low-rank. Therefore, we should try to explore special structure of the given data and use pre-transformation to construct a low-rank matrix with proper sampling set.

In this paper, we build a texture matrix through pre-transformation. This new matrix is either low-rank or can be approximated by a low-rank matrix. With very high probability there is at least one nonzero entry in each row and column of it. Hence it can be reconstructed via regular MC algorithms. We can then recover the original signal matrix through transforming the completed low-rank matrix into the original space. The main contribution of our work is that we attack seismic reconstruction from a completely new angle. By making the connection with MC problem, a lot of efficient MC algorithms from convex optimization can be utilized to recover the seismic data. The proposed method is simpler and faster than existing methods.

This paper is organized as follows. In section II and III, we will give a detailed description about the MC problem and some commonly used algorithms for solving it. Section IV focuses on the construction of texture matrix. The performance of our model is illustrated in section V with comparison to algorithms mentioned in [15]. We will end this work by a short conclusion.

II. MATRIX COMPLETION

Nowadays, a lot of real world models can be categorized as MC problems, such as video denoising ([18]) etc. The general form of the problem is:

$$\min_{X \in \mathbf{R}^{m \times n}} \text{rank}(X), \text{ s.t. } X_{ij} = M_{ij} \text{ for } (i, j) \in \Omega \quad (1)$$

Here $\text{rank}(X)$ is the number of nonzero singular values of X . Sometimes when noise is involved in the measurements, the model is adjusted to

$$\min_{X \in \mathbf{R}^{m \times n}} \mu \cdot \text{rank}(X) + \frac{1}{2} \|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2 \quad (2)$$

with a properly tuned parameter μ . Here \mathcal{P}_Ω stands for the the projection onto the subspace of matrices with nonzero entries restricted to the index subset Ω , and $\|\cdot\|_F$, the Frobenius norm,

is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}. \quad (3)$$

for any matrix $A = (a_{ij})_{m \times n}$. However, solving the above two models are often numerically expensive. Hence people tend to consider their relaxation:

$$\min_{X \in \mathbf{R}^{m \times n}} \|X\|_*, \text{ s.t. } X_{ij} = M_{ij} \text{ for } (i, j) \in \Omega \quad (4)$$

and

$$\min_{X \in \mathbf{R}^{m \times n}} \mu \|X\|_* + \frac{1}{2} \|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M)\|_F^2 \quad (5)$$

Here $\|X\|_*$ stands for the nuclear norm of X , which is the L_1 norm of the singular values. As a matter of fact, similar relaxation also appears in compressed sensing theory, where minimizing the L_1 norm of a vector is used as a relaxation of L_0 minimization.

In raw seismic data $X = (x_{ij})_{m \times n}$, the i denotes temporal sampling and j stands for trace sampling, i.e., the spatial location of receivers. For instance, the first column of X is the 1D time sequence signal acquired at the first receiver. Each column of X is named as one seismic trace.

III. FAST ALGORITHMS FOR MC

Lots of algorithms have been proposed to solve (4) and (5), such as linearized Bregman, fixed point continuation with Bregman iteration etc. See [19], [20], [21], [22] for more details. In this paper we will focus on the following two, accelerated proximal gradient method (APG, it is the same as FISTA) and LMaFit. We need to point out that LMaFit is not designed to solve the standard MC problem. It is actually used for solving the low-rank matrix reconstruction where the low-rank matrix is parameterized in a factorization form. More details about it will be provided in the following.

Accelerated proximal gradient method comes from [23], [24], and it was applied to MC problems in [25]. It is a classic method for solving (5). In order to illustrate this method, let us briefly introduce the idea of proximal mappings.

The proximal mapping (prox-operator) of a convex function h is defined as

$$\mathbf{prox}_h(u) = \arg \min_x h(x) + \frac{1}{2}\|x - u\|^2. \quad (6)$$

Let us assume we want to minimize a function $f(u)$ of the form

$$f(u) = g(u) + h(u), \quad (7)$$

with a convex and differentiable $g(u)$, and a convex $h(u)$ with inexpensive prox-operator. The proximal gradient method computes the minimizer of $f(u)$ iteratively via

$$u^{k+1} = \mathbf{prox}_{t^k h}(u^k - t^k \nabla g(u^k)), \quad (8)$$

with a suitable step size t^k . For (5), by choosing $t^k \equiv \tau$ we have

$$X^{k+1} = S_\tau(X^k - \tau(\mathcal{P}_\Omega(X^k) - \mathcal{P}_\Omega(M))), \quad (9)$$

Here $S_\tau(G) = U \text{diag}((\sigma - \mu\tau)_+) V^T$ and $G = U \text{diag}(\sigma) V^T$ is the singular value decomposition.

APG uses extrapolation on the result from (8) and is usually several times faster than regular proximal gradient method. In order to obtain the optimal X of (5), in each iteration we update X^k in the following way:

$$\begin{aligned} Y^k &= X^k + \frac{t^{k-1}-1}{t^k}(X^k - X^{k-1}) \\ G^k &= Y^k - \tau(\mathcal{P}_\Omega(Y^k) - \mathcal{P}_\Omega(M)) \\ X^{k+1} &= S_\tau(G^k) \\ t^{k+1} &= \frac{1 + \sqrt{1 + 4(t^k)^2}}{2} \end{aligned} \quad (10)$$

Here τ has to be chosen properly. We also want to mention that with Bregman iteration, the solution to (4) can be obtained by solving a sequence of (5) with noise-add-back skill in each iteration:

$$\begin{aligned} X^{k+1} &= \arg \min_X \mu \|X\|_* + \frac{1}{2} \|\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(M^{k+1})\|_F^2 \\ M^{k+1} &= M^k + \mathcal{P}_\Omega(M) - \mathcal{P}_\Omega(X^{k+1}) \end{aligned} \quad (11)$$

LMaFit, proposed in [26], is an efficient algorithm designed to solve

$$\min_{X, Y, Z} \|XY - Z\|_F^2 \text{ s.t. } Z_{ij} = M_{ij} \text{ for } (i, j) \in \Omega. \quad (12)$$

Here $X \in \mathbf{R}^{m \times k}$, $Y \in \mathbf{R}^{k \times n}$, and $Z \in \mathbf{R}^{m \times n}$, where k is a predicted rank bound. With an appropriate k , the product of the optimal X and Y from (12) is also a minimizer of (4). X , Y , Z can be updated in an alternating fashion. Following the idea of nonlinear successive-over-relaxation (SOR) algorithm, the author in [26] used weighted average between this update and the previous iterate. In each iteration, the new X , Y and Z are defined in the following way:

$$\begin{aligned}
 X_{temp}^{k+1} &= Z^k(Y^k)^\dagger \\
 X^{k+1} &= \omega X_{temp}^{k+1} + (1 - \omega)X^k \\
 Y_{temp}^{k+1} &= (X^{k+1})^\dagger Z^k \\
 Y^{k+1} &= \omega Y_{temp}^{k+1} + (1 - \omega)Y^k \\
 Z^{k+1} &= X^{k+1}Y^{k+1} + \mathcal{P}_\Omega(M - X^{k+1}Y^{k+1})
 \end{aligned} \tag{13}$$

with weight parameter $\omega > 1$. Here $(\cdot)^\dagger$ stands for the pseudo inverse of a matrix. As we can see, this algorithm avoids singular value decomposition (SVD) in each iteration, which is the main time cost in most of the algorithms for (4). Hence when timing is the main issue, LMaFit is undoubtedly a great choice.

IV. MODEL CONSTRUCTION: PRE-TRANSFORMATION

In our discussion we assume that the number of consecutive missing columns in the given matrix is bounded above by a constant C . This is a reasonable assumption since the data can not be restored properly if too many successive columns are missing at the same time. As mentioned in introduction, the motivation of pre-transformation involves two aspects: 1) reshape the data with column missing to a new format with pixel missing as required by MC theory; 2) after the transformation, the new data matrix has low rank or can be approximated by a low-rank one. In this paper, we consider the following texture matrix.

Due to the continuity and integrity of seismic data, it is reasonable to believe that our signal only has a few texture patterns, which is defined as base texture in [27]. Inspired by it, if we divide the signal matrix into small $r \times r$ patches, each submatrix should be a combination of these base textures. Furthermore, if we rearrange each patch into a vector with the same ordering, then these vectors will be (highly) linearly dependent, i.e., the patched matrix has lower rank or can be approximated with a low-rank matrix. Later we will justify this claim with numerical test on real data. In [27], Schaeffer and Osher defined the texture matrix as follows:

Definition 1: Given the original signal $X \in \mathbf{R}^{m \times n}$ and patch size r , we can divide X into mn/r^2 subblocks, labeled as $\{B_i\}$. Each B_i can be rearranged into a column vector w_i following the same ordering. The texture matrix T is defined as

$$T = [w_1, w_2, \dots, w_{mn/r^2}]. \quad (14)$$

The nuclear norm of T is defined as the texture norm of X . The nuclear norm minimization is correspondingly shifted to texture norm minimization.

Remark:

- The specific ordering which maps each patch to a vector is not important, as long as it is consistent. In this paper the patches are labeled in the following way:

$$\begin{pmatrix} B_1 & B_2 & \cdots & B_{n/r} \\ B_{n/r+1} & B_{n/r+2} & \cdots & B_{2n/r} \\ \vdots & \vdots & \ddots & B_{mn/r^2} \end{pmatrix}. \quad (15)$$

And for each patch, e.g. B_1 , the patch-vector is defined as:

$$B_1 = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1r} \\ x_{21} & x_{22} & \cdots & x_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ x_{r1} & x_{r2} & \cdots & x_{rr} \end{pmatrix} \implies (x_{11}, \dots, x_{r1}, x_{12}, \dots, x_{r2}, \dots, x_{rr})^{T_t} = w_1. \quad (16)$$

where T_t denotes transpose.

- In the definition of T , the patches are defined in a non-overlapping way. The overlapping case is not considered for two reasons. Firstly, if we construct the texture matrix $T \in \mathbf{R}^{m_1 \times n_1}$ with overlapping patches, we shall have $m_1 n_1 > mn$, hence the new matrix belongs to a space with higher dimension than the original one. This increases the complexity of the problem, and may not be feasible since we must apply MC techniques to this larger matrix and then transform it back to the original space. More importantly, if each pixel contains noise, then the robustness from applying patch based rank reduction techniques will be lost once the patches overlap. The reason is as follows: the overlap causes the patches to become highly correlated, thus the noise is no longer random and cannot be separated via MC and other rank reduction techniques. More explanation about this can be found in the

latest version of [27], to appear soon.

In order to apply MC techniques on the new matrix T , we first need to ensure that if we choose r properly, with high probability there is at least one nonzero entry for each row and column of T . If we have a missing column in the new matrix, then one of the patches B_k has to be all-zero, which means that the consecutive r columns that B_k stays in are all missing. This can not happen if r is greater than C .

On the other hand, if we have a missing row in the new matrix, then there exists $1 \leq i \leq r$, $1 \leq j \leq r$ such that $B_k(i, j) = 0$ for all $1 \leq k \leq mn/r^2$. Hence the $j^{th}, (j+r)^{th}, \dots, (j+n-r)^{th}$ columns should all be zero. Since we are dealing with column missing, finding the probability of this case is the same as the following: given a length n vector with missing rate ρ , where at most C consecutive elements are missing at the same time, we need to find the probability that there exists j between 1 and r such that the $j^{th}, (j+r)^{th}, \dots, (j+n-r)^{th}$ entries are all missing.

The exact probability for this case is hard to measure. Here we consider a sampling model that people often use in application: jittered sampling [6]. We first divide the long vector into several non-overlapping short vectors of length $C/2\rho$, and choose $\rho \cdot C/2\rho$ elements from each part. In this way at most $C/2$ consecutively entries are missing from each short vector, hence the original vector will have at most C consecutive missing entries. Let us choose $r > C/2\rho$, then for each j value between 1 and r , the $j^{th}, (j+r)^{th}, \dots, (j+n-r)^{th}$ entries will stay in different short vectors, and the probability of each element missing is independent and equals ρ . We then know that for a particular j , the probability of the $j^{th}, (j+r)^{th}, \dots, (j+n-r)^{th}$ elements all missing is $\rho^{n/r}$. Since j can be any number between 1 and r , the overall probability is bounded by $r\rho^{n/r}$. Therefore, if we have $r > \max(C, C/2\rho)$ and n/r large, T will have at least one nonzero element for each row with extremely high probability. For example, when we have $n = 128$, $\rho = 0.5$ and $C = 2$, if we choose $r = 8$, the probability of having one row missing in T is less than $1.2 \cdot 10^{-4}$.

Next we use real data to show that for the original seismic data with no missing traces, the transformed T can be approximated by a low-rank matrix. When the data has some missing columns, the associated T may not be close to a low-rank matrix. Here two data sets X_1, X_2 are considered. For each test matrix, we generate the pre-transformation T with the original data and the data with missing columns. Different r values are used in our test, and we display the

singular value plot in the following Figure 1.

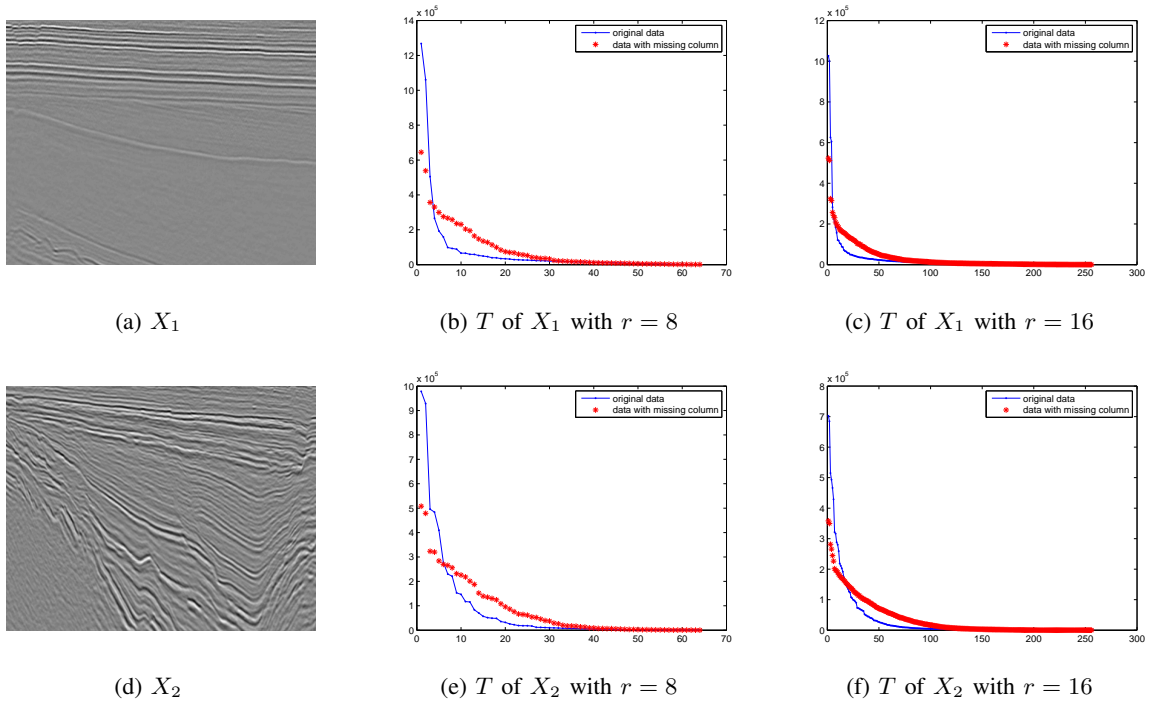


Fig. 1: Singular value plots for the texture matrix of seismic data sets X_1 and X_2 . (a) data set X_1 , (b) the singular value plots for the texture matrix of X_1 (blue) and X_1 with missing columns (red star) with $r = 8$, (c) the same setting with $r = 16$. (d) data set X_2 , (e) the singular value plots for the texture matrix of X_2 (blue) and X_2 with missing columns (red star) with $r = 8$, (f) the same setting with $r = 16$.

We can see from Figure 1 that the singular value plots from the original data (blue) always have sharp decrease. According to our test, for the original data set X_1 , the ratio between the first and the 15th singular values is 25.83 for $r = 8$ case. When $r = 16$, the ratio between the first and the 50th singular values is 44.46. For the original data set X_2 , when we set $r = 8$, the ratio between the first and the 15th singular values is 17.19. When $r = 16$, the ratio between the first and the 50th singular values is 25.78. When part of the columns are missing, we can see clearly from the plots that the associated T can hardly be approximated by low-rank matrices. Based on the shape of the singular value plot and the computed ratio values, it is reasonable for us to assume that T generated from the original (complete) data can be approximated by a low-rank matrix.

V. NUMERICAL RESULTS

We test the above model with different algorithms on real seismic data. In each experiment, we fix the signal matrix and conduct the comparison with different sampling sets. 50% of the traces are extracted with jittered sampling and $C = 2$. The test is conducted with 10 different sampling sets and we record the mean time cost and signal-to-noise ratio (SNR). SNR is denoted by $10 \log_{10}(\|X^*\|_F^2 / \|X - X^*\|_F^2)$, where X^* is the original signal matrix and X is the recovered one. We also display the comparison of one original (missing) trace with reconstructed traces by different methods.

A. Comparison with SSA based algorithm

In our first test, we compare our model with the SSA based algorithm in [15] (algorithm (11) in that paper). For simplicity from now on we will use SSA to denote this method. A detailed description of this algorithm is provided in appendix. For SSA filter, we choose rank threshold $k = 5$ for X_1 and $k = 7$ for X_2 . The number of outer iteration is set to be 10 for X_1 and 15 for X_2 . Since this algorithm tends to take a long time for large data set, we restrict the size of the signal matrix to 128×128 . APG and LMaFit are applied in our model. The result is shown below in Figure 2 and 3. The (f) part of each figure is the trace comparison, where we extract one central missing trace (in red dash line) and compare it with the associated recovered traces from different algorithms (blue line). The order from left to right is: SSA, APG and LMaFit.

We can see clearly from these figures that for both data sets the texture-patch model requires much less time and also gives better results. According to the trace comparison, the results from our model match the true trace perfectly while SSA might miss some parts. We can optimize the results of SSA by taking more outer iterations, but we have to sacrifice even more in time cost. For both tests, APG gives the highest SNR with medium time cost. According to the signal matrix graph, the recovered result from APG is smoother than the other two. When the data becomes more complicated (X_2) with edges and corners, the result from APG and LMaFit are comparable while LMaFit takes much less time.

B. Test with large data sets

This time the original 512×512 signal matrices are used in the experiment. APG and LMaFit are tested and compared. When we use X_1 (Figure 1 (a)) as input signal, we can recover the

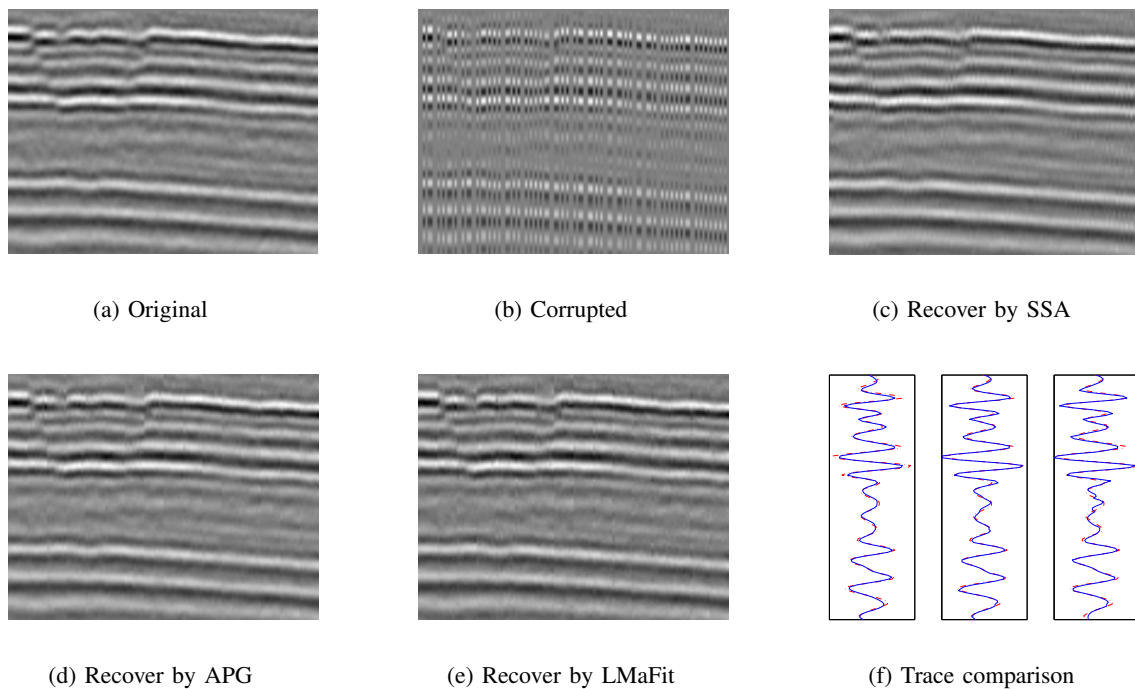


Fig. 2: Signal matrix X_1 (partial) recovery. We restrict its size to 128×128 . (a) the original signal matrix, (b) the corrupted signal, the gray lines are missing traces, (c) the recovered signal via SSA based algorithm. It takes 40.30s and gives SNR 15.29. (d) the recovered signal via APG. It takes 2.61s and gives SNR 17.79. (e) the recovered signal via LMaFit. It takes 0.25s and gives SNR 16.66. (f) comparison of central traces from reconstructed results by different methods. The recovered traces from our model are much better than the one from SSA.

entire signal matrix with less than 7s for both algorithms. Still, APG gives a higher SNR with longer time. The results are displayed in Figure 4.

When the original signal becomes more complicated (Figure 1 (b)), both SNR become undoubtedly lower while still acceptable. According to Figure 5, this time we would rather choose LMaFit since it takes less time and gives similar SNR. We may say that when the original signal has complicated structure, LMaFit should be our choice.

We also consider noisy case for the texture model. X_1 is used as input and Gaussian noise is added to the given data. We define ρ to be the ratio between the variance of the noise and the L_∞ norm of the seismic data. The results for $\rho = 0.01$ and 0.02 are depicted in Figure 6 and Figure 7. For both cases, we are able to get acceptable results. We can see that the unconstrained model is better than the constrained one for the noisy case, since APG always gives a much

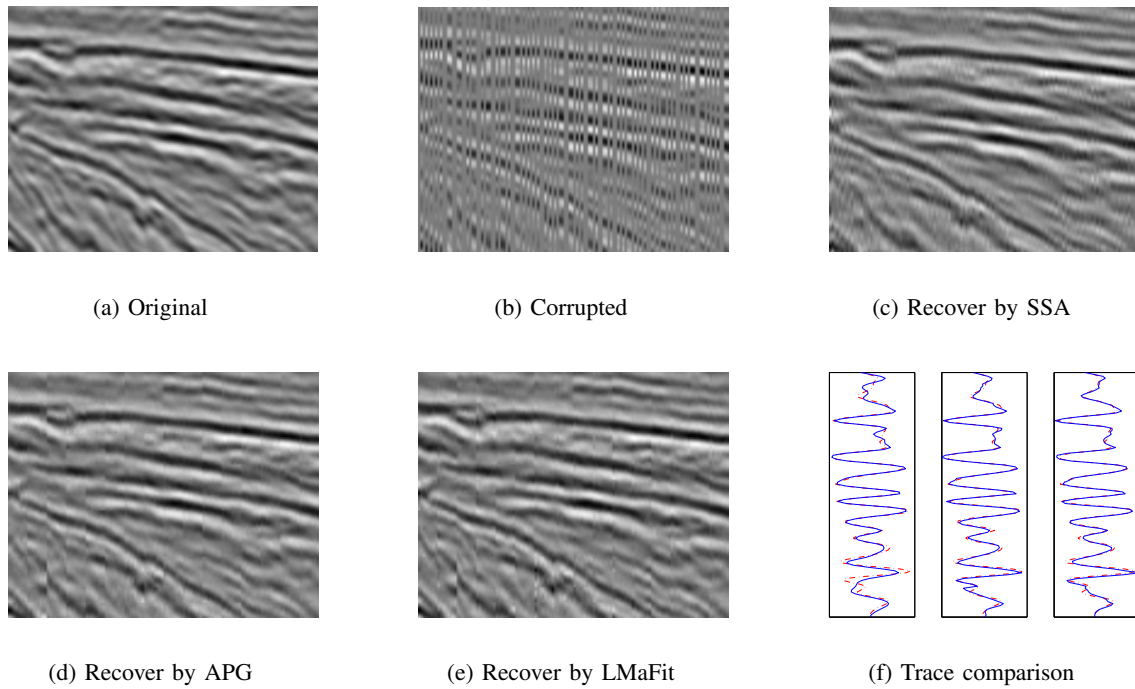


Fig. 3: Signal matrix X_2 (partial) recovery. We restrict its size to 128×128 . (a) the original signal matrix, (b) the corrupted signal, (c) the recovered signal via SSA based algorithm. It takes 57.70s and gives SNR 14.22. (d) the recovered signal via APG. It takes 2.56s and gives SNR 17.02. (e) the recovered signal via LMaFit. It takes 0.13s and gives SNR 16.20. (f) comparison of central traces from reconstructed results by different methods.

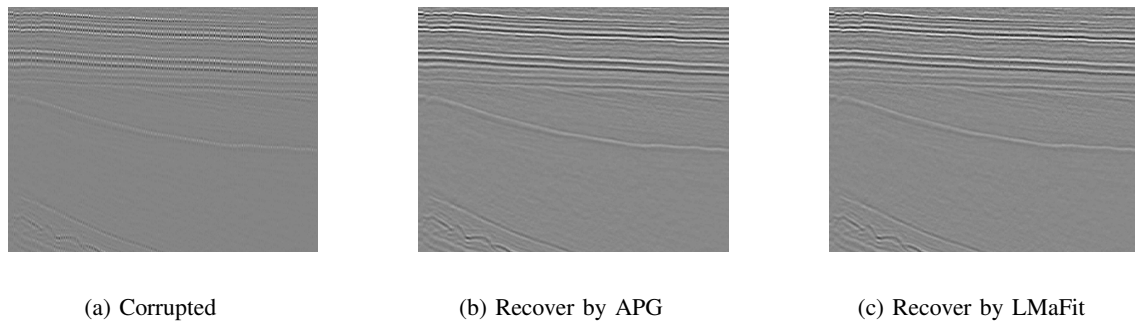


Fig. 4: Signal matrix X_1 recovery with texture matrix. (a) the corrupted signal, (b) the recovered signal via APG, (c) the recovered signal via LMaFit. APG takes 6.23s and gives SNR 18.21. LMaFit takes 2.23s and gives SNR 16.19.

higher SNR and approaches the true missing traces better.

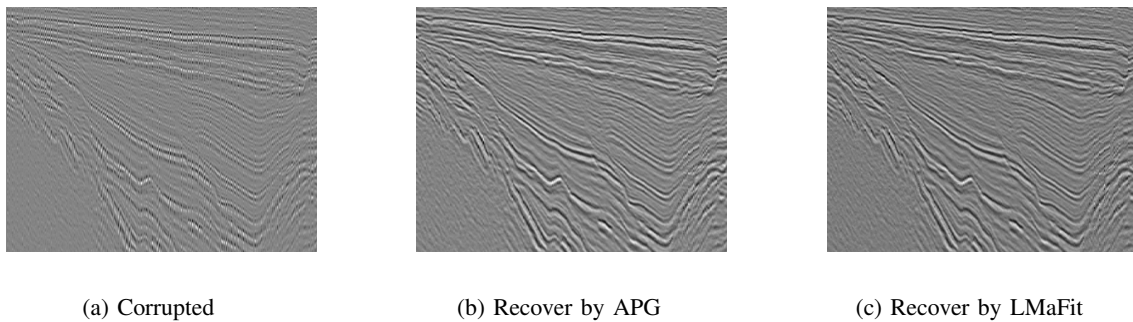


Fig. 5: Signal matrix X_2 recovery with texture matrix. (a) the corrupted signal, (b) the recovered signal via APG, (c) the recovered signal via LMaFit. APG takes 5.60s and gives SNR 14.10. LMaFit takes 1.79s and gives SNR 14.04.

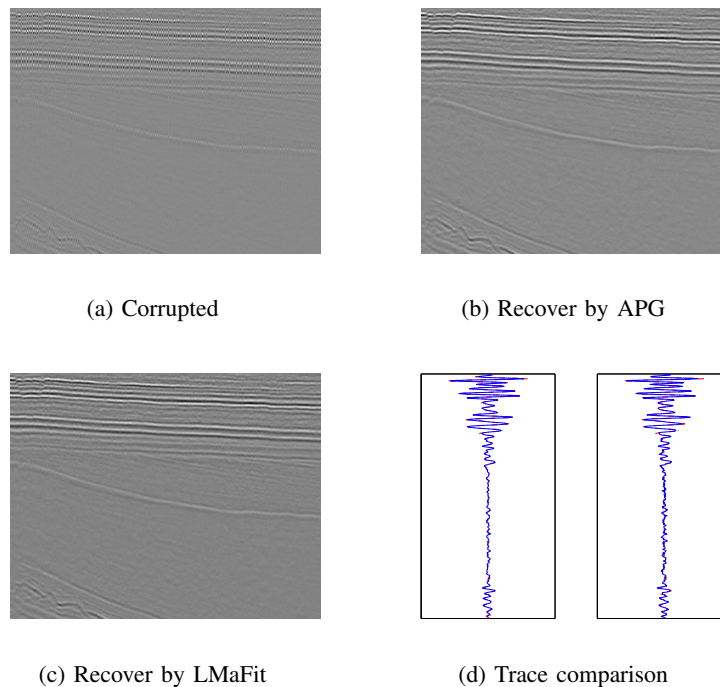


Fig. 6: Signal matrix X_1 recovery with texture matrix. Gaussian noise is added to the given data with ratio $\rho = 0.01$. (a) the corrupted signal with missing traces and noise, (b) the recovered signal via APG, (c) the recovered signal via LMaFit, (d) the central trace comparison with APG on the left and LMaFit on the right. APG takes 5.13s and gives SNR 17.35. LMaFit takes 1.04s and gives SNR 15.34.

VI. CONCLUSION

In this paper, we discover the connection between seismic data reconstruction and the general matrix completion problem. We formulate the seismic data reconstruction as a nuclear-norm

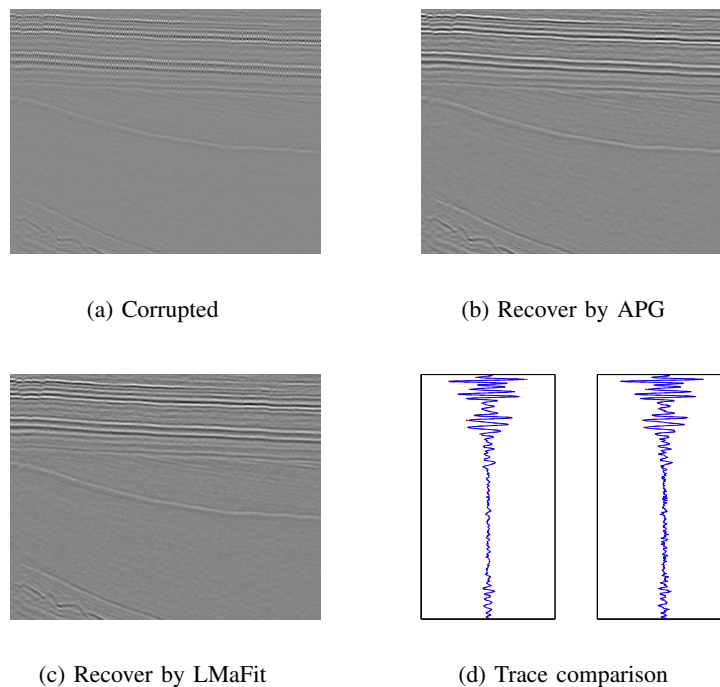


Fig. 7: Signal matrix X_1 recovery with texture matrix. Gaussian noise is added to the given data with ratio $\rho = 0.02$. (a) the corrupted signal with missing traces and noise, (b) the recovered signal via APG, (c) the recovered signal via LMaFit, (d) the central trace comparison with APG on the left and LMaFit on the right. APG takes 5.10s and gives SNR 15.26. LMaFit takes 0.99s and gives SNR 12.90.

minimization for the first time. The texture-patch transformation is provided to build the low-rank matrix with proper sampling sets, which can be recovered with regular MC algorithms. APG and LMaFit are discussed in this paper, and the performances of these two algorithms are shown in the numerical experiments. The reconstruction based on texture matrix tends to cost much less time and give better result compared to SSA based algorithms. When the original signal has complicated structure, our model gives reasonable result within a really short time. In this case, our result can be used as initial input for other refined but time-consuming algorithms.

How to build a joint constraint with rank reduction and sparsity transforms (e.g. curvelets), and how to addresses the problem of interpolating regular missing traces is our next work.

VII. ACKNOWLEDGEMENT

Y. Yang and S. Osher are supported by ONR NO0014-08-1-1119, ONR N00014-11-0719, NSF DMS-1118971, NSF DMS-0914561 and an ARO MURI subcontract from Rice University. J. Ma is supported by Program for New Century Excellent Talents in University under Grant No. NCET-11-0804. The authors would like to thank Mauricio Sacchi for providing the Matlab code for SSA filter and constructive discussions. We would also like to thank Hayden Schaeffer for his helpful suggestions on building the texture-patch matrix.

VIII. APPENDIX: SSA

We first introduce the Hankel matrix construction, which is one step in the SSA filter. For 3D seismic data, a more complicated block Hankel matrix should be built. For a given signal matrix in $f - x$ (frequency-space) domain:

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \ddots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{pmatrix}, \quad (17)$$

the Hankel matrix for row j is constructed via

$$H_j = \begin{pmatrix} x_{j,1} & x_{j,2} & \cdots & x_{j,K_n} \\ x_{j,2} & x_{j,3} & \cdots & x_{j,K_n+1} \\ \vdots & \vdots & \ddots & \ddots \\ x_{j,L_n} & x_{j,L_n+1} & \cdots & x_{j,n} \end{pmatrix}, \quad (18)$$

Usually we choose $L_n = \lfloor n/2 \rfloor + 1$ and $K_n = n - L_n + 1$ to make H_j close to square. The SSA filter is defined as follows: for each row j , we first construct the Hankel matrix H_j , and then apply SVD to get $H_j = U \text{diag}(\sigma) V$. Here σ_i are singular values. Only the first k largest singular values and the corresponding singular vectors of H_j are kept for the next step, i.e., we only treat $H_j^k = U^k \text{diag}(\sigma_1, \dots, \sigma_k) V^k$ as useful information. The final step is averaging the antidiagonal of each block of H_j^k to reform the row j . It has been proven in [28] that if the data consists of k complex exponentials, the associated Hankel matrix of the data is a matrix of rank k . Usually the noise and missing data will increase the rank of Hankel matrix, hence

proper rank reduction technique can be used for data denoising and recovery.

Let us define this SSA filter for matrix X as $\mathcal{F}(X)$, X^{obs} as the original given matrix with missing traces, and \mathcal{P}_I as the projection onto the given data set. The SSA based algorithm we used as comparison (Algorithm 11) is illustrated below:

$$X^0 = X^{obs} \tag{19}$$

$$X^k = X^{obs} + (1 - \mathcal{P}_I)\mathcal{F}(X^{k-1}) \tag{20}$$

The number of iterations taken for this algorithm is denoted as outer iteration in our discussion.

REFERENCES

- [1] M. Sacchi, T. Ulrych, and C. Walker, "Interpolation and extrapolation using a high-resolution discrete fourier transform," *IEEE Transactions on Signal Processing*, vol. 46, pp. 31–38, 1998.
- [2] A. Duijndam, M. Schonewille, and C. Hindriks, "Reconstruction of band-limited signals, irregularly sampled along one spatial direction," *Geophysics*, vol. 64, pp. 524–538, 1999.
- [3] B. Liu and M. D. Sacchi, "Minimum weighted norm interpolation of seismic records," *Geophysics*, vol. 69, pp. 1560–1568, 2004.
- [4] S. Xu, Y. Zhang, D. L. Pham, and G. Lambaré, "Antileakage Fourier transform for seismic data regularization," *Geophysics*, vol. 70, pp. V87 – V95, 2005.
- [5] F. J. Herrmann and G. Hennenfent, "Non-parametric seismic data recovery with curvelet frames," *Geophysical Journal International*, vol. 173, pp. 233–248, 2008.
- [6] R. Shahidi, G. Tang, J. Ma, and F. J. Herrmann, "Application of randomized sampling schemes to curvelet-based sparsity-promoting seismic data recovery," *Geophysical Prospecting*, to appear.
- [7] M. Naghizadeh and M. Sacchi, "Beyond alias hierarchical scale curvelet interpolation of regularly and irregularly sampled seismic data," *Geophysics*, vol. 75, pp. WB189–WB202, 2010.
- [8] S. Hauser and J. Ma, "Seismic data reconstruction via directional weighted shearlet-regularized inpainting," *Preprint, TUK*, 2012.
- [9] S. Spitz, "Seismic trace interpolation in the f-x domain," *Geophysics*, vol. 56, pp. 785–794, 1991.
- [10] S. Crawley, J. Claerbout, and R. Clapp, "Interpolation with smoothly nonstationary prediction-error filters," in *69th Annual International Meeting, SEG, Expanded Abstracts, 1913-1916*, 1999.
- [11] M. Porsani, "Seismic trace interpolation using half-step prediction filters," *Geophysics*, vol. 64, pp. 1461–1467, 1999.
- [12] Y. Liu and S. Fomel, "Seismic data interpolation beyond aliasing using regularized nonstationary autoregression," *Geophysics*, vol. 76, pp. V69–V77, 2011.
- [13] M. Naghizadeh and M. Sacchi, "Seismic data reconstruction using multidimensional prediction filters," *Geophysical Prospecting*, vol. 58, pp. 157–173, 2010.
- [14] S. Trickett, L. Burroughs, A. Milton, L. Walton, and R. Dack, "Rank-reduction-based trace interpolation," *80th Annual meeting, SEG, Expanded Abstracts*, 2010.
- [15] V. Oropeza and M. Sacchi, "Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis," *Geophysics*, vol. 76, no. 3, pp. V25–V32, 2011.
- [16] R. Vautard and M. Ghil, "Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series," *Physica D*, vol. 35, pp. 395–424, 1989.
- [17] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, pp. 717–772, 2009.
- [18] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference*, June 2010.
- [19] J.-F. Cai, E. J. Candes, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, pp. 1956–1982, 2008.
- [20] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and bregman iterative methods for matrix rank minimization," *Mathematical Programming: Series A and B*, vol. 128, pp. 321–353, 2011.

- [21] J. Yang and X. Yuan, “An inexact alternating direction method for trace norm regularized least squares problem,” *Optimization Online*, 2010.
- [22] R. Keshavan, A. Montanari, and S. Oh, “Matrix completion from a few entries,” *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2980–2998, 2010.
- [23] Y. Nesterov, “A method of solving a convex programming problem with convergence rate $o(1/k^2)$,” *Soviet Mathematics Doklady*, vol. 27, pp. 372–376, 1983.
- [24] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Sciences*, pp. 183–202, 2009.
- [25] K.-C. Toh and S. Yun, “An advanced proximal gradient algorithm for nuclear norm regularized linear least squares problems,” pp. 615–640, 2009.
- [26] Z. Wen, W. Yin, and Y. Zhang, “Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm,” *Mathematical Programming Computation*, vol. 4, pp. 333–361, 2012.
- [27] H. Schaeffer and S. Osher, “A low patch-rank interpretation of texture,” *CAM Report*, Nov. 2011.
- [28] Y. Hua, “Estimating two-dimensional frequencies by matrix enhancement and matrix pencil,” *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2267–2280, 1992.