Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
Signal Processing 88 (2008) 1839–1851
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

Multichannel blind seismic deconvolution using dynamic programming☆

Alon Heimer, Israel Cohen*

Department of Electrical Engineering, Technion–Israel Institute of Technology, Technion City, Haifa 32000, Israel

Received 14 December 2006; received in revised form 4 December 2007; accepted 20 January 2008
Available online 4 February 2008

Abstract

In this paper, we present an algorithm for multichannel blind deconvolution of seismic signals, which exploits lateral continuity of earth layers by dynamic programming approach. We assume that reflectors in consecutive channels, related to distinct layers, form continuous paths across channels. We introduce a quality measure for evaluating the quality of a continuous path, and iteratively apply dynamic programming to find the best continuous paths. The improved performance of the proposed algorithm and its robustness to noise, compared to a competitive algorithm, are demonstrated through simulations and real seismic data examples.

Keywords: Wavelet estimation; Multichannel blind deconvolution; Seismic signal; Sparse reflectivity; Reflectivity estimation

1. Introduction

In seismic exploration, a short duration seismic pulse is transmitted from the surface, reflected from boundaries between underground earth layers, and received by an array of sensors on the surface [1]. The received signals, called seismic traces, are analyzed to extract information about the underground structure of the layers in the explored area [2,3]. Pre-processing is applied to the raw data in order to increase the signal-to-noise ratio (SNR) and attenuate surface waves that are unrelated to the underground structure. Subsequently, the traces can be modeled under simplifying assumptions as noisy outcomes of convolutions between reflectivity sequences (channels) and an unknown wavelet. The objective of multichannel blind seismic deconvolution is to estimate both the wavelet and the reflectivity sequences from the measured traces.

Single-channel blind deconvolution is generally an ill-posed problem, and requires some a priori information about the channels or the wavelet. The reflectivity sequence is often modeled as a Bernoulli–Gaussian random sequence, and second-order statistics may be used to partially reconstruct the input signal. Several methods based on high-order statistics have been developed [4,5], which require very long data to properly estimate the output statistics. Alternatively, the wavelet can be modeled as an autoregressive moving-average (ARMA) process, and a maximum likelihood estimator for the reflectivity can be derived [6].

☆Part of this work was presented in [1].

*Corresponding author. Tel.: +972 4 8294731; fax: +972 4 8295757.

E-mail addresses: heimer@tx.technion.ac.il (A. Heimer), icohen@ee.technion.ac.il (I. Cohen).

0165-1684/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2008.01.022

Multichannel blind deconvolution (see [7] and references therein, [8,9]) is often more advantageous and more robust than single-channel blind deconvolution. Sparsity of the reflectivity sequences may be used to cope with the ill-posed nature of the basic blind deconvolution problem [10,11], and to improve the performance of non-blind deconvolution methods [12]. Channel sparsity has been used in [10], together with an assumption of short wavelet, to formulate an efficient channel estimation method suitable for relatively short traces (see also [13]). Lateral continuity of the reflectors across channels is also used to further improve the channel estimates. Idier and Goussard [14] model the two-dimensional structure of the underground reflectivity as a Markov–Bernoulli random field, and impose lateral continuity to generate deconvolution results that are far superior to those obtainable by single-channel deconvolution methods. However, their estimator of the two-dimensional reflectivity pattern is suboptimal, since the dependency between columns is treated locally, i.e., each column of the reflectivity is estimated separately, under prior distributions given by the previous column whose estimate is held fixed.

In this paper, lateral continuity of reflectors across channels is combined with the blind deconvolution algorithm of Kaaresen and Taxt [10]. We employ dynamic programming [15,16] to find the shortest continuous paths of reflectors across channels, and develop an improved multichannel blind deconvolution algorithm for seismic signals, which exploits the lateral continuity of earth layers. Rather than measuring the increase in the fit to the data each single reflector yields, versus the decrease in sparsity of the channel estimates, we measure the increase in the fit to the data obtained by a complete continuous path of reflectors, versus the decrease in the sparsity of paths. This approach is an attempt to look at the data as a whole, and account for dependency between all columns in the data, and not only adjacent ones. The improved performance of the proposed algorithm and its robustness to noise, compared to the blind deconvolution algorithm of Kaaresen and Taxt, are demonstrated by using simulated and real seismic data examples. The rest of this paper is organized as follows: In Section 2, we describe the signal model and briefly review the blind deconvolution algorithm presented in [10]. In Section 3, we describe a dynamic programming method for finding the shortest continuous path in an image. In Section 4, we introduce a multichannel blind deconvolution algorithm, which exploits the continuity of earth layers and utilizes the dynamic programming approach. In Section 5, the performance of the proposed algorithm is demonstrated on simulated and real seismic data, and compared to an existing algorithm. Finally, in Section 6 we discuss the additional complexity of the proposed algorithm.

2. Signal model and basic blind deconvolution process

2.1. Signal model

We assume M received signals (traces), each generated by a single input signal $h[n]$ passing through a channel $x^{(m)}[n]$ and corrupted by additive uncorrelated noise $e^{(m)}[n]$. The output signal of channel m can be written as

$$z^{(m)}[n] = \sum_{k=0}^{K-1} h[k]x^{(m)}[n-k] + e^{(m)}[n]$$
$$\text{for } m = 1, 2, \ldots, M, \;\; n = 1, 2, \ldots, N. \tag{1}$$

The following assumptions are made for the wavelet h and the channels:

(1) All channels are excited by the same wavelet h.
(2) The wavelet h has a finite support of length K, which is shorter than the channel.
(3) Each channel is sparse, i.e., the number of non-zero elements (reflectors) in a channel is small relative to the channel's length.
(4) The dependency between different channels is modeled as follows. Let $P = (n_1, n_2, \ldots, n_M)$ be a vector of M integer line numbers such that n_1 is uniformly distributed in the range $[1, N]$, and $n_m|n_{m-1}$ is uniformly distributed in the range $[n_{m-1}-1, n_{m-1}+1]$ for $m = 2, 3, \ldots, M$. We call the vector P the "location vector of a single layer". Let $\widetilde{\mathbf{a}}(P) = (\widetilde{a}_1^P, \widetilde{a}_2^P, \ldots, \widetilde{a}_M^P)$ denote a vector such that $\widetilde{a}_1^P$ is normally distributed with zero mean and standard deviation σ_a, and such that $\widetilde{a}_m^P|\widetilde{a}_{m-1}^P$ is normally distributed with mean $r\widetilde{a}_{m-1}^P$ and standard deviation $\sigma_a\sqrt{1-r^2}$ for some constant r close to 1 and $m = 2, 3, \ldots, M$. This is a Markov model for the amplitudes of the reflectors along the layer with locations P. The two-dimensional reflectivity pattern is the stacking of the M channels as columns in an $N{*}M$ matrix X. Let $\{P_1, P_2, \ldots, P_L\}$ be the location vectors of L single layers and

$\{\widetilde{\mathbf{a}}(P_1), \widetilde{\mathbf{a}}(P_2), \ldots, \widetilde{\mathbf{a}}(P_L)\}$ be their amplitudes. The model for the structure of the matrix X is as follows: $x^{(m)}[n] = 0$ if $n_m \notin P_l$ for all $l = 1, 2, \ldots, L$ (i.e., no layer passes through that pixel), $x^{(m)}[n] = \widetilde{a}_m^{P_l}$ if P_l is the only location vector such that $n_m \in P_l$ (i.e., only one layer passes through that pixel), and $x^{(m)}[n] \sim \mathrm{N}(0, \sigma_{\mathrm{a}}^2)$ if there is more than one location vector that n_m belongs to (i.e., more than one layer passes through the pixel).

(5) The noise $e^{(m)}[n]$ is white, Gaussian, and independent of $h[n]$ and $x^{(m)}[n]$.

In [10] assumption 4 is replaced by the following two assumptions.

(4.1) The elements in a channel are independent and identically distributed with zero-mean Bernoulli–Gaussian distribution.

(4.2) Some degree of lateral continuity of the reflectors across channels is suggested, without stating any specific model for this continuity.

Assumption 3 makes it useful to write (1) in the following way:

$$z^{(m)}[n] = \sum_{q=1}^{Q} h[n - n_{m,q}] a_{m,q} + e^{(m)}[n], \tag{2}$$

where $n_{m,q}$ is the discrete time of reflection q in channel m, and $a_{m,q}$ is its amplitude. Notice that $a_{m,q}$ is the value of the qth non-zero element in the mth column of X. It equals $\widetilde{a}_m^P$ for some single layer P if this is the only layer that pases through this location, otherwise, if more than a single layer passes through this location, then $a_{m,q} \sim \mathrm{N}(0, \sigma_{\mathrm{a}}^2)$. The matrix representation of (2) is given by

$$\mathbf{z}^{(m)} = \mathbf{H}^{(m)} \mathbf{a}^{(m)} + \mathbf{e}^{(m)}, \tag{3}$$

where $\mathbf{z}^{(m)} = [z^{(m)}[0] \; z^{(m)}[1] \; \ldots \; z^{(m)}[N]]^{\mathrm{T}}$, $\mathbf{H}^{(m)}$ is a matrix with elements $\mathbf{H}_{nq}^{(m)} = h[n - n_{m,q}]$, and $\mathbf{e}^{(m)} = [e^{(m)}[0] \; e^{(m)}[1] \; \ldots \; e^{(m)}[N]]^{\mathrm{T}}$. For later use we adopt the notation in [10] and define the matrix $\mathbf{H}^{\mathrm{w}}$ as the columns of $\mathbf{H}$ corresponding to reflectors inside a certain time window w, $\mathbf{a}^{\mathrm{w}}$ represents the amplitudes of the reflectors inside the window, $\mathbf{H}^{\overline{\mathrm{w}}}$ contains the columns of $\mathbf{H}$ corresponding to reflectors outside the window, and $\mathbf{a}^{\overline{\mathrm{w}}}$ represents the amplitudes of the reflectors outside the window (the channel index m is omitted for convenience).

2.2. Basic single-channel iterative algorithm

The basic algorithm proposed in [10] is applicable to a single channel, and does not require measurements from an array of sensors. It includes an initialization step that provides a coarse approximation of the channel, followed by two steps, which are repeated iteratively until convergence. More specifically, the steps of the algorithm are as follows:

Step 1: Initialization. This step can be performed in several ways. One way, employed in [10], is to place initial reflectors in local maxima points of the absolute value of the received signal z. The reflectors amplitudes are set to the values of z in these locations.

After the initialization step, the following steps are iterated, until a convergence criterion is fulfilled.

Step 2: Wavelet estimation. Writing (1) in a matrix form (omitting the channel index m) we have

$$\mathbf{z} = \mathbf{X}\,\mathbf{h} + \mathbf{e}, \tag{4}$$

where $\mathbf{X}(n,k) = x[n-k]$, $n = 1, 2, \ldots, N$, $k = 1, 2, \ldots, K$, $\mathbf{z}$ and $\mathbf{e}$ are defined as in (3), and $\mathbf{h} = [h_0 : h_1 : \ldots : h_{K-1}]^{\mathrm{T}}$. The wavelet estimate is obtained by least squares fit to the received signal, given the channel estimate $\widehat{x}$:

$$\widehat{\mathbf{h}} = (\widehat{\mathbf{X}}^{\mathrm{T}} \widehat{\mathbf{X}})^{-1} \widehat{\mathbf{X}}^{\mathrm{T}} \mathbf{z}, \tag{5}$$

where $\widehat{\mathbf{X}}(n,k) = \widehat{x}[n-k]$, $n = 1, 2, \ldots, N$, $k = 1, 2, \ldots, K$. Since there is a scaling ambiguity between $\mathbf{h}$ and $\mathbf{a}$, a normalization step is applied to $\mathbf{h}$ after each estimation.

Step 3: Channel estimation. This step utilizes a suboptimal search for updating the channel estimate as follows: Let $\widehat{\mathbf{n}}_q$ be a vector representing the locations of all reflectors in the channel estimate in the qth iteration, and let $\widehat{\mathbf{a}}_q$ denote their amplitudes. Then for each reflector in the channel estimate perform the following:

(3.1) Center a window w around the reflector, and denote by $\widehat{\mathbf{n}}_q^{\mathrm{w}} = [n_1 \; n_2 \; \ldots \; n_p]^{\mathrm{T}}$ the locations of reflectors inside the window w.

(3.2) Examine and compare a set of alternative vectors $\{\widetilde{\mathbf{n}}_i^{\mathrm{w}}\}$ obtained from a predetermined set of alternatives[1] by performing the following

[1] The set of alternatives suggested in [10] is obtained by either removing the middle reflector, adding a single reflector somewhere inside the window, or moving the middle reflector to another location inside the window.

steps (3.2.1)–(3.2.5) for each alternative i:

(3.2.1) Calculate an approximation to the received signal using only existing reflectors outside the window w, and the current wavelet estimate $\widehat{h}$. Then subtract it from the received signal:

$$\widehat{\mathbf{z}}_o = \widehat{\mathbf{H}}_q^{\overline{\mathrm{w}}}\widehat{\mathbf{a}}_q^{\overline{\mathrm{w}}}, \tag{6}$$

$$\mathbf{z}_r = \mathbf{z} - \widehat{\mathbf{z}}_o. \tag{7}$$

(3.2.2) Construct a matrix $\widehat{\mathbf{H}}_i^{\mathrm{w}}$ according to the locations of reflectors inside the window in the examined alternative i, and find their new amplitudes giving the best approximation to the remainder of the data $\mathbf{z}_r$, given their locations and the current wavelet estimate:

$$\widehat{\mathbf{a}}_i^{\mathrm{w}} = [(\widehat{\mathbf{H}}_i^{\mathrm{w}})^{\mathrm{T}}\ \widehat{\mathbf{H}}_i^{\mathrm{w}}]^{-1}(\widehat{\mathbf{H}}_i^{\mathrm{w}})^{\mathrm{T}}\mathbf{z}_r. \tag{8}$$

(3.2.3) Calculate an approximation for $\mathbf{z}_r$ using the new reflector amplitudes from (8):

$$\widehat{\mathbf{z}}_{r,i} = \widehat{\mathbf{H}}_i^{\mathrm{w}}\ \widehat{\mathbf{a}}_i^{\mathrm{w}}. \tag{9}$$

(3.2.4) Calculate the inner product of $\widehat{\mathbf{z}}_{r,i}$ and $\mathbf{z}_r$,

$$q_i = \mathbf{z}_r^{\mathrm{T}}\,\widehat{\mathbf{z}}_{r,i}. \tag{10}$$

(3.2.5) Calculate a quality measure of the alternative by including a penalty term for the sparsity of the alternative:

$$\ell_i = q_i - \theta\, M_i^{\mathrm{w}}, \tag{11}$$

where M_i^{w} represents the number of reflectors inside the window w in this alternative, and θ is some predetermined weighting factor for the penalty term.

(3.3) After calculating the quality measure ℓ_i for all alternative location vectors, choose $i_{\mathrm{opt}} = \arg\max_i \ell_i$, and set the locations vector $\widetilde{\mathbf{n}}_{i_{\mathrm{opt}}}^{\mathrm{w}}$ as the locations of reflectors inside the window w in the channel estimate and the amplitudes $\widehat{\mathbf{a}}_{i_{\mathrm{opt}}}^{\mathrm{w}}$ as their amplitudes.

(3.4) Proceed to the next reflector in this channel estimate (Step 3.1).

Steps 2 and 3 are performed iteratively until some convergence criterion is fulfilled. For example, we may assume convergence when the reflectors locations remain the same and the reflectors amplitudes variation is insignificant, i.e., $\widehat{\mathbf{n}}_q = \widehat{\mathbf{n}}_{q-1}$, and $\|\widehat{\mathbf{a}}_q - \widehat{\mathbf{a}}_{q-1}\|/\|\widehat{\mathbf{a}}_q\| < 0.05$. It is shown in [17] that this process produces a suboptimal maximum a posteriori estimator of $\mathbf{h}, \mathbf{n}$ and $\mathbf{a}$, given $\mathbf{z}$.

2.3. Expansion for multichannel blind deconvolution

An expansion of the single-channel algorithm is proposed in [10] for the case of multichannel measurements, where the assumption of lateral continuity of the reflectors across channels can be made. The initialization step is the same as for the single channel, and performed for each channel independently. The estimation of the wavelet h is also carried out in the same manner, except that the least squares fit to the data is performed with respect to all traces simultaneously:

$$\widehat{\mathbf{h}} = \mathbf{B}^{-1}\mathbf{u}, \tag{12}$$

where $\mathbf{B} = \sum_{m=1}^{M}\widehat{\mathbf{X}}_m^{\mathrm{T}}\widehat{\mathbf{X}}_m$, $\mathbf{u} = \sum_{m=1}^{M}\widehat{\mathbf{X}}_m^{\mathrm{T}}\mathbf{z}_m$, and $\widehat{\mathbf{X}}_m(n,k) = \widehat{x}_m[n-k]$. The resulting wavelet estimate minimizes the mean squared error (MSE) between the received data and the reconstructed data given the channel estimate.

The channel estimation step is performed for each channel at a time in a similar way as before, except that the quality measure ℓ is modified to favor reflectors that form continuous sequences with reflectors in adjacent channels. The new measure ℓ suggested in [10] for the multichannel case is given by

$$\ell = q - \theta M^{\mathrm{w}} + v^{-}M^{-} + v^{\backslash}M^{\backslash} + v^{/}M^{/} - v^{|}M^{|} - v^{\|}M^{\|}, \tag{13}$$

where M^{-} represents the number of reflectors that have neighbor reflectors in adjacent channels in the same time location (horizontal layers), $M^{/}$ and $M^{\backslash}$ represent the number of reflectors that have neighbor reflectors in smaller or greater time locations by 1 sample (descending or ascending layers, respectively, with a slope of one sample per channel), v^{-}, $v^{\backslash}$, and $v^{/}$ are weights that affect the degree to which the algorithm tends to favor these kind of continuities, and $M^{|}$ and $M^{\|}$ represent the number of reflector pairs that are one or two samples apart in the same channel. The last two terms in (13) prevent an undesirable clustering of reflectors in case of several closely spaced layers. Subtraction of these terms introduces a penalty for channel estimates that produce such clustering. In the rest of this paper we refer to this multichannel version of the algorithm as the basic algorithm.

3. Dynamic programming for finding the shortest continuous path

Dynamic programming is an effective way to find a global minimum in some non-convex optimization problems. We now briefly describe the problem of finding the shortest continuous path across an image, and the solution by dynamic programming [16].

3.1. Problem formulation

Assume we have a gray level image of size $N \times M$. The problem is to find a "path", i.e., a sequence of M pixels $\{(n_m, m)\}_{m=1}^{M}$ one in each column, such that the following two conditions are satisfied:

(1) $|n_m - n_{m+1}| \leqslant d$ for all $m = 1, 2, \ldots, M-1$, where d is some small positive integer constant. We refer to this condition as the "continuity condition", and to a path for which this condition is satisfied as a continuous path.

(2) The path is the "shortest" among all continuous paths, in the sense that a certain length measure is minimized for this path among all the continuous paths.

A common length measure for a given path is the sum of gray levels of the pixels, which the path passes through. Fig. 1 demonstrates an image and the shortest path obtained for $d = 1$ (the path is marked with white circles).

In this work we are interested in a length criterion that measures not only the total intensity of the path (as represented by the sum of gray levels) but also the degree to which the gray levels along the path change slowly (since we assume that the amplitude of a reflection from a layer boundary changes slowly along the boundary, as reflected by the Markov model for the amplitudes presented earlier).

3.2. Extraction of shortest path

We now describe a dynamic programming algorithm using a general length measure defined as follows: Let (n, m) be the coordinates of a pixel in row n and column m. Let $P = \{(n_1, m_1), (n_2, m_2), \ldots\}$ denote a path of pixels. Let S_P be the length of path P. We define a concatenation of a pixel (n, m) to a path P of length $m-1$ as the operation of creating a new path $\widetilde{P}$ of length m whose $m-1$ first pixels are the $m-1$ pixels of P and whose last pixel is the pixel (n, m). We use the notation $\widetilde{P} = (P \vdots (n, m))$ to describe the concatenation. Let $s(n, m, P)$ be a value associated with the pixel (n, m) with respect to the path P so that $S_{\widetilde{P}} = S_P + s(n, m, P)$, i.e., the concatenation of the pixel (n, m) to the path P increases the length of the path by $s(n, m, P)$. We assume that the value $s(n, m, P)$ is defined for every combination of pixel (n, m) and path P. We use $\{\}$ to denote the empty path, and define its' length as 0. Let $P_{n,m} = \{(n_1, 1), (n_2, 2), \ldots, (n, m)\}$ denote a continuous path starting at column 1 and ending at pixel (n, m). Denote by $P^o_{n,m}$ the continuous path that starts at column 1 and ends at pixel (n, m) with minimum length among all such paths. The algorithm starts with the following initialization:

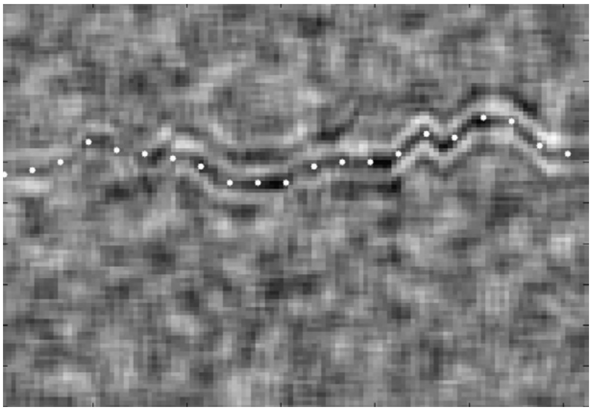

Fig. 1. A noisy gray level image and the shortest continuous path (denoted by white circles) obtained by dynamic programming. The path "length" is the sum of gray levels along the path, and the continuity parameter d is 1.

$$P^o_{n,1} = (n, 1) \quad \text{for } n = 1, 2, \ldots, N, \tag{14}$$

$$S_{P^o_{n,1}} = p(n, 1, \{\}) \quad \text{for } n = 1, 2, \ldots, N. \tag{15}$$

Then for each column $m = 2, \ldots, M$ compute

$$S_{P^o_{n,m}} = \min_{n-d \leqslant k \leqslant n+d} S_{P^o_{k,m-1}} + s(n, m, P_{k,m-1}) \quad \text{for } n = 1, 2, \ldots, N, \tag{16}$$

$$P^o_{n,m} = (P^o_{k^o,m-1} \vdots (n, m)) \quad \text{for } n = 1, 2, \ldots, N, \tag{17}$$

where k^o denotes the value of k achieving the minimum in (16). After the column M is processed, we obtain the desired path P_{opt} by

$$n^o = \arg\min_n S_{P^o_{n,M}} \tag{18}$$

$$P_{\text{opt}} = P^o_{n^o,M}. \tag{19}$$

This procedure achieves the continuous path from columns 1 to M with minimal length.

4. Combining dynamic programming and multichannel blind deconvolution

4.1. Proposed approach

According to the two-dimensional reflectivity model presented in Section 2, we assume that any non-zero element in a channel is a member of a path of non-zero elements across the channels, which satisfies the continuity condition (Condition 1 in Section 3). In the basic algorithm, a decision whether or not to add a reflector to a channel estimate, is made according to the increase in the fit to the data compared to the decrease in the sparsity measure (as reflected in the quality measure ℓ in (11) or (13)). In the multichannel version of the algorithm, a measure of local continuity is added to the decision criterion, which encourages reflectors that are members of a local continuous sequence. The approach presented here allows the process to add, not single reflectors, but complete continuous paths of reflectors to the image of channel estimates. Rather than measuring the increase in the fit to the data each single reflector yields, versus the decrease in sparsity of the channel estimate, we measure the increase in the fit to the data obtained by a complete continuous path of reflectors, versus the decrease in sparsity of paths, i.e., number of continuous reflector paths.

According to the two-dimensional reflectivity model presented in Section 2, all paths of reflectors in the true channels are continuous with continuity parameter $d = 1$. The steps of the new algorithm are as follows:

Step 1: Initialization. The initialization is carried out by finding several continuous paths (with parameter $d = 1$) along which the sum of gray levels is maximal or minimal among all continuous paths in the same region of the data. Finding the path with maximal sum of gray levels (maximal length) can be done by finding the path with minimal length of the negative image. The steps of this initialization are as follows:

(1.1) Use dynamic programming to find the continuous path of minimal length in the original image.

(1.2) Similarly, find the path of maximal length.

(1.3) Choose the path from the above two steps for which the absolute value of the length is maximal and add it to the channel estimate, setting the amplitudes along it to be the value of the data in those locations.

(1.4) Let $S \triangleq \{(n,m) : |n - n_m| < D\}$ be a strip of a certain predetermined width D around the path $\{(n_m, m)\}_{m=1}^M$ found in the previous step. Then, for finding the next extremum path, we replace the values of the true data in that strip by zeros, and subsequently extract the extremum path (the received signal z is randomly distributed with zero mean, and therefore replacing the data values with zeros ensures that this strip will not be a part of the following paths in the initialization step, since zeros are not contributing to an extremum value).

(1.5) If the number of paths found so far is less than a predetermined number, go to step 1.1.

At the end of the initialization step, we replace all the zero values, inserted in Step 1.4, back to their original values.

Step 2: Wavelet estimation. After initialization of the channel estimate, the wavelet estimate is obtained in the same manner as in the basic algorithm (Section 2), which is simply the least squares fit to the data given the current channel estimate and the received signals, and normalizing the estimated wavelet afterwards.

Step 3: Channel estimation. The channel estimates are updated by examining and comparing a few alternatives that include adding a new continuous path of reflectors, or translating an existing path. Since the number of possible continuous paths is much larger than the number of possible single reflectors,[2] we use again dynamic programming as described in Section 3, to find the continuous path that maximizes a criterion that will be defined shortly. That path only is examined as a candidate for inclusion or substitution of other

[2] The number of continuous paths is approximately $N(2d+1)^{M-1}$, while the number of possible reflectors is $N\,M$.

paths. The criterion that is used for finding a path by dynamic programming is explained in the following two steps of the algorithm.

Step 3.1: In this step, we calculate a probability measure for the existence of a new reflector at every possible location in the two-dimensional reflectivity pattern (additional to the currently estimated reflectors), and evaluate the corresponding amplitude given that a reflector indeed exists in that location. For each channel, we go over all locations in this channel, and for every location, we create an alternative reflectivity by placing a new reflector at this location. The probability measure and the amplitude of the new reflector are calculated independently of other channels, in the same manner as in [10], as described in Section 2. During this step we do not assume any prior for the locations or the amplitudes, hence this is actually a maximum likelihood estimation as follows:

Let n^m denote the vector containing the locations of reflectors in column m according to the current reflectivity estimation. The set of examined alternatives for this reflectivity column are denoted by $\{\widetilde{n}_i^m\}_{i=1}^N$ where $\widetilde{n}_i^m$ is created by placing a new reflector in location i. For each $i \in [1, N]$ we calculate: $l_i^m = \max_{\mathbf{a}_i^m} \log(p(\mathbf{z}^m|\mathbf{a}_i^m, \widetilde{\mathbf{n}}_i^m))$ and $\widehat{\mathbf{a}}_i^m = \arg\max_{\mathbf{a}_i^m} \log(p(\mathbf{z}^m|\mathbf{a}_i^m, \widetilde{\mathbf{n}}_i^m))$. Let us denote by $\widehat{a}_i^m$ the amplitude of the reflector from $\widehat{\mathbf{a}}_i^m$ that is located in line i (this is the amplitude of the new reflector we placed). This first stage yields a probability measure for the existence of a new reflector at each possible location, at each column, and maximum likelihood estimate of the reflector amplitude.

The details of this calculation are as follows: For each column m and location i:

(3.1.1) Temporarily place a new reflector at line i.
(3.1.2) Define a window centered at line i.
(3.1.3) Calculate the contribution to the data made by existing reflectors outside the window, using the current wavelet estimate, and subtract it from the data as in (6) and (7).
(3.1.4) Find the amplitudes of reflectors inside the window giving the best fit to the remainder of the received signal $\mathbf{z}_r^m$, given their locations, and the current wavelet estimate as in (8).
(3.1.5) Calculate an approximation for $\mathbf{z}_r^m$ using the new reflector amplitudes from (8):

$$\widehat{\mathbf{z}}_{r,i}^m = \widehat{\mathbf{H}}_i^{\mathrm{w}} \widehat{\mathbf{a}}_i^{\mathrm{w}}, \tag{20}$$

where $\widehat{\mathbf{z}}_{r,i}^m$ is a projection of $\mathbf{z}_r^m$ on the columns of $\widehat{\mathbf{H}}_i^{\mathrm{w}}$ and $\widehat{\mathbf{a}}_i^{\mathrm{w}}$ are the coefficients of each column of $\widehat{\mathbf{H}}_i^{\mathrm{w}}$ in the projection.
(3.1.6) Calculate the inner product $\ell_i^m = (\mathbf{z}_r^m)^{\mathrm{T}} \widehat{\mathbf{z}}_{r,i}^m$as in (10):
Since it is assumed that $\mathbf{z}_r^m = \mathbf{H}_i^{\mathrm{w}} \mathbf{a}_i^{\mathrm{w}} + \mathbf{e}^{\mathrm{w}}$ where $\mathbf{e}^{\mathrm{w}}$ is white Gaussian noise, then $\ell_i^m = \max_{\mathbf{a}_i^{\mathrm{w}}} \log(p(\mathbf{z}_r^m|\mathbf{a}_i^{\mathrm{w}}, \widetilde{\mathbf{n}}_i^m))$ and $\widehat{\mathbf{a}}_i^{\mathrm{w}} = \arg\max_{\mathbf{a}_i^{\mathrm{w}}} \log(p(\mathbf{z}_r^m|\mathbf{a}_i^{\mathrm{w}}, \widetilde{\mathbf{n}}_i))$.

Step (3.1) generates an image of "qualities" which is referred to as the "qualities image", representing the log of probability of a new reflector existing at each possible location at each channel given the existing reflectors.

Step 3.2: Now, for estimating the two-dimensional reflectivity, we recall the priors for the locations of reflectors and their amplitudes in the two-dimensional reflectivity model. We seek a vector of integer line numbers:

$$\begin{aligned}
&(\hat{i}_1, \hat{i}_2, \ldots, \hat{i}_M) \\
&\quad = \arg\max_{i_1,\ldots,i_M} p(\hat{\mathbf{a}}_{i_1}^1, \hat{\mathbf{a}}_{i_2}^2, \ldots, \hat{\mathbf{a}}_{i_M}^M, \\
&\qquad \hat{\mathbf{n}}_{i_1}^1, \hat{\mathbf{n}}_{i_2}^2, \ldots, \hat{\mathbf{n}}_{i_M}^M | \mathbf{z}^1, \mathbf{z}^2, \ldots, \mathbf{z}^M) \\
&\quad = \arg\max_{i_1,\ldots,i_M} p(\mathbf{z}^1, \ldots, \mathbf{z}^M | \hat{\mathbf{a}}_{i_1}^1, \ldots, \hat{\mathbf{a}}_{i_M}^M, \hat{\mathbf{n}}_{i_1}^1, \ldots, \hat{\mathbf{n}}_{i_M}^M) \\
&\qquad \times p(\hat{\mathbf{a}}_{i_1}^1, \ldots, \hat{\mathbf{a}}_{i_M}^M | \hat{\mathbf{n}}_{i_1}^1, \ldots, \hat{\mathbf{n}}_{i_M}^M) p(\hat{\mathbf{n}}_{i_1}^1, \ldots, \hat{\mathbf{n}}_{i_M}^M) \\
&\quad = \arg\max_{i_1,\ldots,i_M} p(\mathbf{z}^1, \ldots, \mathbf{z}^M | \hat{\mathbf{a}}_{i_1}^1, \ldots, \hat{\mathbf{a}}_{i_M}^M, \hat{\mathbf{n}}_{i_1}^1, \ldots, \hat{\mathbf{n}}_{i_M}^M) \\
&\qquad \times p(\hat{\mathbf{a}}_{i_1}^1 \ldots, \hat{\mathbf{a}}_{i_M}^M | \hat{\mathbf{n}}_{i_1}^1, \ldots, \hat{\mathbf{n}}_{i_M}^M) \\
&\quad = \arg\max_{i_1,\ldots,i_M} p(\mathbf{z}^1 | \hat{\mathbf{a}}_{i_1}^1, \hat{\mathbf{n}}_{i_1}^1) p(\hat{\mathbf{a}}_{i_1}^1 | \hat{\mathbf{n}}_{i_1}^1) \prod_{m=2}^{M} p(\mathbf{z}^m | \hat{\mathbf{a}}_{i_m}^m, \hat{\mathbf{n}}_{im}^m) \\
&\qquad \times p(\hat{\mathbf{a}}_{i_m}^m | \hat{\mathbf{a}}_{i_{m-1}}^{m-1}, \hat{\mathbf{n}}_{i_m}^m).
\end{aligned} \tag{21}$$

Here we incorporate the prior for the amplitudes that is given by the Markov model. Omission of the term $p(\widehat{\mathbf{n}}_{i_1}^1, \widehat{\mathbf{n}}_{i_2}^2, \ldots, \widehat{\mathbf{n}}_{i_M}^M)$ is justified since all continuous paths are equally probable. In order to calculate the term $p(\widehat{\mathbf{a}}_{i_m}^m | \widehat{\mathbf{a}}_{i_{m-1}}^{m-1}, \widehat{\mathbf{n}}_{i_m}^m)$ in (21), we need to keep track of all paths that already exist in the two-dimensional reflectivity estimation, at each step, which is a cumbersome task. Usually placing a new reflector and updating its neighbors, causes their amplitudes to change only slightly, hence we can replace the term $p(\widehat{\mathbf{a}}_{i_m}^m | \widehat{\mathbf{a}}_{i_{m-1}}^{m-1}, \widehat{\mathbf{n}}_{i_m}^m)$ with the term $p(\widehat{a}_{i_m}^m | \widehat{a}_{i_{m-1}}^{m-1})$, i.e., incorporate the Markov model with only the new path to be added to the reflectivity estimate rather than all the previously extracted paths. So, finally, if we take the log of (21), use the notation of ℓ_i from step 3.1.6, use the Markov model to evaluate $p(\widehat{a}^m_{i_m}|\widehat{a}^{m-1}_{i_{m-1}})$ and multiply the criterion by -1 we obtain

$$(\widehat{i}_1, \widehat{i}_2, \ldots, \widehat{i}_M) = \arg \min_{i_1,i_2,\ldots,i_M} -\sum_{m=1}^{M} \ell^m_{i_m} + \frac{(\widehat{a}^1_{i_1})^2}{2\sigma_{\mathrm{a}}^2} + \sum_{m=2}^{M} \frac{(\widehat{a}^m_{i_m} - r\widehat{a}^{m-1}_{i_{m-1}})^2}{2(1-r^2)\sigma_{\mathrm{a}}^2}.$$

This vector must also be a valid location vector of a layer, so it must represent a continuous path.

Now we employ dynamic programming as described in Section 3 to find a continuous path (with parameter $d = 1$) in the qualities image, that minimizes this criterion. This is achieved by defining the measure $s(n, m, P)$ in the dynamic programming as follows:

$$s(n, 1, \{\}) = -\ell^1_{n_1} + \frac{(\widehat{a}^1_{n_1})^2}{2\sigma_{\mathrm{a}}^2},$$

$$s(n, m, P) = -\ell^m_{n_m} + \frac{(\widehat{a}^m_{n_m} - r\widehat{a}_{\mathrm{last}}(P))^2}{2(1-r^2)\sigma_{\mathrm{a}}^2},$$

where $\widehat{a}_{\mathrm{last}}(P)$ is the best estimated amplitude for a new reflector placed in the last pixel of the path P.

As long as the channel estimate includes less than the known number of layers, we add new paths to the two-dimensional reflectivity estimate. Each additional path is inserted into a FIFO list, and when we reach the desired number of layers, we sequentially perform the following steps:

(a) Remove the next existing path from the beginning of the list and from the channel estimate.
(b) Find a new best path.
(c) Add the new path to the end of the list and to the channel estimate.

This procedure is similar to the one described in [10], since we actually move each path to a different better location. If a path current location is optimal, then the path is selected again at the same location. The advantage of this procedure is that tuning of a sparsity parameter is unnecessary until the channel estimate has the desired number of reflectors. Tuning of a sparsity parameter depends on the SNR, and involves several executions of the whole process.

Steps 2 and 3 are iterated until a certain stopping criterion is satisfied. In the production of the results in this work the stopping criterion was a predetermined number of iterations.

4.2. Implementation notes

A new reflector affects the optimal values of neighboring reflectors, as explained in the beginning of this section. Hence, after we find the best path of reflectors and add it to the channel estimate, we have to update all reflectors that are in the neighborhood of this path, which requires some management. We propose to temporarily save optimal values of reflectors for all possible locations and after the best path is selected, use the relevant values to update the necessary reflectors. When a path is removed from the channel estimate, we recalculate the optimal values of reflectors using (8), noting that $\widehat{\mathbf{H}}^{\mathrm{w}}$ does not include a column for the removed reflector.

When we calculate the quality of a reflector at a given location, we also calculate the quality of existing reflectors, so that the latter reflectors may be selected as members of a new path. This facilitates the search for a new path which crosses or merges with an existing path. The qualities of existing reflectors are calculated in the same way as if a given reflector is new. A table holding the number of paths that go through each location in the channel estimate is updated every time a path is added or removed (for example, if a path which crosses another path is removed, the crossing point continues to hold a reflector).

4.3. Optimality of the channel estimate

To demonstrate the advantage of the proposed algorithm over the basic algorithm, consider the following example. Suppose that the true reflectors in a certain area are as shown in Fig. 2(a), where each column represents a channel and the non-zero elements are marked in black. Let ℓ_a, ℓ_b, ℓ_c and ℓ_d represent the values of the measure ℓ, as defined in (13), for the reflectors shown in Figs. 2(a), (b), (c) and (d), respectively. Assume that

$$\ell_a > \ell_b > \ell_c \quad \text{and} \quad \ell_b > \ell_d.$$

Then, maximization of the measure ℓ by using the basic algorithm may yield misplaced reflectors as shown in Fig. 2(b), since the search method allows changing the location of a single reflector at a time, and the algorithm is stuck at a local maximum of the measure ℓ.

On the other hand, by using the proposed algorithm and setting d to 1, the channel estimates

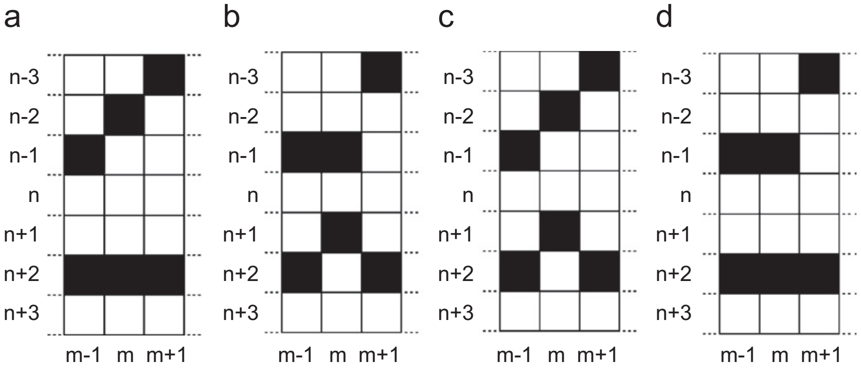

Fig. 2. An example demonstrating a local maximum of the measure ℓ, where the basic algorithm gets stuck. (a) True locations of reflectors ($\ell = \ell_a$); (b) reflector locations obtained by using the basic algorithm ($\ell = \ell_b < \ell_a$). To increase ℓ from ℓ_b to ℓ_a, the algorithm needs to go through either the channel estimate shown in (c) or (d). However, if ℓ for (c) and (d) is lower than ℓ_b, then the basic algorithm is unable to produce the correct reflectors.

shown in Figs. 3(b) and (d) are not possible, since they contain non-continuous paths. Therefore, maximization of the measure ℓ would yield the true reflectors as shown in Fig. 2(a). In this example, the continuity constraint excludes a local maximum of the measure ℓ, which enables the algorithm to produce the correct reflectors. In general, assuming that the basic method of finding reflectors without the continuity constraint is suitable for most reflectors, the continuity constraint further increases the measure ℓ by excluding local maxima, as demonstrated above. However, if the basic method misplaces many reflectors, but the displacement is small in randomly up and down directions, then we may still expect that the continuity constraint would increase the measure ℓ, since the algorithm searches for a continuous path closest to the misplaced reflectors. If the displacement is large or tends to one direction (up or down), then the continuity constraint would generally be insufficient.

4.4. Parameter selection

The most important parameter that affects the performance of the proposed algorithm is the number of expected reflections. This number may be set in advance based on prior knowledge. A better strategy is choosing some threshold for the increase in the fit to the data gained by a given path, and including only paths that contribute “data-fit” above that threshold. The threshold parameter can be set adaptively during the process according to the distribution of data-fit contribution from continuous paths.

5. Experimental results

5.1. High noise example

In this simulation 30 traces of 511 samples length are generated by convolving a certain wavelet with a pattern of reflectors. The pattern of reflectors, shown in Fig. 3(b), is created according to the new model presented in Section 2. The number of layers is taken from a binomial distribution with parameters $N = 511$ and $\lambda = 0.05$, and the parameter r of the Markov model is 0.95.

The wavelet that is used in our simulations is shown in solid line in Fig. 3(e). The traces are corrupted by high level white Gaussian noise ($\mathrm{SNR} = -15\,\mathrm{dB}$). The SNR is defined by $\mathrm{SNR} \triangleq (\sigma_{\mathrm{a}}^2/\sigma_e^2)\sum_k h^2[k]$, where σ_{a} is the standard deviation of the reflectors amplitudes, and σ_e is the standard deviation of the noise. A comparison is made between the basic multichannel algorithm proposed in [10] (with consideration of local continuities of reflectors), and the new version proposed in this work. The results are presented in Fig. 3.

It is clear from the results that the proposed algorithm recovers the true reflectors more accurately than the basic algorithm. Furthermore, the wavelet estimate obtained by the proposed algorithm is characterized by lower MSE ($\mathrm{MSE} = \|\mathbf{h} - \hat{\mathbf{h}}\|^2$). A comparison of Figs. 3(c) and (b) shows that the basic algorithm generates false reflectors in some areas, and misdetects true reflectors in other areas. Hence, the improvement by using the proposed algorithm is not related to the specific choice of sparsity parameter in the basic algorithm, since a

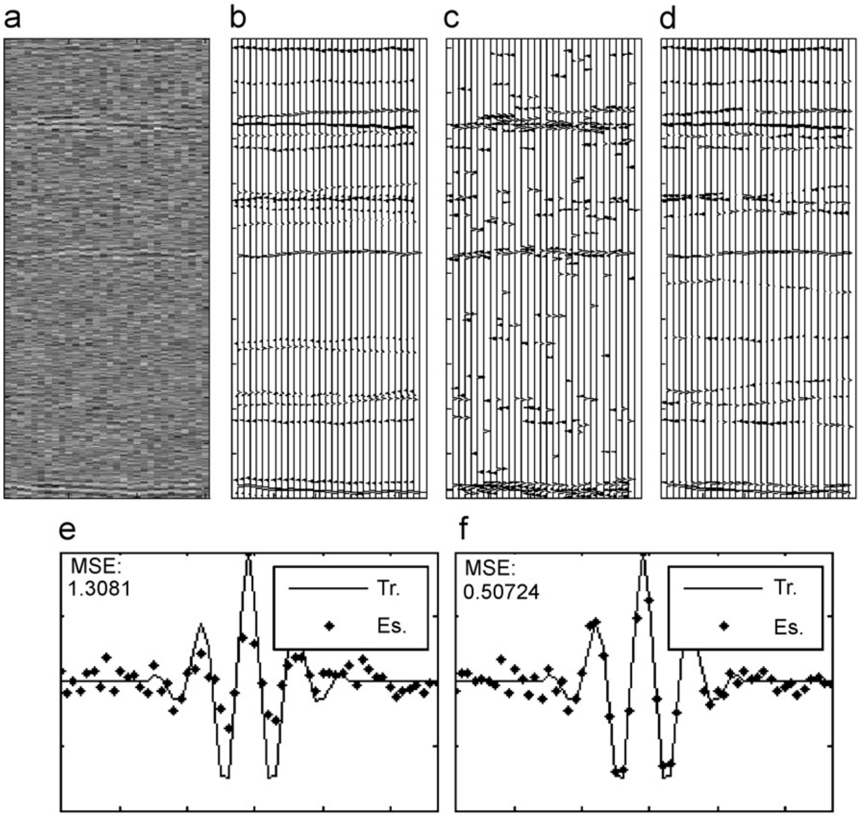

Fig. 3. Results of multichannel blind deconvolution obtained by using the basic and new algorithms in a high noise situation. (a) Received noisy traces (SNR = −15 dB), (b) true channels, (c) channel estimates obtained by using the basic algorithm, (d) channel estimates obtained by using the proposed algorithm, (e) wavelet estimates obtained by using the basic algorithm, (f) wavelet estimates obtained by using the proposed algorithm.

larger sparsity parameter would cause additional true reflectors to disappear, and a smaller sparsity parameter would yield additional false reflectors.

5.2. *No noise example*

In this simulation 30 traces of 511 samples length are generated by convolving the same wavelet as in the previous example with a different pattern of reflectors, shown in Fig. 4(b). The traces do not contain noise. Applying the basic and proposed algorithms to the data shown in Fig. 4(a) yields channel and wavelet estimates as shown in Figs. 4(c)–(f).

The continuity constraint enables the proposed algorithm to avoid some local maxima of the optimality measure ℓ by excluding solutions with non-continuous paths, as can be seen for example in the middle part of the reflectivity pattern. The absence of noise in this example enables to isolate this effect from the effects of noise, but naturally this effect also exists in the presence of noise.

5.3. *Real data results*

Fig. 5(a) shows real seismic data (courtesy of GeoEnergy Inc., Texas) containing 110 traces of 551 samples long. The channel and wavelet estimates obtained by using the basic and proposed algorithms are presented in Fig. 5. Since the true layer structure is unknown, one can only appreciate the continuous nature of the channel estimates obtained by using the proposed algorithm. After the end of the computations it was found out that the wavelet was zero phase, which clearly makes a compelling argument for the new proposed algorithm.

6. Complexity of the proposed algorithm

Both the basic and the proposed algorithms involve calculation of the quality measure ℓ for new reflectors in all possible locations. This part of the algorithms is the most time consuming, but is performed once for every scan of channel estimates.

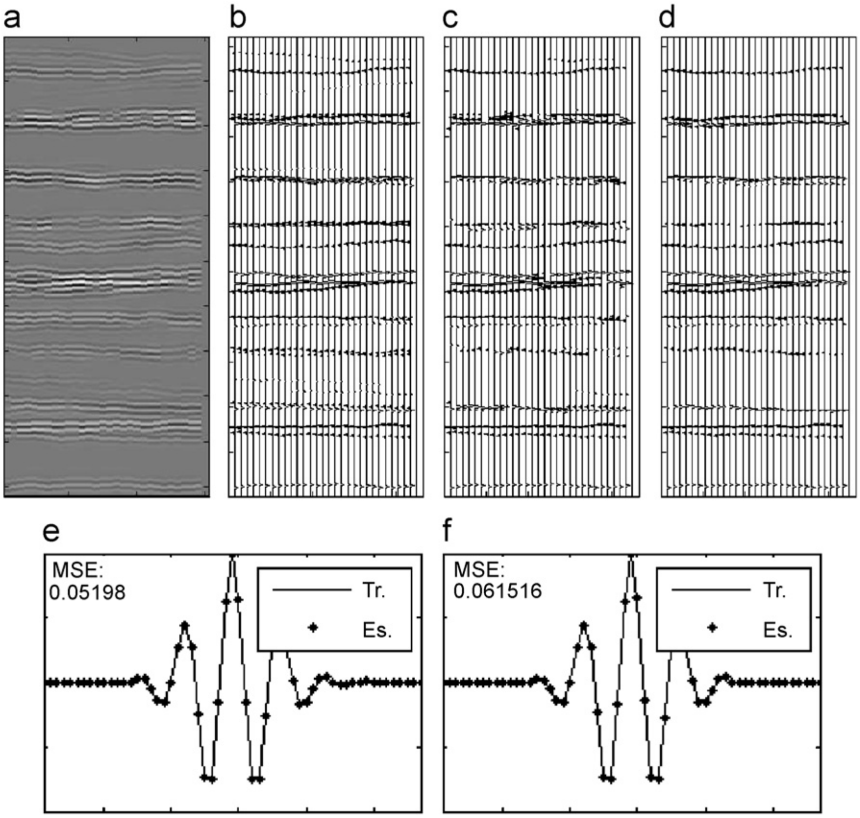

Fig. 4. Results of multichannel blind deconvolution obtained by using the basic and new algorithms in a clean situation. (a) Received clean traces, (b) true channels, (c) channel estimates obtained by using the basic algorithm, (d) channel estimates obtained by using the proposed algorithm, (e) wavelet estimates obtained by using the basic algorithm, (f) wavelet estimates obtained by using the proposed algorithm.

In [10], the reflectors are updated during the calculation, while in the proposed method the computation of ℓ is completed for all channels, and only afterwards the best path is included in the channel estimate. Since a single path may be supplemented after each calculation, the proposed algorithm generally requires more iterations to converge when compared to the basic algorithm. However, assuming that distant paths are sufficiently independent, i.e., jointly adding two distant paths is approximately the same, in terms of quality, as adding each path separately, we may accelerate the convergence of the algorithm by adding or removing after each iteration a few distant paths.

7. Limitations of the model

The main limitation of the proposed model and algorithm is the assumption of paths that start at column 1 and end at column M. In real situations layers can end or start within the region of interest, leave the region of interest at its top or bottom, and also merge or split. These situations present a problem for the proposed algorithm. However, if reflectors' amplitudes are sufficiently large, a continuous path of reflectors can still be extracted which contains also an artificial continuation of the path through the noise-only area. The artificial extension of the path may also cross areas of other layers. An improved model that accounts for such phenomena and appropriate modification of the estimation procedure are topics for future research.

8. Conclusion

We have presented an improved algorithm for multichannel blind deconvolution in seismic applications, where reflectors in channels are sparse and laterally continuous. The improved performance, compared to that obtained by an existing algorithm, is achieved by combining the existing approach

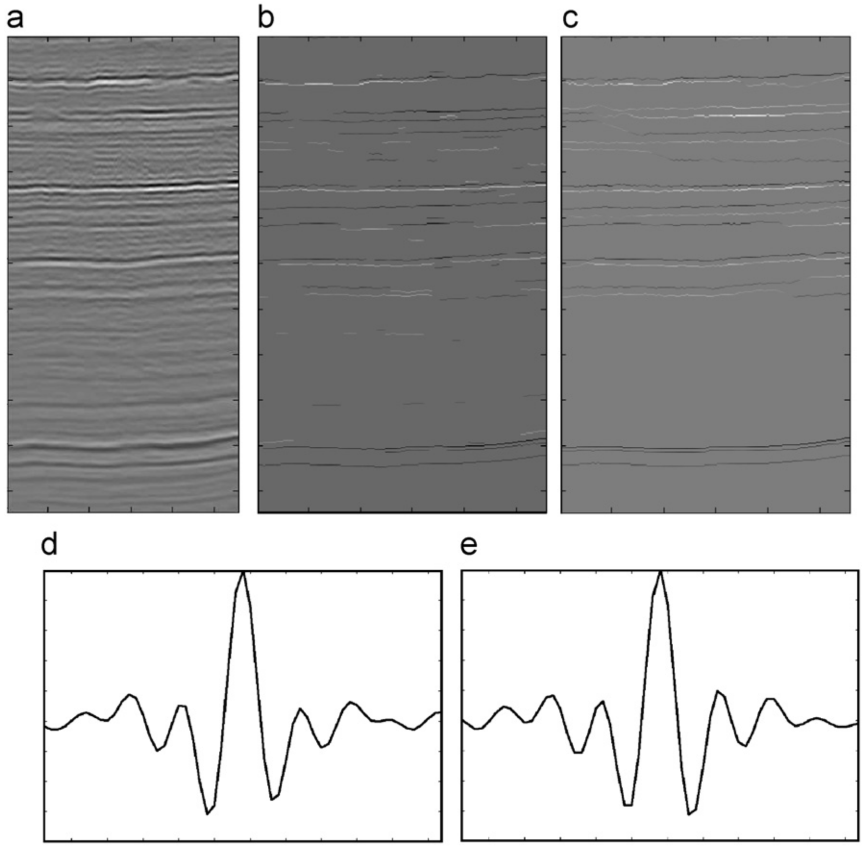

Fig. 5. (a) Real seismic data containing 110 traces of 551 samples long, (b) channel estimates obtained by using the basic algorithm, (c) channel estimates obtained by using the proposed algorithm, (d) wavelet estimates obtained by using the basic algorithm, (e) wavelet estimates obtained by using the proposed algorithm.

with a dynamic programming method for finding continuous lines in images. We have demonstrated the robustness of the proposed algorithm to high noise level, and the mechanism that enables excluding local maxima of the quality measure ℓ. In return, the proposed algorithm is characterized by higher computational complexity and slower convergence rate than the existing algorithm. In some applications the reflectors paths may vary more rapidly between channels, which necessitates increasing the parameter d. However, increasing the parameter d relaxes the continuity constraint and accordingly may reduce the benefits anticipated from the proposed approach.

Acknowledgment

The authors thank Dr. Anthony Vassiliou of GeoEnergy Inc., Houston, Texas, for valuable discussions and helpful suggestions.

References

[1] A. Heimer, I. Cohen, A. Vassiliou, Dynamic programming for multichannel blind seismic deconvolution, in: Proceedings of the Society of Exploration Geophysicist International Conference, Exposition and 77th Annual Meeting, San Antonio, 23–28 September 2007.

[2] A.J. Berkhout, The seismic method in the search for oil and gas: current techniques and future developments, Proc. IEEE 74 (8) (August 1986) 1133–1159.

[3] C.B. Wason, J.L. Black, G.A. King, Seismic modeling and inversion, Proc. IEEE 72 (10) (October 1984) 1385–1393.

[4] J.M. Mendel, Tutorial on high-order statistics (spectra) in signal processing and system theory: theoretical results and some applications, Proc. IEEE 79 (3) (March 1991) 278–304.

[5] G.D. Lazear, Mixed-phase wavelet estimation using fourth order cumulants, Geophysics 58 (7) (July 1993) 1042–1051.

[6] J.M. Mendel, J. Kormylo, F. Aminzadeh, J.S. Lee, A novel approach to seismic signal processing and modeling, Geophysics 46 (10) (October 1981) 1398–1414.

[7] L. Tong, S. Perreau, Multichannel blind identification: from subspace to maximum likelihood methods, Proc. IEEE 86 (10) (October 1998) 1951–1968.

[8] G. Xu, H. Liu, L. Tong, T. Kailath, A least-squares approach to blind channel identification, IEEE Trans. Signal Process. 43 (12) (December 1995).

[9] H. Luo, Y. Li, The application of blind channel identification techniques to prestack seismic deconvolution, Proc. IEEE 86 (10) (October 1998) 2082–2089.

[10] K.F. Kaaresen, T. Taxt, Multichannel blind deconvolution of seismic signals, Geophysics 63 (November–December 1998) 2093–2107.

[11] B.D. Jeffs, Sparse inverse solution methods for signal and image processing applications, in: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, May 1998, pp. 1885–1888.

[12] L.P. Meilhac, E. Moulines, K.A. Meraim, P. Chevalier, Blind identification of multipath channels: a parametric subspace approach, IEEE Trans. Signal Process. 49 (7) (July 2001) 1468–1480.

[13] K.F. Kaaresen, Evaluation and applications of the iterated window maximization method for sparse deconvolution, IEEE Trans. Signal Process. 46 (3) (March 1998) 609–624.

[14] J. Idier, Y. Goussard, Multichannel seismic deconvolution, IEEE Trans. Geosci. Remote Sensing 31 (5) (September 1993) 961–979.

[15] A.A. Amini, T.E. Weymouth, R.C. Jain, Using dynamic programming for solving variational problems in vision, IEEE Trans. Pattern Anal. Mach. Intell. 12 (9) (1990) 855–867.

[16] M. Buckley, J. Yang, Regularized shortest-path extraction, Pattern Recognition Lett. 18 (June 1997) 621–629.

[17] K.F. Kaaresen, Deconvolution of sparse spike trains by iterated window maximization, IEEE Trans. Signal Process. 45 (5) (1997) 1173–1183.