Accelerated plane-wave destruction[a]

[a]Published in Geophysics, 78, no. 1, V1-V9, (2013)

Zhonghuan Chen[1], *Sergey Fomel*[2], *and Wenkai Lu*[1]

ABSTRACT

When plane-wave destruction (PWD) is implemented by implicit finite differences, the local slope is estimated by an iterative algorithm. We propose an analytical estimator of the local slope that is based on convergence analysis of the iterative algorithm. Using the analytical estimator, we design a noniterative method to estimate slopes by a three-point PWD filter. Compared with the iterative estimation, the proposed method needs only one regularization step, which reduces computation time significantly. With directional decoupling of the plane-wave filter, the proposed algorithm is also applicable to 3D slope estimation. We present both synthetic and field experiments to demonstrate that the proposed algorithm can yield a correct estimation result with shorter computational time.

INTRODUCTION

Local slope fields have been widely used in geophysical applications, such as wave-field separation and denoising (Harlan et al., 1984; Fomel et al., 2007), antialiased seismic interpolation (Bardan, 1987), seislet transform (Fomel and Liu, 2010), velocity-independent NMO correction and imaging (Fomel, 2007b; Cooke et al., 2009), predictive painting (Fomel, 2010), seismic attribute analysis (Marfurt et al., 1999), etc.

Several tools exist for local slope estimation: local slant stack (Ottolini, 1983; Harlan et al., 1984), complex trace analysis (Barnes, 1996), multiwindow dip search (Marfurt, 2006), local structure tensor (Fehmers and Höcker, 2003; Hale, 2007), and plane-wave destruction (Claerbout, 1992; Fomel, 2002). Plane-wave destruction (PWD) approximates the local wave-field by a local plane wave, and models it using a linear differential equation (Claerbout, 1992).

When plane-wave destruction is applied on discrete sampled seismic signals, the corresponding differential equation needs to be discretized by finite differences. Claerbout (1992) used explicit finite differences. In this method, plane-wave approximation of the wavefield can be seen as applying a linear finite impulse reponse (FIR) filter to the wavefield. Slope estimation is equivalent to estimating a parameter of the FIR filter. A least-squares estimator of the local slope can be obtained by minimizing the prediction error of the filter. To improve estimation performance of the explicit finite difference scheme, Schleicher et al. (2009) proposed total least-squares estimation.

The implicit finite difference scheme was applied to the differential equation by Fomel (2002). Using an infinite impulse response (IIR) filter, known as the *Thiran allpass filter* (Thiran, 1971), to approximate the phase-shift operator, the plane-wave destruction equation becomes a nonlinear equation of the slope. An iterative algorithm was designed to estimate the slope. In order to improve stability in the iterative algorithm, a smoothing regularization (Fomel, 2007a) of the increment can be applied at each iteration. Iterations of regularization can be time consuming, however, particularly in the 3D case.

In this paper, we prove the fact that the plane-wave destruction equation is a polynomial equation of an unkown slope. In the case of a three-point approximation of *Thiran's filter*, the convergence results of the iterative algorithm can be analytically analyzed. In this case, we obtain an analytical estimator of the local slope and show that the smoothing regularization can be applied on the final estimator only once. This approach reduces the computational time significantly. We present both 2D and 3D examples, which demonstrate that the proposed algorithm can obtain a slope-estimation result faster than the iterative algorithm, with a similar or even better accuracy.

THEORY

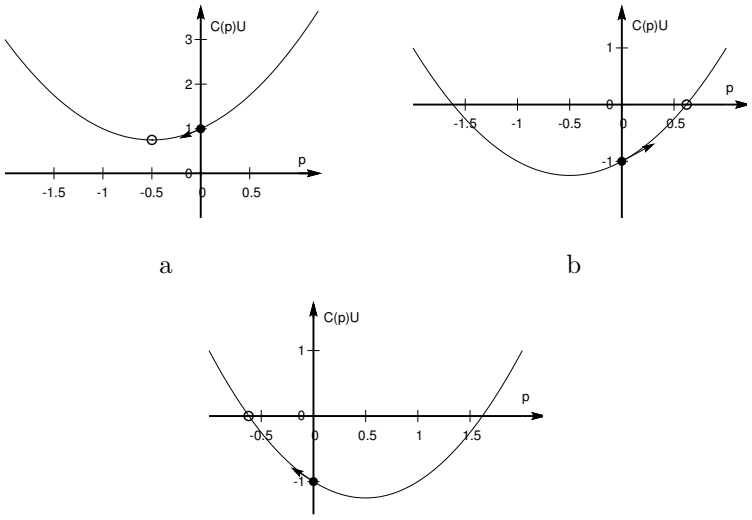


Figure 1: The iterative results of Newton's algorithm. The initial point is $p_0 = 0$ (solid circle), and the convergence result is marked by a blank circle: (a) when $D \leq 0$, (b) when $D > 0$ and $a_1 a_2 > 0$, (c) when $D > 0$ and $a_1 a_2 < 0$.

Review of PWD

The local plane wave can be represented by the following differential equation (Claerbout, 1992):

$$\frac{\partial u}{\partial x} + \sigma \frac{\partial u}{\partial t} = 0 \ , \tag{1}$$

where σ is the local slope in continuous space, with dimension time/length. The wavefields observed at the two positions x_1, x_2 have a time delay which is proportional to their distance, $\sigma|x_1 - x_2|$. In the sampled system with space and time intervals Δx and Δt, we define the discrete space slope in the unit of $\Delta t/\Delta x$, as $p = \sigma \Delta x/\Delta t$. As p is independent of the sampling interval, it can be directly used in irregular dataset (in this case, the unit of the slopes becomes space variant). The time delay between two adjacent positions is then the slope $p\Delta t$:

$$u(x,t) = u(x + \Delta x, t + p\Delta t) \ . \tag{2}$$

With the Z transform applied along both time and space directions, the above equation becomes

$$(1 - Z_x Z_t^p) U(Z_x, Z_t) = 0 \ , \tag{3}$$

where Z_t is the unit time-shift operator, Z_x denotes the unit space-shift operator and $U(Z_x, Z_t)$ is the Z transform of $u(x,t)$. The operator $1 - Z_x Z_t^p$ is the plane-wave destructor. Using *Thiran's fractional delay filter* $H(Z_t) = \dfrac{B(1/Z_t)}{B(Z_t)}$ (Thiran, 1971) to approximate the time-shift operator $Z_t^p = e^{j\omega p}$, where ω is the circular frequency, the plane-wave destructor can be expressed as (Fomel, 2002),

$$C(p) = B(Z_t) - Z_x B(\frac{1}{Z_t}), \tag{4}$$

where

$$B(Z_t) = \sum_{k=-N}^{N} b_k(p) Z_t^{-k}, \tag{5}$$

N is the order of the noncausal temporal filter and $b_k(p)$ are functions of the local slope p.

Equation 4 is a 2D filter. Applying the filter at an arbitrary point in the wavefield, the plane-wave destruction equation 3 becomes a nonlinear equation for the local slope p:

$$C(p, Z_x, Z_t) U(Z_x, Z_t) \approx 0 \ . \tag{6}$$

An iterative method, such as Newton's method, can be applied to find the slope. In practice, wavefields are polluted by noise and the plane wave assumption may not hold true where faults and conflicting boundaries exist. To obtain a stable slope estimation, an additional smoothing regularization process (Fomel, 2007a) is needed at each step. The total computational cost of slope estimation by plane-wave destruction becomes

$O(N_d N_f N_l N_n)$, where N_d is the size of the data, $N_f = 2N + 1$ is the size of the filter, N_l is the number of linear iterations for regularization, and N_n is the number of nonlinear iterations for solving equation 6. Typical values are $N_f = 3, 5$, $N_l =$ 10-50, and $N_n =$ 5-10.

Accelerated PWD

Gauss-Newton's iteration searches the solutions for nonlinear equation 6 as follows: Let p_k be the estimated slope at step k, with estimating error (or destructive error) $e_k = C(p_k)U(Z_x, Z_t)$. In order to find the correct solution p_{k+1} that minimizes e_{k+1}, we need to find the increment Δp_k from the local linearization:

$$e_{k+1} = C(p_k)U(Z_x, Z_t) + C'(p_k)U(Z_x, Z_t)\Delta p_k \approx 0 \ , \tag{7}$$

where $C'(p_k)$ is the derivative of $C(p)$ at p_k with respect to p.

The iterative algorithm stops when a stationary point or a root of $C(p)U$ is reached. They are:

1. Points where $C'(p_k)U = 0$: When p_k satisfies $C'(p)U = 0$, then $e_{k+1} = e_k$, and the Δp_k dependency in equation 7 is removed, stopping further iterations on p_k.
2. Points where $C(p_k)U = 0$ and $C'(p)U \neq 0$: In this case, $\Delta p_k = 0$, thus $p_{k+1} = p_k$, eliminating the need for further improvements on p_k.

The iterative algorithm for equation 6 may converge at different points, depending on the initial point that we chose; $p_0 = 0$ is a common practical choice for the initial solution. In this case, the iterative algorithm may converge to the least absolute root, which denotes the event with smallest dip angle.

In order to analyze the convergence results, the maximally flat fractional delay filter (Thiran, 1971; Zhang, 2009) is designed with polynomial coefficients:

$$b_k(p) = \frac{(2N)!(2N)!}{(4N)!(N+k)!(N-k)!} \prod_{m=0}^{N-1-k} (m - 2N + p) \prod_{m=0}^{N-1+k} (m - 2N - p). \tag{8}$$

Details on how to design the filter can be found in the Appendix.

Since $b_k(p)$ is a polynomial of p, expanding it, we get $b_k(p) = \sum_{i=0}^{2N} c_{ki} p^i$ and

$$B(Z_t, p) = \sum_{k=-N}^{N} \sum_{i=0}^{2N} c_{ki} Z_t^{-k} p^i \ . \tag{9}$$

From equation 8, it is obvious that $b_k(p) = b_{-k}(-p)$, therefore $c_{ki} = (-1)^i c_{-k,i}$ and

$$B(Z_t, p) = B(\frac{1}{Z_t}, -p). \tag{10}$$

Substituting the above two equations, the nonlinear equation 6 becomes a $2N$-th degree polynomial equation for p:

$$\sum_{i=0}^{2N} a_i p^i = 0, \tag{11}$$

and the coefficients of the polynomial plane-wave destruction can be expressed as

$$a_i = [1 - (-1)^i Z_x] \sum_{k=-N}^{N} c_{ki} Z_t^{-k} U, \tag{12}$$

which says that the coefficients of the polynomial PWD can be obtained by applying a 2D filter on the wavefield u. Moreover, the 2D filter can be decoupled into the cascade of two 1D directional filters: the temporal filter $\sum_{k=-N}^{N} c_{ki} Z_t^{-k}$ and the spatial filter $1 - (-1)^i Z_x$.

In the special case of $N = 1$, we get a three-point approximation of $B(Z_t)$. It takes the following form (Fomel, 2002):

$$B(Z_t) = \frac{(1+p)(2+p)}{12} Z_t^{-1} + \frac{(2+p)(2-p)}{6} + \frac{(1-p)(2-p)}{12} Z_t \ . \tag{13}$$

The plane-wave destruction equation 11 is a quadratic equation. The coefficients $a_i (i = 0, 1, 2)$ can be solved for and expressed as

$$a_i = \frac{1}{12}[1 - (-1)^i Z_x] v_i \ , \tag{14}$$

where v_i are outputs of the following three-point temporal filters:

$$v_0 = 2(Z_t^{-1} + 4 + Z_t)U \ , \tag{15}$$
$$v_1 = 3(Z_t - Z_t^{-1})U \ , \tag{16}$$
$$v_2 = (Z_t^{-1} - 2 + Z_t)U \ . \tag{17}$$

In this case, the quadratic plane-wave destruction equation 11 has one stationary point and two roots, which can be analytically expressed as: $\left\{ \frac{-a_1}{2a_2}, \quad \frac{-a_1 \pm \sqrt{D}}{2a_2} \right\}$, where $D = a_1^2 - 4a_0 a_2$.

The plots in Figure 1 show the convergence process of the iterative algorithm when we choose $p_0 = 0$ as the starting value. Geometrically when $D \leq 0$, the iteration converges to the stationary point $\frac{-a_1}{2a_2}$, as shown in Figure 1a. When $D > 0$, it

converges to the least absolute solution of equation 11. Figure 1b and 1c shows the convergence process to the least absolute solution in different cases. We can summarize the convergence result of the iterative algorithm as follows:

$$p = \begin{cases} \dfrac{-a_1}{2a_2} & D \le 0 \\ \dfrac{-2a_0}{a_1 - \sqrt{D}} & D > 0,\ a_1 < 0\ . \\ \dfrac{-2a_0}{a_1 + \sqrt{D}} & D > 0,\ a_1 > 0 \end{cases} \tag{18}$$

As $\dfrac{-2a_0}{a_1 \pm \sqrt{D}} = \dfrac{-a_1 \pm \sqrt{D}}{2a_2}$ in the above equation, we use $\dfrac{-2a_0}{a_1 \pm \sqrt{D}}$ instead of $\dfrac{-a_1 \pm \sqrt{D}}{2a_2}$ to obtain better numerical stability.

When the data is polluted by noise, in order to obtain a robust slope estimation, we can combine the equations in a local window into the following equation set:

$$\mathbf{F}\mathbf{p} \approx \mathbf{g}, \tag{19}$$

where $\mathbf{F}$ is a normalized diagonal matrix and $\mathbf{g}$ is a vector. Their elements are denuminators and numerators of equation 18 respectively. When we are solving the above equation set by least squares, we can use Tikhonov's regularization (Fomel, 2002) or the shaping regularization (Fomel, 2007a, equation 13) to obtain a smooth solution as follows

$$\mathbf{p} = \mathbf{H}[\mathbf{I} + \mathbf{H}^T(\mathbf{F}^T\mathbf{F} - \mathbf{I})\mathbf{H}]^{-1}\mathbf{H}^T\mathbf{F}^T\mathbf{g}\ , \tag{20}$$

where $\mathbf{H}$ is an appropriate smoothing operator. In this case, the regularization runs only once, therefore the computational cost is reduced to $O(N_d N_f N_l)$.

In 3D applications, there are two polynomial PWD equations for inline and crossline slopes separately. Note that, using the decoupling, inline and crossline slope estimations can share the temporal filtering results in equations 15−17. We can obtain the coefficients of the crossline plane-wave destruction equation as

$$a_i = \frac{1}{12}[1 - (-1)^i Z_y]v_i \tag{21}$$

The five-point or longer approximations of $B(Z_t)$ can achieve higher accuracy. Equation 6 in this case becomes a higher-order polynomial equation (see details in the Appendix), which can be solved numerically. However, there are multiple stationary points, and it is difficult to determine the right one analytically. For applications that need five-point or higher accuracy, we suggest obtaining an initial slope estimation by the proposed three-point method and using it to make the iterative algorithm converge faster (to decrease N_n).

EXAMPLES

Synthetic examples

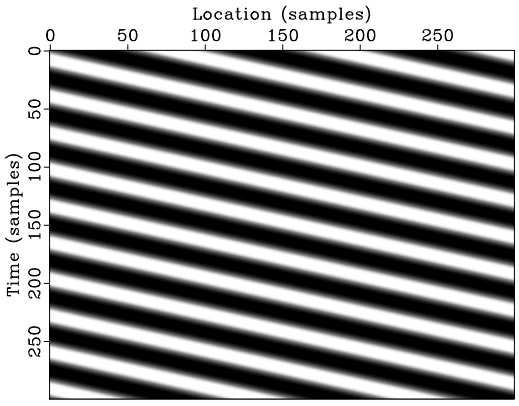


Figure 2: Harmonic waves with constant slope $p_0 = 0.3$.

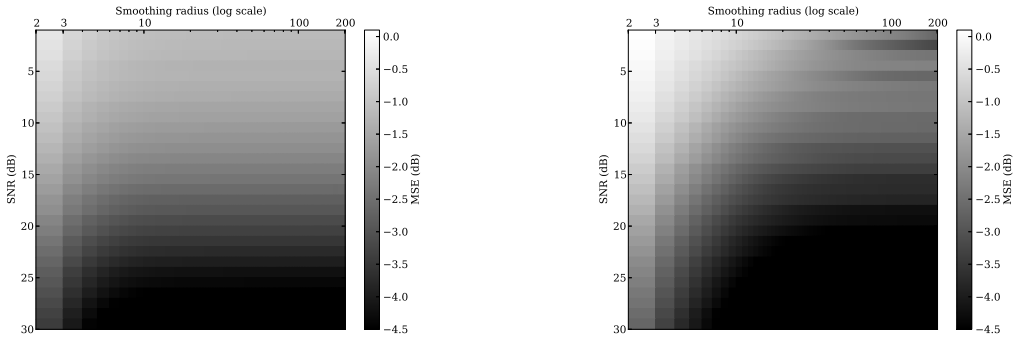
To test the performance of the proposed slope-estimation method, we generated a harmonic wave field with constant slope $p_0 = 0.3$ shown in Figure 2. We added different scales of additive white Gaussian noise (AWGN) to the wave field and estimate the slope by the proposed method. To compare with the iterative algorithm by Fomel (2002), the mean square error (MSE) is used as the criterion:

$$\mathrm{MSE}(p) = \mathrm{E}\{(p - p_0)^2\}. \tag{22}$$

We use five iterations in the iterative method, $N_n = 5$. For constant slope model, using a large smoothing window, the smoothing regularization can converge faster (with less N_l) and we can obtain a better estimation accuracy. For each methods, we try the smoothing windows from 2 to 200 and show the mean square errors in Figure 3. Compared with the iterative method, the proposed method has better accuracy at the left upper (low SNR and small smoothing window) and worse accuracy at right bottom corner (high SNR and large smoothing window).

We show the total runtime of all the noise scale data in Figure 4. For all smoothing windows in the regularization, the proposed method (solid line) only uses about one fifth run time of the three-point ($N = 1$) iterative method (dash line).

A more complicated model from Claerbout (1999) and Fomel (2002) is shown in Figure 5a. It has variable slopes in synclines, anticlines, and faults. The slope estimated by the proposed method is shown in Figure 5b. The estimation takes about 20 ms. The three-point ($N = 1$) iterative method can obtain a similar estimation (shown in Figure 5c) by five iterations, which takes about 130ms. Both methods use a 4-point smoothing window in the regularization, but the proposed method obtains a slightly smoother estimation. In Figure 5d we show the faults detected by the residuals of the proposed plane-wave destruction.



a b

Figure 3: Mean square errors of of the slope estimations by the proposed method (a) and the three-point ($N = 1$) iterative method (b).

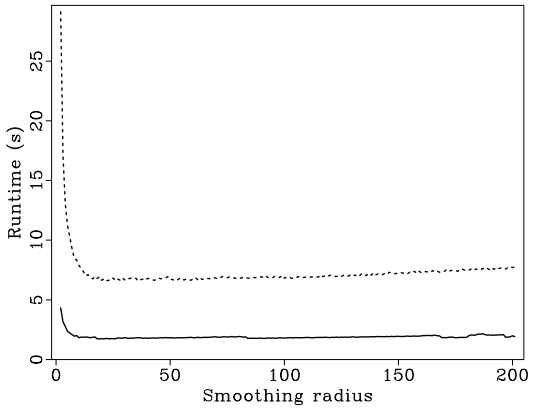


Figure 4: Run time of the proposed method (solid line) and the three-point ($N = 1$) iterative method (dash line).

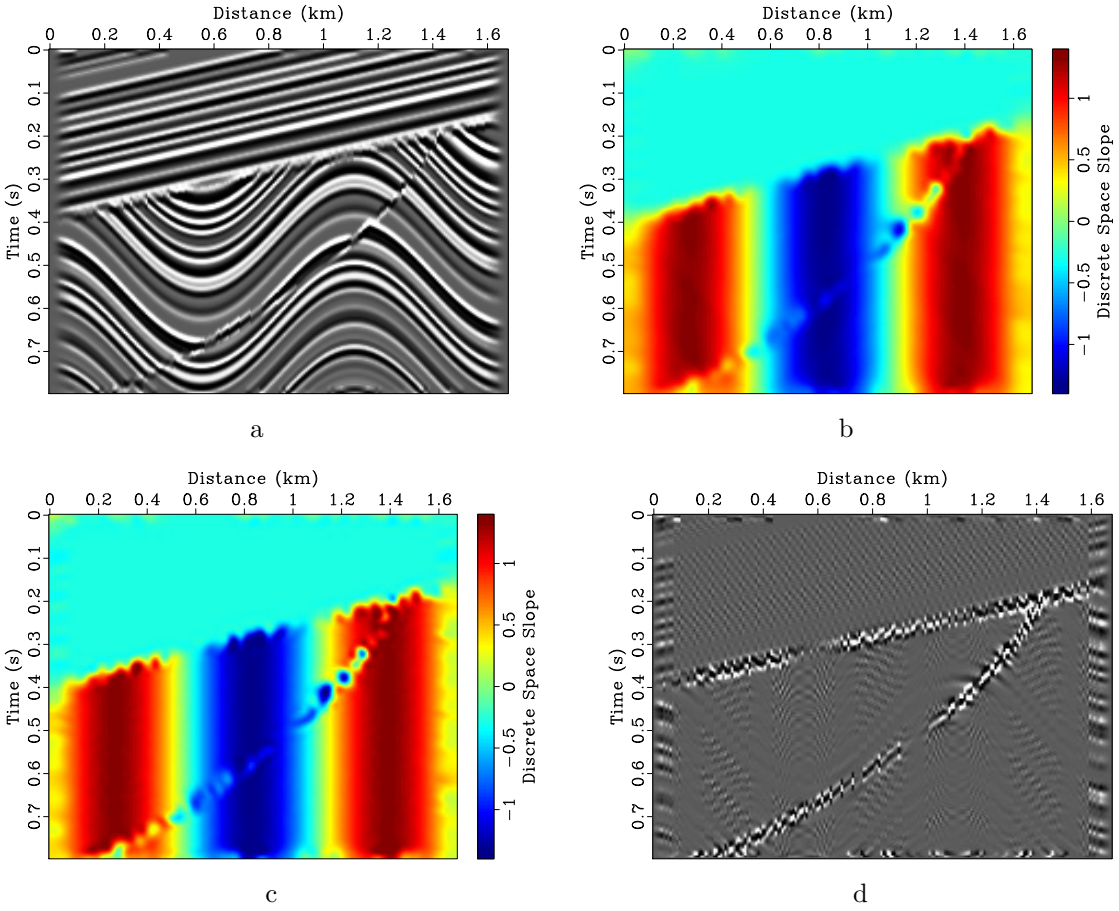


Figure 5: 2D slope estimation example: (a) synthetic data, (b) slope field estimated by the proposed algorithm, (c) slope field estimated by the iterative algorithm after five iterations, (d) faults detection by plane-wave destruction with the estimated slope.

Application

The 2D seislet transform (Fomel and Liu, 2010) uses local slopes to predict and update even and odd traces in the wavelet lifting scheme. The seislet transform itself is fast, but the slope estimation step is comparatively slow. The 3D seislet transform can be constructed in the same way by using 3D slopes and cascading 2D transforms in inline and crossline directions. So that an efficient transform can be built, the proposed accelerated plane-wave destruction is applied to slope estimation.

Figure 6 shows a part of the Teapot Dome image from Wyoming. The 3D slope estimation yields two slope fields along two space directions. Figure 7 shows the local inline and crossline slope fields estimated by the proposed algorithm.

The two slopes are used by the lifting scheme in the seislet transform to obtain the seislet transform coefficients. The coefficients of the 2D seislet transform are concentrated at the planes near the zero inline plane, as shown in Figure 8a, whereas in Figure 8b, the 3D transform coefficients are concentrated in the corner region near the origin.

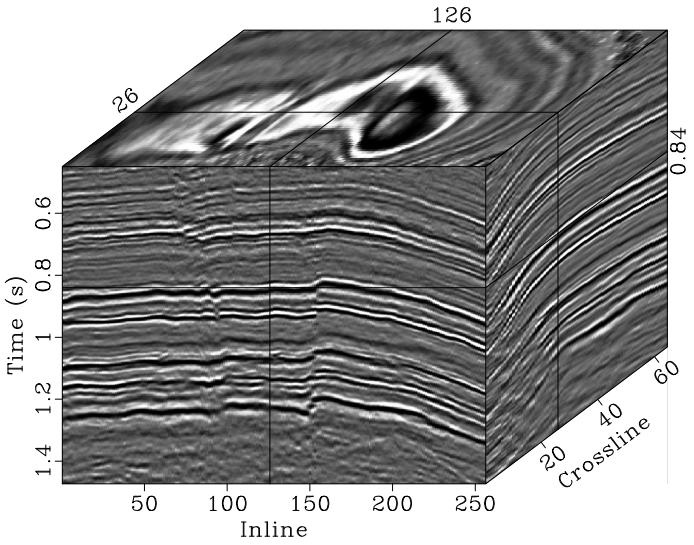


Figure 6: A portion of the 3D data from the Teapot dataset.

To illustrate the compressive performance of the 3D seislet transform, Figure 9 shows the reconstruction results at different percentages of the compression. Most of the data can be reconstructed using only 1% of the seislet coefficients (1:100 compression ratio). In order to compare with the iterative methods quantitatively, we define the following normalized cross-correlation (NCC)

$$NCC(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}^T\mathbf{y}}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2}. \tag{23}$$

In the seislet transform, the more accurate dip we use, the better compression we can obtain. The NCC between the reconstructed data and the original data can be

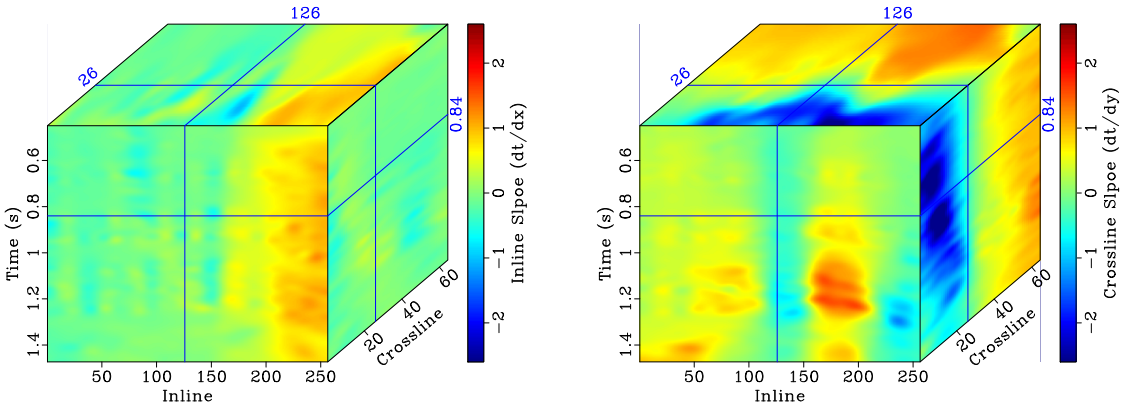


Figure 7: 3D slopes estimated by the proposed algorithm: (a) inline slope, (b) crossline slope.

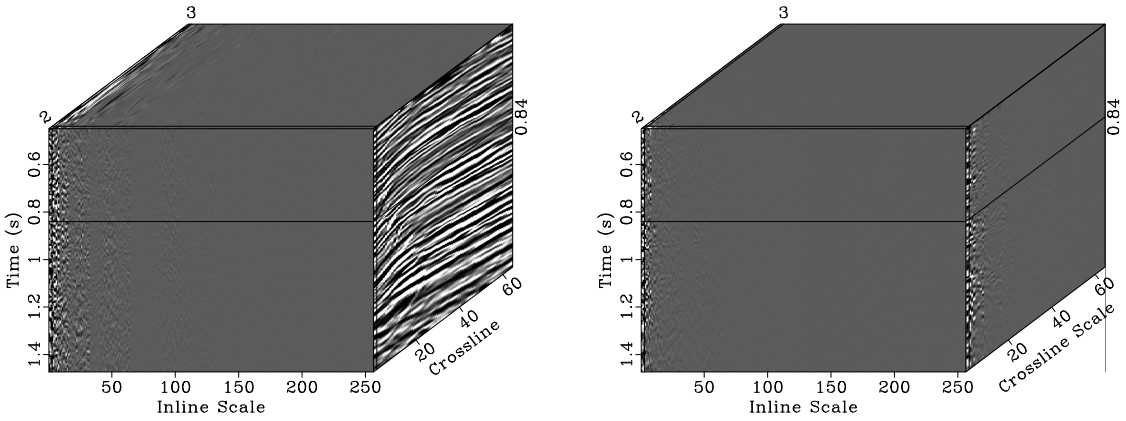


Figure 8: Coefficients of: (a) the 2D seislet transform along inline direction only, (b) the 3D seislet transform.

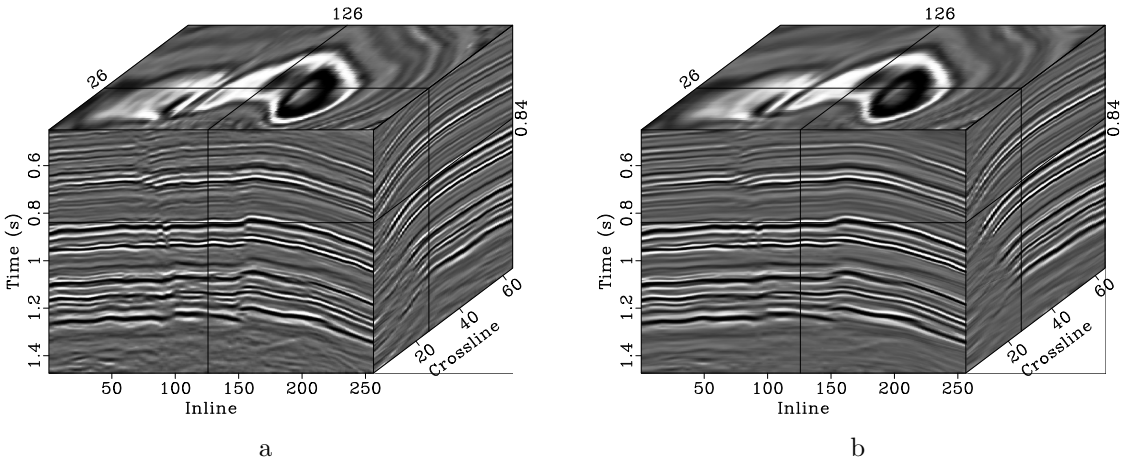


Figure 9: Reconstruction of 3D-seislet-compressed data using: (a) 5% of the seislet coefficients, (b) 1% of the seislet coefficients.

used to quantify the compressive performance. In Table 1, the proposed method is compared with both three-point ($N = 1$) and five-point ($N = 2$) iterative methods.

In all the three methods, we use ten-point smoothing windows in both inline and crossline dimensions, and use five-point smoothing window in time direction. In order to obtain a similar slope estimation as the proposed method, the three-point ($N = 1$) and five-point ($N = 2$) iterative mathods need about six and five iterations respectively. The five-point method can obtain a better compression ratio than the three-point method, because of its better accuracy in slope estimation. However, although the iterative methods have smaller PWD residuals, neighter of them achieves a better NCC than the proposed method. In this case, using the noniterative estimation as the initial in five-point estimation can save at least 250s. That is to say, the computational time cost is reduced by a factor of about 6.

Method	N	Iterations	Runtime (s)	RES-inline	RES-xline	NCC-1	NCC-5
noniterative	1	0	54.63	0.2314	0.2242	0.8858	0.9618
iterative	1	6	334.5	0.2267	0.2172	0.8814	0.961
iterative	2	5	301.6	0.2194	0.2075	0.8838	0.9615

Table 1: Performance of different dip estimation methods in 3D seislet transform: Runtime is the run time of the slope estimation process; RES-inline is the inline residual; RES-xline is the crossline residual; NCC-1 is the normalized cross-correlation between the original data and the reconstruction using one percent of the seislet coefficients; NCC-5 uses five percents of the coefficients. The run time for 2D and 3D seislet transform are about 22.45 s and 45.0 s respectively.

CONCLUSIONS

In this paper, we derived an analytical estimator of the local slope in three-point implicit plane-wave destruction. On the basis of this result, we built an accelerated slope estimation algorithm. Examples show that the proposed method can produce a result that is similar to that of the iterative algorithm, at a reduced computation time.

Two or more conflicting slopes can be estimated simultaneously by the iterative algorithm. In this case, the PWD equation is a multi-variable polynomial, and the convergence analysis becomes complicated. The proposed noniterative method is not yet suitable for multi-slope estimation. However, we believe that it can find many applications in situations where one dominant slope is sufficient.

ACKNOWLEDGMENTS

We thank Alexander Klokov for his helpful technical suggestions. We also thank the reviewers and the editors for their careful review of the paper.

The Teapot Dome data is provided by the Rocky Mountain Oilfield Testing Center, sponsored by the U.S. Department of Energy. Zhonghuan Chen's joint research is sponsored by China Scholarship Council and China State Key Science and Technology Projects (2011ZX05023-005-007).

APPENDIX A: POLYNOMIAL FORM OF PWD

If all the coefficients of $B(Z_t)$ are polynomials of p, equation 4 is also a polynomial of p, and the plane-wave destruction equation becomes in turn a polynomial equation of p. The problem is to design a $2N+1$ points filter $B(Z_t)$ with polynomial coefficients such that the allpass system $H(Z_t) = \frac{B(1/Z_t)}{B(Z_t)}$ can approximate the phase-shift operator $Z_t^p = e^{j\omega p}$. Denoting the phase response of the system as $\theta(\omega)$, that is $H(e^{j\omega}) = e^{j\theta(\omega)}$, the group delay of the system is

$$\tau(\omega) = \frac{\partial \theta(\omega)}{\partial \omega}. \tag{A-1}$$

The maximally flat criteria designs a filter with a smoothest phase response. There are $2N$ unknown coefficients in $H(Z_t)$, so we can add $2N$ flat constraints for the first $2N-$th order deviratives of the phase response. It becomes (Zhang, 2009, equation 7)

$$\begin{cases} \tau(\omega) = p \\ \dfrac{\partial^n \tau(\omega)}{\partial \omega^n} = 0 \qquad n = 1, 2, \ldots, 2N \end{cases}, \tag{A-2}$$

which is equivalent to the following linear maximally flat conditions (Thiran, 1971):

$$\sum_{k=-N}^{N} (d-k)^{2n+1} b_k = 0, \tag{A-3}$$

where $n = 0, 1, \ldots, 2N-1$ and $d = p/2$ is the fractional delay of $B(1/Z_t)$ or $1/B(Z_t)$.

In order to solve b_k from the above equations, Thiran (1971) used an additional condition $b_0 = 1$, which leads $b_k (k \neq 0)$ to be a fractional function of p. Differently from that, we use the following condition,

$$\sum_{k=-N}^{N} b_k = 1, \tag{A-4}$$

where b_k can be proved to be polynomials of p.

Let vector $\mathbf{b} = [b_0, b_N, \ldots, b_1, b_{-1}, \ldots, b_{-N}]^{\mathrm{T}}$. Combining equations A-3 and A-4, we rewrite them into the following matrix form:

$$\left[\begin{array}{c|ccccccc} 1 & 1 & \ldots & 1 & 1 & \ldots & 1 \\ \hline d & d-N & \ldots & d-1 & d+1 & \ldots & d+N \\ d^3 & (d-N)^3 & \ldots & (d-1)^3 & (d+1)^3 & \ldots & (d+N)^3 \\ \vdots & \vdots & \ldots & \ldots & \ldots & \ldots & \vdots \\ d^{4N-1} & (d-N)^{4N-1} & \ldots & (d-1)^{4N-1} & (d+1)^{4N-1} & \ldots & (d+N)^{4N-1} \end{array}\right] \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

The matrix on the left side, denoted as $\mathbf{V}$, can be split into four blocks $\left[\begin{array}{c|c}\mathbf{A}&\mathbf{B}\\\hline\mathbf{C}&\mathbf{D}\end{array}\right]$ as shown above. Following the *lemma of matrix inversion*,

$$\mathbf{V}^{-1}=\left[\begin{array}{cc}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}&-(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\mathbf{B}\mathbf{D}^{-1}\\-\mathbf{D}^{-1}\mathbf{C}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}&\mathbf{D}^{-1}+\mathbf{D}^{-1}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\mathbf{B}\mathbf{D}^{-1}\end{array}\right], \quad \text{(A-5)}$$

therefore the coefficients

$$\mathbf{b}=\mathbf{V}^{-1}[1,0,\ldots,0]^{\mathrm{T}}=\left[\begin{array}{c}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\\-\mathbf{D}^{-1}\mathbf{C}(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\end{array}\right]. \quad \text{(A-6)}$$

Let subindex $i=-N,-N+1,\ldots,-1,1,2,\ldots,N$ and $x_i=d+i$. Submatrix $\mathbf{D}$ can be expressed as

$$\begin{aligned}\mathbf{D}&=\mathbf{E}\mathbf{X}\\&=\left[\begin{array}{cccccc}1&\ldots&1&1&\ldots&1\\(d-N)^2&\ldots&(d-1)^2&(d+1)^2&\ldots&(d+N)^2\\(d-N)^4&\ldots&(d-1)^4&(d+1)^4&\ldots&(d+N)^4\\\vdots&\ldots&\vdots&\vdots&\ldots&\vdots\\(d-N)^{4N-2}&\ldots&(d-1)^{4N-2}&(d+1)^{4N-2}&\ldots&(d+N)^{4N-2}\end{array}\right]\mathrm{diag}\left[\begin{array}{c}x_{-N}\\\vdots\\x_{-1}\\x_1\\\vdots\\x_N\end{array}\right],\end{aligned}$$

so $\mathbf{D}^{-1}=\mathbf{X}^{-1}\mathbf{E}^{-1}$. Denoting $\mathbf{U}=\mathbf{E}^{-1}$ with elements $u_{ij},\ j=1,2,\ldots,2N$, as $\mathbf{E}$ is a Vandermonde matrix, u_{ij} and Lagrange intepolating polynomials have the following relationship:

$$\sum_{j=1}^{2N}u_{ij}x^{2j-2}=\ell_i(x), \quad \text{(A-7)}$$

where $i=-N,\ldots,-1,1,\ldots,N$, and $\ell_i(x)$ is the Lagrange polynomial related to the basis $d+i$,

$$\ell_i(x)=\prod_{-N\le m\le N}^{m\neq i,m\neq 0}\frac{x^2-(d+m)^2}{(d+i)^2-(d+m)^2}. \quad \text{(A-8)}$$

Substituting the above equation, u_{ij} and x into equation A-7, we can prove equation A-7. It follows that

$$[\mathbf{E}^{-1}\mathbf{C}]_i=d\ell_i(d), \quad \text{(A-9)}$$

$$[\mathbf{D}^{-1}\mathbf{C}]_i=[\mathbf{X}^{-1}\mathbf{E}^{-1}\mathbf{C}]_i=\frac{d}{d+i}\ell_i(d), \quad \text{(A-10)}$$

with

$$\ell_i(d)=\frac{(-1)^{i+1}N!N!}{(N+i)!(N-i)!}\frac{d+i}{d}\prod_{m=-N}^{N}\frac{2d+m}{2d+m+i}. \quad \text{(A-11)}$$

Thus hence

$$\begin{aligned}\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C} &= \sum_{i=-N}^{N}(-1)^i\frac{N!N!}{(N+i)!(N-i)!}\prod_{m=-N}^{N}\frac{p+m}{p+m+i} \\ &= \frac{(4N)!N!N!}{(2N)!(2N)!}\frac{1}{\prod_{m=N+1}^{2N}(m^2-p^2)} \end{aligned} \tag{A-12}$$

and

$$[\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C}]^{-1} = \frac{(2N)!(2N)!}{(4N)!N!N!}\prod_{m=N+1}^{2N}(m^2-p^2). \tag{A-13}$$

It is the coefficient b_0, a $2N$-th degree polynomial of p. Substituting it into equation A-6, the coefficients at $k=\pm1,\pm2,\dots\pm N$ are expressed as

$$\begin{aligned} b_k &= -[\mathbf{D}^{-1}\mathbf{C}]_k[\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C}]^{-1} \\ &= \frac{(2N)!(2N)!}{(4N)!(N+k)!(N-k)!}\prod_{m=0}^{N-1-k}(m-2N+p)\prod_{m=0}^{N-1+k}(m-2N-p). \end{aligned} \tag{A-14}$$

With the additional condition A-4 in $2N+1$ points approximation, all the coefficients are polynomials of p of $2N$-th degree. Thus the plane-wave destruction equation 6 therefore is proved to be a polynomial equation of $2N$-th degree.

REFERENCES

Bardan, V., 1987, Trace interpolation in seismic data processing: Geophysical Prospecting, **35**, 343–358.

Barnes, A. E., 1996, Theory of 2-D complex seismic trace analysis: Geophysics, **61**, 264–272.

Claerbout, J. F., 1992, Earth soundings analysis: Processing versus inversion: Blackwell Scientific Publications.

———, 1999, Geophysical estimation by example: Geophysical soundings image construction: Stanford Exploration Project.

Cooke, D., A. Bóna, and B. Hansen, 2009, Simultaneous time imaging, velocity estimation, and multiple suppression using local event slopes: Geophysics, **74**, WCA65–WCA73.

Fehmers, G. C., and C. F. W. Höcker, 2003, Fast structural interpretation with structure-oriented filtering: Geophysics, **68**, 1286–1293.

Fomel, S., 2002, Applications of plane-wave destruction filters: Geophysics, **67**, 1946–1960.

———, 2007a, Shaping regularization in geophysical-estimation problems: Geophysics, **72**, R29–R36.

———, 2007b, Velocity-independent time-domain seismic imaging using local event slopes: Geophysics, **72**, S139–S147.

———, 2010, Predictive painting of 3D seismic volumes: Geophysics, **75**, A25–A30.

Fomel, S., E. Landa, and M. T. Taner, 2007, Poststack velocity analysis by separation and imaging of seismic diffractions: Geophysics, **72**, U89–U94.

Fomel, S., and Y. Liu, 2010, Seislet transform and seislet frame: Geophysics, **75**, V25–V38.

Hale, D., 2007, Local dip filtering with directional laplacians: CWP Report, **567**.

Harlan, W. S., J. F. Claerbout, and F. Rocca, 1984, Signal/noise separation and velocity estimation: Geophysics, **49**, 1869–1880.

Marfurt, K. J., 2006, Robust estimates of 3d reflector dip and azimuth: Geophysics, **71**, P29–P40.

Marfurt, K. J., V. Sudhaker, A. Gersztenkorn, K. D. Crawford, and S. E. Nissen, 1999, Coherency calculations in the presence of structural dip: Geophysics, **64**, 104–111.

Ottolini, R., 1983, Signal/noise separation in dip space: SEP Report, **37**.

Schleicher, J., J. C. Costa, L. T. Santos, A. Novais, and M. Tygel, 2009, On the estimation of local slopes: Geophysics, **74**, P25–P33.

Thiran, J., 1971, Recursive digital filters with maximally flat group delay: IEEE Transactions on Circuit Theory, **18**, 659–664.

Zhang, X., 2009, Maxflat fractional delay iir filter design: IEEE Transactions on Signal Processing, **57**, 2950 –2956.