A new sparse representation of seismic data using adaptive easy-path wavelet transform

Jianwei Ma [1,3], Gerlind Plonka [2], Hervé Chauris [3]

[1] Institute of Seismic Exploration, School of Aerospace, Tsinghua University, Beijing 100084, China

[2] Department of Mathematics, University of Duisburg-Essen, Campus Duisburg, 47048 Duisburg, Germany

[3] Centre of Géoscience, Mines ParisTech, 35 rue Saint-Honoré, 77305 Fontainebleau, France

Abstract

Sparse representation of seismic data is a crucial step for seismic forward modeling and seismic processing such as multiple separation, migration, imaging, and sparsity-promoting data recovery. In this paper, a new locally adaptive wavelet transform, called easy-path wavelet transform (EPWT), is applied for the sparse representation of seismic data. The EPWT is an adaptive geometric wavelet transform that works along a series of special pathways through the input data and exploits the local correlations of the data. The transform consists of two steps: reorganizing the data following the pathways according to the data values, and then applying a one-dimensional wavelet transform along the pathways. This leads to a very sparse wavelet representation. In comparison to conventional wavelets, the EPWT concentrates most energy of signals at smooth scales and needs less significant wavelet coefficients to represent signals. Numerical experiments show that the new method is really superior over the conventional wavelets and curvelets in terms of sparse representation and compression of seismic data.

Keywords: Easy-path wavelet transform, seismic processing, adaptive wavelets, sparse representation, curvelets

1 Introduction

Current seismic acquisitions deploy a few ten thousands of shot gathers [25]. Each shot point may also contain more than 10^4 receivers for land seismic data. For a single source-receiver pair, the signal is typically recorded during a few seconds at a rate of a few ms, leading to a few thousands samples per trace. In total, three-dimensional acquisitions easily contain data volumes of a few terabytes. In that context, an efficient transform for compressing data and later processing the data is a key element.

The most striking feature in seismic data is the presence of texture-like wave fronts. In the last two decades, wavelets have been used as one popular tool for seismic data

processing. A primary aim of wavelet expansions is to find a sparse representation for seismic data. A signal representation is sparse when it can capture a signal with a small number of significant coefficients or components. Generally, a sparser transform is attractive for signal processing tasks, e.g., data compression and fast forward modeling, taking advantage of the multiscale and local temporal-frequency analysis.

Tensor product two-dimensional wavelets are not optimal for representing geometric structures because their support is not suited to directional geometric properties. Recently, several geometric wavelets including curvelets [1, 2], shearlets [10], contourlets [7] were proposed to represent the directional features. These non-adaptive highly redundant function frames have strong anisotropic directional selectivity. For a smooth object f with discontinuities along C^2-continuous curves, the best m-term approximation $\tilde{f}_m$ by curvelets/shearlets thresholding obeys $\|f - \tilde{f}_m\|_2^2 \leq Cm^{-2} (\log m)^3$, while for wavelets the decay rate is only m^{-1}. In the last few years, the curvelets have been applied to the fields of seismic exploration, see e.g., seismic denoising [11, 19], data recovery [12], multiple removal [13], migration [3], imaging [8] and forward modeling of wave propagation [24]. A detailed comparison between wavelets, contourlets, curvelets, and their combination for seismic denoising was also examined [23]. More applications of curvelets can be found in recent review papers by two of the authors [16, 17].

However, curvelets are quite redundant (about $3 - 7$ times more than for wavelets in two-dimensional problems), i.e., the number of coefficients in curvelet domain is larger than the number of pixels in the original data. This redundancy factor is useful for denoising, but it prevents the use of curvelets in other fields such as data compression and forward modeling, where a sparse representation is very important. Also, curvelets or shearlets are not adaptive with respect to the regularity of images. They lose their near optimal approximation properties when the data is composed of features having singularities along curves being not exactly C^2-smooth. Some adaptive wavelet transforms have been presented in the image processing community. For instance, Le Pennec and Mallat [15] proposed bandlet orthogonal bases and frames that adapt the geometric regularity of data. The main idea of bandlets is that they warp wavelets along a geometric flow and generate bandlet orthogonal bases in different bands. Dekel and Leviatan [6] introduced a geometric wavelet transform, based on an adaptive binary partition of image domain to match the geometric features of the images. Krommweh [14] proposed an adaptive Haar wavelet transform, named tetrolet transform, which allows so-called tetromino partitions such that the local image geometry is taken into account. The tetrominoes are borrowed from a famous computer game classic 'Tetris' where shapes are built by connecting four equal-sized squares, each joined together with at least one other square along an edge. Some other approaches for locally adaptive image representations also exist, based on the lifting scheme, see e.g. [4, 9]. Using the seislet transform, one needs a priori information such as the local dip along which the signal is smooth. But then the stability of the wavelet transform strongly suffers, and the obtained results only marginally outperform the usual

fast wavelet transform.

Recently, a nonlinear locally adaptive easy-path wavelet transform was proposed by Plonka [20] for a sparse representation of two-dimensional images. The EPWT is related with the idea of grouplets [18], where one applies a weighted Haar wavelet transform to points that are grouped by a so-called association field. The concept of EPWT is very effective and simple to understand. Starting with some suitable point of a given data set, one seeks a path through all data points, such that there is a strong correlation between neighboring data points in this path. By choosing the "best neighbor" strategy using a given optimality rule, all data points are used in the pathways only once. Then one can apply a suitable one-dimensional (1D) discrete wavelet transform to the function along the path. The choice of the path vector ensures that most wavelet coefficients remain small. The same idea is repeated to the wavelet low-pass part. The EPWT has a generalized multiresolution structure and adaptive scaling and wavelet functions depend on the considered images. It also works well for other data structures as e.g. for data on the sphere by suitable extension [21]. It has been proved that the EPWT leads, for a suitable choice of the pathways, to optimal N-term approximation for piecewise Hölder continuous functions with singularities along curves [22]. The idea of EPWT is not restricted to the two-dimensional case, it can be also applied to higher-dimensional data.

In this paper, we apply the EPWT, for the first time, to obtain sparse representations of seismic data in the oil and gas exploration. Numerical experiments show that it is promising for potential applications in seismic fields.

2 Easy-path wavelet transform

In this section, we briefly introduce the easy-path wavelet transform [20]. In each decomposition level, this transform includes two basic steps: (1) finding the path vector and then (2) applying a one-dimensional discrete wavelet transform along the path vector.

Let N_1, N_2 be two positive integers with $N_1 N_2 = 2^L s$, with L, $s \in \mathbb{N}$. Let $f = (f(i,j))_{(i,j)\in I}$ be a digital function where $I = \{(i,j) : i \in [0, N_1 - 1], j \in [0, N_2 - 1]\}$ denotes an index set. In the fields of seismic exploration for a 2D (two-dimensional) shot gather, the i denotes an index of time coordinate, and the j denotes an index of spatial receiver coordinate. Let $J = J(I)$ be a 1D index set for I by rearranging the $I(i,j)$ column by column, i.e., $J(i,j) := i + jN_1$. Further, we define a neighborhood of an index $(i,j) \in I$ by

$$N(i,j) = \{(i_1, j_1) \in I \setminus \{(i,j)\} : |i - i_1| \le A_\Omega, |j - j_1| \le A_\Omega\}.$$

In this paper, we consider $A_\Omega = 1$. Hence, an index that does not lie at the boundary has eight neighbors, an index at the boundary has five neighbors, and an index at a corner has only three neighbors. The next point of the pathway is always taken from the direct neighbors, so that the pathway is continued in the two-dimensional graph. If using

$A_\Omega > 1$, one can choose the next point from an extended neighboring area. This "nonlocal strategy" can lead to better performances at the cost of more computations.

Let us consider an L-level decomposition of the EPWT. At the first level, we determine a complete path vector p^L through I resp. $J(I)$ and then apply a 1D periodic wavelet transform to the function values along this path p^L. For simplicity, we use the 1D index set $J(I)$. We start with $p^L(0) := 0$ and search the minimum of absolute differences of the function values corresponding to the neighborhood of the index 0, to determine the second index $p^L(1)$

$$p^L(1) = \arg\min_k \{|f^L(0) - f^L(k)|,\, k \in \{1, N_1, N_1 + 1\}\}.$$

For instance, assuming the second value $p^L(1)$ is equal to $N_1 + 1$, we can find $p^L(2)$ by looking for the minimum of absolute difference of the function values corresponding to the neighborhood of the index $N_1 + 1$ expect the index 0 that has been used already in the path vector p^L. We then have

$$p^L(2) := \arg\min_k \{|f^L(N_1 + 1) - f^L(k)|,\, k \in \{1, 2, N_1, N_1 + 2, 2N_1, 2N_1 + 1, 2N_1 + 2\}\}.$$

Generally, given the index $p^L(l)$, $l \in [0, N_1N_2 - 1]$, we determine the next value $p^L(l + 1)$ by

$$p^L(l + 1) := \arg\min_k \{|f^L(p^L(l)) - f^L(k)|,\, k \in N(p^L(l)),\, k \neq p^L(v), v = 0, ..., l.\}.$$

We proceed in this simple manner to determine a path vector through the index set $J(I)$ that is locally adapted to the function f. This is the so-called easy path. In the path vector, we only reorder the data points according to their values, and put the points with similar values together. These points can be called local neighborhood (if $A_\Omega = 1$) or nonlocal neighborhood (if $A_\Omega > 1$). The aim is to find a path vector such that the absolute differences between neighbored function values along the path remain as small as possible (i.e., make the function smoother), so that the wavelet transform along the path can result in less significant coefficients. In this way we achieve a sparse representation of the data.

If the choice of the neighborhood is not unique, one may use a pre-given rule (e.g., a favorite direction) to determine a unique pathway. If, for a given $p^L(l)$, no neighbor can be chosen (i.e., all indices in the neighborhood have already been used in path p^L), one needs to interrupt the path and starts with a new index $p^L(l + 1)$ from the remaining unused indices in $J(I)$. After finding the complete path vector p^L, we apply a one-level discrete orthogonal or biorthogonal wavelet transform to the vector of function values $(f^L(p^L(l)))$ along the path p^L. We obtain low-pass smooth coefficients and high-pass wavelet coefficients after the one-level decomposition.

Then, we apply the same strategy to the low-pass part and carry out a path vector p^{L-1} for the second-level wavelet decomposition. For this purpose, we relate the low-pass coefficients c_l^{L-1} to two neighboring indices $p^L(2l)$ and $p^L(2l + 1)$, and seek a path

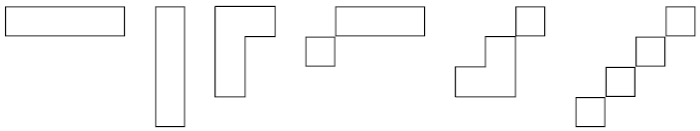


Figure 1: Some shapes of supports of EPWT adaptive basic functions.

vector through the obtained index sets $\{p^L(2l),\, p^L(2l+1)\}$, $l = 0, \ldots, N_1N_2/2 - 1$, where we again exploit the local correlations between neighbored data values. We repeat the same strategy in further levels. The number of decomposition levels depend on the data. Usually, one can apply five to ten levels to obtain satisfying results. The inverse EPWT needs to use the indices of the path vector $p = (p^L, p^{L-1}, \ldots)$. Therefore, the path vector p has to be stored in memory. According to the strategy of this storage, two versions named rigorous EPWT and relaxed EPWT have been constructed by Plonka [20]. The searching and storage of the path vectors cause essential costs in EPWT. In Section 3, we offer a way, how these costs can be considerably reduced for seismic data.

The EPWT can be understood as an adaptive geometric wavelet transform. Some examples of the adaptive supports of the EPWT basis functions are shown in Figure 1, when the 1D Haar wavelet transform is applied in the EPWT algorithm. Here we just outline the forward and inverse EPWT algorithm in Table 1 and 2. For more details on the EPWT, we refer to [20].

Input: Decomposition level L, seismic data f
or its 1D reordered version $c^L := (f(l))_{l=0}^{N_1N_2-1}$.
For $j = L, L-1, \ldots, 1$ **do**
1) Find a suitable path vector p^j through indices (or index sets for $j < L$).
2) Apply the 1D periodic wavelet transform to the vector c^j along the path vector p^j to obtain low-pass coefficients c^{j-1} and high-pass coefficients d^{j-1}.
3) Store the high-pass coefficients in d^{j-1} and the path vector p^j and go to the first step for low-pass components.
end
Output: coefficient vector $g = (d^{L-1}, d^{L-1}, ..., d^0, c^0)$
and the path vector $p = (p^L, p^{L-1}, ..., p^1)$.

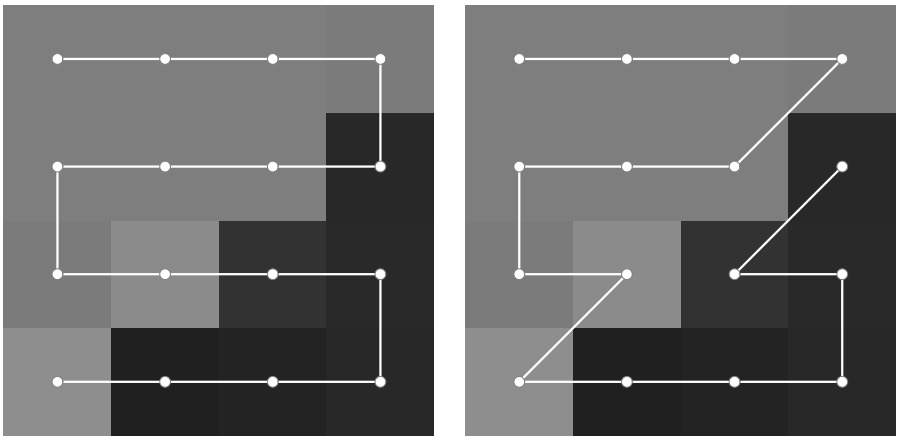
Table 1: Forward EPWT algorithm.

Input: coefficient vectors c^0 and $d^0, \ldots, d^{L-1}$, path vector p, and the decomposition level L.
For $j = 0, ..., L-1$ **do**
1) Apply the inverse 1D periodic wavelet transform to the vector (c^j, d^j) in order to obtain c_p^{j+1}.
2) Apply the permutation to reorder the data $c^{j+1}(p^{j+1}(k)) := c_p^{j+1}(k)$.
end
Output: reconstructed data c^L or f.

Table 2: Inverse EPWT algorithm.

3 Coding strategies for the path vectors of EPWT

The bottleneck of the EPWT is certainly the cost related to store the path vectors p^j. However, especially for seismic data, one can reduce these costs essentially if a large portion of data does not contain essential information. The cost for storage of p^L can even fall away if the path works like for usual wavelet transform along data rows or columns, see Figure 2 (a). However, the EPWT is much more effective than the tensor product wavelet transform according to their ability to exploit the local correlations of the data. The path vector p^L in Fig. 2 (b) can be stored with minimal memory allocation. Compared to Figure 1 (a) with the "usual path", where horizontal directions are preferred, we only need to store the direction from the fourth to the fifth index (being different from the usual way) and from the 9-th to the 10-th index. Then the complete path vector in Fig. 2 (b) is already determined by the specification that each index is contained exactly one times in the path vector.



(a) (b)

Figure 2: A simple example to show the performance of easy path. (a) The path without storage cost. (b) Path that takes the data values into account.

In Figure 3, we show another example (a detail of Figure 4), where the path vector p^L of length 384 is chosen such that the previous direction of the path is preferred as long as the data values of the corresponding pixels do not differ too much. For that purpose, we can fix a certain bound C and say that for given indices $p^L(l-1)$ and $p^L(l)$ the next path component $p^L(l+1)$ is taken in "path direction" if

$$|f(p(l)) - f(p(l+1))| < C.$$

That means, we are not longer seeking for the optimal neighbor but prefer the first neighbor (in the known direction) if the corresponding function value is good enough. Obviously, with a large bound C we end up on a path vector like in Fig. 2 (a) as for the usual wavelet transform along the rows of the data, and without adaptivity costs. The path p^L in Fig. 3 has now (relatively) small entropy, since we need to store only a change of path direction. This can be done with small numbers. Also, we can take into account that each index occurs only one times in the path vector, such that the number of admissible neighbors of an index gets smaller.

For further levels of the EPWT, similar strategies can be developed. For example, our experiments for data in Fig. 4 with gray values between 0 and 255 have shown that p^L can be stored with an entropy of about 0.8 bit per pixel for $C = 20$; 0.4 bpp for $C = 40$ and 0.07 bpp for $C = 100$.

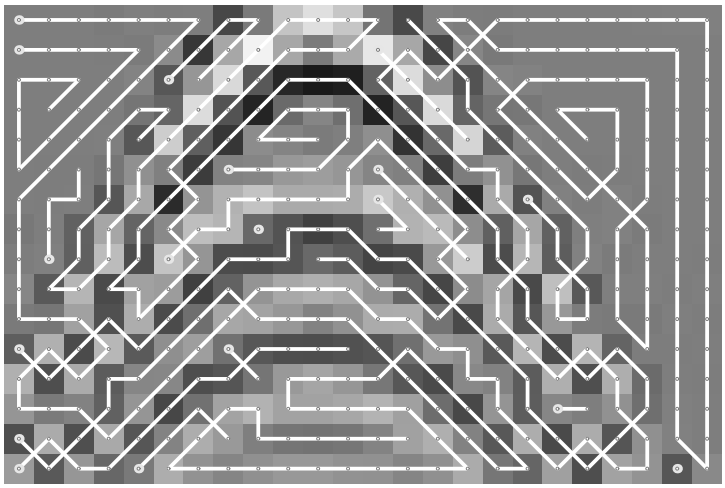


Figure 3: Possible path vector in a detail data of Figure 4.

4 Numerical experiments

In this section, we want to demonstrate the good performance of the EPWT method for seismic data compression, in comparison with wavelets and curvelets. In the experiments, we use the Daubechies DB4 wavelets [5] for the wavelet transform and the second-generation curvelets [2] for the discrete curvelet transform .

Figure 4 (a) is a typical simulated 2D shot gather with 128×128 size. Figure 4 (b), (c), and (d) are the results reconstructed by using the largest 1024 coefficients of wavelets, curvelets, and EPWT. The EPWT provides a much better sparse representation than the wavelets and curvelets, in terms of both SNR (signal-to-noise ratio) values and preservation of wave fronts. Due to the high redundancy of curvelets, the fixed-number reconstruction has a lower SNR value than the wavelet reconstruction. Unfortunately, wavelet reconstructions result in serious oscillating artifacts (mosaic phenomenon) along the edges, due to its poor ability to analyze curve singularities.

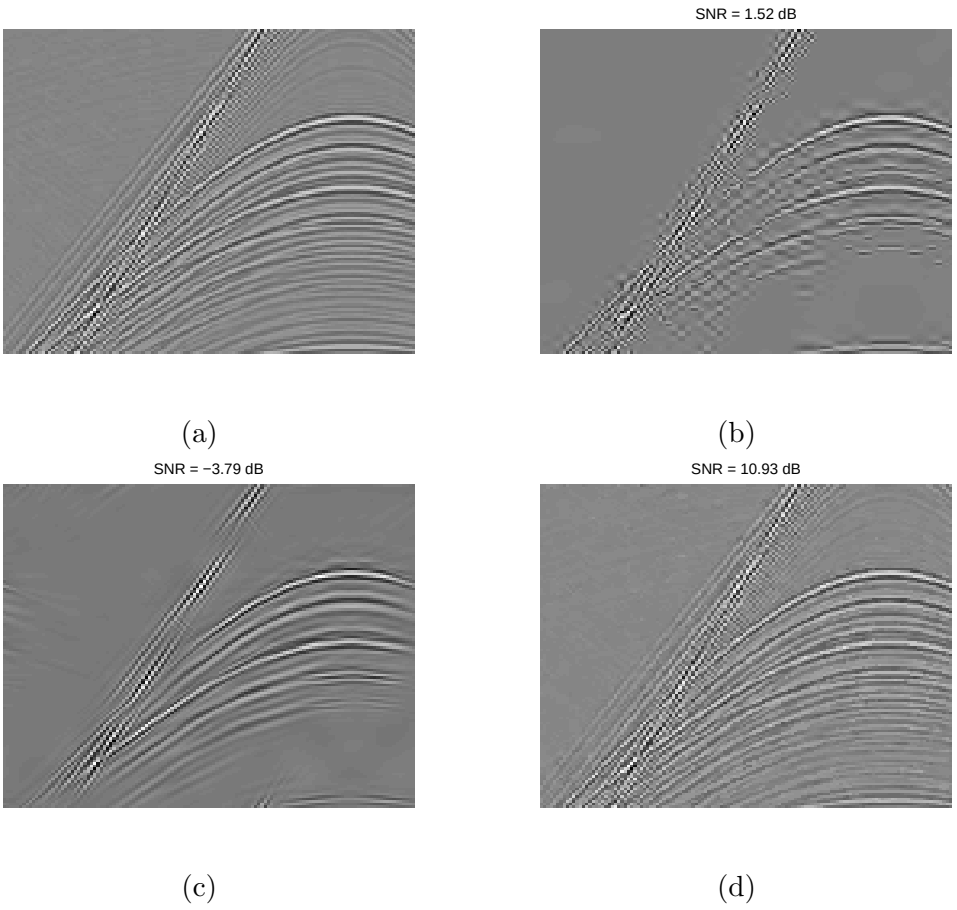
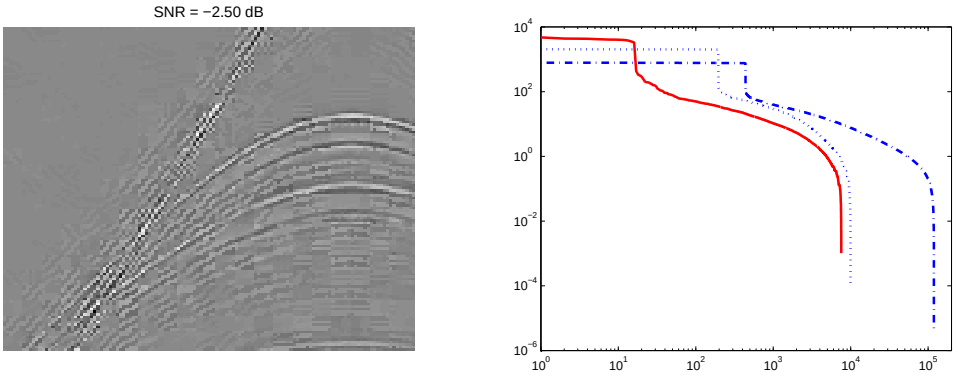


Figure 4: Reconstruction by the largest 1024 coefficients. (a) Original simulated seismic data. (b) Wavelet reconstruction. (c) Curvelet reconstruction. (d) EPWT reconstruction.

Figure 5(a) shows a comparing result by bandlet transform. We select the largest 20000 bandlet coefficients in this case. We do not select 1024 coefficients because the reconstruction of 1024 coefficients almost can not result in visual geometric features. Although the bandlet transform works along geometric edges, it can not capture enough features by these coefficients because of its large redundancy. Another similar transform is grouplet transform that performs a wavelet transform along a geometric path. However, current grouplet version is only for Haar transforms and there are again essential adaptivity costs

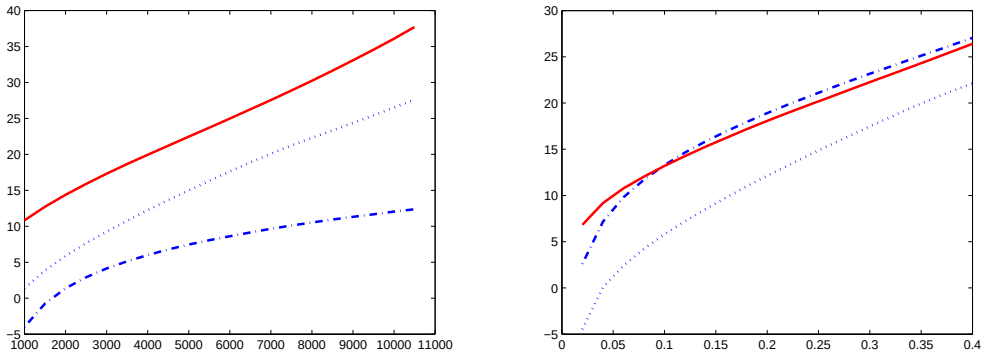
since the association field has to be stored in each level.

Figure 5(b) displays the coefficients after sorting for different transforms, showing that the EPWT coefficients have the fastest decay.



(a) (b)

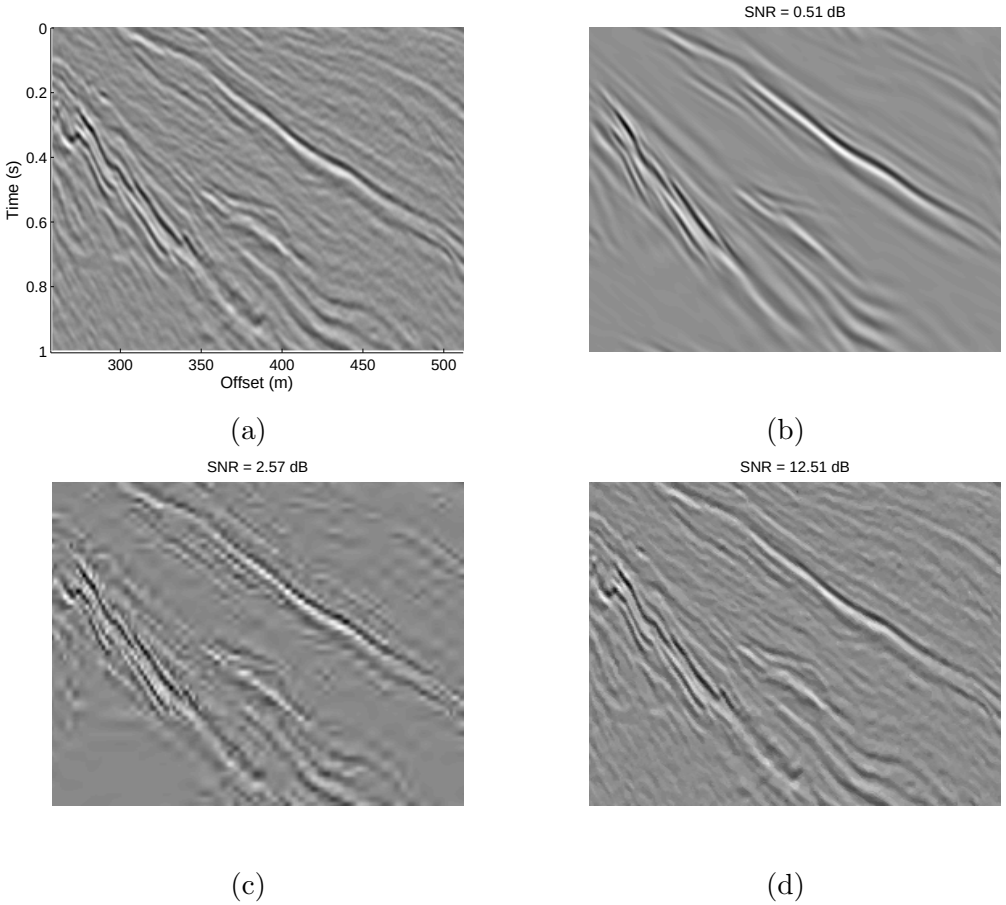
Figure 5: (a) Reconstruction using 20000 coefficients in the bandlet transform. (b) Coefficients after sorting from large to small amplitudes. Solid line, dotted line, and dot-dashed line denote EPWT, wavelets, curvelets, respectively.



(a) (b)

Figure 6: (a) SNR vs number of reconstructed coefficients. (b) SNR vs percentage of reconstructed coefficients to total coefficients. Dotted line, dot-dashed line, and solid line denote wavelets, curvelets, and EPWT, respectively.

Figure 6 shows the change of SNR of reconstructed data as the number of coefficients increases. The EPWT displays the good ability for sparse representation of seismic data, because the EPWT can find the pathways easily along the wave fronts. Even using the same percentage of coefficients (where the curvelet transform uses almost three times as many coefficients) as shown in Figure 6 (b), the EPWT displays similar SNR values as curvelets known to be optimal for sparse representation of data containing smooth discontinuities [1].



(a) (b)

(c) (d)

Figure 7: Reconstruction of real seismic data by using 1024 coefficients. (a) Original data. (b) Curvelet reconstruction. (c) Wavelet reconstruction. (d) EPWT reconstruction.

In Figure 7, we test the methods on a real seismic data. The original size of the data is 1125×1511, i.e., the vertical time coordinate (0 to 4.5 seconds) indicates 1125 samples in each trace, and the horizontal spatial coordinate indicates 1511 traces. The interval between traces is 4 milliseconds. In our test, we take a central part of the data with size 256×256, as shown in Fig. 7 (a). The vertical axis corresponds to time and the horizontal axis corresponds to the position of the receivers. We consider the reconstructions by using the largest 1024 coefficients of curvelets, wavelets and EPWT, respectively. The EPWT displays outstanding superiority for the edge-preserving sparse reconstruction. Figure 8 (a)-(c) show the differences between the input data and reconstructed data by using the curvelets, wavelets and the EPWT, respectively. Fig. 8 (d) shows a comparison of central traces taken from Fig. 7 (a)-(d). The dotted lines display the trace from input data. The solid lines, from left to right, denote curvelet reconstruction, wavelet reconstruction, and EPWT reconstruction. It can be seen that the EPWT method preserves the amplitude of data well, which is important for practical application.

Figure 9 shows the ability of the EPWT method for compressing seismic data with

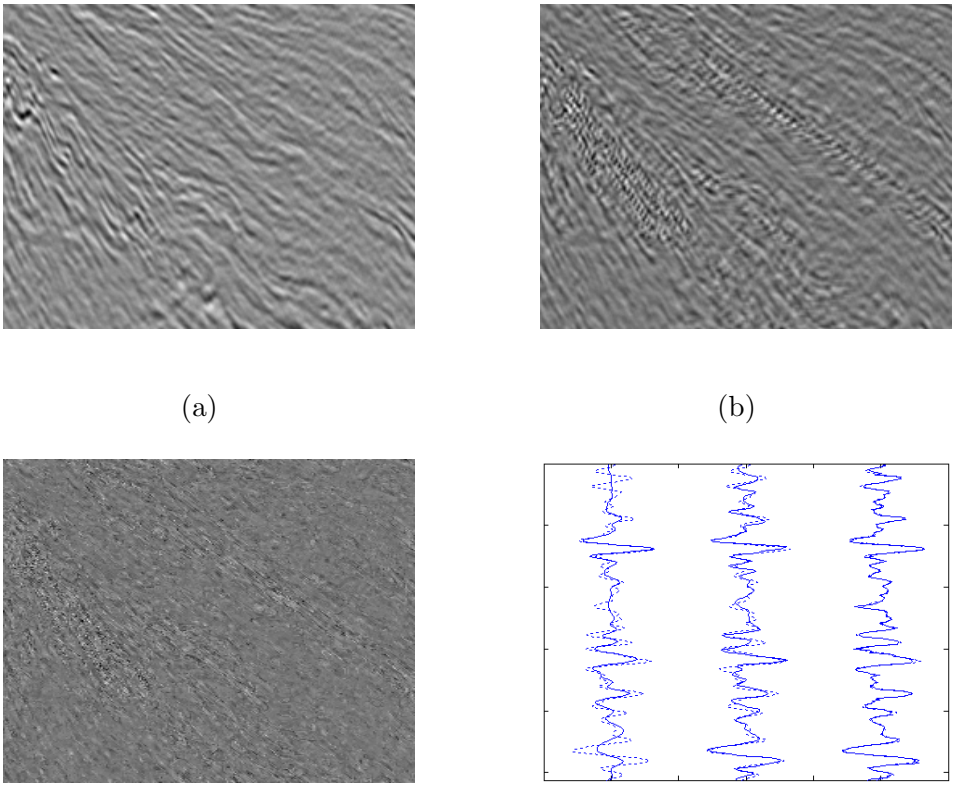


Figure 8: Reconstructed errors by (a) curvelets, (b) wavelets, and (c) EPWT. (d) A comparison of central traces from reconstructed results by different methods, as shown in Fig. 7. The dotted lines denote original data. The solid lines, from left to right, denote curvelet, wavelet, and EPWT reconstruction.

aliased dips. In order to get aliased data with "geometrical aliasing", We for example select one trace over ten from original input data with 1511 traces. The new input data now consists of 152 traces. Fig. 9 (a) shows the data with aliased dips, where the wave fronts are not continue due to the aliased dips. The sampling in the horizontal axis indicates the selected traces over 10, i.e., $1, 11, ..., 1511$. Fig. 9 (b), (c) and (d) show the compressed results by reconstructing the 1024 largest coefficients of curvelets, wavelets, and EPWT, respectively. Once more, the EPWT better works for the compression of data even with aliased dips. The signal is well-preserved in the compressed result.

Seismic tomography consists of retrieving the Earth properties from measurements at the surface, typically travel times picked on shot gathers. The unknown sub-surface parameters are defined in the depth domain. There is thus a need for defining a velocity model for data depth structures from time recorded data. First arrival travel time tomography is a widely used method in that context (e.g. [26]). Only a part of the seis-

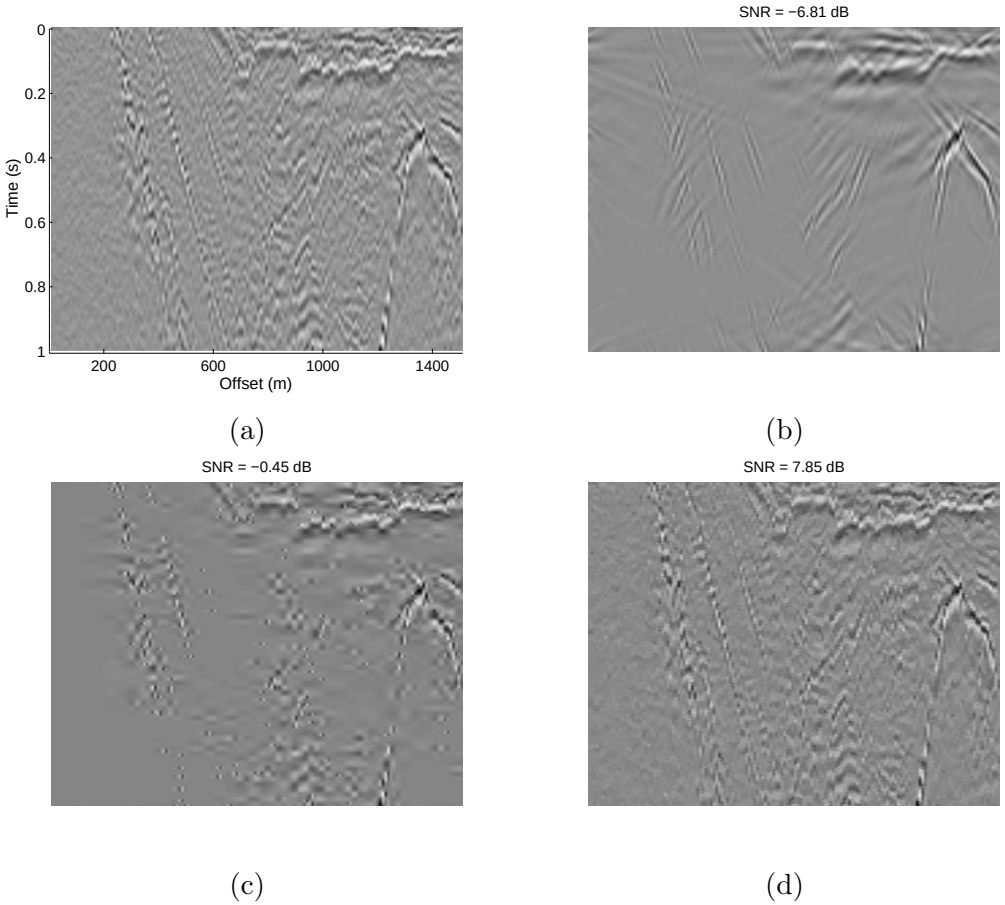
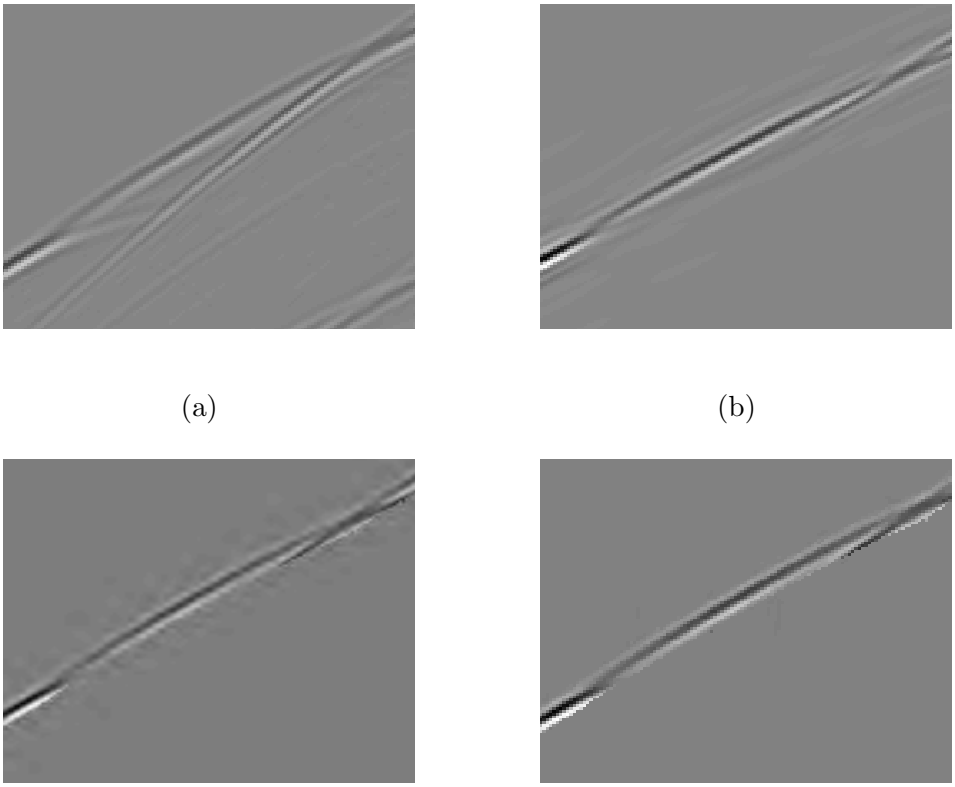


Figure 9: Compression of data with aliased dips. (a) Original data with aliased dips (missed traces). (b) Curvelet compression. (c) Wavelet compression. (d) EPWT compression.

mic information is selected, namely the first arrival waves. The inversion scheme later consists of inverting the picked travel times to estimate the velocity model. The sparse representation of the first arrival waves is useful for the tomography. Figure 10 (a) shows a close-up of a synthetic seismic data. Figure 10 (b)-(d) shows the sparse reconstruction by 500 largest coefficients of the curvelets, wavelets and EPWT, following by a simple amplitude detector (i.e., to detect the first 20 nonzero values in each trace) to extract the first arrival waves. For clarity, we only give the close-up of their reconstructed results. In above all comparisons, we only consider sparse approximation (without coding step) for data compression. The additional coding step can gain the performance of compressive methods naturally.

5 Conclusion

In this paper, we apply a new adaptive easy-path wavelet transform for sparse representation and compression of seismic data. As a first step, numerical results for two-dimensional



(a) (b)

(c) (d)

Figure 10: Reconstruction of first arrive waves by using 500 coefficients. (a) Original data. (b)-(d) Reconstruction by curvelets, wavelets, and EPWT. Only local close-up parts are shown, in order to see the difference more clearly.

(x, t) seismic data show good performances of the EPWT method in this field. The next step would be to develop high-dimensional EPWT algorithm for sparse representation of large volumes of data depending on possibly five dimensions: time and two spatial coordinates for both sources and receivers.

Although the EPWT has surprising performances in terms of sparse representation of seismic data, at the current stage, the main lack of the EPWT is its high computation cost for searching and storing the path p. An alternative strategy to overcome this problem to some extent is to apply the EPWT in wavelet sub-bands. That means, one first applies a two-dimensional discrete wavelet transform to the data, and then applies the EPWT in each wavelet coefficient subband. Due to the wavelet subband is 2^j subsampling, the biggest size of subband (e.g., the finest-scale subband) only has one-quarter size of the original data. From experiments, this strategy based method only needs half computation time (including the pre-computing time of the two-dimensional wavelet transform) for a 256×256 data. Another motivation is that the wavelet coefficients represent edge

information so that the EPWT could follow along the edges better than the previous spatial-domain version. However, the additional pre-computed wavelet transform may cause possible conventional wavelet-like artifacts.

Acknowledgment

The authors would like to thank the Editor and the anonymous reviewers for their constructive comments and fruitful suggestions, which have contributed in improving the quality of this paper. The research in this paper is funded by the National Natural Science Foundation of China under Grand No. 40704019, and the projects PL 170/11-2 and PL 170/13-1 of the Deutsche Forschungsgemeinschaft (DFG). This is gratefully acknowledged.

References

[1] E. Candès, D. Donoho, New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities, *Comm. Pure and Appl. Math.* **56**, 216–266 (2004).

[2] E. Candès, L. Demanet, D. Donoho, L. Ying, Fast discrete curvelet transforms, *Multiscale Model. Simul.* **5** (3), 861–899 (2006).

[3] H. Chauris, T. Nguyen, Seismic demigration/migration in the curvelet domain, *Geophysics* **73** (2), S35–S46 (2008).

[4] R. Claypoole, G. Davis, W. Sweldens, R. Baraniuk, Nonlinear wavelet transforms for image coding via lifting, *IEEE Trans. Image Process.* **12** (2003) 1449–1459.

[5] I. Daubechies, Ten lectures on wavelets, SIAM, Philadelphia, 1992.

[6] S. Dekel, D. Leviatan, Adaptive multivariate approximation using binary space partitions and geometric wavelets, *SIAM J. Numer. Anal.* **43**, 707–732 (2006).

[7] M. Do, M. Vetterli, The contourlet transform: an efficient directional multiresolution image representation, *IEEE Trans. Image Process.* **14**(12), 2091–2106 (2005).

[8] H. Douma, M. de Hoop, Leading-order seismic imaging using curvelets, *Geophysics* **72** (6), S231–S248 (2007).

[9] S. Fomel, Towards the seislet transform, SEG, New Orleans Annual Meeting, p. 2847–2851 (2006).

[10] K. Guo, D. Labate, Optimally sparse multidimensional representation using shearlets, *SIAM J. Math. Anal.* **39**, 298–318 (2007).

[11] G. Hennenfent, F. Herrmann, Seismic denoising with nonuniformly sampled curvelets, *Comput. Sci. Eng.* **8** (3), 16–25 (2006).

[12] F. Herrmann, G. Hennenfent, Non-parametric seismic data recovery with curvelet frames, *Geophysical J. Int.* **173** (1), 233–248 (2008).

[13] F. Herrmann, D. Wang, D. Verschuur, Adaptive curvelet-domain primary-multiple separation, *Geophysics* **73** (3), A17–A21 (2008).

[14] J. Krommweh, Tetrolet transform: A new adaptive Haar wavelet algorithm for sparse image representation, preprint, 2009.

[15] E. Le Pennec, S. Mallat, Sparse geometric image representations with bandelets, *IEEE Trans. Image Process.* **14** (4), 423–438 (2005).

[16] J. Ma, G. Plonka, A review of curvelets and recent applications, *IEEE Signal Processing Magazine*, to appear, 2009.

[17] J. Ma, G. Plonka, Computing with curvelets: from image processing to turbulent flows, *Computing in Science and Engineering* **11** (2), 72–80 (2009).

[18] S. Mallat, Geometrical grouplets, *Appl. Comput. Harmon. Anal.* **26**,161–180 (2009).

[19] R. Neelamani, A. Baumstein, D. Gillard, Cohenrent and random noise attenuation using the curvlet transform, *The Leading Edge* **27** (2), 240–248 (2008).

[20] G. Plonka, The easy path wavelet transform: a new adaptive wavelet transform for sparse representation of two-dimensional data, *SIAM J. Multiscale Modelling Simul.* **7**, 1474–1496 (2009).

[21] G. Plonka, D. Rosca, Easy path wavelet transform on triangulations of the sphere, *Mathematical Geosciences*, to appear.

[22] G. Plonka, S. Tenorth, A. Iske, Optimally sparse image representation by the easy path wavelet transform, preprint, 2009.

[23] H. Shan, J. Ma, H. Yang, Comparisons of wavelets, contourlets, and curvelets in seismic denoising, *Journal of Applied Geophysics* **69**, 103–115 (2009).

[24] B. Sun, J. Ma, H. Chauris, H. Yang, Solving the wave equation in the curvelet domain: a multi-scale and multi-directional approach, *Journal of Seismic Exploration* **18**, 385–399 (2009).

[25] A. Vesnaver, Yardsticks for industrial tomography, *Geophysical Prospecting* **56** (4), 457–465 (2008).

[26] C. Zelt, A. Azaria, A. Levander, 3D seismic refraction traveltime tomography at a groundwater contamination site, *Geophysics* **71** (5), H67–H78 (2006).