

Signal Reconstruction using Sparse Tree Representations

Chinh La and Minh N. Do

Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign

ABSTRACT

Recent studies in linear inverse problems have recognized the sparse representation of unknown signal in a certain basis as an useful and effective prior information to solve those problems. In many multiscale bases (e.g. wavelets), signals of interest (e.g. piecewise-smooth signals) not only have few significant coefficients, but also *those significant coefficients are well-organized in trees*. We propose to exploit the *tree-structured sparse representation* as additional prior information for linear inverse problems with limited numbers of measurements. We present numerical results showing that exploiting the sparse *tree* representations lead to better reconstruction while requiring less time compared to methods that only assume sparse representations.

1. INTRODUCTION

Recently, there have been a series of remarkable work showing that one can effectively exploit the knowledge that the unknown signal has a sparse representation in a certain basis, and use it as a prior information for linear inverse problems¹⁻⁵ (see also⁶ for a related result in sampling). The information that an unknown signal can be well approximated using only K coefficients indicates that perhaps the M signal samples can be reconstructed from only K linear measurements ($K \ll M$). The main difficulty, however, lies in the fact that the K coefficients used in the approximation must be chosen adaptively, i.e., by computing all M coefficients and keeping only the K most significant ones; whereas the measurements in inverse problems are typically collected non-adaptively. Despite this difficulty, it is shown in^{2,3} that under certain conditions, with high probability, the sparse representation can be reconstructed accurately using $O(K \log M)$ non-adaptive linear measurements.

These new results could have great impact in a wide range of inverse problems, from remote sensing to medical imaging, where due to physical constraints only a limited number of measurements can be acquired from the unknown object. Note that these results only rely on the fact that the unknown object has sparse representation or fast decay in a certain transform.

Fig. 1 shows an example of sparse representations where only a few coefficients in the transform domain are significant (i.e., have magnitude above a certain threshold). The power of the sparse representations is demonstrated via accurate reconstruction of the original signal by keeping only K most significant coefficients, where K is much smaller than the number M of samples in the original signals. Furthermore, theoretical studies show that the convergence rates for such nonlinear approximations are optimal for certain classes of signals, such as the class of piecewise smooth signals.⁷

Examining the sparse representations in Fig. 1(c), we notice that not only do they contain few significant coefficients, but also *those significant coefficients are well-organized in trees*. Significant wavelet coefficients of piecewise-smooth signals or images are localized at discontinuity points.⁸ In fact, these embedded tree structures for significant wavelet coefficients have been used successfully in compression,⁹ modeling,^{10,11} and approximation.¹²

We propose to exploit the *tree-structured sparse representation* as additional prior information for linear inverse problems with limited numbers of measurements. Heuristically, a general sparse representation with K coefficients can be specified with $2K$ numbers: K for the values of the significant coefficients and another K for the locations of these coefficients. If the K significant coefficients are known to be organized in trees then the indexing cost is significantly reduced and hence the total description is reduced. Exploiting this embedded tree structure in addition to the sparse representation prior in inverse problems would potentially lead to: (1) better reconstructed signals; (2) reconstruction using fewer measurements; and (3) faster reconstruction algorithms.

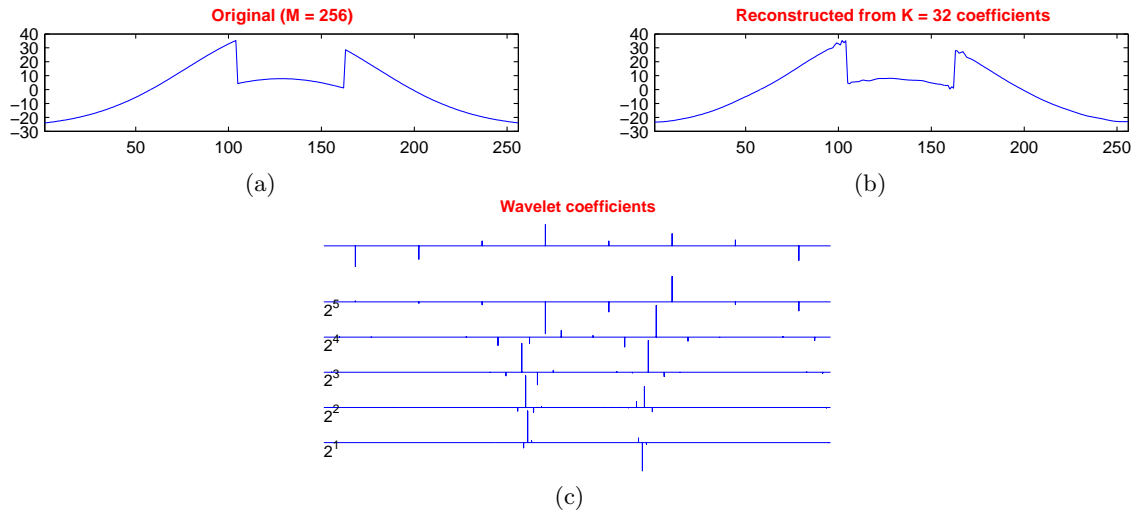


Figure 1. Examples of sparse representations: (a) original signal with $M = 256$ samples; (b) wavelet approximation using $K = 32$ coefficients; (c) wavelet transform.

In this paper, we introduce a new pursuit algorithm which tracks for significant coefficients on trees, and thus is named *Tree-based Orthogonal Matching Pursuit* (TOMP). We also present experimental results showing that TOMP run faster and give better reconstructions compared to existing algorithms such as Basis Pursuit¹³ and Orthogonal Matching Pursuit.¹⁴

The organization of this paper is as follows. Section 2 formulates the inverse problem with sparse tree representation. Section 3 reviews prior methods used for the inverse problem, namely Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP). Our new method, named Tree-based Orthogonal Matching Pursuit (TOMP), is introduced in Section 4. Section 5 investigates numerical results from the experiments employing the TOMP algorithm. We then conclude our paper in Section 6.

2. PROBLEM FORMULATION

Suppose that there is an unknown length- M signal \mathbf{s} that we can only make a limited number of K linear measurements. Normally, K is much smaller than M . Moreover, these measurements are non-adaptive, using K measurement kernels \mathbf{m}_i 's as follows:

$$\langle \mathbf{m}_i, \mathbf{s} \rangle = d_i, \quad i = 1, 2, \dots, K, \quad (1)$$

where d_i 's are the measurement data. The matrix \mathbf{M} consists of all \mathbf{m}_i 's as its rows is called the measurement matrix. Matrix \mathbf{M} has size $K \times M$ and is non-adaptive, independent of the signal \mathbf{s} . Equation (1) can be rewritten as:

$$\mathbf{M}\mathbf{s} = \mathbf{d}. \quad (2)$$

We suppose that \mathbf{s} has a basis expansion as

$$\mathbf{s} = \mathbf{T}\mathbf{c}, \quad (3)$$

so solving \mathbf{c} will give \mathbf{s} . Let $\mathbf{N} = \mathbf{M}\mathbf{T}$, then our measurement equation in (2) becomes

$$\mathbf{N}\mathbf{c} = \mathbf{d}. \quad (4)$$

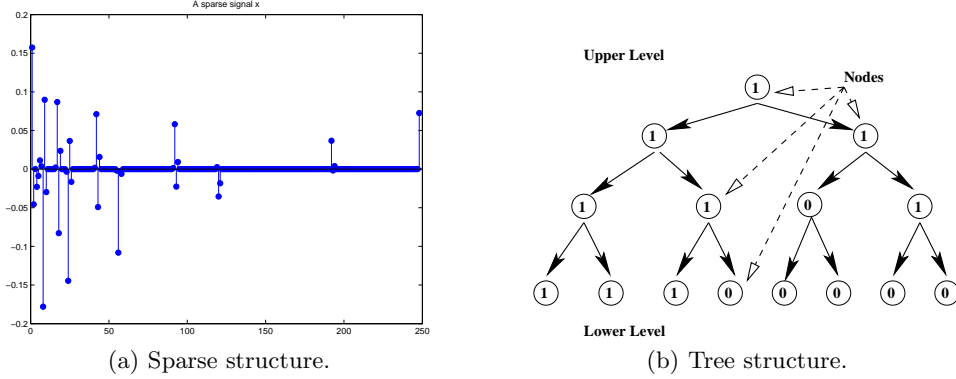


Figure 2. The sparse and tree structures of an example wavelet coefficient set.

For the case of interest where \mathbf{T} is a wavelet transform matrix, \mathbf{c} can be divided into two parts as

$$\mathbf{c} = \begin{pmatrix} \mathbf{l} \\ \mathbf{x} \end{pmatrix}, \quad (5)$$

where \mathbf{l} contains all the scaling coefficients, and \mathbf{x} contains all the wavelet coefficients of \mathbf{s} . We furthermore suppose that \mathbf{N} can be constructed so that \mathbf{l} and \mathbf{x} are measured separately. This scheme is called the *hybrid compress sensing* in.^{3,4} Specifically, we suppose that (4) can be written as

$$\mathbf{N}\mathbf{c} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \end{pmatrix} \begin{pmatrix} \mathbf{l} \\ \mathbf{x} \end{pmatrix} = \begin{pmatrix} \mathbf{l} \\ \mathbf{b} \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{d}. \quad (6)$$

Thus, our inverse problem becomes solving for the wavelet coefficient vector \mathbf{x} from the equation

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \quad (7)$$

We recall that under the limited measurement condition, \mathbf{A} has much less rows than columns; i.e. (7) is a under-determined system. To solve this inverse problem, we notice that \mathbf{x} has the following two special properties.

P1 Vector \mathbf{x} has sparse structure; i.e. only few entries in \mathbf{x} are nonzero or significant.

P2 Those significant entries of \mathbf{x} are well organized in a tree structure (see below).

The tree structure of \mathbf{x} is depicted in Figure 2(b). All wavelet coefficients in \mathbf{x} are positioned in a tree where levels (from top to bottom) are associated with the wavelet scales (from coarse to fine). In this paper, we use the terms ‘‘father’’ and ‘‘child’’ to denote the obvious relationships among the nodes on the tree. In Figure 2(b), nodes of value 1 stand for nonzero coefficients, and nodes of 0 stand for zero coefficients.

As can be seen from Figure 1(c), property **P2** is due to the fact that nonzero coefficients correspond to discontinuity points in the original signal \mathbf{s} . From this, we notice that *if a coefficient is nonzero or significant then its father and all of its ancestors are likely nonzero or significant*. Exploiting this property **P2** is the key point of our proposed algorithm, whereas existing algorithms such as BP and OMP only employ **P1** to solve the inverse problem (7).

3. EXISTING METHODS

3.1. Basis Pursuit (BP)

In the Basis Pursuit algorithm, which has been studied extensively in,^{1-4, 13} the solution of (7) is obtained by solving the following l_1 -norm minimization problem.

$$\hat{\mathbf{x}}_{BP} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad (8)$$

As described in,^{8, 13} problem (8) can be converted into a linear programming problem which can be implemented easily, for example using the function `linprog` in Matlab.

Figure 3(a) and Figure 3(c) show the reconstructed results of piecewise second degree polynomials using the BP algorithm. The original test signal is of length 256 with two discontinuity points, one is in the middle and one is due to the border effect*. In this experiment, the 5-level Daubechies 3 wavelet decomposition is applied. Therefore, in the wavelet domain, \mathbf{l} has length 8, while \mathbf{x} has length 248. The total number of measurements is equal to 8 (number of scaling coefficients) plus the number of rows of \mathbf{A} . As could be seen, BP reconstruction using 8 + 55 measurements gives a noisy reconstruction. However, BP reconstruction using 8 + 80 measurements, about one third of the signal's length, can provide good reconstruction with SNR = 33.99 dB.

An extensive study of BP reconstructions with varying number of measurements is plotted in Figure 3(e). The same wavelet decomposition as the above experiment is used. The reconstructed SNRs are averaged over 10 randomly generated \mathbf{A} 's and different positions of discontinuity points in the unknown signal \mathbf{s} . A key observation in⁴ suggests that acceptable BP reconstruction can be made from the number of measurements which is three times the number of nonzero entries in \mathbf{x} . This matches with the result in Figure 3(e), since the number of nonzeros in \mathbf{x} is observed to vary from 20 to 30.

3.2. Orthogonal Matching Pursuit (OMP)

As described in^{5, 14} OMP is a greedy method to solve equation (7). In this method, the measurement matrix \mathbf{A} is considered as a dictionary with all columns \mathbf{a}_i 's of \mathbf{A} as atoms. In each iteration, OMP selects out from the dictionary one atom that minimizes the difference, also called the residual, between \mathbf{b} and its approximation. Specially, start with $\mathbf{r}_0 = \mathbf{b}$, the OMP algorithm selects the k -th atom as

$$\lambda_k = \arg \max_i |\langle \mathbf{r}_{k-1}, \mathbf{a}_i \rangle|. \quad (9)$$

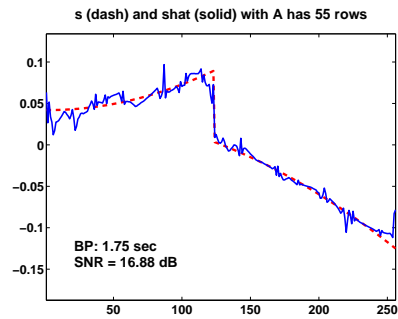
And it updates the residual as

$$\mathbf{r}_k = \mathbf{b} - P_{\text{span}\{\mathbf{a}_{\lambda_1}, \mathbf{a}_{\lambda_2}, \dots, \mathbf{a}_{\lambda_k}\}} \mathbf{b}. \quad (10)$$

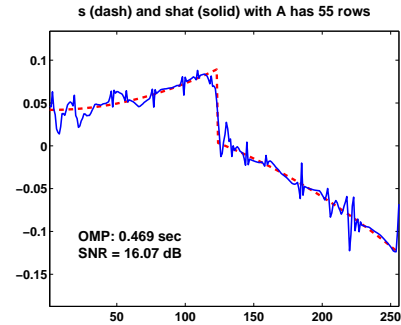
Here, P_V denotes the orthogonal projection onto the subspace V . The OMP algorithm in parallel applies the Gram-Schmidt orthogonalization upon chosen atoms for efficient computation of projections.¹⁴ Ideally, the number of iterations is equal to the number of nonzeros in \mathbf{x} and then the residual becomes zero.

Numerical experiments with OMP reconstructions, using the same setup as the BP method described above, are shown in Figure 3(b), (d), and (e). Through these experiments, we observe that using the same number of measurements OMP can provide reconstructions that are compatible to those by BP. However, OMP requires much less execution time than BP.

*We use the periodic extension for all wavelet transforms in this paper.



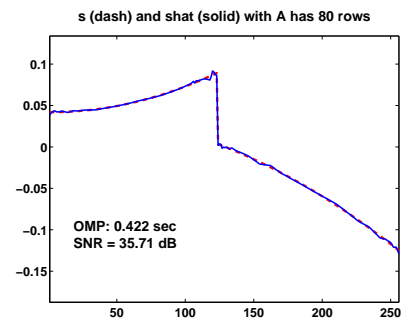
(a) BP reconstruction from 55 measurements.



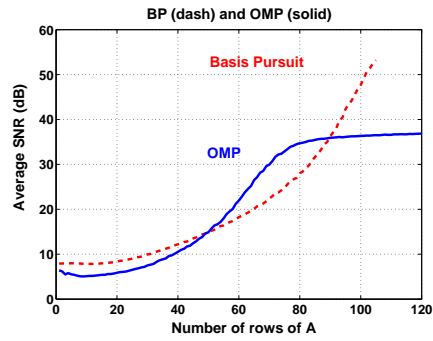
(b) OMP reconstruction from 55 measurements.



(c) BP reconstruction from 80 measurements.



(d) OMP reconstruction from 80 measurements.



(e) SNRs of BP and OMP reconstructions with varying number of measurements.

Figure 3. Reconstructions of piecewise second degree polynomial using BP and OMP using different number of measurements (counting the part for the 248 wavelet coefficients only).

4. TREE-BASED ORTHOGONAL MATCHING PURSUIT

The BP and OMP methods described in Section 3 only exploit the property **P1** of the wavelet representation \mathbf{x} in the inverse problem $\mathbf{A}\mathbf{x} = \mathbf{b}$. We now motivate an improved version of OMP, named *Tree-based Orthogonal Matching Pursuit (TOMP)*, that additionally exploits the property **P2** of \mathbf{x} .

The main idea of TOMP is to *select from \mathbf{A} the columns associated with nodes on the tree* that are corresponding to significant coefficients in \mathbf{x} . Let Λ be the index set of all columns $\{\mathbf{a}_i\}$ of matrix \mathbf{A} , and Λ_k be the index set of all \mathbf{a}_i 's that are selected from the beginning until after step k . At a k -th iteration step, TOMP selects a new set of columns (or atoms) of \mathbf{A} via a two-stage selection process. In the first stage, like OMP, TOMP exams the inner products of the current residual with the remaining atoms.

$$c_i^{(k)} = |\langle \mathbf{r}_{k-1}, \mathbf{a}_i \rangle|, \quad \text{where } i \in \Lambda \setminus \Lambda_{k-1}, \quad (11)$$

$$c_i^{*(k)} = \max\{c_i^{(k)} : i \in \Lambda \setminus \Lambda_{k-1}\}. \quad (12)$$

Based on these inner products, TOMP narrows the search down to a small set of potential candidates:

$$S_k = \{i : c_i^{(k)} \geq \alpha c_i^{*(k)}\}, \quad (13)$$

where α is a relaxation coefficient, $0 < \alpha \leq 1$.

As we mention in Section 2, due to the property **P2**, if \mathbf{a}_i is the next selected atom then all of the ancestors of \mathbf{a}_i are likely correspond to nonzero or significant entries in \mathbf{x} , and thus should also be selected. We denote F_i as the index collection of \mathbf{a}_i and its ancestors.

In the second stage, TOMP selects from the candidate set S_k the atom that together with its ancestors provide the maximal reduction of residual, or

$$i_k = \arg \min_{i \in S_k} \|\mathbf{b} - P_{\text{span}\{\mathbf{a}_l : l \in \Lambda_{k-1} \cup F_i\}} \mathbf{b}\|_2. \quad (14)$$

And then we add the whole family F_{i_k} of the chosen node i_k to our collection Λ_k :

$$\Lambda_k = \Lambda_{k-1} \cup F_{i_k}. \quad (15)$$

The residue is updated by deducting the projection of \mathbf{b} onto the selected family:

$$\mathbf{r}_k = \mathbf{b} - P_{\text{span}\{\mathbf{a}_l : l \in \Lambda_k\}} \mathbf{b} \quad (16)$$

Similar to OMP, in TOMP we apply a Gram-Schmidt orthogonalization step to each newly selected atom to obtain efficient computation for projections.

Equations (14)-(16) mark the major differences between OMP and our proposed TOMP. Intuitively, the OMP algorithm sequentially selects the next "good" atom based only on its alignment (i.e. large inner product) with the current residual. In contrast, the TOMP algorithm sequentially selects the whole next "good" family of atoms. The property **P2** of \mathbf{x} can be interpreted that if an atom is "good" then it should have a "good track record", i.e. all of its ancestors should also be "good". Thus, we can see that by realizing on more information when making decision, TOMP is more robust than OMP in selecting correct atoms. This is especially true when there are only limited number of measurements.

The iteration steps of TOMP are repeated until the number of selected atoms in Λ_k reaches a predefined portion (e.g. half) of the number of rows in \mathbf{A} . This is because we cannot expect to recover a signal with more degrees of freedom than the number of measurements. Another stopping rule in the noisy case is $\|\mathbf{r}_k\|_2^2 \leq \varepsilon$ where ε is an optionally selected limit according to the noise level in measurement data. For example, in the case when \mathbf{b} is affected by an additive white Gaussian noise of variance σ^2 , ε can be chosen as $N\sigma^2$, the noise energy, where N is the length of \mathbf{b} . These choices of the stopping limits regularize the effects of noise upon the reconstructed signal $\hat{\mathbf{s}}$ and will be investigated later in Section 5.

Summary of the TOMP algorithm

INPUT:

- An $N \times M$ measurement matrix \mathbf{A} where $N \ll M$
- A data vector \mathbf{b} of length N

OUTPUT:

- A reconstruction vector $\hat{\mathbf{x}}$ of length M that solves $\mathbf{Ax} = \mathbf{b}$ and has sparse tree structure

PROCEDURE:

1. Initialize the residual $\mathbf{r}_0 = \mathbf{b}$, the index set $\Lambda_0 = \emptyset$, and the counter $k = 1$. Create the index set Λ of all columns in \mathbf{A} .
2. Find columns of \mathbf{A} which have large inner products with the residual:

$$c_i^{(k)} = |\langle \mathbf{r}_{k-1}, \mathbf{a}_i \rangle|, \quad \text{where } i \in \Lambda \setminus \Lambda_{k-1}, \quad (17)$$

$$c_i^{*(k)} = \max\{c_i^{(k)} : i \in \Lambda \setminus \Lambda_{k-1}\}, \quad (18)$$

$$S_k = \{i : c_i^{(k)} \geq \alpha c_i^{*(k)}\}. \quad (19)$$

3. For each i in S_k , form the family set F_i that consists of i and all of its ancestors.
4. Search among the candidate set S_k for the item that together with its family maximize the reduction of residual:

$$i_k = \arg \min_{i \in S_k} \|\mathbf{b} - P_{\text{span}\{\mathbf{a}_l : l \in \Lambda_{k-1} \cup F_i\}} \mathbf{b}\|_2 \quad (20)$$

5. Set $\Lambda_k = \Lambda_{k-1} \cup F_{i_k}$.

6. Update the residual:

$$\mathbf{r}_k = \mathbf{b} - P_{\text{span}\{\mathbf{a}_l : l \in \Lambda_k\}} \mathbf{b} \quad (21)$$

7. Compare $\|\mathbf{r}_k\|_2^2$ with a preselected limit ε , and compare the number of selected items in Λ_k with a preselected limit ρN . If these limits are not reached then increase k by one and return to step 2.
8. Locations of nonzero coefficients of $\hat{\mathbf{x}}$ are listed in Λ_k . The values of those coefficients are in the expansion:

$$P_{\text{span}\{\mathbf{a}_l : l \in \Lambda_k\}} \mathbf{b} = \sum_{\lambda \in \Lambda_k} \hat{x}_\lambda \mathbf{a}_\lambda \quad (22)$$

Finally, we put $\hat{\mathbf{x}}$ and the scaling vector \mathbf{l} together, then apply the inverse wavelet transform to get a reconstruction $\hat{\mathbf{s}}$ of the original signal \mathbf{s} .

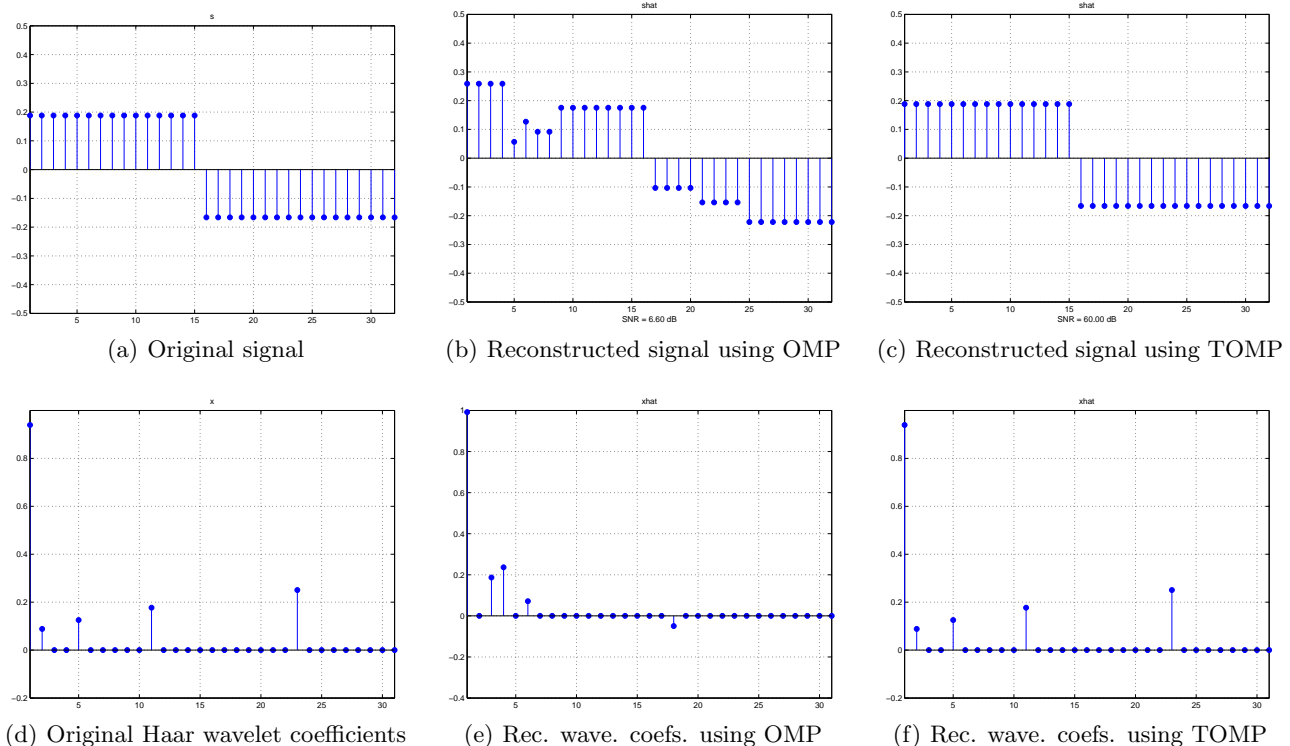


Figure 4. Examples of reconstructions using OMP (orthogonal matching pursuit) and TOMP (tree-based orthogonal matching pursuit). Both methods reconstruct the wavelet coefficients from 8 linear measurements.

5. EXPERIMENTAL RESULTS AND DISCUSSIONS

5.1. Experiments in no-noise cases

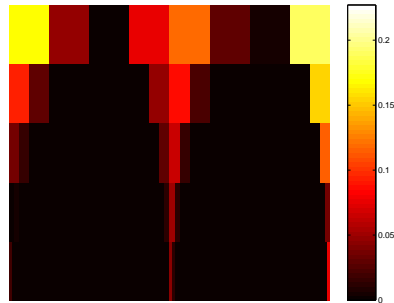
Throughout this section, there is no measurement noise and the stopping limit of OMP and TOMP is selected as $\varepsilon \cong 0$. In the first experiment, we test the OMP and TOMP reconstruction algorithms with piecewise-constant signals and Haar wavelet transform. Figure 4(a) shows an example of a piecewise-constant signal of length $M = 32$ with a discontinuity at sample $k = 15$. Figure 4(d) shows the signal in the wavelet domain with $K = 5$ non-zero coefficients, which can be organized in a tree. Suppose that N randomly chosen linear measurements of the signal are given. Figure 4(b) and (c) shows example reconstructions with $N = 8$ random measurements. In this case, OMP fails to reconstruct the original signal, while TOMP perfectly reconstructs the signal.

In the second experiment, OMP and TOMP are used to reconstruct a piecewise second degree polynomial signal with two discontinuities. The wavelet transform is 5-level with Daubechies 3 wavelets. The original signal of length 256 are reconstructed is reconstructed from a small number of measurements, only 40 for 248 wavelet coefficients. As in Figure 5(b) and Figure 5(c), TOMP captures the essential structure of the wavelet representation, while OMP fails. This leads to a reconstruction SNR gain of about 9 dB by using TOMP over OMP.

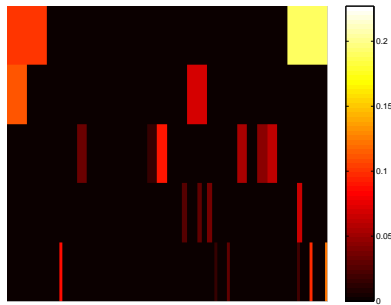
5.2. Experiments in noisy cases

In this section, signals are reconstructed from noisy data using TOMP and OMP algorithms. White Gaussian noise is added to the data \mathbf{b} at SNR = 10 dB. The stopping limit ε is selected in consideration of the noise energy $E = N\sigma^2$, where N is the length of \mathbf{b} and σ^2 is the noise variance. In our experiments, the values used for ε are $0.5 E$, E and $2 E$.

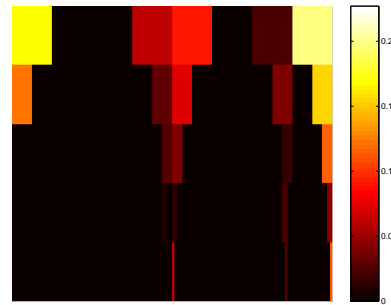
Figure 6 shows reconstructed results of a piecewise 3rd degree polynomial from noisy data using TOMP and OMP. Since the wavelet decomposition is 5-level Daubechies 4, which has higher order than the signal, the



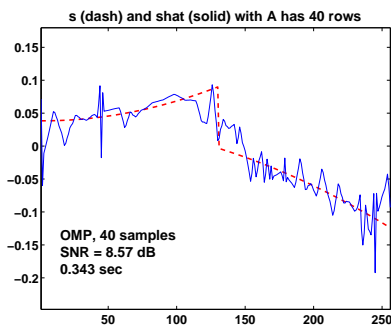
(a) The wavelet tree representation of the original signal



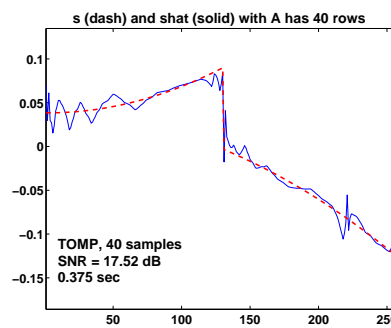
(b) The reconstructed wavelet tree representation using OMP



(c) The reconstructed wavelet tree representation using TOMP



(d) The reconstructed signal using OMP



(e) The reconstructed signal using TOMP

Figure 5. Reconstructions of length-256 piecewise second degree polynomial from 40 measurements using OMP and TOMP.

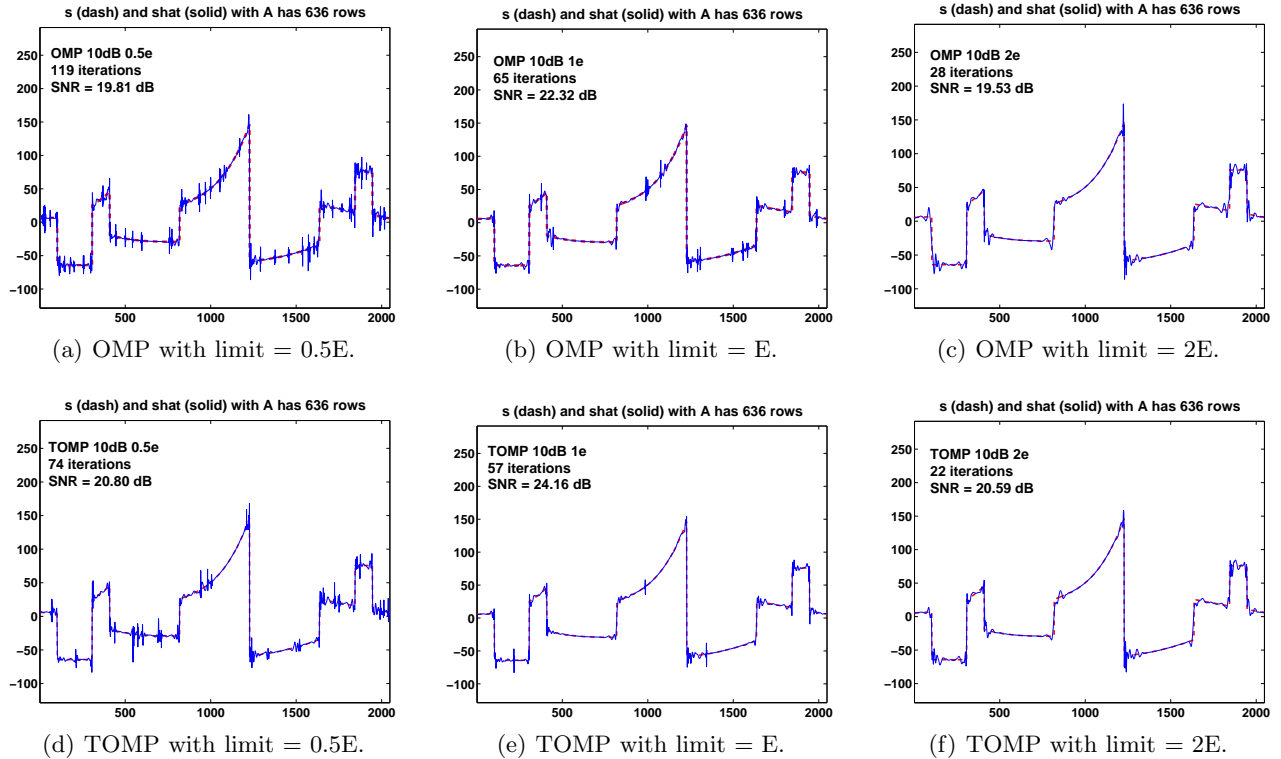


Figure 6. Reconstruction of piecewise 3rd degree polynomial signal from noisy data of SNR = 10 dB using OMP and TOMP with different values of stopping limit.

wavelet coefficients have perfect tree structure. The number of measurements is selected about three times the number of nonzero entries in \mathbf{x} . We observe that TOMP provides higher reconstruction SNR and requires less number of iterations in all cases. Each iteration performs the searching step in (11) once which is the most time consuming stage in the whole algorithm. Therefore, TOMP not only gives higher SNR but also gives faster reconstruction than OMP.

We also notice the effects of choosing the stopping limit ε on the results in Figure 6. When the limit is small, TOMP and OMP copy noise into the reconstructed signals. When the limit is large, our algorithms miss some entries in the wavelet vector \mathbf{x} and hence transition edges in the reconstructed signals are not sharp. We observe that the limit around the noise energy $N\sigma^2$ gives the highest reconstruction SNR. At this limit, TOMP provides a reconstruction SNR of 24.16 dB, which is more than twice the measurement SNR.

6. CONCLUSION

Current algorithms such as BP and OMP employ the sparse structure of unknown signals as prior knowledge to solve the inverse problems. In this paper, we go further by introducing a new algorithm named TOMP (Tree-based Orthogonal Matching Pursuit) which in additionally exploits the tree structure of the signals. Our experiments show that TOMP can be better than BP and OMP in reconstruction quality. The number of iterations in TOMP is less than in OMP since TOMP selects not only one element but the whole tree at a time. When the data is affected by noise, TOMP still provides good reconstruction. This requires a stopping limit around the noise energy as shown in our experiments.

In summary, with the tree structure employed as a prior knowledge, we have achieved better reconstruction in both time and SNR. However, the proposed TOMP algorithm in this paper only presents a first version of signal reconstruction using sparse tree representations. As for future studies, we are investigating other reconstruction

algorithms that can exploit more the tree structures, and studying the performance analysis of these algorithms under sparse tree models.

REFERENCES

1. E. J. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information.” Manuscript, <http://www.acm.caltech.edu/~emmanuel/publications.html>, 2004.
2. E. J. Candès and T. Tao, “Near optimal signal recovery from random projections: Universal encoding strategies?.” Manuscript, <http://www.acm.caltech.edu/~emmanuel/publications.html>, 2004.
3. D. L. Donoho, “Compressed sensing.” Manuscript, <http://www-stat.stanford.edu/~donoho/reports.html>, 2004.
4. Y. Tsaig and D. L. Donoho, “Extensions of compressed sensing.” Manuscript, <http://www-stat.stanford.edu/~donoho/reports.html>, 2004.
5. J. A. Tropp, “Greed is good: Algorithmic results for sparse approximation,” *IEEE Trans. Inform. Th.* **50**, pp. 2231–2242, Oct. 2004.
6. M. Vetterli, P. Marziliano, and T. Blu, “Sampling signals with finite rate of innovation,” *IEEE Trans. Signal Proc.* **50**, pp. 1417–1428, Jun. 2002.
7. D. L. Donoho, M. Vetterli, R. A. DeVore, and I. Daubechies, “Data compression and harmonic analysis,” *IEEE Trans. Inform. Th.* **44**, pp. 2435–2476, October 1998.
8. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 2nd ed., 1999.
9. A. Said and W. A. Pearlman, “A new, fast, and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits and Systems for Video Technology* **6**, pp. 243–250, Jun. 1996.
10. M. Crouse, R. D. Nowak, and R. G. Baraniuk, “Wavelet-based signal processing using hidden Markov models,” *IEEE Trans. Signal Proc. (Special Issue on Wavelets and Filterbanks)* **46**, pp. 886–902, Apr. 1998.
11. M. J. Wainwright, E. P. Simoncelli, and A. S. Willsky, “Random cascades on wavelet trees and their use in modeling and analyzing natural images,” *Journal of Appl. and Comput. Harmonic Analysis* **11**, pp. 89–123, Jul. 2001.
12. A. Cohen, W. Dahmen, I. Daubechies, and R. DeVore, “Tree approximation and optimal encoding,” *Journal of Appl. and Comput. Harmonic Analysis* **11**, pp. 192–226, Dec. 2001.
13. S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Sci. Comput.* **20**(1), pp. 33–61, 1999.
14. G. Davis, S. Mallat, and M. Avellaneda, “Greedy adaptive approximation,” *J. Constr. Approx.* **13**, pp. 57–98, 1997.