Sparse Signal Recovery with Pruning-based Tree search

Jinhong Kim*, and Byonghyo Shim**
* School of Electrical and Computer Engineering , Seoul National University
**School of Electrical and Computer Engineering, Seoul National University

요 약

In this paper, we propose an efficient sparse signal recovery algorithm referred to as the matching pursuit with a tree pruning (TMP). Two key ingredients of TMP are the pre-selection to put a restriction on columns of the sensing matrix to be investigated and the tree pruning to eliminate unpromising paths from the search tree. In our analysis, we show that the sparse signal is accurately reconstructed when the sensing matrix satisfies the restricted isometry property. In our simulations, we confirm that TMP is effective in recovering sparse signals and outperforms conventional sparse recovery algorithms.

1. Introduction

In recent years, compressive sensing (CS) has received much attention as a means to recover sparse signals in underdetermined system [1]-[4]. In CS paradigm, the key finding is that as long as the signal to be recovered is sparse, one can recover the signal with far less number of measurements than traditional approaches.

Well known problem to recover the sparse signal x using the measurement y=Φx is formulated as the l_0-norm minimization problem. Since solving this problem is known to be NP-hard, greedy algorithm has received much attention for cost-effective implementation for sparse signal recovery (e.g., orthogonal matching pursuit, OMP). However, although the greedy approaches are computationally efficient, their performance in general is not satisfactory, especially for large sparsity.

The goal of this paper is to introduce an efficient sparse recovery algorithm based on the tree search with pruning, referred to as the matching pursuit with a tree pruning (TMP). Two key features of TMP are the 'pre-selection' to put a restriction on columns of the sensing matrix to be investigated and the 'pruning-based tree search' to remove the unpromising paths from the search tree. In our analysis, we show that TMP can accurately reconstruct the sparse signal under more relaxed condition than that of existing approaches. From the simulations, we show that our approach significantly reduces the computational burden of exhaustive tree search yet achieves excellent reconstruction performance

2. Matching Pursuit with a Tree Pruning

As mentioned, our proposed algorithm consists of two steps: pre-selection and pruning-based tree search.

The purpose of the pre-selection is to estimate column indices which are highly likely to be the support $\mathbf{T}$ (index set of nonzero entries). Denoting the set of indices as Θ, then the search set is reduced from $\Omega = \{1,2,\ldots,N\}$ to Θ, a subset of Ω. When we perform the tree search, we only use elements of the pre-selected set Θ as a new element in the branches so that we can limit the number of paths in the tree and lessen the search complexity. In the pre-selection, one can basically use any sparse recovery algorithm to build Θ, such as the OMP algorithm running more than $\mathbf{K}$-iterations [3] or the generalized OMP algorithm [4].

When Θ is constructed, TMP performs tree search for sparse signal reconstruction. The tree has a maximum depth $\mathbf{K}$, and the goal is to find a path with depth K (candidate with cardinality $\mathbf{K}$) that has the smallest cost function $J(\Lambda) = \|y - \Phi_\Lambda x_\Lambda\|_2$. In each layer, new child path is generated by adding new element to the existing path. If we denote the path at layer (iteration) $\mathbf{i}$ as $\hat{s}_1^i$, then $\hat{s}_1^i$ is the causal set chosen in first $\mathbf{i}$ iterations.. Since visiting all possible paths is prohibitive, we introduce a pruning strategy to remove unpromising paths from the tree. This pruning decision is done by comparing the cost function of the path and the and the pruning threshold (smallest cost function among all paths investigated).

To make a proper decision, therefore, we have no way but to consider the cost function of full-blown path

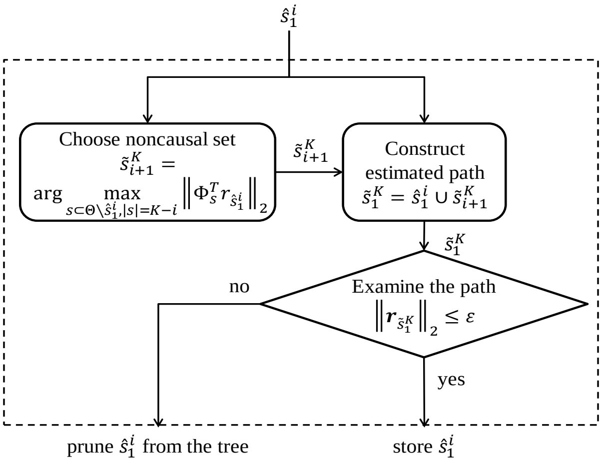

Figure 1 Pruning operation of TMP

and hence need a noncausal set $\tilde{s}_{i+1}^K$ in the pruning process. This noncausal set $\tilde{s}_{i+1}^K$ is temporarily needed for the pruning operation and can be easily obtained by choosing $K-i$ indices of columns in $\Omega \backslash \hat{s}_1^i$ whose magnitude of the correlation with the residual $r_{\hat{s}_1^i}$ is maximal, where

$$r_{\hat{s}_1^i} = y - \Phi_{\hat{s}_1^i} \hat{x}_{\hat{s}_1^i} \tag{1}$$

and

$$\hat{x}_{\hat{s}_1^i} = \Phi_{\hat{s}_1^i}^{+} y. \tag{2}$$

For example, if $K-i=2$, $\Omega \backslash \hat{s}_1^i = \{2,5,6,10,11,\ldots\}$ and

$$\left|\phi_5' r_{\hat{s}_1^i}\right| > \left|\phi_2' r_{\hat{s}_1^i}\right| > \left|\phi_{11}' r_{\hat{s}_1^i}\right| > \left|\phi_{10}' r_{\hat{s}_1^i}\right| > \left|\phi_6' r_{\hat{s}_1^i}\right| > \cdots,$$

then the noncausal set is $\tilde{s}_{i+1}^K = \{2,5\}$.

Once the noncausal set is obtained, it is combined with the causal set as $\tilde{s}_1^K = \hat{s}_1^i \cup \tilde{s}_{i+1}^K$ and then its residual $r_{\tilde{s}_1^K} = y - \Phi_{\tilde{s}_1^K} \hat{x}_{\tilde{s}_1^K}$ ($\hat{x}_{\tilde{s}_1^K} = \Phi_{\tilde{s}_1^K}^{+} y$) is computed. Using the l_2-norm of the residual $r_{\tilde{s}_1^K}$, TMP decides whether to prune the path $\hat{s}_1^i$ or not. To be specific, if the magnitude of $r_{\tilde{s}_1^K}$ is greater than the pruning threshold ε, then $\hat{s}_1^i$ is regarded to be hopeless and hence is removed from the tree. After the path examination in i-th layer is finished, the pruning threshold is updated to the minimum l_2-norm of the residual among all surviving paths.

3. Recovery Condition of TMP

In this section, we derive the exact recovery condition ensuring that TMP accurately recovers the K-sparse signals. In our analysis, we use the restricted isometry property (RIP) of the sensing matrix.

As mentioned, TMP consists of pre-selection and tree search. In our analysis, we show that the recovery condition of TMP is not much different from the condition of the pre-selection only and in fact guaranteed under more relaxed RIP bound.

In order to ensure the accurate identification of the support, TMP should satisfy the following two conditions:

1) At least one support index should be selected in the pre-selection process.

2) At least one true path5 should be survived in the tree pruning process.

The following Theorem describes the condition ensuring that at least one support is identified by the pre-selection stage.

Theorem 1 (Recovery condition of gOMP in first iteration [4]) At least one true index is chosen in the first iteration of gOMP under

$$\delta_{L+K} < \frac{\sqrt{L}}{\sqrt{L}+\sqrt{K}}. \tag{4}$$

Next, we provide the condition under which the true path is not removed from the tree.

Theorem 2 If $\hat{s}_1^i \subset T$, then $\tilde{s}_{i+1}^K$ is also true (i.e., $\tilde{s}_{i+1}^K \subset T$) under

$$\delta_M < \frac{1}{3}. \tag{4}$$

Proof: We skip the proof due to the page limitation.

Since at least one true index is chosen and the noncausal set of true causal path is also true, $\tilde{s}_1^K \subset T$ for any true path $\hat{s}_1^i \subset T$. Furthermore, since $\|r_T\|_2 = 0$, $\|r_T\|_2 < \varepsilon$ for any positive ε. In this scenario, that is, when Theorem 1 and 2 are jointly satisfied, the support is accurately identified by TMP.

Theorem 3 (Recovery condition of TMP) TMP recovers the sparse signals accurately under

$$\delta_Z < \frac{1}{3} \quad \text{if } K < 4L \tag{5}$$

$$\delta_Z < \frac{\sqrt{L}}{\sqrt{L}+\sqrt{K}} \quad \text{otherwise} \tag{6}$$

where $Z = \max\{M, L+K\}$.

Proof: The conditions (5) and (6) are obtained by choosing more strict condition between Theorem 1 and 2.

4. Empirical Results

In this section, we provide the empirical performance of TMP with existing sparse signal recovery algorithms. The sensing matrix of size 100×256 where each entry is from independent Gaussian random variable is used for the simulation. In order to measure the performance, the exact recovery ratio (ERR) and the mean squared error (MSE) are used for the noiseless and the noisy settings, respectively.

Fig. 2 shows the ERR performance of sparse signal recovery algorithms. Overall, we observe that the addition of tree search process provides substantial gain in performance. In particular, when K is large, performance gain caused by the tree search stage is noticeable. For example, the ERR of TMP at $K=35$ is 0.94 while that of OMP and CoSaMP are 0.23 and 0.62, respectively.

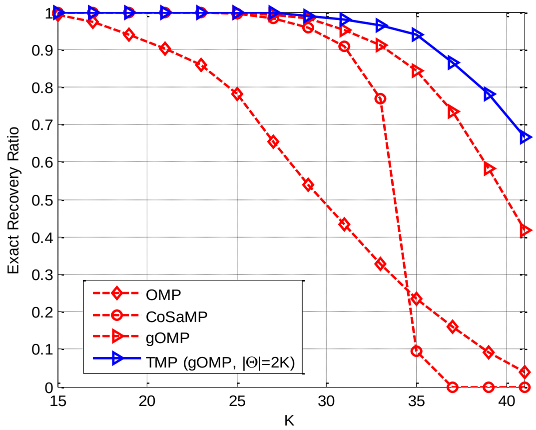

Figure 2 ERR performance

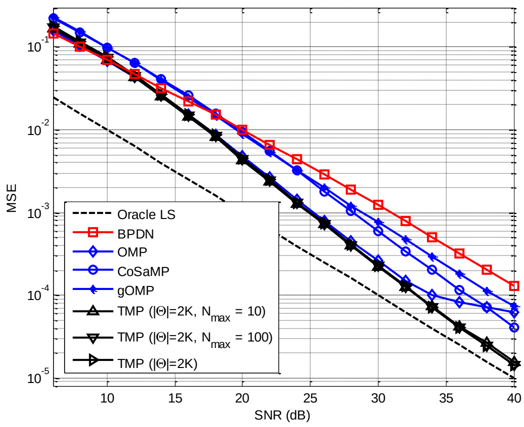

Figure 3 MSE performance

In Fig. 3, we plot the MSE performance of the sparse recovery algorithms as a function of signal−to−noise ratio (SNR) in the noisy setting. In this test, we set the sparsity level to K = 20 so that 8% of entries of $\mathbf{x}$ are nonzero. Overall, we observe that the performance gain of TMP improves with SNR. While the performance gap between the conventional sparse recovery algorithms and the Oracle estimator is maintained across the board, the performance gap between TMP and the oracle estimator gets smaller as SNR increases.

Fig. 4 shows the running time complexity of the sparse recovery algorithms as a function of the sparsity level $\mathbf{K}$. As seen in the figure, among greedy algorithms under test, OMP exhibits the smallest running time. Since TMP performs tree search to investigate multiple promising paths, it is no wonder that the running time complexity of TMP is higher than the rest of greedy algorithms. Nevertheses, by limiting the number of branching operations, computational burden of TMP can be saved dramatically. Due to the reduction in

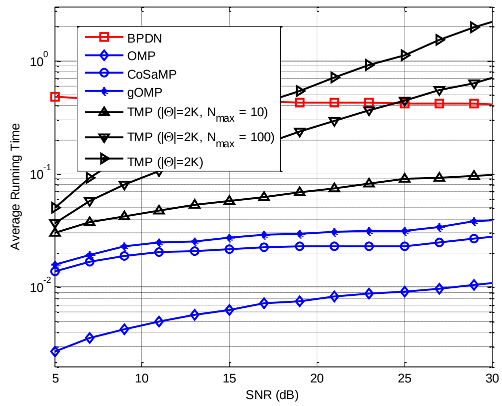

Figure 4 Complexity

number of investigated paths, we can observe that the running time complexity of TMP with limited branching is much smaller than that without limitation.

5. Conclusion

In this paper, we proposed an effective sparse signal recovery algorithm referred to as TMP. From the RIP−based analysis, we provided the sufficient condition under which the proposed approach selects the support. In addition, from the simulation, we could observe that the proposed TMP algorithm is effective in recovering sparse signals for both noiseless and noisy scenarios. In particular, we observed that TMP outperforms conventional sparse signal recovery algorithms and its performance is close to that of the Oracle estimator in high SNR regime.

References

[1] E. J. Candes, J. Romberg, and T. Tao, " Robustuncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489− 509, Feb. 2006.

[2] T. Tony Cai and L. Wang, " Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4680− 4688, July 2011.

[3] T. Zhang, " Sparse recovery with orthogonal matching pursuit under rip," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 6215− 6221, Sept. 2011.

[4] J. Wang, S. Kwon, and B. Shim, " Generalized orthogonal matching pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6202− 6216, Dec. 2012.

Affiliation

This work was partly supported by the ICT R&D program of MSIP/IITP [B0126-15-1017, Spectrum Sensing and Future Radio Communication Platforms] and the National Research Foundation of Korea (NRF) grant funded by the Korean government(MSIP)(2014R1A5A1011478)