ADAPTIVE REDUCED-SET MATCHING PURSUIT FOR COMPRESSED SENSING RECOVERY

Michael M. Abdel-Sayed, Ahmed Khattab, Mohamed F. Abu-Elyazeed

EECE Department – Cairo University, Giza, Egypt

ABSTRACT

Compressed sensing enables the acquisition of sparse signals at a rate that is much lower than the Nyquist rate. Various greedy recovery algorithms have been proposed to achieve a lower computational complexity compared to the optimal ℓ_1 minimization, while maintaining a good reconstruction accuracy. We propose a new greedy recovery algorithm for compressed sensing, called the Adaptive Reduced-set Matching Pursuit (ARMP). Our algorithm achieves higher reconstruction accuracy at a *significantly* low computational complexity compared to existing greedy recovery algorithms. It is even superior to ℓ_1 minimization in terms of the normalized time-error product, a metric that we introduced to measure the trade-off between the reconstruction time and error.

Index Terms— compressed sensing, matching pursuit, sparse signal reconstruction, restricted isometry property

1. INTRODUCTION

Traditionally, a signal is sampled at least at the Nyquist rate, which is double the signal bandwidth, for perfect reconstruction. On one hand, the Nyquist rate of some applications is so high that it is too expensive or even impossible to implement [1]. On the other hand, many applications compress the sampled signal for efficient storage purposes or for transmission over a much limited bandwidth. For example, a digital camera has millions of imaging sensors, but the acquired image is usually compressed into a few hundred kilobytes. Thus, a significant amount of the acquired data – the least significant information content – is sacrificed [2].

Compressed sensing simultaneously performs sensing and compression, thus the signal is sensed in a compressed form [1, 2, 3, 4]. This results in a considerable reduction in the costs of sampling and computation. The signal to be acquired should be either sparse or compressible, i.e. it has a few significant coefficients in a suitable basis or domain (e.g. Fourier, Wavelets, ..., etc.). This includes a large variety of signals such as natural images, videos, MRI, and radar signals [5]. The original signal can be recovered by convex optimization or greedy recovery algorithms.

In this paper, we introduce the Adaptive Reduced-set Matching Pursuit (ARMP): a new thresholding-based greedy recovery algorithm for compressed sensing. The ARMP algorithm has a good reconstruction capability at a *significantly* low computational complexity compared to existing greedy recovery algorithms [6, 7, 8, 9, 10]. ARMP correlates the residual signal with the columns of the sensing matrix in each iteration, attempting to find the support of the sparse signal (its nonzero indices). One or more of the elements of the correlation vector are selected, and their indices are merged into the support set, which is used to approximate the signal. The main idea behind ARMP is limiting the set of elements from which the new ones are selected in each iteration, and then adaptively picking the largest elements. ARMP then estimates the signal based on the identified support set and prunes it to only its largest samples. Unlike related works, ARMP adapts the number of the selected elements based on the distribution of the correlation values.

The rest of this paper is organized as follows. Section 2 presents compressed sensing preliminaries. Section 3 reviews the main greedy recovery algorithms ideas. We propose our ARMP algorithm in Section 4, and evaluate its performance in Section 5. We conclude the paper in Section 6.

2. COMPRESSED SENSING PRELIMINARIES

Consider a sparse signal $x \in \mathbb{R}^n$, of sparsity level k, and a measurement system that acquires m linear measurements. The measurement system samples the signal as

$$y = \Phi x, \tag{1}$$

where $\Phi \in \mathbb{R}^{m \times n}$ is the sensing or measurement matrix, and $y \in \mathbb{R}^m$ is the measured vector.

Alternatively, the signal x may not be itself sparse, but it may be sparse in a certain basis Ψ, i.e. $x = \Psi s$, where s is a sparse vector. Therefore, (1) can be rewritten as

$$y = \Phi \Psi s = As, \tag{2}$$

where Ψ is an $n \times n$ matrix whose columns form a basis in which x is sparse, and $A = \Phi\Psi$ is an $m \times n$ matrix.

In compressed sensing, the measured vector is of a much less dimension than the original signal, i.e. $m \ll n$. It was shown that the sparse (or compressible) signal x can be recovered from the measured signal y provided that the sensing matrix satisfies the Restricted Isometry Property (RIP) [1, 3].

Definition 1. A matrix A satisfies the restricted isometry property of order k if there exists a $\delta_k \in (0, 1)$ such that

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2 \tag{3}$$

holds for all k-sparse signals x.

Random matrices of certain distributions satisfy the RIP with high probability [11]. More specifically, if the entries of a matrix are independent and identically distributed (i.i.d.) and follow a Gaussian, Bernoulli or sub-Gaussian distribution, then the probability that the matrix does not satisfy the RIP is exponentially small.

Donoho originally suggested using ℓ_1 minimization for reconstructing the sparse signal as follows [4]

$$\hat{x} = \arg\min_z \|z\|_1 \text{ subject to } y = \Phi z \tag{4}$$

While ℓ_1 minimization is a powerful solution for the sparse problem, this solution is computationally expensive [1].

3. GREEDY RECOVERY ALGORITHMS

Motivated by the need to reach computationally inexpensive solutions, various greedy algorithms have been proposed in the literature for signal recovery. Greedy recovery algorithms iteratively attempt to find the signal support. In each iteration, the sparse signal is estimated based on the identified support set through least square minimization. Figure 1 shows a generic block diagram of the main steps for such greedy algorithms. The function of each block is described as follows:
1. Correlation. The residual r is correlated with the columns of the sensing matrix Φ to form a proxy signal g.
2. Selection and Support Merging. One or more of the elements of g with the largest absolute values are selected in each iteration. The indices of the selected elements are merged into the identified support set to approximate the signal.
3. Signal Estimation. The sparse signal is estimated based on the identified support using least square minimization. Some algorithms (thresholding-based) perform a pruning step to the estimated signal, keeping only the k largest absolute values of the signal, and setting the rest to zeros.
4. Residual Calculation. The residual is calculated based on the estimated signal.

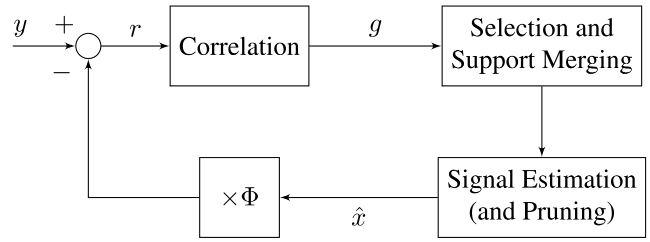

Fig. 1. General Block Diagram of Recovery Algorithms

3.1. Matching Pursuit Greedy Recovery Algorithms

One of the basic greedy recovery algorithms is the Orthogonal Matching Pursuit (OMP), which selects only one element from the correlation vector per iteration and adds its index to the identified support set [6]. Least square minimization is performed to estimate the signal without pruning. For a k-sparse signal, k iterations are required for signal reconstruction. Some algorithms add more than one index per iteration, resulting in faster convergence. The Generalized Orthogonal Matching Pursuit (GOMP) selects a fixed number of elements per iteration [7]. The Regularized Orthogonal Matching Pursuit (ROMP) chooses a set of k largest nonzero elements, then divides them into groups of comparable magnitudes, and selects the group of maximum energy [12]. The Stagewise Weak Orthogonal Matching Pursuit (SWOMP) selects the elements with absolute values larger than or equal to $\alpha \max_l |g_l|$, where $0 < \alpha < 1$ and $\max_l |g_l|$ is the largest magnitude element in the correlation vector [13]. The Stagewise Orthogonal Matching Pursuit (StOMP) [14] selects the elements larger than a threshold determined by the constant false alarm rate (CFAR) adaptive strategy used in radar systems [8]. Some algorithms speed up the minimization step using iterative matrix inversion techniques [15]. Other algorithms exploit the structure of the signal sparsity such as the Tree-based Orthogonal Matching Pursuit (TOMP) [16].

Analysis: Since OMP adds only one index per iteration, it requires a larger number of iterations than the rest of the algorithms. While ROMP improves the speed of OMP by selecting multiple elements per iteration, its reconstruction error is larger, especially for higher sparsity levels. The algorithm often results in adding a larger number of indices per iteration than is necessary, which usually includes ones not belonging to the support of the original signal. SWOMP and StOMP attempt to improve the selection stage. However, SWOMP still suffers from the same drawback of ROMP. Meanwhile, StOMP gives closer error performance to OMP, while requiring less execution time for higher sparsity levels. Furthermore, none of the aforementioned algorithms contain a pruning step. Thus, incorrectly selected indices will appear in the signal estimate, which degrades the performance.

3.2. Thresholding-based Greedy Recovery Algorithms

A common drawback in all the aforementioned greedy algorithms is that if an incorrect index is added to the support set in a certain iteration, it remains in all subsequent iterations, possibly degrading the performance. Thresholding-based algorithms handle this problem by applying a hard thresholding operator which removes one or more of the indices having the least energy from the identified support set. An example is the Compressive Sampling Matching Pursuit (CoSaMP) [9], which selects $2k$ (generally αk) elements per iteration and performs pruning after signal estimation. The Subspace Pursuit (SP) is another thresholding-based algorithm which selects k elements per iteration [10]. Pruning is then performed, followed by an extra least square minimization step.

Analysis: Such thresholding-based algorithms add a pruning step at the end of each iteration. However, both algorithms select a fixed number of elements per iteration ($2k$ and k, respectively). Such a selection is constant for all iterations and does not adapt to the distribution of the values of correlation. Furthermore, it usually results in selecting too many elements causing a larger reconstruction time, since more than necessary components are sorted in each iteration. A large and non-adaptive selection further increases the iteration time as more than necessary nonzero values have to be estimated by least square minimization. Selecting too many elements may also reduce the accuracy of the signal estimate, especially for larger sparsity levels, when incorrect indices are selected and kept through the subsequent pruning steps.

4. ADAPTIVE REDUCED-SET MATCHING PURSUIT

In this paper, we propose the Adaptive Reduced-set Matching Pursuit (ARMP), a thresholding-based greedy recovery algorithm. ARMP is based on two main ideas. First, ARMP limits the set of elements to be searched in each iteration to a reduced set containing only the βk largest magnitude elements in g. Second, ARMP selects from the reduced set the elements to be added to the identified support set as only those whose magnitudes are at least a fraction $0 < \alpha < 1$ of the maximum element. This contrasts with SWOMP [13] in which the selection is made from the entire g and not from a reduced set. ARMP results in selecting the indices which most likely belong to the support of the original signal, without taking too many indices per iteration. Such indices correspond to columns of the sensing matrix that have the highest correlation with the residual. The performance of ARMP depends on the proper choice of α and β.

Moreover, ARMP adapts to the distribution of the values of g. The number of selected elements is not constant for all iterations. For steeper (flatter) distributions of the absolute values of g, fewer (more) elements are selected.

After performing the selection step, the newly identified support set is merged with the previous one. Based on the merged set, a new signal estimate is generated using least square minimization. This gives the projection of y onto the subspace spanned by the columns of the sensing matrix corresponding to the identified support set. Then the signal is pruned keeping only its k largest values. The residual is then calculated based on the pruned signal. The previous steps are repeated until a stopping condition is met.

4.1. ARMP Algorithm

Initially, the signal estimate is set to a zero vector and the residual to the measured vector y. In each iteration, the following steps are performed:

1. Signal Proxy Formation. A signal proxy, g, is formed by correlating the residual with the sensing matrix columns.

2. Selection and Support Merging. The vector g is sorted in a descending order of absolute values. The elements which absolute values are larger than or equal to $\alpha \max_l |g_l|$, where $0 < \alpha < 1$, are selected from a reduced set containing the βk largest magnitude elements. The indices of the selected elements are united with the already identified support set.

3. Signal estimation. An estimate of the signal is formed by least square minimization. This is done via multiplication by the pseudo-inverse of the sensing matrix.

4. Pruning. The k largest magnitude components in the signal estimate are retained. The rest are set to zero.

5. Residual Calculation. The new residual is calculated from the pruned signal.

The algorithm terminates if the norm of the residual is less than ϵ_1 or if the difference between the norms of the residuals in two successive iterations is less than ϵ_2, whichever occurs first. Otherwise, a maximum of k iterations are performed.

Algorithm 1 summarizes the ARMP algorithm. The operator $L_k(\cdot)$ returns the index set of the k largest absolute values of the elements of its argument vector. The hard thresholding operator $H_k(\cdot)$ retains only the k elements with the largest absolute values and sets the rest to zero.

Algorithm 1 Adaptive Reduced-set Matching Pursuit

Input: Sensing matrix Φ, measurement vector y, sparsity level k, parameters α and β.
Initialize: $\hat{x}^{[0]} = 0, r^{[0]} = y, T^{[0]} = \emptyset$.
for $i = 1; i := i + 1$ until the stopping criterion is met **do**
  $g^{[i]} \leftarrow \Phi^* r^{[i-1]}$ {Form signal proxy}
  $J \leftarrow L_{\beta k}(g^{[i]})$ {Indices of βk largest magnitude elements in g}
  $W \leftarrow \{j : |g_j^{[i]}| \geq \alpha \max_l |g_l^{[i]}|, j \in J\}$ {Indices of elements in J larger than or equal to $\alpha \max_l |g_l^{[i]}|$}
  $T \leftarrow W \cup \text{supp}(\hat{x}^{[i-1]})$ {Support merging}
  $b|_T \leftarrow \Phi_T^\dagger y, b|_{T^c} \leftarrow 0$ {Signal estimation}
  $\hat{x}^{[i]} \leftarrow H_k(b)$ {Prune approximation}
  $r \leftarrow y - \Phi \hat{x}^{[i]}$ {Update residual}
end for
Output: Reconstructed signal $\hat{x}$

4.2. Performance Metrics

We next evaluate the performance of ARMP against existing related techniques as well as the original ℓ_1 minimization. Our performance metrics are the recovery time in seconds, the recovery error (defined as $\|x-\hat{x}\|_2/\|x\|_2$), and the percentage of correctly identified components of the signal. Furthermore, we introduce the normalized time-error product in which the product of the time and error of each algorithm is normalized

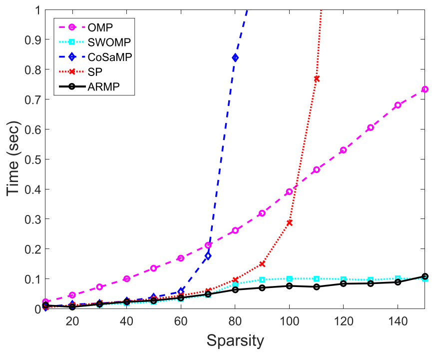

Fig. 2. Reconstruction time.

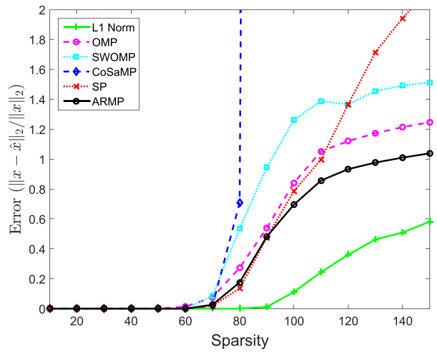

Fig. 3. Reconstruction error.

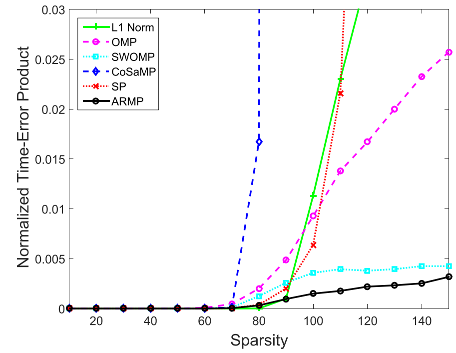

Fig. 4. Normalized time-error product.

over the largest product value of all algorithms. This metric accounts for the trade-off between time and error, since some algorithms give higher reconstruction accuracy at the expense of higher computational complexity.

5. PERFORMANCE EVALUATION

5.1. Simulation Setup

For each algorithm, the reported results are the average of the metrics evaluated for 100 independent trials. In each trial, we generate a random sparse signals of length n=1000 of uniformly distributed integers from 0 to 100. We take m=250 measurements. The sensing matrix A of dimensions $m \times n$ is randomly generated from i.i.d. Gaussian distribution with columns having unit ℓ_2 norm.

We plot the metrics versus sparsity levels from 10 to 150. We present the simulation results of six algorithms: ℓ_1 minimization, OMP, SWOMP, CoSaMP, SP, and ARMP. For SWOMP, we use $\alpha = 0.7$, as in [13]. The optimal values of α and β used for our ARMP algorithm were found to be 0.7 and 0.25, respectively. We defer how these values were obtained and the detailed impact of α and β to a sequel publication.

5.2. Simulation Results

Figure 2 depicts the recovery time versus the signal sparsity level. ℓ_1 minimization was omitted since it takes considerably longer time. ARMP and SWOMP have the least reconstruction times. However, our ARMP outperforms SWOMP at high sparsity levels above 70. The reconstruction time of other thresholding-based algorithms increases rapidly at sparsity levels of 70 for CoSaMP and 100 for SP.

Figure 3 compares the recovery error for all algorithms as a function of the sparsity level. For low sparsity levels, most of the algorithms produce very low errors, giving accurate signal estimates. However, as the sparsity of the signal increases, the differences between the reconstruction capability of the algorithms start to become clear. As expected, ℓ_1 minimization has the least error. Our proposed algorithm, ARMP, has the lowest error compared to all other greedy algorithms.

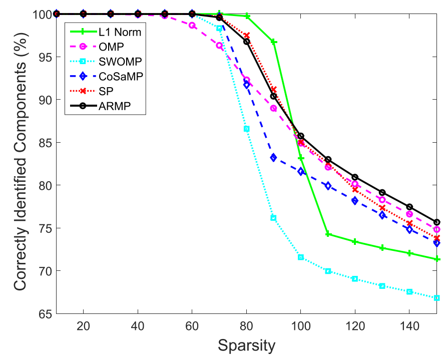

Fig. 5. Percentage of correctly identified signal components.

Figure 4 shows the normalized time-error product as a function of sparsity. ARMP gives the smallest product for most sparsity levels except for sparsity levels around 80 where ℓ_1 minimization was slightly smaller. This means that ARMP achieves a *high* reconstruction accuracy at *low* complexity compared to other algorithms including ℓ_1 minimization (which achieves slightly higher accuracy but at the expense of significantly longer time).

Figure 5 illustrates the percentage of correctly reconstructed components of the signal. ℓ_1 minimization is the best until a sparsity level of 100, followed by SP and ARMP which are close. After this point ARMP has the highest percentage, while the percentage of ℓ_1 minimization deteriorates. It is worth noting that it is not of vital importance to *perfectly* reproduce *all* the signal components in most applications.

6. CONCLUSION

In this paper, we have introduced ARMP: a new thresholding-based greedy algorithm for compressed sensing recovery. Simulation results have shown that the proposed ARMP algorithm is superior to main greedy recovery algorithms both in terms of reconstruction time and accuracy. Furthermore, ARMP is even superior to ℓ_1 minimization in terms of normalized time-error product, a measure which accounts for the trade-off between the reconstruction time and error.

7. REFERENCES

[1] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[2] E. J. Candès, "Compressive sampling," in *Proceedings of the international congress of mathematicians*, vol. 3. Madrid, Spain, 2006, pp. 1433–1452.

[3] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[4] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[5] Z. Li, W. Xu, Y. Wang, and J. Lin, "A tree-based regularized orthogonal matching pursuit algorithm," in *Telecommunications (ICT), 2015 22nd International Conference on*, 2015, pp. 343–347.

[6] J. Tropp and A. C. Gilbert, "Signal recovery from partial information via orthogonal matching pursuit," 2005.

[7] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *Signal Processing, IEEE Transactions on*, vol. 60, no. 12, pp. 6202–6216, 2012.

[8] M. A. Richards, *Fundamentals of radar signal processing*. Tata McGraw-Hill Education, 2005.

[9] D. Needell and J. A. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples," *Communications of the ACM*, vol. 53, no. 12, pp. 93–100, 2010.

[10] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *Information Theory, IEEE Transactions on*, vol. 55, no. 5, pp. 2230–2249, 2009.

[11] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[12] D. Needell and R. Vershynin, "Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 310–316, 2010.

[13] T. Blumensath and M. E. Davies, "Stagewise weak gradient pursuits," *Signal Processing, IEEE Transactions on*, vol. 57, no. 11, pp. 4333–4346, 2009.

[14] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 58, no. 2, pp. 1094–1121, 2012.

[15] G. Huang and L. Wang, "High-speed signal reconstruction for compressive sensing applications," *Journal of Signal Processing Systems*, pp. 1–12, 2014.

[16] H. Bui, C. La, and M. Do, "A fast tree-based algorithm for compressed sensing with sparse-tree prior," *Signal Processing*, vol. 108, pp. 628–641, 2015.