A COMPLEMENTARY MATCHING PURSUIT ALGORITHM FOR SPARSE APPROXIMATION

Gagan Rath and Christine Guillemot

IRISA-INRIA, Rennes, France

ABSTRACT

This paper introduces the concept of a complementary matching pursuit for sparse approximation. The algorithm is analogous to the classical matching pursuit but done in the row-space of the dictionary matrix. A deeper analysis of the algorithm shows that the residual error at any iteration may not be orthogonal to the immediately selected atom, however, this brings about the possibility of increasing the convergence speed and improving the sparsity of the solution vector. This is validated through simulations with a random dictionary created using the K-SVD algorithm.

1. INTRODUCTION

Consider the following system of equations:

$$A\mathbf{x} = \mathbf{b}, \tag{1}$$

where A is a matrix of dimension $K \times N$, $K < N$, and $\mathbf{b}$ is a known vector of dimension K. The columns of A are assumed to make a redundant basis for the K-dimensional space. The problem here is to find the unknown vector of coefficients $\mathbf{x}$ which has the minimum number of nonzero elements and which satisfies the above system of equations either exactly, or with some approximation error smaller than a specified threshold. In approximation theory, this problem is known as sparse approximation in a redundant basis. Referring to the above system, the exact sparse approximation problem thus can be posed as

$$\min\{\|\mathbf{x}\|_0 \ : \ A\mathbf{x} = \mathbf{b}\}, \tag{2}$$

where the L_0 norm denotes the number of non-zero elements. The problem which allows some approximation error can be posed as

$$\min\{\|\mathbf{x}\|_0 \ : \ \|A\mathbf{x} - \mathbf{b}\|_p \leq \delta\}, \tag{3}$$

for some $\delta \geq 0$. The norm p is usually 2, but could be 1 or ∞ as well.

This problem arises in many areas of signal processing, such as compression, denoising, indexing, signal recovery, etc., where the goal is to express a known signal as a linear sum of the fewest signals from a set of elementary signals. The elementary signals typically have well-defined properties, and thus it is easier to analyze the specified signal in terms of them. Using the terminology of sparse approximation, such elementary signals are called atoms and the collection is called a dictionary.

The above problem is NP-hard [1]. Except for the exhaustive combinatorial approach, there is no known method to find the exact solution under general conditions on the matrix A. There are basically two heuristic approaches in the literature which find approximate solutions with tractable complexity. One is called the matching pursuit (MP), which is a greedy algorithm [2, 3]. The other is known as the basis pursuit (BP), which relaxes the L_0 norm condition by the L_1 norm and solves the problem through linear programming [4, 5, 6]. BP algorithms can produce more accurate solutions than the matching pursuit algorithms but require higher complexity.

In this paper, we introduce the concept of a complementary matching pursuit (CMP). The algorithm is similar to the classical MP, but performed in the row-space of the dictionary matrix. Unlike MP, this may result in residual vectors that are not orthogonal to the immediately selected atoms. On the other hand, it chooses the coefficients which are closer to the actual true coefficients. As a result, the convergence speed and the sparsity are improved over the MP algorithm. We also show that, under the orthogonality of the rows of the dictionary matrix, the CMP is equivalent to the MP algorithm.

The goal here is not to present an algorithm which performs better than the state-of-the-art algorithms such as the orthogonal matching pursuit (OMP) [3] and basis pursuit (BP) [4, 5, 6], but to introduce a concept on par with the simplest pursuit algorithm. The idea here is to show that the classical algorithm done in a subspace of a higher dimensional space can have improved performance in terms of convergence speed and sparsity of solution. The orthogonal extension of the algorithm [7] is a suitable candidate to be compared with the advanced techniques. Because of the page limitation, we do not include this extension in this paper.

2. MATCHING PURSUIT (MP)

Matching pursuit [2] is an iterative greedy approach that selects the atom having the highest correlation with the residual vector at each iteration. Let α_i, $1 \leq i \leq N$, denote the ith atom with $\|\alpha_i\|_2 = 1$, $\forall i$. At the jth iteration, the algorithm finds

$$\alpha_j^{opt} = \underset{\alpha_i \in \mathcal{A}}{\arg\max} | < \mathbf{r}_{j-1}, \alpha_i > |, \tag{4}$$

where $\mathcal{A}$ denotes the dictionary of atoms, $< . >$ denotes the inner-product operation, and $\mathbf{r}_{j-1}$ denotes the residual at the $(j-1)$th iteration with $\mathbf{r}_0 = \mathbf{b}$. The inner product $< \mathbf{r}_{j-1}, \alpha_j^{opt} >$ represents the coefficient associated with the atom α_j^{opt}. Let us denote it as c_j. The algorithm then updates the residual as

$$\mathbf{r}_j = \mathbf{r}_{j-1} - c_j \alpha_j^{opt}. \tag{5}$$

The approximation at the jth iteration is given as $\mathbf{b}_j = \sum_{k=1}^{j} c_k \alpha_k^{opt}$ with $\mathbf{b}_0 = \mathbf{0}$.

The algorithm terminates when the norm of the residual falls below the desired approximation error bound, or when the number of distinct atoms in the approximation equals the desired limit. Notice that the selected atoms α_j^{opt}, $j = 1, 2, 3, ...$, may not be distinct when $j > 2$. Hence the sparse solution vector is obtained by adding up the coefficients c_j's with respect to the same atom.

The matching pursuit algorithm is very simple. But because of the sub-optimality [3], it suffers from slow convergence and poor sparsity result. The orthogonal matching pursuit (OMP) presented later in [3] removes this drawback by projecting the signal vector to the subspace spanned by the atoms selected up to any iteration. In the following, we present a simple modification of the MP which improves these factors without requiring the orthogonal extension.

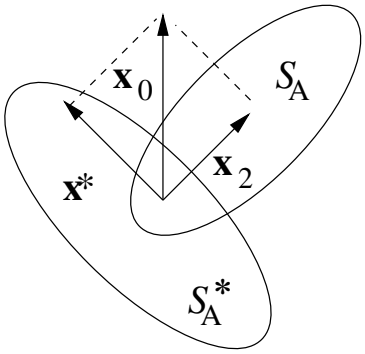

Figure 1: Subspace representations of $\mathbf{x}_2$, $\mathbf{x}_0$, and $\mathbf{x}^*$

3. COMPLEMENTARY MATCHING PURSUIT (CMP)

Consider the matrix A, whose columns are atoms. Since the columns of A are assumed to make a redundant basis, its rows are linearly independent. Thus the columns of A^T span a K dimensional subspace in the N-dimensional space. Let us denote this subspace by $\mathcal{S}_A$. The orthogonal complement of this subspace has dimension $N-K$. Let us denote this subspace by $\mathcal{S}_A^*$. Let G denote the $N \times (N-K)$ matrix whose columns are orthogonal and which span $\mathcal{S}_A^*$. G could be derived through well-known matrix operations such as QR factorization or Singular Value Decomposition (SVD) of A^T. Clearly, $AG = \mathbf{0}_{K\times(N-K)}$.

Now consider the system of equations in Eqn.1. Let us assume that the signal has an exact sparse representation and it is observed without any additive noise. The minimum L_2 norm solution to this system of equation is the pseudo-inverse:

$$\mathbf{x}_2 = A^T(AA^T)^{-1}\mathbf{b}.$$

Clearly, $\mathbf{x}_2$ lies in $\mathcal{S}_A$. The sparsest solution, denoted by $\mathbf{x}_0$, can be expressed as

$$\mathbf{x}_0 = \mathbf{x}_2 + \mathbf{x}^*,$$

where $\mathbf{x}^*$ is some nonzero vector. Since $A\mathbf{x}_0 = A\mathbf{x}_2 = \mathbf{b}$, $A\mathbf{x}^* = \mathbf{0}_K$. Therefore $\mathbf{x}^*$ lies in $\mathcal{S}_A^*$. This is shown in Fig. 1. Now finding $\mathbf{x}_0$ is equivalent to finding $\mathbf{x}^*$.

Since $\mathbf{x}_0$ is sparse, there is a relation between the components of $\mathbf{x}_2$ and $\mathbf{x}^*$. $\|\mathbf{x}_0\|_0 \leq K$ (which follows from the redundancy of the basis) implies that at least $N-K$ components of $\mathbf{x}^*$ have the same magnitudes but opposite signs as the corresponding components of $\mathbf{x}_2$[1].

Let us assume that these components of $\mathbf{x}^*$ are somehow known. Then the remaining components of $\mathbf{x}^*$ can be determined provided the rows of G having the same indices as these components make a spanning set for the $(N-K)$-dimensional space. This can be demonstrated as follows.

Let $\mathcal{I}$ denote the set of indices $\{1, 2, \ldots, N\}$, and let $\mathcal{I}_R$ denote the subset of indices for which the components of $\mathbf{x}^*$ and $\mathbf{x}_2$ are equal in magnitude but opposite in sign. Let $\mathcal{I}_E$ denote the complementary subset of indices, i.e., $\mathcal{I}_E = \mathcal{I} \backslash \mathcal{I}_R$. Let $\mathbf{x}_R$ and $\mathbf{x}_E$ denote vectors containing the corresponding components of $\mathbf{x}^*$, that is, $\mathbf{x}_R$ contains the components of $\mathbf{x}^*$ with indices $\mathcal{I}_R$ and $\mathbf{x}_E$ contains the remaining components. Let us partition G into G_R and G_E similarly, i.e., G_R contains the rows having indices $\mathcal{I}_R$ and G_E contains the remaining rows.

Now, since $\mathbf{x}^*$ lies in $\mathcal{S}_A^*$, there is a unique $(N-K)$-dimensional vector $\mathbf{z}_0$ such that

$$\mathbf{x}^* = G\mathbf{z}_0. \tag{6}$$

[1]If any of these components of $\mathbf{x}_2$ is 0, the corresponding component of $\mathbf{x}^*$ is also 0.

Using the above partitioning, we can write:

$$\mathbf{x}_R = G_R\mathbf{z}_0, \quad \mathbf{x}_E = G_E\mathbf{z}_0.$$

If the rows of G_R make a spanning set for the $(N-K)$-dimensional space, the columns of G_R are linearly independent. Therefore, given $\mathbf{x}_R$, we can compute $\mathbf{z}_0$ using the pseudo-inverse as

$$\mathbf{z}_0 = (G_R^T G_R)^{-1} G_R^T \mathbf{x}_R. \tag{7}$$

Substituting this above, we get

$$\mathbf{x}_E = G_E(G_R^T G_R)^{-1} G_R^T \mathbf{x}_R. \tag{8}$$

This shows that $\mathbf{x}_E$ can be recovered given $\mathbf{x}_R$. It is easy to see from above that, for a sparse solution, the associated G_R must be full column-rank. Because, otherwise, we can have infinitely many reconstructions for $\mathbf{x}_E$ resulting in infinitely many sparse representations with the same set of atoms (atoms with indices $\mathcal{I}_E$). This is a contradiction since the set of atoms in a sparse representation must be linearly independent.

Using the above result, the exact solution can be obtained by exhaustively searching over all possible combinations of t, $N > t \geq N-K$, components from the L_2 norm solution $\mathbf{x}_2$. We can assume that a given combination equals $\mathbf{x}_R$, and then compute $\mathbf{z}_0$ as in Eqn. 7 provided the corresponding G_R is full column-rank. Then we can compute $\mathbf{x}^*$ as in Eqn. 6 and subtract it from $\mathbf{x}_2$ to derive the sparse vector. The sparsest solution corresponds to the combination where this subtraction results in the minimum number of nonzero elements. This is similar to the known combinatorial approach. An alternative approach is to follow a greedy pursuit algorithm. But, before we describe the algorithm, let us first prove the following result, which we will use subsequently.

Lemma: Consider an $N \times M$ matrix Q, $M < N$, whose columns are orthogonal with identical Euclidean norm l and which has non-null rows with Euclidean norms less than l. Then any $(N-1)$ rows of Q span the M-dimensional space.

Proof: Let $\mathbf{q}$ denote any row of Q and let Q_r denote the matrix containing the remaining rows. Since Q is orthogonal with identical column norms, $Q^TQ = lI_M$, where I_M denotes the identity matrix of order M. Thus, $\mathbf{q}^T\mathbf{q} + Q_r^TQ_r = lI_M$. Multiplying $\mathbf{q}$ on the left side we get, $\mathbf{q}\mathbf{q}^T\mathbf{q} + \mathbf{q}Q_r^TQ_r = l\mathbf{q}$. Denoting the product $\mathbf{q}Q_r^T$ by $\mathbf{q}_r$, and simplifying, we get $(l - \mathbf{q}\mathbf{q}^T)^{-1}\mathbf{q}_rQ_r = \mathbf{q}$. This proves that $\mathbf{q}$ can be expressed as a linear sum of the rows of matrix Q_r. Since Q has rank M, there must be M linearly independent rows in Q_r and this proves the result. ■

We can now apply the above result to the matrix G. Let us assume that G is derived through QR factorization or SVD of A^T. Then $G^TG = I_{N-K}$ and the row norms of G are less than 1. Therefore, any combination of $N-1$ rows spans the $(N-K)$-dimensional space. This result helps us to apply the matching pursuit in the complementary space.

3.1 A greedy algorithm

This algorithm is similar to the MP algorithm. At every iteration, it identifies the atom that produces the smallest residual vector and computes the corresponding coefficient. However, the residual vector is defined differently in this case. Instead of a K-dimensional vector as in the case of MP, here it is an N-dimensional vector.

Let $\mathbf{r}_j$ denote the residual at the jth iteration with initialization $\mathbf{r}_0 = \mathbf{x}_2$. Let R_i denote the set of indices except the ith index, i.e., $R_i \equiv \mathcal{I}/\{i\}$. Let $\mathbf{r}_j[R_i]$ denote the vector containing all the elements of $\mathbf{r}_j$ except the ith element. Similarly, let G_{R_i} denote the matrix containing all the rows of matrix G except the ith row. At the jth iteration, the algorithm computes

$$\mathbf{e}_j^i = \mathbf{r}_{j-1} - G(G_{R_i}^T G_{R_i})^{-1} G_{R_i}^T \mathbf{r}_{j-1}[R_i], \quad 1 \leq i \leq N. \tag{9}$$

The second term on the right computes the orthogonal vector in $\mathcal{S}_A^*$ for the atom α_i. Because of the result stated earlier, G_{R_i} has full column rank for any i, and therefore the above matrix inversion operation is valid.

The optimal atom is found by minimizing the pth norm:

$$\alpha_j^{opt} = \arg\min_{\alpha_i \in \mathcal{A}} \|\mathbf{e}_j^i[R_i]\|_p, \tag{10}$$

where $\mathbf{e}_j^i[R_i]$ denotes the vector $\mathbf{e}_j^i$ without the ith component, and p can be 1, 2, or ∞. If k denotes the index of the optimal atom, the coefficient with respect to the optimal atom and the new residual vector are computed as

$$c_j = \mathbf{e}_j^k[k] \quad \text{and} \quad \mathbf{r}_j = \mathbf{e}_j^k \odot \mathbf{1}_{k=0}, \tag{11}$$

where $\mathbf{1}_{k=0}$ denotes a vector with all elements 1 except the kth element, which is 0, and '$\odot$' denotes the element-wise multiplication.

The approximation at the jth iteration is given as $\mathbf{b}_j = \sum_{k=1}^{j} c_k \alpha_k^{opt}$. Thus the approximation error at the jth iteration is given as

$$\epsilon_j = \mathbf{b} - \mathbf{b}_j = \mathbf{b} - \sum_{k=1}^{j} c_k \alpha_k^{opt}. \tag{12}$$

Notice that, unlike the MP, here the approximation error is not the same as the residual error. The algorithm terminates when the norm of the approximation error falls below the desired error bound, or when the number of distinct atoms in the approximation equals the desired limit. The coefficients of the sparse solution vector are derived by adding up the coefficients c_j's with respect to the same atom.

Also, notice that there is a zero element in the residual vector at each iteration, which corresponds to the index of the optimal atom in that iteration. Hence, what is minimized in (10) is actually the norm of the residual. Obviously, if all the remaining elements of the residual vector are zero, the algorithm terminates at that iteration. Further, the residual vector at each iteration is orthogonal to the columns of G. This can be proved easily as follows: Let k denote the index of the optimal atom at the jth iteration. Then,

$$G^T \mathbf{r}_j = G_{R_k}^T \mathbf{e}_j^k[R_k].$$

Using the expression from Eqn. 9, we get

$$G^T \mathbf{r}_j = G_{R_k}^T (\mathbf{r}_{j-1}[R_k] - G_{R_k} (G_{R_k}^T G_{R_k})^{-1} G_{R_k}^T \mathbf{r}_{j-1}[R_k]) = \mathbf{0}.$$

Therefore $\mathbf{r}_j$ lies in the subspace $\mathcal{S}_A$, like $\mathbf{x}_2$. As a consequence of this, in the $(j+1)$th iteration, the second term in Eqn. 9 will be equal to zero for the kth atom, and thus the residual will remain unchanged for that atom. This is similar to the orthogonality of the residual vector to the immediately selected atom in the MP algorithm.

4. CONVERGENCE OF CMP

It is well known that the MP algorithm is convergent [3]. This is seen from the fact that the residual vector is orthogonally projected onto the atoms that make a redundant basis, and thus the residual is a decreasing function of the iteration number. The same argument holds for the CMP as well.

Referring back to the expression in Eqn. 9,

$$\begin{aligned} \mathbf{r}_j[R_k] &= \mathbf{r}_{j-1}[R_k] - G_{R_k} (G_{R_k}^T G_{R_k})^{-1} G_{R_k}^T \mathbf{r}_{j-1}[R_k] \\ &= (I_{N-K} - G_{R_k} (G_{R_k}^T G_{R_k})^{-1} G_{R_k}^T) \mathbf{r}_{j-1}[R_k]. \end{aligned}$$

This shows that the non-null part of the residual vector at the jth iteration is actually the difference between $\mathbf{r}_{j-1}[R_k]$ and its projection onto the subspace spanned by the columns of G_{R_k}. Since the atoms make a redundant basis, there is at least one atom for which this projection is non-zero. This implies that $\mathbf{r}_j$ is smaller than $\mathbf{r}_{j-1}$ and thus the algorithm is guaranteed to converge.

We have shown that G_{R_k} is full column-rank, i.e., it has rank $N-K$. Thus the residual error lies in a $(K-1)$-dimensional subspace. This is also true for the MP since the residual error is the difference between a K-dimensional vector and its projection on an atom (column vector). However, in the case of CMP, the projection being onto a subspace of dimension $(N-K)$, the error may die out faster with each iteration (faster convergence) than in the case of MP. Compare this with the case of OMP [3] where the error lies in a subspace whose dimension decreases by 1 at each iteration.

5. CMP VS MP

From the above explanation, it seems that the CMP must be somehow related to the MP. In the following we derive the expressions for the coefficient of the optimal atom and the residual energy after the first iteration and relate them to those in MP. The coefficient and the residual energy after the jth iteration can be obtained by replacing $\mathbf{x}_2$ by the residual vector $\mathbf{r}_{j-1}$.

At the first iteration, the nonzero components of the residual error with respect to the ith atom is given as

$$\mathbf{e}_1^i[R_i] = (I_{N-K} - G_{R_i} (G_{R_i}^T G_{R_i})^{-1} G_{R_i}^T) \mathbf{x}_2[R_i]. \tag{13}$$

Therefore

$$\|\mathbf{e}_1^i[R_i]\|^2 = \mathbf{x}_2^T[R_i] (I_{N-K} - G_{R_i} (G_{R_i}^T G_{R_i})^{-1} G_{R_i}^T) \mathbf{x}_2[R_i]. \tag{14}$$

Let $\mathbf{g}_i$ denote the ith row of G. Since $G^T G = I_{N-K}$, $G_{R_i}^T G_{R_i} = I_{N-K} - \mathbf{g}_i^T \mathbf{g}_i$. Using the matrix inversion lemma [8], and the orthogonality between the columns of G and $\mathbf{x}_2$, the above expression can be simplified as

$$\|\mathbf{e}_1^i[R_i]\|^2 = \|\mathbf{x}_2\|^2 - \frac{(\mathbf{x}_2[i])^2}{1 - \mathbf{g}_i \mathbf{g}_i^T} \tag{15}$$

Now, using the orthogonality between G and A^T, it can be proved that $1 - \mathbf{g}_i \mathbf{g}_i^T = \alpha_i^T (AA^T)^{-1} \alpha_i$. Therefore

$$\|\mathbf{e}_1^i[R_i]\|^2 = \|\mathbf{x}_2\|^2 - \frac{(\mathbf{x}_2[i])^2}{\alpha_i^T (AA^T)^{-1} \alpha_i}. \tag{16}$$

If the kth atom produces the minimum error, then the residual energy is

$$\|\mathbf{r}_1\|^2 = \|\mathbf{x}_2\|^2 - \frac{(\mathbf{x}_2[k])^2}{\alpha_k^T (AA^T)^{-1} \alpha_k}. \tag{17}$$

The coefficient associated with the kth atom is

$$c_1 = \mathbf{e}_1^k[k] = \mathbf{x}_2[k] - \mathbf{g}_k (G_{R_k}^T G_{R_k})^{-1} G_{R_k}^T \mathbf{x}_2[R_k]. \tag{18}$$

Using the matrix inversion lemma as before and simplifying, we get

$$c_1 = \frac{\mathbf{x}_2[k]}{1 - \mathbf{g}_k \mathbf{g}_k^T} = \frac{\alpha_k^T (AA^T)^{-1} \mathbf{b}}{\alpha_k^T (AA^T)^{-1} \alpha_k}. \tag{19}$$

From Eqn. 12, the approximation error energy is

$$\|\epsilon_1\|^2 = \|\mathbf{b} - c_1 \alpha_k\|^2 = \mathbf{b}^T \mathbf{b} - 2 c_1 \alpha_k^T \mathbf{b} + c_1^2. \tag{20}$$

To compare these quantities with those in the MP, let us recall the MP algorithm. At the first iteration, the residual error with the ith atom is given as

$$\mathbf{e}_{MP}^{i} = (I_K - \alpha_i \alpha_i^T)\mathbf{b}, \tag{21}$$

where we have used the subscript 'MP' to distinguish the error from that in the case of the CMP. Therefore the error energy is

$$\|\mathbf{e}_{MP}^{i}\|^2 = \mathbf{b}^T (I_K - \alpha_i \alpha_i^T)\mathbf{b}. \tag{22}$$

If the kth atom produces the minimum error, then its coefficient is $c_{MP} = \alpha_k^T \mathbf{b}$ and the residual error is $\mathbf{b}^T (I_K - \alpha_k \alpha_k^T)\mathbf{b}$.

Now, if the rows of A are orthogonal, then it is easy to show that $\|\mathbf{e}_1^i[R_i]\|^2$ in Eqn. 16 is equal to $\|\mathbf{e}_{MP}^i\|^2$ in Eqn. 22. Therefore both the CMP and the MP identify the same atom as the optimal atom. Further, it is also easy to show that $c_1 = c_{MP}$. These results are true for all subsequent iterations as well. Therefore the CMP will produce the same sparse solution as the MP.

Let us consider the general case when the rows of A are not orthogonal. Let us assume that the MP algorithm also identifies α_k as the optimal atom at the first iteration. Thus, we can write $\mathbf{b} = c_{MP}\alpha_k + \mathbf{e}_{MP}^k$. Substituting this in the expression for c_1 in Eqn. 19, we get

$$c_1 = c_{MP} + \frac{\alpha_k^T (AA^T)^{-1} \mathbf{e}_{MP}^k}{\alpha_k^T (AA^T)^{-1} \alpha_k}. \tag{23}$$

The above expression is interesting. First of all, it shows that, if $\mathbf{b}$ is collinear with any of the atoms, then c_1 is equal to c_{MP}. This is so because $\mathbf{e}_{MP}^k$ is a null vector in this case. This also shows that, in general, the approximation error with CMP is not orthogonal to the optimal atom. As a result the approximation error after the first iteration is more than that of the MP algorithm. This is corroborated by Eqn. 20, which can now be expressed as

$$\|\epsilon_1\|^2 = \mathbf{b}^T\mathbf{b} - 2c_1 c_{MP} + c_1^2 = \|\mathbf{e}_{MP}^k\|^2 + (c_1 - c_{MP})^2. \tag{24}$$

However, this result does not extend to all subsequent iterations. On the contrary, the offset term may improve the accuracy of the atoms and the convergence speed by getting closer to the actual coefficient magnitudes. The offset term is actually the ratio of the kth elements of the L_2 norm solutions of equations $A\mathbf{x} = \mathbf{e}_{MP}^k$ and $A\mathbf{x} = \alpha_k$. To see its effect clearly, consider the trivial example of a dictionary having 2 atoms each having 2 elements. Assume that the two atoms are as shown in Fig. 2. The known vector $\mathbf{b}$ has a unique representation in terms of these two atoms (vectors representing the sides of the parallelogram). Now the MP algorithm will identify atom α_1 and α_2 alternately and each iteration will reduce the residual error. The solution will converge to the unique solution ultimately. The CMP algorithm, however, will find the true coefficients in 2 iterations. The offset term helps in finding the true coefficient of α_1 in the first iteration. Though it makes the residual error larger than that of MP, it is collinear with α_2. Therefore, the second iteration results in zero residual error. This example provides us some intuitive idea about how, in a general case, the CMP proceeds as against the MP. In general, we can expect a faster convergence and better sparsity compared to the MP.

If we look very closely, we find that the CMP is equivalent to an MP performed to solve the following system of equations:

$$A^T (AA^T)^{-1} A\mathbf{x} = \mathbf{x}_2. \tag{25}$$

The equivalence can be proved by deriving the expressions for the coefficient and the residual error for the MP performed on this transformed system of equations. This system of equations is obtained by pre-multiplying the original system in Eqn. 1 by $A^T(AA^T)^{-1}$. In the transformed system, the atom α_i gets replaced by the atom $A^T(AA^T)^{-1}\alpha_i$, which lies in the subspace $\mathcal{S}_A$. So the CMP is basically an MP applied in the row-space of the dictionary matrix.

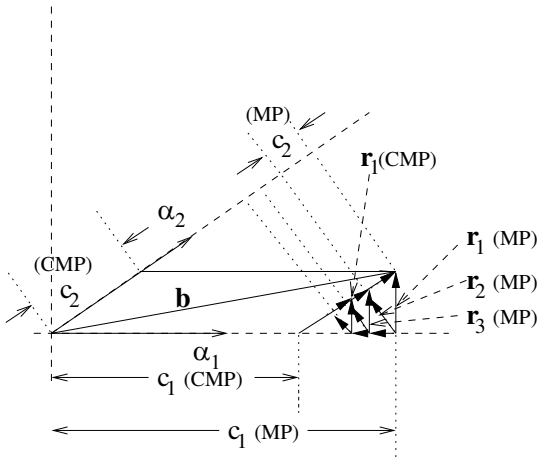

Figure 2: CMP vs MP with two 2-D atoms. MP iterations consist of alternate orthogonal projections onto the two atoms, which converges to the sides of the parallelogram ultimately. CMP converges only in two iterations.

6. SIMULATION RESULTS

In order to compare the two algorithms, we performed experiments with a random dictionary. The dictionary was created using the recently proposed K-SVD algorithm [9]. The dictionary consisted of 55 atoms with each atom having 32 elements. The atoms were normalized with respect to the L_2 norm. Then 1000 signal vectors of dimension 32 were generated, each created by a linear combination of certain number of atoms with randomly generated coefficients, but no additive noise. The atoms themselves were selected from random combinations.

In the first experiment, we compared the convergence of the CMP with those of the MP and OMP algorithms. We generated signals with 8 random atoms and used them as input signals to the three algorithms. Fig. 3 shows the mean square residual errors at different number of iterations. The mean was computed over all the 1000 signal blocks and then it was normalized with respect to the mean signal energy. We observe that, after some initial few iterations, the error dies out faster with the CMP than the MP. The convergence speed of the CMP relative to the OMP is also noteworthy.

In the second experiment, we varied the number of generating atoms for the input signal. We specified an error bound of 0.001 for each component of the signal vector and a maximum of 16 atoms in the approximation. Fig. 4 displays the number of atoms identified by different algorithms. We observe that, when the number of generating atoms is more than 3, the CMP results in a sparser solution than the MP. Furthermore, the sparsity of the CMP is comparable to that of the OMP up to about 7 generating atoms. Fig. 5 displays the fraction of true atoms identified in the approximation. It is interesting to see that the CMP outperforms not only MP, but also OMP. Finally Fig. 6 shows the mean square error resulting from different algorithms. We see that the CMP results in lesser error than the MP. However, the error produced by the OMP is the smallest among the three algorithms. This result motivates us to test the orthogonal version of the CMP, whose results will be presented in a future paper.

7. CONCLUSIONS

In this paper, we have introduced the concept of a complementary matching pursuit. Using the complementary subspace of the row-space of the dictionary matrix, we have presented the matching pursuit from another perspective. We have derived the expressions for the resulting coefficients and the residual errors and shown that the algorithm is equivalent to an MP algorithm performed in the row-space of the dictionary matrix. Simulations with a K-SVD opti-

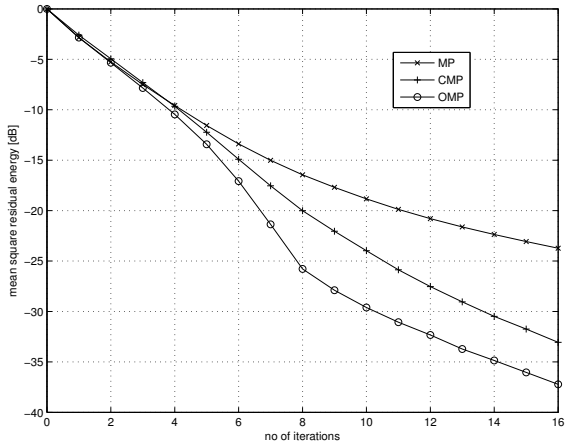

Figure 3: Residual energy for different number of iterations. Number of generating atoms is 8.

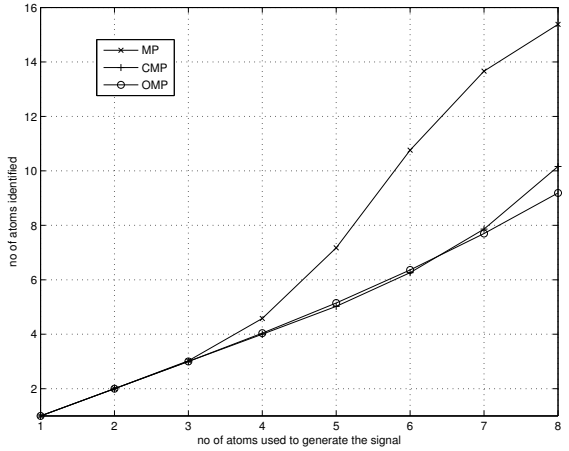

Figure 4: No of atoms identified versus the number of actual atoms. Error bound per component 0.001, maximum 16 atoms.

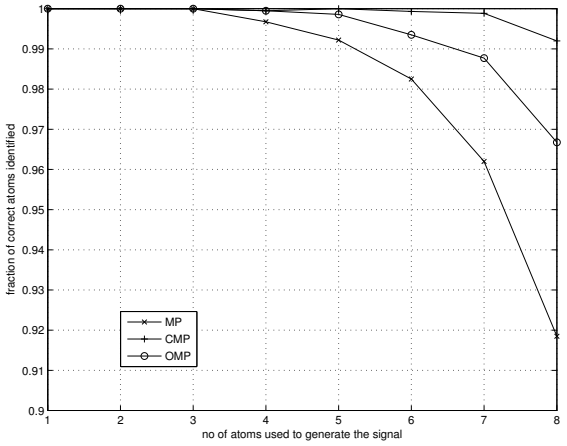

Figure 5: Fraction of correct atoms detected versus the number of atoms. Error bound per component 0.001, maximum 16 atoms.

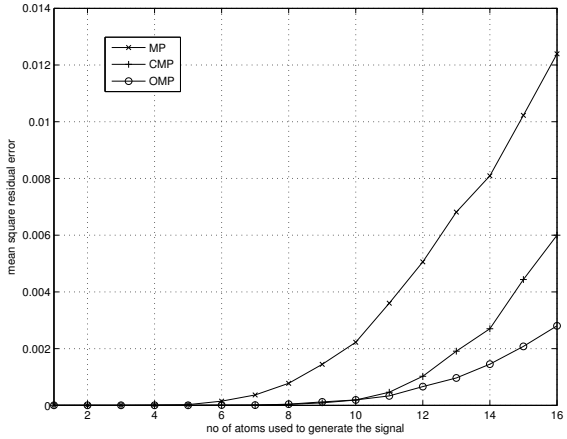

Figure 6: Mean square residual error versus the number of atoms. Error bound per component 0.001, maximum 16 atoms.

mized dictionary showed that the CMP converges faster and produces a sparser solution than the MP. Furthermore, in terms of identifying true atoms, it outperforms both the MP and the OMP algorithms.

This work can be extended in several ways. First, it is interesting to test the orthogonal version of the CMP and to compare it with the OMP. Second, from the theoretical point of view, it is important to analyze the presented algorithm for the number of iterations required for convergence, and for the resulting number of atoms, given a certain error bound. Third, for comparison with the MP, it is interesting to investigate the convergence and the sparsity of CMP vis-à-vis MP in a precise manner. The results presented here considered a dictionary of a fixed size. Since the dimension of the subspace $\mathcal{S}_A^*$ depends on the size of the dictionary, the convergence and the sparsity of the CMP solution are expected to be increasingly better as the size of the dictionary increases. Finally, the presentation here has considered only noiseless signals. For noisy signals, the algorithm can be highly sensitive depending on the stability of the matrix $(AA^T)^{-1}$. Hence it is important to investigate the algorithm for noisy signals and to develop techniques to circumvent the stability issues.

REFERENCES

[1] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.,* 24, 2, 227–234, April 1995.

[2] S. Mallat and Z. Zhang, "Matching pursuit in a time-frequency dictionary," *IEEE Trans. SP,* 41, 12, 3397–3415, 1993.

[3] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad,"Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decompositions," *Proc. 27th Asilomar Conf. on Sig., Sys. and Comp.,* 1993.

[4] S. S. Chen, D. L. Donoho, and M. A. Saunders,"Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.,* 20, 1, 33–61, 1988.

[5] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution," *Comm. Pure Appl. Math.,* 59, 6, 797–829, 2006.

[6] J. J. Fuchs, "Recovery of exact sparse representations in the presence of bounded noise," *IEEE Trans. Inform. Theory,* 51, 10, 3601–3608, 2005.

[7] G. Rath and C. Guillemot, "Sparse approximation with an orthogonal complementary matching pursuit algorithm," submitted to *IEEE SiPS 2008*, Oct. 2008, Washington, USA.

[8] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra,* SIAM, 2000.

[9] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. SP,* 54, 11, 4311–4322, 2006.