Abstract—This paper presents an Sparsity Update Subspace Pursuit (SUSP) algorithm for compressed sparse signal reconstruction with unknown sparsity. From practical point of view, the sparsity information is usually unavailable in many applications. In particular, the compressed spectrum sensing application is considered in this paper. The proposed SUSP algorithm begins with sparsity estimation and iteratively updates the sparsity based on the residual value with subspace pursuit approach. A termination criterion is developed to facilitate the convergence of the sparse update iteration. Moreover, a tail biting rule is devised to refine the reconstruction. Consequently, the sparse signal is recovered and the reconstruction performance is improved. The recovery rate performance is numerically evaluated for both the known sparsity and the unknown sparsity cases. For each case, the recovery mean square errors are also presented for the noisy environment. The results show that the resulting performance outperforms the popular methods using the maximum pursuit or the subspace pursuit based approaches.

Index Terms—Compressive sensing (CS), sparse signal recovery, greedy algorithm, sparsity estimation.

I. INTRODUCTION

The compressive sensing (CS) technology has attracted much attention which has wide applications in many fields. Satisfying the sparsity and the incoherent properties, the sparse signal can be reconstructed from far fewer samples than the source bandwidth [1] [2].

To recover the sparse signal, many reconstruction methods have been studied. In general, these algorithms can be classified into two categories: the basis pursuit (BP) [3] approach and matching pursuit (MP) approach [4] [5] [6]. The BP based approach explores the use of convex relaxation to approximate the sparse signal [7] [8]. We can reconstruct $x$ by using $\ell_1 - \text{norm}$ minimization instead of $\ell_0 - \text{norm}$ if the measurement matrix $A$ satisfies the Restricted Isometry Property (RIP) with a constant parameter [1]. While BP plays an important role in CS recovery, its high complexity prevents it from many practical applications [9] [10].

Several low complexity reconstruction algorithms have been proposed based on the concept of Matching Pursuit which are generally greedy search methods and use vector projection. Mallat and Zhang first to solve the under-determined system by correlation maximization at each step [4]. The Orthogonal Matching Pursuit (OMP) method takes advantage of orthogonal projection to improve the reconstruction [11] [12] [13] [14]. Many variants based on MP approach include the Stagewise OMP (stOMP) [15], the Regularized OMP (ROMP) [16] and the Ordered OMP (OOMP) [17]. More recently, backtracking algorithm include Compressive Sampling Matching Pursuit (CoSaMP) [18] and Subspace Pursuit (SP) [19] offers a significant improvement that enables the algorithms to eliminate wrong selected atoms. Although the greedy MP based algorithms provide a low complexity way to reconstruct the $s$-sparse signal, the sparse level $s$ is usually required as a prior knowledge for signal recovery.

To detect the sparse signal without prior knowledge of sparsity level is a practical issue in the compressive spectrum sensing application. Though the compressed spectrum sensing has been studied extensively, most literatures assume prior knowledge of sparsity about primary users and the situation of unknown sparsity is much less addressed. The Sparsity Adaptive Matching Pursuit (SAMP) method modifies from the SP method and deals with unknown sparsity [20]. The Sparsity Update Compressive Sampling Matching Pursuit (CSAMP) is quite similar to SAMP which changes the updating function of SP to CoSaMP. The Backtracking-based Adaptive OMP (BAOMP) [21] is improved based on OMP incorporating a simple backtracking technique to approach the sparse level.

In this paper, we propose an Sparsity Update Subspace Pursuit (SUSP) algorithm which deals with unknown sparsity problem. The proposed SUSP algorithm begins with sparsity estimation and progressively update the sparsity based on the subspace pursuit approach. By adaptively increasing the size of candidate list, the SUSP solves the sparse solutions problem when the sparsity $s$ is not available. A tail biting rule is devised to refine the reconstruction. Moreover, termination criterion is developed to facilitate the convergence of the algorithm. The target signal of reconstruction is updated and the recovery performance is improved. If the sparsity is given in the proposed SUSP, our algorithm also outperforms most greedy algorithms with sparsity pre-known.

The remainder of this paper is organized as follows. In Section II, the signal model is described. The greedy algorithms to solve compressive sensing problem are discussed briefly. Section III presents the proposed Sparsity Update Subspace Pursuit algorithm which solves both pre-known and unknown sparsity problems. Section IV presents the numerical simulation and compare the performance of proposed SUSP with the existing arts. Finally, Section V gives the conclusion.
II. SIGNAL MODEL

Consider the received signal \( y = Ax \), where \( x \in \mathbb{R}^{N \times 1} \) is the sparse signal under recovery, and \( A \in \mathbb{R}^{M \times N} \) is the measurement matrix. The signal \( x \) is called \( s \)-sparse when there are \( s \) non-zero entries, i.e.,

\[
\|x\|_0 = |\{j : x_j \neq 0\}| \leq s \ll N \tag{1}
\]

In this paper, we assume that the non-zero entries in \( x \) are i.i.d. Gaussian random variables with zero mean and unit variance and are independent of the entries in \( A \). The sparse signal \( x \) can be reconstructed through the linear projection \( y = Ax \), where \( A \) satisfies the RIP condition with size \( M \times N, s \log(N/s) \leq M \ll N \) [3].

The sparse signal reconstruction are recovered based on the measured signal \( y \). The matching pursuit is the most popular sparse signal construction approach for low complexity. Based on greedy search and vector projection and greedy search, the sparse signal can be recovered with high probability. With the RIP condition being satisfied for the measurement matrix \( A \), the columns of measurement matrix are nearly orthogonal. The projection value of \( y \) onto the columns of \( A \) will be large on those columns corresponding to the non-zero entries in original \( x \). So the columns in \( A \) with large correlation are more likely to be the indices to the original signal \( x \). The matching pursuit approach exploits this correlation property and chooses the atom with largest inner product in each iteration.

1) Correlation Test: The MP selects a column in measurement matrix \( A \) that maximizes the correlation between the residual vector and the column of the sensing matrix.

\[
\alpha_i = \arg\max_{A_j} |\langle r^{(i-1)}, A_j \rangle| \tag{2}
\]

where \( \alpha_i \) represents the atom chosen at the \( i \)th iteration, \( r^{(i-1)} \) is the residual vector of \( y \) after the \( (i-1) \)th iteration, and \( A_j \) is the \( j \)th column of \( A \).

2) Orthogonal Projection: Once \( \alpha_i \) has been selected, project the residual vector \( r^{(i-1)} \) onto \( \alpha_i \). The \( j \)th entry in \( x^{(i-1)} \) will be replaced by the value of projection. The new estimation of \( x \) is denoted as \( x^{(i)} \).

3) Residual Update: After orthogonal projection, MP updates the residual vector

\[
r^{(i)} = y - Ax^{(i)}. \tag{3}
\]

The procedure keeps searching solution to identify the large components in the residual until the iteration number \( i \) meets the sparsity \( s \). After \( s \) iterations, the MP algorithm comes to end and obtains the output of the estimated \( x_s \).

III. PROPOSED SPARSITY UPDATE SUBSPACE PURSUIT

The proposed SUSP algorithm deals with compressive sensing reconstruction problem with unknown sparsity which consists of four stages, (1) initial sparsity estimation, (2) correlation test, (3) sparsity update, and (4) tail biting. In the beginning the sparsity is estimated. The correlation test and the sparsity update procedures iterate till the termination is satisfied. Finally, the tail biting rule is applied to prune the atoms. The details are described as follows.

A. Sparsity Estimation

The sparsity level \( s \) of the sparse signal \( x \) plays an important role in compressive sampling recovery. The problem of solving the underdetermined system \( y = Ax \) without prior knowledge of \( s \) is a practical issue in CS applications. In SP and CoSaMP, the algorithms are based on the assumption that the residual value would decrease as the reliable candidates are merged into the support set. In SUSP, we estimate the sparsity based on observation of the measurement vector.

**Proposition 1.** Let the measurement matrix \( A \in \mathbb{R}^{M \times N} \) where each entries \( a_{i,j} \) are i.i.d. Gaussian random variables with zero mean and variance \( 1/M \) and the signal vector \( x \) be \( s \)-sparse. The non-zero element of \( x \) are i.i.d. Gaussian random variables with zero mean and unit variance. Then, given the measured signal vector \( y = Ax \), the sparsity \( s \) can be estimated by the expectation of the power of the length of \( y \) over the length of any column in \( A \). That is,

\[
s_{\text{est}} = E\left[ \frac{\|y\|_2}{\|A_{j}\|_2^2} \right], \forall j, \tag{4}
\]

where \( A_j \) represents the \( j \)th column of \( A \).

**Proof:** Let \( n_A \) denote the expectation of the 2-norm of the \( A_j \), then

\[
n_A = E\left[\|A_{j}\|_2^2\right] = E\left[\sum_{i=1}^{M} a_{ij}^2\right] = \sum_{i=1}^{M} E[\alpha_{ij}^2] = 1. \tag{5}
\]

The measurement vector \( y \) is formed by randomly choosing \( s \) columns from \( A \). The \( k \)th entry of \( y \) is \( i.i.d. \) with \( E[y_k] = 0 \) and \( \text{Var}[y_k] = s/M, k = 1, \ldots, M \). The expectation of 2-norm of \( y \) is

\[
E\left[\sqrt{\sum_{k=1}^{M} y_k^2}\right] = \sqrt{s} \tag{6}
\]

Hence the expectation of the square of 2-norm of \( y \) divided by 2-norm of \( i \)th column in \( A \) is

\[
E\left[\frac{\|y\|_2^2}{\|A_{j}\|_2^2}\right] = E\left[\frac{\|y\|_2^2}{\|A_{j}\|_2^2}\right] = E\left[\frac{\sum_{k=1}^{M} y_k^2}{\sum_{i=1}^{M} a_{ij}^2}\right] = s \tag{7}
\]

which yields the initial sparsity estimate \( s_{\text{est}} \).

The first step in SUSP estimates the sparsity if \( s \) is not known a priori. However, in the case that the sparse level \( s \) is given, we can simply set the initial sparsity estimate \( s_{\text{est}} \) to be \( s \).

B. Correlation Test

The step of correlation test is similar to that in SP or CoSaMP algorithms. The SUSP contains preliminary and final correlation tests. The correlation between the residual vector, \( r_p \), and the columns of measurement matrix is calculated. Then \( s_{\text{est}} \) candidate atoms of largest absolute values are selected as support set \( T \),

\[
T = \text{Large}(A^T r_p, s_{\text{est}}), \tag{8}
\]
where \( \text{Large}(u, k) \) denotes the function that picks \( k \) indices corresponding to the largest \( k \) absolute values in vector \( u \). Let \( F_p \) denote the support set from the previous iteration. The selected coordinates in support set \( T \) merges with the set of coordinates in \( F_p \) into candidate set, denoted as set \( C \)

\[
C = F_p \cup T.
\]

Let \( A_c \) be the submatrix with columns selected from \( A \) based on the indices of \( C \). The final correlation test is applied to prune the atoms in the candidates,

\[
F = \text{Large}(A_c^\top y, s_{\text{est}}),
\]

where \( F \) denote the set of the indices selected from the final correlation test.

C. Sparsity Update

In SUSP, the reconstruction on different values of \( s_{\text{est}} \) represents different stages. The value of \( s_{\text{est}} \) is updated adaptively. The stage changing is determined by the improvement of residual between two consecutive iterations. The SUSP selects atoms with maximum correlation in each stage of iteration, so the residual will be reduced gradually. If the current residual is greater or equal to the past residual,

\[
\|r\|_2 \geq \|r_p\|_2,
\]

it represents a residual floor has reached. In this case, there are two possibilities. The residual may be small enough and the iteration could terminate. Otherwise, when residual is not sufficiently small, the \( s_{\text{est}} \) needs to be updated to find better support set. Based on the principle, there are two possible situations: stop with termination criterion being satisfied, or forward update sparsity estimate based on the residual vector.

1) Termination Criterion: In the noiseless case, if the reconstruction of SUSP algorithm works perfectly, the residual vector approaches to a null vector. So we set the termination criterion

\[
\|r\|_2 = \|y - A\hat{x}\|_2 \leq \varepsilon,
\]

where \( \varepsilon \) is set to be an arbitrary small number. For noisy environment, the received signal would be \( y_r = y + e = Ax + e \), where \( e \) is an \( M \times 1 \) noisy vector with each entry to be Gaussian random variable with 0 mean and variance \( \sigma_n^2 \). If SUSP finds the correct estimated signal \( \hat{x} \), the residual will be

\[
\|r\|_2 = \|y_r - A\hat{x}\|_2 = \|e\|_2.
\]

For many applications where \( M \) is large enough,

\[
\|e\|_2 \cong \sqrt{M} \cdot \sigma_n.
\]

Hence the value of threshold \( \varepsilon \) should be set depending on the environment. In this paper, \( \varepsilon = 2\sqrt{M} \cdot \sigma_n \) is chosen for \( \hat{x} \) to be the reconstruction of \( x \).

2) Forward Estimation: The estimated sparsity \( s_{\text{est}} \) would not be equal to the true sparsity \( s \) with probability one. If the sparsity is under estimated, \( s_{\text{est}} < s \), and \( M \geq 2s \), the termination criterion would not be satisfied. In this condition, the SUSP algorithm increases the size of candidate list by

\[
s_{\text{est}} \leftarrow s_{\text{est}} + \left\lceil \frac{\|r\|_2^2}{n_A} \right\rceil,
\]

and the correlation test repeats based on the updated \( s_{\text{est}} \).

It is worthwhile to note that, even if the sparsity is pre-known, the recovery rate is not always 100% and residual value does not guaranteed to satisfy the termination criterion. By increasing the \( s_{\text{est}}(> s) \), the size of support set is increased and the recover rate can be further improved. The false atoms in the augmented support set can be pruned in the tail biting in the final step.

D. Tail Biting

In the sparsity update procedure, the \( s_{\text{est}} \) is likely to be greater than the true \( s \) to obtain a larger pool of support set. Though the residual is sufficiently small and the termination criterion is satisfied, the augmented support set need to be pruned by nulling the significantly smaller atoms. In SUSP, we post-process the resulting support set with tail biting to prune the false atoms. If the sparsity is pre-known, we can simply keep the largest \( s \) elements. When sparsity is not pre-known, the SUSP eliminates the false atoms by retain the minimum \( \hat{s} \) elements in support set \( F \) corresponding to the largest \( \hat{s} \) elements in \( \hat{x} \) while ensuring termination criterion to hold. The value of \( \hat{s} \) is determined by the infimum number of largest elements in \( \hat{x} \) that can keep the termination criterion satisfied. During the tail biting process, the termination criterion is sustained at all time to avoid pruning the true atoms. Therefore, We can recover the sparse signal and prune the false atoms from the final support list. The elements \( \hat{x} \) following tail biting rule in the final support can be expressed as

\[
\hat{x} = \{ \text{The infimum } \hat{s} \text{ largest elements in } A_{F_p}^\top y \text{ that termination criterion is true} \},
\]

where \( A_{F_p} \) is the matrix with columns selected from \( A \) based on the indices in support set \( F \).

IV. SIMULATIONS

The numerical performances for compressed spectrum sensing for cognitive radio are presented in this section. We first compare SUSP with existing sparse signal recovery algorithms for the sparsity pre-known problem. For the second part, we present the performance comparison between SUSP and SAMP for the sparsity unknown case. In both cases, the noiseless and the noisy scenarios are studied as well.

A. Performance of SUSP when Sparsity Pre-known

The performance between SUSP and existing algorithms is shown in Fig. 1. The number of measurement is fixed to \( M = 50 \) and the signal \( x \) with length \( N = 100 \). We investigate
Algorithm 1: Sparsity Update Subspace Pursuit Algorithm

**Input:** received vector $y$; measurement matrix $A$;

**Output:** estimated $x$ of the sparse signal estimated sparsity $\hat{s}$

**Initialization:**
- $r = y$, $r_p = y$, $F = \emptyset$
- $s_{est} = \left\lceil \frac{\|r\|_2^2}{n_A} \right\rceil$ or $s_{est} = s$ (Sparsity estimation)

repeat

- $T = \text{Large}(A^* r_p, s_{est})$ \{Correlation support set\}
- $C = F_p \cup T$ \{Solutions candidate list\}
- $F = \text{Large}(A^\dagger y, s_{est})$ \{Final test\}
- $r = y - A F A^\dagger y$
- if $\|r\|_2 \geq \|r_p\|_2$ then
  - if termination criterion true then quit iteration
  - else
    - $s_{est} \leftarrow s_{est} + \left\lceil \frac{\|r\|_2^2}{n_A} \right\rceil$ \{Sparsity update \}
    - $F_p = F$, $r_p = r$
  - end if
- else
  - $F_p = F$ and $r_p = r$
- end if

until termination criterion true

$x' = A^\dagger F y$

Tail Biting Rules

---

the recovery rate vs. the sparsity $s$ for a fixed $M$ and $N$. It is clear to see that the SUSP outperforms the SP and the OMP.

Fig. 2 shows the recovery rate over the number of measurement when the sparsity $s = 20$ and the length of sparse signal $N = 256$ is fixed. We see that SUSP yields the best probability for recovering sparse signal.

Fig. 3 shows the mean squared error in a noisy environment with $M = 50$ measurements and the signal $x$ with length of $N = 100$. Results show that the performance of SUSP is better than the SP and the OMP.

---

**B. Performance of SUSP when Sparsity-Unknown**

The recovery ability dealing with unknown sparsity problem is the main focus of this work. This section compares the simulation results of proposed SUSP with the SAMP. The number of measurements is fixed to $M = 50$, the signal length of $x$ is $N = 100$, and the tail biting threshold $\delta = 0.001$.

Fig. 4 shows the mean squared error in a noisy environment with measurements $M = 50$, the signal $x$ with length of $N = 100$ and the sparsity $s = 15$. The results show that the performance of SUSP is better than that of SAMP. The ability of estimating the sparsity of unknown sparse signal...
The SUSB results in better reconstruction performance than that of well-known SAMP method. In summary, the proposed SUSB algorithm can handle both sparsity pre-known and sparsity unknown cases with improved recovery performance.

ACKNOWLEDGEMENT

The authors would like to acknowledge the support of this work in parts by the Ministry of Science and Technology, Taiwan (MOST 103-2221-E-007-030) and joint collaboration with MediaTek Corp. (MOST 103-2622-E-002-034).

REFERENCES