

Compressed Sensing for Wireless Communications : Useful Tips and Tricks

Jun Won Choi *Member, IEEE*, Byonghyo Shim *Senior Member, IEEE*, Yacong Ding, *Student Member, IEEE*
Bhaskar Rao *Fellow, IEEE*, Dong In Kim, *Senior Member, IEEE*

Abstract—As a paradigm to recover the sparse signal from a small set of linear measurements, compressed sensing (CS) has stimulated a great deal of interest in recent years. In order to apply the CS techniques to wireless communication systems, there are a number of things to know and also several issues to be considered. However, it is not easy to grasp simple and easy answers to the issues raised while carrying out research on CS. The main purpose of this paper is to provide essential knowledge and useful tips and tricks that wireless communication researchers need to know when designing CS-based wireless systems. First, we present an overview of the CS technique, including basic setup, sparse recovery algorithm, and performance guarantee. Then, we describe three distinct subproblems of CS, viz., sparse estimation, support identification, and sparse detection, with various wireless communication applications. We also address main issues encountered in the design of CS-based wireless communication systems. These include potentials and limitations of CS techniques, useful tips that one should be aware of, subtle points that one should pay attention to, and some prior knowledge to achieve better performance. Our hope is that this article will be a useful guide for wireless communication researchers and even non-experts to get the gist of CS techniques.

Index Terms—Compressed sensing, sparse signal, underdetermined systems, wireless communication systems, ℓ_1 -norm, greedy algorithm, performance guarantee.

I. INTRODUCTION

Compressed sensing (CS) is an attractive paradigm to acquire, process, and recover the sparse signals [1]. This new paradigm is very competitive alternative to conventional information processing operations including sampling, sensing, compression, estimation, and detection. Traditional way to acquire and reconstruct analog signals from sampled signals is based on the celebrated Nyquist-Shannon's sampling theorem [3] which states that the sampling rate should be at least twice the bandwidth of an analog signal to restore it from the

discrete samples accurately. In case of a discrete signal regime, fundamental theorem of linear algebra states that the number of observations in a linear system should be at least equal to the length of the desired signal to ensure the accurate recovery of the desired signal. While these fundamental principles hold true always, they might be too stringent in a situation where signals of interest are sparse, meaning that the signals can be represented using a relatively small number of nonzero coefficients.

The CS paradigm provides a new perspective on the way we process the information. While approaches exploiting the sparsity of signal vector have been used for a long time in image processing and transform coding, this paradigm has attracted wide attention ever since the works of Donoho [1] and Candes, Romberg, and Tao [2]. At the heart of the CS lies the fact that a sparse signal vector can be recovered from the underdetermined linear system in a computationally efficient way. In other words, a small number of linear measurements (projections) of the signal contain enough information for its reconstruction. Main wisdom behind the CS is that essential knowledge in the large dimensional signals is just handful, and thus measurements with the size being proportional to the sparsity level of the input signal are enough to reconstruct the original signal. In fact, in many real-world applications, signals of interest are sparse or can be approximated as a sparse vector in a properly chosen basis. Sparsity of underlying signals simplifies the acquisition process, reduces memory requirement and computational complexity, and further enables to solve the problem which has been believed to be unsolvable.

In the last decade, CS techniques have spread rapidly in many disciplines such as medical imaging, machine learning, radar detection, seismology, computer science, statistics, and many others. Also, various wireless communication applications exploiting the sparsity of a target signal have been proposed in recent years. Notable examples, among many others, include channel estimation, interference cancellation, direction estimation, spectrum sensing, and symbol detection (see Fig. 1). These days, many tutorials, textbooks, and papers are available [4]–[6], but it might not be easy to grasp the essentials and useful tips tailored for wireless communication engineers. One reason is because the seminal works are highly theoretic and based on harmonic analysis, group theory, random matrix theory, and convex optimization so that it is not easy to figure out the crux from these works. Another reason is that the CS-based wireless communication works are mostly case studies so that one cannot grasp a general idea from these studies. Due to these reasons, CS remains somewhat esoteric

Copyright © 2015 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

J. Choi is with Dept. of Electrical and Biomedical Engineering, Hanyang University, Korea (e-mail: junwchoi@hanyang.ac.kr).

B. Shim is with Institute of New Media and Communications and School of Electrical and Computer Engineering, Seoul National University, Korea (e-mail: bshim@snu.ac.kr).

Y. Ding and B. Rao are with Dept. of Electrical and Computer Engineering, University of California at San Diego, CA, USA (e-mail: {yding, brao}@ucsd.edu)

D. Kim is with School of Information and Communication Engineering, Sungkyunkwan University, Korea (e-mail: dikim@ieee.org).

This work was sponsored by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) 2016R1A2B3015576 and 2014R1A5A1011478.

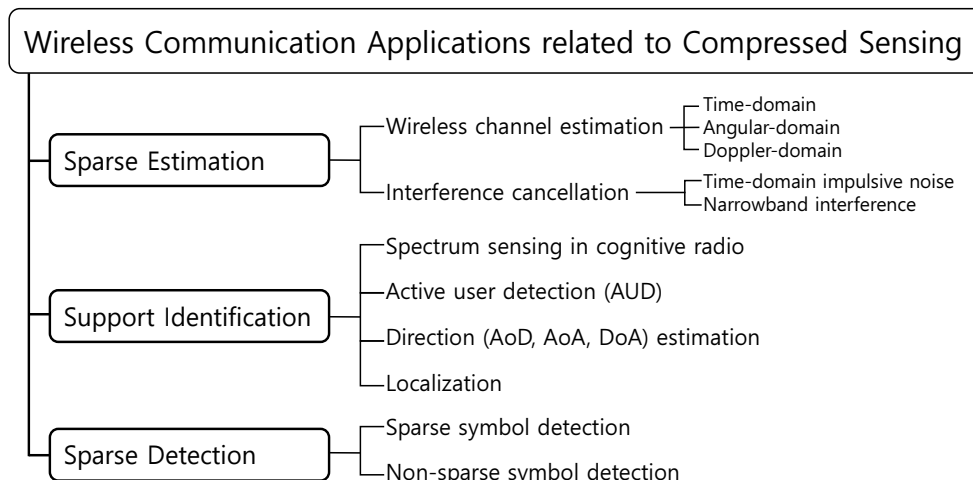


Fig. 1. Outline of subproblems of CS related to wireless communications and their examples.

and vague field for many wireless communication researchers and practitioners who want to catch the gist of CS and use it in their applications. Notwithstanding the foregoing, much of the fundamental principle and basic knowledge is simple, intuitive, and easy to understand.

The purpose of this paper is neither to describe the complicated mathematical expressions required for the characterization of the CS, nor to describe the details of state-of-the-art CS techniques and sparse recovery algorithms, but to bridge the gap between the wireless communications and CS technique by providing essentials and useful tips that communication engineers and researchers need to be aware of. With this purpose in mind, we organized this article as follows. In Section II, we provide an overview of the CS techniques. We review how to solve the systems with linear equations for both overdetermined and underdetermined systems and then address the scenario where the input vector is sparsely represented in the underdetermined setting. We also discuss the sparse signal recovery algorithm and performance guarantee of the CS technique under which accurate or stable recovery of the sparse signal is ensured. In Section III, we describe the basic wireless communication system model and then describe subproblems of CS related to wireless communications one by one. Depending on the sparse structure of the desired signal vector, CS problems can be divided into three subproblems: *sparse estimation*, *support identification*, and *sparse detection*. We explain the subproblem with the specific wireless communication applications. Developing the successful CS technique for the given wireless system requires good understanding on key design issues. These include properties of system matrix and input vector, algorithm selection/modification/development, system setup and performance requirements. In Section IV, we go over main issues in a way of answering to seven fundamental questions. In each issue, we provide useful tips, essential knowledge, and benefits and limitations so that readers can quickly understand the gist and thus take advantage of CS techniques. We conclude the paper in Section V by summarizing our contributions and discussing open issues. Our hope is that this paper would provide better

view, insight, and understanding of the potentials and limitations of CS techniques to wireless communication researchers.

In the course of this writing, we observe a large body of researches on CS, among which we briefly summarize some notable tutorial and survey results here. Short summary of CS is presented by Baraniuk [7]. Extended summary can be found in Candes and Wakin [5]. Foroucar and Rauhut provided a tutorial of CS with an emphasis on mathematical properties for performance guarantee [8] and similar approach can be found in [9]. Comprehensive treatment on various issues, such as sparse recovery algorithms, performance guarantee, and CS applications, can be found in the book of Eldar and Kutyniok [4]. Book of Han, Li, and Yin summarized the CS techniques for wireless network applications [6] and Hayashi, Nagahara, and Tanaka discussed the applications of CS to the wireless communication systems [10].

Our work is distinct from previous studies in the following aspects. First, we divide the wireless communication problems into three distinct subproblems (sparse estimation, sparse detection, support identification) and then explain the details of each problem, which could provide more systematic view and also help researchers easily classify the problem under investigation. Second, we focus more on physical layer issues with various state-of-the-art examples. While examples of previous studies are unclassified and a bit limited, we introduce plentiful examples including channel estimation and impulsive noise cancellation (Section III.B), spectrum sensing, active user detection, and direction estimation (Section III.C), and non-sparse detection (Section III.D) and then classify each into one of subproblems. From these examples, researchers and practitioners can figure out the similarity and disparity between the examples and their own problem and therefore realize how the CS technique can be applied to their wireless applications. We further present useful tips and tricks in a way of answering to seven important issues raised in the wireless system design for which there is no counterpart in previous studies. These include properties of system matrix and target vector, system and algorithm design when multiple measurements are available, selection of sparse recovery algorithm,

and how to take advantage of the constraints and additional hints (Section IV).

II. BASICS OF COMPRESSED SENSING

A. Solutions of Linear Systems

We begin with a linear system having m equations and n unknowns given by

$$\mathbf{y} = \mathbf{H}\mathbf{s} \quad (1)$$

where \mathbf{y} is the measurement vector, \mathbf{s} is the desired signal vector to be reconstructed, and $\mathbf{H} \in \mathcal{R}^{m \times n}$ is the system matrix (often referred to as the sensing matrix). In this case, the measurement vector \mathbf{y} can be expressed as a linear combination of the columns of \mathbf{H} , that is, $\mathbf{y} = \sum_i s_i \mathbf{h}_i$ (s_i and \mathbf{h}_i are the i -th entry of \mathbf{s} and i -th column of \mathbf{H} , respectively) so that \mathbf{y} lies in the subspace spanned by the columns of \mathbf{H} .

We first consider the scenario where the number of measurements is larger than or equal to the size of unknown vector ($m \geq n$). In this case, often referred to as overdetermined scenario, one can recover the desired vector \mathbf{s} using a simple algorithm (e.g., Gaussian elimination) as long as the system matrix is a full rank (i.e., $\text{rank}(\mathbf{H}) = \min\{m, n\}$). Even if this is not the case, one can find an approximate solution minimizing the error vector $\mathbf{e} = \mathbf{y} - \mathbf{H}\mathbf{s}$. The conventional approach to recover \mathbf{s} from the measurement vector \mathbf{y} is to find the vector \mathbf{s}^* minimizing the ℓ_2 -norm of the error vector, i.e.,

$$\mathbf{s}^* = \arg \min_{\mathbf{s}} \|\mathbf{e}\|_2. \quad (2)$$

Since $\|\mathbf{e}\|_2^2 = \mathbf{s}^T \mathbf{H}^T \mathbf{H} \mathbf{s} - 2\mathbf{y}^T \mathbf{H} \mathbf{s} + \mathbf{y}^T \mathbf{y}$, by setting the derivative of $\|\mathbf{e}\|_2^2$ with respect to \mathbf{s} to zero, we have $\frac{\partial}{\partial \mathbf{s}} \|\mathbf{e}\|_2^2 = 2\mathbf{H}^T \mathbf{H} \mathbf{s} - 2\mathbf{H}^T \mathbf{y} = 0$, and

$$\mathbf{s}^* = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{y}. \quad (3)$$

The obtained solution \mathbf{s}^* is often called the least squares (LS) solution and the operator $(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T$ is called the pseudo inverse and denoted as \mathbf{H}^\dagger . Note that $\mathbf{H}\mathbf{s}^*$ is closest to the measurement vector \mathbf{y} among all possible points in the range space of \mathbf{H} .

While finding the solution in an overdetermined scenario is straightforward and fairly accurate, the task to recover the input vector in an underdetermined scenario where the measurement size is smaller than the size of unknown vector ($m < n$) is challenging and problematic, since one cannot find out the unique solution in general. As a simple example, consider the example where $\mathbf{H} = [1 \ 1]$ and the original vector is $\mathbf{s} = [s_1 \ s_2]^T = [1 \ 1]^T$ (and hence $y = 2$). Since the system equation is $2 = s_1 + s_2$, one can easily observe that there are infinitely many possible solutions satisfying this. This is because for any vector $\mathbf{v} = [v_1 \ v_2]^T$ satisfying $0 = v_1 + v_2$ (e.g., $v_1 = -1$ and $v_2 = 1$), $\mathbf{s}' = \mathbf{s} + \mathbf{v}$ also satisfies $\mathbf{y} = \mathbf{H}\mathbf{s}'$. Indeed, there are infinitely many vectors in the null space $N(\mathbf{H}) = \{\mathbf{v} \mid \mathbf{H}\mathbf{v} = 0\}$ for the underdetermined scenario so that one cannot find out the unique solution satisfying (1). In this scenario, because $\mathbf{H}^T \mathbf{H}$ is not full rank and hence non-invertible, it is not possible to compute the LS solution in

(3). Alternative approach is to find a solution minimizing the ℓ_2 -norm of \mathbf{s} while satisfying the constraint $\mathbf{y} = \mathbf{H}\mathbf{s}$:

$$\mathbf{s}^* = \arg \min \|\mathbf{s}\|_2 \text{ s.t. } \mathbf{y} = \mathbf{H}\mathbf{s}. \quad (4)$$

Using the Lagrangian multiplier method, one can easily obtain¹

$$\mathbf{s}^* = \mathbf{H}^T (\mathbf{H}\mathbf{H}^T)^{-1} \mathbf{y}. \quad (5)$$

Since the solution \mathbf{s}^* is a vector satisfying the constraint ($\mathbf{y} = \mathbf{H}\mathbf{s}$) with the minimum ℓ_2 -norm, it is often referred to as the minimum norm solution. Since the system has more unknowns than measurements, the minimum norm solution in (5) cannot guarantee to recover the original input vector. This is well-known bad news. However, as we will discuss in the next subsection, sparsity of the input vector provides an important clue to recover the original input vector in this underdetermined setting.

B. Solutions of Underdetermined Systems for Sparse Input Vector

As mentioned, one cannot find out the unique solution of the underdetermined system since there exist infinitely many solutions. If one wish to narrow down the choice to convert ill-posed problem into well-posed one, additional hint (side information) is indispensable. In fact, the CS technique exploits the fact that the desired signal vector is sparse in finding the solution. A vector is called *sparse* if the number of nonzero entries is sufficiently smaller than the dimension of the vector. As a metric to check the sparsity, we use ℓ_0 -norm $\|\mathbf{s}\|_0$ of a vector \mathbf{s} , which is defined as²

$$\|\mathbf{s}\|_0 = \#\{i : s_i \neq 0\}.$$

For example, if $\mathbf{s} = [3 \ 0 \ 0 \ 0 \ 1 \ 0]$, then $\|\mathbf{s}\|_0 = 2$. In the simple example we discussed ($2 = s_1 + s_2$), if $\mathbf{s} = [s_1 \ s_2]$ is sparse, then at least s_1 or s_2 should be zero (i.e., $s_1 = 0$ or $s_2 = 0$). Interestingly, by invoking the sparsity constraint, the number of possible solutions is dramatically reduced from infinity to two (i.e., $(s_1, s_2) = (2, 0)$ or $(0, 2)$).

Since the ℓ_0 -norm counting the number of nonzero entries in a vector is a sparsity promoting function, the problem to find the sparsest input vector from the measurement vector is readily expressed as

$$\mathbf{s}^* = \arg \min \|\mathbf{s}\|_0 \text{ s.t. } \mathbf{y} = \mathbf{H}\mathbf{s}. \quad (6)$$

Since the ℓ_0 -norm counts the number of nonzero elements in \mathbf{s} , has no way but to rely on the combinatoric search to get the solution in (6). In other words, we need to investigate all possible subsystems $\mathbf{y} = \mathbf{H}_\Lambda \mathbf{s}_\Lambda$, where \mathbf{H}_Λ is the submatrix of \mathbf{H} that contains columns indexed by Λ .³ Initially, we investigate the solution with the sparsity one by

¹By setting derivative of the Lagrangian $L(\mathbf{s}, \lambda) = \|\mathbf{s}\|_2^2 + \lambda^T (\mathbf{y} - \mathbf{H}\mathbf{s})$ with respect to \mathbf{s} to zero, we obtain $\mathbf{s}^* = -\frac{1}{2} \mathbf{H}^T \lambda$. Using this together with the constraint $\mathbf{y} = \mathbf{H}\mathbf{s}$, we get $\lambda = -2(\mathbf{H}\mathbf{H}^T)^{-1} \mathbf{y}$ and $\mathbf{s}^* = \mathbf{H}^T (\mathbf{H}\mathbf{H}^T)^{-1} \mathbf{y}$.

²One can alternatively define as $\|\mathbf{s}\|_0 = \lim_{p \rightarrow 0} \|\mathbf{s}\|_p = \lim_{p \rightarrow 0} \sum_i |s_i|^p$

³For example, if $\Lambda = \{1, 3\}$, then $\mathbf{H}_\Lambda = [\mathbf{h}_1 \ \mathbf{h}_3]$.

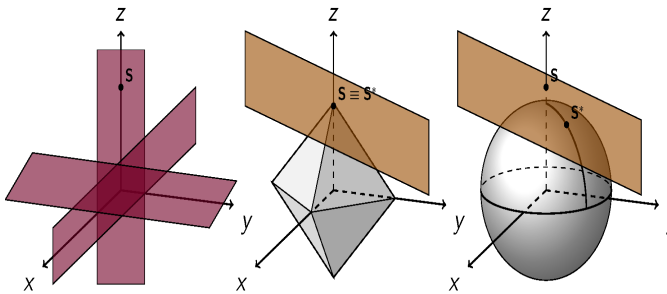


Fig. 2. Illustration of ℓ_0 , ℓ_1 , and ℓ_2 -norm minimization approach. If the sparsity of the original vector \mathbf{s} is one, then \mathbf{s} lies in the coordinate axes.

testing $\mathbf{y} = \mathbf{h}_i s_i$ for each i . If the solution is found (i.e. a scalar value s_i satisfying $\mathbf{y} = \mathbf{h}_i s_i$ is found), then the solution $\mathbf{s}^* = [0 \cdots 0 s_i 0 \cdots 0]$ is returned and the algorithm is finished. Otherwise, we investigate the solution with the sparsity two by checking if the measurement vector is constructed by a linear combination of two columns of \mathbf{H} . This step is repeated until the solution satisfying $\mathbf{y} = \mathbf{H}_\Lambda \mathbf{s}_\Lambda$ is found. Since the complexity of this combinatoric search increases exponentially in n , ℓ_0 -norm minimization approach is infeasible for most real-world applications.

Alternative approach suggested by Donoho [1] and Candes and Tao [5] is ℓ_1 -norm minimization approach given by

$$\mathbf{s}^* = \arg \min \|\mathbf{s}\|_1 \text{ s.t. } \mathbf{y} = \mathbf{H}\mathbf{s}. \quad (7)$$

While the ℓ_1 -norm minimization problem in (7) lies in the middle of (6) and (4), it can be cast into the convex optimization problem so that the solution of (7) can be obtained by the standard linear programming (LP) [1]. In Fig. 2, we illustrate ℓ_0 , ℓ_1 and ℓ_2 -norm minimization techniques. If the original vector is sparse (say the sparsity is one), then the desired solution can be found by the ℓ_0 -norm minimization since the points being searched are those in the coordinate axes (sparsity one). Since ℓ_1 -norm has a diamond shape (referred to as cross-polytope), one can observe from Fig. 2(b) that the solution of this approach corresponds to the vertex, not the face of the cross-polytope in most cases. Since the vertex of the diamond lies on the coordinate axes, it is highly likely that the ℓ_1 -norm minimization technique returns the desired sparse solution. In fact, it has been shown that under the mild condition the solution of ℓ_1 -norm minimization problem is equivalent to the original vector [5]. Whereas, the solution of the ℓ_2 -norm minimization corresponds to the point closest to the origin among all points \mathbf{s} satisfying $\mathbf{y} = \mathbf{H}\mathbf{s}$ so that the solution has no special reason to be placed at the coordinate axes. Thus, as depicted in Fig. 3, the ℓ_1 -norm minimization solution often equals the original sparse signal while the ℓ_2 -norm minimization solution does not guarantee this.

We also note that when the measurement vector \mathbf{y} is corrupted by the noise, one can modify the equality constraint as

$$\mathbf{s}^* = \arg \min \|\mathbf{s}\|_1 \text{ s.t. } \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2 \leq \epsilon \quad (8)$$

where ϵ is a (pre-determined) noise level of the system. This type of problem is often called basis pursuit de-noising

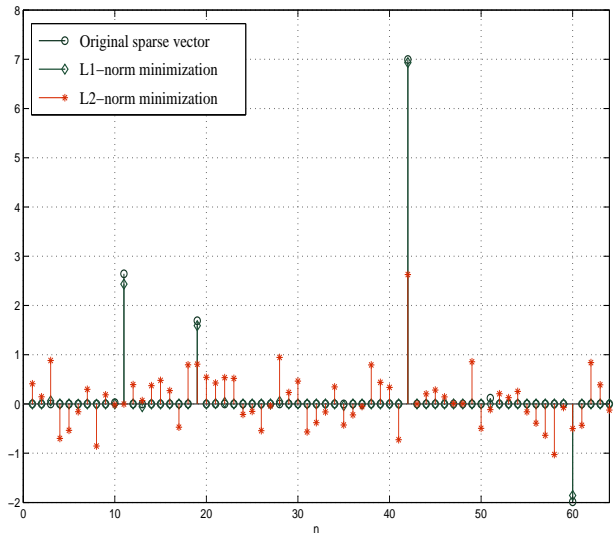


Fig. 3. Illustration of performance of ℓ_1 and ℓ_2 -norm minimization techniques. Entries of the system matrix $\mathbf{H} \in \mathbb{R}^{16 \times 64}$ are chosen from standard Gaussian and the sparsity of \mathbf{s} is set to 5.

(BPDN) [11], [12]. This problem has been well-studied subject in the convex optimization field and there are a number of approaches to solve the problem (e.g., interior-point method [13]). Nowadays, there are many optimization packages (e.g., CVX [14] or L1-magic [15]) so that one can save the programming effort by using these software tools.

C. Greedy algorithm

While the LP technique to solve ℓ_1 -norm minimization problem is effective in reconstructing the sparse vector, it requires substantial computational cost, in particular for large-scale applications. For example, a solver based on the interior point method has an associated computational complexity order of $O(m^2 n^3)$ [1]. For many real-time applications including wireless communication applications, computational cost and time complexity of ℓ_1 -norm minimization solver would be burdensome.

Over the years, numerous algorithms to recover the sparse signals have been proposed. Notable one among them is a greedy algorithm. By the greedy algorithm, we mean an algorithm to make a local optimal selection at each time with a hope to find the global optimum solution in the end. Perhaps the most popular greedy algorithm is the orthogonal matching pursuit (OMP) [16]. In the OMP algorithm, a column of the matrix \mathbf{H} is chosen one at a time using a greedy rule. Specifically, in each iteration, a column maximally correlated with the (modified) observation is chosen. Obviously, this is not necessarily optimal since the choice does not guarantee to pick the column associated with the nonzero element of \mathbf{s} . Let \mathbf{h}_{π_i} be the column chosen in the i -th iteration, then the (partial) estimate of \mathbf{s} is $\hat{\mathbf{s}}_i = \mathbf{H}_i^\dagger \mathbf{y}$ and the estimate of \mathbf{y} is $\hat{\mathbf{y}}_i = \mathbf{H}_i \hat{\mathbf{s}}_i = \mathbf{H}_i \mathbf{H}_i^\dagger \mathbf{y}$ where $\mathbf{H}_i = [\mathbf{h}_{\pi_1} \mathbf{h}_{\pi_2} \cdots \mathbf{h}_{\pi_i}]$. By subtracting $\hat{\mathbf{y}}_i$ from \mathbf{y} , we obtain the modified observation

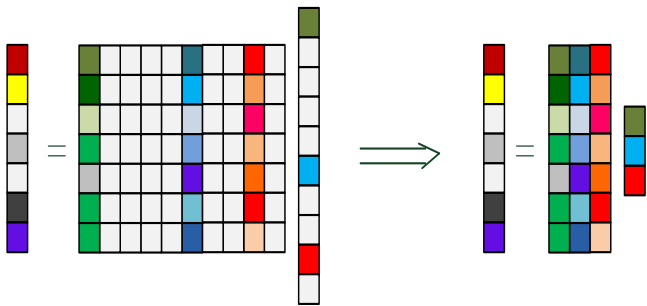


Fig. 4. Principle of the greedy algorithm. If the right columns are chosen, then we can convert an underdetermined system into an overdetermined system.

$\mathbf{r}_i = \mathbf{y} - \hat{\mathbf{y}}_i$, called the residual, used for the next iteration. By removing the contribution of $\hat{\mathbf{s}}_i$ from the observation vector \mathbf{y} , we can focus on the identification of the remaining nonzero elements in the next iteration.

One can observe that when the column selection is right, the OMP algorithm can reconstruct the original sparse vector accurately. This is because columns corresponding to the zero element in \mathbf{s} do not contribute to the observation vector \mathbf{y} so that they can be removed from the system model. As a result the underdetermined system can be converted into overdetermined system (see Fig. 4). As mentioned, LS solution for the overdetermined system generates an accurate estimate of the original sparse vector. Since the computational complexity of the greedy algorithm is typically much smaller than that of the LP techniques, the greedy algorithm has received much attention in recent years. Interestingly, even for the simple greedy algorithm like OMP algorithm, recent results show that the recovery performance is comparable to the LP technique while requiring much lower computational overhead [17]. We will discuss more on the sparse recovery algorithm in Section IV-D.

D. Performance Guarantee

In order to analyze the performance guarantee of the sparse recovery algorithm, various analysis tools have been suggested. For the sake of completeness, we briefly go over some of these in this subsection. First, a simple yet intuitive property is the spark of the matrix \mathbf{H} . Spark of a matrix \mathbf{H} is defined as the smallest number of columns of \mathbf{H} that are linearly dependent.⁴ From this definition, we see that a vector \mathbf{v} in a null space $N(\mathbf{H}) = \{\mathbf{v} \mid \mathbf{H}\mathbf{v} = \mathbf{0}\}$ should satisfy $\|\mathbf{v}\|_0 \geq \text{spark}(\mathbf{H})$ since a vector \mathbf{v} in the null space linearly combines columns in \mathbf{H} to make the zero vector, and at least $\text{spark}(\mathbf{H})$ columns are needed to do so. Following results provide the minimum level of spark over which uniqueness of the k -sparse solution is ensured.

Theorem 1 (Corollary 1 [5]): There is at most one k -sparse solution for a system of linear equations $\mathbf{y} = \mathbf{H}\mathbf{s}$ if and only if $\text{spark}(\mathbf{H}) > 2k$.

Proof See Appendix A

⁴Note that $\text{rank}(\mathbf{H})$ is the largest number of columns in \mathbf{H} that are linearly independent. In general, $\text{spark}(\mathbf{H}) \neq \text{rank}(\mathbf{H}) + 1$.

From the definition, it is clear that $1 \leq \text{spark}(\mathbf{H}) \leq n + 1$. If entries of \mathbf{H} are i.i.d. random, then no m columns in \mathbf{H} would be linearly dependent with high probability so that $\text{spark}(\mathbf{H}) = m + 1$. Using this together with Theorem 1, one can conclude that the uniqueness is guaranteed for every solution satisfying $k \leq \frac{m}{2}$.

It is worth mentioning that it is not easy to compute the spark of a matrix since it requires a combinatoric search over all possible subsets of columns in \mathbf{H} . Clearly, it is better to use a property that is easily computable. A tool that meets this purpose is the mutual coherence. The mutual coherence $\mu(\mathbf{H})$ is defined as the largest magnitude of normalized inner product between two distinct columns of \mathbf{H} :

$$\mu(\mathbf{H}) = \max_{i \neq j} \frac{|\langle \mathbf{h}_i, \mathbf{h}_j \rangle|}{\|\mathbf{h}_i\|_2 \|\mathbf{h}_j\|_2}. \quad (9)$$

One can see that $\mu(\mathbf{H}) > 0$ for a matrix having more columns than rows. In [18], it has been shown that for a full rank matrix, $\mu(\mathbf{H})$ satisfies

$$1 \geq \mu(\mathbf{H}) \geq \sqrt{\frac{n-m}{m(n-1)}}.$$

In particular, if $n \gg m$, we obtain an approximate lower bound as $\mu(\mathbf{H}) \geq \frac{1}{\sqrt{m}}$. It has been shown that $\mu(\mathbf{H})$ is related to $\text{spark}(\mathbf{H})$ via $\text{spark}(\mathbf{H}) \geq 1 + \frac{1}{\mu(\mathbf{H})}$ [19]. Using this together with Theorem 1, we get the following uniqueness condition.

Theorem 2 (Corollary 1 [5]): If $k < \frac{1}{2}(1 + \frac{1}{\mu(\mathbf{H})})$, then for each measurement vector, there exists at most one k -sparse signal \mathbf{s} satisfying $\mathbf{y} = \mathbf{H}\mathbf{s}$.

While the mutual coherence is relatively easy to compute, the bound obtained from this is too strict in general. These days, restricted isometry property (RIP), introduced by Candes and Tao [2], has been used popularly to establish the performance guarantee.

Definition 3: A system matrix \mathbf{H} is said to satisfy RIP if for all K -sparse vector \mathbf{s} , the following condition holds

$$(1 - \delta)\|\mathbf{s}\|_2^2 \leq \|\mathbf{H}\mathbf{s}\|_2^2 \leq (1 + \delta)\|\mathbf{s}\|_2^2. \quad (10)$$

In particular, the smallest δ , denoted as δ_k is referred to as a RIP constant. When we say the matrix satisfying the RIP, it implies that all possible selection of k columns behaves like and orthonormal system. That is, δ_k indicates how well the system matrix preserves the energy of the original signal. On one hand, if $\delta_k \approx 0$, the system matrix is close to orthonormal so that the reconstruction of \mathbf{s} would be guaranteed almost surely with a simple matching filtering operation (e.g., $\hat{\mathbf{s}} = \mathbf{H}^H \mathbf{y}$). On the other hand, if $\delta_k \approx 1$, it might be possible that $\|\mathbf{H}\mathbf{s}\|_2^2 \approx \mathbf{0}$ (i.e., \mathbf{s} is in the nullspace of \mathbf{H}) so that the measurements $\mathbf{y} = \mathbf{H}\mathbf{s}$ might not preserve any information on \mathbf{s} . In this case, obviously, the recovery of \mathbf{s} would be nearly impossible.

Note that RIP is useful to analyze performance when the measurements are contaminated by the noise [5], [20]–[22]. Additionally, by the help of random matrix theory, one can perform probabilistic analysis when the entries of the system matrix are i.i.d. random. Specifically, it has been shown that many random matrices (e.g., random Gaussian, Bernoulli, and

partial Fourier matrices) satisfy RIP with exponentially high probability, when the number of measurements scales linearly in the sparsity level [4]. As a well-known example, if $\delta_{2k} < \sqrt{2} - 1$, then the solution in (7) obeys

$$\|\mathbf{s}^* - \mathbf{s}\|_2 \leq C_0 \|\mathbf{s} - \mathbf{s}_k\|_1 / \sqrt{k} \quad (11)$$

$$\|\mathbf{s}^* - \mathbf{s}\|_1 \leq C_0 \|\mathbf{s} - \mathbf{s}_k\|_1 \quad (12)$$

for some constant C_0 , where \mathbf{s}_k is the vector \mathbf{s} with all but the largest k components set to 0. One can easily see that if \mathbf{s} is k -sparse, then $\mathbf{s} = \mathbf{s}_k$, and thus the recovery is exact. If \mathbf{s} is not k -sparse, then quality of recovery is limited by the difference of the true signal \mathbf{s} and its best k approximation \mathbf{s}_k . For a signal which is not exact sparse but can be well approximated by a k -sparse signal (i.e., $\|\mathbf{s} - \mathbf{s}_k\|_1$ is small), we can still achieve fairly good recovery performance.

While the performance guarantees obtained by RIP or other tools provide a simple characterization of system parameters (number of measurements, system matrix, algorithm) of the recovery algorithm, these results need to be taken with a grain of salt, in particular when designing the practical wireless systems. This is because the performance guarantee, expressed as a sufficient condition, might be too stringent and working in asymptotic sense in many cases. Also, some of them are, from the wireless communications perspective, based on impractical assumptions (e.g., Gaussianity of the system matrix, strict sparsity of input vector). Additionally, it is in general difficult to check whether the system setup satisfies the recovery condition or not.⁵

III. COMPRESSED SENSING FOR WIRELESS COMMUNICATIONS

A. System Model

In this section, we describe three distinct CS subproblems related to the wireless communications: sparse estimation, support identification, and sparse detection. We begin with the basic system model where the signal of interest is transmitted over linear channels with additive white Gaussian noise (AWGN). The input-output relationship in this model is

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{v}, \quad (13)$$

where \mathbf{y} is the vector of received signals, $\mathbf{H} \in \mathbb{C}^{m \times n}$ is the system matrix,⁶ \mathbf{s} is the desired signal vector we want to recover, and \mathbf{v} is the noise vector ($\mathbf{v} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I})$). In this article, we are primarily interested in the scenario where the desired vector \mathbf{s} is sparse, meaning that the portion of nonzero entries in \mathbf{s} is far smaller than its dimension. It is worth mentioning that even when the desired vector is non-sparse, one can either approximate it to a sparse vector or convert it to the sparse vector using a proper transform. For example, when the magnitude of nonzero elements is small, we can obtain an approximately sparse vector by ignoring negligible nonzero elements. For example, if $\mathbf{s} = [2 \ 0 \ 0 \ 0 \ 0 \ 3 \ 0.1 \ 0.05 \ 0.01 \ 0]^T$, then we can approximate it

⁵For example, one need to check $\binom{n}{2k}$ submatrices of \mathbf{H} to identify the RIP constant δ_{2k} .

⁶In the compressed sensing literatures, \mathbf{y} and \mathbf{H} are referred to as measurement vector and sensing matrix (or measurement matrix), respectively.

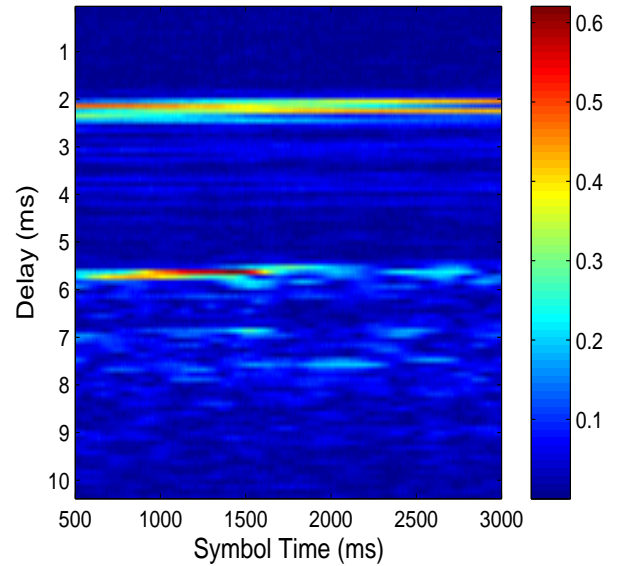


Fig. 5. A record of the channel impulse response (CIR) of underwater acoustic channels measured at the coast of Martha's vinyard, MA, USA. At the given time, CIR can be well approximated as a sparse vector.

to 2-sparse vector $\mathbf{s}' = [2 \ 0 \ 0 \ 0 \ 0 \ 3 \ 0 \ 0 \ 0 \ 0]^T$. In this case, the effective system model would be $\mathbf{y} = \mathbf{H}\mathbf{s}' + \mathbf{v}'$ where $\mathbf{v}' = \mathbf{H}_\nu \mathbf{s}_\nu + \mathbf{v}$ ($\mathbf{H}_\nu = [\mathbf{h}_7 \ \mathbf{h}_8 \ \mathbf{h}_9]$) and $\mathbf{s}_\nu = [0.1 \ 0.05 \ 0.01]^T$. Also, even in the case where the desired vector is not sparse, one might choose a basis $\{\psi_i\}$ to express the signal as a linear combination of basis. In the image/video processing society, for example, discrete Fourier transform (DFT), discrete cosine transform (DCT), and wavelet transform have been popularly used to perform this task. Using a properly chosen basis matrix $\Psi = [\psi_1 \ \dots \ \psi_n]$, the input vector can be expressed as $\mathbf{s} = \sum_{i=1}^n x_i \psi_i = \Psi \mathbf{x}$ and thus

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{v} = \mathbf{H}\Psi \mathbf{x} + \mathbf{v}, \quad (14)$$

where \mathbf{x} is a representation of \mathbf{s} in Ψ domain. By the proper choice of the basis, one can convert the original non-sparse vector \mathbf{s} into the sparse vector \mathbf{x} . Since this new representation does not change the system model, in the sequel, we will use a standard model in (13). Depending on the way the desired vector is constructed, the CS-related problem can be classified into three distinctive subproblems. In the following subsections, we will discuss these subproblems one by one.

B. Sparse Estimation

In wireless communications, traditional way to estimate the information vector \mathbf{s} from the model $\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{v}$ is the linear minimum mean square (LMMSE) estimation, i.e.,

$$\hat{\mathbf{s}} = \mathbf{R}_s \mathbf{H}^H (\mathbf{H} \mathbf{R}_s \mathbf{H}^H + \mathbf{R}_v \mathbf{I})^{-1} \mathbf{y}, \quad (15)$$

$$= (\mathbf{R}_s^{-1} + \mathbf{H}^H \mathbf{R}_v \mathbf{H})^{-1} \mathbf{H}^H \mathbf{R}_v^{-1} \mathbf{y} \quad (16)$$

where \mathbf{R}_s and \mathbf{R}_v are the covariance matrix of \mathbf{s} and \mathbf{v} , respectively. However, when the size of measurement vector \mathbf{y} is smaller than the size of the vector \mathbf{s} , the system matrix \mathbf{H} is rank-deficient and the quality of the LMMSE solution

degrades significantly. When the signal vector is sparse, the problem to recover \mathbf{s} from \mathbf{y} is classified into *sparse estimation* problem and the CS technique becomes efficient means to recover the target vector \mathbf{s} .

1) *Channel Estimation*: Channel estimation is a typical example of the sparse estimation. It has been shown from various measurement campaigns in wireless channels that the scattering effect of the environment is limited. In other words, propagation paths tend to be clustered within a small spread and there are only a few dominant scattering clusters. As a result, the communication channel can be sparsely represented in the delay-Doppler domain or the angular domain. In the delay-Doppler domain channel model, if the maximal delay and Doppler spreads are large and there are only few dominant paths, the channel can be readily modeled as a sparse vector [23]. Typical examples include the ultra-wideband channel [24]–[26], underwater acoustic channel [27]–[29], and even some cellular channels (e.g., extended vehicular-A (EVA) or extended typical urban (ETU) channel model in long term evolution (LTE) systems [30]). In the massive multi-input multi-output (MIMO) [31], [32] or millimeter wave [34], [35] systems where the transmitter and/or receiver are equipped with large antenna array, due to the limited number of scattering clusters and the increased spatial resolvability, the channel can be sparsely represented in the angular domain [36]–[38]. Also, recent experiments performed on millimeter wave channels [39], [40] reveal the limited scattering clusters in angular domain.

When the channel can be sparsely represented, CS-based channel estimation can offer significant performance gain over the conventional approaches. For example, we consider single carrier systems where the channel can be modeled as a m -tapped channel impulse response $\mathbf{h} = [h_0 \ h_1 \ \cdots \ h_{m-1}]^T$. The received signal $\mathbf{y} = [y_0 \ y_1 \ \cdots \ y_{m-1}]^T$ is expressed as a linear convolution of the channel impulse response \mathbf{h} and the pilot sequence $\mathbf{p} = [p_0 \ p_1 \ \cdots \ p_{m-n-2}]^T$, i.e. $y_t = \sum_{i=0}^{n-1} h_i * p_{t-i}$ for $0 \leq t \leq m-1$. By constructing the system matrix using the known pilot sequence, one can express the relationship between \mathbf{y} and \mathbf{h} using a matrix-vector form

$$\mathbf{y} = \mathbf{P}\mathbf{h} + \mathbf{v} \quad (17)$$

where \mathbf{P} is the Toeplitz matrix constructed from the pilot symbols $\mathbf{p} = [p_0 \ p_1 \ \cdots \ p_{m-n-2}]^T$. On the other hand, in the orthogonal frequency division multiplexing (OFDM) systems, we allocate the pilot symbols $\mathbf{p}_f = [p_0 \ p_1 \ \cdots \ p_m]^T$ in the frequency domain so that the linear convolution operation is converted to element-by-element product between \mathbf{p}_f and the frequency-domain channel response [41]–[44]

$$\mathbf{y} = \text{diag}(\mathbf{p}_f)\mathbf{\Phi}\mathbf{D}\mathbf{h} + \mathbf{v} \quad (18)$$

$$= \mathbf{P}\mathbf{h} + \mathbf{v} \quad (19)$$

where \mathbf{D} is the $n \times n$ DFT matrix which plays a role to convert the time-domain channel response \mathbf{h} to frequency domain response, and $\mathbf{\Phi}$ is the $m \times n$ row selection matrix determining the location of pilots in frequency domain. For both scenarios, the number of nonzero taps in \mathbf{h} is small due to the limited number of scattering clusters, and their positions are unknown.

Note that this problem has the same form as (13) and CS techniques can be used to recover \mathbf{h} from the measurements \mathbf{y} [45], [46].

As an example to exploit the sparsity in the angular domain, we consider a downlink massive MIMO system where the base-station is equipped with n antennas and the user has only single antenna. The user estimates the channel vector from the n base-station antennas to itself. For this purpose, the base-station transmits the training pilots from n antennas over m time slots. Under the narrowband block fading channel model, the pilot measurement acquired by the user over m time slots can be expressed as $\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{v}$, where $\mathbf{h} \in \mathbb{C}^{n \times 1}$ is the downlink channel vector, $\mathbf{A} \in \mathbb{C}^{n \times m}$ is the matrix containing the training pilots, and \mathbf{y} and \mathbf{v} are the received signal and noise vectors, respectively. When the number of antennas n is large and there are only limited number of scattering clusters in environment, we can use the angular domain channel representation [29], [36], [37], i.e., $\mathbf{h} = \mathbf{D}\mathbf{s}$, where \mathbf{D} is the $n \times n$ DFT matrix (whose columns correspond to the steering vectors corresponding to n equally spaced directions) and \mathbf{s} is the angular domain coefficients with only a few number of nonzeros. Then, the received downlink training signal can be written as $\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{v} = \mathbf{A}\mathbf{D}\mathbf{s} + \mathbf{v}$ which is the same form as (14). After obtaining the estimate $\hat{\mathbf{s}}$ using the CS technique, we can reconstruct the channel from $\hat{\mathbf{h}} = \mathbf{D}\hat{\mathbf{s}}$.

One clear benefit of the CS-based channel estimation is to achieve the reduction in the training overhead. In the above examples, the size m of the measurement vector indicates the amount of time and frequency resource needed to transmit pilot symbols. With conventional channel estimation schemes such as the LMMSE estimator, m should exceed the channel length n , imposing substantial training overhead in the system. However, by employing the CS techniques, it only requires m to be proportional to the number of nonzero elements in the sparse vector. In other words, when the channel can be sparsely represented, the channel can be acquired by the CS technique using much smaller amount of resources than what conventional methods require [44], [47]–[50].

In Fig. 6, we compare the channel estimation performance of the CS algorithm (OMP) and the conventional LMMSE technique in the massive MIMO systems. We set the number of antennas n to 64, the training overhead m to 16, and the number of multipaths k is 4. As shown in the figure, the CS algorithm outperform the conventional LMMSE channel estimator by a large margin for most of SNR regimes under test.

2) *Impulse Noise Cancellation in OFDM Systems*: While OFDM is well suited for frequency selective channel with Gaussian noise, when the unwanted impulsive noise is added, the performance would be degraded severely. In fact, since the impulse in the time domain corresponds to the constant in the frequency domain, very strong time domain impulses will give negative impact to most of frequency domain symbols. Since the span of impulse noise is short in time and thus can be considered as a sparse vector, we can exploit the CS technique to mitigate this noise [51], [52]. First, the discrete-time complex baseband equivalent channel model for the

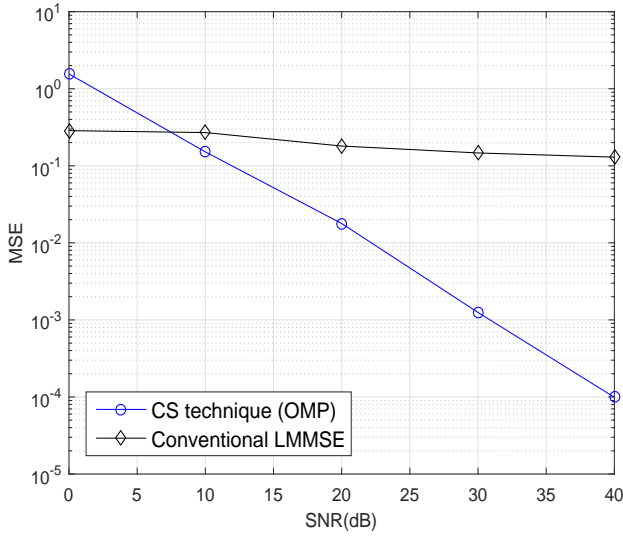


Fig. 6. Performance of the CS technique and the conventional LMMSE for angular domain channel estimation in massive MIMO systems.

OFDM signal is expressed as

$$\mathbf{y} = \mathbf{H}\mathbf{x}_t + \mathbf{n} \quad (20)$$

where \mathbf{y} and \mathbf{x}_t are the time-domain receive and transmit signal blocks (after the cyclic prefix removal), \mathbf{H} is the circulant matrix generated by the cyclic prefix, and \mathbf{n} is additive Gaussian noise vector. When the impulse noise \mathbf{e} is added, the received signal vector becomes

$$\mathbf{y} = \mathbf{H}\mathbf{x}_t + \mathbf{e} + \mathbf{n}. \quad (21)$$

Note that the circulant matrix \mathbf{H} can be eigen-decomposed by the DFT matrix \mathbf{F} , i.e., $\mathbf{H} = \mathbf{F}^H \mathbf{\Lambda} \mathbf{F}$ [53]. Also, the time-domain transmit signal in OFDM systems is expressed as $\mathbf{x}_t = \mathbf{F}^H \mathbf{x}_f$, where \mathbf{x}_f is the frequency-domain symbol vector. Let q be the number of subcarriers in which symbols are being transmitted, then the relationship between \mathbf{x}_f and the true (nonzero) symbol vector \mathbf{s} with dimension $q (\leq n)$ is

$$\mathbf{x}_f = \mathbf{\Pi} \mathbf{s}$$

where $\mathbf{\Pi}$ is $n \times q$ selection matrix containing only one element being one in each column and rest being zero. With these, (21) can be rewritten as

$$\begin{aligned} \mathbf{y} &= (\mathbf{F}^H \mathbf{\Lambda} \mathbf{F}) \mathbf{x}_t + \mathbf{e} + \mathbf{n} \\ &= (\mathbf{F}^H \mathbf{\Lambda} \mathbf{F}) (\mathbf{F}^H \mathbf{\Pi} \mathbf{s}) + \mathbf{e} + \mathbf{n} \\ &= \mathbf{F}^H \mathbf{\Lambda} \mathbf{\Pi} \mathbf{s} + \mathbf{e} + \mathbf{n}. \end{aligned} \quad (22)$$

Let \mathbf{y}' be the received vector after the DFT operation ($\mathbf{y}' = \mathbf{F}\mathbf{y}$). Then, we have

$$\mathbf{y}' = \mathbf{\Lambda} \mathbf{\Pi} \mathbf{s} + \mathbf{F} \mathbf{e} + \mathbf{n}' \quad (23)$$

where $\mathbf{n}' = \mathbf{F}\mathbf{n}$ is also Gaussian having the same statistic of \mathbf{n} . In removing the impulse noise, we use the subcarriers free of modulation symbols. By projecting \mathbf{y}' onto the space

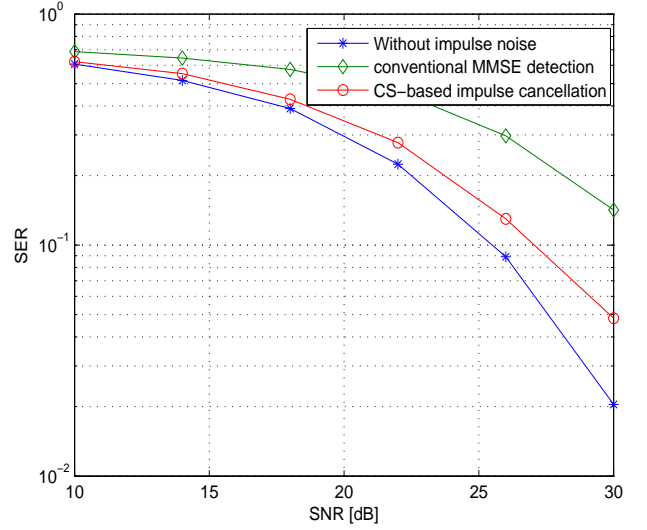


Fig. 7. Performance of OFDM systems with impulsive noise ($n = 64$, $q = 12$). Span of impulse is 2 among 64 samples, and the impulse noise power is set to 20 dB higher than the average sample power. Symbols are generated by the quadrature phase shift keying (QPSK) modulation and the OMP algorithm is used for the sparse impulse noise recovery.

where symbol is unassigned (i.e., orthogonal complement of the signal subspace), we obtain⁷

$$\mathbf{y}'' = \mathbf{P}\mathbf{y}' = \mathbf{P}\mathbf{F}\mathbf{e} + \mathbf{v} \quad (24)$$

where $\mathbf{v} = \mathbf{P}\mathbf{n}'$ is the sub-sampled noise vector. Note that \mathbf{y}'' is a projection of n -dimensional impulse noise vector onto a subspace of dimension $m (\ll n)$. In the CS perspective, \mathbf{y}'' and $\mathbf{P}\mathbf{F}$ are the observation vector and the sensing matrix, and hence the task is to estimate \mathbf{e} from \mathbf{y}'' . Once this task is finished, by subtracting $\mathbf{F}\hat{\mathbf{e}}$ from the received vector \mathbf{y}' , we obtain the modified received vector

$$\begin{aligned} \hat{\mathbf{y}}' &= \mathbf{y}' - \mathbf{F}\hat{\mathbf{e}} \\ &= \mathbf{\Lambda} \mathbf{\Pi} \mathbf{s} + \mathbf{F}(\mathbf{e} - \hat{\mathbf{e}}) + \mathbf{n}'. \end{aligned} \quad (25)$$

Due to the mitigation of the impulsive noise, as shown in Fig. 7, we observe improved detection performance.

C. Support Identification

Set of indices corresponding to nonzero elements in \mathbf{s} is called the support Ω_s of \mathbf{s} ⁸ and the problem to identify the support is called support identification problem. Support identification is important as a sub-step for the sparse estimation algorithm. It is also useful in its own right when an accurate estimation of nonzero values is unnecessary.

⁷As a simple example, if \mathbf{F} is 4×4 DFT matrix and the first and third subcarrier is being used $\mathbf{s} = \begin{bmatrix} s_1 \\ s_3 \end{bmatrix}$, then the selection matrix is $\mathbf{\Pi} =$

$\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$ and the projection operator is $\mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$. Since the 1st and 3rd elements of $\mathbf{\Lambda} \mathbf{\Pi} \mathbf{s}$ are nonzero and 2nd and 4th columns contain nonzero elements, we have $\mathbf{P} \mathbf{\Lambda} \mathbf{\Pi} \mathbf{s} = \mathbf{0}$.

⁸If $\mathbf{s} = [0 \ 0 \ 1 \ 0 \ 2]$, then $\Omega_s = \{3, 5\}$.

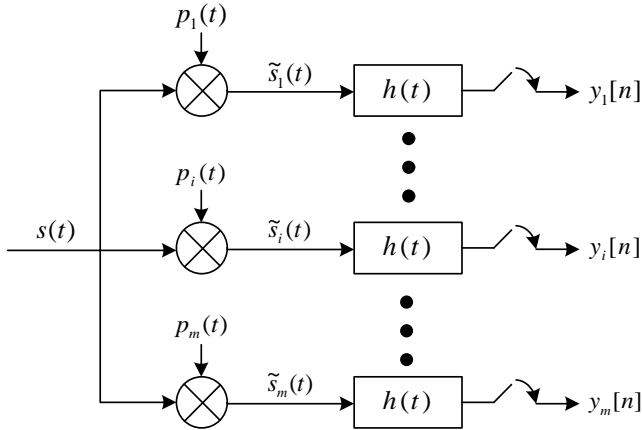


Fig. 8. Block diagram of modulated wideband converter for spectrum sensing.

1) *Spectrum Sensing*: As a means to improve the overall spectrum efficiency, cognitive radio (CR) has received much attention recently [54]. CR technique offers a new way of exploiting temporarily available spectrum. Specifically, when a primary user (license holder) does not use a spectrum, a secondary user may access it in such a way that they do not cause interference to primary users. Clearly, key to the success of the CR technology is the accurate sensing of the spectrum (whether the spectrum is empty or used by a primary user) so that secondary users can safely use the spectrum without hindering the operation of primary users. Future CR systems should have a capability to scan a wideband of frequencies, say in the order of a few GHz. In this case, design and implementation of high-speed analog to digital converter (ADC) become a challenge since the Nyquist rate might exceed the sampling rate of the state-of-the-art ADC devices, not to mention the huge power consumption. One can therefore think of an option of scanning each narrowband spectrum using conventional techniques. However, conventional approach is also undesirable since it takes too much time to process the whole spectrum (if done in sequential manner) or it is too expensive in terms of cost, power consumption, and implementation complexity (if done in parallel).

Recently, CS-based spectrum sensing technique has received much attention for its potential to alleviate the sampling rate issue of ADC and the cost issue of RF circuitry. In the CS perspective, the spectrum sensing problem can be translated into the problem to find the nonzero position of vector, which is often referred to as the *support identification* or *model selection* problem. One popular approach of the CS-based spectrum sensing problem is the modulated wideband converter (MWC) [55], [56]. In this scheme, we first multiply a pseudo random function $p(t)$ with period T_p to the time domain continuous signal $s(t)$. Since $p(t)$ is a periodic function, it can be represented as a Fourier series ($p(t) = \sum_k c_k e^{j2\pi k t/T_p}$). Let $s(f)$ be the frequency domain representation of $s(t)$, then the Fourier transform of the modulated signal $\tilde{s}(t) = p(t)s(t)$ is

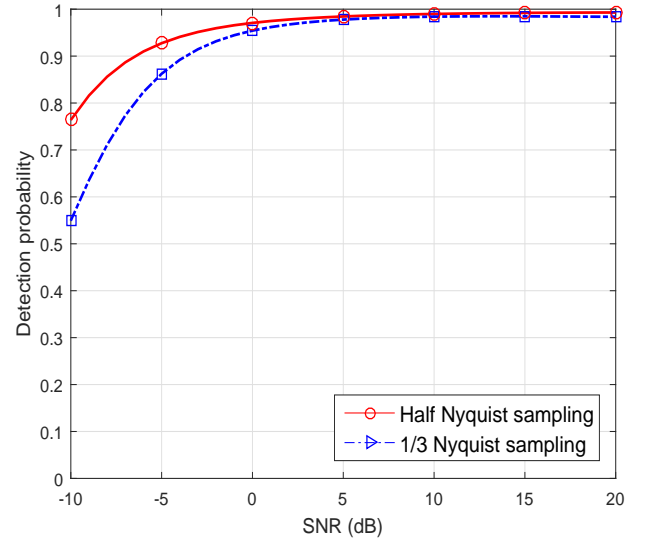


Fig. 9. Probability of correct support detection in the CS-based spectrum sensing. The signal occupies only 3 bands, each of width 50MHz, in a wide spectrum 5GHz.

expressed as

$$\tilde{s}(f) = \sum_{k=-\infty}^{\infty} c_k s(f - kf_p), \quad (26)$$

where $f_p = 1/T_p$. The low-pass filtered version $\tilde{s}'(f)$ will be expressed as $\tilde{s}'(f) = \sum_{k=-L}^L c_k s(f - kf_p)$. Denoting $y[n]$ as the discretized sequence of $\tilde{s}'(t)$ after the sampling (with rate T_s), we obtain the frequency domain relationship as⁹

$$y(e^{j2\pi f T_s}) = \sum_{k=-L}^L c_k s(f - kf_p). \quad (27)$$

When this operation is done in parallel for different modulating functions $p_i(t)$ ($i = 1, 2, \dots, m$), we have multiple measurements $y_i(e^{j2\pi f T_s})$. After stacking these, we obtain

$$\mathbf{y} = [y_1(e^{j2\pi f T_s}) \dots y_m(e^{j2\pi f T_s})]^T$$

and the corresponding matrix-vector form $\mathbf{y} = \mathbf{H}\mathbf{s}$ where $\mathbf{s} = [s(f - Lf_p) \dots s(f + Lf_p)]^T$ and \mathbf{H} is the sensing matrix relating \mathbf{y} and \mathbf{s} . Since large portion of the spectrum band is empty, \mathbf{s} can be readily modeled as a sparse vector, and our task is to find \mathbf{s} from $\mathbf{y} = \mathbf{H}\mathbf{s}$. It is worth mentioning that this problem is distinct from the sparse estimation problem since an accurate estimation of nonzero values is unnecessary. Recalling that the main purpose of the spectrum sensing is to identify the empty band, not the occupied one, it would not be a serious problem to slightly increase the false alarm probability (by false alarm we mean that the spectrum is empty but decided as an occupied one). However, special attention should be paid to avoid the misdetection event since the penalty would be severe when the occupied spectrum is falsely declared to be an empty one.

⁹If $u[n] = w(t)$ at $t = nT_s$, then $u(e^{j\Omega}) = w(f)$ where $\Omega = 2\pi f T_s$.

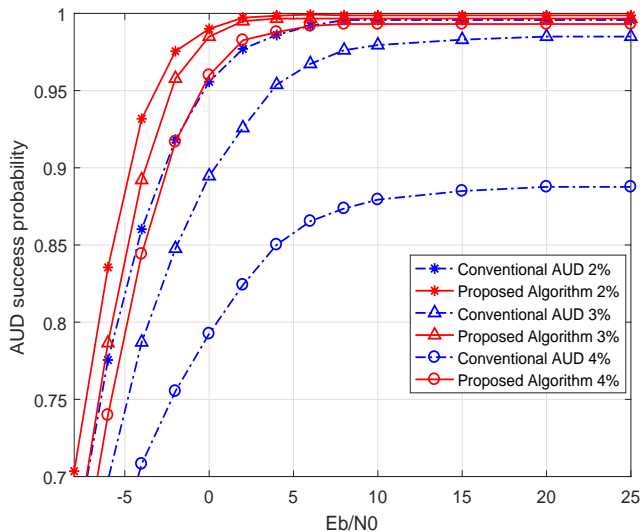


Fig. 10. Performance of AUD (FFT size is 512, length of CIR is 36, and among them three taps are dominant.)

2) *Detection of Active Devices for IoT Systems:* In recent years, internet of things (IoT), providing network connectivity of almost all things at all times, has received much attention for its plethora of applications such as healthcare, automatic metering, environmental monitoring (temperature, humidity, moisture, pressure), surveillance, automotive systems [57]. Recently, it has been shown that the CS can be useful for various IoT applications including energy saving [58], data acquisition and streaming [59], [60], positioning [61]–[64], monitoring [67]. See [68] for comprehensive survey. Common feature of the IoT networks is that the node density is much higher than the cellular network, yet the data rate is very low and not every device transmits information at a given time. Hence, the number of active devices in IoT environments is in general much smaller than the total number of devices. Since it is not easy to acquire the active user information from complicated message handshaking, it is of importance to easily identify what devices are active at the access point (AP). This problem is one of key issues in massive machine type communications (mMTC) in 5G wireless communications [69], [70].

Suppose there are n devices in the networks and only $k (\ll n)$ devices among them are trying to access to AP. In each device, the quasi-orthogonal signature (codeword) with length $m (< n)$ is assigned from the codebook $Q = \{\mathbf{q}_1, \dots, \mathbf{q}_n\}$. Since $m < n$, orthogonality among code sequences cannot be guaranteed (i.e., $\mathbf{q}_i \mathbf{q}_j^T \neq 0$ for $i \neq j$). When multiple devices send their signature sequences to AP, the first thing to be done at the AP is to identify which devices are transmitting information to the AP. Channel estimation, symbol detection, and decoding are performed after this *active user detection* (AUD). Under the flat fading channel assumption, which is true for narrowband IoT systems (e.g., eMTC and NB-IoT in 3GPP Rel. 13 [69]), the received vector at the AP is expressed

as

$$\begin{aligned} \mathbf{y} &= \sum_{i=1}^n h_i \mathbf{q}_i p_i + \mathbf{v} \\ &= [\mathbf{q}_1 \ \cdots \ \mathbf{q}_n] \begin{bmatrix} h_1 p_1 \\ \vdots \\ h_n p_n \end{bmatrix} + \mathbf{v} \\ &= \mathbf{H} \mathbf{s} + \mathbf{v} \end{aligned} \quad (28)$$

where p_i is the symbol transmitted from the i th device, h_i is the (scalar) channel from the i th device to the AP, $\mathbf{H} = [\mathbf{q}_1 \ \cdots \ \mathbf{q}_n]$ is the $m \times n$ matrix generated by the codeword vector \mathbf{q}_i ¹⁰, and \mathbf{v} is the $m \times 1$ noise vector. In this setup, the problem to identify which devices are active is translated into the problem to find out the support of \mathbf{s} . As depicted in Fig. 10, the CS-based AUD performs better than conventional approach. After the AUD, columns \mathbf{q}_i and symbol element s_i corresponding to inactive users can be removed from the system model, and as a result, we obtain the overdetermined system. Specifically, if the set $S = \{s_1, \dots, s_k\}$ is the support (index set of active users), then the system model to estimate the channel is

$$\begin{aligned} \mathbf{y} &= [p_{s_1} \mathbf{q}_{s_1} \ \cdots \ p_{s_k} \mathbf{q}_{s_k}] \begin{bmatrix} h_{s_1} \\ \vdots \\ h_{s_k} \end{bmatrix} + \mathbf{v} \\ &= \mathbf{T} \mathbf{h} + \mathbf{v}. \end{aligned} \quad (29)$$

As long as $k < m$, (29) is modeled as an overdetermined system so that one can estimate the channel vector \mathbf{h} accurately using the conventional LS or LMMSE technique.

3) *Localization for Wireless Sensor Network:* In wireless sensor networks, location information of the sensor nodes is needed to perform the location-aware resource allocation and scheduling. Location information is also an important ingredient for the location-based services. In a nutshell, the localization problem is to estimate the position of the target nodes using the received signals at multiple APs [72]. For example, we consider the wireless sensor network equipped with multiple APs at different places. These APs receive the signals from multiple targets and then send the collected signals to the server where the localization task is performed (see Fig. 11). Assume that there are k targets and m APs measure the received signal strength (RSS). The location of the targets is represented in the two-dimensional location grid of size n . The conventional localization algorithm calculates the average RSS for all candidates over the grid, and searches for the one closest to the measurements. Due to a large number of elements in the location grid, this approach will incur high computational complexity. Since the number of target nodes is much smaller than the number of elements in the grid, the location information of the targets is sparse, and thus the CS technique can be used to perform the efficient localization [61]–[63]. Let the average RSS from the j th grid element to the i th AP be $p_{i,j}$. The average RSS can be obtained from offline measurement at certain reference

¹⁰There are various approaches to generate the codebook for machine type devices. See, e.g., sparse coding multiple access (SCMA) [71].

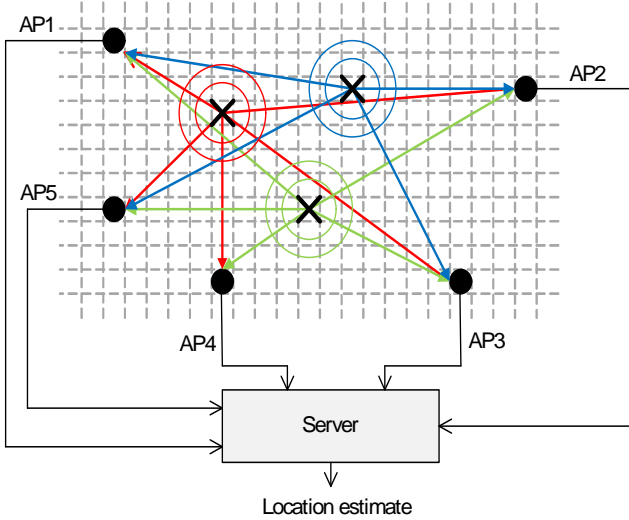


Fig. 11. Illustration of localization in wireless sensor network.

points (i.e., fingerprinting method [65], [66]) or calculated via appropriate path loss models. In the fingerprinting method, we obtain the average RSS at non-reference points via two dimensional interpolation. The measurement received by the m APs $\mathbf{y} = [y_1, \dots, y_m]^T$ is expressed as

$$\mathbf{y} = \mathbf{P}\mathbf{s} + \mathbf{v}, \quad (30)$$

where \mathbf{P} is the $m \times n$ matrix and the i th element of \mathbf{s} has zero value if the target is not located at the i th grid element, and \mathbf{v} is the perturbation vector capturing the noise and modeling error. The support of \mathbf{s} represents the location of the target nodes and the CS technique can be employed to identify the support.

4) *Direction Estimation*: Yet another important application of the support identification problem is the estimation of the angle of arrival (AoA) and angle of departure (AoD) in wireless communications. In the millimeter wave (mmWave) communications, the carrier frequency increases up to tens or hundreds of GHz so that the transmit signal power decays rapidly with distance and wireless channels exhibit a few strong multipaths components caused by the small number of dominant scatterers. In fact, signal components departing and arriving from particular angles are very few compared to the total number of angular bins [33]. When the estimates of AoA and AoD are available, beamforming with high directivity is desirable to overcome the path loss of mmWave wireless channels. The sparsity of the channel in the angular domain can be exploited for the effective estimation of the AoA and AoD [34], [35], [73]. Consider a mmWave system where the transmitter and the receiver are equipped with N_t and N_r antennas, respectively. When employing the uniform linear array antennas, MIMO channels in the angular domain is expressed as

$$\mathbf{H} = \mathbf{A}_r \Phi_a \mathbf{A}_t^H, \quad (31)$$

where $\mathbf{A}_r = [\mathbf{a}_r(\phi_1), \dots, \mathbf{a}_r(\phi_{L_r})] \in \mathcal{C}^{N_r \times L_r}$, $\mathbf{A}_t = [\mathbf{a}_t(\phi_1), \dots, \mathbf{a}_t(\phi_{L_t})] \in \mathcal{C}^{N_t \times L_t}$, L_r and L_t are the number of

total angular bins for the receiver and the transmitter, respectively, $\mathbf{a}_r(\phi_i)$ and $\mathbf{a}_t(\phi_i)$ are the steering vectors corresponding to the i -th angular bin for AoA and AoD, respectively, and Φ_a is the $L_r \times L_t$ path-gain matrix whose (i, j) th entry contains the path gain from j th angular bin for the transmitter to i th angular bin for the receiver. Note that due to the sparsity of the channel in the angular domain, only a few elements of Φ_a (i.e., k) are nonzero. In order to estimate AoA and AoD, the base-station transmits the known symbols at the designated directions over T time slots, where the transmit beamforming vector $\mathbf{f}_i \in \mathcal{C}^{N_t \times 1}$ is used for the i th beam transmission. Then, the received vector $\mathbf{z}_i \in \mathcal{C}^{N_r \times 1}$ corresponding to the i th beam transmission can be expressed as

$$\mathbf{z}_i = \mathbf{H}\mathbf{f}_i x_i + \mathbf{n}_i, \quad (32)$$

where \mathbf{n}_i is the noise vector and x_i is the known symbol. When the receiver applies the combining matrix $\mathbf{W} \in \mathcal{C}^{N_r \times Q}$, we can obtain the measurement vector $\mathbf{y}_i \in \mathcal{C}^{Q \times 1}$ as

$$\mathbf{y}_i = \mathbf{W}^H \mathbf{H}\mathbf{f}_i x_i + \mathbf{W}^H \mathbf{n}_i, \quad (33)$$

$$= \mathbf{W}^H \mathbf{H}\mathbf{f}_i x_i + \mathbf{v}_i, \quad (34)$$

If we collect the received signals over T beam transmissions, i.e., $\mathbf{y} = [\mathbf{y}_1^T, \dots, \mathbf{y}_T^T]^T$ and let $x_i = 1$, then we have [35]

$$\mathbf{y} = \text{vec}(\mathbf{W}^H \mathbf{H}\mathbf{F}) + \mathbf{v} \quad (35)$$

$$= (\mathbf{F}^T \otimes \mathbf{W}^H) \text{vec}(\mathbf{A}_r \Phi_a \mathbf{A}_t^H) + \mathbf{v} \quad (36)$$

$$= ((\mathbf{F}^T (\mathbf{A}_t)^*) \otimes (\mathbf{W}^H \mathbf{A}_r)) \mathbf{s} + \mathbf{v}, \quad (37)$$

$$= \mathbf{P}\mathbf{s} + \mathbf{v} \quad (38)$$

where $\text{vec}(\cdot)$ and $(\cdot)^*$ are the vectorization and the conjugation operations, respectively, \otimes is the Kronecker product, $\mathbf{v} = [\mathbf{v}_1^T, \dots, \mathbf{v}_T^T]^T$, $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_T]$, and $\mathbf{s} = \text{vec}(\Phi_a)$. Note that the indices of the nonzero elements in \mathbf{s} correspond to the AoA and AoD information. The dimension of the vector \mathbf{s} (i.e., $n = L_r \times L_t$) needs to be large to estimate the AoA and AoD with high resolution. On the other hand, the size of the measurement vector $n (= T \times Q)$ cannot be as large as the value of m due to the resource constraints. Since the number of nonzero elements is small, \mathbf{s} is modeled by the sparse vector and the CS technique is useful to find the support of \mathbf{s} .

Fig. 12 shows the performance of the CS technique and the conventional LMMSE method. We assume that the transmitter and receiver use the beamforming and combining vectors steered to equally spaced directions with $T = Q = 16$. We consider the mmWave channels with three multi-path components (i.e., $k = 3$) and both the transmitter and receiver have 16 antennas. The resolutions for AoA and AoD are set to $L_r = 32$ and $L_t = 32$, respectively. In this setup, the size of the system matrix \mathbf{P} is 256×1024 . In the LMMSE-based approach, we obtain the support by picking the largest k absolute values of the LMMSE estimate of \mathbf{s} . While the performance of this approach is not appealing, the CS technique exhibits fairly good performance due to the exploitation of the sparsity structure of the angular-domain channel.

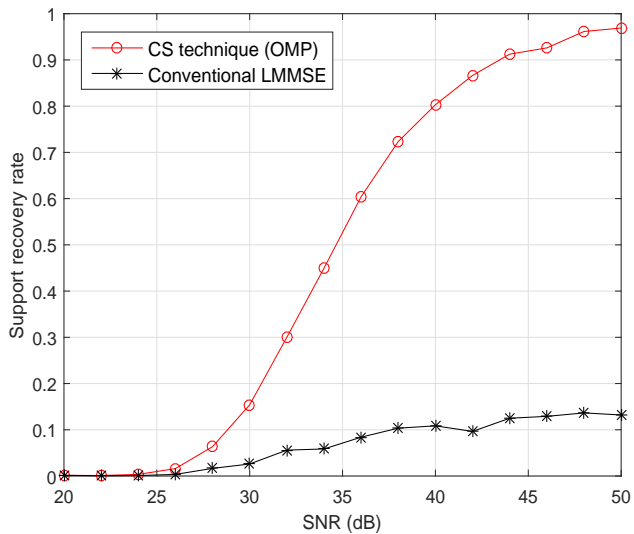


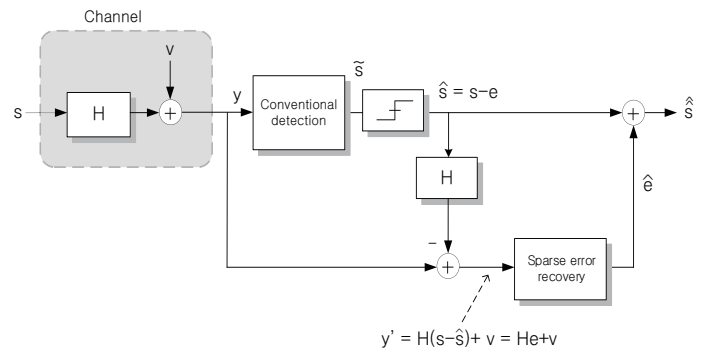
Fig. 12. Recovery performance of the CS technique in comparison with the conventional LMMSE method for mmWave communications.

D. Sparse Detection

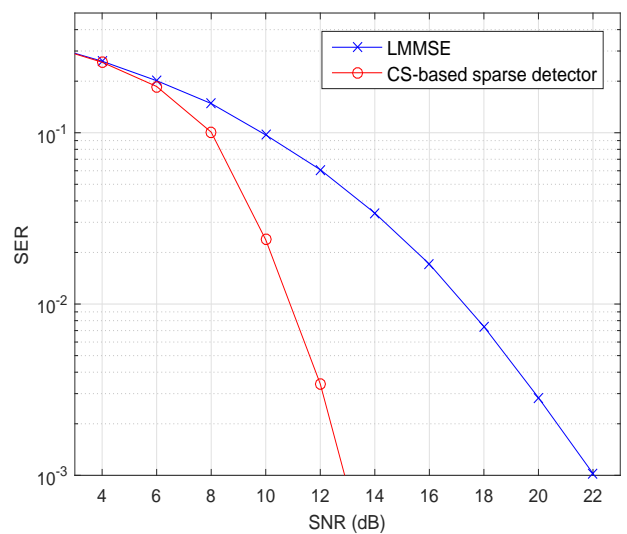
Even in the case where the transmit symbol vector is non-sparse, we can still use CS techniques to improve the performance of the symbol detection. There are a number of applications where the transmit vector cannot be modeled as a sparse signal. Even in this case, by the deliberate combination of conventional linear detection and sparse recovery algorithm, one can improve the detection performance [74]. In this scheme, conventional linear detection such as LMMSE is performed initially to generate a rough estimate (denoted by \hat{s} in Fig. 13) of the transmit symbol vector. Since the quality of detected (sliced) symbol vector \hat{s} is generally acceptable in the operating regime of the target systems, the error vector $e = s - \hat{s}$ after the detection would be readily modeled as a sparse signal. Now, by a simple transform of this error vector, one can obtain the new measurement vector y' whose input is the sparse error vector e . This task is accomplished by the retransmission of the detected symbol \hat{s} followed by the subtraction (see Fig. 13 (a)). As a result, newly obtained received vector y' is expressed as $y' = y - H\hat{s} = He + v$, where \hat{s} is the estimate of s obtained by the conventional detector. In estimating the error vector e from y' , a sparse recovery algorithm can be employed. By adding the estimate \hat{e} of the error vector to the sliced symbol vector \hat{s} , more reliable estimate of the transmit vector $\hat{\hat{s}} = \hat{s} + \hat{e} = s + (\hat{e} - e)$ can be obtained. In Fig. 13 (b), we observe that the CS-based sparse detection outperforms the conventional approach by a large margin.

IV. ISSUES TO BE CONSIDERED WHEN APPLYING CS TECHNIQUES TO WIRELESS COMMUNICATION SYSTEMS

As more things should be considered in the design of wireless communication systems, such as wireless channel environments, system configurations (bandwidth, power, number of antennas), and design requirements (computational



(a) Block diagram



(b) SER performance

Fig. 13. Sparse detection using the CS technique: (a) block diagram of CS-based sparse error detection (SED) technique and (b) symbol error rate (SER) performance.

complexity, peak data rate, latency), the solution becomes more challenging, ambitious, and complicated. As a result, applying the CS techniques to wireless applications is not any more copy-and-paste type task and one should have good knowledge on fundamental issues. Some of the questions that wireless researchers can come across when they design a CS-based technique are listed as follows:

- Is sparsity important for applying CS technique? If yes, then what is the desired sparsity?
- How can we convert non-sparse vector into sparse one? Should we know the sparsity in a priori?
- What is the desired property of the system (sensing) matrix?
- What kind of recovery algorithms are there and what are pros and cons of these?
- What should we do if more than one observations are available?
- Can we do better if the input vector consists of finite alphabet symbols?

In this section, we address these issues in a way of answering to each question. In each issue, we provide useful tips and tricks for the successful development of CS techniques for wireless communication systems.

A. Is Sparsity Important?

If you have an application that you think CS-based technique might be useful, then the first thing to check is whether the signal vector to be recovered is sparse or not. Many natural signals, such as image, sound, or seismic data are in themselves sparse or can be sparsely represented in a properly chosen basis. Even though the signal is not strictly sparse, often it can be well approximated as a sparse signal. For example, most of wireless channels exhibit power-law decaying behavior due to the physical phenomena of waves (e.g., reflection, diffraction, and scattering) so that the received signal is expressed as a superposition of multiple attenuated and delayed copies of the original signal. Since a few of delayed copies contain most of the energy, a vector representing the channel impulse response can be readily modeled as a sparse vector. Obviously, stronger sparsity, i.e., lower k leads to better recovery performance (see Fig. 14). Regarding the sparsity, an important question that one might ask for is what level of sparsity is enough to apply the CS techniques? Put it alternatively, what is the desired dimension of the observation vector when the sparsity k is given? Although there is no clean-cut boundary on the measurement size under which CS-based techniques do not work properly,¹¹ it has been shown that one can recover the original signals using $m = O(k \log(\frac{n}{k}))$ measurements via many of state-of-the-art sparse recovery algorithms. Since the logarithmic term can be well approximated to a constant, one can set $m = \epsilon k$ as a starting point (e.g., $\epsilon = 4$ by four-to-one practical rule [5]). This essentially implies that measurement size is a linear function of k and unrelated to n . Note, however, that if the measurement size is too small and comparable to the sparsity (e.g., $m < 2k$ in Fig. 14), performance of the sparse recovery algorithms might not be appealing.

In summary, if the desired vector is sparse, then it is natural to consider the CS technique. As a rule of thumb, if the sparsity k satisfies $m = 4k$, one can safely try the CS technique. We will discuss in the next subsection that even though the target vector is non-sparse, we can still use the CS technique by finding the basis over which the target vector can be sparsely represented.

B. Predefined Basis or Learned Dictionary?

As discussed, to use CS techniques in wireless communication applications, we should ensure that the target signal has a sparse representation. Traditional CS algorithm is performed when the signal can be sparsely represented in an orthonormal basis, and many robust recovery theories are based on this assumption [5]. Although such assumption is valid in many applications, there are still plenty of scenarios where the

¹¹In fact, this is connected to many parameters such as dimension of vector and quality of system matrix.

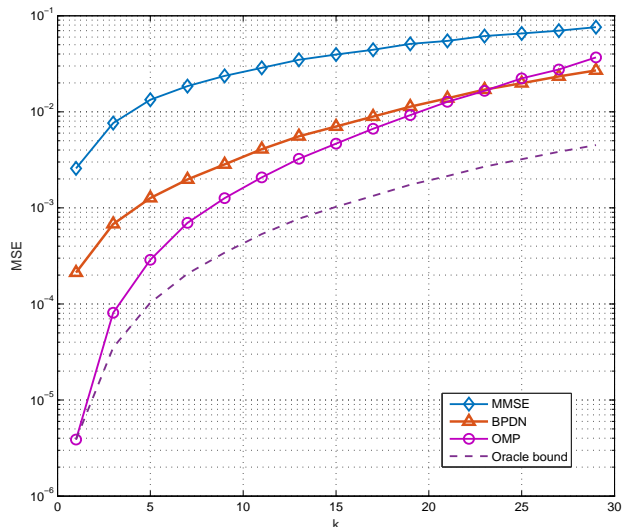


Fig. 14. Recovery performance as a function of sparsity. Elements of the sensing matrix is generated by the Gaussian random variable. We can clearly see that the CS techniques outperform the conventional MMSE technique by a large margin ($m = 100$, $n = 256$, SNR= 20 dB).

target signal may not be sparsely represented in an orthonormal basis, but in an overcomplete dictionary [75]. Overcomplete dictionary refers to a dictionary having more columns than rows. Since such dictionary is usually unknown beforehand, it should be learned from a set of training data. This task, known as *dictionary learning* [76], [77], is to learn an overcomplete dictionary $\mathbf{D} \in \mathbb{C}^{n \times m}$ ($n < m$) from a set of signals \mathbf{x}_i such that \mathbf{x}_i can be approximated as $\mathbf{x}_i \approx \mathbf{D}\mathbf{s}_i$, where $\mathbf{s}_i \in \mathbb{C}^{m \times 1}$ and $\|\mathbf{s}_i\|_0 \ll m$. Specifically, by using a training set \mathbf{X} which contains L realizations of \mathbf{x} , i.e., $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L]$, we solve the following optimization problem

$$\min_{\mathbf{D}, \mathbf{s}_1, \dots, \mathbf{s}_L} \lambda \|\mathbf{X} - \mathbf{D}\mathbf{S}\|_F^2 + \sum_{i=1}^L \|\mathbf{s}_i\|_0 \quad (39)$$

where $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_L]$ is the matrix formed from all sparse coefficients satisfying $\mathbf{x}_k \approx \mathbf{D}\mathbf{s}_k$. Note that λ is the parameter that trades off the data fitting error $\|\mathbf{X} - \mathbf{D}\mathbf{S}\|_F^2$ and the sparsity of the representation $\sum_{i=1}^L \|\mathbf{s}_i\|_0$. As a result of this, we can express the system model $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$ into $\mathbf{y} \approx \mathbf{A}\mathbf{D}\mathbf{s} + \mathbf{n} = \mathbf{H}\mathbf{s} + \mathbf{n}$ where $\mathbf{H} = \mathbf{A}\mathbf{D}$. After obtaining $\hat{\mathbf{s}}$ from the CS technique, we generate the estimated signal $\hat{\mathbf{x}} = \mathbf{D}\hat{\mathbf{s}}$.

As an example to show the benefit of the dictionary learning, we consider the downlink channel estimation of the massive MIMO systems. Consider the pilot-aided channel estimation where the base-station transmits pilots symbols $\mathbf{A} \in \mathbb{C}^{T \times n}$ during the training period T . In the multiple-input-single-output (MISO) scenario where the base-station has n antennas and the mobile user has a single antenna, the received vector is $\mathbf{y} = \mathbf{A}\mathbf{h} + \mathbf{n}$ where \mathbf{h} is the channel vector. Traditional channel estimation schemes such as the LS or LMMSE estimation require more than or at least equal to n measurements to reliably estimate the channel. In the massive MIMO regime

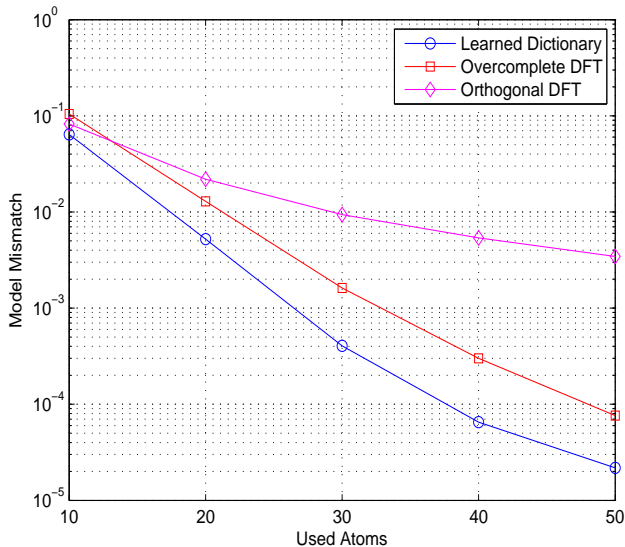


Fig. 15. MSE performance of overcomplete learned dictionary, overcomplete DFT dictionary and orthogonal DFT basis. The channel \mathbf{h}_i is generated using 3GPP spatial channel model (SCM) [79], each \mathbf{h}_i consists of 6 scatter clusters (3 far scatter and 3 local scatter).

where n is in the order of hundred or more, this approach might not be feasible since it consumes too much downlink resources. From our discussion, if \mathbf{h} is sparse in some basis or dictionary \mathbf{D} (i.e., $\mathbf{h} \approx \mathbf{D}\mathbf{s}$, $\|\mathbf{s}\|_0 \ll n$), then with the knowledge of \mathbf{A} and \mathbf{D} , \mathbf{s} can be recovered from $\mathbf{y} = \mathbf{A}\mathbf{D}\mathbf{s} + \mathbf{n}$ via the CS technique, and subsequently the channel \mathbf{h} is estimated as $\hat{\mathbf{h}} = \mathbf{D}\hat{\mathbf{s}}$. Since the training period proportional to the sparsity of \mathbf{s} ($T \propto \|\mathbf{s}\|_0$) is enough, channel estimation using the CS technique is effective in reducing the pilot training overhead.

One can easily see that key issue in this process is to find \mathbf{D} such that \mathbf{h} can be sparsely represented, that is, to express \mathbf{h} as $\mathbf{h} \approx \mathbf{D}\mathbf{s}$ where \mathbf{s} is the sparse vector. A commonly used basis is the DFT basis which is derived from a uniform linear array deployed at the base-station [29], [34]. However, the sparsity assumption under the orthogonal DFT basis is valid only when the scatters in the environment are extremely limited (e.g., point scatter) and the number of antennas at the base-station goes to infinity [36], which is not applicable in many cases. Fig. 15 depicts the model mismatch error $\sum_{i=1}^L \|\mathbf{h}_i - \mathbf{D}\mathbf{s}_i\|_2^2 / L$ as a function of the number of atoms in \mathbf{D} being used. For each k , we set the constraint $\|\mathbf{s}_i\|_0 \leq k$ for all i and then compare three types of \mathbf{D} : orthogonal DFT basis, overcompleted DFT dictionary, and overcompleted learned dictionary. We observe that an approach using overcomplete dictionary achieves much smaller model mismatch error than the orthogonal basis, while the learned dictionary is even better than overcomplete DFT dictionary (see also [78]). In this figure, we also observe that even with relatively weak sparsity (e.g., $\|\mathbf{s}_i\|_0 = 30$), the model mismatch error of the learned dictionary is pretty small $\|\mathbf{h}_i - \mathbf{D}\mathbf{s}_i\|_2^2 < 10^{-3}$. This example clearly demonstrates that the essential degree of freedom of the channel is much smaller than the actual dimension of the channel ($k = 30 \ll n = 100$).

The main takeaway in this subsection is that even though the vector is non-sparse, one can sparsify the desired vector using a technique such as dictionary learning. Using simple dictionary like DFT matrix would be handy but finding an elaborate dictionary is a bit challenging task and also interesting open research problem. In general, computational cost associated with the learning in (39) might not be trivial. However, this would not be a serious concern since this task is done in offline in most cases.

C. What is the Desired Property for System Matrix?

From the discussion above, one might think that the desired signal can be recovered accurately as long as the original signal vector is sparse. Unfortunately, this is not always true since the accurate recovery of the sparse vector would not be possible when a poorly designed system matrix is used. For example, suppose the support Ω of \mathbf{s} is $\Omega = \{1, 3\}$ and the first and third columns of \mathbf{H} are the same, then by no means the recovery algorithm will work properly. This also motivates that the columns in \mathbf{H} should be designed to be as orthogonal to each other as possible. Intuitively, the more the system matrix preserves the energy of the original signals, the better the quality of the recovered signal would be. The system matrices supporting this idea need to be designed such that each element of the measurement vector contains similar amount of information on the input vector \mathbf{s} . That is the place where the random matrix comes into play. Although an exact quantification of the system matrix is complicated (see also next subsection), good news is that most of random matrices, such as Gaussian ($\mathbf{H}_{i,j} \sim N(0, \frac{1}{m})$) or Bernoulli with equal probability ($\mathbf{H}_{i,j} = \pm \frac{1}{m}$), well preserve the energy of the original sparse signal.

When the CS technique is applied to the wireless communications, the system matrix \mathbf{H} can be determined by the process of generating the transmit signal and/or wireless channel characteristics. Fortunately, many of system matrices in wireless communication systems behave like a random matrix. Similarly, the system matrix is modeled by a Bernoulli random matrix when the channel estimation is performed for code division multiplexing access (CDMA) systems. Fading channel is often modeled as Gaussian random variables so that the channel matrix whose columns correspond to the channel vectors between mobile terminal and the base-station can be well modeled as a Gaussian random matrix.

While the system matrix of many wireless applications works well in many cases, in some case we can also design the system matrix to improve the reconstruction quality. This task, called *sensing matrix design*, is classified into two approaches. In the first approach, we assume that the desired signal \mathbf{x} is sparsely represented in a dictionary \mathbf{D} . Then, the system model is expressed as $\mathbf{y} = \mathbf{H}\mathbf{x} = \mathbf{H}\mathbf{D}\mathbf{s}$. In this setup, the goal is to design \mathbf{H} adapting to dictionary \mathbf{D} such that columns in the combined dictionary $\mathbf{E} = \mathbf{H}\mathbf{D}$ has good geometric properties [80]–[82]. In other words, we design \mathbf{H} such that the columns in \mathbf{E} are as orthogonal to each other as possible. In the second type, rows of \mathbf{H} are *sequentially* designed using previously collected measurements as guidance [83], [84]. The

main idea of this approach is to estimate the support from previous measurements and then allocate the sensing energy to the estimated support element. Recently, the system design strategies to generate a nice structure of system matrix in terms of recovery performance for massive MIMO systems were proposed [44], [85].

In summary, in order to make the CS technique effective, columns of the system matrix should be as uncorrelated as possible. Simple and effective way to do so is to design the elements of the system matrix as random as possible. One can check this by computing the mutual coherence $\mu(\mathbf{H})$ or average coherence (taking average of all inner product instead of finding maximum). Also, one can infer from the bound in Section II.D ($\mu(\mathbf{H}) \geq \frac{1}{\sqrt{m}}$) that the coherence of the system matrix improves (decreases) with the dimension and so would be the resulting recovery performance. Also, similar to the dictionary learning, we can design (learn) the system matrix to achieve better recovery performance. This is interesting research topic since the recovery performance depends strongly on the system matrix, simple and efficient design strategy would be beneficial to improve the performance.

D. What Recovery Algorithm Should We Use?

When the researchers consider the CS techniques in their applications, they can be confused by a plethora of algorithms. There are hundreds of sparse recovery algorithms in the literatures, and still many new ones are proposed every year. The tip for not being flustered in a pile of algorithms is to clarify the main issues like the target specifications (performance requirements and complexity budget), system environments (quality of system matrix, operating SNR regime), dimension of measurements and signal vectors, and also availability of the extra information. Perhaps two most important issues in the design of CS-based wireless communication systems are the mapping of the wireless communication problem into an appropriate CS problem and the identification of the right recovery algorithm. Often, one should modify the algorithm to meet the system requirements. Obviously, identifying the best algorithm for the target application is by no means straightforward and one should have basic knowledge of the sparse recovery algorithm. In this subsection, we provide a brief overview on four major approaches: *ℓ_1 -norm minimization*, *greedy algorithm*, *iterative algorithm*, and *statistical sparse recovery technique*. Although not all sparse recovery algorithms can be grouped into these categories, these four are important in various standpoints such as popularity, effectiveness, and historical value.

- **Convex optimization approach (ℓ_1 -norm minimization):** As mentioned, with the knowledge of the signal \mathbf{s} being sparse, the most natural way is to find a sparse input vector under the system constraint ($\arg \min \|\mathbf{s}\|_0$ s.t. $\mathbf{y} = \mathbf{H}\mathbf{s}$). Since the objective function $\|\mathbf{s}\|_0$ is non-convex, solution of this problem can be found in a combinatoric way. As an approach to overcome the computational bottleneck of ℓ_0 -norm minimization, ℓ_1 -norm minimization has been used. If the noise power is bounded to ϵ , ℓ_1 -minimization problem is expressed as

$$\arg \min \|\mathbf{s}\|_1 \text{ s.t. } \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2 < \epsilon.$$

Basis pursuit de-noising (BPDN) [11], also called Lasso [86], relaxes the hard constraint on the reconstruction error by introducing a soft weight λ as

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2 + \lambda \|\mathbf{s}\|_1. \quad (40)$$

The recovery performance of such ℓ_1 -minimization method can be further enhanced by solving a sequence of weighted ℓ_1 optimization [87]. Although the ℓ_1 -minimization problem is a convex optimization problem and thus efficient solvers exist, computational complexity of this approach is still burdensome in implementing real-time wireless communication systems.

- **Greedy algorithm:** In principle, main principle of the greedy algorithm is to successively identify the subset of support (index set of nonzero entries) and refine them until a good estimate of the support is found. Suppose the support is found accurately, then the estimation of support elements would be straightforward since one can convert the underdetermined system into overdetermined one by removing columns corresponding to the zero element in \mathbf{s} and then use a conventional estimation scheme like MMSE or least squares (LS) estimator. In many cases, greedy algorithm attempts to find the support in an iterative fashion, obtaining a sequence of estimates $(\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_n)$. While the OMP algorithm picks an index of column of \mathbf{H} one at a time using a greedy strategy [16], recently proposed variants of OMP, such as generalized OMP (gOMP) [20], compressive sampling matching pursuit (CoSaMP) [88], subspace pursuit (SP) [21], and multipath matching pursuit (MMP) [89], have a refined step to improve the recovery performance. For example, gOMP selects multiple promising columns in each iteration. CoSaMP [88] and SP [21] incorporate special procedures to refine the set of column indices by 1) choosing more than k columns of \mathbf{H} , 2) recovering the signal coefficients based on the projection onto the space of the selected columns, and 3) rejecting those might not be in the true support. MMP finds the best candidate among multiple promising candidates obtained from the tree search. In general, these approaches outperform the OMP algorithm at the cost of higher computational complexity. In summary, the greedy algorithm has computational advantage over the convex optimization approach while achieving comparable (sometimes better) performance.
- **Iterative algorithm:** Sparse solution can be found by refining the sparse signal estimate in an iterative fashion. This approach called iterative hard thresholding (IHT) [90], [91] performs the following update step iteratively
$$\hat{\mathbf{s}}^{(i+1)} = T \left(\hat{\mathbf{s}}^{(i)} + \mathbf{H}^H (\mathbf{y} - \mathbf{H}\hat{\mathbf{s}}^{(i)}) \right), \quad (41)$$
 where $\hat{\mathbf{s}}^{(i)}$ is the estimate of the signal vector \mathbf{s} at the i th iteration and $T(\cdot)$ is the thresholding operator. Algorithms based iterative thresholding yet exhibiting improved performance have been proposed in [92], [93].
- **Statistical sparse recovery:** Statistical sparse recovery algorithms treat the signal vector \mathbf{s} as a random vector

TABLE I
SUMMARY OF SPARSE RECOVERY ALGORITHMS

Approach	Algorithm	Features	Computational Complexity
Convex optimization	BPDN [11]	Reconstruction error $\ \mathbf{y} - \mathbf{H}\mathbf{s}\ _2$ regularized with ℓ_1 norm $\ \mathbf{s}\ _1$ is minimized.	Complexity is $O(m^2n^3)$. Speed of optimization for the BPDN has been substantially improved in [13].
	Reweighted ℓ_1 minimization [87]	The BPDN can be improved via iterative reweighted ℓ_1 -minimization.	Complexity is $O(m^2n^3 \cdot \text{iter})$. Computational complexity of this approach is higher than the BPDN.
Greedy algorithm	OMP [16], gOMP [20]	The indices of nonzero elements of \mathbf{s} are identified in an iterative fashion.	Complexity is $O(mnk)$. Complexity is low for small k .
	CoSaMp [88], SP [21]	More than k indices of the nonzero elements of \mathbf{s} are found and then candidates of poor quality are pruned afterwards.	Complexity is $O(mn \cdot \text{iter})$. It requires higher complexity than the OMP.
	MMP [89]	Tree search algorithm is adopted to search for the indices of the nonzero elements in \mathbf{s} efficiently.	The complexity is higher than that of the OMP. The tree-based search offers the trade-off between performance and complexity.
Iterative algorithm	IHT [90]	Iterative thresholding step in (41) is performed repeatedly. It works well under limited favorable scenarios.	Complexity is $O(mn \cdot \text{iter})$. Implementation cost is low.
	AMP [93]	Gaussian approximations in message passing are used to derive the algorithm.	Complexity is comparable to that of the IHT.
Statistical sparse recovery	MAP with Laplacian prior [97]	MAP estimation of the sparse vector is derived using Laplacian distribution as sparsity-promoting prior distribution.	Complexity is comparable to that of the BPDN.
	SBL [100], BCS [98]	Hyper-parameter is used to model the sparsity of the sparse signals. The EM algorithm is used to find the hyper-parameter and signal vector iteratively.	Complexity is $O(n^3)$. Computational complexity can be high for large size problems due to matrix inversion operation.

and then infer it using the Bayesian framework. In the maximum-a-posteriori (MAP) approach, for example, an estimate of \mathbf{s} is expressed as

$$\hat{\mathbf{s}} = \arg \max_{\mathbf{s}} \ln f(\mathbf{s}|\mathbf{y}) = \arg \max_{\mathbf{s}} \ln f(\mathbf{y}|\mathbf{s}) + \ln f(\mathbf{s}),$$

where $f(\mathbf{s})$ is the prior distribution of \mathbf{s} . To model the sparsity nature of the signal vector \mathbf{s} , $f(\mathbf{s})$ is designed in such a way that it decreases with the magnitude of \mathbf{s} . Well-known examples include i.i.d. Gaussian and Laplacian distribution. For example, if i.i.d. Laplacian distribution is used, then the prior distribution $f(\mathbf{s})$ is expressed as

$$f(\mathbf{s}) = \left(\frac{\lambda}{2}\right)^n \exp\left(-\lambda \sum_{i=1}^n |s_i|\right).$$

Note that the MAP-based approach with the Laplacian prior model leads to the algorithm similar to the BPDN in (40). When one chooses other super-Gaussian priors, the model reduces to a regularized least squares problem [94]–[96], which can be solved by a sequence of reweighted ℓ_1 or ℓ_2 algorithms.

Yet another statistical sparse recovery algorithms include sparse Bayesian learning (SBL) [120] and Bayesian compressed sensing [98]. In these approaches, the priori distribution of the signal vector \mathbf{s} is modeled as zero-mean Gaussian with the variance parameterized by a hyper-parameter. In SBL, for example, it is assumed that each element of \mathbf{s} is a zero-mean Gaussian random

variable with variance γ_k (i.e., $s_k \sim \mathcal{N}(0, \gamma_k)$). A suitable prior on the variance γ_k allows the modeling of several super-Gaussian densities. Often a non-informative prior is used and found to be effective. Let $\boldsymbol{\gamma} = \{\gamma_k, \forall k\}$, then the hyperparameters $\Theta = \{\boldsymbol{\gamma}, \sigma^2\}$ which control the distribution of \mathbf{s} and \mathbf{y} can be estimated from data by marginalizing over \mathbf{s} and then performing evidence maximization or Type-II maximum-likelihood estimation [99]:

$$\begin{aligned} \hat{\Theta} &= \arg \max_{\Theta} p(\mathbf{y}; \boldsymbol{\gamma}, \sigma^2) \\ &= \arg \max_{\Theta} \int p(\mathbf{y}|\mathbf{s}; \sigma^2) p(\mathbf{s}; \boldsymbol{\gamma}) d\mathbf{s}. \end{aligned} \quad (42)$$

The signal \mathbf{s} can be inferred from the maximum-a-posterior (MAP) estimate after obtaining $\hat{\Theta}$:

$$\mathbf{s} = \arg \max_{\mathbf{s}} p(\mathbf{s}|\mathbf{y}; \hat{\Theta}). \quad (43)$$

By solving (42), we obtain the solution of $\boldsymbol{\gamma}$ with most of elements being zero. Note that $\boldsymbol{\gamma}$ controls the variance of \mathbf{s} . When $\gamma_k = 0$, it implies $s_k = 0$, which is a sparse solution. It has been shown that with appropriately chosen parameters, the SBL algorithm outperforms ℓ_1 norm minimization and iteratively reweighted algorithms [100].

In Table I, we summarize the key features of the sparse recovery algorithms and briefly comment on their computational complexity.

To sum up, choosing the right algorithm is very important but it is not easy to find the algorithm fulfilling the designer's

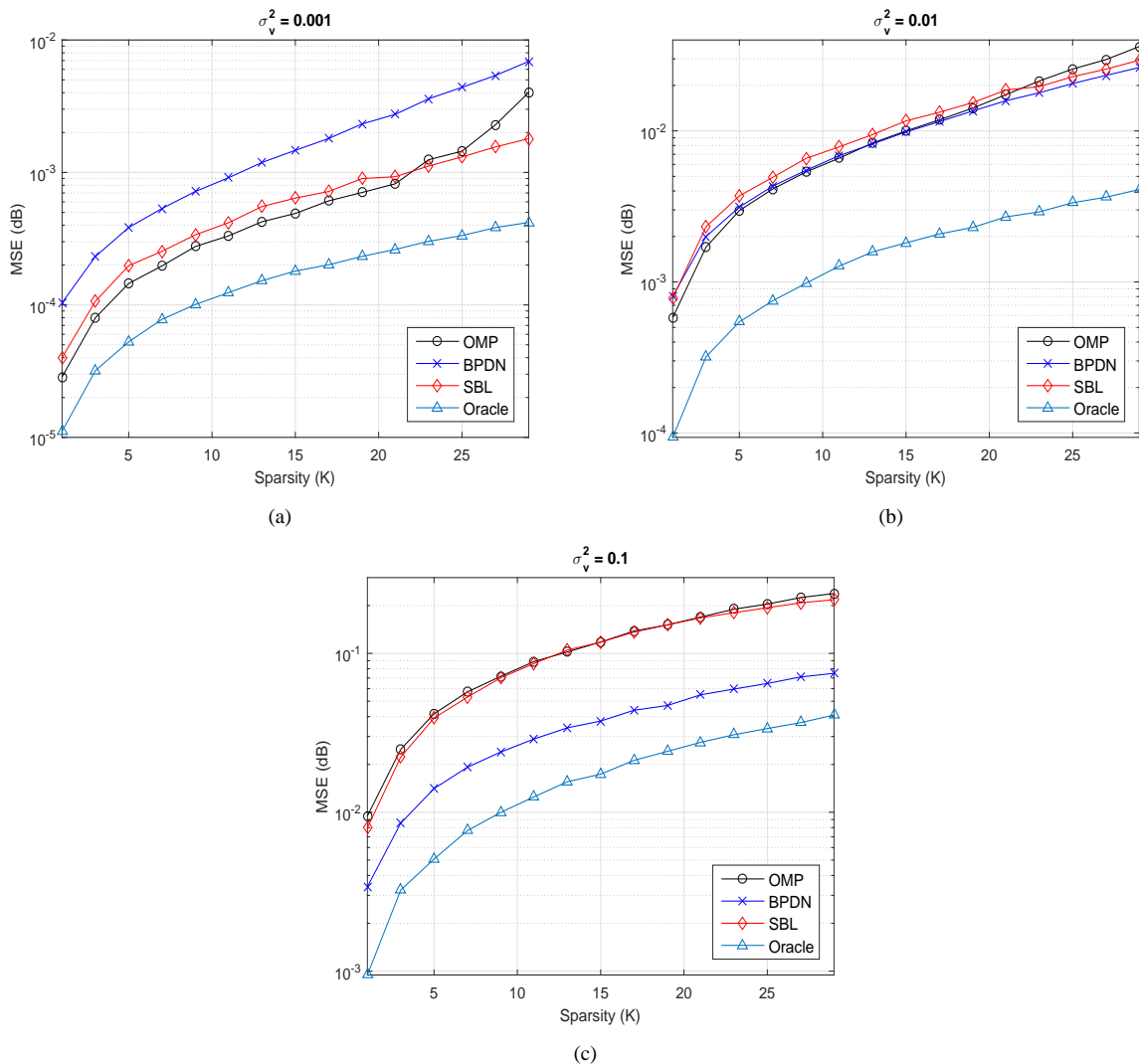


Fig. 16. Performance comparison of several recovery algorithms (a) in high SNR ($\sigma_v^2 = 0.001$), (b) in mid SNR ($\sigma_v^2 = 0.01$), and (c) in low SNR ($\sigma_v^2 = 0.1$).

expectation in a pile of algorithms. The performance and complexity depend on various parameters like problem size m and n , sparsity level k , quality of sensing matrix, and noise intensity. In order to provide simple idea, we present the performance of three well-known recovery algorithms BPDN, OMP, and SBL for different setups. The sensing matrix \mathbf{H} is generated from $N(0, 1/n)$ and the nonzero elements of the signal vector \mathbf{s} are from i.i.d. Gaussian distribution $N(0, 1)$. We fix $m = 100$ and $n = 256$ and vary the sparsity level k and the noise variance σ_v^2 . As a reference, we also plot the Oracle estimation where the signal recovery is performed using the perfect knowledge of the support. Fig. 16 shows the MSE performance as a function of sparsity level k for several recovery algorithms. We consider different levels of noise variance $\sigma_v^2 = 0.001, 0.01$ and 0.1 in Fig. 16 (a), (b) and (c), respectively. When the noise level is small (i.e., high SNR) and the sparsity level is less than 20, the OMP algorithm outperforms the BPDN and SBL. This is not a surprise because the support found by the OMP is very accurate for this scenario so that the quality of the resulting estimate is also very reliable. Note that the SBL slightly outperforms

the OMP when the sparsity level exceeds 20. When noise variance increases to the mid-level, the performance of all three algorithms is more or less similar. Whereas, when the noise level is high, the BPDN performs significantly better than the OMP and SBL. It is clear from these results that the performance of the recovery algorithms depends strongly on the system parameters. Performance is also sensitive to the structure of sensing matrix and the distribution of the signal vector. Hence, for the given setup and operating condition, one should check the empirical performance to identify the right algorithm.

E. Can We Do Better If Multiple Measurement Vectors Are Available?

In many wireless communication applications, such as the wireless channel estimation problem and AoA and AoD estimation in mmWave communication systems [35], [101], multiple snapshots (more than one observation) are available and further the nonzero positions of these vectors are invariant or varying slowly. The problem to recover the sparse

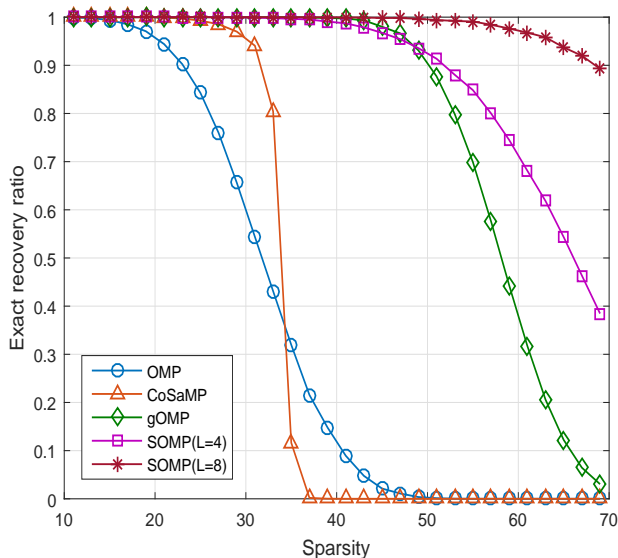


Fig. 17. Performance comparison between OMP and SOMP ($M = 32$, $N = 64$, $k = 8$).

vector from multiple observations, often called multiple measurement vectors (MMV) problem, received much attention recently due to its superior performance compared to the single measurement vector (SMV) problem (see Fig. 17). Group of measurements sharing common support are useful in many wireless communication applications since multiple measurements exploit correlation among sparse signal vectors and also filter out noise component and interference. Wireless channel estimation is a good example since the support of the channel impulse response does not change much over time and across different MIMO antenna pairs [31], [44], [102], [103]. In addition, temporal correlations between multiple source vectors (e.g., correlated fading of the channel gain) can be exploited in the algorithm design [46], [104]–[106].

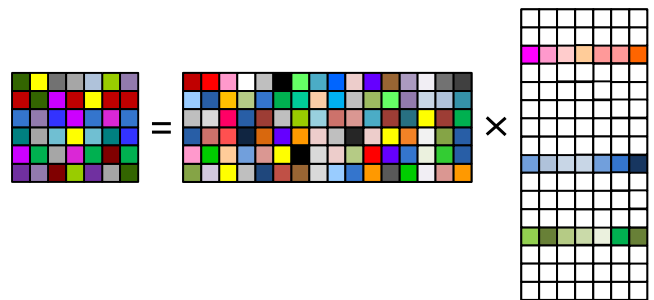
Depending on how the support information is shared among multiple measurement vectors, MMV scenario can be divided into three distinct cases:

- 1) The supports of multiple measurement vectors are the same but the values for nonzero positions are distinct. The system matrix for each measurement vector is identical.
- 2) The supports of multiple measurement vectors are the same but the values for nonzero positions are distinct. The system matrix for all measurement vectors are also distinct.
- 3) The support of multiple measurement vectors are slightly different.

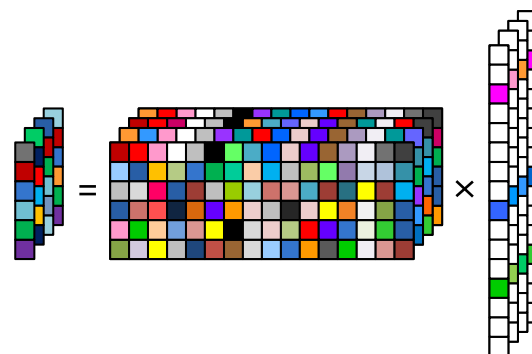
The first scenario is the most popular scenario of the MMV problem (see Fig. 18(a)). In this scenario, we express the measurement vectors as

$$\mathbf{Y} = \mathbf{H}\mathbf{S} + \mathbf{N} \quad (44)$$

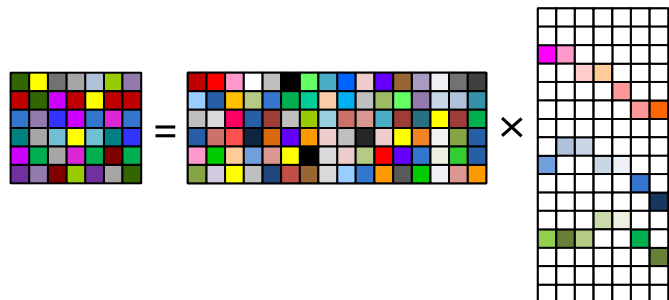
where $\mathbf{Y} = [\mathbf{y}_1 \cdots \mathbf{y}_N]$, $\mathbf{S} = [\mathbf{s}_1 \cdots \mathbf{s}_N]$, and $\mathbf{N} = [\mathbf{n}_1 \cdots \mathbf{n}_N]$. The recovery algorithm finds the column indices



(a) Scenario 1: supports are the same and the system matrices are also the same.



(b) Scenario 2: supports are the same but the system matrices are different.



(c) Scenario 3: supports vary slightly in time.

Fig. 18. Illustration of three MMV scenarios.

of \mathbf{H} corresponding to the nonzero row vectors of \mathbf{S} using the measurement matrix \mathbf{Y} . It has been shown from theoretic analysis that the performance of the MMV-based algorithm improves exponentially with the number of measurements [117], [118]. In fact, MMV-based sparse recovery algorithms perform much better than the SMV-based recovery algorithms [107], [109], [119], [120]. Various recovery algorithms have been proposed for MMV scenario. In [108], the convex relaxation method based on the mixed norm has been proposed. In [107], the greedy algorithm called simultaneous OMP (SOMP) is proposed. Statistical sparse estimation techniques for MMV scenario include MSBL [120], AR-SBL [105], and TSBL [104]. In [110], an approach to identify the direction of arrival (DoA) in array signal processing using the MMV model has been investigated. Further improvement in the recovery performance can be achieved by exploiting the statistical correlations between the signal amplitudes [104], [105], [111].

The second scenario is a generalized version of the first scenario in the sense that system matrices are different for all

TABLE II
SUMMARY OF MMV-BASED SPARSE RECOVERY ALGORITHMS

Scenario	References	Remark
Scenario 1	SOMP [107]	Extension of OMP for the MMV setup. Computational complexity of the SOMP is lower than other candidate algorithms.
	Convex relaxation [108]	Mixed ℓ_1 norm is used to replace ℓ_0 norm. The convex optimization package is used for algorithm.
	MSBL [120]	Extension of SBL for the MMV setup. It offers excellent recovery performance but the computational complexity is a bit higher.
	MUSIC-augmented CS [110]	The subspace criterion of MUSIC algorithm is used to identify the support.
	TSBL [104]	Equivalence between block sparsity model and MMV model was used to exploit the correlations between the source vectors.
	AR-SBL [105], Kalman-filtered CS [111]	The multiple source vectors are modeled by auto-regressive process. The support and amplitude of the source vectors is jointly estimated via iterative algorithm.
Scenario 2	KSBL [46]	The auto-regressive process is used to model the dynamics of the source vectors. Kalman filter is incorporated to estimate the support and gains sequentially.
	AMP-MMV [112]	Graphical model is used to describe the variations of the source vectors. Message passing over a part of graph having dense connections is handled via the AMP method [92].
	sKTS [106]	The deterministic binary vector is used to model the sparsity structure of the source vectors. The EM algorithm is used for joint estimation of sparsity pattern and gains.
Scenario 3	Modified-CS [113]	The new elements added to the support detected in the previous measurement vector is found via ℓ_1 optimization. The candidates of poor quality are eliminated via thresholding.
	DCS-AMP [114]	The dynamic change of the support is modeled by the markov process and efficient message passing algorithm based on AMP is applied.

measurement vectors. Extensions of OMP algorithm [121], iteratively reweighted algorithm [122], sparse Bayesian learning algorithm [123], and Kalman-based sparse recovery algorithm [106] has been proposed. In [112], it has been shown that the graph-based inference method is effective in this scenario. In the third scenario, the recovery algorithms need to keep track of temporal variations of the support since the sparsity pattern changes over time. However, since the change is usually very small, the sparsity pattern can be tracked by estimating the difference between two support sets for consecutive measurement vectors [48], [113]. The algorithm employing approximate message passing (AMP) is used for this scenario in [114]. In Table II, we summarize the recovery algorithms based on the MMV model.

In summary, main point in this subsection is that having multiple *correlated* measurements is very helpful in improving the performance. Depending on system setup and model, one may achieve sub-linear, linear, or super-linear performance gain proportional to the number of measurements. Often we spend lots of time on the algorithm selection yet have hard time satisfying required performance. Remembering that good recovery performance is what we want at the end of the day, easy and pragmatic way to achieve the goal is to use multiple measurements (see Fig. 17). In fact, in many static or slowly-varying environments, measurements of the previous sample time are largely correlated to the current ones and one can simply use multiple measurements for better performance. To come up with an efficient CS technique for the time-varying scenario is still an interesting open problem.

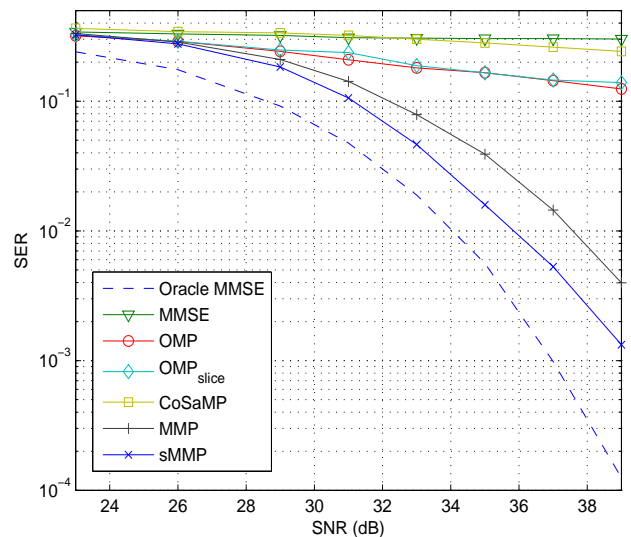


Fig. 19. SER performance when the nonzero positions of input vector is chosen from 16-quadrature amplitude modulation (QAM) ($m = 12, n = 24, k = 5$). sMMP refers to the MMP algorithm equipped with slicing operation.

F. Can We Do Better If Integer Constraint Is Given?

When the nonzero elements of the target vector \mathbf{s} are from the set of finite alphabets, one can exploit this information for the better reconstruction of the sparse vector. In order to

incorporate the integer constraint into the sparse input vector, one can either modify the conventional detection algorithm or incorporate an integer constraint into the sparse recovery algorithm.

First, when the detection approach is used, one can simply add the zero into the constellation set Θ . For example, if nonzero elements of \mathbf{s} are chosen from binary phase shift keying (BPSK) (i.e., $s_i \in \Theta = \{-1, 1\}$), then the modified constellation set becomes $\Theta' = \{-1, 0, 1\}$. Sparsity constraint $\|\mathbf{s}\|_0 = k$ can also be used to limit the search space of the detection algorithm. For example, if the maximum likelihood (ML) tree search (e.g., sphere decoding [115], [116]) is performed, one can stop the search if the cardinality of the symbol vector equals the sparsity.

On the other hand, when the sparse recovery algorithm is used, one should incorporate the quantization step to map the real (complex) value into the symbol. As a simple example, suppose the OMP algorithm is used with BPSK modulation ($\Theta = \{-1, 1\}$). Then, we have $\hat{\mathbf{s}}_i = [0.7 \ -0.6]^T$, then the quantized output becomes $Q_{\Theta}(\hat{\mathbf{s}}_i) = [1 \ -1]^T$. Note, however, that just using the quantized output might not be effective, in particular for the sequential greedy algorithms due to the error propagation. For example, if an index is chosen incorrectly in one iteration, then the estimate will also be incorrect and thus the quantized output will bring additional quantization error, deteriorating the subsequent detection process. In this case, parallel tree search strategy can be a good option to alleviate the error propagation phenomenon. For example, a tree search algorithm performs the parallel search to find multiple promising candidates (e.g., MMP [89]). Among the multiple candidates, the best one minimizing the residual magnitude is chosen in the last minute ($\hat{\mathbf{s}} = \arg \min_{\|\mathbf{s}\|_0=k, \mathbf{s} \in \Theta^k} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|$). The main benefit of tree search method, from the perspective of incorporating the integer slicer, is that it deteriorates the quality of incorrect candidate yet enhances the quality of correct one. This is because the quality of incorrect candidates gets worse due to the additional quantization noise caused by the slicing while no such phenomenon happens to the correct one (note that the quantization error is zero for the correct symbol). As a result, as shown in Fig. 19, the recovery algorithms accounting for the integer constraint of the symbol outperform those without considering this property.

Moral of the story in this subsection is that if additional hint exists, then one can actively use it for better performance. Integer constraint discussed above would be a useful hint in the detection of the sparse vector. However, these hints might be a double-edged sword because it will just do harm if it is improperly used. Thus, one should use with caution in order not to make situation worse. Since the parallel detection strategy we mentioned has cost issue, one can come up with solution to trade-off the cost and performance. Soft quantization can be one option. Iterative detection using a prior information from the decoder would also be a viable option to be investigated.

G. Should We Know Sparsity a Priori?

Some algorithm requires the sparsity of an input signal while others do not need this. For example, sparsity informa-

tion is unnecessary for the ℓ_1 -norm minimization approaches but many greedy algorithms need this since the sparsity is used as stopping criteria of the algorithm. When needed, one should estimate the sparsity using various heuristics. Before we discuss on this, we need to consider what will happen if the sparsity information is incorrect. In a nutshell, setting the number of iterations not being equivalent to the sparsity leads to either early or late termination of the greedy algorithm. In the former case (i.e., *underfitting* scenario), the desired signal will not be fully recovered while some of the noise vector is treated as a desired signal for the latter case (i.e., *overfitting* scenario). Both cases are undesirable, but performance loss is typically more severe for underfitting due to the loss of the signal. Thus, it might be safe to use slightly higher sparsity, especially when the noise effect is not that severe. For example, if the sparsity estimate is \hat{k} , one can take $1.2\hat{k}$ as an iteration number of OMP.

As a sparsity estimation strategy, the residual-based stopping criterion and cross validation [124] are well-known. The residual based stopping criterion is widely used to identify the sparsity level (or iteration number) of the greedy algorithm. Basically, this scheme terminates the algorithm when the residual power is smaller than the pre-specified threshold ϵ (i.e., $\|\mathbf{r}^i\|_2 < \epsilon$). The iteration number at the termination point is set to the sparsity level. However, since the residual magnitude decreases monotonically and the rate of decay depends on the system parameters, it might not be easy to identify the optimal terminating point.

Cross validation (CV) is another technique to identify the model order (sparsity level k in this case) [124]. In this scheme, the measurement vector \mathbf{y} is divided into two parts: a training vector $\mathbf{y}^{(t)}$ and a validation vector $\mathbf{y}^{(v)}$. In the first step, we generate a sequence of possible estimates $\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_n$ using a training vector $\mathbf{y}^{(t)}$, where $\hat{\mathbf{s}}_i$ denotes the estimate of \mathbf{s} obtained under the assumption that the sparsity equals i . In the second step, the sparsity is predicted using the validation vector $\mathbf{y}^{(v)}$. Specifically, for each estimate $\hat{\mathbf{s}}_i$, the validation error $\epsilon_i = \|\mathbf{y}^{(v)} - \mathbf{H}^{(v)}\hat{\mathbf{s}}_i\|_2$ is computed. Initially, when the count i increases, the quality of the estimate improves since more signal elements are added and thus the validation error ϵ_i decreases. However, when the count i exceeds the sparsity, that is, when we choose more columns than needed, we observe no more decrease in ϵ_i and just noise will be added to the validation error. In summary, the validation error roughly looks like a convex function, the number generating the minimum validation error is returned as the sparsity estimate ($\hat{k} = \arg \min_i \epsilon_i$).

When one uses the CS algorithm in which the sparsity is used as an input parameter, one needs to be aware of the sparsity information. More importantly, in order to decide whether the CS technique is useful, one should know if the target signal vector is sparse. There are a variety of ways to check the sparsity. In order to pursue an accurate evaluation of the sparsity, it is desired to use a bit expensive option like cross validation. In many stationary situations, fortunately, one can do it once and use it for a while.

V. CONCLUSION AND FUTURE DIRECTION

In this article, we have provided an overview of the CS technique for wireless communication systems. We discussed basics of CS techniques, three subproblems of CS related to wireless communications, and various wireless applications which CS techniques can be applied to. We also discussed several main issues that one should be aware of and subtle points that one should pay attention to. There are a broad class of wireless applications to which the CS technique would be beneficial and much work remains to be done. We list here some of future research directions.

- It would be interesting to design a flexible CS technique and sparse recovery algorithm that can adapt to various and diverse wireless environments and input conditions. It is very difficult and cumbersome task for wireless practitioners to do calibration and tuning on the various channel conditions and parameters (e.g., measurement size and input sparsity). Thus, it would be useful to come up with solution that adapts to diverse wireless environments.
- Most of the system matrices discussed in the CS theory are Gaussian and random matrices. In many real situations, however, we cannot use the “purely” random matrices and need to use the deterministic matrices. Unfortunately, there is no well-known guideline or design principle for the deterministic sensing matrix. Thus, it would be good to have practical design guideline for the system using the deterministic matrices. Related to this, it would be useful to come up with a simple and effective dictionary learning technique suited for wireless communication systems.
- Most of performance metrics in wireless communication systems are statistical (e.g., BER, BLER) and it would be good to have more realistic analysis tool based on statistical approach. As mentioned, analytic tools to date (e.g., RIP or mutual coherence) are too stringent for wireless communication purpose so it would be helpful to have more flexible tool that bridges the theory and practice.
- It might be worth investigating if the machine learning technique would be helpful in classifying whether the CS technique can be applied or not in a given scenario. Note that it is very difficult to judge if the CS technique is effective in the the complicated wireless communication scenarios. Since we already have large collection of data at the base-station, state-of-the-art machine learning approach like deep neural network (DNN) might be useful to provide a simple answer to this problem.
- What if the system matrix is sparse, not the input vector. In this work, we have primarily discussed the scenario where the desired input vector is sparse. But there are some scenarios where the input-output connection is sparse, not the input vector (e.g., massive multiple access scheme). Solution of this problem will be useful in various machine-type applications. Related to this, applying the matrix completion technique to wireless communication problem would also be interesting research direction

to be pursued in the future.

- As the dimension of wireless communication systems increases, design time and computational complexity increase significantly. We often want an algorithm whose complexity scales linearly with the problem size. Without doubt, for the successful development of CS-based wireless communication systems, low-complexity and fast implementation is of great importance. Development of implementation-friendly algorithm and architecture would expedite the commercialization of CS-based wireless systems.

As a final remark, we hope that this article will serve as a useful guide for wireless communication researchers who want to use the CS technique in their applications. Since our treatment in this paper is rather casual and non-analytical, one should dig into details with further study. However, if the readers learn in mind that essential knowledge in a pile of information is always *sparse*, the journey will not be that painful.

ACKNOWLEDGMENT

The authors would like to thank J. Kim and H. Seo for their help in simulations and the anonymous reviewers for their invaluable comments and suggestions, which helped to improve the quality of the paper.

VI. PROOF OF THEOREM 1

We first consider *only if* case. We assume that there are more than one vector, say \mathbf{s}_1 and \mathbf{s}_2 , satisfying $\mathbf{y} = \mathbf{H}\mathbf{s}$ and both have at most k nonzero elements. Then, by letting $\mathbf{u} = \mathbf{s}_1 - \mathbf{s}_2$, we have $\mathbf{H}\mathbf{u} = \mathbf{0}$ where \mathbf{u} has at most $2k$ nonzero elements. Since $\text{spark}(\mathbf{H}) > 2k$ from the hypothesis, any $2k$ columns in \mathbf{H} are linearly independent, implying that $\mathbf{u} = \mathbf{0}$ and hence $\mathbf{s}_1 = \mathbf{s}_2$. We next consider the *if* case. We assume that, for given \mathbf{y} , there exists at most one k -sparse signal \mathbf{s} satisfying $\mathbf{y} = \mathbf{H}\mathbf{s}$ and $\text{spark}(\mathbf{H}) \leq 2k$. Under this hypothesis, there exist a set of $2k$ columns that are linearly dependent, implying that there exists $2k$ -sparse vector \mathbf{u} in $N(\mathbf{H})$ (i.e., $\mathbf{H}\mathbf{u} = \mathbf{0}$). Since \mathbf{u} is $2k$ -sparse, we can express it into the difference of two k -sparse vectors \mathbf{s}_1 and \mathbf{s}_2 ($\mathbf{u} = \mathbf{s}_1 - \mathbf{s}_2$). Since $\mathbf{H}\mathbf{u} = \mathbf{0}$, $\mathbf{H}(\mathbf{s}_1 - \mathbf{s}_2) = \mathbf{0}$ and hence $\mathbf{H}\mathbf{s}_1 = \mathbf{H}\mathbf{s}_2$, which contradicts the hypothesis that there is at most one k -sparse solution satisfying $\mathbf{y} = \mathbf{H}\mathbf{s}$. Thus, we should have $\text{spark}(\mathbf{H}) > 2k$.

REFERENCES

- [1] D. Donoho, “Compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289-1306, April 2006.
- [2] E. Candes, J. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207-1223, Aug. 2006.
- [3] A. V. Oppenheim and R. W. Schaffer, *Discrete-time Signal Processing*, Prentice Hall, 2010.
- [4] Y. C. Eldar and G. Kutyniok, *Compressed Sensing : Theory and Applications*, Cambridge Univ. Press, 2012.
- [5] E. J. Candes and M. B. Wakin, “An introduction to compressive sampling,” *IEEE Signal Processing Mag.*, vol. 25, pp. 21-30, March 2008.

- [6] Z. Han, H. Li, and W. Yin, *Compressive Sensing for Wireless Networks*, Cambridge Univ. Press, 2013.
- [7] R. G. Baraniuk, "Compressive sensing," *IEEE Signal Processing Mag.*, vol. 24, pp. 118-121, July 2007.
- [8] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Applied and Numerical Harmonic Analysis, Birkhauser, 2013.
- [9] M. Fornasier and H. Rauhut, *Compressive sensing*, In Handbook of Mathematical Methods in Imaging, pp 187-228. Springer, 2011.
- [10] K. Hayashi, M. Nagahara, and T. Tanaka, "A user's guide to compressed sensing for communications systems," *IEICE Trans. Commun.*, vol. E96-B, no. 3, pp. 685-712, March 2013.
- [11] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Scientif. Comput.*, vol. 20, no. 1, pp. 33-61, 1998.
- [12] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, pp. 489-509, Feb. 2006.
- [13] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1 -regularized least squares," *IEEE Journal on Sel. Topics in Signal Process.* vol. 1, pp. 606-617, Dec. 2007.
- [14] <http://cvxr.com/cvx/>
- [15] <http://users.ece.gatech.edu/justin/l1magic/>
- [16] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655-4666, Dec. 2007.
- [17] J. Wang and B. Shim, "Exact recovery of sparse signals using orthogonal matching pursuit: How many iterations do we need?," *IEEE Trans. Signal Process.*, vol. 64, no. 16, pp. 4194-4202, Aug. 2016.
- [18] T. Strohmer and R. Heath, "Grassmannian frames with applications to coding and communication measurements," *Appl. Comput. Harmon. Anal.*, vol. 53, no. 14, pp. 257-275, 2004.
- [19] D. Donoho and M. Elad, "Optimally sparse representation in general (non-orthogonal) dictionaries via L1 minimization," *Proc. Natl. Acad. Sci.*, vol. 100, pp. 2197-2202, 2003.
- [20] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6202-6216, Dec. 2012.
- [21] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, pp. 2230-2249, May 2009.
- [22] T. Zhang, "Sparse recovery with orthogonal matching pursuit under RIP," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 6215-6221, Sept. 2011.
- [23] C. R. Berger and W. Zhaohui, H. Jianzhong, and Z. Shengli, "Application of compressive sensing to sparse channel estimation," *IEEE Commun. Magazine*, vol. 48, no. 11, pp. 164-174, Nov. 2010.
- [24] J.L. Paredes, G.R. Arce and Z. Wang, "Ultra-wideband compressed sensing: channel estimation," *IEEE Journal of Selected Topics in Signal Process.*, vol. 1, no. 3, pp. 383-395, Oct. 2007.
- [25] N. Michelusi, U. Mitra, A.F. Molisch and M. Zorzi, "UWB sparse/diffuse channels, part I: Channel models and Bayesian estimators," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5307-5319, Oct. 2012.
- [26] N. Michelusi, U. Mitra, A.F. Molisch and M. Zorzi, "UWB sparse/diffuse channels, part II: Estimator analysis and practical channels," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5320-5333, Oct. 2012.
- [27] W. Li and J. C. Preisig, "Estimation of rapidly time-varying sparse channels," *IEEE J. Oceanic Eng.*, vol. 32, pp. 927-939, Oct. 2007.
- [28] C. R. Berger, S. Zhou, J. C. Preisig and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: from subspace methods to compressed sensing," *IEEE Trans. Signal Process.*, vol. 58, pp. 1708-1721, March 2010.
- [29] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058-1076, June 2010.
- [30] 3GPP TS 36.101. "User Equipment (UE) Radio Transmission and Reception." *3rd Generation Partnership Project; Technical Specification Group Radio Access Network (E-UTRA)*. URL: <http://www.3gpp.org>.
- [31] X. Rao, and V. K. N. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261-3271, Jun. 2014.
- [32] X. Rao and V. K. N. Lau, "Compressive sensing with prior support quality information and application to massive MIMO channel estimation with temporal correlation," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4914-4924, Sept. 2015.
- [33] T. S. Rappaport, S. Shu, R. Mayzus, Z. Hang, Y. Azar, K. Wang, G. N. Wang, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!," *IEEE Access*, vol. 1, pp. 335-349, 2013.
- [34] O. E. Ayach, S. Rajagopal, S. Abu-Surra, and P. Zhouyue, and R. W. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1499-1513, March 2014.
- [35] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, pp. 831-846, Oct. 2014.
- [36] A. M. Sayeed, "Deconstructing multi-antenna fading channels," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2563-2579, Oct. 2002.
- [37] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge Univ. Press, 2005.
- [38] H.L. Van Trees, *Detection, estimation, and modulation theory, optimum array processing* John Wiley & Sons, 2004.
- [39] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164-1179, June 2014.
- [40] M.K. Samimi and T.S. Rappaport, "3-d millimeter-wave statistical channel model for 5G wireless system design," *IEEE Trans. Microwave Theory and Techniques*, vol. 64, no. 7, pp. 2207-2225, July 2016.
- [41] J. Meng, Y. Yin, Y. Li, N. T. Nguyen, and Z. Han, "Compressive sensing based high-resolution channel estimation for OFDM system," *IEEE Journal of Selected Topics in Signal Process.*, vol. 7, no. 1, pp. 15-25, Feb. 2012.
- [42] W. Ding, C. Pan, L. Dai, and J. Song, "Compressive sensing based channel estimation for OFDM systems under long delay channels," *IEEE Trans. Broadcasting*, vol. 60, no. 2, pp. 313-321, June 2014.
- [43] G. Taubock and F. Hlawatsch, "A compressed sensing technique for OFDM channel estimation in mobile environments: exploiting channel sparsity for reducing pilots," *IEEE Proc. ICASSP 2008*, April 2008, pp. 2885-2888.
- [44] J. W. Choi, B. Shim, and S. Chang, "Downlink Pilot Reduction for Massive MIMO Systems via Compressed Sensing," *IEEE Commun. Letter*, vol. 19, no. 11, pp. 1889-1892, Nov. 2015.
- [45] S. F. Cotter, and B. D. Rao. "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. Commun.*, vol. 50 no. 3, pp. 374-377, March 2002.
- [46] R. Prasad, C. R. Murphy and B. D. Rao, "Joint approximately sparse channel estimation and data detection in OFDM systems using sparse Bayesian learning," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3591-3603, July 2014.
- [47] Z. Gao, L. Dai, W. Dai, B. Shim, and Z. Wang, "Structured compressive sensing-based spatio-temporal joint channel estimation for FDD massive MIMO," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 601-617, Feb. 2016.
- [48] X. Rao and V. K. N. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261-3271, June 2014.
- [49] Z. Gao, L. Dai, Z. Wang, and S. Chen, "Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO," *IEEE Trans. Signal Process.*, vol. 63, no. 23, pp. 6169-6183, Dec. 2015.
- [50] M. Masood, L. H. Afify, and T. Y. Al-Naffouri, "Efficient coordinated recovery of sparse channels in massive MIMO," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 104-118, Jan. 2015.
- [51] T. Al-Naffouri, A. Quadeer, F. Al-Shaalani, and H. Hmida, "Impulsive noise estimation and cancellation in DSL using compressive sampling," *Proc. IEEE International Symp. Circuits and Systems (ISCAS)*, May 2011, pp. 2133-2136.
- [52] A. B. Ramirez, R. E. Carrillo, G. Arce, K. E. Barner, and B. Sadler, "An overview of robust compressive sensing of sparse signals in impulsive noise," *Proc. European Signal Process. Conf. (EUSIPCO) 2015*, pp. 2859-2863.
- [53] T. K. Moon and W. Stirling, *Mathematical Methods And Algorithms For Signal Processing*, Prentice Hall, 1999.
- [54] B. Wang and K. J. R. Liu, "Advances in cognitive radio networks: a survey," *IEEE Journal of Selected Topics on Signal Process.*, vol. 5, no. 1, pp. 5-23, Nov. 2010.
- [55] D. Cohen and Y. C. Eldar, "Sub-Nyquist sampling for power spectrum sensing in cognitive radios: a unified approach," *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3897-3910, August 2014.
- [56] S. Romero and G. Leus, "Wideband spectrum sensing from compressed measurements using spectral prior information," *IEEE Trans. Signal Process.*, vol. 61, no. 24, pp. 6232-6246, Dec. 2013.
- [57] L. Atzori, A. Lera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 11, pp. 2787-2805, Oct. 2010.

- [58] D. Liu, Q. Zhou, Z. Zhang, and B. Liu, "Cluster-based energy-efficient transmission using a new hybrid compressed sensing in WSN," *IEEE Proc. INFOCOM 2016*, pp. 372-376.
- [59] S. Li, L. D. Xu, and X. Wang, "Compressed sensing signal and data acquisition in wireless sensor networks and internet of things," *IEEE Trans. Industrial Informatics*, vol. 9, no. 4, pp. 2177-2186, Nov. 2013.
- [60] S. Pudlewski, A. Prasanna, and T. Melodia, "Compressed-sensing-enabled video streaming for wireless multimedia sensor networks," *IEEE Trans. Mobile Computing*, vol. 11, no. 6, pp. 1060-1072, Jun. 2012.
- [61] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Compressive sensing-based positioning using RSS of WLAN access points," *IEEE Proc. INFOCOM 2010*, pp. 1-9.
- [62] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, Q. Liang, "Sparse target counting and localization in sensor networks based on compressed sensing," *IEEE Proc. INFOCOM 2011*, pp. 2255-2263.
- [63] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Received-signal-strength-based indoor positioning using compressive sensing," *IEEE Trans. Mobile Computing*, vol. 11, no. 12, pp. 1983-1993, Dec. 2012.
- [64] J. Wang, D. Fang, X. Chen, Z. Yang, T. Xing, and L. Cai, "LCS: compressive sensing based device-free localization for multiple targets in sensor networks," *IEEE Proc. INFOCOM 2013*, pp. 145-149.
- [65] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Trans. Systems, Man, and Cybernetics, Part C*, vol. 37, no. 6, pp. 1067-1080, Nov. 2007.
- [66] M. Bshara, U. Orguner, F. Gustafsson, and L. V. Biesen, "Fingerprinting localization in wireless networks based on received-signal-strength measurements: a case study on WiMAX networks," *IEEE Trans. Vehicular Technology*, vol. 59, no. 1, pp. 283-294, Jan. 2010.
- [67] C. Wang, B. Zhang, K. Ren, J. M. Roveda, C. W. Chen, and Z. Xu, "A privacy-aware cloud-assisted healthcare monitoring system via compressive sensing," *IEEE Proc. INFOCOM 2014*, pp. 2130-2138.
- [68] H. Hong, S. Misra, W. Tang, H. Barani, and H. Al-Azzawi, "Applications of compressed sensing in communications networks," arXiv preprint arXiv:1305.3002, 2013.
- [69] 3GPP Technical Report (TR) 45.820, "Cellular system support for ultralow complexity and low throughput Internet of Things (CIoT)."
- [70] METIS D6.6, "Final report on the METIS system concept and technology road map," 2015.
- [71] X. Wang, "Sparse coding multiple (SCMA)," *IEEE*, vol. 5, pp. 5-23, Nov. 2015.
- [72] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor network" *IEEE Signal Proc. Magazine*, pp. 54-69, July 2005.
- [73] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE Journal of Selected Topics Signal Process.*, vol. 10, no. 3, pp. 436-453, Feb. 2016.
- [74] J. W. Choi and B. Shim, "New approach for massive MIMO detection using sparse error recovery," Proc. of *IEEE GLOBECOM conf.* 2014.
- [75] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Comp. Harmonic Anal.*, vol. 31, no. 1, pp. 59-73, July 2011.
- [76] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, no. 2, pp. 349-396, Feb. 2003.
- [77] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311-4322, Nov. 2006.
- [78] Y. Ding and B. D. Rao, "Channel estimation using joint dictionary learning in fdd massive mimo systems," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2015, pp. 185-189.
- [79] 3GPP TS 25.996. "Universal Mobil Telecommunications System (UMTS); Spatial channel model for Multiple Input Multiple Output (MIMO) simulations," version 12.0.0 Release 12, Sep 2014. URL: <http://www.3gpp.org>.
- [80] M. Elad, "Optimized projections for compressed sensing," *IEEE Trans. Signal Process.*, vol. 55, no. 12, pp. 5695-5702, 2007.
- [81] J.M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1395-1408, 2009.
- [82] Y. Yu, A.P. Petropulu, and H.V. Poor, "Measurement matrix design for compressive sensing-based mimo radar," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5338-5352, 2011.
- [83] M.L. Malloy and R.D. Nowak, "Near-optimal adaptive compressed sensing," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4001-4012, 2014.
- [84] J. Haupt, R. Baraniuk, R. Castro, and R. Nowak, "Sequentially designed compressed sensing," in *Statistical Signal Processing Workshop (SSP)*, 2012, pp. 401-404.
- [85] C. Qi and W. Wu, "A study of deterministic pilot allocation for sparse channel estimation in OFDM systems," *IEEE Commun. Letter*, vol. 16, no. 5, pp. 742-744, May 2012.
- [86] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267-288, 1996.
- [87] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, pp. 877-905, 2008.
- [88] D. Needell and J. A. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, pp. 301-321, 2009.
- [89] S. Kwon, J. Wang, and B. Shim, "Multipath matching pursuit," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2986-3001, May 2014.
- [90] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265-274, 2009.
- [91] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Process.*, vol. 4, no. 2, pp. 330-341, April 2010.
- [92] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. of the National Academy of Sciences*, vol. 106, pp. 18914-18919, Nov. 2009.
- [93] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, pp. 764-785, Feb. 2011.
- [94] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600-616, March 1997.
- [95] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Fourier anal. and appl.* vol. 14, no. 5-6, pp. 877-905, Dec. 2008.
- [96] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," *IEEE Proc. ICASSP 2008*, pp. 3869-3872, March 2008.
- [97] M. Figueiredo, "Adaptive sparseness using Jeffreys prior," *Advances in Neural Information Processing Systems (NIPS)*, 2002.
- [98] S. Ji, Y. Xue, and L. Carin, "Bayesian compressed sensing," *IEEE Trans. Signal Process.*, vol. 56, pp. 2346-2356, June 2008.
- [99] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211-244, 2001.
- [100] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153-2164, August 2004.
- [101] Y. Y. Lee, C. Wang, and Y. H. Huang, "A Hybrid RF/Baseband precoding processor based on parallel-index-selection matrix-inversion-bypass simultaneous orthogonal matching pursuit for millimeter wave MIMO systems," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 305-317, Jan. 2015.
- [102] Y. Barbotin, A. Hormati, S. Rangan, and M. Vetterli, "Estimation of sparse MIMO channels with common support," *IEEE Trans. Commun.*, vol. 60, no. 12, pp. 3705-3716, Dec. 2012.
- [103] Z. Gao, L. Dai, W. Dai, B. Shim, and Z. Wang, "Structured compressive sensing-based spatio-temporal joint channel estimation for FDD massive MIMO," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 601-617, Feb. 2016.
- [104] Z. Zhang and B. D. Rao, "Sparse signal recovery with temporally correlated source vectors using sparse Bayesian learning," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, pp. 912-926, Sept. 2011.
- [105] Z. Zhang and B. D. Rao, "Sparse signal recovery in the presence of correlated multiple measurement vectors," *Proc. ICASSP 2010*, Dallas, TX USA, 2010, pp. 3986-3989.
- [106] J. W. Choi and B. Shim, "Statistical recovery of simultaneously sparse time-varying signals from multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6136-6148, Nov. 2015.
- [107] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. Part I: greedy pursuit," *Signal Process.*, vol. 86, no. 3, pp. 572-588, March 2006.
- [108] J. A. Tropp, A. C. Gilbert, M. J. Strauss, "Algorithms for simultaneous sparse approximation. Part II: convex relaxation," *Signal Process.*, vol. 86, pp. 589-602, 2006.
- [109] S. F. Cotter, B. D. Rao, K. Engan, K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 53, pp. 2477-2488, July 2005.
- [110] K. Lee, Y. Bresler, and M. Junge, "Subspace methods for joint sparse recovery," *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 3613-3641, 2012.

- [111] N. Vaswani, "Kalman filtered compressed sensing," *Proc. ICIP* San Diego, CA USA, 2008, pp. 893-896.
- [112] J. Ziniel and P. Schniter, "Efficient high-dimensional inference in the multiple measurement vector problem," *IEEE Trans. Signal Process.*, vol. 61, pp. 340-354, Jan. 2013.
- [113] N. Vaswani and W. Lu, "Modified-CS: modifying compressive sensing for problems with partially known support," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4595-4607, Sept. 2010.
- [114] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270-5284, Nov. 2015.
- [115] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communication," *IEEE Trans. Signal Process.*, vol. 53, pp. 1474-1484, April 2005.
- [116] B. Shim and I. Kang, "Sphere decoding with a probabilistic tree pruning," *IEEE Trans. Signal Process.*, vol. 56, pp. 4867-4878, Oct. 2008.
- [117] Y. C. Eldar and H. Rauhut, "Average case analysis of multichannel sparse recovery using convex relaxation," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 505-519, Jan. 2010.
- [118] R. Gribonval, H. Rauhut, K. Schnass, and P. Vandergheynst, "Atoms of all channels, unite! Average case analysis of multi-channel sparse recovery using greedy algorithms," *J. Fourier Anal. Appl.*, vol. 14, no. 5, pp. 655-687, 2008.
- [119] Y. C. Eldar and H. Rauhut, "Average case analysis of multichannel sparse recovery using convex relaxation," *IEEE Trans. Inf. Theory*, vol. 6, no. 1, pp. 505-519, Jan. 2010.
- [120] D. P. Wipf and B. D. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Trans. Signal Process.* vol. 55, no. 7, pp. 3704-3716, July 2007.
- [121] M. F. Duarte, S. Sarvotham, and D. Baron, M. B. Wakin, and R. G. Baraniuk, "Distributed compressed sensing of jointly sparse signals," *Proc. Asilomar Conf. Signals, Sys. Comput.*, pp. 1537-1541, May 2005.
- [122] Y. Ding and B. D. Rao, "Joint dictionary learning and recovery algorithms in a jointly sparse framework," in *2015 49th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2015, pp. 1482-1486.
- [123] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 92-106, Jan. 2009.
- [124] R. Ward, "Compressed sensing with cross validation," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5773-5782, Dec. 2009.

Jun Won Choi received the B.S. and M.S. degrees in Electrical and Computer Engineering at Seoul National University. In 2010, he earned Ph. D. degree in Electrical and Computer Engineering at University of Illinois at Urbana-Champaign. From 2010 to 2013, he worked for Qualcomm at San Diego USA, where he conducted research on the design of next generation wireless communication systems and advanced baseband signal processing algorithms. Since 2013, he has been on the faculty of the Department of Electrical and Biomedical Engineering at Hanyang University. His research interest includes wireless communications, sparsity-based signal processing, and machine learning.

Byonghyo Shim (SM' 09) received the B.S. and M.S. degrees in control and instrumentation engineering from Seoul National University, Korea, in 1995 and 1997, respectively. He received the M.S. degree in mathematics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, USA, in 2004 and 2005, respectively. From 1997 and 2000, he was with the Department of Electronics Engineering, Korean Air Force Academy as an Officer (First Lieutenant) and an Academic Full-time Instructor. From 2005 to 2007, he was with Qualcomm Inc., San Diego, CA, USA, as a Staff Engineer. From 2007 to 2014, he was with the School of Information and Communication, Korea University, Seoul, as an Associate Professor. Since September 2014, he has been with the Seoul National University (SNU), where he is currently a Professor with the Department of Electrical and Computer Engineering. His research interests include wireless communications, statistical signal processing, estimation and detection, compressed sensing, and information theory.

Dr. Shim is a member of Signal Processing for Communications and Networking Technical Committee of the IEEE Signal Processing Society. He has also been an Associate Editor of the IEEE Wireless Communications Letters, Journal of Communications and Networks, and a Guest Editor of the IEEE Journal on Selected Areas in Communications (JSAC).

Yaogong Ding received the B.E. degree in communication engineering from Xiamen University, Xiamen, China, and the M.S. degree in electrical engineering from the University of California, San Diego, La Jolla, in 2012 and 2015 respectively. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, University of California, San Diego. His research interests lie in the areas of signal processing and communications, including sparse signal processing, compressive sensing, channel modeling and channel estimation for massive MIMO and millimeter wave communication systems.

Bhaskar Rao (S80-M83-SM91-F00) received the B.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 1979 and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California, Los Angeles, in 1981 and 1983, respectively. Since 1983, he has been with the University of California San Diego, La Jolla, where he is currently a Distinguished Professor with the Electrical and Computer Engineering Department and holder of the Ericsson endowed chair in wireless access networks. He was the Director of the Center for Wireless Communications from 2008-2011. His interests are in the areas of digital signal processing, estimation theory, and optimization theory, with applications to digital communications, speech signal processing, and human-computer interactions.

Dr. Rao was elected IEEE Fellow in 2000 for his contributions in high resolution spectral estimation. His research group has received several paper awards. His paper received the best paper award at the 2000 Speech Coding Workshop and his students have received student paper awards at the 2005 and 2006 International Conference on Acoustics, Speech and Signal Processing as well as the best student paper award at the 2006 Neural Information Processing Systems (NIPS). A paper he co-authored with B. Song and R. Cruz received the 2008 Stephen O. Rice Prize Paper Award in the Field of Communications Systems. Another paper, he co-authored with David Wipf, received the 2012 signal processing society best paper award. He is also the recipient of the 2016 signal processing society technical achievement award.

Dong In Kim received the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1990. He was a Tenured Professor with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada. Since 2007, he has been with Sungkyunkwan University (SKKU), Suwon, Korea, where he is currently a Professor with the College of Information and Communication Engineering. Dr. Kim is a first recipient of the NRF of Korea Engineering Research Center in Wireless Communications for RF Energy Harvesting (2014-2021). From 2002 to 2011, he served as an Editor and a Founding Area Editor of Cross-Layer Design and Optimization for the IEEE Transactions on Wireless Communications. From 2008 to 2011, he served as the Co-Editor-in-Chief of the IEEE/KICS Journal of Communications and Networks. He served as the Founding Editor-in-Chief of the IEEE Wireless Communications Letters from 2012 to 2015. From 2001 to 2014, he served as an Editor of Spread Spectrum Transmission and Access for the IEEE Transactions on Communications. He is currently serving as an Editor-at-Large of Wireless Communication I for the IEEE Transactions on Communications.