# Signal Space CoSaMP for Sparse Recovery With Redundant Dictionaries

Mark A. Davenport, *Member, IEEE*, Deanna Needell, and Michael B. Wakin, *Member, IEEE*

***Abstract*—Compressive sensing (CS) has recently emerged as a powerful framework for acquiring sparse signals. The bulk of the CS literature has focused on the case where the acquired signal has a sparse or compressible representation in an orthonormal basis. In practice, however, there are many signals that cannot be sparsely represented or approximated using an orthonormal basis, but that *do* have sparse representations in a redundant dictionary. Standard results in CS can sometimes be extended to handle this case provided that the dictionary is sufficiently incoherent or well conditioned, but these approaches fail to address the case of a truly redundant or overcomplete dictionary. In this paper, we describe a variant of the iterative recovery algorithm CoSaMP for this more challenging setting. We utilize the $D$-RIP, a condition on the sensing matrix analogous to the well-known restricted isometry property. In contrast to prior work, the method and analysis are "signal-focused"; that is, they are oriented around recovering the *signal* rather than its dictionary *coefficients*. Under the assumption that we have a near-optimal scheme for projecting vectors in signal space onto the model family of candidate sparse signals, we provide provable recovery guarantees. Developing a practical algorithm that can provably compute the required near-optimal projections remains a significant open problem, but we include simulation results using various heuristics that empirically exhibit superior performance to traditional recovery algorithms.**

***Index Terms*—Compressive sensing (CS), greedy algorithms, redundant dictionaries, sparse approximation.**

## I. Introduction

### A. Overview

COMPRESSIVE sensing (CS) is a powerful new framework for signal acquisition, offering the promise that we can acquire a vector $\boldsymbol{x} \in \mathbb{C}^n$ via only $m \ll n$ linear measurements provided that $\boldsymbol{x}$ is *sparse* or *compressible*.[1] Specifically, CS considers the problem where we obtain measurements of the form $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}$, where $\boldsymbol{A}$ is an $m \times n$ sensing matrix and $\boldsymbol{e}$ is a noise vector. If $\boldsymbol{x}$ is sparse or compressible and $\boldsymbol{A}$ satisfies certain conditions, then CS provides a mechanism to recover the signal $\boldsymbol{x}$ from the measurement vector $\boldsymbol{y}$ efficiently and robustly.

Typically, however, signals of practical interest are not themselves sparse, but rather have a sparse expansion in some dictionary $\boldsymbol{D}$. By this, we mean that there exists a sparse coefficient vector $\boldsymbol{\alpha}$ such that the signal $\boldsymbol{x}$ can be expressed as $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$. One could then ask the simple question: How can we account for this signal model in CS? In some cases, there is a natural way to extend the standard CS formulation—since we can write the measurements as $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{\alpha} + \boldsymbol{e}$ we can use standard CS techniques to first obtain an estimate $\widehat{\boldsymbol{\alpha}}$ of the sparse coefficient vector. We can then synthesize an estimate $\widehat{\boldsymbol{x}} = \boldsymbol{D}\widehat{\boldsymbol{\alpha}}$ of the original signal. Unfortunately, this is a rather restrictive way to proceed for two main reasons: 1) the application of standard CS results to this problem will require that the matrix given by the product $\boldsymbol{A}\boldsymbol{D}$ satisfy certain properties that will not be satisfied for many interesting choices of $\boldsymbol{D}$, as discussed further below, and 2) we are not really interested in recovering $\boldsymbol{\alpha}$ *per se*, but rather in obtaining an accurate estimate of $\boldsymbol{x}$. If the dictionary $\boldsymbol{D}$ is poorly conditioned, the signal space recovery error $\|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|_2$ could be significantly smaller or larger than the coefficient space recovery error $\|\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}}\|_2$. It may be possible to recover $\boldsymbol{x}$ in situations where recovering $\boldsymbol{\alpha}$ is impossible, and even if we could apply standard CS results to ensure that our estimate of $\boldsymbol{\alpha}$ is accurate, this would not necessarily translate into a recovery guarantee for $\boldsymbol{x}$.

In this paper, we will consider an alternative approach to this problem and develop an algorithm for which we can provide guarantees on the recovery of $\boldsymbol{x}$ while making no direct assumptions concerning our choice of $\boldsymbol{D}$. Before we describe our approach, however, it will be illuminating to see precisely what goes wrong in an attempt to extend the standard CS formulation. Toward this end, let us return to the case where $\boldsymbol{x}$ is itself sparse (when $\boldsymbol{D} = \boldsymbol{I}$). In this setting, there are many possible algorithms that have been proposed for recovering an estimate of $\boldsymbol{x}$ from measurements of the form $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}$, including $\ell_1$-minimization approaches [5], [12] and greedy/iterative methods such as iterative hard thresholding (IHT) [4], orthogonal matching pursuit (OMP) [18], [22], [28], and compressive sampling matching pursuit (CoSaMP) [21]. For any of these algorithms, it can be shown (see [4], [5], [10], [21]) that $\boldsymbol{x}$ can be accurately recovered from the measurements $\boldsymbol{y}$ if the

[1] When we say that a vector $z$ is $k$-sparse, we mean that $\|z\|_0 \overset{\text{def}}{=} |\text{supp}(z)| \leq k \ll n$. A compressible vector is one that is well-approximated as being sparse. We discuss compressibility in greater detail in Section II-B.

Manuscript received August 01, 2012; revised February 07, 2013; accepted June 17, 2013. Date of publication July 16, 2013; date of current version September 11, 2013. This work was supported in part by the NSF under Grant DMS-1004718 and CCF-0830320, NSF CAREER Grant CCF-1149225, and AFOSR Grant FA9550-09-1-0465. This paper was presented in part at the 2012 Asilomar Conference on Signals, Systems, and Computers and in part at the 2013 Signal Processing with Adaptive Sparse Structured Representations (SPARS) Workshop.

M. A. Davenport is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: mdav@gatech.edu).

D. Needell is with the Department of Mathematics and Computer Science, Claremont McKenna College, Claremont, CA 91711 USA (e-mail: dneedell@cmc.edu).

M. B. Wakin is with the Department of Electrical Engineering and Computer Science, Colorado School of Mines, Golden, CO 80401 USA (e-mail: mwakin@mines.edu).

Communicated by Y. Ma, Associate Editor for Signal Processing.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2013.2273491

matrix $\boldsymbol{A}$ satisfies a condition introduced in [8] known as the restricted isometry property (RIP) with a sufficiently small constant $\delta_k$ (the precise requirement on $\delta_k$ varies from method to method). We say that $\boldsymbol{A}$ satisfies the RIP of order $k$ with constant $\delta_k \in (0,1)$ if

$$\sqrt{1-\delta_k} \leq \frac{\|\boldsymbol{Ax}\|_2}{\|\boldsymbol{x}\|_2} \leq \sqrt{1+\delta_k} \qquad (1)$$

holds for all $\boldsymbol{x}$ satisfying $\|\boldsymbol{x}\|_0 \leq k$. Importantly, if $\boldsymbol{A}$ is generated randomly with i.i.d. entries drawn from a suitable distribution, and with a number of rows roughly proportional to the sparsity level $k$, then with high probability $\boldsymbol{A}$ will satisfy (1) [2], [19], [24].

We now return to the case where $\boldsymbol{x}$ is sparse with respect to a dictionary $\boldsymbol{D}$. If $\boldsymbol{D}$ is unitary (i.e., if the dictionary is an orthonormal basis), then the same arguments used to establish (1) can be adapted to show that for a fixed $\boldsymbol{D}$, if we choose a random $\boldsymbol{A}$, then with high probability $\boldsymbol{AD}$ will satisfy the RIP, i.e.,

$$\sqrt{1-\delta_k} \leq \frac{\|\boldsymbol{AD\alpha}\|_2}{\|\boldsymbol{\alpha}\|_2} \leq \sqrt{1+\delta_k} \qquad (2)$$

will hold for all $\boldsymbol{\alpha}$ satisfying $\|\boldsymbol{\alpha}\|_0 \leq k$. Thus, standard CS algorithms can be used to accurately recover $\boldsymbol{\alpha}$, and because $\boldsymbol{D}$ is unitary, the signal space recovery error $\|\boldsymbol{x}-\widehat{\boldsymbol{x}}\|_2$ will exactly equal the coefficient space recovery error $\|\boldsymbol{\alpha}-\widehat{\boldsymbol{\alpha}}\|_2$.

Unfortunately, this approach would not do in cases where $\boldsymbol{D}$ is not unitary and especially in cases where $\boldsymbol{D}$ is highly redundant/overcomplete. For example, $\boldsymbol{D}$ might represent the overcomplete discrete Fourier transform (DFT), the undecimated discrete wavelet transform, a redundant Gabor dictionary, or a union of orthonormal bases. The challenges that we must confront when dealing with $\boldsymbol{D}$ of this form include

1) Redundancy in $\boldsymbol{D}$ will mean that in general, the representation of a vector $\boldsymbol{x}$ in the dictionary is not unique—there may exist many possible coefficient vectors $\boldsymbol{\alpha}$ that can be used to synthesize $\boldsymbol{x}$.
2) Coherence (correlations) between the columns of $\boldsymbol{D}$ can make it difficult to satisfy (2) with a sufficiently small constant $\delta_k$ to apply existing theoretical guarantees. For instance, while the DFT forms an orthonormal basis, a 2 × overcomplete DFT is already quite coherent, with adjacent columns satisfying $|\langle d_i, d_{i+1}\rangle| > 2/\pi > 0.63$. Since the coherence provides a bound on $\delta_k$ for all $k \geq 2$, this means that $\boldsymbol{D}$ itself cannot satisfy the RIP with a constant $\delta_{2k} < 0.63$. For the random constructions of $\boldsymbol{A}$ typically considered in the context of CS, with high probability $\boldsymbol{A}$ will preserve the conditioning of $\boldsymbol{D}$ (good or bad) on each subspace of interest. (For such $\boldsymbol{A}$, one essentially needs $\boldsymbol{D}$ itself to satisfy the RIP in order to expect $\boldsymbol{AD}$ to satisfy the RIP.) Thus, in the case of the 2 × overcomplete DFT, we would expect the RIP constant for $\boldsymbol{AD}$ to be *at least* roughly 0.63—well outside the range for which any of the sparse recovery algorithms described above are known to succeed.
3) As noted above, if the dictionary $\boldsymbol{D}$ is poorly conditioned, the signal space recovery error $\|\boldsymbol{x}-\widehat{\boldsymbol{x}}\|_2$ could differ substantially from the coefficient space recovery error $\|\boldsymbol{\alpha}-\widehat{\boldsymbol{\alpha}}\|_2$, further complicating any attempt to understand how well we can recover $\boldsymbol{x}$ by appealing to results concerning the recovery of $\boldsymbol{\alpha}$.

All of these problems essentially stem from the fact that extending standard CS algorithms in an attempt to recover $\boldsymbol{\alpha}$ is a *coefficient-focused* recovery strategy. By trying to go from the measurements $\boldsymbol{y}$ all the way back to the coefficient vector $\boldsymbol{\alpha}$, one encounters all the problems above due to the lack of orthogonality of the dictionary.

In contrast, in this paper we propose a *signal-focused* recovery strategy for CS. Our algorithm employs the model of sparsity in an arbitrary dictionary $\boldsymbol{D}$ but directly obtains an estimate of the signal $\boldsymbol{x}$, and we provide guarantees on the quality of this estimate in signal space. Our algorithm is a modification of CoSaMP [21], and in cases where $\boldsymbol{D}$ is unitary, our "Signal-Space CoSaMP" algorithm reduces to standard CoSaMP. However, our analysis requires comparatively weaker assumptions. Our bounds require only that $\boldsymbol{A}$ satisfy the $\boldsymbol{D}$-RIP [6] (which we explain below in Section I-C)—this is a different and less-restrictive condition to satisfy than requiring $\boldsymbol{AD}$ to satisfy the RIP. The algorithm does, however, require the existence of a near-optimal scheme for projecting a vector $\boldsymbol{x}$ onto the set of signals admitting a sparse representation in $\boldsymbol{D}$. While the fact that we require only an *approximate* projection is a significant relaxation of the requirements of traditional algorithms like CoSaMP (which require *exact* projections), showing that a practical algorithm can provably compute the required near-optimal projection remains a significant open problem. Nevertheless, as we will see in Section III, various practical algorithms do lead to empirically favorable performance, suggesting that this challenge might not be insurmountable.

### B. Related Work

Our work most closely relates to Blumensath's Projected Landweber Algorithm (PLA) [3], an extension of IHT [4] that operates in signal space and accounts for a union-of-subspaces signal model. In several ways, our paper is a parallel of this one, except that we extend CoSaMP rather than IHT to operate in signal space. Both works assume that $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP, and implementing both algorithms requires the ability to compute projections of vectors in the signal space onto the model family. (These requirements are described more thoroughly in Section I-C below.) One critical difference, however, is that our analysis allows for near-optimal projections whereas the PLA analysis does not.[2] Other fine differences are noted below.

Also related are works that employ an assumption of "analysis sparsity," in which a signal $\boldsymbol{x}$ is analyzed in a dictionary $\boldsymbol{D}$, and recovery from CS measurements is possible if $\boldsymbol{D}^*\boldsymbol{x}$ is sparse or compressible. Conventional CS algorithms such as $\ell_1$-minimization [6], [20], IHT [9], [14], and CoSaMP [14] have been adapted to account for analysis sparsity. These works are similar to ours in that they provide recovery guarantees in signal space and do not require $\boldsymbol{AD}$ to satisfy the RIP. However, the assumption of analysis sparsity is in general different from the

[2] Technically, the analysis of the PLA [3] allows for near-optimal projections but only with an additive error term. For sparse models, however, such projections could be made arbitrarily accurate simply by rescaling the signal before projecting. Thus, we consider the PLA to require exact projections for sparse models.

"synthesis sparsity" that we assume, where there exists a sparse coefficient vector $\boldsymbol{\alpha}$ such that $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$. For example, in the analysis case, exact sparsity implies that the analysis vector $\boldsymbol{D}^*\boldsymbol{x}$ is sparse, whereas in our setting exact sparsity implies the coefficient vector $\boldsymbol{\alpha}$ is. Under both of these assumptions, both the $\ell_1$-analysis method and our signal space CoSaMP algorithm provide recovery guarantees proportional to the norm of the noise in the measurements, $\|\boldsymbol{e}\|_2$ [6]. Without exact sparsity, $\ell_1$-analysis adds an additional factor $\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_k\|_1/\sqrt{k}$, where $(\boldsymbol{D}^*\boldsymbol{x})_k$ represents the $k$ largest coefficients in magnitude of $(\boldsymbol{D}^*\boldsymbol{x})$. In the synthesis sparsity setting, the analogous "tail-term" is less straightforward (see Section II-B below for details). In summary, these algorithms are intended for different signal families and potentially different dictionaries. Nevertheless, there are some similarities between our work and analysis CoSaMP (ACoSaMP) [14] as we will see below.

Finally, it is worth mentioning the loose connection between our work and that in "model-based CS" [1]. In the case where $\boldsymbol{D}$ is unitary, IHT and CoSaMP have been modified to account for structured sparsity models, in which certain sparsity patterns in the coefficient vector $\boldsymbol{\alpha}$ are forbidden. This paper is similar to ours in that it involves a projection onto a model set. However, the algorithm (including the projection) operates in coefficient space (not signal space) and employs a different signal model; the requisite model-based RIP is more similar to requiring that $\boldsymbol{AD}$ satisfy the RIP; and extensions to nonorthogonal dictionaries are not discussed. In fact, our paper is in part inspired by our recent efforts [11] to extend the "model-based CS" framework to a nonorthogonal dictionary in which we proposed a similar algorithm to the one considered in this paper.

### C. Requirements

First, to establish notation, suppose that $\boldsymbol{A}$ is an $m \times n$ matrix and $\boldsymbol{D}$ is an arbitrary $n \times d$ matrix. We suppose that we observe measurements of the form $\boldsymbol{y} = \boldsymbol{Ax} + \boldsymbol{e} = \boldsymbol{AD\alpha} + \boldsymbol{e}$. For an index set $\Lambda \subset \{1, 2, \ldots, d\}$ (sometimes referred to as a *support set*), we let $\boldsymbol{D}_\Lambda$ denote the $n \times |\Lambda|$ submatrix of $\boldsymbol{D}$ corresponding to the columns indexed by $\Lambda$, and we let $\mathcal{R}(\boldsymbol{D}_\Lambda)$ denote the column span of $\boldsymbol{D}_\Lambda$. We also use $\mathcal{P}_\Lambda : \mathbb{C}^n \to \mathbb{C}^n$ to denote the orthogonal projection operator onto $\mathcal{R}(\boldsymbol{D}_\Lambda)$ and $\mathcal{P}_{\Lambda^\perp} : \mathbb{C}^n \to \mathbb{C}^n$ to denote the orthogonal projection operator onto the orthogonal complement of $\mathcal{R}(\boldsymbol{D}_\Lambda)$.[3]

We will approach our analysis under the assumption that the matrix $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP [6]. Specifically, we say that $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP of order $k$ if there exists a constant $\delta_k \in (0, 1)$ such that

$$\sqrt{1-\delta_k} \leq \frac{\|\boldsymbol{AD\alpha}\|_2}{\|\boldsymbol{D\alpha}\|_2} \leq \sqrt{1+\delta_k} \tag{3}$$

holds for all $\boldsymbol{\alpha}$ satisfying $\|\boldsymbol{\alpha}\|_0 \leq k$. We note that this is different from requiring that $\boldsymbol{A}$ satisfy the RIP—although (1) and (3) appear similar, the RIP requirement demands that this condition holds for vectors $\boldsymbol{x}$ containing few nonzeros, while the $\boldsymbol{D}$-RIP requirement demands that this condition holds for vectors $\boldsymbol{x}$ having a sparse representation in the dictionary $\boldsymbol{D}$. We also note that, compared to the requirement that $\boldsymbol{AD}$ satisfy the RIP (2), it is relatively easy to ensure that $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP. In particular, we have the following lemma.

*Lemma I.1 ([11, Corollary 3.1]):* For any choice of $\boldsymbol{D}$, if $\boldsymbol{A}$ is populated with i.i.d. random entries from a Gaussian or sub-Gaussian distribution, then with high probability, $\boldsymbol{A}$ will satisfy the $\boldsymbol{D}$-RIP of order $k$ as long as $m = O(k \log(d/k))$.

In fact, using the results of [17] one can extend this result to show that given any matrix $\boldsymbol{A}$ satisfying the traditional RIP, by applying a random sign matrix one obtains a matrix that with high probability will satisfy the $\boldsymbol{D}$-RIP.

Next, recall that one of the key steps in the traditional CoSaMP algorithm is to project a vector in signal space onto the model family of candidate sparse signals. In the traditional setting (when $\boldsymbol{D}$ is an orthonormal basis), this step is trivial and can be performed by simple thresholding of the entries of the coefficient vector. Our signal space CoSaMP algorithm (described more completely in Section II) involves replacing thresholding with a more general projection of vectors in the signal space onto the signal model. Specifically, for a given vector $\boldsymbol{z} \in \mathbb{C}^n$ and a given sparsity level $k$, define

$$\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k) := \underset{\Lambda:|\Lambda|=k}{\arg\min} \|\boldsymbol{z} - \mathcal{P}_\Lambda \boldsymbol{z}\|_2 .$$

The support $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$—if we could compute it—could be used to generate the best $k$-sparse approximation to $\boldsymbol{z}$; in particular, the nearest neighbor to $\boldsymbol{z}$ among all signals that can be synthesized using $k$ columns from $\boldsymbol{D}$ is given by $\mathcal{P}_{\Lambda_{\mathrm{opt}}(\boldsymbol{z},k)}\boldsymbol{z}$. Unfortunately, computing $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$ may be difficult in general. Therefore, we allow for near-optimal projections to be used in our algorithm. For a given vector $\boldsymbol{z} \in \mathbb{C}^n$ and a given sparsity level $k$, we assume a method is available for producing an estimate of $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$, denoted $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ and having cardinality $|\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)| = k$, that satisfies

$$\begin{aligned} &\left\|\mathcal{P}_{\Lambda_{\mathrm{opt}}(\boldsymbol{z},k)}\boldsymbol{z} - \mathcal{P}_{\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z},k)}\boldsymbol{z}\right\|_2 \\ &\quad \leq \min\left(\epsilon_1 \left\|\mathcal{P}_{\Lambda_{\mathrm{opt}}(\boldsymbol{z},k)}\boldsymbol{z}\right\|_2, \epsilon_2 \left\|\boldsymbol{z} - \mathcal{P}_{\Lambda_{\mathrm{opt}}(\boldsymbol{z},k)}\boldsymbol{z}\right\|_2\right) \end{aligned} \tag{4}$$

for some constants $\epsilon_1, \epsilon_2 \geq 0$. Setting $\epsilon_1$ or $\epsilon_2$ equal to 0 would lead to the requirement that $\mathcal{P}_{\Lambda_{\mathrm{opt}}(\boldsymbol{z},k)}\boldsymbol{z} = \mathcal{P}_{\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z},k)}\boldsymbol{z}$ exactly. Note that our metric for judging the quality of an approximation to $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$ is entirely in terms of its impact in signal space. It might well be the case that $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ could satisfy (4) while being substantially different (or even disjoint) from $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$. Thus, while computing $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$ may be extremely challenging when $\boldsymbol{D}$ is highly redundant, there is hope that efficiently computing an approximation that satisfies (4) might still be possible. However, determining whether this is the case remains an open problem.

It is important to note that although computing a near-optimal support estimate that satisfies the condition (4) remains a challenging task in general, several important related works have run into the same problem. As we previously mentioned, the existing analysis of the PLA [3] actually requires exact computation of $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$. The analysis of ACoSaMP [14] allows a near-optimal projection to be used, with a near-optimality criterion that differs slightly from ours. Simulations of ACoSaMP, however, have relied on practical (but not

[3] Note that $\mathcal{P}_{\Lambda^\perp}$ does *not* represent the orthogonal projection operator onto $\mathcal{R}(\boldsymbol{D}_{\{1,2,\ldots,d\}\setminus\Lambda})$.

theoretically backed) methods for computing this projection. In Section III, we present simulation results for signal space CoSaMP using practical (but not theoretically backed) methods for computing $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$. We believe that computing provably near-optimal projections is an important topic worthy of further study, as it is really the crux of the problem in all of these settings.

## II. Algorithm and Recovery Guarantees

Given noisy compressive measurements of the form $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}$, our signal space CoSaMP algorithm for recovering an estimate of the signal $\boldsymbol{x}$ is specified in Algorithm 1.

**Algorithm 1** Signal Space CoSaMP

```
input: $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, k$, stopping criterion
initialize: $\boldsymbol{r} = \boldsymbol{y}, \boldsymbol{x}^0 = 0, \ell = 0, \Gamma = \emptyset$
while not converged
    proxy:    $\widetilde{\boldsymbol{v}} = \boldsymbol{A}^*\boldsymbol{r}$
    identify:    $\Omega = \mathcal{S}_{\boldsymbol{D}}(\widetilde{\boldsymbol{v}}, 2k)$
    merge:    $T = \Omega \cup \Gamma$
    update:    $\widetilde{\boldsymbol{x}} = \arg\min_{\boldsymbol{z}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{z}\|_2$
                   s.t. $\boldsymbol{z} \in \mathcal{R}(\boldsymbol{D}_T)$
        $\Gamma = \mathcal{S}_{\boldsymbol{D}}(\widetilde{\boldsymbol{x}}, k)$
        $\boldsymbol{x}^{\ell+1} = \mathcal{P}_\Gamma \widetilde{\boldsymbol{x}}$
        $\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^{\ell+1}$
        $\ell = \ell + 1$
end while
output: $\widehat{\boldsymbol{x}} = \boldsymbol{x}^\ell$
```

### *A. Bound for the Recovery of Sparse Signals*

For signals having a sparse representation in the dictionary $\boldsymbol{D}$, we have the following guarantee.

*Theorem II.1:* Suppose there exists a $k$-sparse coefficient vector $\boldsymbol{\alpha}$ such that $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$, and suppose that $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP of order $4k$. Then, the signal estimate $\boldsymbol{x}^{\ell+1}$ obtained after $\ell + 1$ iterations of signal space CoSaMP satisfies

$$\left\|\boldsymbol{x} - \boldsymbol{x}^{\ell+1}\right\|_2 \leq C_1 \left\|\boldsymbol{x} - \boldsymbol{x}^{\ell}\right\|_2 + C_2 \left\|\boldsymbol{e}\right\|_2, \tag{5}$$

where $C_1$ and $C_2$ are constants that depend on the isometry constant $\delta_{4k}$ and on the approximation parameters $\epsilon_1$ and $\epsilon_2$. In particular,

$$C_1 = ((2 + \epsilon_1)\delta_{4k} + \epsilon_1)(2 + \epsilon_2)\sqrt{\frac{1 + \delta_{4k}}{1 - \delta_{4k}}}$$

and

$$C_2 = \left(\frac{(2 + \epsilon_2)\left((2 + \epsilon_1)(1 + \delta_{4k}) + 2\right)}{\sqrt{1 - \delta_{4k}}}\right).$$

Our proof of Theorem II.1 appears in the Appendix and is a modification of the original CoSaMP proof [21]. Through various combinations of $\epsilon_1$, $\epsilon_2$, and $\delta_{4k}$, it is possible to ensure that $C_1 < 1$ and thus that the accuracy of signal space CoSaMP improves at each iteration. Taking $\epsilon_1 = \frac{1}{10}$, $\epsilon_2 = 1$, and $\delta_{4k} = 0.029$ as an example, we obtain $C_1 \leq 0.5$ and $C_2 \leq 12.7$. Applying the relation (5) recursively, we then conclude the following.

*Corollary II.1:* Suppose there exists a $k$-sparse coefficient vector $\boldsymbol{\alpha}$ such that $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$, and suppose that $\boldsymbol{A}$ satisfies the $\boldsymbol{D}$-RIP of order $4k$ with $\delta_{4k} = 0.029$. Suppose that signal space CoSaMP is implemented using near-optimal projections with approximation parameters $\epsilon_1 = \frac{1}{10}$ and $\epsilon_2 = 1$. Then, the signal estimate $\boldsymbol{x}^\ell$ obtained after $\ell$ iterations of signal space CoSaMP satisfies

$$\left\|\boldsymbol{x} - \boldsymbol{x}^{\ell}\right\|_2 \leq 2^{-\ell} \left\|\boldsymbol{x}\right\|_2 + 25.4 \left\|\boldsymbol{e}\right\|_2. \tag{6}$$

By taking a sufficient number of iterations $\ell$, the first term on the right-hand side of (6) can be made arbitrarily small, and ultimately the recovery error depends only on the level of noise in the measurements. For a precision parameter $\eta$, this shows that at most $O(\log(\|\boldsymbol{x}\|_2 / \eta))$ iterations are needed to ensure that

$$\|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|_2 = O(\eta + \|\boldsymbol{e}\|_2) = O(\max\{\eta, \|\boldsymbol{e}\|_2\}).$$

The cost of a single iteration of the method is dominated by the cost of the *identify* and *update* steps, where we must obtain sparse approximations to $\widetilde{\boldsymbol{v}}$ and $\widetilde{\boldsymbol{x}}$, respectively. We emphasize again that there is no known algorithm for computing the approximation $\mathcal{S}_{\boldsymbol{D}}$ efficiently, and the ultimate computational complexity of the algorithm will depend on this choice. However, in the absence of a better choice, one natural option for estimating $\mathcal{S}_{\boldsymbol{D}}$ is to use a greedy method such as OMP or CoSaMP (see Section III for experimental results using these choices). The running time of these greedy methods on an $n \times d$ dictionary $\boldsymbol{D}$ are $O(knd)$ or $O(nd)$, respectively, [21]. Therefore, using these methods as approximations in the *identify* and *update* steps yields an overall running time of $O(knd \log(\|\boldsymbol{x}\|_2 / \eta))$ or $O(nd \log(\|\boldsymbol{x}\|_2 / \eta))$. For sparse signal recovery, these running times are in line with state-of-the-art bounds for traditional CS algorithms such as CoSaMP [21], except that signal space CoSaMP can be applied in settings where the dictionary $\boldsymbol{D}$ is not unitary. The error bounds of Corollary II.1 also match those of classical results, except that here we assume suitable accuracy of $\mathcal{S}_{\boldsymbol{D}}$. Of course, since no efficient near-optimal projection method is known, this presents a weakness in these results, but it is one shared by all comparable results in the existing literature.

### *B. Discussion Concerning the Recovery of Arbitrary Signals*

We can extend our analysis to account for signals $\boldsymbol{x}$ that do not exactly have a sparse representation in the dictionary $\boldsymbol{D}$. For the sake of illustration, we again take $\epsilon_1 = \frac{1}{10}$, $\epsilon_2 = 1$, and $\delta_{4k} = 0.029$, and we show how (6) can be extended.

We again assume measurements of the form $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}$ but allow $\boldsymbol{x}$ to be an arbitrary signal in $\mathbb{C}^n$. For any vector $\boldsymbol{\alpha}_k \in \mathbb{C}^d$ such that $\|\boldsymbol{\alpha}_k\|_0 \leq k$, we can write

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{\alpha}_k + \boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k) + \boldsymbol{e}.$$

The term $\widetilde{\boldsymbol{e}} := \boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k) + \boldsymbol{e}$ can be viewed as noise in the measurements of the $k$-sparse signal $\boldsymbol{D}\boldsymbol{\alpha}_k$. Then, by (6) we have

$$\begin{aligned}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k - \boldsymbol{x}^\ell\right\|_2 &\\ &\le 2^{-\ell}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 + 25.4\left\|\widetilde{\boldsymbol{e}}\right\|_2\\ &\le 2^{-\ell}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 + 25.4\left(\left\|\boldsymbol{e}\right\|_2 + \left\|\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k)\right\|_2\right), \quad (7)\end{aligned}$$

and so using the triangle inequality,

$$\begin{aligned}\left\|\boldsymbol{x} - \boldsymbol{x}^\ell\right\|_2 \le 2^{-\ell}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 + 25.4\left\|\boldsymbol{e}\right\|_2 + \left\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2&\\ + 25.4\left\|\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k)\right\|_2. \quad (8)&\end{aligned}$$

One can then choose the coefficient vector $\boldsymbol{\alpha}_k$ to minimize the right-hand side of (8).

Bounds very similar to this appear in the analysis of both the PLA [3] and ACoSaMP [14]. Such results are somewhat unsatisfying since it is unclear how the term $\left\|\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k)\right\|_2$ might behave. Unfortunately, these bounds are difficult to improve upon when $\boldsymbol{D}$ is not unitary. Under some additional assumptions, however, we can make further modifications to (8). For example, the following proposition allows us to bound $\left\|\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k)\right\|_2$.

*Proposition II.1 ([21, Proposition 3.5]):* Suppose that $\boldsymbol{A}$ satisfies the upper inequality of the RIP, i.e., that $\left\|\boldsymbol{A}\boldsymbol{x}\right\|_2 \le \sqrt{1+\delta_k}\left\|\boldsymbol{x}\right\|_2$ holds for all $\boldsymbol{x} \in \mathbb{C}^n$ with $\left\|\boldsymbol{x}\right\|_0 \le k$. Then, for every signal $\boldsymbol{z} \in \mathbb{C}^n$,

$$\left\|\boldsymbol{A}\boldsymbol{z}\right\|_2 \le \sqrt{1+\delta_k}\left[\left\|\boldsymbol{z}\right\|_2 + \frac{1}{\sqrt{k}}\left\|\boldsymbol{z}\right\|_1\right]. \quad (9)$$

Plugging this result in to (8), we have

$$\begin{aligned}\left\|\boldsymbol{x} - \boldsymbol{x}^\ell\right\|_2 \le 2^{-\ell}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 &+ 25.4\left\|\boldsymbol{e}\right\|_2\\ &+ (25.4\sqrt{1+\delta_k} + 1)\left\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2\\ &+ \frac{25.4\sqrt{1+\delta_k}}{\sqrt{k}}\left\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k\right\|_1. \quad (10)\end{aligned}$$

For any $\boldsymbol{x} \in \mathbb{C}^n$, one could define the model mismatch quantity

$$M(\boldsymbol{x}) := \inf_{\boldsymbol{\alpha}_k : \left\|\boldsymbol{\alpha}_k\right\|_0 \le k}\left[\left\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 + \frac{1}{\sqrt{k}}\left\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k\right\|_1\right].$$

We remark that this mismatch quantity is analogous to the tail bounds in the literature for methods which do not allow for redundant dictionaries. In particular, the $\ell_1$-norm term in the classical setting is required on account of geometric results about Gelfand widths [13], [16]. If this quantity is large, then the signal is far from compressible and we are not in a setting for which our method is designed. Plugging this definition into (10), we obtain

$$\begin{aligned}\left\|\boldsymbol{x} - \boldsymbol{x}^\ell\right\|_2 \le 2^{-\ell}\left\|\boldsymbol{D}\boldsymbol{\alpha}_k\right\|_2 + 25.4\left\|\boldsymbol{e}\right\|_2&\\ + 26.4\sqrt{1+\delta_k}\cdot M(\boldsymbol{x}). \quad (11)&\end{aligned}$$

In some sense, one can view $M(\boldsymbol{x})$ as the "distance" from $\boldsymbol{x}$ to the set of signals that are $k$-sparse in the dictionary $\boldsymbol{D}$, except that the actual "distance" being measured is a mixed $\ell_2/\ell_1$ norm. In cases where one expects this distance to be small, (11) guarantees that the recovery error will be small.

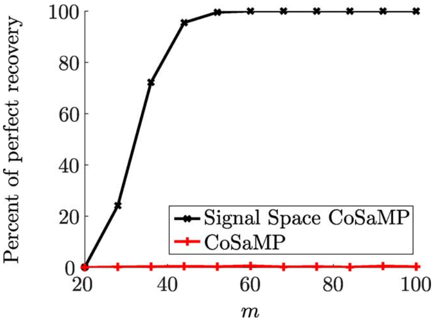


Fig. 1. Performance in recovering signals having a $k = 8$ sparse representation in a dictionary $\boldsymbol{D}$ with orthogonal, but not normalized, columns. The plot shows, for various numbers of measurements $m$, the percent of trials in which each algorithm recovers the signal exactly. Signal space CoSaMP (in which we can compute optimal projections) outperforms an unmodified CoSaMP algorithm.

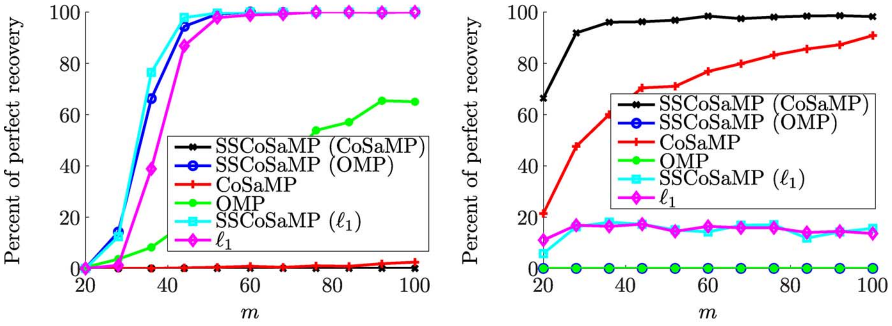
We close this discussion by noting that if we make the stronger assumption that $\boldsymbol{A}\boldsymbol{D}$ actually satisfies the RIP, we can also measure the model mismatch in the coefficient space rather than the signal space. Let $\boldsymbol{\alpha} \in \mathbb{C}^d$ be any vector that satisfies $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$. Then using a natural extension of Proposition II.1, we conclude that

$$\begin{aligned}\left\|\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}_k)\right\|_2 &\\ &= \left\|\boldsymbol{A}\boldsymbol{D}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_k)\right\|_2\\ &\le \sqrt{1+\delta_k}\left(\left\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_k\right\|_2 + \frac{1}{\sqrt{k}}\left\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_k\right\|_1\right).\end{aligned}$$

When $\boldsymbol{\alpha}$ is compressible, the recovery error (8) will be reasonably small.

## III. Simulations

As we discussed in Section I-C, the main difficulty in implementing our algorithm is in computing projections of vectors in the signal space onto the model family of candidate sparse signals. One such projection is required in the *identify* step in Algorithm 1; another such projection is required in the *update* step. Although our theoretical analysis can accommodate near-optimal support estimates $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ that satisfy the condition (4), computing even near-optimal supports can be a challenging task for many dictionaries of practical interest. In this section, we present simulation results using practical (but heuristic) methods for attempting to find near-optimal supports $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$. Specifically, we find ourselves in a situation that mirrors the early days of the sparse recovery literature—we would like to identify a sparse vector that well approximates $\boldsymbol{z}$. This is precisely the scenario where recovery algorithms like OMP and $\ell_1$-minimization were first proposed, so despite the lack of a theoretical guarantee, we can still apply these algorithms. Of course, if we are leaving the solid ground of theory and entering the world of heuristics, we can also just consider applying standard algorithms like OMP, CoSaMP, and $\ell_1$-minimization algorithms directly to the CS recovery problem to see how they perform. We will see, however, that the "signal space CoSaMP" algorithms resulting from using



(a) (b)

Fig. 2. Performance in recovering signals having a $k = 8$ sparse representation in a 4 × overcomplete DFT dictionary. Two scenarios are shown: (a) one in which the $k = 8$ nonzero entries of $\boldsymbol{\alpha}$ are randomly positioned but well separated, and (b) one in which the $k = 8$ nonzero entries all cluster together in a single, randomly positioned block. Algorithms involving OMP and $\ell_1$-minimization perform well in the former scenario; algorithms involving CoSaMP perform well in the latter. In general, the Signal Space CoSaMP variants outperform the corresponding traditional CS algorithm.

standard solvers for $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$—even though they are not quite covered by our theory—can nevertheless outperform these classical CS reconstruction techniques.

In all simulations that follow, we set the signal length $n = 256$. We let $\boldsymbol{D}$ be an $n \times d$ dictionary (two different dictionaries are considered below), and we construct a length-$d$ coefficient vector $\boldsymbol{\alpha}$ with $k = 8$ nonzero entries chosen as i.i.d. Gaussian random variables. We set $\boldsymbol{x} = \boldsymbol{D\alpha}$, construct $\boldsymbol{A}$ as a random $m \times n$ matrix with i.i.d. Gaussian entries, and collect noiseless measurements $\boldsymbol{y} = \boldsymbol{Ax}$. After reconstructing an estimate of $\boldsymbol{x}$, we declare this recovery to be perfect if the SNR of the recovered signal estimate is above 100 dB. All of our simulations were performed via a MATLAB software package that we have made available for download at http://users.ece.gatech.edu/~mdavenport/software.

### A. Renormalized Orthogonal Dictionary

As an instructive warm-up, we begin with one almost trivial example where the optimal projection can be computed exactly: we construct $\boldsymbol{D}$ by taking an orthobasis and renormalizing its columns while maintaining their orthogonality. To compute $\Lambda_{\mathrm{opt}}(\boldsymbol{z}, k)$ with such a dictionary, one merely computes $\boldsymbol{D}^*\boldsymbol{z}$, divides this vector elementwise by the column norms of $\boldsymbol{D}$, and sets $\Lambda$ equal to the positions of the $k$ largest entries.

For the sake of demonstration, we set $\boldsymbol{D}$ equal to the $n \times n$ identity matrix, but we then rescale its first $n/2$ diagonal entries to equal 100 instead of 1. We construct sparse coefficient vectors $\boldsymbol{\alpha}$ as described above, with supports chosen uniformly at random. As a function of the number of measurements $m$, we plot in Fig. 1 the percent of trials in which signal space CoSaMP recovers $\boldsymbol{x}$ exactly. We see that with roughly 50 or more measurements, the recovery is perfect in virtually all trials. In contrast, this figure also shows the performance of traditional CoSaMP using the combined dictionary $\boldsymbol{AD}$ to first recover $\boldsymbol{\alpha}$. Because of the nonnormalized columns in $\boldsymbol{D}$, CoSaMP almost never recovers the correct signal; in fact its support estimates almost always are contained in the set $\{1, 2, \ldots, n/2\}$. Of course, if presented with this problem in practice one would naturally want to modify traditional CoSaMP to account for the various column norms in $\boldsymbol{D}$; the point here is merely that our algorithm gives a principled way to make this (trivial) modification.

### B. Overcomplete DFT Dictionary

As a second example, we set $d = 4n$ and let $\boldsymbol{D}$ be a 4 × overcomplete DFT dictionary. In this dictionary, neighboring columns are highly coherent, while distant columns are not. We consider two scenarios: one in which $k = 8$ nonzero entries of $\boldsymbol{\alpha}$ are randomly positioned but well separated (with a minimum spacing of eight zeros in between any pair of nonzero entries), and one in which the $k = 8$ nonzero entries all cluster together in a single, randomly positioned block. Because of the nature of the columns in $\boldsymbol{D}$, we see that many recovery algorithms perform differently in these two scenarios.

*1) Well-Separated Coefficients:* Fig. 2(a) plots the performance of six different recovery algorithms for the scenario where the nonzero entries of $\boldsymbol{\alpha}$ are well separated. Two of these algorithms are the traditional OMP and CoSaMP algorithms from CS, each using the combined dictionary $\boldsymbol{AD}$ to first recover $\boldsymbol{\alpha}$. We actually see that OMP performs substantially better than CoSaMP in this scenario, apparently because it can select one coefficient at a time and is less affected by the coherence of $\boldsymbol{D}$. It is somewhat remarkable that OMP succeeds at all, given that $\boldsymbol{AD}$ will not satisfy the RIP and we are not aware of any existing theory that would guarantee the performance of OMP in this scenario.

We also show in Fig. 2(a) two variants of signal space CoSaMP: one in which OMP is used for computing $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ (labeled "SSCoSaMP (OMP)"), and one in which CoSaMP is used for computing $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ (labeled "SSCoSaMP (CoSaMP)"). That is, these algorithms actually use OMP or CoSaMP as an inner loop inside of signal space CoSaMP to find a sparse solution to the equation $\boldsymbol{z} = \boldsymbol{D\alpha}$. In this scenario, we see that the performance of SSCoSaMP (OMP) is substantially better than OMP, while the performance of SSCoSaMP (CoSaMP) is poor. We believe that this happens for the same reason that traditional OMP outperforms traditional CoSaMP. In general, we have found that when OMP performs well, SSCoSaMP (OMP) may perform even better, and when CoSaMP performs poorly, SSCoSaMP (CoSaMP) may still perform poorly.

Fig. 2(a) also shows the performance of two algorithms that involve convex optimization for sparse regularization. One, labeled "$\ell_1$," uses an $\ell_1$-minimization approach [27] to find a sparse coefficient vector $\boldsymbol{\alpha}'$ subject to the constraint that $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{D}\boldsymbol{\alpha}'$. This algorithm actually outperforms traditional OMP in this scenario. The other, labeled "SSCoSaMP ($\ell_1$)," is a variant of signal space CoSaMP in which $\ell_1$-minimization is used for computing $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$.[4] Specifically, to compute $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$, we search for the vector $\boldsymbol{\alpha}'$ having the smallest $\ell_1$ norm subject to the constraint that $\boldsymbol{z} = \boldsymbol{D}\boldsymbol{\alpha}'$, and we then choose the support that contains the $k$ largest entries of this vector. Remarkably, this algorithm performs best of all. We believe that this is likely due to the fact that, for the overcomplete DFT dictionary, $\ell_1$-minimization is known to be capable of finding $\Lambda_{\rm opt}(\boldsymbol{z}, k)$ exactly when $\boldsymbol{z} = \mathcal{P}_{\Lambda_{\rm opt}(\boldsymbol{z},k)}\boldsymbol{z}$ and the entries of $\Lambda_{\rm opt}(\boldsymbol{z}, k)$ are sufficiently well separated [7]. While we do not guarantee that this condition will be met within every iteration of signal space CoSaMP, the fact that the original coefficient vector $\boldsymbol{\alpha}$ has well-separated coefficients seems to be intimately related to the success of $\ell_1$ and SSCoSaMP ($\ell_1$) here.

We note that all of the above algorithms involve a step where a least-squares problem must be solved on an estimated support set.[5] This might seem somewhat contrary to our signal-focused approach, since solving this least-squares problem essentially involves recovering a set of coefficients $\boldsymbol{\alpha}$ and then computing $\widetilde{\boldsymbol{x}} = \boldsymbol{D}_T\boldsymbol{\alpha}$. However, recall that at this point in the algorithm we are solving an overdetermined system, so there is no significant difference between solving for $\boldsymbol{\alpha}$ versus $\widetilde{\boldsymbol{x}}$. The problem with coefficient-focused strategies and analysis is that they rely on identifying the subset $T$ containing the "correct" subset of coefficients, whereas for us, $T$ could be very different from the "correct" subset, and the vector $\boldsymbol{\alpha}$ has no particular significance—all that matters is the vector $\widetilde{\boldsymbol{x}}$ that this step ultimately synthesizes. Nevertheless, it is worth noting that when a dictionary $\boldsymbol{D}$ is highly coherent, it can be numerically challenging to solve this problem, as the resulting submatrix can be very poorly conditioned. Following our discussion in [11], we employ *Tikhonov regularization* [15], [23], [25], [26] and solve a norm-constrained least-squares problem to improve its conditioning. For this we must provide our algorithms with an upper bound on the norm of the sparse coefficient vector. In the simulations above, we have selected this bound to be 10 × the true norm of the original $\boldsymbol{\alpha}$. The selection of this bound does not have a substantial impact on the performance of OMP or SSCoSaMP (OMP), but we have noticed that CoSaMP and SSCoSaMP (CoSaMP) perform somewhat better when the true norm $\|\boldsymbol{\alpha}\|_2$ is provided as an oracle.

*2) Clustered Coefficients:* Fig. 2(b) plots the performance of the same six recovery algorithms for the scenario where the nonzero entries of $\boldsymbol{\alpha}$ are clustered into a single block. Although one could of course employ a block-sparse recovery algorithm in this scenario, our intent is more to study the impact that neighboring active atoms have on the algorithms above.

In this scenario, between the traditional greedy algorithms, CoSaMP now outperforms OMP, apparently because it is designed to select multiple indices at each step and will not be as affected by the coherence of neighboring active columns in $\boldsymbol{D}$. We also see that the performance of SSCoSaMP (CoSaMP) is somewhat better than CoSaMP, while the performance of SSCoSaMP (OMP) is poor. We believe that this happens for the same reason that traditional CoSaMP outperforms traditional OMP. In general, we have found that when CoSaMP performs well, SSCoSaMP (CoSaMP) may perform even better, and when OMP performs poorly, SSCoSaMP (OMP) may still perform poorly.

In terms of our condition for perfect recovery (estimating $\boldsymbol{x}$ to within an SNR of 100 dB or more), neither of the algorithms that involve $\ell_1$-minimization perform well in this scenario. However, we do note that both $\ell_1$ and SSCoSaMP ($\ell_1$) do frequently recover an estimate of $\boldsymbol{x}$ with an SNR of 50 dB or more, though still not quite as frequently as SSCoSaMP (CoSaMP) does.

In these simulations, we again use Tikhonov regularization with a norm bound that is 10 × the true norm of the original $\boldsymbol{\alpha}$. However, we have not found that changing this norm bound has a significant impact on CoSaMP or SSCoSaMP (CoSaMP) in this scenario. We also note that in this scenario we found it beneficial to run CoSaMP and the three signal space CoSaMP methods for a few more iterations than in the case of well-separated coefficients; convergence to exactly the correct support can be slow in this case where multiple neighboring atoms in a coherent dictionary are active.

*3) Additional Investigations:* We close with some final remarks concerning additional investigations. First, our simulations above have tested three heuristic methods for attempting to find near-optimal supports $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$, and we have evaluated the performance of these methods based on the ultimate success or failure of SSCoSaMP in recovering the signal. In some problems of much smaller dimension (where we could use an exhaustive search to find $\Lambda_{\rm opt}(\boldsymbol{z}, k)$), we monitored the performance of CoSaMP, OMP, and $\ell_1$-minimization for computing $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$ in terms of the effective $\epsilon_1$ and $\epsilon_2$ values they attained according to the metric in (4). For scenarios where the nonzero entries of $\boldsymbol{\alpha}$ were well separated, we observed typical $\epsilon_1$ and $\epsilon_2$ values for OMP and $\ell_1$-minimization on the order of 1 or less. For CoSaMP, these values were larger by one or two orders of magnitude, as might be expected based on the signal recovery results presented in Section III-B.1.[6] For scenarios where the nonzero entries of $\boldsymbol{\alpha}$ were clustered, the $\epsilon_2$ values for OMP and $\ell_1$-minimization increased by about one order of magnitude, but the $\epsilon_1$ and $\epsilon_2$ values for CoSaMP did not change significantly. The primary reason for this, despite the superior signal recovery performance of SSCoSaMP (CoSaMP) in Section III-B.2, appears to be that even when the nonzero entries of $\boldsymbol{\alpha}$ are clustered, the support of the optimal approximation of $\widetilde{\boldsymbol{v}}$ in the *identify* step will not necessarily be clustered, and so CoSaMP will struggle to accurately identify this support.

Second, we remark that our simulations in Sections III-B.1 and III-B.2 have tested two extremes: one scenario in which the nonzero entries of $\boldsymbol{\alpha}$ were well separated, and one scenario in which the nonzero entries clustered together in a single block. Among the heuristic methods that we have used for attempting to find near-optimal supports $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z}, k)$, the question of which

[4] We are not unaware of the irony of using $\ell_1$-minimization inside of a greedy algorithm.

[5] We use this for debiasing after running $\ell_1$.

[6] The occasional exception in some of these simulations occurred when it happened that $\left\|\boldsymbol{z} - \mathcal{P}_{\Lambda_{\rm opt}(\boldsymbol{z},k)}\boldsymbol{z}\right\|_2 \approx 0$ but $\left\|\mathcal{P}_{\Lambda_{\rm opt}(\boldsymbol{z},k)}\boldsymbol{z} - \mathcal{P}_{\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z},k)}\boldsymbol{z}\right\|_2$ was not correspondingly small, and so the effective $\epsilon_2$ value was large or infinite.

method performs best has been shown to depend on the sparsity pattern of $\boldsymbol{\alpha}$. Although we do not present detailed results here, we have also tested these same algorithms using a hybrid sparsity model for $\boldsymbol{\alpha}$ in which half of the nonzero entries are well separated while the other half are clustered. As one might expect based on the discussions above, all three of the SS-CoSaMP methods struggle in this scenario (as do the three standard CS methods). This is yet another reminder that more work is needed to understand what techniques are appropriate for approximating $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{z},k)$ and how to optimize these techniques depending on what is known about the signal's sparsity pattern.

## Appendix
## Proof of Theorem II.1

The proof of Theorem II.1 requires four main lemmas, which are listed below and proved in Sections A–D. In the lemmas below, $\boldsymbol{v}=\boldsymbol{x}-\boldsymbol{x}^{\ell}$ denotes the recovery error in signal space after $\ell$ iterations.

*Lemma A.1 (Identify):*

$$\|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 \le ((2+\epsilon_1)\delta_{4k}+\epsilon_1)\|\boldsymbol{v}\|_2 + (2+\epsilon_1)\sqrt{1+\delta_{4k}}\,\|\boldsymbol{e}\|_2 .$$

*Lemma A.2 (Merge):*

$$\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 \le \|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 .$$

*Lemma A.3 (Update):*

$$\|\boldsymbol{x}-\widetilde{\boldsymbol{x}}\|_2 \le \sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}\,\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 + \frac{2}{\sqrt{1-\delta_{4k}}}\,\|\boldsymbol{e}\|_2 .$$

*Lemma A.4 (Estimate):*

$$\left\|\boldsymbol{x}-\boldsymbol{x}^{\ell+1}\right\|_2 \le (2+\epsilon_2)\,\|\boldsymbol{x}-\widetilde{\boldsymbol{x}}\|_2 .$$

Combining all four statements above, we have

$$\begin{aligned}
&\left\|\boldsymbol{x}-\boldsymbol{x}^{\ell+1}\right\|_2\\
&\quad\le (2+\epsilon_2)\,\|\boldsymbol{x}-\widetilde{\boldsymbol{x}}\|_2\\
&\quad\le (2+\epsilon_2)\sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}\,\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 + \frac{4+2\epsilon_2}{\sqrt{1-\delta_{4k}}}\,\|\boldsymbol{e}\|_2\\
&\quad\le (2+\epsilon_2)\sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}\,\|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 + \frac{4+2\epsilon_2}{\sqrt{1-\delta_{4k}}}\,\|\boldsymbol{e}\|_2\\
&\quad\le (2+\epsilon_2)\sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}((2+\epsilon_1)\delta_{4k}+\epsilon_1)\,\|\boldsymbol{v}\|_2\\
&\qquad\quad + (2+\epsilon_2)\sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}(2+\epsilon_1)\sqrt{1+\delta_{4k}}\,\|\boldsymbol{e}\|_2\\
&\qquad\quad + \frac{4+2\epsilon_2}{\sqrt{1-\delta_{4k}}}\,\|\boldsymbol{e}\|_2\\
&\quad= ((2+\epsilon_1)\delta_{4k}+\epsilon_1)(2+\epsilon_2)\sqrt{\frac{1+\delta_{4k}}{1-\delta_{4k}}}\,\left\|\boldsymbol{x}-\boldsymbol{x}^{\ell}\right\|_2\\
&\qquad\quad + \left(\frac{(2+\epsilon_2)\left((2+\epsilon_1)(1+\delta_{4k})+2\right)}{\sqrt{1-\delta_{4k}}}\right)\|\boldsymbol{e}\|_2 .
\end{aligned}$$

This completes the proof of Theorem II.1.

*A) Proof of Lemma A.1:* In order to prove the four main lemmas, we require two supplemental lemmas, the first of which is a direct consequence of the $\boldsymbol{D}$-RIP.

*Lemma A.5 (Consequence of $\boldsymbol{D}$-RIP):* For any index set $B$ and any vector $\boldsymbol{z}\in\mathbb{C}^n$,

$$\|\mathcal{P}_B\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_B\boldsymbol{z}-\mathcal{P}_B\boldsymbol{z}\|_2 \le \delta_{|B|}\,\|\boldsymbol{z}\|_2 .$$

*Proof:* We have

$$\begin{aligned}
\delta_{|B|} &\ge \sup_{\boldsymbol{x}:|B|-\text{sparse in }\boldsymbol{D}} \frac{\left|\,\|\boldsymbol{A}\boldsymbol{x}\|_2^2-\|\boldsymbol{x}\|_2^2\,\right|}{\|\boldsymbol{x}\|_2^2}\\
&\ge \sup_{\boldsymbol{x}} \frac{\left|\,\|\boldsymbol{A}\mathcal{P}_B\boldsymbol{x}\|_2^2-\|\mathcal{P}_B\boldsymbol{x}\|_2^2\,\right|}{\|\mathcal{P}_B\boldsymbol{x}\|_2^2}\\
&\ge \sup_{\boldsymbol{x}} \frac{\left|\,\|\boldsymbol{A}\mathcal{P}_B\boldsymbol{x}\|_2^2-\|\mathcal{P}_B\boldsymbol{x}\|_2^2\,\right|}{\|\boldsymbol{x}\|_2^2}\\
&= \sup_{\|\boldsymbol{x}\|_2=1} \left|\,\|\boldsymbol{A}\mathcal{P}_B\boldsymbol{x}\|_2^2-\|\mathcal{P}_B\boldsymbol{x}\|_2^2\,\right|\\
&= \sup_{\|\boldsymbol{x}\|_2=1} \left|\langle\mathcal{P}_B^*\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_B\boldsymbol{x}-\mathcal{P}_B^*\mathcal{P}_B\boldsymbol{x},\boldsymbol{x}\rangle\right|\\
&= \sup_{\|\boldsymbol{x}\|_2=1} \left|\langle\mathcal{P}_B\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_B\boldsymbol{x}-\mathcal{P}_B\boldsymbol{x},\boldsymbol{x}\rangle\right|\\
&= \|\mathcal{P}_B\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_B-\mathcal{P}_B\|_2 ,
\end{aligned}$$

where the third line follows because $\|\mathcal{P}_B\boldsymbol{x}\|_2\le\|\boldsymbol{x}\|_2$ for all $\boldsymbol{x}$, and the last line follows from the fact that $\mathcal{P}_B\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_B-\mathcal{P}_B$ is self-adjoint. □

We will also utilize an elementary fact about orthogonal projections.

*Lemma A.6:* For any pair of index sets $A,B$ with $A\subset B$, $\mathcal{P}_A=\mathcal{P}_A\mathcal{P}_B$.

Now, to make the notation simpler, note that $\widetilde{\boldsymbol{v}}=\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v}+\boldsymbol{A}^*\boldsymbol{e}$ and that $\Omega=\mathcal{S}_{\boldsymbol{D}}(\widetilde{\boldsymbol{v}},2k)$. Let $\Omega^*=\Lambda_{\text{opt}}(\widetilde{\boldsymbol{v}},2k)$ denote the optimal support of size $2k$ for approximating $\widetilde{\boldsymbol{v}}$, and set $R=\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{v},2k)$. Using this notation we have

$$\begin{aligned}
&\|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2\\
&\quad= \|\boldsymbol{v}-\mathcal{P}_\Omega\boldsymbol{v}\|_2\\
&\quad\le \|\boldsymbol{v}-\mathcal{P}_\Omega\widetilde{\boldsymbol{v}}\|_2\\
&\quad= \|(\boldsymbol{v}-\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}})+(\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}})\\
&\qquad\quad +(\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_\Omega\widetilde{\boldsymbol{v}})\|_2\\
&\quad\le \|\boldsymbol{v}-\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}\|_2+\|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}\|_2\\
&\qquad\quad +\|\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_\Omega\widetilde{\boldsymbol{v}}\|_2\\
&\quad\le \|\boldsymbol{v}-\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v}\|_2+\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2\\
&\qquad\quad +\|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}\|_2+\|\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_\Omega\widetilde{\boldsymbol{v}}\|_2 ,
\end{aligned}\tag{12}$$

where the second line follows from the fact that $\mathcal{P}_\Omega\boldsymbol{v}$ is the nearest neighbor to $\boldsymbol{v}$ among all vectors in $\mathcal{R}(\boldsymbol{D}_\Omega)$ and the fourth and fifth lines use the triangle inequality.

Below, we will provide bounds on the first and second terms appearing in (12). To deal with the third term in (12), note that for any $\Pi$ which is a subset of $R\cup\Omega^*$, we can write

$$\widetilde{\boldsymbol{v}}-\mathcal{P}_\Pi\widetilde{\boldsymbol{v}}=(\widetilde{\boldsymbol{v}}-\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}})+(\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}-\mathcal{P}_\Pi\widetilde{\boldsymbol{v}}),$$

where $\widetilde{\boldsymbol{v}} - \mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}$ is orthogonal to $\mathcal{R}(\boldsymbol{D}_{R\cup\Omega^*})$, and $\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Pi}\widetilde{\boldsymbol{v}}$ is contained in $\mathcal{R}(\boldsymbol{D}_{R\cup\Omega^*})$. Thus, we can write

$$\|\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Pi}\widetilde{\boldsymbol{v}}\|_2^2 = \|\widetilde{\boldsymbol{v}} - \mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}\|_2^2 + \|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Pi}\widetilde{\boldsymbol{v}}\|_2^2 .$$

Recall that over all index sets $\Pi$ with $|\Pi| = 2k$, $\|\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Pi}\widetilde{\boldsymbol{v}}\|_2$ is minimized by choosing $\Pi = \Omega^*$. Thus, over all $\Pi$ which are subsets of $R \cup \Omega^*$ with $|\Pi| = 2k$, $\|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Pi}\widetilde{\boldsymbol{v}}\|_2^2$ must be minimized by choosing $\Pi = \Omega^*$. In particular, we have the first inequality below

$$\begin{aligned}
&\|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}\|_2 \\
&\quad \le \|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{R}\widetilde{\boldsymbol{v}}\|_2 \\
&\quad = \|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{R}\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}\|_2 \\
&\quad \le \|\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{R}(\boldsymbol{v} + \boldsymbol{A}^*\boldsymbol{e})\|_2 \\
&\quad = \|(\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}) + (\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e} - \mathcal{P}_{R}\boldsymbol{A}^*\boldsymbol{e})\|_2 \\
&\quad \le \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 + \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e} - \mathcal{P}_{R}\boldsymbol{A}^*\boldsymbol{e}\|_2 \\
&\quad \le \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 + \|(I - \mathcal{P}_{R})\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 \\
&\quad \le \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 + \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 . \qquad (13)
\end{aligned}$$

The second line above uses Lemma A.6, the third line follows from the fact that $\mathcal{P}_{R}\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}$ must be the nearest neighbor to $\mathcal{P}_{R\cup\Omega^*}\widetilde{\boldsymbol{v}}$ among all vectors in $\mathcal{R}(\boldsymbol{D}_R)$, the fourth line uses the fact that $\mathcal{P}_{R}\boldsymbol{v} = \boldsymbol{v}$ because $R = \mathcal{S}_{\boldsymbol{D}}(\boldsymbol{v}, 2k)$ and both $\boldsymbol{x}$ and $\boldsymbol{x}^\ell$ are $k$-sparse in $\boldsymbol{D}$, the fifth line uses the triangle inequality, the sixth line uses Lemma A.6, and the seventh line follows from the fact that $(I - \mathcal{P}_R)$ is an orthogonal projection and hence has norm bounded by 1.

To deal with the fourth term in (12), note that from the definition of $\Omega^*$ and from (4), we have the first inequality below

$$\begin{aligned}
&\|\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}} - \mathcal{P}_{\Omega}\widetilde{\boldsymbol{v}}\|_2 \\
&\quad \le \epsilon_1 \|\mathcal{P}_{\Omega^*}\widetilde{\boldsymbol{v}}\|_2 \\
&\quad = \epsilon_1 \|\mathcal{P}_{\Omega^*}(\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} + \boldsymbol{A}^*\boldsymbol{e})\|_2 \\
&\quad \le \epsilon_1 \|\mathcal{P}_{\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v}\|_2 + \epsilon_1 \|\mathcal{P}_{\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 \\
&\quad = \epsilon_1 \|\mathcal{P}_{\Omega^*}\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v}\|_2 + \epsilon_1 \|\mathcal{P}_{\Omega^*}\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 \\
&\quad \le \epsilon_1 \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v}\|_2 + \epsilon_1 \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 \\
&\quad \le \epsilon_1 \|\boldsymbol{v}\|_2 + \epsilon_1 \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 \\
&\qquad + \epsilon_1 \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 . \qquad (14)
\end{aligned}$$

The third line above uses the triangle inequality, the fourth line uses Lemma A.6, the fifth line uses the fact $\mathcal{P}_{\Omega^*}$ is an orthogonal projection and hence has norm bounded by 1, and the sixth line uses the triangle inequality.

Combining (12), (13), and (14) we see that

$$\begin{aligned}
\|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 &\le (2+\epsilon_1)\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 \\
&\quad + (2+\epsilon_1)\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{e}\|_2 + \epsilon_1\|\boldsymbol{v}\|_2 .
\end{aligned}$$

Since $\boldsymbol{v} \in \mathcal{R}(\boldsymbol{D}_R)$, it follows that $\boldsymbol{v} \in \mathcal{R}(\boldsymbol{D}_{R\cup\Omega^*})$, and so

$$\begin{aligned}
\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{v} - \boldsymbol{v}\|_2 &= \|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\boldsymbol{A}\mathcal{P}_{R\cup\Omega^*}\boldsymbol{v} - \mathcal{P}_{R\cup\Omega^*}\boldsymbol{v}\|_2 \\
&\le \delta_{4k}\|\boldsymbol{v}\|_2 ,
\end{aligned}$$

where we have used Lemma A.5 to get the inequality above. In addition, we know that the operator norm of $\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*$ satisfies

$$\begin{aligned}
\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*\|_2 &= \|(\mathcal{P}_{R\cup\Omega^*}\boldsymbol{A}^*)^*\|_2 \\
&= \|\boldsymbol{A}\mathcal{P}_{R\cup\Omega^*}\|_2 \\
&\le \sqrt{1+\delta_{4k}},
\end{aligned}$$

which follows from the $\boldsymbol{D}$-RIP. Specifically, for any $\boldsymbol{x}$,

$$\frac{\|\boldsymbol{A}\mathcal{P}_{R\cup\Omega^*}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} \le \frac{\|\boldsymbol{A}\mathcal{P}_{R\cup\Omega^*}\boldsymbol{x}\|_2}{\|\mathcal{P}_{R\cup\Omega^*}\boldsymbol{x}\|_2} \le \sqrt{1+\delta_{4k}}.$$

Putting all of this together, we have

$$\|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 \le ((2+\epsilon_1)\delta_{4k} + \epsilon_1)\|\boldsymbol{v}\|_2 + (2+\epsilon_1)\sqrt{1+\delta_{4k}}\,\|\boldsymbol{e}\|_2 .$$

*B) Proof of Lemma A.2:* First note that by the definition of $T$, $\boldsymbol{x}^\ell \in \mathcal{R}(\boldsymbol{D}_T)$, and hence $\mathcal{P}_{T^\perp}\boldsymbol{x}^\ell = 0$. Thus, we can write

$$\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 = \left\|\mathcal{P}_{T^\perp}(\boldsymbol{x} - \boldsymbol{x}^\ell)\right\|_2 = \|\mathcal{P}_{T^\perp}\boldsymbol{v}\|_2 .$$

Finally, since $\Omega \subseteq T$, we have that

$$\|\mathcal{P}_{T^\perp}\boldsymbol{v}\|_2 \le \|\mathcal{P}_{\Omega^\perp}\boldsymbol{v}\|_2 .$$

*C) Proof of Lemma A.3:* To begin, we note that $\boldsymbol{x} - \widetilde{\boldsymbol{x}}$ has a $4k$-sparse representation in $\boldsymbol{D}$, thus, applying the $\boldsymbol{D}$-RIP (of order $4k$) we have

$$\|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 \le \frac{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\widetilde{\boldsymbol{x}}\|_2}{\sqrt{1-\delta_{4k}}}.$$

By construction,

$$\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\widetilde{\boldsymbol{x}} + \boldsymbol{e}\|_2 \le \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\boldsymbol{z} + \boldsymbol{e}\|_2$$

for any $\boldsymbol{z} \in \mathcal{R}(\boldsymbol{D}_T)$, in particular for $\boldsymbol{z} = \mathcal{P}_T\boldsymbol{x}$. Thus,

$$\begin{aligned}
\|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 &\le \frac{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\widetilde{\boldsymbol{x}}\|_2}{\sqrt{1-\delta_{4k}}} \\
&\le \frac{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\widetilde{\boldsymbol{x}} + \boldsymbol{e}\|_2 + \|\boldsymbol{e}\|_2}{\sqrt{1-\delta_{4k}}} \\
&\le \frac{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\mathcal{P}_T\boldsymbol{x} + \boldsymbol{e}\|_2 + \|\boldsymbol{e}\|_2}{\sqrt{1-\delta_{4k}}} \\
&\le \frac{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\mathcal{P}_T\boldsymbol{x}\|_2 + 2\|\boldsymbol{e}\|_2}{\sqrt{1-\delta_{4k}}}
\end{aligned}$$

where the second and fourth lines use the triangle inequality. By applying the $\boldsymbol{D}$-RIP, we obtain

$$\begin{aligned}
\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{A}\mathcal{P}_T\boldsymbol{x}\|_2 &\le \sqrt{1+\delta_{4k}}\,\|\boldsymbol{x} - \mathcal{P}_T\boldsymbol{x}\|_2 \\
&= \sqrt{1+\delta_{4k}}\,\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 .
\end{aligned}$$

Combining all of this,

$$\|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 \le \frac{\sqrt{1+\delta_{4k}}\,\|\mathcal{P}_{T^\perp}\boldsymbol{x}\|_2 + 2\|\boldsymbol{e}\|_2}{\sqrt{1-\delta_{4k}}}.$$

*D) Proof of Lemma A.4:* Using the triangle inequality, we have

$$\begin{aligned}
\left\|\boldsymbol{x} - \boldsymbol{x}^{\ell+1}\right\|_2 &= \left\|\boldsymbol{x} - \widetilde{\boldsymbol{x}} + \widetilde{\boldsymbol{x}} - \boldsymbol{x}^{\ell+1}\right\|_2 \\
&\le \|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 + \left\|\widetilde{\boldsymbol{x}} - \boldsymbol{x}^{\ell+1}\right\|_2 .
\end{aligned}$$

Recall that $\Gamma = \mathcal{S}_{\boldsymbol{D}}(\widetilde{\boldsymbol{x}}, k)$ and $\boldsymbol{x}^{\ell+1} = \mathcal{P}_{\Gamma}\widetilde{\boldsymbol{x}}$. Let $\Gamma^* = \Lambda_{\mathrm{opt}}(\widetilde{\boldsymbol{x}}, k)$ denote the optimal support of size $k$ for approximating $\widetilde{\boldsymbol{x}}$. Then, we can write

$$\begin{aligned}\left\|\widetilde{\boldsymbol{x}} - \boldsymbol{x}^{\ell+1}\right\|_2 &\le \|\widetilde{\boldsymbol{x}} - \mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}}\|_2 + \|\mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}} - \mathcal{P}_{\Gamma}\widetilde{\boldsymbol{x}}\|_2 \\ &\le \|\widetilde{\boldsymbol{x}} - \mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}}\|_2 + \epsilon_2 \|\widetilde{\boldsymbol{x}} - \mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}}\|_2 ,\end{aligned}$$

where the first line follows from the triangle inequality, and the second line uses (4). Combining all of this, we have

$$\begin{aligned}\left\|\boldsymbol{x} - \boldsymbol{x}^{\ell+1}\right\|_2 &\le \|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 + (1+\epsilon_2)\|\widetilde{\boldsymbol{x}} - \mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}}\|_2 \\ &\le \|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 + (1+\epsilon_2)\|\widetilde{\boldsymbol{x}} - \boldsymbol{x}\|_2 \\ &= (2+\epsilon_2)\|\boldsymbol{x} - \widetilde{\boldsymbol{x}}\|_2 ,\end{aligned}$$

where the second line follows from the fact that $\mathcal{P}_{\Gamma^*}\widetilde{\boldsymbol{x}}$ is the nearest neighbor to $\widetilde{\boldsymbol{x}}$ among all vectors having a $k$-sparse representation in $\boldsymbol{D}$.

## References

[1] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.

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

[3] T. Blumensath, "Sampling and reconstructing signals from a union of linear subspaces," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4660–4671, Jul. 2011.

[4] T. Blumensath and M. Davies, "Iterative hard thresholding for compressive sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, 2009.

[5] E. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus de l'Académie des Sciences, Série I*, vol. 346, no. 9-10, pp. 589–592, 2008.

[6] E. Candès, Y. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Appl. Comput. Harmon. Anal.*, vol. 31, no. 1, pp. 59–73, 2011.

[7] E. Candès and C. Fernandez-Granda, "Towards a Mathematical Theory of Super-Resolution," to appear in *Comm. Pure Appl. Math.*, 2013 [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/cpa.21455/abstract

[8] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[9] V. Cevher, "An ALPS view of sparse recovery," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 5808–5811.

[10] M. Davenport and M. Wakin, "Analysis of orthogonal matching pursuit using the restricted isometry property," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4395–4401, Sept. 2010.

[11] M. Davenport and M. Wakin, "Compressive sensing of analog signals using discrete prolate spheroidal sequences," *Appl. Comput. Harmon. Anal.*, vol. 33, no. 3, pp. 438–472, 2012.

[12] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[13] A. Garnaev and E. Gluskin, "On widths of the euclidean ball," *Sov. Math. Dokl.*, vol. 30, pp. 200–204, 1984.

[14] R. Giryes, S. Nam, M. Elad, R. Gribonval, and M. Davies, "Greedy-Like Algorithms for the Cosparse Analysis Model," to appear in *Linear Alg. Appl.*, 2013 [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0024379513001870

[15] P. Hansen, "Regularization tools version 4.0 for Matlab 7.3," *Numer. Algorithms*, vol. 46, pp. 189–194, 2007.

[16] B. Kashin, "The widths of certain finite dimensional sets and classes of smooth functions," *Izvestia*, vol. 41, pp. 334–351, 1977.

[17] F. Krahmer and R. Ward, "New and improved Johnson Lindenstrauss embeddings via the restricted isometry property," *SIAM J. Math. Anal.*, vol. 43, no. 3, pp. 1269–1281, 2011.

[18] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[19] S. Mendelson, A. Pajor, and N. Tomczak-Jaegermann, "Uniform uncertainty principle for Bernoulli and subgaussian ensembles," *Constructive Approx.*, vol. 28, no. 3, pp. 277–289, 2008.

[20] S. Nam, M. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Appl. Comput. Harmon. Anal.*, vol. 34, no. 1, pp. 30–56, 2013.

[21] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.

[22] Y. Pati, R. Rezaifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," presented at the Asilomar Conf. Signals, Syst., Comput., Pacific Grove, CA, USA, Nov. 1993.

[23] D. Phillips, "A technique for the numerical solution of certain integral equations of the first kind," *J. ACM*, vol. 9, pp. 84–97, 1962.

[24] M. Rudelson and R. Vershynin, "Sparse reconstruction by convex relaxation: Fourier and Gaussian measurements," presented at the IEEE Conf. Inf. Sci. Syst., Princeton, NJ, USA, Mar. 2006.

[25] A. Tikhonov, "Solution of incorrectly formulated problems and the regularization method," *Dokl. Akad. Nauk. SSSR*, vol. 151, pp. 1035–1038, 1963.

[26] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC, USA: Winston & Sons, 1977.

[27] E. van den Berg and M. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. Sci. Comput.*, vol. 31, no. 2, pp. 890–912, 2008.

[28] T. Zhang, "Sparse recovery with orthogonal matching pursuit under RIP," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 6215–6221, Sep. 2011.

**Mark A. Davenport** (S'01–M'10) is an Assistant Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. Prior to this, he spent 2010–2012 as an NSF Mathematical Sciences Postdoctoral Research Fellow in the Department of Statistics at Stanford University and as a visitor with the Laboratoire Jacques-Louis Lions at the Université Pierre et Marie Curie. He received the B.S.E.E., M.S., and Ph.D. degrees in electrical and computer engineering in 2004, 2007, and 2010, and the B.A. degree in managerial studies in 2004, all from Rice University. His research interests include compressive sensing, low-rank matrix recovery, nonlinear approximation, and the application of low-dimensional signal models in signal processing and machine learning.

Dr. Davenport shared the Hershel M. Rich Invention Award from Rice University, in 2007 for his work on the single-pixel camera and compressive sensing. In 2011, he was awarded the Ralph Budd Thesis Award from Rice.

**Deanna Needell** is an Assistant Professor in the Mathematical Sciences department at Claremont McKenna College. Prior to this, she spent 2009–2011 as a postdoctoral fellow in the Mathematics and Statistics departments at Stanford University. She received her B.S. in Mathematics and Computer Science from the University of Nevada and her M.A. and Ph.D. in Mathematics from the University of California, Davis. Her research interests include Compressed Sensing, Randomized Algorithms, Functional Analysis, Computational Mathematics, Probability, and Statistics. Prof. Needell has been the recipient of awards including the Simons Foundation Collaboration grant, 2012 IEEE Signal Processing Society Young Author Best Paper Award, and ScienceWatch Fast-Breaking paper award.

**Michael B. Wakin** (S'01–M'07) received the B.S. degree in electrical engineering and the B.A. degree in mathematics in 2000 (summa cum laude), the M.S. degree in electrical engineering in 2002, and the Ph.D. degree in electrical engineering in 2007, all from Rice University. He was an NSF Mathematical Sciences Postdoctoral Research Fellow at the California Institute of Technology from 2006–2007 and an Assistant Professor at the University of Michigan in Ann Arbor from 2007–2008. He is now an Associate Professor in the Department of Electrical Engineering and Computer Science at the Colorado School of Mines. His research interests include sparse, geometric, and manifold-based models for signal and image processing, approximation, compression, compressive sensing, and dimensionality reduction. In 2007, Dr. Wakin shared the Hershel M. Rich Invention Award from Rice University for the design of a single-pixel camera based on compressive sensing; in 2008, Dr. Wakin received the DARPA Young Faculty Award for his research in compressive multi-signal processing for environments such as sensor and camera networks; and in 2012, Dr. Wakin received the NSF CAREER Award for research into dimensionality reduction techniques for structured data sets.