

4-1-2014

Guaranteed Sparse Signal Recovery with Highly Coherent Sensing Matrices

Guangliang Chen
Claremont McKenna College

Atul Divekar
Alcatel-Lucent

Deanna Needell
Claremont McKenna College

Recommended Citation

Chen, G., Divekar, A., Needell, D., "Guaranteed sparse signal recovery with highly coherent sensing matrices", *Sampling Theory in Signal Analysis and Image Processing*, To appear, arXiv:1311.0314v2, 2013.

This Article - postprint is brought to you for free and open access by the CMC Faculty Scholarship at Scholarship @ Claremont. It has been accepted for inclusion in CMC Faculty Publications and Research by an authorized administrator of Scholarship @ Claremont. For more information, please contact scholarship@cuc.claremont.edu.

Guaranteed sparse signal recovery with highly coherent sensing matrices

Guangliang Chen

Department of Mathematics, Claremont McKenna College
Claremont, California 91711, USA
gchen@cmc.edu

Atul Divekar

Alcatel-Lucent,
Naperville, Illinois 60563, USA
atul.divekar@alcatel-lucent.com

Deanna Needell

Department of Mathematics, Claremont McKenna College
Claremont, California 91711, USA
dneedell@cmc.edu

April 2, 2014

Abstract

Compressive sensing is a methodology for the reconstruction of sparse or compressible signals using far fewer samples than required by the Nyquist criterion. However, many of the results in compressive sensing concern random sampling matrices such as Gaussian and Bernoulli matrices. In common physically feasible signal acquisition and reconstruction scenarios such as super-resolution of images, the sensing matrix has a non-random structure with highly correlated columns. Here we present a compressive sensing recovery algorithm, called *Partial Inversion (PartInv)*, that shows better performance than existing greedy methods for random matrices, and is especially suitable for matrices that have subsets of highly correlated columns. We provide theoretical justification as well as empirical comparisons.

1 Introduction

Consider the problem of image super-resolution, where one or more low-resolution images of a scene are used to synthesize a single image of higher resolution. If multiple images are used, they are commonly assumed to be subpixel-shifted and downsampled versions of the original high resolution image that is to be reconstructed [1]. Alternatively, super-resolution from a single low resolution image using a dictionary of image patches and compressive sensing recovery has been proposed in [2]. The relationship between the available low resolution and desired high resolution images is commonly modeled by a linear filtering and downsampling operation. Suppose that we wish to reconstruct a size $N \times N$ high resolution image from a lower resolution image, for example of size

$\frac{N}{2} \times \frac{N}{2}$, or smaller. Let x and y represent the vectorized high and low resolution images respectively. We model the formation of y from x by the equation $y = SHx + \eta$ where η is the sensor noise, S is a downsampling matrix of size $\frac{N^2}{4}$ by N^2 , and H is an N^2 by N^2 matrix that represents the filtering (antialiasing) operation. In order to consider super-resolution as a compressive sensing recovery problem we write $x = \Psi c$ where Ψ is a sparsifying basis for the class of images under consideration and c is the coefficient vector corresponding to image x with respect to the basis Ψ . Typically, one assumes the signal x has a sparse representation, meaning that the coefficient vector c has a small number of non-zero coefficients: $\|c\|_0 := |\text{supp}(c)| = K \ll N$. We call such vectors K -sparse, and for approximately sparse vectors we write x_K to denote the best K -sparse approximation of x . In the simplest case, Ψ is an $N^2 \times N^2$ orthogonal matrix, but can also be generalized to an overcomplete dictionary. Here we have $y = SH\Psi c + \eta = \Phi c + \eta$, where $\Phi = SH\Psi$ is the sampling matrix.

Most of the work in the compressive sensing literature assumes Φ to be a random matrix, such as a partial DFT or one drawn from a Gaussian or Bernoulli distribution. However, in this scenario the matrix is not random, but instead has correlated columns whose structure may impair conventional compressive sensing recovery. Here H may not be a perfect low pass filter, so that it is possible for $\Phi = SH\Psi$ to preserve enough high frequency information for recovery to be possible; SH and Ψ have sufficient incoherency to allow c to be recovered with acceptable error.

Compressed sensing provides techniques for stable sparse recovery [3, 4, 5], but results for coherent sensing matrices have been limited [6, 7, 8].

Contribution. In this work we present an algorithm called the Partial Inversion (PartInv) method. PartInv eliminates a source of noise in the proxy used by existing greedy algorithms such as CoSaMP to identify the nonzero coefficients, which gives it better performance, since there is always some correlation among the columns of a rectangular sensing matrix (i.e. one with more columns than rows). PartInv works especially well when the sensing matrix has subsets of heavily correlated columns, which is seen in the super-resolution problem. We present theoretical justification for PartInv in the setting of coherent sensing matrices and provide rigorous results for the setting of more incoherent matrices. Our experiments confirm that PartInv yields improved recovery for both cases -when matrices are coherent (the super-resolution case) and when they are more incoherent. We believe that these results can be extended to the coherent case, and leave such an analysis for future work.

Organization. The correlation structure in the columns of the sensing matrix is first described in Section 2. In Section 3 we present an algorithm which tackles the correlation in the sensing matrix to recover the signal, and provide theoretical justification for it. Section 4 contains experimental results for the algorithm and comparisons to other existing methods, as well as a discussion of the particular case involving wavelet sparsity. We conclude in Section 5 and include the proofs of our main results in the appendix.

2 Correlation Structure

Typical examples of sparsifying bases Ψ for images are wavelets and blockwise discrete cosine transform bases. Images exhibit correlation at each scale: neighboring pixels are heavily correlated except across edges, local averages of neighboring blocks are heavily correlated except across edges, and so on. This makes wavelet-like bases, which have locally restricted atoms, suitable for sparsifying the image. For the super-resolution setting with the low resolution image of size $\frac{N}{2} \times \frac{N}{2}$,

the rows of SH consist of shifted versions of the filtering kernel with shifts of 2 horizontally and vertically. Due to the localized nature of wavelet bases, we expect columns of Φ that correspond to spatially distant bases in Ψ to have little correlation. If Ψ is a tree structured orthogonal wavelet basis matrix, columns of Ψ that overlap spatially are orthogonal, however when filtered by H , they result in significant correlation. Then we expect columns in Φ to show significant correlation in tree structured patterns.

We illustrate this with an example. For simplicity we consider only one-dimensional signals, though the discussion is equally valid for images. Suppose that Ψ is a 256×256 matrix whose columns consist of the length 256 Haar basis vectors, and SH is a 128×256 matrix obtained by shifting the filter kernel $h = \{0.1, 0.2, 0.4, 0.2, 0.1\}$ by two from one row to the next. SH represents the filtering and downsampling operation that generates the low resolution signal $y = SHx$ from the length 256 signal x . Then $\Phi = SH\Psi$ is the sampling matrix.

Fig. 1 shows the absolute values of the correlation matrix $C = \Phi^*\Phi$ (here and throughout A^* denotes the adjoint of A). This shows that only a small number of pairs of columns of Φ are strongly correlated to each other. Each filtered wavelet basis is correlated with other spatially overlapping bases at coarser and finer scale and in the immediate neighborhood, but has no correlation with spatially distant bases.

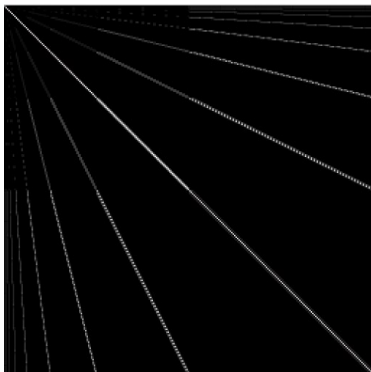


Figure 1: Absolute values of $\Phi^*\Phi$.

One of the central results in compressive sensing is that if matrix Φ exhibits a property called the Restricted Isometry Property (RIP) [9, 10], convex optimization can recover the sparse signal exactly [11, 12] via the following program

$$\min \|c\|_1 \quad \text{such that} \quad y = \Phi c \quad (2.1)$$

where $\|c\|_1 = \sum_i |c_i|$ is the ℓ_1 -norm of c . However, the sampling matrix $\Phi = SH\Psi$ described above does not obey the RIP and these results are not readily applicable. We propose a simple modification to CoSaMP[13], called *Partial Inversion (PartInv)* and described by Algorithm 1, to deal with correlated columns. As we shall see below, it provides an improvement in recovery performance over CoSaMP in this setting.

3 Partial Inversion

Consider the usual CS setting: Given a length M sample vector $y = \Phi c + \eta$ where Φ is an $M \times N$ sampling matrix and c a length N vector with sparsity $K < M$, we wish to obtain the best K -

sparse approximation \hat{c} to c . At each step let I be an index set and \hat{c}_I represent an estimate of the components of c corresponding to the column indices in I . The vector \hat{c} by itself is an estimate for all the columns $\{1..N\}$. Let L for $K \leq L < M$ be an adjustable parameter for the size of the set I . The value of L that provides the best performance depends on the matrix, a detailed is given in a later section. Let Φ_I denote the matrix of columns from Φ corresponding to indices in the set I . Similarly, c_I denotes the vector c with entries in the complement of I set to zero. At times we will write c_I instead to be the vector in $\mathbb{C}^{|I|}$ consisting of the elements of c indexed by I , in which the notation will be clear from context. Let $\tilde{I} = \{1..N\} \setminus I$ denote the complement of I . For any full rank matrix A , define $A^\dagger = (A^*A)^{-1}A^*$.

Algorithm 1 Given $y = \Phi c$, return best K -sparse approximation \hat{c}

```

1:  $\hat{c} \leftarrow \Phi^* y$ ;  $I^{(0)} \leftarrow$  indices of the  $L$ -largest magnitudes of  $\hat{c}$ ;  $k \leftarrow 0$ 
2: while Stopping condition not met do
3:    $\hat{c}_{I^{(k)}} \leftarrow \Phi_{I^{(k)}}^\dagger y$ 
4:    $r \leftarrow y - \Phi_{I^{(k)}} \hat{c}_{I^{(k)}}$ 
5:    $J^{(k)} \leftarrow \tilde{I}^{(k)}$ 
6:    $\hat{c}_{J^{(k)}} \leftarrow \Phi_{J^{(k)}}^* r$ 
7:    $I^{(k+1)} \leftarrow$  indices of the  $L$ -largest magnitude components of  $\hat{c}$ .
8:    $k \leftarrow k + 1$ 
9: end while

```

For the noiseless case $\eta = 0$, the stopping condition can be obtained by testing the magnitude of $r_2 = y - \Phi \hat{c}$ at the start of each iteration. If the set I does not vary from one iteration to the next, the algorithm cannot progress further and should be stopped immediately. In practice the inversion of line 3 can be done efficiently by Richardson's algorithm (see e.g. [14, Sec. 7.2]).

This algorithm demonstrates improvement relative to CoSaMP when the accurate recovery region is considered on a plot of $\frac{K}{M}$ versus $\frac{M}{N}$. The motivation is the following (for simplicity we drop the iteration indicator k): From line 3, we obtain

$$\hat{c}_I = \Phi_I^\dagger y = \Phi_I^\dagger (\Phi_I c_I + \Phi_{\tilde{I}} c_{\tilde{I}}) \quad (3.1)$$

$$= c_I + (\Phi_I^* \Phi_I)^{-1} \Phi_I^* \Phi_{\tilde{I}} c_{\tilde{I}}. \quad (3.2)$$

In CoSaMP, Φ^* rather than Φ_I^\dagger is used to form a proxy and identify large coefficients of the signal. In this case, the proxy \hat{c} restricted to the index set I when $r = y$ satisfies

$$\begin{aligned} \hat{c}_I &= \Phi_I^* y \\ &= \Phi_I^* \Phi_I c_I + \Phi_I^* \Phi_{\tilde{I}} c_{\tilde{I}} \\ &= c_I + (\Phi_I^* \Phi_I - I) c_I + \Phi_I^* \Phi_{\tilde{I}} c_{\tilde{I}} \end{aligned}$$

If the index set I contains several nonzero coefficients (which we hope is true), then $(\Phi_I^* \Phi_I - I) c_I$, which results from the mutual interference between the columns of Φ_I , is significant and is a source of noise in \hat{c}_I . This term is eliminated in (3.1). Partial inversion does add $(\Phi_I^* \Phi_I)^{-1}$ to the remaining noise term, however, the singular values of this term can be kept from significantly amplifying the noise term by a conservative choice of L , the size of the index set I . For example, empirically we find that $L = K$ tends to be a safe choice, but the value of L that produces optimal performance

depends on the type of matrix, besides K and M . The improved estimate \hat{c}_I further produces an improved estimate $\hat{c}_{J^{(k)}}$, which leads to a better selection of nonzero coefficients in the next iteration.

The expression (3.1) also indicates how the correlation structure may be used to improve recovery. The noise term $(\Phi_I^* \Phi_I)^{-1} \Phi_I^* \Phi_{\tilde{I}} c_{\tilde{I}}$ depends upon the correlation between the sets Φ_I and $\Phi_{\tilde{I}}$ given by $\Phi_I^* \Phi_{\tilde{I}}$. This correlation is weak if Φ_I and $\Phi_{\tilde{I}}$ are sufficiently spread. However, the correlation is likely to remain large if L is significant compared to M , as will be the case when $\frac{K}{M}$ is large.

In the wavelet case, if set I contains several wavelet subsets and \tilde{I} contains columns that are not from any of the subsets in I , then the correlation in the noise term is small. By these arguments, we see that if the matrix Φ satisfies some mild assumptions, then Partial Inversion will converge to the sparse solution in a fixed number of iterations. Experimentally we see that even with high correlations in the matrix Φ , PartInv provides accurate recovery (see Sec. 4). Under slightly stronger assumptions, we have the following mathematical justification, whose proof can be found in the appendix. Here and throughout, we use $\|\cdot\|_2$ and $\|\cdot\|$ to denote the usual ℓ_2 norm and spectral norm, respectively.

Theorem 3.1. *Let $c \in \mathbb{C}^N$ be a K -sparse vector with support set T satisfying*

$$|c_i| \geq 3\delta \|c\|_2, \quad \forall i \in T, \quad (3.3)$$

for some fixed constant $0 < \delta \leq \frac{1}{3\sqrt{K}}$. Assume that the dictionary Φ satisfies the following properties:

$$\|\Phi_{T_1}^* \Phi_{T_1} c_{T_1}\|_2 \geq (1 - \delta)^2 \|c_{T_1}\|_2, \quad \forall T_1 \subseteq T \quad (3.4)$$

$$\|\Phi_I\| \leq A, \quad \forall |I| \leq L \quad (3.5)$$

$$\|\Phi_I^\dagger\| \leq A, \quad \forall |I| \leq L \quad (3.6)$$

$$\|\Phi_I \Phi_I^\dagger \Phi_{I^c \cap T}\| \leq \delta/A, \quad \forall |I| \leq L. \quad (3.7)$$

where L is the parameter used in the Partial Inversion (PartInv) algorithm, and $1 \leq A < \sqrt{L}$ is another fixed constant. Then PartInv reconstructs the signal, $\hat{c} = c$ in at most K iterations.

Remarks.

1. First, we remark that the assumptions of this theorem restrict not only the sparsity of the signal, but also the distribution of its non-zero coefficients, contrary to typical results in compressive sensing. We also comment that if (3.3) does not hold, then the proof guarantees that all coefficients c_i of c that do satisfy that bound are still recovered.

2. We next relate the first three assumptions on Φ of Theorem 3.1 to the Restricted Isometry Property (RIP) [9], which states that

$$(1 - \delta) \|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta) \|x\|_2^2 \quad \text{for all } k\text{-sparse } x. \quad (3.8)$$

It is now well-known that many classes of random matrices satisfy this property with high probability for $\delta < \epsilon$ when the number of measurements M is on the order of $k \log(N)/\epsilon^2$ [11, 10]. Note that the RIP is equivalent to asking that

$$\sqrt{1 - \delta} \leq \sigma_{\min}(\Phi_I) = \sigma_k(\Phi_I) \leq \dots \leq \sigma_1(\Phi_I) = \sigma_{\max}(\Phi_I) \leq \sqrt{1 + \delta} \quad (3.9)$$

for all $|I| \leq k$. Here, $\sigma_1(\cdot), \dots, \sigma_k(\cdot)$ denote the k sorted singular values (in decreasing order), and $\sigma_{\min}(\cdot), \sigma_{\max}(\cdot)$ denote the smallest and largest singular values. In contrast, our first three assumptions on Φ above are all relaxations of the RIP condition (in both directions). Indeed, we may rewrite them as follows:

$$\begin{aligned} \sigma_{\min}(\Phi_{T_1}) &\geq 1 - \delta, & \forall T_1 \subseteq T \\ \sigma_{\max}(\Phi_I) &\leq A, & \forall |I| \leq L \\ \sigma_r(\Phi_I) &\geq 1/A \quad (r = \text{rank}(\Phi_I)), & \forall |I| \leq L. \end{aligned}$$

Note in particular that the last condition above says that the smallest *nonzero* singular value of Φ_I is sufficiently large (i.e., at least $1/A$). The matrix Φ_I may still have several zero singular values, which is in contrast to the RIP which requires the matrix Φ_I to have no zero singular values and its smallest singular value to be close to one. This means that off the support, the matrix may have significant correlations. The most restrictive condition is (3.7), which we believe can be improved. We leave a detailed analysis of the requirement on M in the correlated case for future work.

3. We note that if the matrix Φ satisfies the RIP with parameter L , then the first three Φ assumptions of the theorem hold with constant $A = \sqrt{1 + \delta}$ (see [15, Prop. 3.2] and [16, Prop. 3.2 and 3.3]). Since many classes of $M \times N$ random matrices satisfy the RIP when M is on the order of $k \log(N/k)/\delta^2$ [11, 10], for the assumptions of Theorem 3.1 to hold, one needs $M \approx k^2 \log(N/k)$ measurements. We expect that the squared dependence on k is an artifact of the proof and can be improved.

4. Experimentally, we see that PartInv is also robust to noise, and doesn't necessarily require the columns of Φ to satisfy strong incoherence bounds. We leave this theoretical analysis as future work, and provide experimental evidence in Section 5.

4 Experimental Results

We compare the recovery performance of Partial Inversion with CoSaMP and convex optimization (2.1) for two classes of matrices: Gaussian random matrices, and matrices constructed to have highly correlated subsets of columns with low correlation across subsets.

In the first case, we construct M by N matrices with $N(0, 1)$ elements along with the coefficient vector c containing K nonzero entries taken from a $N(0, 1)$ distribution. The nonzero locations are selected uniformly at random from $\{1 \dots N\}$. The columns in each matrix are normalized to have unit l_2 norm. We set $N = 256$ and vary $\delta = \frac{M}{N}$ from 0.1 to 0.9 in steps of 0.1. For each δ we vary $\rho = \frac{K}{M}$ from 0.1 to 0.9 in steps of 0.1. For each (δ, ρ) point we carry out 25 trials, and declare success if $\frac{1}{N} \|c - \hat{c}\|^2 < 10^{-5}$. For PartInv we considered two cases for the size of subset I : $L = K$ and $L = \max\{K, 0.8M\}$. We see better performance in the $L = K$ case. For l_1 minimization we use the l_1 -magic package [17].

We show the results in Fig. 2.

In the second case, we construct M by N matrices with $N = 256$ and variable M and a block diagonal structure. The columns are divided into 16 column subsets. In each subset we set $M/16$ rows to 1 and add Gaussian noise with zero mean and variance 0.0625. In addition, to every element of the matrix we add noise drawn from a zero-mean normal distribution with variance $\frac{1}{M}$. This produces heavy intra-subset correlation and light correlation across subsets. We let the coefficient vector c contain K non-zero elements drawn from a $N(0, 1)$ distribution. We select 4 of

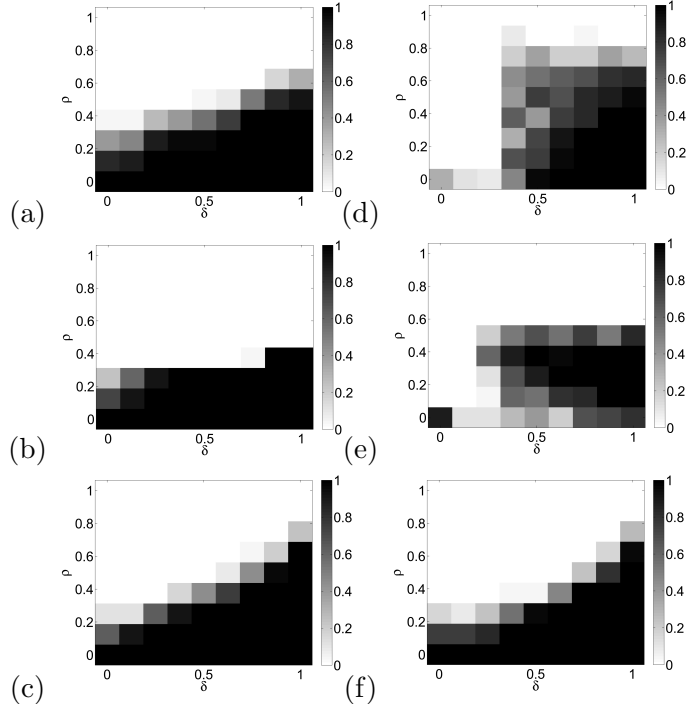


Figure 2: Proportion of successes on Gaussian matrices using (a) PartInv, (b) CoSaMP and (c) ℓ_1 -minimization, and proportion of successes on correlated column subset matrices using (d) PartInv, (e) CoSaMP and (f) ℓ_1 -minimization for various values of $\delta = \frac{M}{N} \in (0, 1)$ (horizontal axis) and $\rho = \frac{K}{M} \in (0, 1)$ (vertical axis).

the 16 subsets at random and in each subset select $\frac{K}{4}$ of the indices to have nonzero values, again uniformly at random. If some of the nonzeros were left over, they are accommodated in a fifth subset. For PartInv we set $L = \max\{K, 0.8M\}$.

4.1 Sensitivity of Recovery Performance to the Size of the Selected Subset

We study the sensitivity of recovery performance to variations in parameter L , the size of the selected subset. We select three representative values of M and K that lie near the transition boundary between the high recovery and low recovery regions in the $\delta - \rho$ diagram (see Fig. 2). For each (M, K) pair, we vary L over the range $\{K..0.8M\}$ in steps of 2, for the Gaussian and correlated column subsets studied earlier. The results are shown in Fig. 3.

We see that for Gaussian matrices, the performance drops as L increases from K towards $0.8M$, due to the presence of increasingly smaller singular values of Φ_I as L approaches M , which lead to amplification of the noise component $\Phi_I^\dagger \Phi_I c_I$. For correlated column subset matrices, the performance is mixed, with performance improving as L increases for the the first two cases, while it decreases for the last. Understanding this behavior is a topic for future work.

We also obtain the best performance of Partial Inversion as L is varied from K to M for each δ and ρ in the range $(0, 1)$ with 100 trials for each (M, K, L) combination. The results are shown in Fig. 4. When compared with Fig. 2 we see that there is a small improvement in the Gaussian case and a more significant improvement for the correlated columns case, especially for small (δ, ρ) values. This is consistent with the sensitivity to L seen in Fig. 3, where we see that for correlated

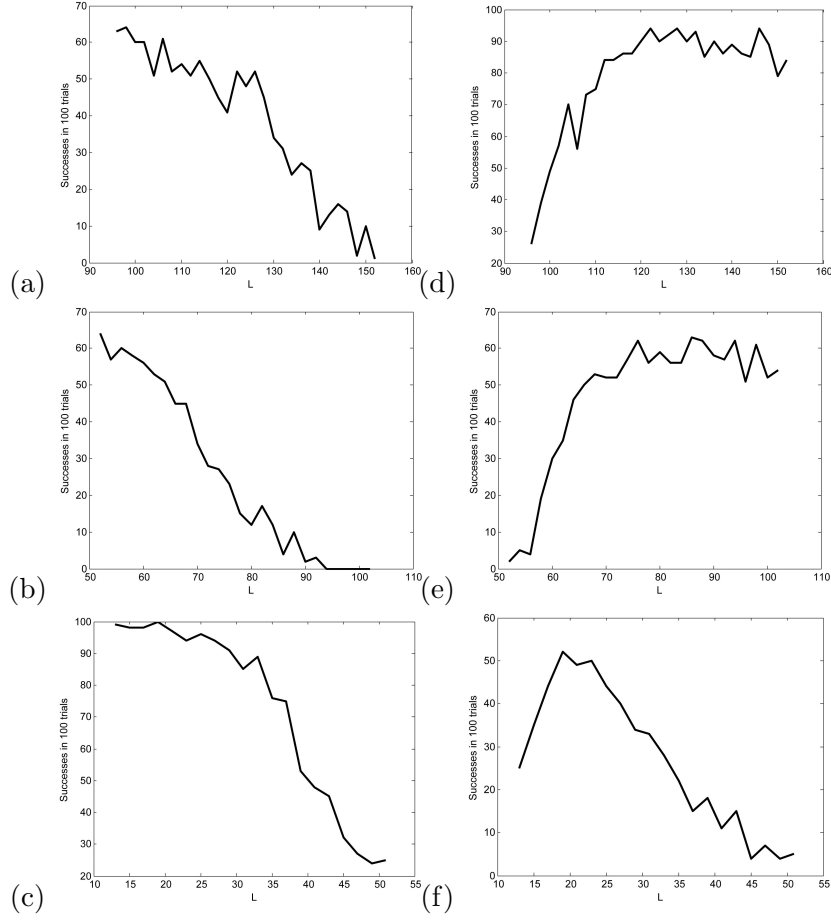


Figure 3: Proportion of successes using PartInv on Gaussian matrices(left) and correlated column subset matrices(right) for $M = 192, K = 96$ (a,d), for $M = 128, K = 52$ (b,e) and for $M = 64, K = 13$ (c,f) as L is varied from K upto $0.8M$ in steps of 2.

columns with $M = 64, K = 13$, the performance is best at lower values of L close to K . We also provide the optimal values of L for each (δ, ρ) in Table 1.

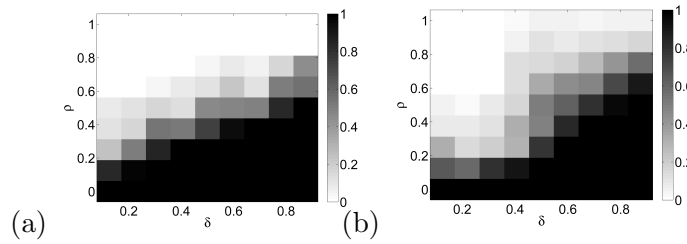


Figure 4: Best proportion of successes using PartInv as L is varied from K upto M in steps of 2 for each (M, K) , on (a)Gaussian matrices (left) and (b) correlated column subset matrices(right).

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	90	108	125	142	169
0	0	54	61	76	91	109	128	162
13	25	38	51	66	76	99	112	141
10	20	30	44	51	65	77	83	92
7	17	22	36	44	45	53	61	69
5	10	17	20	25	30	35	40	46
4	5	7	10	12	15	17	20	23

0	0	0	101	115	153	173	203	219
0	0	0	99	126	130	177	199	206
0	0	0	99	127	133	155	168	193
0	0	0	85	116	129	133	164	180
18	28	42	85	100	116	137	140	163
16	20	76	100	87	93	131	109	112
9	19	30	58	74	85	73	73	83
9	14	21	50	69	46	49	48	54
10	11	13	22	20	21	23	24	29

Table 1: Values of L that produce optimal performance for Gaussian(top) and correlated column subset matrices(bottom); a 0 indicates that the number of successes is zero

4.2 Recovery of Coefficients Concentrated on Wavelet Trees

We next use Partial Inversion to recover nonzero coefficients that are concentrated on wavelet trees, which is commonly seen when a signal or image with discontinuities is decomposed in a wavelet basis. When the coefficients are concentrated on an isolated set (a set of columns that have low correlation with columns outside the set), a setwise estimator is especially useful to identify the sets on which the coefficients are nonzero.

Consider the 2D wavelet case. Suppose that I is the index set of columns of the wavelet basis belonging to a particular tree rooted at a coarse scale and containing finer scale coefficients. We have

$$z_I = \Phi_I^* y = \Phi_I^* \Phi_I c_I + \Phi_I^* \Phi_{\bar{I}} c_{\bar{I}}. \quad (4.1)$$

Because Φ_I is relatively isolated from the columns in $\Phi_{\bar{I}}$, the second term is small, and because most of the elements of c_I are nonzero, the first term is large. This is further intensified by the mutual correlation of the columns of Φ_I which is high because of the spatial overlap of the support of the wavelet bases in the tree. This motivates a simple selection criterion for measuring the strength of the nonzero coefficients in each wavelet tree I : $s_I = \sum_{j \in I} |z_j|$. We use this criterion along with

PartInv to select wavelet trees that are known to be nonzero. We denote the number of subsets by SetNum. This modified method is described by Algorithm 2.

Algorithm 2 Given $y = \Phi c$, with K nonzero coefficients concentrated on wavelet trees, return best K -sparse approximation \hat{c}

```

1:  $\hat{c} \leftarrow \Phi^* y$ ;
2:  $k \leftarrow -1$ 
3: for  $j = 1 \rightarrow \text{SetNum}$  do
4:    $s_j \leftarrow \sum_{l \in I_j} |\hat{c}_l|$ 
5: end for
6:  $I^{k+1} \leftarrow$  indices of columns contained in the sets with the largest magnitude  $s_k$ , to include at least  $K$  coefficients.
7:  $k \leftarrow k + 1$ 
8: while Stopping condition not met do
9:    $\hat{c}_{I^{(k)}} \leftarrow \Phi_{I^{(k)}}^\dagger y$ 
10:   $r \leftarrow y - \widetilde{\Phi}_{I^{(k)}} \hat{c}_{I^{(k)}}$ 
11:   $J^{(k)} \leftarrow I^{(k)}$ 
12:   $\hat{c}_{J^{(k)}} \leftarrow \Phi_{J^{(k)}}^* r$ 
13:  Repeat lines 3 – 6
14:   $k \leftarrow k + 1$ 
15: end while

```

4.3 Experimental Results using wavelet trees

To test this algorithm, we use the Daubechies-5 wavelet basis in two dimensions over 32×32 size patches with 5 levels of decomposition. This gives a size 1024 by 1024 matrix Ψ . We divide this matrix into 49 sets: 1 set of the coarsest scale coefficients in a block of size 4×4 containing the two coarsest scales, and 48 other sets rooted at the coefficients at the next finer scale. Each of these sets contains $1 + 4 + 16 = 21$ coefficients in a quadtree structure. To create matrix Φ we first apply a blurring filter H with a symmetric 5×5 kernel that is close to a delta function. This simulates practical optical sample acquisition effects such as diffraction and helps prevent rank deficiency problems when carrying out inversion. We use different 2D sampling patterns to carry out the subsampling operation represented by matrix S . Hence the acquisition process is represented by $y = \Phi c$ where $\Phi = SH\Psi$. The sampling patterns are shown in Table 2 for each sampling rate $\delta = \frac{M}{N}$ used to generate the results. Each pattern is replicated 8 times in horizontal and vertical directions to give the 32×32 sampling pattern used for matrix S . The sampling patterns were selected to allow δ to increase in constant steps over the whole range $(0, 1)$, while distributing the samples as uniformly and symmetrically as possible. Common subsampling patterns used in superresolution problems would have sampled only a part of this interval, and are likely to give similar results as the patterns used here that are closest to them in density. The filter kernel is a 5×5 kernel with 0.29 at the center and 0.02 in other locations. The signals are generated by uniformly selecting at random wavelet trees to make the sparsity of the signal the specified value. The coefficients in these trees are set to values chosen from a standard normal distribution, and the rest are set to zero.

For each data point we carry out 100 trials. We declare success if $\frac{1}{N} \|c - \hat{c}\|^2 < 10^{-5}$ where $N = 32 \times 32$. This shows improvement in selection performance with the sum estimator.

<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table>	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table>	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td></tr> </table>	1	0	1	0	0	1	0	1	1	0	0	0	0	0	1	0	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> </table>	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> </table>	0	1	0	1	1	0	1	0	0	1	1	1	1	1	0	1
0	0	0	0																																																																																	
0	1	0	0																																																																																	
0	0	0	0																																																																																	
0	0	0	1																																																																																	
1	0	0	0																																																																																	
0	0	1	0																																																																																	
0	1	0	0																																																																																	
0	0	0	1																																																																																	
1	0	1	0																																																																																	
0	1	0	1																																																																																	
1	0	0	0																																																																																	
0	0	1	0																																																																																	
1	0	1	0																																																																																	
0	1	0	1																																																																																	
1	0	1	0																																																																																	
0	1	0	1																																																																																	
0	1	0	1																																																																																	
1	0	1	0																																																																																	
0	1	1	1																																																																																	
1	1	0	1																																																																																	
(a) $\delta = \frac{2}{16}$	(b) $\delta = \frac{4}{16}$	(c) $\delta = \frac{6}{16}$	(d) $\delta = \frac{8}{16}$	(e) $\delta = \frac{10}{16}$																																																																																
<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>1</td></tr> </table>		1	1	0	1	0	1	1	1	1	1	1	0	1	0	1	1	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px;"> <tr><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td></tr> </table>		1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1																																																	
1	1	0	1																																																																																	
0	1	1	1																																																																																	
1	1	1	0																																																																																	
1	0	1	1																																																																																	
1	1	1	1																																																																																	
1	1	0	1																																																																																	
1	1	1	1																																																																																	
0	1	1	1																																																																																	
(f) $\delta = \frac{12}{16}$		(g) $\delta = \frac{14}{16}$																																																																																		

Table 2: Sampling Patterns

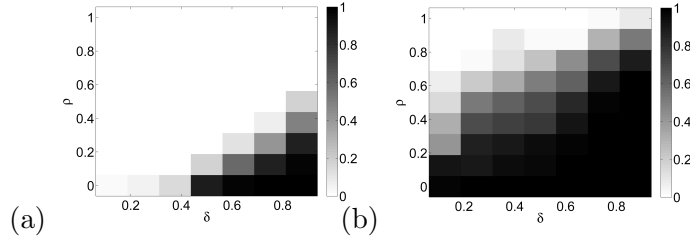


Figure 5: Proportion of successes with nonzero coefficients concentrated on wavelet trees from (a) ℓ_1 -minimization and (b) PartInv.

5 Conclusion and Future Work

We consider methods of compressive sensing recovery for sampling matrices with correlated columns. This structure commonly arises in physical sample acquisition/reconstruction scenarios such as image super-resolution. We describe Partial Inversion, an algorithm that improves compressive sensing recovery by removing a source of noise in the initial estimator, and demonstrate its performance by simulations on Gaussian and correlated column subset matrices. We consider compressive sensing recovery when the nonzero coefficients are concentrated on wavelet trees, and demonstrate a simple estimator that improves selection of the trees that carry the nonzero coefficients.

We provide an analysis of the proposed method, which shows under mild conditions on the sensing matrix Φ that exact recovery is provided in the noiseless regime. Our experimental results suggest that the method is also robust to noise, and that the assumptions placed on the correlation in the matrix can be weakened. We believe an extension of our analysis is a good direction for future work.

We also plan to consider compressive sensing recovery where the columns of the sampling matrix Φ can be grouped into nearly-isolated sets, such that correlation among pairs of columns within a set may be significant, but correlation between two columns that belong to different sets is relatively small. Future research will exploit this structure to efficiently reconstruct the signal.

A Proof of Theorem 3.1

We start by pointing out that the first three Φ assumptions in Theorem 3.1 imply the following bounds:

$$\begin{aligned}\|\Phi_I^* \Phi_{I^c \cap T}\| &\leq \delta, \quad \forall |I| \leq L \\ \|\Phi_I^\dagger \Phi_{I^c \cap T}\| &\leq \delta, \quad \forall |I| \leq L.\end{aligned}$$

To see this, write $\Phi_I = U\Sigma V^*$, the SVD decomposition in reduced form (i.e., the diagonals of Σ are all strictly positive). Then the third condition becomes

$$\|\Phi_I \Phi_I^\dagger \Phi_{I^c \cap T}\| = \|UU^* \Phi_{I^c \cap T}\| = \|U^* \Phi_{I^c \cap T}\| \leq \delta/A,$$

which holds because U, V are both orthogonal matrices. Therefore,

$$\begin{aligned}\|\Phi_I^* \Phi_{I^c \cap T}\| &= \|V\Sigma U^* \Phi_{I^c \cap T}\| = \|\Sigma U^* \Phi_{I^c \cap T}\| \\ &\leq \|\Sigma\| \cdot \|U^* \Phi_{I^c \cap T}\| = \|\Phi_I\| \cdot \|U^* \Phi_{I^c \cap T}\| \\ &\leq A \cdot \delta/A = \delta. \\ \|\Phi_I^\dagger \Phi_{I^c \cap T}\| &= \|V\Sigma^{-1} U^* \Phi_{I^c \cap T}\| = \|\Sigma^{-1} U^* \Phi_{I^c \cap T}\| \\ &\leq \|\Sigma^{-1}\| \cdot \|U^* \Phi_{I^c \cap T}\| = \|\Phi_I^\dagger\| \cdot \|U^* \Phi_{I^c \cap T}\| \\ &\leq A \cdot \delta/A = \delta.\end{aligned}$$

We fix an iteration k and consider $\hat{c}_{I^{(k)}}$, the approximation of c at the beginning of iteration k . Let $T = \text{supp}(c)$ be the true support of the signal c , so that $c = c_T$. Denote $J^{(k)} = \widetilde{I^{(k)}}$, the complement of $I^{(k)}$ in the set of all column indices $1 : N$. We then have

$$\begin{aligned}\hat{c}_{I^{(k)}} &= \Phi_{I^{(k)}}^\dagger y = \Phi_{I^{(k)}}^\dagger \Phi c = \Phi_{I^{(k)}}^\dagger (\Phi_{I^{(k)}} c_{I^{(k)}} + \Phi_{J^{(k)}} c_{J^{(k)}}) \\ &= c_{I^{(k)}} + \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)}} c_{J^{(k)}}.\end{aligned}\tag{A.1}$$

This equation is the foundation for the analysis below. We consider the following three cases, depending on whether T has been partially recovered, fully recovered, or not at all recovered. Considering these cases separately serves to highlight how the algorithm gathers the support, and why each assumption is needed. We aim to show that at each iteration, at least one new support element is identified, and that no correct support elements are lost.

Case 1: $T \subset I^{(k)}$

In this case we know that $c_{J^{(k)}} = 0$ (which also implies that $y = \Phi_{I^{(k)}} c_{I^{(k)}}$). So we obtain from (A.1) that $\hat{c}_{I^{(k)}} = c_{I^{(k)}}$, and hence, $r = y - \Phi_{I^{(k)}} \hat{c}_{I^{(k)}} = 0$. This yields that $\hat{c}_{J^{(k)}} = \Phi_{J^{(k)}}^* r = 0$. As a result, the estimator of c at iteration k is $\hat{c}_{(k)} = c$. We have recovered the original signal c without error.

In sum, if at one iteration $I^{(k)}$ contains the entire support of the signal, then the corresponding estimator \hat{c} coincides with c .

Case 2: $T \cap I^{(k)} = \emptyset$

In this case we have $c_{I^{(k)}} = 0$ and $T \subset J^{(k)}$, so that $y = \Phi_{J^{(k)}} c_{J^{(k)}} = \Phi_T c_T$. Starting with

$$\hat{c}_{I^{(k)}} = \Phi_{I^{(k)}}^\dagger y,$$

we get

$$\begin{aligned} r &= y - \Phi_{I^{(k)}} \hat{c}_{I^{(k)}} = (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) y; \\ \hat{c}_{J^{(k)}} &= \Phi_{J^{(k)}}^* r = \Phi_{J^{(k)}}^* (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) y. \end{aligned}$$

Now plug in $y = \Phi_T c_T$ to the last three equations:

$$\begin{aligned} \hat{c}_{I^{(k)}} &= \Phi_{I^{(k)}}^\dagger \Phi_T c_T \\ r &= (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) \Phi_T c_T \\ \hat{c}_{J^{(k)}} &= \Phi_{J^{(k)}}^* (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) \Phi_T c_T \\ &= \Phi_{J^{(k)}}^* \Phi_T c_T - \Phi_{J^{(k)}}^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T \end{aligned}$$

Since $T \subset J^{(k)}$, we may split $\hat{c}_{J^{(k)}}$ into \hat{c}_T and $\hat{c}_{J^{(k)}-T}$ and handle them separately:

$$\begin{aligned} \hat{c}_T &= \Phi_T^* \Phi_T c_T - \Phi_T^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T; \\ \hat{c}_{J^{(k)}-T} &= \Phi_{J^{(k)}-T}^* \Phi_T c_T - \Phi_{J^{(k)}-T}^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T. \end{aligned}$$

Using the assumptions we have the following estimates

$$\begin{aligned} \|\hat{c}_{I^{(k)}}\|_2 &= \|\Phi_{I^{(k)}}^\dagger \Phi_T c_T\|_2 \leq \delta \|c_T\|_2 = \delta \|c\|_2 \\ \|\hat{c}_T\|_2 &= \|\Phi_T^* \Phi_T c_T - \Phi_T^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T\|_2 \\ &\geq \|\Phi_T^* \Phi_T c_T\|_2 - \|\Phi_T^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T\|_2 \\ &\geq (1 - \delta)^2 \|c_T\|_2 - \delta \|\Phi_{I^{(k)}}^\dagger \Phi_T c_T\|_2 \\ &\geq (1 - \delta)^2 \|c_T\|_2 - \delta \cdot \delta \|c_T\|_2 \\ &= (1 - 2\delta) \|c\|_2 \end{aligned}$$

These two inequalities imply that the largest (in magnitude, same below) possible element in $\hat{c}_{I^{(k)}}$ is $\delta \|c\|$, while there is at least one element in \hat{c}_T that exceeds $\frac{1-2\delta}{\sqrt{K}} \|c\|$, where $K = |T|$ is the sparsity of the signal c . Regarding the elements in $\hat{c}_{J^{(k)}-T}$, we upper bound them individually: For any $i \in J^{(k)} - T$, we have

$$\begin{aligned} |\hat{c}_i| &= |\Phi_i^* \Phi_T c_T - \Phi_i^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T| \\ &\leq |\Phi_i^* \Phi_T c_T| + |\Phi_i^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T| \\ &\leq \delta \|c_T\|_2 + \|\Phi_i\|_2 \cdot \|\Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_T c_T\|_2 \\ &\leq \delta \|c_T\|_2 + 1 \cdot \frac{\delta}{A} \|c_T\|_2 \\ &\leq 2\delta \|c\|_2. \end{aligned}$$

That is, the largest possible element in $\hat{c}_{J^{(k)}-T}$ is $2\delta\|c\|$. Therefore, if we have that

$$\frac{1-2\delta}{\sqrt{K}} > 2\delta, \quad \text{or} \quad \delta < \frac{1}{2\sqrt{K}+2} \quad (\text{A.2})$$

then at least one index from T will be selected at the end of this iteration. Indeed, this inequality is true (provided that $K > 3$) because $\delta < \frac{1}{3\sqrt{K}}$.

To summarize, if the initial $I^{(k)}$ satisfies $I^{(k)} \cap T = \emptyset$, at least one index from T will be selected at the end of the iteration.

Case 3: $I^{(k)} \cap T \neq \emptyset$, and $J^{(k)} \cap T \neq \emptyset$

This case is in between the first two cases. Our goal is to show that at the end of iteration k , all indices in $I^{(k)} \cap T$ will be preserved and at least one new index from $J^{(k)} \cap T$ will be selected.

To establish this, first note that

$$y = \Phi_T c_T = \Phi_{I^{(k)} \cap T} c_{I^{(k)} \cap T} + \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T}. \quad (\text{A.3})$$

We continue directly from Equation (A.1) as follows:

$$\begin{aligned} \hat{c}_{I^{(k)}} &= c_{I^{(k)}} + \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)}} c_{J^{(k)}} \\ &= c_{I^{(k)} \cap T} + \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} \end{aligned}$$

Under the same assumptions, we split $\hat{c}_{I^{(k)}}$ as $\hat{c}_{I^{(k)} \cap T} + \hat{c}_{I^{(k)}-T}$ and estimate them separately. Since

$$\|\Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T}\|_2 \leq \delta \|c_{J^{(k)} \cap T}\|_2,$$

we know that every element of $\hat{c}_{I^{(k)}-T}$ is at most $\delta \|c_{J^{(k)} \cap T}\|$ and every element of $\hat{c}_{I^{(k)} \cap T}$ is at least

$$3\delta \|c\|_2 - \delta \|c_{J^{(k)} \cap T}\|_2 > 2\delta \|c_{J^{(k)} \cap T}\|_2.$$

We derive a formula for $\hat{c}_{J^{(k)}}$ and will discuss its two parts $\hat{c}_{J^{(k)} \cap T} + \hat{c}_{J^{(k)}-T}$ also separately.

$$\begin{aligned} r &= y - \Phi_{I^{(k)}} \hat{c}_{I^{(k)}} \\ &= y - \Phi_{I^{(k)}} c_{I^{(k)}} - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)}} c_{J^{(k)}} \\ &= \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} \\ &= (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} \\ \hat{c}_{J^{(k)}} &= \Phi_{J^{(k)}}^* r \\ &= \Phi_{J^{(k)}}^* (I - \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger) \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} \\ &= \Phi_{J^{(k)}}^* \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} - \Phi_{J^{(k)}}^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T}. \end{aligned}$$

From above and using similar arguments, we have

$$\begin{aligned} \|\hat{c}_{J^{(k)} \cap T}\|_2 &= \|\Phi_{J^{(k)} \cap T}^* \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T} - \Phi_{J^{(k)} \cap T}^* \Phi_{I^{(k)}} \Phi_{I^{(k)}}^\dagger \Phi_{J^{(k)} \cap T} c_{J^{(k)} \cap T}\|_2 \\ &\geq (1-\delta)^2 \|c_{J^{(k)} \cap T}\|_2 - \delta^2 \|c_{J^{(k)} \cap T}\|_2 \\ &= (1-2\delta) \|c_{J^{(k)} \cap T}\|_2 \end{aligned}$$

and each element of $\hat{c}_{J^{(k)} \cap T}$ is no more than

$$\delta \cdot \|c_{J^{(k)} \cap T}\|_2 + 1 \cdot \frac{\delta}{A} \cdot \|c_{J^{(k)} \cap T}\|_2 \leq 2\delta \|c_{J^{(k)} \cap T}\|_2. \quad (\text{A.4})$$

Considering $\hat{c}_{I^{(k)} \cap T}, \hat{c}_{I^{(k)} \cap T}, \hat{c}_{J^{(k)} \cap T}, \hat{c}_{J^{(k)} \cap T}$ all together, in order for all elements in $\hat{c}_{I^{(k)} \cap T}$ and at least one element from $\hat{c}_{J^{(k)} \cap T}$ to be among the L largest entries of \hat{c} and hence selected at the end of the iteration, we only need

$$\frac{1 - 2\delta}{\sqrt{|J^{(k)} \cap T|}} \|c_{J^{(k)} \cap T}\|_2 > 2\delta \|c_{J^{(k)} \cap T}\|_2$$

or equivalently,

$$\delta < \frac{1}{2\sqrt{|J^{(k)} \cap T|} + 2}.$$

This inequality is indeed true because we already have $\delta < \frac{1}{2\sqrt{K+2}}$ from Case 2 and $|J^{(k)} \cap T| < K$.

References

- [1] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, “Advances and challenges in super-resolution,” *Int. J. Imag. Syst. Tech.*, vol. 14, no. 2, pp. 47–57, 2004.
- [2] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image superresolution via sparse representation,” *IEEE T. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
- [3] E. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *IEEE T. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.
- [4] E. Candès, J. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [5] D. Donoho and P. Stark, “Uncertainty principles and signal recovery,” *SIAM J. Appl. Math.*, vol. 49, no. 3, pp. 906–931, 1989.
- [6] E. Candès, Y. Eldar, D. Needell, and P. Randall, “Compressed sensing with coherent and redundant dictionaries,” *Appl. Comput. Harmon. A.*, vol. 31, no. 1, pp. 59–73, 2011.
- [7] A. Fannjiang and W. Liao, “Coherence pattern-guided compressive sensing with unresolved grids,” *SIAM J. Imaging Sci.*, vol. 5, no. 1, pp. 179–202, 2012.
- [8] E. Candès and C. Fernandez-Granda, “Towards a mathematical theory of super-resolution,” *Comm. Pure Appl. Math. (online)*, 2013.
- [9] E. Candès and T. Tao, “Decoding by Linear Programming,” *IEEE T. Inform. Theory*, vol. 51, no. 12, pp. 4203 – 4215, dec. 2005.
- [10] M. Rudelson and R. Vershynin, “On sparse reconstruction from Fourier and Gaussian measurements,” *Comm. Pure Appl. Math.*, vol. 61, pp. 1025–1045, 2008.

- [11] E. Candes and T. Tao, “Near optimal signal recovery from random projections: Universal encoding strategies?” *IEEE T. Inform. Theory*, vol. 52, pp. 5406–5425, Dec. 2006.
- [12] E. Candes, “The Restricted Isometry Property and its implications for Compressed Sensing,” *Cr. Acad. Sci. I-Math.*, pp. 589–592, Dec. 2008.
- [13] D. Needell and J. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. Comput. Harmon. A.*, vol. 26, pp. 301–321, 2009.
- [14] Å. Björck, *Numerical Methods for Least Squares Problems*. Philadelphia: SIAM, 1996.
- [15] D. Needell and R. Vershynin, “Uniform Uncertainty Principle and signal recovery via Regularized Orthogonal Matching Pursuit,” *Foundations of Computational Mathematics*, vol. 9, pp. 317–334, 2009.
- [16] D. Needell and J. Tropp, “Cosamp: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. Comput. Harmon. A.*, vol. 26, no. 3, pp. 301–321, 2009.
- [17] ℓ_1 -magic. [Online]. Available: <http://www.acm.caltech.edu/l1magic/>