Complex Sparse Projections for Compressed Sensing

Abdolreza Abdolhosseini Moghadam and Hayder Radha
Dep. of Electrical and Computer Engineering, Michigan State University, Email: {abdolhos,radha}@msu.edu

***Abstract*—Sparse projections for compressed sensing have been receiving some attention recently. In this paper, we consider the problem of recovering a k-sparse signal (x) in an n-dimensional space from a limited number (m) of linear, noiseless compressive samples (y) using *complex sparse projections*. Our approach is based on constructing complex sparse projections using strategies rooted in combinatorial design and expander graphs. We are able to recover the non-zero coefficients of the k-sparse signal (x) iteratively using a low-complexity algorithm that is reminiscent of well-known iterative channel decoding methods. We show that the proposed framework is optimal in terms of sample requirements for signal recovery $(m = O(k \log(n/k)))$ and has a decoding complexity of $O(m \log(n/m))$, which represents a tangible improvement over recent solvers. Moreover we prove that using the proposed complex-sparse framework, on average $2k < m \leq 4k$ real measurements (where each complex sample is counted as two real measurements) suffice to recover a k-sparse signal perfectly.**

***Index Terms*—Compressed Sensing, sparse projections, channel decoding**

I. Introduction

The design of a variety of new ensembles of projection matrices for compressed sensing (CS) [1]-[2] has been receiving a great deal of attention. In particular, the utility of sparse projections [7]-[11] for compressed sensing has shown to lead to low complexity decoding algorithms. To highlight these recent CS developments, let's consider the key parameters of a typical CS problem. Consider a length n signal s which has a k sparse real-valued decomposition x in a known basis $\Psi_{n\times n}$: $s = \Psi x$. By k sparse decomposition we mean that x is non-zero only in k coordinates: $k = \|x\|_0 := \#\{i : x_i \neq 0\}$. Instead of sensing n samples $(\{s_i\})$, under compressed sensing the encoder projects s into an incoherent frame $\phi_{m\times n}$ (with respect to the sparsifying domain Ψ) and senses m (where $k < m < n$) compressive samples y according to the projection (sensing) matrix $P = \phi\Psi$, i.e. $y = \phi s = Px$. Accordingly, the decoder recovers the sparse representation x (or equivalently s) by utilizing the compressive samples y and the projection matrix P. Generally the under-determined system of equations $y = Px$ has an infinite set of solutions. However, if certain properties outlined in [1]-[2] are met then it is possible to find a unique solution using the following optimization problem:

$$x = \arg\min \|\hat{x}\|_0 = \arg\min \|\hat{x}\|_1 : y = P\hat{x} \qquad (1)$$

Early CS approaches were primarily based on using dense projections in conjunction with some form of optimization or greedy algorithms [3]-[6]. In particular, an arguably benchmark approach for CS evolved around the utility of dense Gaussian projections in conjunction with an ℓ_1 minimization algorithm that is based on basis pursuit (BP) decoding for the recovery of the sparse signal x. This approach has shown to require the tight bound for the number of measurements $m = O(k \log(n/k))$ needed for the recovery of x [1]-[2] [6]. Despite the optimality of the BP solver (in terms of the measurement bound), the high complexity of this convex relaxation decoding approach has led to many efforts for the reducing the complexity of the CS decoder. More recently, *binary sparse projections* have received a great deal of attention, especially those that are based on expander graphs [9]-[10]. Recent efforts have shown that such projections, and in conjunction with belief-propagation type of algorithms, can lead to signal recovery using $m = O(k \log(n/k))$ measurements and with decoder complexity $O(n \log(n/k))$.

In this paper, we develop a new compressed sensing approach based on complex sparse projections that have some random structure. Hence, we refer to our approach as Complex Randomness-in-Structured Projection (CRISP) compressed sensing. CRISP, which enables a low complexity recovery for compressive sampling of k-sparse signals, is based on constructing sparse projections using strategies rooted in combinatorial design such as Balanced Incomplete Block Designs (BIBD) and expander graphs. Although there are some similarities between CRISP and recent sparse compressed sensing approaches [9][10], it is important to highlight that our utility of *complex* sparse projections lead to decoding framework that is different from sparse-projection based decoding methods. The key idea of our proposed combinatorial approaches for compressed sensing is finding samples spanning one non-zero coefficient and then iteratively recovering other non-zero coefficients. This approach is reminiscent of iterative erasure channel-decoding algorithms that recover degree one lost symbols, and then higher degree symbols are recovered from lower degree symbols. Meanwhile, recent CS approaches are based on using binary projection matrices (e.g., based on regular LDPC matrices and expander graphs) in conjunction with belief-propagation type decoding algorithms [7]-[11].

The contributions of this papers are 1) generalizing the combinatorial approaches from binary matrices to complex valued matrices and reducing the complexity furthermore (while keeping the sample requirement $O(k \log(n/k))$ fixed) and 2) providing an average behavior analysis for our proposed method. Our simulation results in section III verify our analysis, and they show that under CRISP, *on average* $2k < m \leq 4k$ real valued samples are sufficient to recover a k sparse signal perfectly. We also show that the complexity of the CRISP solver is $O(m \log(n/m))$ which is faster than similar CS solutions. Section IV concludes the paper.

II. Sparse Projection Matrices

It has been shown [1] that $m \geq O\left(k \log(n)\right)$ compressive samples ($y = Px$) are sufficient to recover x, a k-sparse signal of length n using linear programming algorithms such as Basis Pursuit (BP) [6]. Despite the tractability of BP, its complexity ($\Omega(n^3)$) is so high that many alternative approaches [3]-[5] [7]-[11] have been proposed to lower the complexity of the solver even at the price of increasing the number of compressive samples.

In contrast to traditional CS solutions, where each sample is a function of all signal coefficients, a sparse-projection based method naturally projects different (small) subsets $\chi = \{\chi_1, \chi_2, \ldots, \chi_m\}$ of the signal x into the observations $y = \{y_1, y_2, \ldots, y_m\}$. Here, χ is a superset where each of its members is a subset of x: $\forall i : \chi_i \subset x = \{x_1, \ldots, x_n\}$. For example, if the i^{th} row of a sparse projection matrix P has zero entries everywhere except for non-zero entries in some locations, say $M_i = \{3, 8, 31\}$, then we have a corresponding subset $\chi_i = \{x_3, x_8, x_{31}\} = x_{M_i}$. Consequently, the i^{th} compressive sample y_i is a function of the subset χ_i ($y_i = g_i(\chi_i)$), and y_i does not depend on the values of other elements of the signal x[1].

Meanwhile combinatorial algorithms for compressed sensing are based on the (provable) premises that a sparse projection matrix (P) *disperses* non-zero coefficients of the signal x into different compressive samples ($y = Px$). Now if P is a binary matrix and the values of some numbers of compressive samples are exactly the same and these samples span one common coefficient, then that (common) coefficient is non-zero and can be recovered easily. Subtracting the effect of the identified coefficients from the vector of compressive samples (y) and repeating this process iteratively leads to identifying the unique solution x. Although the *empirical* number of samples which combinatorial algorithms require for the perfect recovery is not as good as BP (at least for all signal classes), there exist combinatorial algorithms guaranteeing the exact recovery from an optimal *order* of compressive samples ($O\left(k \log(n/k)\right)$) with a complexity of $O\left(n \log(n/k)\right)$ [10].

For the class of combinatorial algorithms, the main reason for choosing binary projection matrices (besides keeping the complexity of encoding low) is to keep the complexity of the decoder low. Hence it seems that employing any non-binary projection matrix would worsen the decoder complexity. On contrary, in this paper, we show that employing complex numbers in the projection matrix would decrease the solver complexity (significantly). Using well established results in case of random sparse matrices and expander graphs [10], we show that our proposed algorithm is optimal in terms of order of required samples for the perfect recovery. Then we present a simple proof showing that our proposed method on average requires only $2k < m \leq 4k$ samples for the perfect recovery and the solver complexity is only $O\left(m \log(n/m)\right)$.

The following notation will be used in this paper. Let A be a $b \times c$ matrix and let $i \subseteq [b] = \{1, \ldots, b\}$ and $j \subseteq [c]$. A^T represents the transpose of A. By $A_{b,}$ and $A_{,c}$ we respectively mean submatrices of A given by restricting A on rows and columns indexed by those members of b and c. Moreover let D be a $d \times 1$ vector and let $e \subseteq [d]$. Then $\hat{D}_e$ is a $d \times 1$ vector given by keeping the entries of D indexed by e and setting the rest (indices of $[d] \backslash e$) to zero.

A. CRISP Projection Matrices

As before, let us denote the number of rows of the projection matrix P by m and the number of columns of the projection matrix (or equivalently the signal length) by n. We form the final projection matrix P from a *base projection matrix* $p_{m \times n}$. Under the proposed CRISP framework, the base projection matrix p has binary entries (ones and zeros), and hence, this base matrix provides an underlying structure that can be used to populate the "ones" entries of p with random complex numbers (as explained later). This process of replacing the "ones" of p by complex numbers generates the final projection matrix P. Consequently, the underlying "structure" (and associated sparsity) of both the base matrix p and the final projection matrix P is the same. One straightforward approach for constructing p is to randomly select the entries of p where "ones" are placed (Algorithm 1). This simple approach does not impose any constraints on the underlying structure of p except for the parameter K, which is the number of "ones" that are placed randomly in each column of p. Note that K has to be small enough (relative to m and n) in order to have a sparse underlying structure for the (final) projection matrix P. Note that the resulting p (in this case) may be thought as the (transpose of) incidence matrix of a bipartite left regular graph of degree K. Hence with high probability [10][13] p corresponds to an expander graph.

```
Input: $m$, $n$ and $K \geq 2$
Output: The base projection matrix $p$
for $i = 1$ to $n$ (all columns of $p$) do
    Select $K$ distinct random integers ($l = \{l_1, \ldots, l_K\}$)
    uniformly from $[m] = \{1, \ldots, m\}$
    For the $i^{th}$ column of $p$, set the entries in the rows
    indexed by $l$ to one and put zero in other indices:
    $p_{e,} = \hat{\mathbf{1}}_l$
end
```

Algorithm 1: Generating the base projection matrix p.

One might impose certain constraints on the underlying structure of the base projection matrix which leads to the same structure for P. Most notably, approaches that are inspired by channel coding methods for constructing parity check matrices, and in particular ones related to Low-Density-Parity-Check (LDPC) codes can be used for the binary base matrix

[1] We say that the i^{th} measurement sample spans the indices of M_i.

p. For instance, one might utilize the incidence matrices of Balanced Incomplete Block Designs (BIBD) [12] to improve the performance of the CRSIP framework as we explain further below in this paper.

In general, one can perform some form of matrix manipulation of p prior to replacing the "ones" of p onto some complex numbers. Further, one can have certain constraints on the complex numbers used in the projection matrix P. In this paper, we focus on applying two steps to the base matrix p, random permutation and employing complex numbers with unique phases. For example, the overall approach used when permuting (uniformly) the columns of p and substituting the non-zero entries of p by normalized complex numbers with unique phases in range of $[0, \pi)$ is shown below (Algorithm 2).

```
Input: The base projection matrix $p$
Output: The projection matrix $P$
Permute randomly the columns of $p$
for $i = 1$ to $m$ (all rows of $P$) do
    Let $M_i$ be the non-zero entries of the $i^{th}$ row of $p$ and let $w = |M_i|$.
    Generate a row vector $f^{(i)}$ composed of $w$ normalized complex numbers with unique phases in range of $[0, \pi)$.
    For the $i^{th}$ row of $P$, insert $f^{(i)}$ in the column indices of $M_i$: $P_{i,M_i} = f^{(i)}$ and $\forall l \notin M_i : P_{i,l} = 0$
end
```

Algorithm 2: Generating the projection matrix from the base projection matrix.

Now we say few words regarding the BIBD design procedure. As we eluded above, the incidence matrix of a Balanced Incomplete Block Design [12] represents a special case for the design of the base projection matrix p. Here, we briefly review a few key properties of such matrices. Let m, n, K and λ be positive integers such that $m > K \geq \lambda$. The incidence matrix of a (m, n, K, λ)-BIBD is a $m \times n$, matrix with the following properties:

- Each row is non-zero exactly in $w = \lambda(m-1)/(K-1)$ column indices. Also each of the $n = mw/K$ columns is non-zero exactly in K entries[2] and thus we have $\forall i, j \in [m] : w = |M_i| = |M_j|$. Note that this matrix corresponds to the (transpose of) incidence matrix of a bipartite left-regular of degree K and right-regular of degree w.
- If we consider two distinct rows of this matrix, both are non-zero in exactly λ columns. That is: $\forall i, j \in [m], i \neq j : |M_i \cap M_j| = \lambda$.

Due to the special characteristics (listed above) of the proposed projection matrix, the following properties hold: 1) Since each column of our projection matrix is non-zero in K row indices, hence each signal coefficient x_i appears exactly in K random (different) equations/samples. 2) If the base projection matrix p is an incidence matrix of a BIBD then two distinct equations (compressive samples) have exactly λ common coefficients/unknowns.

B. CRISP Solver

We first highlight an important (yet arguably clear) proposition, which is critical for the successful working of the HCS solver: if a subset of the coefficients has at most one non-zero coefficient, then only one complex valued random compressive sample is sufficient for the perfect recovery of that subset.

Proposition 1. *Let $M_i \subset [n], |M_i| = w$ be the (ordered) set of indices of non-zero coefficients for the i^{th} row of a $P^{(c)}$. And let $P^{(c)}_{i,} = f^{(i)} = \left[\, e^{j\phi^{(i)}_1} \ldots e^{j\phi^{(i)}_w} \,\right]$ be the corresponding vector of complex exponentials (on unit circle) such that the phases of $f^{(i)}$ are distinct in the range of $[0, \pi)$, $\forall l, t \in [w], l \neq t : \phi^{(i)}_l \in [0, \pi), \phi^{(i)}_t \neq \phi^{(i)}_l$. If x_{M_i} is non-zero in at most one index ($\|x_{M_i}\|_0 \leq 1$) then having $y_i = P^{(c)}_{i,} x = f^{(i)} x_{M_i}$ and $f^{(i)}$, one can recover x_{M_i} deterministically. Moreover having y_i and $f^{(i)}$, we can detect the event that x_{M_i} is non-zero in at least two indices. Specifically if $y_i = 0$ then $x_{M_i} = \mathbf{0}$. If the phase of y_i equals to the phase of l^{th} entry of $f^{(i)}$ ($\angle y_i = \angle f^{(i)}_l = \phi^{(i)}_l$) then x_{M_i} is non-zero only in the index of $M_{i,l}$ and has the value of $x_{M_{i,l}} = y_i / f^{(i)}_l = \|y_i\|$. If the phase of y_i is not among the phases of $f^{(i)}$ ($\{\phi^{(i)}_1, \ldots, \phi^{(i)}_w\}$), then x_{M_i} is non-zero in at least two indices. Proof is presented in the Appendix.*

The decoding scheme of the CRISP solver is iterative: we examine compressive samples to find some subsets of the signal spanning only one non-zero coefficient. This can be achieved by looking for the phases of the compressive samples ($\angle y = \{\angle y_1, \ldots, \angle y_m\}$) among the phases of the entries of the random vectors $f^{(\cdot)}$. For instance, assume that $\angle y_i$ equals to $\angle f^{(i)}_l$. Therefore using Proposition 1, one can determine the signal values in the i^{th} subset (x_{M_i}). Subtracting the effect of the identified non-zero coefficient from the compressive samples leads to the following new under-determined system of equations: $y^{(1)} = y - P\hat{x}_{M_i} = P\hat{x}_{[n]/M_i}$. Note that $\|\hat{x}_{[n]/M_i}\|_0 = \|x\|_0 - 1$. Since there are exactly K distinct samples spanning the same non-zero coefficient, this reduction affects K more samples (equations). Specifically, if the base projection matrix p is the incidence matrix of a BIBD, then identifying the signal values in one subset reduces the number of unknowns (un-identified coefficients) in all equations $y_i = \sum_{e=1}^{w} f^{(i)}_e x_{M_{i,e}}$ by λ. Therefore the algorithm converges to the solution very fast. More specifically, consider a compressive sample y_t spanning two non-zeros including $x_{M_{i,l}}$: $y_t = \alpha x_{M_{i,l}} + \beta x_e,\ e \in [n]/M_{i,l}$, where α and β are two distinct entries of $f^{(t)}$. Thus $y^{(1)}_t = y_t - P_{i,} x_{M_i} = \beta x_e$. Consequently after this reduction, all compressive samples spanning two non-zeros such that one of these non-zeros is among the identified coefficients in the first iteration, turn into new samples spanning only one non-zero coefficient. Therefore in the next iteration, the phases of these compressive samples can be found among the available phases in the

[2] Note that the size of support of a signal (or sparsity measure) is denoted by k, while K is a parameter of a BIBD.

random vectors $f^{(\cdot)}$. This event results in identifying the signal values in new subsets (and consequently new coordinates). Algorithm 3 shows the CRISP decoding process.

```
Input: $y$ and $P$
Output: the sparse vector $\bar{x}$ which $y = P\bar{x}$
Set $I = 0$, $y^{(0)} = y$ and $\bar{x} = \mathbf{0}$.
repeat
    Find all indices $\beta = \{i : \angle y_i^{(I)} \in \angle f^{(i)}\}$
    for all $i \in \beta$ do
        Suppose $\angle y_i^{(I)} = \angle f_l^{(i)}$ then:
        $\bar{x}_e = \left\{ \begin{array}{cc} \|y_i^{(I)}\| & e = M_{i,l} \\ 0 & e \in M_i/M_{i,l} \end{array} \right\}$
        $y^{(I+1)} = y^{(I)} - P\hat{\bar{x}}_{M_i}$
    end
    $I = I + 1$
until $y^{(I)}$ is a zero vector ;
```

Algorithm 3: CRISP solver

C. Optimality

As stated before, the projection matrix (P) inherits best properties of the base projection p. Therefore, by design and with high probability, P corresponds to (randomly permuted) transpose of the incidence of an expander graph. It has been shown in [10] that with high probability, having $m = O\left(k \log n/k\right)$ compressive samples, 1) in the first iteration of the solver, there exists at least $1+[K/2]$ samples spanning only one specific non-zero coefficient (where K is the number of non-zero entries in each column of the projection matrix) and 2) the algorithm never halts and finally 3) it converges in $O(k)$ iterations. Therefore the algorithm is optimal in terms of the *order* of the number of samples for perfect recovery. However now instead of real valued samples, we are sensing complex samples which (albeit keeping the order of m fixed) doubles the number of sensed real samples (each complex sample is counted as two real measurements). On the other hand, as opposed to traditional combinatorial algorithms which require at least $1 + [K/2]$ samples that span only one common non-zero coefficient (for a voting like decoding algorithm), here we only need one sample (instead of $1 + [K/2]$) that spans one non-zero, which in turn lowers the required number of samples for perfect recovery. This significant reduction in the required samples is attributed to our utility of unique complex phases that make it feasible to detect and identify isolated non-zero coefficients. And finally, this generalization (from binary matrices to matrices with complex entries) reduces the complexity of the solver significantly from $O\left(n \log(n/k)\right)$ to $O\left(k \log(n/k) \log(n/m)\right)$.

Recall that the first iteration of the solver we look for the phases of compressive samples among a predetermined set of phases (available phases in each row of the projection matrix). Assume for each row of P, non-zero entries are sorted by their phases:

$$a, b \in [w], a > b \Leftrightarrow \angle f_a^{(i)} = \phi_a^{(i)} > f_b^{(i)} = \angle \phi_b^{(i)} \tag{2}$$

Then one might utilize a quick search algorithm (such as binary search) in the CRISP solver. Now for each sample, we need to search among $w = O(n/m)$ distinct phases and the complexity of such search is only $\log(w)$. Hence the first iteration of CRISP has a complexity of order $O\left(m \log(w)\right)$. For the next iteration of the solver, we need to subtract the effect of identified coefficients from the compressive samples. Since each non-zero coefficient appears in K (distinct) samples, hence the complexity of such update is $O(K)$ which is a constant. However the algorithm converges in $O(k)$ iterations, hence the overall complexity of CRISP is $O\left(m \log r + kK\right) = O\left(m \log(n/m)\right)$.

D. The Average Behavior Analysis

In this subsection, we present the proof for our claim that CRISP on average needs only $k < m \leq 2k$ complex (or equivalently $2k < m \leq 4k$ real) valued compressive samples for the perfect recovery of a k-sparse signal.

Lemma 1. *On average, CRISP requires $2k < m \leq 4k$ real-valued compressive samples for the perfect recovery of a k sparse signal of length n if $k \gg 1$.*

Proof. Let $C = m/k$ be the oversampling factor. Then the probability that a given sample y_j spans exactly i non-zeros is:

$$\Pr\left(\|x_{M_j}\|_0 = i\right) = \Lambda_i = \frac{\binom{k}{i}\binom{m-1}{K-1}^i \binom{m-1}{K}^{k-i}}{\binom{m}{K}^k} \tag{3}$$

It is important to note that since the columns of CRISP projection matrix are permuted randomly, hence the probability that any given sample spans i non-zero coefficients, is constant for all samples: $\forall l, j \in [m] : \Pr\left(\|x_{M_j}\|_0 = i\right) = \Pr\left(\|x_{M_l}\|_0 = i\right) = \eta_i$. Thus the expected number of samples, spanning only one non-zero is $m\eta_1$. To simplify our analysis, let us consider the asymptotic case where k (and hence m and n) tends to infinity. Then the expected number of samples spanning only one non-zero would be:

$$\underset{k \to \infty}{E\left(\#j : \|x_{M_j}\|_0 = 1\right)} \approx kKe^{-\frac{K}{C}} \tag{4}$$

Note that for large values of k and fixed values of $m = Ck$, the expected number of samples spanning only one non-zero is a linear function of k. These samples identify the locations of isolated non-zero coefficients which consequently lead to the recovery of the associated subsets of coefficients.

Here we should emphasize that not all of these isolated coefficients provide new information. For instance, it is possible that there are several samples spanning only one specific non-zero coefficient x_i. So it is important to exclude this redundant information from our analysis. Let $\mho$ be the probability of the event that one sample y_a spans only one non-zero coefficient x_i, given that there exists another compressive sample y_b ($b \neq a$) spanning the same coefficient x_i. Then we have:

$$\mho = \frac{k(m-1)\binom{m-2}{K-2}\binom{m-2}{K}^{k-1}}{\binom{m}{K}^k} \tag{5}$$

That is choose a non-zero coefficient x_i among k non-zeros and choose another sample y_a (except y_b) and let y_a be only function of x_i. Now randomly select $K-2$ other samples which would be functions of x_i and distribute the remaining $k-1$ non-zeros (counting their K repetitions) among remaining $m-2$ samples. Fix $m = Ck$, for large values of k (5) can be approximated to:

$$\underset{k\to\infty}{\mho} \approx \frac{k(K^2-K)}{m}e^{-\frac{2K}{C}} \tag{6}$$

which is a linear function of sparsity measure k. Thus the expected number of samples which do not result to discovery of new coefficients (μ) is:

$$\mu = \frac{m}{K}\mho \approx \frac{m}{K}\frac{kK(K-1)}{m}e^{-\frac{2K}{C}} = k(K-1)e^{-\frac{2K}{C}} \tag{7}$$

Recall that (on average) $kKe^{-\frac{K}{C}}$ samples span only one non-zero coefficient. Hence, the expected number of compressive samples spanning distinct non-zeros in the first iteration of the solver is:

$$\mu' = kKe^{-\frac{K}{C}} - \mu = k\left(Ke^{-\frac{K}{C}} - (K-1)e^{-\frac{2K}{C}}\right) \tag{8}$$

In words, CRISP solver (on average) recovers μ' non-zero coefficients in the first iteration. Since the columns of the CRISP projection matrix are permuted independently and uniformly, we can say that in the next iteration, there are $k^{(1)}$ non-zeros left to recover from m samples, where:

$$k^{(1)} = k - \mu' = kq,\ q = \left(1 - Ke^{-\frac{K}{C}} + (K-1)e^{-\frac{2K}{C}}\right) \tag{9}$$

Note that for fixed values of $C = m/k$, q is independent of k. It is straightforward to see that, (on average) in the i^{th} iteration of the algorithm, there are: $k^{(i)} = kq^i$ non-zero coefficients which have not been recovered. Also note that even for $1 < C \leq 2$, we have $q < 1$. In other words, on average, $k < m \leq 2k$ complex valued samples (and hence $2k < m \leq 4k$ real valued samples) is sufficient for the perfect recovery. ■

III. Simulation Results

We tested the proposed CRISP on a large number of standard signals from SparseLab [14]. Specifically we compared the performance of CRISP (in terms of quality of the recovered signal as a function of the number of compressive samples and the required time for recovery) in comparison with the popular Gaussian random projection matrix and dominant Basis Pursuit (BP) and OMP solvers. In all plots m, n and k denote the number of compressive samples, the signal length and the sparsity of the signal in the sparsifying domain (Wavelet/DCT). To provide fairness, we have counted each complex sample as two samples. We performed our simulations under various configurations of signal length and sparsity ratio which we present some of them in Fig. 1. As clearly demonstrated CRISP has significantly lower complexity compare to BP and OMP.

Here we should highlight an important note: due to the strong structures and rich properties of BIBDs, we expect that a sensing matrix generated based on an incidence matrix of a BIBD boosts CRISP performances in terms of the solver complexity and also the sample requirements for the perfect recovery. For instance, in Fig. 1b, we have used a Hamming matrix of weight two (all m tuples with $K = 2$ non-entries in each column) as the base projection matrix p. Hence p is an incidence matrix of a $(m, m(m-1)/2, 2, 1)$-BIBD. In that simulation, we observed that if the incidence of a $(200, 4950, 2, 1)$-BIBD was used as the base projection matrix, then $m = 4k$ (real valued) samples was enough for the perfect recovery. On the other hand, if we have used Algorithm 1 to generate the base projection, then $m \approx 6k$ samples was required for the perfect recovery and also the decoding time would increase as well. However we should recall that: although there are numerous methods for generating a BIBD with some given parameters [12], it is possible that there are no BIBD designs that exist in some configurations of (m, n, K, λ).

IV. Conclusion

In this paper, we showed that if we sense one complex valued compressive sample from random subsets of a sparse signal, such that any two subsets intersect in a few indices, then the recovery algorithm is optimal in terms of sample requirements for perfect recovery ($O\left(k\log(n/k)\right)$). Moreover, on average we only need $2k < m \leq 4k$ compressive samples to recover a signal from these samples. The complexity of CIRSP is $O\left(m\log(n/m)\right)$ which is lower than the complexity of similar combinatorial algorithms.

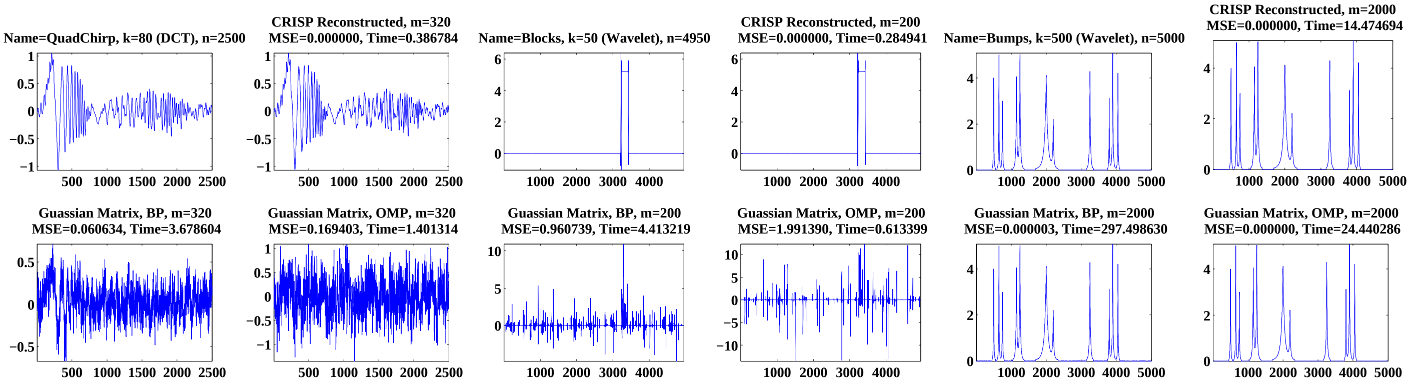
V. Appendix

Proof of Proposition 1 Clearly if x_{M_i} is a zero vector then $y_i = 0$. Now assume $y_i = 0$. We show that with high probability $x_{M_i} = \mathbf{0}$. Note that if x_{M_i} is non-zero only in index of $M_{i,l}$ ($\|x_{M_i}\|_0 = 1$ and $x_{M_{i,l}} \neq 0$) then $y_i = x_{M_{i,l}}e^{j\phi_l^{(i)}} \neq 0$. Now suppose x_{M_i} is non-zero in at least two indices ($\|x_{M_{i,l}}\|_0 \geq 2$). Hence y_i is a linear combination of at least two independent complex numbers. Suppose the values of all entries of x_l are bounded by η, i.e. $\forall l : |x_l| < \eta,\ \eta > 0$. Then the magnitude of y_i is a continuous random variable (at least) in the range of $[0, \eta)$. Thus the probability of the event that the magnitude of y_i takes *exactly* the value of zero, is zero. Hence with high probability: $x_{M_i} = \mathbf{0} \Leftarrow y_i = 0$ and therefore $x_{M_i} = \mathbf{0} \Leftrightarrow y_i = 0$.

Now assume x_{M_i} is non-zero only in the index of $M_{i,l}$. Hence: $y_i = x_{M_{i,l}}e^{j\phi_l^{(i)}}$. Note that in this case, the phase of y_i (i.e. $\angle y_i = \phi_l^{(i)}$) can be found in the ordered set of available phases in $f^{(i)}$ ($\angle f^{(i)} = \{\phi_1^{(i)}, \ldots, \phi_w^{(i)}\}$). In other words, knowledge of the phase $\angle y_i = \phi_l^{(i)}$ translates into knowledge of the index l associated with the location of the (single) non-zero coefficient $x_{M_{i,l}}$. Consequently having y_i and $f^{(i)}$, we can recover x_{M_i} by:

$$\angle y_i = \angle f_l^{(i)} \Leftarrow x_u = \left\{ \begin{array}{cc} \|y_i\| & u = M_{i,l} \\ 0 & u \in M_i \backslash M_{i,l} \end{array} \right\} \tag{10}$$

Now we prove (by contradiction) that if the phase of y_i can be found in $\angle f^{(i)}$ then with high probability x_{M_i} is non-zero exactly in one index. Suppose x_{M_i} is non-zero in at least two



(a) (b) (c)

Fig. 1: m, n and k represent number of samples, signal length and signal sparsity respectively.

indices and $\angle y_i$ can be found among $\angle f^{(i)}$. By the problem setup, $\angle y_i$ is a continuous random variable in the range of $[0, \pi)$. However, we know that the probability of the event that a continuous random variable ($\angle y_i$) attains the values of a finite set of numbers ($\angle f^{(i)}$), is zero and this is a contradiction. Thus:

$$\angle y_i = \angle f_l^{(i)} \Leftrightarrow x_u = \left\{ \begin{array}{cc} \|y_i\| & u = M_{i,l} \\ 0 & u \in M_i \backslash M_{i,l} \end{array} \right\} \tag{11}$$

Finally if $y_i \neq 0$ and the phase of y_i is not found in the ordered set $\angle f^{(i)}$, then x_{M_i} is non-zero at least in two indices. ■

Acknowledgment

This work was supported in part by NSF.

References

[1] Emmanuel Cands, Justin Romberg, and Terence Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information". IEEE Trans. on Information Theory, 52(2) pp. 489 - 509, February 2006.

[2] David Donoho and Yaakov Tsaig, "Extensions of compressed sensing". (Signal Processing, 86(3), pp. 533-548, March 2006).

[3] David L. Donoho, Yaakov Tsaig, Iddo Drori, and Jean-Luc Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit". Preprint, 2007.

[4] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples". Preprint, 2008.

[5] Y. C. Pati R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition", Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers, Nov. 1993.

[6] S. Chen, , D. Donoho, and M. Saunders. "Atomic decomposition by basis pursuit". Technical Report 479, Department of Statistics, Stanford University, May 1995.

[7] Radu Berinde and Piotr Indyk, "Sparse recovery using sparse random matrices", Preprint, 2008.

[8] Shriram Sarvotham, Dror Baron, and Richard Baraniuk, "Sudocodes - Fast measurement and reconstruction of sparse signals". Proc. IEEE Int. Symposium on Information Theory (ISIT), Seattle,Washington, July 2006.

[9] Piotr Indyk and R. Berinde "Sequential Sparse Matching Pursuit", Allerton, 2009.

[10] Sina Jafarpour, Weiyu Xu, Babak Hassibi, and Robert Calderbank, "Efficient compressed sensing using high-quality expander graphs". to appear in IEEE Trans Info Theory, 2009.

[11] Dror Baron, Shriram Sarvoham, Richard G. Baraniuk, "Bayesian compressive sensing via belief propagation". (To appear in IEEE Trans. Signal Processing, 2009)

[12] "Combinatorial Designs: Construction and Analysis" by Douglas R. Stinson, Springer-Verlag, New York inc, 2004.

[13] L. Bassalygo and M. Pinsker, "Complexity of an Optimum Nonblocking Switching Network without Reconnections", Problem in Information Transmission, vol 9 no 1, pp. 289-313, 1973.

[14] http://sparselab.stanford.edu