On combinatorial approaches to compressed sensing

Abdolreza Abdolhosseini Moghadam and Hayder Radha
Department of Electrical and Computer Engineering,
Michigan State University, East Lansing, MI, U.S.
Emails: {abdolhos,radha}@msu.edu

***Abstract*—In this paper, we look at combinatorial algorithms for Compressed Sensing from a different perspective. We show that certain combinatorial solvers are in fact recursive implementations of convex relaxation methods for solving compressed sensing, under the assumption of sparsity for the projection matrix. We extend the notion of sparse binary projection matrices to sparse real-valued ones. We prove that, contrary to their binary counterparts, this class of sparse-real matrices has the Restricted Isometry Property. Finally, we generalize the voting mechanism (employed in combinatorial algorithms) to notions of isolation/alignment and present the required solver for real-valued sparse projection matrices based on such isolation/alignment mechanisms.**

Keywords: Compressed sensing, combinatorial approaches

I. INTRODUCTION

The theory of Compressed Sensing (CS)[1][2] deals with finding the unique sparse (or compressible) signal $x \in \mathbb{R}^n$ from a limited number of linear samples $y = \boldsymbol{P}x$ where $\boldsymbol{P} \in \mathbb{R}^{m \times n}$ and $m < n$. Roughly speaking, there are three approaches to finding the solution x: convex relaxation methods [2][3][4], greedy algorithms [5] and combinatorial approaches [6][7][8][11]. Convex relaxation methods find the unique sparsest solution to the under-determined system of linear equations $y = \boldsymbol{P}x$ by solving a convex optimization problem in the form of:

$$\arg\min_{\hat{x}} \|\hat{x}\|_p \;\; s.t. \;\; y = \boldsymbol{P}\hat{x} \qquad (1)$$

where $p = 1$. Although being tractable and demanding the fewest number of samples/equations (m) to guarantee perfect recovery, solving such linear program generally could be very complex (e.g. $O(n^3)$ for Basis Pursuit [3]). On the other extreme, combinatorial algorithms have much less complexities (compared to convex relaxation methods) but at the cost of higher sample/equation requirements. The low computational costs of combinatorial algorithms stem from the fact that these approaches are based on binary sparse projection matrices $\boldsymbol{P}$ and light voting-like decoding stages.

Although there have been some works (e.g.[8]) highlighting connections between convex relaxation methods and combinatorial algorithms, in this paper we present a different perspective onto such link. We show that, a typical combinatorial decoder to CS is in fact a simple recursive ℓ_1 minimizer in case of a binary projection matrix. Then, we answer the following question: what will happen if non-zero entries of the binary projection matrix $\boldsymbol{P}$ are replaced by (specifically Gaussian) random variables? We prove that, such modification provides us RIP-2 [2] in case of sparse projection matrices, a property which binary sparse projection matrices do not possess [10] for $m = O(k \log n/k)$ where k is the sparsity of the signal x ($k = \|x\|_0 = \#i: x_i \neq 0$). Unfortunately, this modification (going from a sparse binary matrix to a sparse real-valued one) implies that the voting mechanism of combinatorial approaches [6][7] will no longer work in case of such new projection matrices. Consequently, we present concepts of "alignment" and "isolation" for generic (real-valued) sparse projection matrices. We show that the voting mechanism is in fact a special implementation of those concepts in case of binary projections. Finally, we present necessary and sufficient conditions under which the new voting mechanism (alignment/isolation) is guaranteed to find the sparsest solution.

This paper is structured as follows: In section II, we show how a basic combinatorial algorithm could be viewed as a recursive ℓ_1 minimizer and extend the notion of using sparse binary projections to sparse real-valued matrices. In section III, we prove that RIP-2 holds for such new class of projection matrices. Concepts of alignment/isolation and conditions for their equivalency are presented in Section IV. Simulation results are brought in Section V and Section VI concludes this paper.

II. A GENERAL FRAMEWORK FOR CS COMBINATORIAL APPROACHES

First, we introduce the notation used in this paper. For $q \in \mathbb{N}$, define $[q] \coloneqq \{1,2,\dots,q\}$. For a projection matrix $\boldsymbol{P} \in \mathbb{R}^{m \times n}$ and for a row index $i \in [m]$, define $\omega_i = \{j: \boldsymbol{P}_{i,j} \neq 0\}$. Hence, ω_i is the set of column indices where the i-th row of $\boldsymbol{P}$ is non-zero. We generalize such notation for any set of rows $i \subseteq [m]$ as: $\omega_i = \bigcup_{j \in i} \omega_j$. The set of row indices where the i-th column of $\boldsymbol{P}$ are non-zero, is Ω_i, i.e. $\Omega_i = \{j: \boldsymbol{P}_{j,i} \neq 0\}$. Similarly, one can generalize this notation for any set of columns $i \subseteq [n]$ by: $\Omega_i = \bigcup_{j \in i} \Omega_j$. A zero vector of length l is denoted by $\mathbf{0}_l$.

Throughout this paper, we consider the class of combinatorial algorithms in their most basic form. Hence, our discussion excludes message passing algorithms [9] and we focus only on those algorithms which estimate signal values at each coordinate only once (e.g. [6]).

Algorithm 1 shows how a typical combinatorial approach to CS operates for solving (1) under the zero pseudo norm $p = 0$. It can be easily verified that Algorithm 2 could be perceived as the recursive formulation of Algorithm 1 when p is set to zero in the third line of Algorithm 2.

Inputs : $\boldsymbol{P} \in \mathbb{R}^{m\times n}$and $y \in \mathbb{R}^m$
Output : $\hat{x} \in \mathbb{R}^n$
1) $\hat{y} = y, z = \emptyset$;
2) **While** $z \neq [n]$
3) Find $l \subseteq [m]$ such that $\|x_{\omega_l}\|_0 = 1$
4) Let $\hat{x}_{\omega_l}$ denotes estimate of values x_{ω_l}
5) For $\theta = \omega_l$: $\hat{y}_{\Omega_\theta} \leftarrow \hat{y}_{\Omega_\theta} - P_{\Omega_\theta,\omega_l}\hat{x}_{\omega_l}$, $z \leftarrow z \cup \omega_l$
6) **End**

Algorithm 1. A typical combinatorial algorithm for CS. The set of coefficients which their values are determined, is denoted by $\boldsymbol{z}$.

Inputs : $\boldsymbol{P} \in \mathbb{R}^{m\times n}, \theta$ and $y \in \mathbb{R}^m$
Output : $\hat{x} \in \mathbb{R}^n$
1) $\hat{x} \leftarrow \mathbf{0}_n$
2) Find $l \subseteq [m]$ and $\theta_1 = \omega_l \subseteq \theta$ such that $\|x_{\theta_1}\|_0 = 1$
3) $\hat{x}_{\theta_1} = arg\ min_{\{\xi\}}\|\xi\|_p$ subject to $y_l = \boldsymbol{P}_{l,\theta_1}\xi$
4) $\hat{x}_{\theta/\theta_1} \leftarrow \Delta(\boldsymbol{P}, \theta/\theta_1, y - \boldsymbol{P}\hat{x})$

Algorithm 2. $\boldsymbol{\Delta(P, \theta, y)}$ the recursive formulation of Algorithm 1.

Initializing with $\theta = [n]$ and after finding θ_1 such that $\|\hat{x}_{\theta_1}\|_0 = 1$ (finding these subsets is discussed in Section IV), the algorithm recursively finds $\theta_{i+1} \subseteq [n]/\cup_{j\in[i]}\theta_j$ such that $\|\hat{x}_{\theta_{i+1}}\|_0 = 1$ where θ_j denotes the value of θ_1 (line 2 in Algorithm 2) in the j-th level of the stack. This process continues until all non-zero coefficients are recovered. Note that, for a vector ξ with a single non-zero entry ($\|\xi\|_0 = 1$) and $z = \boldsymbol{P}\xi$, the solutions to (P1) for $p = 0$ and $p = 1$ are equal by the triangle inequality:

$$\text{(P1)}\ \xi_1 = \arg\min_{\hat{\xi}} \|\hat{\xi}\|_p\ s.t.\ z = \boldsymbol{P}\hat{\xi} \tag{2}$$

Also note that for any signal x and two disjoint subsets of $a \subseteq [n]$ and $a^c = [n]/a$ one has:$\|x_a\|_p + \|x_{a^c}\|_p = \|x\|_p$both for $p = 0$ and $p = 1$ (see line 3 and 4 in Algorithm 2). Combining this observation with the equality of solutions of (P1) for $p = 0$ and $p = 1$, one can conclude that in Algorithm 2, the solutions for$p = 0$ and $p = 1$ are exactly the same. Hence, Algorithm 1 could be recast as a recursive ℓ_1 minimizer. The success of the aforementioned recursive algorithm depends on: (a) the ability of step 3 in Algorithm 1 to find *isolated non-zeros* and (b) the quality of the projection matrix $\boldsymbol{P}$ in *recoverability*. More specifically, the first condition demands that the decoder at the i-th iteration could truly find $\theta_i = \omega_l$, a subset of signal coefficients, containing only one non-zero ($\|x_{\theta_i}\|_0 = 1$). Meanwhile, the second requirement states that, at least one isolated non-zero must exist throughout the recovery process and hence such process shall not be halted before the perfect recovery. It has been shown [7] that certain sparse matrices, for instance the adjacency matrices of high quality expander graphs provide the second condition. In the rest of this paper, we assume that the employed sparse projection matrix has the second condition, meaning that if isolated non-zeros are correctly identified in step 3 of Algorithm 1, then the recovery process will not halt. To that end and to make our analysis simple, we adapt the adjacency matrices of random left regular expander graphs [7]. That is, each column of $\boldsymbol{P}$ is non-zero in exactly $d = O(\log\frac{n}{k})$ random indices. However, we extend the notion of using binary sparse projection matrices to random (real valued) sparse projections. More specifically, we simply substitute the non-zero entries of a sparse binary $\boldsymbol{P}$ (i.e. entries with values of one) with a zero-mean Gaussian random variable with variance of $1/d$. We show that by such simple modification, RIP-2 will hold true in case of the proposed $\boldsymbol{P}$, the property which binary sparse projection matrices do not possess [10] (for $m = k\log n/k$).

Having the guarantee of recoverability, now our concern will be to satisfy the first condition, i.e. correctly finding isolated non-zeros. The most common approach for finding isolated non-zeros (in case of a binary sparse projection matrix) is to apply a voting mechanism (see [6][7]). More specifically, assume that there exists a set of sample indices of $l \subseteq [m]$ such that all of them vote the same value, i.e.: $\forall p, q \in l: y_p = y_q$ and they span the same single coefficient (i.e. $\cap_{j\in l}\ \omega_j = i$). Then typical combinatorial algorithms infer that $\hat{x}_i$ equals to that vote and $\hat{x}_j = 0$ for $j \in \omega_l\ /i$. Such conclusion might imply certain presumptions about the underlying signal. For example, it has been assumed in [11] that no two subsets $\theta_1, \theta_2 \subseteq [n]$ exist such that $\|x_{\theta_2}\|_1 \approx \|x_{\theta_2}\|_1$ or other assumptions in some other methods such as non-negativity of the signal and so on. Consequently, the applicability of the aforementioned approaches will be restricted to signals satisfying corresponding conditions. On the other hand, due to the non-binary nature of the proposed sparse projection matrix $\boldsymbol{P}$, the mechanism of voting is not applicable in case of $\boldsymbol{P}$. Consequently, we introduce the notions of "*alignment*" and "*isolation*" in section IV and show that the method of voting is a special case of such notions in case of binary matrices. Furthermore, we present necessary and sufficient conditions for detecting isolated non-zeros. Before that, we prove that the RIP property holds for the proposed sparse-real matrices; whereas it has been shown that RIP does not hold for the corresponding sparse-binary matrices [10].

III. Restricted Isometry Property

A matrix $\boldsymbol{P} \in \mathbb{R}^{m\times n}$ has RIP-2 of order $k \in \mathbb{N}$ [2], if for every k-sparse signal $x \in \mathbb{R}^n$, there exists a constant δ_k such that:

$$(1 - \delta_k)\|x\|_2^2 \leq \|\boldsymbol{P}x\|_2^2 \leq (1 + \delta_k)\|x\|_2^2$$

In this section, we present the proof sketch that certain sparse real-valued projection matrices have RIP-2.

Theorem 1. Assume $\boldsymbol{P} \in \mathbb{R}^{m\times n}$ ***has the following properties: (a) each column is non-zero in exactly*** $d = O(\log\frac{n}{k})$ ***random indices and (b) the non-zeros entries are zero mean Gaussian random variables with variance of*** $1/d$. ***Then for any*** $x \in \mathbb{R}^n$ ***with*** $k = \|x\|_0 = O(n)$ ***and sufficiently large*** $m = O(k\log\frac{n}{k}) \geq de^{(1+\epsilon)}$, ***there exists constant*** $c_0(\epsilon)$ ***such that:***

$$\Pr(|\|\boldsymbol{P}x\|_2^2 - \|x\|_2^2| \geq \epsilon\|x\|_2^2) \leq 2e^{-mc_0(\epsilon)} \tag{3}$$

Proof: We follow simple and elegant approaches of [12][13] to show (3) holds. Recall that non-zero entries of $\boldsymbol{P}$ are i.i.d

$N(0,1/d)$. Hence $E(y_i) = \sum_{j\in\omega_i} x_j E(\boldsymbol{P}_{i,j}) = 0$. Also by using the fact that non-zero entries of $\boldsymbol{P}$ are independent, it can be easily shown that:

$$E(y_i^2) = \sum_{j\in\omega_i} x_j^2 E(\boldsymbol{P}_{i,j}^2) = \frac{\|x_{\omega_i}\|_2^2}{d} \tag{4}$$

Consequently for $E(\|y\|_2^2) = E(\sum_{i=1}^m y_i^2) = \sum_i E(y_i^2)$:

$$E(\|y\|_2^2) = \frac{\sum_{i=1}^m \|x_{\omega_i}\|_2^2}{d} = \frac{\sum_{i=1}^n c_i x_i^2}{d} \tag{5}$$

where c_i is the number of times that $i \in [n]$ occurs in ω_i. Recall that, all columns of $\boldsymbol{P}$ are non-zero in exactly d row indices. Hence, for all $i \in [n]$we have $c_i = d$. Consequently: $E(\|y\|_2^2) = \|x\|_2^2$. Now we focus on finding how much $\|y\|_2^2$ could deviate from $\|x\|_2^2$. Here, we derive bounds for the lower and upper tails separately. Note that (see [13]): $y_i \sim \|x_{\omega_i}\|_2/\sqrt{d}\ N(0,1)$. Hence:

$$\|y\|_2^2 \sim \frac{\sum_{i=1}^m \|x_{\omega_i}\|_2^2 \chi^2(1)}{d} \tag{6}$$

where $\chi^2(1)$ is a chi-square with one degree of freedom. Since non-zero entries of $\boldsymbol{P}$ are independent, y_i and y_j are independent for $i \neq j$. Consequently the moment generating function for $\|y\|_2^2$ is:

$$M(t, \|y\|_2^2) = \mathrm{E}\left(e^{t\|y\|_2^2}\right) = \prod_{i=1}^m \left(1 - \frac{2\|x_{\omega_i}\|_2^2 t}{d}\right)^{-1/2} \tag{7}$$

for $t < 1/2$. Applying the method of generating functions and then the Markov's inequality yields:

$$\Pr(\|y\|_2^2 \geq (1+\epsilon)\|x\|_2^2) \leq g(t) = \frac{\mathrm{E}\left(e^{t\|y\|_2^2}\right)}{e^{t(1+\epsilon)\|x\|_2^2}} = \frac{\prod_{i=1}^m \left(1 - \frac{2\|x_{\omega_i}\|_2^2 t}{d}\right)^{-\frac{1}{2}}}{e^{t(1+\epsilon)\|x\|_2^2}} \tag{8}$$

For simplicity and without loss of generality assume x is a unit norm vector. Thus for all i, we have $\|x_{\omega_i}\|_2^2 \leq \|x\|_2^2 = 1$ and consequently:

$$g(t) \leq \left(1 - \frac{2t}{d}\right)^{-\frac{m}{2}} e^{-t(1+\epsilon)} \tag{9}$$

To get the tightest bounds, one needs to set the derivate of $g(t)$ with respect to t to zero. Since $t < 1/2$, only the solution of $t = \frac{d}{2} - \frac{m}{2(1+\epsilon)}$ would be of importance. Plugging this value into (9) and then (8) yields:

$$\Pr(\|y\|_2^2 \geq (1+\epsilon)\|x\|_2^2) \leq e^{-\frac{d}{2}(1+\epsilon)} \left(\frac{m}{de(1+\epsilon)}\right)^{-\frac{m}{2}} \tag{10}$$

It is straightforward to verify that, under the assumption of $m \geq de^{(1+\epsilon)}$ (recall $d = O\left(\log\frac{n}{k}\right)$ and $m = O\left(k\log\frac{n}{k}\right)$):

$$\Pr(\|y\|_2^2 \geq (1+\epsilon)\|x\|_2^2) \leq e^{-m(\frac{\epsilon^2}{4} - \frac{\epsilon^3}{6})} \tag{11}$$

where $\exp\{-m\left(\frac{\epsilon^2}{4} - \frac{\epsilon^3}{6}\right)\}$ is the bound for a dense projection matrix from a Gaussian ensemble[12].

Similarly for finding the bound for the lower tail, one can follow the same approach to get:

$$\Pr(\|y\|_2^2 \leq (1+\epsilon)\|x\|_2^2) \leq \frac{\mathrm{E}\left(e^{-t\|y\|_2^2}\right)}{e^{-t(1-\epsilon)}} = e^{t(1-\epsilon)} \prod_{i=1}^m \left(1 + \frac{2\|x_{\omega_i}\|_2^2 t}{d}\right)^{-1/2} \tag{12}$$

for the unit-norm vector x and $t < 1/2$. Define $S = \{i: x_i \neq 0\}$. Hence S is the set of indices where the signal x is non-zero. Then define the constant δ (note that this constant is different from the RIP constant δ_k) as follow:

$$\delta = \inf\{|x_i|\} \quad i \in S \tag{13}$$

In words, if $x_i \neq 0$ then $|x_i| \geq \delta$. Note that, since $k = \|x\|_0 = O(n)$ and $\boldsymbol{P}$ (when its non-zero entries are replaced with one) corresponds to a (k, β) expander graph ($\beta < 1$), hence (a) there exists indices Ξ where $\Xi \subseteq [m]$, $(1-\beta)dk \leq \xi = |\Xi| \leq dk$ and (b) $\forall i \in \Xi: \delta \leq \|x_{\omega_i}\|_2^2 \leq 1$. Thus ξ is in the order of $O(m)$. Consequently:

$$\Pr(\|y\|_2^2 \leq (1-\epsilon)\|x\|_2^2) \leq f(t) \tag{14}$$

Where:

$$f(t) = e^{t(1-\epsilon)} \left(1 - \frac{\delta t}{d}\right)^{\xi/2} \tag{15}$$

Letting $t = \frac{\epsilon}{2(1-\epsilon)} < \frac{1}{2}$ and calculating Taylor series of $f(t)$ around $\epsilon = 0$ yields:

$$f(t) = 1 + \left(\frac{1}{2} - \frac{\delta\xi}{4d}\right)\epsilon + \frac{(4d^2 - 12d\delta\xi - 2\delta^2\xi + \delta^2\xi^2)}{32d^2}\epsilon^2 + O(\epsilon^3)$$

It can be verified that, when:

$$\frac{2d}{\xi} < \delta \tag{16}$$

then:

$$f(t) \leq 2\exp\left\{\left(\frac{1}{2} - \frac{\delta\xi}{4d}\right)\epsilon\right\} \tag{17}$$

up to order $O(\epsilon^3)$. Note that in the asymptotic case when $k \to \infty$, constraint (16) demands that $\delta > 0$ which by definition of (13) is trivially true. Assume $m = O\left(k\log\frac{n}{k}\right) =$

$C_1 \log\frac{n}{k}$ and $d = O\left(\log\frac{n}{k}\right) = C_2 \log\frac{n}{k}$. Then combining (17) with (14) yields:

$$\Pr(\|y\|_2^2 \le (1-\epsilon)\|x\|_2^2) \le e^{-mC_0} \tag{18}$$

Where:

$$C_0 \ge \frac{(1-\beta)C_2\delta\epsilon}{4dC_1} \tag{19}$$

Thus, under the assumption of $k = O(n)$ (or equivalently $d = O\left(\log\frac{n}{k}\right) = O(1)$), we would have: $C_0 = O(1)$.

■

Having Theorem 1, one can apply Theorem 5.2 in [12] to prove that the sparse random projection matrix $\boldsymbol{P}$ (with the aforementioned conditions) has RIP-2. Consequently, BP [3] or other convex relaxation approaches [4] could be utilized for the proposed projection matrix to have a robust and stable recovery in the presence of noise and in case of compressible (as opposed to exactly sparse) signals. In the next section, we find conditions for Algorithm 1 to guarantee perfect recovery in case of exactly sparse signals.

IV. Necessary and Sufficient Conditions for Finding Isolated Non-Zeros

As before, assume $\boldsymbol{P} \in \mathbb{R}^{m\times n}$ is a matrix where each of its columns is non-zero in exactly $d = O(\log\frac{n}{k})$ random indices. Also assume that, non-zero entries of $\boldsymbol{P}$ are zero-mean Gaussian random variables with variance of $1/d$. If non-zero entries of $\boldsymbol{P}$ are replaced with one, then the resulting binary matrix would correspond to the adjacency matrix of a high quality expander graph (with high probability) [7]. Since the analysis of [7] is not based on values of $\boldsymbol{P}$, therefore the recoverability holds for $\boldsymbol{P}$. Consequently in here, we only discuss lines 3 and 4 in Algorithm 1. We show, how one can find subsets of signal coefficients containing only one non-zero, by examining compressive samples. To that end, we need to define two basic concepts. We say "x_i *is isolated in sample indices of* $l \subseteq [m]$" if $\|x_{\omega_l}\|_0 = 1, x_i \neq 0$. In other words, x_i is the only non-zero signal coefficient (among indices ω_l) spanned by samples y_l. Clearly, since $\boldsymbol{P}_{:,i}$ is non-zero only in indices Ω_i, therefore $l \subseteq \Omega_i$.

For $q \subseteq [m]$ we say "y_q *is aligned with the* j*-th column of* $\boldsymbol{P}$" if: $\forall i \in q: \boldsymbol{P}_{i,j} \neq 0$ and $\exists \alpha \neq 0: y_q = \alpha \boldsymbol{P}_{q,j}$. Note that, if $\boldsymbol{P}$ is binary (i.e. $\boldsymbol{P}_{q,j}$ is an all one vector), then the definition of alignment reads as "*there exists a set of samples* (y_q), *all voting the same value of* α". This is exactly the voting mechanism, employed in combinatorial algorithms. This verifies our claim that the voting mechanism is in fact an alignment in the special case of binary matrices. Assume that $x_i \neq 0$ is isolated in sample indices of $l \subseteq \Omega_i \subseteq [m]$. Then:

$$y_l = \boldsymbol{P}_{l,\omega_l} x_{\omega_l} = x_i \boldsymbol{P}_{l,i} \tag{20}$$

In words, if x_i is isolated in sample indices of l then y_l will be aligned with the i-th column of $\boldsymbol{P}$. Thus isolation is a necessary condition for alignment. Now, if isolation is a sufficient condition for alignment as well, then isolated non-zero coefficients in the signal could be found by looking for alignment of samples among the columns of $\boldsymbol{P}$. For now assume that isolation and alignment are equivalent. Then step 3 and 4 in Algorithm 1 could be implemented as follows: to see whether x_i is isolated in a set of unknown samples, one might consider $\boldsymbol{P}_{\Omega_i,i}$ and looks for $q \subseteq \Omega_i$ such that $y_q = \alpha \boldsymbol{P}_{q,j}$. If there exists any q with those conditions, then (20) could be applied to infer $x_i = \alpha$. In the rest of this section, we identify the underlying conditions under which isolation is a sufficient condition for alignment.

Here, it is important to highlight the impact of the size of the set $l \subseteq \Omega_i \subseteq [m]$ in our analysis. Since, in an extreme case when $|l| = 1$ in (20), there always exists a meaningless alignment. Recall that, each column of $\boldsymbol{P}$ is non-zero in exactly $d = O(\log n/k)$ entries. In [7], it has been shown that in case of a sparse binary projection matrix based on the adjacency matrix of a left d-regular random graph, there exists at least $\tau \ge 1 + d/2$ samples which span only one (and the same) non-zero in all stages of decoding. Hence, in here we have $l = \tau > d/2 = O(\log n/k)$ and ignore alignments with size smaller than $d/2$ in the decoder.

As before, assume y_l is aligned with the i-th column of $\boldsymbol{P}$, where $l > d/2$. Since by definition $y_l = \boldsymbol{P}_{l,\omega_l} x_{\omega_l}$ and $\|y_l\|_2 > 0$, there are two possible cases here: (a) $\|x_{\omega_l}\|_0 = 1$ or (b) $\|x_{\omega_l}\|_0 > 1$. In the first case, since y_l is non-zero in all entries ($\|y_l\|_0 = |l|$) and for all $j \in \omega_l/i$ we have $\|\boldsymbol{P}_{l,j}\|_0 < |l|$ (this comes from the expander property of $\boldsymbol{P}$) thus one can conclude that, x_i is the only non-zero among indices of ω_l and its value can be easily computed by (20). In words, in the first case, isolation is a sufficient and necessary condition for alignment. Unfortunately, a similar argument generally does not hold true for the second case ($\|x_{\omega_l}\|_0 > 1$) for $\boldsymbol{P}$. Nevertheless, here we show how one can modify $\boldsymbol{P}$ such that, the second case would never happen with a very high probability. To that end, we follow these steps (a) for any subset of samples $l \subseteq [m]$ with $l \ge 1 + \frac{d}{2} = O(\log\frac{n}{k})$ first we find $Q = \max\|x_{\omega_l}\|_0$, the maximum number of non-zeros which will be spanned (with high probability) by samples l. (b) Then we modify the projection matrix, such that every Q columns of $\boldsymbol{P}_{l,\omega_l}$ would be independent. Finally by using elementary arguments, one can show that it is impossible to have $\|x_{\omega_l}\|_0 > 1$ and y_l to be aligned with the i-th column of $\boldsymbol{P}$.

Define the binary random variable X_i as:

$$X_i = \begin{cases} 1 & \text{if } x_i \neq 0 \text{ and } i \in \omega_l \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

It can be easily verified that $Q = \max\sum_{i=1}^{n} X_i$. Now since the probability that $x_i \neq 0$ is independent from the probability of the event that $i \in \omega_l$, thus:

$$\Pr(X_i = 1) = \frac{k}{n}\left(1 - \frac{\binom{m-l}{d}}{\binom{m}{d}}\right) \tag{22}$$

Assuming $k \gg 1$ and to simplify derivations, here use the approximation: $1/(m-d+1) \approx 1/m$. Using the inequality $1+\alpha \le e^{\alpha}$, one can show: $\mu = E[\mathrm{Q}] = n\Pr(X_i = 1) \le k(1-\exp(-l/m))^d$. Recalling that $d = O\left(\log\frac{n}{k}\right) = c_1 \log\frac{n}{k}$, $d/2 < l \le d$ and $m = O\left(\log\frac{n}{k}\right) = c_2 k \log\frac{n}{k}$, we have:

$$\mu \le k\left(1-\left(\frac{k}{n}\right)^{\frac{c_1^2}{c_2 k}}\right) \qquad (23)$$

Here we would like to note that μ is extremely small. In fact, it can be easily shown that for a constant $\alpha \ge 1$ and under the assumption of $n > c_1^2 \log\left(\frac{k}{n}\right)/c_2 \log\left(1-\frac{\alpha}{k}\right)$: we get $\mu \le \alpha$. Nevertheless, here we assume that $\mu = \alpha = O(1)$ for a properly chosen value of α. Now it only remains to bound the deviation of Q from μ. Fortunately, since Q is the sum of Bernoulli random variables, there exists good concentration inequalities for it. For instance, by using Chernoff bound for Poisson trials, it can be easily shown that for $\log(1+\delta) > 2$ (or $\delta \ge 6.389$) we have: $\Pr(Q \ge \mu + \delta\beta \log n) \le \frac{1}{n^{\beta}}$.

Now if the spark [14] of $\boldsymbol{P}_{l,\omega_l}$ is more then Q, then it is impossible to have y_l aligned with the i-th column of $\boldsymbol{P}$ and at the same time $\|x_{\omega_l}\|_0 > 1$. This can be proved by contradiction. More specifically, assume $1 < \|x_{\omega_l}\|_0 < Q$ and y_l is aligned with the i-th column of $\boldsymbol{P}$ ($y_l = \boldsymbol{P}_{l,\omega_l} x_{\omega_l} = \alpha \boldsymbol{P}_{l,j}$). Define a new vector x' where $x'_j = x_j - \alpha$ and $x'_i = x_i$ for $i \in \omega_l / j$. Then $\boldsymbol{P}_{l,\omega_l} x' = 0$ and $\|x'\|_0 \le Q$ which contradicts the presumption that the spark of $\boldsymbol{P}_{l,\omega_l}$ is more than Q.

In summary, as long as every $Q = O(\log n)$ columns of $\boldsymbol{P}_{l,\omega_l}$ for every possible subset $l \subseteq [m]$ with $|l| \ge O\left(\log\frac{n}{k}\right)$ are independent, then isolation would be a sufficient and necessary condition for alignment. Unfortunately, this requirement does not hold for a projection matrix constructed from a random left regular bi-partite graph, since it is quite possible to have two columns with indices u and v in $\boldsymbol{P}_{l,\omega_l}$ such that, both of them are non-zero exactly in one (and the same) row index j. Clearly, these two columns in $\boldsymbol{P}_{l,\omega_l}$ are linearly dependent which reduces the spark of $\boldsymbol{P}_{l,\omega_l}$ to a minimal of one. This problem could be resolved by using Algorithm 3 to modify $\boldsymbol{P}$.

Inputs : $\boldsymbol{P} \in \mathbb{R}^{m\times n}$ and Q
Output : $\boldsymbol{P}'$
For $i = 1$ to n do
  $T = \mathbf{0}_{Qm}$
  For $j \in \Omega_i$ do
    $\forall l \in \{Q(j-1)+1, \dots, Qj\}: T_l \sim N\left(0, \frac{1}{dQ}\right)$
  End
  $\boldsymbol{P}'_{:,i} = T$
End

Algorithm 3 Increasing the spark of sub-matrices of $\boldsymbol{P}$.

The output matrix $\boldsymbol{P}'$ has the following properties: (a) $\boldsymbol{P}' \in \mathbb{R}^{Qm\times n}$ (b) $\Omega'_i = \{j: \boldsymbol{P}'_{j,i} \ne 0\} = \cup_{j\in\Omega_i}\{j(Q-1)+1, j(Q-1)+2, \dots, jQ\}$ (c) Let $\omega'_i = \{j: \boldsymbol{P}'_{i,j} \ne 0\}$. Then $\forall i \in \{j(Q-1)+1, j(Q-1)+2, \dots, jQ\}$ we have $\omega'_i = \omega_j$. Thus for instance, if x_i is isolated in sample indices l, then the same coefficient x_i shall be isolated in samples y'_b where $y' = \boldsymbol{P}'x$ and $b = \cup_{j\in l}\{j(Q-1)+1, j(Q-1)+2, \dots, jQ\}$. Consequently, the decoding algorithm and all the analysis we presented in case of the projection matrix $\boldsymbol{P}$ stay the same in case of the new projection matrix $\boldsymbol{P}'$. However, $\boldsymbol{P}'$ has a feature that $\boldsymbol{P}$ does not have and that is: for any subset $l \subseteq [m]$ with $|l| = O(d)$, $\boldsymbol{P}'_{l,\omega'_l}$ would have a spark of at least Q. This can be shown by the following argument. If $u, v \in \omega_l{}'$ then (by properties of $\boldsymbol{P}'$) either $\Omega'_u \cap \Omega'_v = \emptyset$ (meaning those two columns are orthogonal) or $|\Omega'_u \cap \Omega'_v| \ge Q$. Now since the non-zero entries of $\boldsymbol{P}'$ are populated randomly ($\sim N(0, \frac{1}{dQ})$), thus every Q columns of $\boldsymbol{P}'$ would be independent. Note that, this property is gained at the cost of increasing the sample requirement from $O(k\log(n/k))$ to $O(Qk\log(n/k)) = O(k\log(n)\log(n/k))$. Admittedly our analysis could be improved furthermore and one can come up with tighter bounds for Q. Our simulation results show that, in practice an optimal number of samples is required for perfect recovery.

V. Simulation Results

In this section, we present simulation results validating our claims/proofs in this paper. First, the joint performance of the proposed combinatorial algorithm and sparse real-valued projection matrix for recovering sparse signals is investigated. To that end, for a fixed signal length of $n = 1000$, ten different levels of m/n and seven levels of k/m, we find probability of exact recovery as functions of those parameters. For each configuration of m/n and k/m, one hundred simulations are performed as follows: (a) we select uniformly at random k locations among $[n]$ (b) In selected indices, the test signal x would be populated according to a normal distribution. (c) We generate $\boldsymbol{P}$, a sparse real-valued projection matrix based on assumptions Section II and Algorithm 4, with parameters $d = 3$ and $Q = 2$. (d) The compressive samples $y = \boldsymbol{P}x$ are then given to Algorithm 1 to compute $\hat{x}$. Figure 1, shows the ratio of perfect recovery ($\hat{x} = x$) among one hundred simulations for each configuration of m/n and k/m.

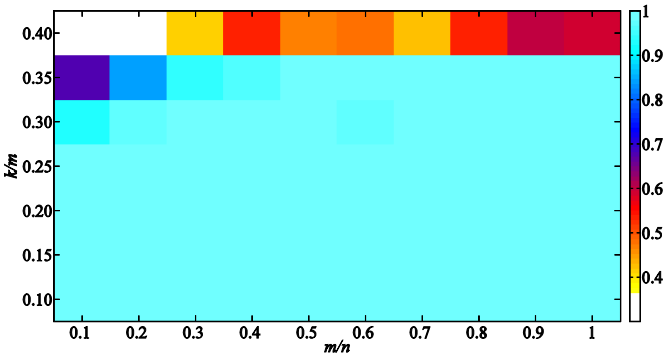

Figure 1. The probability of exact recovery of with signal of length $n = 1000$ as a function of k/m and m/n.

To evaluate performance of introduced projection matrices in the presence of noise, we considered a signal x with length $n = 2000$ where $k = 150$ of its entries are selected uniformly at random and their values are uniformly chosen in the range of $[-0.5,0.5]$. To assess, how the proposed projection matrix would act in conjunction with a robust solver such as BP, we generated two matrices $\boldsymbol{P}, \boldsymbol{P}_d \in \mathbb{R}^{m \times n}$ where $m = 600$, $\boldsymbol{P}$ is a sparse real-valued projection matrix with parameters $d = 3$ and $Q = 2$ and $\boldsymbol{P}_d$ is a dense random matrix with Gaussian entries. Correspondingly, two sets of compressive samples were generated and contaminated by the same noise ϵ, i.e. $y = \boldsymbol{P}x + \epsilon$ and $y_d = \boldsymbol{P}_d x + \epsilon$ and fed into BP. We chose ϵ to be a Gaussian noise ($\epsilon \sim N(0, \sigma^2)$) and considered ten different levels for σ^2. For each variance level, we performed one hundred simulations and recorded the maximum ℓ_2 norm of recovery error ($\|\hat{x} - x\|_2$). The results are presented in Figure 2. As clearly shown in that figure, BP shows very similar stabilities under both types of (dense Gaussian and the proposed) projection matrices.

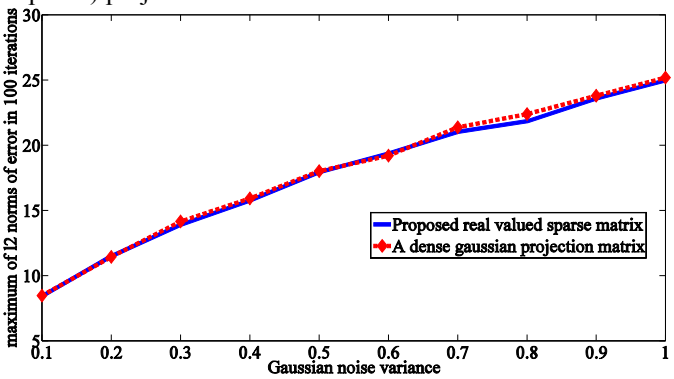

Figure 2. ℓ_2 norm of recovery error in a noisy setting.

VI. Conclusion

In this paper, we show how certain classes of combinatorial algorithms are linked to convex relaxation methods for CS. We considered the scenario when non-zero entries of a sparse binary projection matrix are replaced with random Gaussian numbers and show how the notion of voting (employed for sparse binary projections) extends to isolation/alignment. The RIP-2 is proved for this new class of projections. Sufficient and necessary conditions for the equivalency of isolation and alignment are presented.

References

[1] David Donoho, "Compressed sensing". IEEE Trans. on Information Theory, 52(4), pp. 1289 - 1306, April 2006.

[2] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information". IEEE Trans. on IT, 52(2) pp. 489 - 509, February 2006

[3] S.S. Chen, D. L. Donoho, Michael, A. Saunders, "Atomic decomposition by Basis Pursuit", Siam J. of Scientific Comp, pp. 33-61, 1998.

[4] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems". IEEE J. of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing, 1(4), pp. 586-598, 2007.

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

[6] R. Berinde and P. Indyk, "Sparse recovery using sparse random matrices". (Preprint, 2008)

[7] S. Jafarpour, W. Xu, B. Hassibi, and R. Calderbank, "Efficient compressed sensing using high-quality expander graphs", IEEE Trans Info Theory, 2009

[8] R. Berinde, A. C. Gilbert, P. Indyk, H. Karloff, and M. J. Strauss, "Combining geometry and combinatorics: A unified approach to sparse signal recovery". (Preprint, 2008).

[9] D.L. Donoho, A. Maleki, and A. Montanari. "Message passing algorithms for compressed sensing", Proc. Natl Acad. Sci., 2009.

[10] Venkat Chandar, "A negative result concerning explicit matrices with the restricted isometry property". (Preprint, 2008)

[11] S. Sarvotham, D. Baron, and R. Baraniuk, "Sudocodes-Fast measurement and reconstruction of sparse signals". IEEE ISIT 2006.

[12] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices". Constructive Approximation, 28(3), pp. 253-263, December 2008.

[13] D. Achlioptas, "Database-friendly random projections", 20th Annual Symposium on Principles of Database Systems, 2001, pp. 274–281.

[14] http://www.egr.msu.edu/waves/people/nima/isit-2