RECOVERING SIMPLE SIGNALS

ANNA C. GILBERT, BRETT HEMENWAY, ATRI RUDRA, MARTIN J. STRAUSS, AND MARY WOOTTERS

Abstract. The primary goal of compressed sensing and (non-adaptive) combinatorial group testing is to recover a sparse vector x from an underdetermined set of linear equations $\Phi x = y$. Both problems entail solving $\Phi x = y$ given Φ and y but they use different models of arithmetic, different models of randomness models for Φ, and different guarantees upon the solution x and the class of signals from which it is drawn. In [32], Lipton introduced a model for error correction where the channel is computationally bounded, subject to standard cryptographic assumptions, and produces the error vector x that must be found and then corrected. This has been extended in [24, 34] to create more efficient schemes against polynomial and logspace bounded channels. Inspired by these results in error correction, we view compressed sensing and combinatorial group testing as an adversarial process, where Mallory the adversary produces the vector x to be measured, with limited information about the matrix Φ. We define a number of computationally bounded models for Mallory and show that there are significant gains (in the minimum number of measurements) to be had by relaxing the model from adversarial to computationally or information-theoretically bounded, and not too much (in some cases, nothing at all) is lost by assuming these models over oblivious or statistical models. We also show that differences in adversarial power give rise to different lower bounds for the number of measurements required to defeat such an adversary. By contrast we show that randomized one pass log space streaming Mallory is almost as powerful as a fully adversarial one for group testing while for compressed sensing such an adversary as weak as an oblivious one.

compressed sensing, group testing, computationally bounded adversary

ACG and MW are at the Department of Mathematics, 2074 East Hall, University of Michigan, Ann Arbor, MI 48109. BH and MJS are with the Departments of Mathematics and Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109. AR is at the Department of Computer Science at the University at Buffalo SUNY, Buffalo, NY. ACG is an Alfred P. Sloan Research Fellow and has been supported in part by NSF DMS 0354600. MJS has been supported in part by NSF DMS 0354600 and NSF DMS 0510203. ACG and MJS have been partially supported by DARPA ONR N66001-06-1-2011. AR is supported by NSF CAREER grant CCF-0844796. Email: {annacg,bhemen,martinjs,wootters}@umich.edu and atri@buffalo.edu.

1. Introduction

Group testing was introduced by Dorfman in his seminal 1943 paper to identify soldiers in WWII who had syphilis [15]. Combinatorial group testing[1] has found applications in drug and DNA library screening (the literature is large, see [37, 26, 46, 25] and references thereof), multiple access control protocols [5, 45], live baiting of DoS attackers [27], data forensics [23] and data streams [13], among others. The reader is referred to the standard monograph on group testing for more details [16].

Compressed sensing was introduced much more recently in 2006 by [8, 14] as a method for acquiring signals in a "compressed" fashion and then reconstructing good (i.e., sparse) approximations to the signals from those observations. The problem has many practical applications from analog-to-digital conversion [28, 43, 31], novel optical device design [47, 42, 17], pooling designs for rare allele detection [20], and video acquisition [44], to cite a few examples.

Even though they were introduced more than half a century apart, compressed sensing and (non-adaptive) combinatorial group testing are similar. Their primary goal is to recover a sparse vector x from an underdetermined set of linear equations $\Phi x = y$. Both problems include solving $\Phi x = y$ given Φ and y but they use different models of arithmetic, different randomness models for Φ and different guarantees upon the solution x and the class of signals from which it is drawn. Compressed sensing attempts to recover a sparse signal $x \in \mathbb{R}^N$ from a small number of linear measurements Φx. The signal x is usually assumed to be composed of a "head", the k largest entries (in magnitude) of the vector, supported on a small number of indices, and a tail whose ℓ_1 or ℓ_2 norm is small. The goal is to recover a close approximation [2] $\widehat{x}$ to x from the measurements Φx. In the combinatorial group testing model the binary matrix Φ represents a pooling design and the goal is to recover a small set (of size d) of "defective" elements from a set of tests Φx. (Here $x \in \{0,1\}^N$.) In the group testing model, a test fails if any items in the group being tested are defective, so the matrix vector product Φx is done using boolean OR.

Compressed sensing and group testing share another similarity, which is a limited range of types of models that generate the input signal x. Existing models typically fall into one of two categories:

- **Adversarial or "for all" models.** An ambitious goal is to successfully recover all signals that satisfy a certain geometric property. In the compressed sensing this usually corresponds to a bound on size of the support of the head, and an ℓ_1- or ℓ_2-norm bound on the tail, although more restrictive models on the form of the support set (e.g., k non-zero elements must lie within $B < k$ contiguous blocks or on contiguous paths in a tree [3]). In combinatorial group testing settings these correspond to restrictions on the Hamming weight of the signal.
- **Probabilistic or "for each" models.** These are two slightly different cases. One goal (that of the "for each" model in compressed sensing) is to recover a fixed, arbitrary vector with high probability over the construction of the matrix Φ. A similar goal is to successfully recover a random signal from a specified distribution with high probability over the *signal*, which has been considered in compressed sensing. Statistical models that have been considered include the uniform distribution or graphical models for the k-sparse head [7, 10, 3], possibly combined with i.i.d. Gaussian noise for the tail [7]. Somewhat surprisingly, neither the "for each" model nor the random signal model has received much attention in the (non-adaptive) group testing setting, to the best of our knowledge[3].

Both these extremal types of models have benefits and drawbacks. The advantage of the "for all" model is that it places minimal assumptions on the signal, which typically implies that schemes developed in this model also work in more benign models. The wide applicability of the "for

[1] In this paper we will only consider non-adaptive schemes.

[2] The output should satisfy $\|x - \widehat{x}\|_p \leq C\|x - x_k\|_q$ where $x - x_k$ is the tail of the vector. There are three combinations of norms ℓ_p/ℓ_q of interest: ℓ_2/ℓ_2, ℓ_2/ℓ_1, and ℓ_1/ℓ_1.

[3] There are, however, a number of adaptive group testing results for a variety of probabilistic models. See [29, 41, 33]

all" model comes at a cost, and usually implies that positive results are harder to achieve. More importantly, it is highly unlikely that a natural process would generate worst-case signals. On the other hand, the "for each" model typically leads to stronger positive results but is not applicable in as many situations. Schemes that specify particular distributions on the input tend to be specialized, and it is debatable if a natural signal producing process would be oblivious to Φ, or that a natural process would follow a simple, fully specified random process. One of the goals of this paper is to develop a model for signals in such a way that it captures natural process and the same time has the benefits of both the "for all" model (wider applicability of the schemes developed) and the "for each" model (stronger positive results).

While schemes in the "for each" model are the easiest to construct, in many situations a "for each" guarantee may not be sufficient. One problem that arises is that of feedback. In real-world applications the measurement matrix, Φ, will be fixed and used to recover many signals. This is acceptable if we can prove a "for all" guarantee on the matrix Φ, but if Φ only satisfies a "for each" guarantee problems may arise. One of the simplest types of problems is when future signals may depend on *the measurements* of current signals. As a simple example exhibiting feedback, imagine a vector x indicating the monthly sales counts of items in a store. The management may wish to keep a sketch, Φx, of the best selling items.[4] The store may then choose to update its layout or sales strategy depending on the result of the sketch.[5] Next month's sales will be influenced by this updated layout or strategy, and hence are dependent on Φx. A more adversarial situation could arise as follows. An attacker launches a missile at a ship. The ship, using some form of compressed sensing radar [4], detects the incoming missile and takes evasive action. The ship's evasive action is viewed by the attacker, and the attacker can adjust his second attack accordingly. Since the ship cannot be expected to choose a new measurement matrix, Φ, between successive attacks, the second attack may not be detected using a matrix that only satisfies a "for each" guarantee.[6] In both these cases, successive signals are subject to some sort of feedback. Signals that depend on some sort of feedback arise naturally, but these signals are not covered by the "for each" model. These types of signals fit naturally into a bounded adversarial model. We point out as well, that even if the attacker cannot be modeled as a simple process, the amount of information the attacker receives from viewing the target's evasive action is limited, so the attacker's view of Φx is naturally modeled as having to pass through an "information-theoretic bottleneck."

When considering intermediate signal models, it is instructive to consider the landscape in a related field: coding theory. Compressed sensing and group testing are closely related to coding, the parallel is most easily seen by viewing Φ as the parity-check matrix of a linear code, and the signal x as the error pattern [9]. Coding theory has a worst-case (for all) model pioneered by Hamming, as well as a probabilistic ("for each") models introduced by Shannon. Unlike compressed sensing and group testing, coding theory also has a rich literature on intermediate models. One such example is the arbitrarily varying channel [30]. Another example, that is more in line with computational thinking, is the notion of computationally bounded adversaries. In [32], Lipton introduced a model for error correction where the channel is computationally bounded, and is subject to standard cryptographic assumptions. This has been extended in [24, 34] to create more efficient schemes against polynomial and logspace bounded channels. Note that the "for all" model corresponds to a computationally unbounded adversarial model while the probabilistic models correspond to much weaker computationally bounded or oblivious adversaries.

[4]Suppose the sketch indicates that antihistamines are selling quite well, but sunscreen is doing poorly.

[5]In the example of the previous footnote, the manager suspects a high pollen count, which is confirmed through some arbitrary process involving checking the pollen count online or in a weather forecast. The store moves the sunscreen to a back shelf and, to be safe, also bug spray, and orders more facial tissues.

[6]In practice the measurement matrix, Φ, is often built into hardware and is extremely difficult to change on the fly.

Given the importance of compressed sensing and group testing, we believe it is important to develop an understanding of their properties under more than just the most extreme classes of signal. The primary goal of this paper is to initiate the study of intermediate models for both compressed sensing and group testing:

- **Computationally bounded models.** The goal here is to recover a signal generated by a computationally bounded adversary that must be recovered with high probability over the construction of Φ. This corresponds to the computationally bounded channels in coding theory. Like the bounded channel model for error correcting codes, we often imagine that the input for these problems is generated by some natural process; not necessarily powerful, but also not necessarily well-behaved, and, in particular, not necessarily independent of the measurement matrix Φ. Furthermore, computational conditions are arguably more natural for algorithms than geometric or statistical ones: a geometric model is useful mathematically, but does not give insight into how difficult it is to choose problematic vectors. Statistical models put the burden on the problem designer to produce a natural model, if one even exists. In contrast, a computationally bounded model is quite broad. As a special case we consider the *streaming adversary* where one is only allowed a single pass over Φ and uses only logarithmic amount of space.
- **Information theoretically bounded models.** Another way to have a spectrum of adversaries it to characterize recoverable signals in terms of information content. Here, the goal to recover all signals whose mutual information with Φ is small. Note that the "for each" model corresponds to zero mutual information and the "for all" model corresponds to the full information case.

A second goal of this paper is to compare the compressed sensing and group testing frameworks. On the one hand group testing is an easier task than compressed sensing as we only need to handle binary vectors (and there is no measurement noise[7]). On the other hand each measurement gives very little information about x as each measurement in group testing is just an OR of the bits). Our metric of comparison is the minimum number of measurements required to recover the signal or the defective set, as in both problems a primary goal is to minimize the number of measurements (the rows of Φ) necessary for recovery.

Let k denote both the size of the "head" of the signal and of the defective set. Then, if we compare group testing to compressed sensing with ℓ_1/ℓ_1 error guarantees, it would seem that compressed sensing is more powerful as $O(k \log(N/k))$ measurements suffice [2] to perform (even) "for all" ℓ_1 compressed sensing while one needs $\Omega(k^2 \log_k N)$ measurements [18, 19, 21] to recover k sparse binary vectors by group testing. On the other hand, if we are interested in ℓ_2/ℓ_2 error bounds in compressed sensing, then we must use $\Omega(N)$ measurements in the "for all" setting [12]. By contrast, group testing only needs $O(k^2 \log(N/k))$ measurements [16] and thus, compressed sensing seems harder.[8]

Only $O(k \log(N/k))$ measurements are needed [22] in the "for each" model for ℓ_2/ℓ_2 recovery, which is optimal [2], and far away from the lower bound of $\Omega(N)$ in the "for all" model. Since this is the main gap between the "for all" and "for each" for compressed sensing, and we are interested in interpolating between these cases, we will consider only ℓ_2/ℓ_2 compressed sensing for the rest of the paper. Furthermore, ℓ_2/ℓ_2 norms are the most natural for signal processing as the ℓ_2 norm squared is the signal energy and the error can be expressed in terms of signal-to-noise ratio (SNR), a standard metric.

1.1. **Our results.** Our results are summarized in Table 1.

[7]Error-tolerant group testing is a well-studied topic but we will only concentrate on "error-less" group testing in this paper.

[8]Since in group testing we deal with binary vectors, all ℓ_p norms are the same whereas in compressed sensing the choice of the norm matters.

Combinatorial group testing		
Mallory	Num. Measurements	Reference
Adversarial	$\Omega(d^2 \log(N/d)/\log(d))$	[21, 18, 19]
Information-Theoretically bounded (logspace)	$O(d \log(N))$	this paper
Oblivious	$O(d \log(N))$	this paper
Binary symmetric channel	$\Omega(d \log(N/d))$, $O(d \log(N))$	this paper
Logspace streaming (one pass over the rows)	$\Omega(d^2/\log d)$	this paper
Deterministic $O(\log d \log N)$ space	$\Omega(d^2/\log d)$	this paper
Sparse signal recovery		
Mallory	Num. Measurements	Reference
Adversarial	$\Omega(n)$	[12]
Adversarial, but restricted so that $\|x - x_k\|_1 \leq \sqrt{k}\|x - x_k\|_2$	$O(k \log(N/k)$	[8, 14]
Information-Theoretically bounded (logspace)	$O(k \log(N/k))$	this paper
Logspace streaming (one pass over the rows)	$O(k \log(N/k))$	this paper
Oblivious	$O(k \log(N/k))$	[22]

TABLE 1. Results on the number of measurements in combinatorial group testing and sparse recovery various access models. For sparse recovery, all results obtain an ℓ_2/ℓ_2 error bound, that is, $\|x - \hat{x}\|_2 < C\|x - x_k\|_2$, where $\hat{x}$ is the recovered signal and x_k is the best k-term approximation.

We begin with our results on the "for each" or probabilistic models for group testing. We show that $O(d \log N)$ measurements suffice to recover a d-sparse binary vector x in the "for each" model as well as the case when x is generated by the well-known binary symmetric channel (each bit is one independently with probability d/N). This shows that (a) we gain in the number of measurements for group testing when we go from the "for all" to "for each" model and (b) group testing and (both ℓ_1/ℓ_1 and ℓ_2/ℓ_2) compressed sensing have similar complexities in the "for each" model. A natural follow-up question is whether we can distinguish between group testing and ℓ_2/ℓ_2 compressed sensing by an adversary that is weaker than the "for all" model (where we know there is a gap).

Somewhat surprisingly we are able to show that a randomized one pass $O(\log N)$ space streaming adversary suffices to distinguish between group testing and ℓ_2/ℓ_2 compressed sensing. In particular, we are able to show that $O(k \log(n/k))$ measurements suffice against such an adversary for ℓ_2/ℓ_2 compressed sensing, i.e., in some sense such an adversary is just as weak as a "for each" adversary. We use lower bounds in communication complexity to show upper bounds for recovery against a streaming adversary. On the other hand, we show that for group testing a randomized one pass log space streaming adversary is almost as powerful as a "for all" adversary by showing that $\Omega(d^2/\log d)$ measurement are needed to perform group testing against such an adversary. Using limited-wise hash functions, we can de-randomize this result to show that $\Omega(d^2/\log d)$ measurements are necessary to recover a d sparse binary vector against a deterministic $O(\log d \log N)$ space bounded adversary (with no a priori limit on the number of passes).

Finally, we show that for information-theoretically bounded adversaries with $O(\log N)$ bits of mutual information do no better than the oblivious model for both compressed sensing and group testing.

2. Preliminaries

2.1. **Notation.** There are three key components to both group testing and sparse signal recovery designs: the measurement matrix Φ, the unknown sample or signal x, and the observations y. We assume that x is of length N and that the matrix Φ has dimensions $m \times N$. In the combinatorial group testing literature, the number of rows m is typically denoted by t for tests but we will stick with the sparse signal recovery notation. The final parameter of interest in either problem is the number of defectives d that we seek (for group testing) or the sparsity k (for signal recovery). For this parameter, we use the convention of the two separate bodies of literature and express the number of measurements m in terms of d and N for group testing and k and N for sparse signal

recovery. One important distinction between group testing and sparse recovery signals is that for group testing, the signal x is a binary vector with d 1s, while for sparse recovery, the signal $x \in \mathbb{R}^N$ consists of two pieces: x_k, the k-largest entries in absolute value (called the "head" of the vector), and $x - x_k$, the remaining $N - k$ entries (called the "tail" of the vector).

Since we are considering these problems in an adversarial context, we will name the process generating inputs x Mallory. She will take on various powers throughout the paper.

2.2. **Models for Mallory.** We consider several models for Mallory.

- **Binary symmetric channel:** Entries in the signal x are 1 with probability p and 0 with probability $1 - p$, independent of the matrix Φ (analogous to the error correcting code setting in which bits in the encoded message are flipped with probability p). We consider this model for combinatorial group testing only and note that in this model, the number of defectives present in the signal is a random variable.
- **Oblivious:** Mallory cannot see the matrix Φ and generates the signal x independent from Φ. In the combinatorial group testing problem, Mallory chooses a set of size d to be the defective set. For sparse signal recovery, this model is equivalent to the "for each" signal model.
- **Information-Theoretic:** Mallory's output has bounded mutual information with the matrix. To cast this in a computational light, we say that an algorithm M is *(log-)information-theoretically-bounded* if $M(x) = M_2(M_1(x))$, where the output of M_1 consists of at most $O(\log(|x|))$ bits. Lemma 1 shows that this requirement can be expressed as a bound on the success probability of an oblivious adversary. We distinguish this case from the oblivious case because it provides a natural abstraction in a number of settings. As mentioned in the introduction, in a situation with feedback, it is reasonable to assume that an adversary (powerful or not) might only see a few bits about Φ based on our actions. This might also arise in a situation where a space-bounded streaming adversary does not start to generate x until it has seen all of Φ. This is obviously a much weaker adversary that a general information-theoretic bounded one, but it comes up naturally. For example, suppose measurements must be coordinated between several sensors, and Φ (or a seed for Φ) must be passed between them, through an environment which then generates the signals. The sensors do not begin to take measurements until after Φ has been initialized. The signals generated then depend on Φ, and hence an algorithm designed for an oblivious adversary would not give guaranteed results. However, the assumption is that the environment can be modeled as a simple process, perhaps with logarithmic space. A recovery algorithm designed for an information-theoretic bounded adversary would apply.
- **Streaming log-space:** Mallory can stream over Φ but only has log space with which to store information about Φ and to compute an error message. In contrast to the information theoretic model above, which puts restrictions on the amount of mutual information between the signal and Φ, this is a computational restriction. A logspace streaming adversary is the natural "weakest" computationally bounded adversary, and thus is a reasonable place to start our analysis. Additionally, it has practical applications: both the sensor example and the error correction example above could fit into this model as well.
- **Adversarial:** Mallory has full computational power.

Before we dive into technical results, we give a general statement about randomized algorithms and information theoretically bounded adversaries that we will use in the proceeding sections.[9] For a randomized algorithm A with failure probability $\epsilon > 0$, if an omnipotent adversary sees the seed r for A before choosing the input x, then A will fail. However, if the adversary has limited space, success is still possible:

[9]Despite the simplicity of this observation, or, perhaps because of its simplicity, we cannot find an appropriate reference for this result.

Lemma 1. *Pick $\ell = \ell(N)$, and fix $0 < \alpha < 1$. Let A be any randomized algorithm which takes input $x \in \{0,1\}^N$, $r \in \{0,1\}^m$, which "succeeds" with probability $1 - \epsilon$. Then for any information theoretically bounded algorithm M with space ℓ, $A(M(r), r)$ succeeds with probability at least $\min\{1 - \alpha,\ 1 - \ell/\log(\alpha/\epsilon)\}$ over the choice of r.*

The proof is in the full version.

3. Combinatorial group testing

The goal of group testing is to identify a set of d "defective" elements from a population of size N. This is accomplished by creating a pooling design, where each element is placed in multiple "pools" and each pool is tested separately. A pool will fail if it contains at least one defective element. The primary goal in this scenario is to minimize the number of pools, which is equivalent to minimizing the number of tests needed. It is customary to identify the set of defectives as a vector, x, of weight d in $\{0,1\}^N$, and a pool as a vector in $\{0,1\}^N$ identifying which elements are in the pool. The entire design, consisting of m pools, can be written concisely as a matrix $\Phi \in \{0,1\}^{m \times N}$. The outcome of the m tests then corresponds to the matrix product Φx, where the matrix product is computed using boolean AND an OR, since each test fails if *any* element in that pool is defective.

Combinatorial group testing schemes have focused on developing pooling designs that are robust against "worst-case" (adversarial) distribution of defectives. In particular, this means, that the group testing scheme must be able to correctly identify any subset of size d from a population of size N. In this worst-case adversarial model, explicit schemes are known which make $m = O(d^2 \log(N))$ tests [40], and it is known that any scheme must have $m = \Omega(d^2 \log(N)/\log(d))$ [11].

Because we are not aware of any results about group testing in the random model (where the signal is composed of N independent Bernoulli random variables) or in the oblivious model, we first consider these models and note that $O(d \log(N))$ measurements suffice in either case. We give an explicit scheme that achieves this in the random model.

The rather large gap between the upper bound of $d \log(N)$ measurements in the random and oblivious cases, and the lower bound of $d^2 \log(N)$ measurements in the "for all" case indicates a need for intermediate models. As a step towards filling this gap, we consider the similar case where x is generated by a information-theoretically bounded adversary, and by computationally bounded adversaries.

Our results show that moving from the restrictive "for each" model to the more general space bounded adversary does not result in a decrease in efficiency. On the other hand, we show lower bounds even for very weak computationally bounded adversaries. We show that any scheme where Φ has $m = O(d^2/\log(d))$ rows will fail against a randomized log-space adversary with one-pass streaming access to Φ, or by a deterministic adversary with $O(\log N \log m)$ space. As we will see in Section 4, this contrasts with the sparse recovery case, where a logspace streaming adversary can be defeated with the optimal number of measurements.

3.1. Binary symmetric channel.

In this section, we give a concrete construction of a group testing design which recovers a random set of defectives with high probability over the choice of defectives. We consider a model where each item in the population is defective with probability d/N, independent of all other items. Thus, the expected number of defective items is d. In the setting of error-correcting codes, this corresponds to switching from Hamming's adversarial model of noise to Shannon's binary symmetric channel.

Proposition 2. *For any $0 < \delta < 1$, there is a group testing scheme with $O\left(d \log \frac{d}{\delta} \log\left(\frac{N}{d} \log \frac{d}{\delta}\right)\right)$ tests that succeeds with probability $1 - \delta$ against input distributions where each item is defective independently with probability d/N.*

Proof. The idea is very simple: divide up domain (length N) into N/t chunks of size t each. If t is not too large, in each chunk the expected number of ones is small. Thus with high probability, each chunk will have very few ones (call that number x).

Our idea is then to create a block a block diagonal matrix, where we use a standard x-disjunct matrix for each of the N/t chunks. The construction of [40] gives an x-disjunct matrix over a population of size t with $O(x^2 \log(t))$ rows. Our test is simply a block diagonal matrix consisting of d copies of this x-disjunct matrix, so our scheme requires $(N/t)x^2 \log(t)$ tests.

It remains to choose x so that the probability of error is bounded by δ; the fact that we can choose an appropriate x follows from some union bounds. The details are in the full version. □

There are, in fact, explicit constructions of x-disjunct matrices that can be constructed in polynomial time [40]; furthermore, these can be decoded in sublinear time [38].

This scheme is nearly optimal:

Lemma 3. *Any (non-adaptive) group testing scheme to recover from error rate d/N in the random model with probability $(1-\delta) = \Omega(1)$ requires at least $NH(d/N) = d \log\left(\frac{N}{d}\right) + (N-d)\log\left(\frac{N}{N-d}\right)$ tests.*

The proof can be found in the full version of this paper.

While Lemma 3 relies on the fact that every group testing scheme is a compression algorithm, the converse is not true. Group testing schemes are compression algorithms that can be represented as a matrix product Φx, where the arithmetic is boolean AND and OR. This is similar to using the parity-check matrix of binary error-correcting code for noiseless compression [6] except the matrix multiplication is done using OR instead of XOR.

3.2. **Oblivious adversary.** We also consider the oblivious model, in which an adversary Mallory, with no knowledge of the design matrix Φ, chooses a pattern of defectives to create $x \in \{0,1\}^N$. For any x she chooses, we demand that with high probability over the construction of Φ, x is recoverable from Φx and Φ. The construction in Proposition 2, post-multiplied by a random permutation matrix, works in the oblivious setting with the same argument and the same number of measurements. We observe that, using a random matrix, we may correct the $\log(d/\delta)$ factor at the cost of switching from $\log(N/d)$ to $\log N$:

Proposition 4. *For any distribution of vectors $x \in \{0,1\}^N$, where $\Pr[|\mathrm{Supp}(x)| > d] < \delta$, there is a distribution of design matrices Φ with $O(d \log(N))$ rows such that with probability at least $1-2\delta$ over the choice of Φ, and the choice of x, the signal x can be exactly recovered from Φx and Φ, using boolean arithmetic.*

The distribution where each entry of Φ is indepently 1 or 0 will work. It suffices to show that within any fixed set of d columns, no one column is contained in the union of any of the others, except with small probability. The details are in the full version.

Corollary 5. *A $m \times n$ matrix where each entry is 1 with probability $1/d$, and 0 with probability $1-1/d$ represents a pooling design against an oblivious adversary which recovers any weight d input probability at least $1-\delta$ over the choice of the matrix, where $m = 4d \log\left(\frac{N}{\delta}\right)$.*

3.3. **Group testing against a information-theoretically bounded adversary.** The error probability in Proposition 4 along with Observation 1 immediately implies that a logspace information theoretically bounded adversary with access to the a matrix Φ drawn from the distribution described above will also not succeed:

Corollary 6. *Let Φ be a matrix so that each entry is 1 independently with probability $1/d$, with $m = O(d \log(N))$ rows. For any information-theoretically bounded algorithm M with $O(\log N)$ space which sees Φ and then generates x, x can be recovered exactly from Φx and Φ with probability at least $2/3$.*

3.4. **Lower bounds against computationally bounded adversaries.** In this section, we show that the good news in group testing stops at oblivious or information-theoretically bounded adversaries. For even very simple computationally bounded adversaries, $O(d^2/\log d)$ measurements are needed. We begin with a randomized adversary and then later show it can be de-randomized to obtain the second adversary.

Theorem 7. *Suppose Φ is an $m \times N$ binary matrix with $m = O(d^2/\log d)$ rows, and additionally assume $m = o(N)$. There is a algorithm with $O(\log(N))$ space which streams over the rows of Φ and outputs a set $T \subset [N]$ so that $|T| \leq d$ and so that the characteristic vector x of T satisfies the following property: there is some $i \notin T$ so that if y is the characteristic vector of $T \cup \{i\}$, $\Phi x = \Phi y$. In particular, accurate recovery of x is impossible.*

To prove Theorem 7, we describe in Algorithm 1 a one pass low space adversary who streams over Φ row-by-row and outputs a list of indices which will determine the set T of defectives.

Algorithm 1: A logspace streaming adversary which outputs a set T of defectives which will resist recovery.

(1) Independently at random output each of the N column indices with probability $p = \Theta\left(\frac{x \log m}{N}\right)$, where x will be chosen in the proof. Note that the above is done even without looking at the matrix Φ.
(2) Pick a random $i \in [N]$ and remember it.
(3) For each row index $r \in [m]$:
  (i) Count the number of ones y in row r;
  (ii) Remember the first column $j \neq i$ such that $\Phi_{r,j} = 1$.
  (iii) Set $B = \Phi_{r,i}$.
  (iv) If ($B \wedge y \leq N/x \wedge j$ is defined), output j. Otherwise, do nothing.
  (v) Stop if the algorithm has output $d/2$ (not necessarily distinct) j indices in Step 3(iv).

It is easy to check that the above algorithm can be implemented in one pass and $O(\log N)$ space.

The proof of correctness is can be found in the full version of this paper. The basic idea is that this adversary will take care of the "heavy" rows (those with many ones) and the "light" rows separately. The goal is to choose x large enough so that enough of the heavy rows are covered, and small enough that not more than d indices are selected. This is possible as long as $m = O(d^2/\log(d))$.

Algorithm 1 can be de-randomized to produce an equivalent deterministic low space algorithm. (Note that in this setting we will allow the adversary to make multiple passes over Φ.

Theorem 9. *For any matrix Φ with $O(d^2/\log(d))$ rows, there is a deterministic algorithm with space $O(\log m \log N)$ which generates $x \in \{0,1\}^N$ with at most d nonzeroes so that for some $y \in \{0,1\}^N$ with at most d non-zeros, $\Phi x = \Phi y$. In particular, recovery of x is impossible.*

The proof is in the full version.

4. Sparse signal recovery

In sparse signal recovery, we wish to recover a signal x from measurements Φx with as little error as possible. We will use the notation that $x = y + z$, where y is the head (the largest k terms) and z is the tail. In this section, we suppose an adversary who generates the tail z, while the head y is assumed to be worst-case.[10]

[10]Note that, beyond the distiction between a statistical model and a computational model, this is more general than many random signal models which assume that *both* the head and the tail are random.

The strongest possible bound on error between the original signal x and the recovered approximation $\hat{x}$ is an ℓ_2/ℓ_2 requirement:

$$\|x - \hat{x}\|_2 \leq C\|x - x_k\|_2. \tag{1}$$

This bound is achievable with $m = O(k \log(N/k))$ (see [22]) in the oblivious model, meeting a lower bound [2]. On the other hand, [12] show that in the adversarial model, (1) is impossible unless $N = O(m)$.

In this section, we show that even when relaxing the adversary to the logspace (information theoretically bounded) model or logspace streaming model, (1) is still attainable with an optimal number of rows, circumventing the lower bound in [12].

4.1. **Information-Theoretically Bounded Adversaries.** In this section we consider an adversary who must pass through a $\log(N)$-space information-theoretic bottleneck.

The Restricted Isometry Property (RIP) is a useful criterion for generating matrices for sparse recovery:

Definition 11. *A matrix Φ satisfies the **Restricted Isometry Property** with constant δ if for every k-sparse vector x, $(1-\delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta)\|x\|_2^2$.*

In [12] it is shown that, when Φ has the Restricted Isometry Property, a sufficient condition for unique recovery against an oblivious adversary is that no "tail" z is very stretched by Φ. This implies that unique recovery is possible with ℓ_2/ℓ_2 error as long as Mallory cannot find z so that $\|\Phi z\|_2 \geq C\|z\|_2$. From the fact that for several standard ensembles (Gaussian, Bernoulli, etc) an oblivious adversary is very unlikely to find such z (see for example [1]), Lemma 1 implies that neither is Mallory. This does not imply that *efficient* recovery is possible, but in fact several existing (efficient) algorithms will work. Many algorithms ([36], [35]) recover an exactly k-sparse x in the adversarial setting which are stable in the ℓ_2/ℓ_2 sense against some post-measurement noise. That is, if x is k-sparse, given $\Phi x + e$, such algorithms recover $\hat{x}$ so that

$$\|x - \hat{x}\|_2 \leq C\|e\|_2. \tag{2}$$

This immediately gives an algorithm which works against an information-theoretically bounded adversary with logarithmic space.

Proposition 13. *Suppose that Φ is chosen to have independent Bernoulli or Gaussian entries. Suppose A is an algorithm which recovers an exactly k-sparse vector x from $\Phi x + e$ so that (2) holds. Then A will succeed with high probability on any vector x generated by a logspace information theoretically bounded adversary with access to Φ.*

A downside of Proposition 13 is that we would like to use a more combinatorial approach, for several reasons. First, these schemes tend to use sparser matrices and have faster recovery times. Secondly, and more importantly for the purposes of this paper, we will see in Section 4.2 that combinatorial algorithms will extend to work against a logspace streaming adversary as well as an information theoretically bounded adversary.

Our construction is based on several constructions in the literature, including [22] and [39]. Unlike those constructions, however, our algorithm will have superlinear runtime. It follows we can afford several simplifications. We do need strong guarantees of failure to take a union bound over all possible heads of signals.

At a high level, the idea is to create Φ out of $O(\log N)$ hash matrices with $O(k)$ buckets each. Additionally, each nonzero element in the hash matrix is subject to a random sigh flip. With high probability, at most one element of the head is hashed into each bucket, and the other (smaller) elements in the bucket are likely to cancel each other out. The savings in the number of rows and the error probability over other hashing-type algorithms comes from the recursive nature of the reconstruction algorithm. To recover, a block of hash matrices is used to identify the top half of

the heavy hitters, which are then subtracted off. The process is repeated with the remaining hash matrices.

The algorithm itself and a proof of its correctness can be found in the full version of the paper.

4.2. **A Streaming Adversary.** In this section, we claim that the above combinatorial algorithm continues to work with a logspace streaming adversary, at the cost of some constant factors.[11]

Theorem 17. *Suppose $x = y + z$, where y is an arbitrary k-sparse vector and z is generated synchronously by an adversary with $O(\log(N))$ space streaming over the rows of a matrix Φ. Suppose that $k = o(N/(\log(N/k)\log N))$ and $k = \Omega(\log N)$. There is a distribution on $m \times N$ matrices Φ with $m = O(\epsilon^{-1}k\log(N/k))$ measurements and an efficient recovery algorithm which returns $\hat{x}$ so that $\|x - \hat{x}\|_2 \leq O(\epsilon)\|x - x_k\|_2$ with probability at least $2/3$.*

The analysis uses communication complexity. Briefly, the idea is that as Mallory streams over the rows, she only outputs a few entries relevant to that row, for most of the rows. She outputs the other entries of the tail based only on her (limited) memory. Because the relevant analysis is resilient to a constant fraction of errors in each block, we recover from her informed outputs, and we may consider only those that she is trying to remember.

To make things easier on Mallory, we assume that she has unlimited computational power while inspecting any one row, but that she can only carry $\log(N)$ bits with her between rows. We cast this as a communication problem for players Alice and Bob, who are trying to solve the augmented indexing. Given an instance of augmented indexing, we construct a matrix Φ so that if Mallory could remember enough information after each block to find a problematic tail for the recover algorithm, then Alice could use Mallory's memory to send Bob a short message which would allow him to solve their problem. This would violate a known bound in communication complexity.

The details are in the full version of the paper.

5. Conclusions and Future Work

We present several new models for signal generation in combinatorial group testing and sparse recovery. These models capture the many natural situations in which the signal has a weak dependence on the measurement matrix, or when subsequent signals one observes have some weak dependence on the measurement matrix or upon the measurements obtained. It is often more natural to assume that the process generating this signal is either computationally bounded, or has limited information about the measurement matrix Φ than to assume that this process conforms to particular geometric requirements or follows a certain distribution. We show that there are significant gains (in the minimum number of measurements required) to be had by relaxing the model from adversarial to computationally or information-theoretically bounded, and not too much (in some cases, nothing at all) is lost by assuming these models over oblivious or statistical models. We also show that in the group testing case, there is a difference between information-theoretically bounded and computationally bounded (streaming) models, which contrasts the situation in sparse recovery.

One model we have not discussed is that of a polynomial-time bounded adversary, with cryptographic assumptions, which is a natural next step. It is perhaps of more practical use to consider sparse recovery or group testing against such an adversary.

Acknowledgements

The authors thank Hung Ngo for many helpful discussions.

[11] It seems likely that no "geometric" algorithm would be able to succeed against such an adversary, where "geometric" means based on the RIP and ℓ_2 distances. Indeed, a streaming adversary can generate a vector z in the row space of Φ via matrix multiplication, which would have $\|\Phi z\|$ much greater than $\|z\|$.

References

[1] D Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 66(4):671–687, 2003.

[2] K Do Ba, P Indyk, E Price, and D.P Woodruff. Lower bounds for sparse recovery. *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1190–1197, 2010.

[3] R Baraniuk, V Cevher, M Duarte, and C Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, Jan 2008.

[4] R Baraniuk and P Steeghs. Compressive radar imaging. *IEEE Radar Conference 2007*, pages 128—133, Jan 2007.

[5] T. Berger, N. Mehravari, D. Towsley, and J. Wolf. Random multiple-access communications and group testing. *IEEE Trans. Commun.*, 32(7):769–779, 1984.

[6] G. Caire, S. Shamai, and S. Verd'u. Noiseless data compression with low-density parity-check codes. In *Advances in Network Information Theory (DIMACS '04)*, pages 263–284, 2004.

[7] Robert Calderbank, Stephen Howard, and Sina Jafarpour. A sublinear algorithm for sparse reconstruction with l2/l2 recovery guarantees. *arXiv*, cs.IT, Jun 2008.

[8] Candes, Romberg, and Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2), 2006.

[9] E Candes, M Rudelson, T Tao, and R Vershynin. Error correction via linear programming. 2005.

[10] V. Cevher, M.F. Duarte, C. Hegde, and R.G. Baraniuk. Sparse signal recovery using markov random fields. In *Proc. Workshop on Neural Info. Proc. Sys.(NIPS)*. Citeseer, 2008.

[11] Shiva Chaudhuri and Jaikumar Radhakrishnan. Deterministic restrictions in circuit complex- ity. pages 30–36, 1996.

[12] A Cohen, W Dahmen, and R DeVore. Compressed sensing and best k-term approximation. *American Mathematical Society*, 22(1):211–231, 2009.

[13] Graham Cormode and S. Muthukrishnan. What's hot and what's not: tracking most frequent items dynamically. *ACM Trans. Database Syst.*, 30(1):249–278, 2005.

[14] D Donoho. Compressed sensing. *Information Theory*, 52(4):1289–1306, Jan 2006.

[15] R. Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943.

[16] Ding-Zhu Du and Frank K. Hwang. *Combinatorial group testing and its applications*, volume 12 of *Series on Applied Mathematics*. World Scientific Publishing Co. Inc., River Edge, NJ, second edition, 2000.

[17] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 2008.

[18] A. G. Dýachkov and V. V. Rykov. Bounds on the length of disjunctive codes. *Problemy Peredachi Informatsii*, 18(3):7–13, 1982.

[19] A. G. Dýachkov, V. V. Rykov, and A. M. Rashad. Superimposed distance codes. *Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform.*, 18(4):237–250, 1989.

[20] Yaniv Erlich, Kenneth Chang, Assaf Gordon, Roy Ronen, Oron Navon, Michelle Rooks, and Gregory J. Hannon. Dna sudoku—harnessing high-throughput sequencing for multiplexed specimen analysis. *Genome Research*, 19:1243—1253, 2009.

[21] Zoltán Füredi. On r-cover-free families. *J. Comb. Theory, Ser. A*, 73(1):172–173, 1996.

[22] A.C Gilbert, Y Li, E Porat, and M.J Strauss. Approximate sparse recovery: Optimizing time and measurements. *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 475–484, 2010.

[23] Michael T. Goodrich, Mikhail J. Atallah, and Roberto Tamassia. Indexing information for data forensics. In *Third International Conference on Applied Cryptography and Network Security (ANCS)*, pages 206–221, 2005.

[24] V Guruswami and A Smith. Codes for computationally simple channels: Explicit constructions with optimal rate. *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 723–732, 2010.

[25] Kainkaryam. Pooling in high-throughput drug screening. *Current Opinion in Drug Discovery & Development*, 12(3):339–350, May 2009.

[26] Raghunandan Kainkaryam and Peter Woolf. poolhits: A shifted transversal design based pooling strategy for high-throughput drug screening. *BMC Bioinformatics*, 9(1), 2008.

[27] Sherif M. Khattab, Sameh Gobriel, Rami G. Melhem, and Daniel Mossé. Live baiting for service-level dos attackers. In *INFOCOM*, pages 171–175, 2008.

[28] Sami Kirolos, Jason Laska, Michael Wakin, Marco Duarte, Dror Baron, Tamer Ragheb, Yehia Massoud, and Richard Baraniuk. Analog-to-information conversion via random demodulation. In *IEEE Dallas Circuits and Systems Workshop (DCAS)*, Dallas, Texas, Oct. 2006.

[29] E. Knill. Lower bounds for identifying subset members with subset queries. In *Proceedings of the sixth annual ACM-SIAM symposium on Discrete algorithms*, SODA '95, pages 369–377, Philadelphia, PA, USA, 1995. Society for Industrial and Applied Mathematics.

[30] Amos Lapidoth and Prakash Narayan. Reliable communication under channel uncertainty. *IEEE Transactions on Information Theory*, 44:2148–2177, 1998.

[31] Jason Laska, Sami Kirolos, Yehia Massoud, Richard Baraniuk, Anna Gilbert, Mark Iwen, and Martin Strauss. Random sampling for analog-to-information conversion of wideband signals. In *IEEE Dallas Circuits and Systems Workshop (DCAS)*, Dallas, Texas, Oct. 2006.

[32] Richard J. Lipton. A new approach to information theory. In *STACS '94: Proceedings of the 11th Annual Symposium on Theoretical Aspects of Computer Science*, pages 699–708, London, UK, 1994. Springer-Verlag.

[33] Marc Mézard and Cristina Toninelli. Group testing with random pools: optimal two-stage algorithms. *CoRR*, abs/0706.3104, 2007.

[34] Silvio Micali, Chris Peikert, Madhu Sudan, and David A. Wilson. Optimal error correction against computationally bounded noise. In *TCC*, pages 1–16, 2005.

[35] D Needell and J.A Tropp. Cosamp: iterative signal recovery from incomplete and inaccurate samples. *Communications of the ACM*, 53(12):93–100, 2010.

[36] D Needell and R Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):310–316, 2010.

[37] Hung Q. Ngo and Ding-Zhu Du. A survey on combinatorial group testing algorithms with applications to DNA library screening. In *Discrete mathematical problems with medical applications (New Brunswick, NJ, 1999)*, volume 55 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 171–182. Amer. Math. Soc., Providence, RI, 2000.

[38] Hung Q. Ngo, Ely Porat, and Atri Rudra. Efficiently decodable error-correcting list disjunct matrices and applications. In *Proceedings of the 38th International Colloquium on Automata, Languages and Programming (ICALP)*, 2011. To appear.

[39] E Porat and M J Strauss. Sublinear time, measurement-optimal, sparse recovery for all. *submitted to SODA 2012*, 2011.

[40] Ely Porat and Amir Rothschild. Explicit non-adaptive combinatorial group testing schemes. In *ICALP '08*, volume 5125, pages 748–759, 2008.

[41] Milton Sobel and Phyllis A. Groll. Binomial group-testing with an unknown proportion of defectives. *Technometrics*, 8(4):pp. 631–656, 1966.

[42] Dharmpal Takhar, Jason Laska, Michael B. Wakin, Marco F. Duarte, Dror Baron, Shriram Sarvotham, Kevin Kelly, and Richard G. Baraniuk. A new compressive imaging camera architecture using optical-domain compression. In *Proc. IS&T/SPIE Symposium on Electronic Imaging*, 2006.

[43] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Trans. Signal Proc.*, 50(6), June 2002.

[44] Michael Wakin, Jason Laska, Marco Duarte, Dror Baron, Shriram Sarvotham, Dharmpal Takhar, Kevin Kelly, and Richard Baraniuk. Compressive imaging for video representation and coding. In *Proc. Picture Coding Symposium 2006*, Beijing, China, Apr. 2006.

[45] J. K. Wolf. Born again group testing: multiaccess communications. *IEEE Transaction on Information Theory*, IT-31:185–191, 1985.

[46] Xiaofeng Xin, Jean-François F. Rual, Tomoko Hirozane-Kishikawa, David E. Hill, Marc Vidal, Charles Boone, and Nicolas Thierry-Mieg. Shifted transversal design smart-pooling for high coverage interactome mapping. *Genome research*, 19(7):1262–1269, July 2009.

[47] Y.H. Zheng, D. J. Brady, M. E. Sullivan, and B. D. Guenther. Fiber-optic localization by geometric space coding with a two-dimensional gray code. *Applied Optics*, 44(20):4306–4314, 2005.

RECOVERING SIMPLE SIGNALS

ANNA C. GILBERT, BRETT HEMENWAY, ATRI RUDRA, MARTIN J. STRAUSS, AND MARY WOOTTERS

Abstract. The primary goal of compressed sensing and (non-adaptive) combinatorial group testing is to recover a sparse vector x from an underdetermined set of linear equations $\Phi x = y$. Both problems entail solving $\Phi x = y$ given Φ and y but they use different models of arithmetic, different models of randomness models for Φ, and different guarantees upon the solution x and the class of signals from which it is drawn. In [35], Lipton introduced a model for error correction where the channel is computationally bounded, subject to standard cryptographic assumptions, and produces the error vector x that must be found and then corrected. This has been extended in [25, 37] to create more efficient schemes against polynomial and logspace bounded channels. Inspired by these results in error correction, we view compressed sensing and combinatorial group testing as an adversarial process, where Mallory the adversary produces the vector x to be measured, with limited information about the matrix Φ. We define a number of computationally bounded models for Mallory and show that there are significant gains (in the minimum number of measurements) to be had by relaxing the model from adversarial to computationally or information-theoretically bounded, and not too much (in some cases, nothing at all) is lost by assuming these models over oblivious or statistical models. We also show that differences in adversarial power give rise to different lower bounds for the number of measurements required to defeat such an adversary. By contrast we show that randomized one pass log space streaming Mallory is almost as powerful as a fully adversarial one for group testing while for compressed sensing such an adversary as weak as an oblivious one.

compressed sensing, group testing, computationally bounded adversary

ACG and MW are at the Department of Mathematics, 2074 East Hall, University of Michigan, Ann Arbor, MI 48109. BH and MJS are with the Departments of Mathematics and Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109. AR is at the Department of Computer Science at the University at Buffalo SUNY, Buffalo, NY. ACG is an Alfred P. Sloan Research Fellow and has been supported in part by NSF DMS 0354600. MJS has been supported in part by NSF DMS 0354600 and NSF DMS 0510203. ACG and MJS have been partially supported by DARPA ONR N66001-06-1-2011. AR is supported by NSF CAREER grant CCF-0844796. Email: {annacg,bhemen,martinjs,wootters}@umich.edu and atri@buffalo.edu.

1. Introduction

Group testing was introduced by Dorfman in his seminal 1943 paper to identify soldiers in WWII who had syphilis [15]. Combinatorial group testing[1] has found applications in drug and DNA library screening (the literature is large, see [40, 28, 50, 27] and references thereof), multiple access control protocols [5, 49], live baiting of DoS attackers [29], data forensics [24] and data streams [13], among others. The reader is referred to the standard monograph on group testing for more details [16].

Compressed sensing was introduced much more recently in 2006 by [8, 14] as a method for acquiring signals in a "compressed" fashion and then reconstructing good (i.e., sparse) approximations to the signals from those observations. The problem has many practical applications from analog-to-digital conversion [30, 47, 34], novel optical device design [51, 46, 17], pooling designs for rare allele detection [20], and video acquisition [48], to cite a few examples.

Even though they were introduced more than half a century apart, compressed sensing and (non-adaptive) combinatorial group testing are similar. Their primary goal is to recover a sparse vector x from an underdetermined set of linear equations $\Phi x = y$. Both problems include solving $\Phi x = y$ given Φ and y but they use different models of arithmetic, different randomness models for Φ and different guarantees upon the solution x and the class of signals from which it is drawn. Compressed sensing attempts to recover a sparse signal $x \in \mathbb{R}^N$ from a small number of linear measurements Φx. The signal x is usually assumed to be composed of a "head", the k largest entries (in magnitude) of the vector, supported on a small number of indices, and a tail whose ℓ_1 or ℓ_2 norm is small. The goal is to recover a close approximation [2] $\widehat{x}$ to x from the measurements Φx. In the combinatorial group testing model the binary matrix Φ represents a pooling design and the goal is to recover a small set (of size d) of "defective" elements from a set of tests Φx. (Here $x \in \{0,1\}^N$.) In the group testing model, a test fails if any items in the group being tested are defective, so the matrix vector product Φx is done using boolean OR.

Compressed sensing and group testing share another similarity, which is a limited range of types of models that generate the input signal x. Existing models typically fall into one of two categories:

- **Adversarial or "for all" models.** An ambitious goal is to successfully recover all signals that satisfy a certain geometric property. In the compressed sensing this usually corresponds to a bound on size of the support of the head, and an ℓ_1- or ℓ_2-norm bound on the tail, although more restrictive models on the form of the support set (e.g., k non-zero elements must lie within $B < k$ contiguous blocks or on contiguous paths in a tree [3]). In combinatorial group testing settings these correspond to restrictions on the Hamming weight of the signal.
- **Probabilistic or "for each" models.** These are two slightly different cases. One goal (that of the "for each" model in compressed sensing) is to recover a fixed, arbitrary vector with high probability over the construction of the matrix Φ. A similar goal is to successfully recover a random signal from a specified distribution with high probability over the *signal*, which has been considered in compressed sensing. Statistical models that have been considered include the uniform distribution or graphical models for the k-sparse head [7, 10, 3], possibly combined with i.i.d. Gaussian noise for the tail [7]. Somewhat surprisingly, neither the "for each" model nor the random signal model has received much attention in the (non-adaptive) group testing setting, to the best of our knowledge[3].

Both these extremal types of models have benefits and drawbacks. The advantage of the "for all" model is that it places minimal assumptions on the signal, which typically implies that schemes developed in this model also work in more benign models. The wide applicability of the "for

[1] In this paper we will only consider non-adaptive schemes.

[2] The output should satisfy $\|x - \widehat{x}\|_p \leq C\|x - x_k\|_q$ where $x - x_k$ is the tail of the vector. There are three combinations of norms ℓ_p/ℓ_q of interest: ℓ_2/ℓ_2, ℓ_2/ℓ_1, and ℓ_1/ℓ_1.

[3] There are, however, a number of adaptive group testing results for a variety of probabilistic models. See [31, 45, 36]

all" model comes at a cost, and usually implies that positive results are harder to achieve. More importantly, it is highly unlikely that a natural process would generate worst-case signals. On the other hand, the "for each" model typically leads to stronger positive results but is not applicable in as many situations. Schemes that specify particular distributions on the input tend to be specialized, and it is debatable if a natural signal producing process would be oblivious to Φ, or that a natural process would follow a simple, fully specified random process. One of the goals of this paper is to develop a model for signals in such a way that it captures natural process and the same time has the benefits of both the "for all" model (wider applicability of the schemes developed) and the "for each" model (stronger positive results).

While schemes in the "for each" model are the easiest to construct, in many situations a "for each" guarantee may not be sufficient. One problem that arises is that of feedback. In real-world applications the measurement matrix, Φ, will be fixed and used to recover many signals. This is acceptable if we can prove a "for all" guarantee on the matrix Φ, but if Φ only satisfies a "for each" guarantee problems may arise. One of the simplest types of problems is when future signals may depend on *the measurements* of current signals. As a simple example exhibiting feedback, imagine a vector x indicating the monthly sales counts of items in a store. The management may wish to keep a sketch, Φx, of the best selling items.[4] The store may then choose to update its layout or sales strategy depending on the result of the sketch.[5] Next month's sales will be influenced by this updated layout or strategy, and hence are dependent on Φx. A more adversarial situation could arise as follows. An attacker launches a missile at a ship. The ship, using some form of compressed sensing radar [4], detects the incoming missile and takes evasive action. The ship's evasive action is viewed by the attacker, and the attacker can adjust his second attack accordingly. Since the ship cannot be expected to choose a new measurement matrix, Φ, between successive attacks, the second attack may not be detected using a matrix that only satisfies a "for each" guarantee.[6] In both these cases, successive signals are subject to some sort of feedback. Signals that depend on some sort of feedback arise naturally, but these signals are not covered by the "for each" model. These types of signals fit naturally into a bounded adversarial model. We point out as well, that even if the attacker cannot be modeled as a simple process, the amount of information the attacker receives from viewing the target's evasive action is limited, so the attacker's view of Φx is naturally modeled as having to pass through an "information-theoretic bottleneck."

When considering intermediate signal models, it is instructive to consider the landscape in a related field: coding theory. Compressed sensing and group testing are closely related to coding, the parallel is most easily seen by viewing Φ as the parity-check matrix of a linear code, and the signal x as the error pattern [9]. Coding theory has a worst-case (for all) model pioneered by Hamming, as well as a probabilistic ("for each") models introduced by Shannon. Unlike compressed sensing and group testing, coding theory also has a rich literature on intermediate models. One such example is the arbitrarily varying channel [33]. Another example, that is more in line with computational thinking, is the notion of computationally bounded adversaries. In [35], Lipton introduced a model for error correction where the channel is computationally bounded, and is subject to standard cryptographic assumptions. This has been extended in [25, 37] to create more efficient schemes against polynomial and logspace bounded channels. Note that the "for all" model corresponds to a computationally unbounded adversarial model while the probabilistic models correspond to much weaker computationally bounded or oblivious adversaries.

[4]Suppose the sketch indicates that antihistamines are selling quite well, but sunscreen is doing poorly.

[5]In the example of the previous footnote, the manager suspects a high pollen count, which is confirmed through some arbitrary process involving checking the pollen count online or in a weather forecast. The store moves the sunscreen to a back shelf and, to be safe, also bug spray, and orders more facial tissues.

[6]In practice the measurement matrix, Φ, is often built into hardware and is extremely difficult to change on the fly.

Given the importance of compressed sensing and group testing, we believe it is important to develop an understanding of their properties under more than just the most extreme classes of signal. The primary goal of this paper is to initiate the study of intermediate models for both compressed sensing and group testing:

- **Computationally bounded models.** The goal here is to recover a signal generated by a computationally bounded adversary that must be recovered with high probability over the construction of Φ. This corresponds to the computationally bounded channels in coding theory. Like the bounded channel model for error correcting codes, we often imagine that the input for these problems is generated by some natural process; not necessarily powerful, but also not necessarily well-behaved, and, in particular, not necessarily independent of the measurement matrix Φ. Furthermore, computational conditions are arguably more natural for algorithms than geometric or statistical ones: a geometric model is useful mathematically, but does not give insight into how difficult it is to choose problematic vectors. Statistical models put the burden on the problem designer to produce a natural model, if one even exists. In contrast, a computationally bounded model is quite broad. As a special case we consider the *streaming adversary* where one is only allowed a single pass over Φ and uses only logarithmic amount of space.
- **Information theoretically bounded models.** Another way to have a spectrum of adversaries it to characterize recoverable signals in terms of information content. Here, the goal to recover all signals whose mutual information with Φ is small. Note that the "for each" model corresponds to zero mutual information and the "for all" model corresponds to the full information case.

A second goal of this paper is to compare the compressed sensing and group testing frameworks. On the one hand group testing is an easier task than compressed sensing as we only need to handle binary vectors (and there is no measurement noise[7]). On the other hand each measurement gives very little information about x as each measurement in group testing is just an OR of the bits). Our metric of comparison is the minimum number of measurements required to recover the signal or the defective set, as in both problems a primary goal is to minimize the number of measurements (the rows of Φ) necessary for recovery.

Let k denote both the size of the "head" of the signal and of the defective set. Then, if we compare group testing to compressed sensing with ℓ_1/ℓ_1 error guarantees, it would seem that compressed sensing is more powerful as $O(k \log(N/k))$ measurements suffice [2] to perform (even) "for all" ℓ_1 compressed sensing while one needs $\Omega(k^2 \log_k N)$ measurements [18, 19, 22] to recover k sparse binary vectors by group testing. On the other hand, if we are interested in ℓ_2/ℓ_2 error bounds in compressed sensing, then we must use $\Omega(N)$ measurements in the "for all" setting [12]. By contrast, group testing only needs $O(k^2 \log(N/k))$ measurements [16] and thus, compressed sensing seems harder.[8]

Only $O(k \log(N/k))$ measurements are needed [23] in the "for each" model for ℓ_2/ℓ_2 recovery, which is optimal [2], and far away from the lower bound of $\Omega(N)$ in the "for all" model. Since this is the main gap between the "for all" and "for each" for compressed sensing, and we are interested in interpolating between these cases, we will consider only ℓ_2/ℓ_2 compressed sensing for the rest of the paper. Furthermore, ℓ_2/ℓ_2 norms are the most natural for signal processing as the ℓ_2 norm squared is the signal energy and the error can be expressed in terms of signal-to-noise ratio (SNR), a standard metric.

1.1. **Our results.** Our results are summarized in Table 1.

[7]Error-tolerant group testing is a well-studied topic but we will only concentrate on "error-less" group testing in this paper.

[8]Since in group testing we deal with binary vectors, all ℓ_p norms are the same whereas in compressed sensing the choice of the norm matters.

Combinatorial group testing		
Mallory	Num. Measurements	Reference
Adversarial	$\Omega(d^2 \log(N/d)/\log(d))$	[22, 18, 19]
Information-Theoretically bounded (logspace)	$O(d \log(N))$	this paper
Oblivious	$O(d \log(N))$	this paper
Binary symmetric channel	$\Omega(d \log(N/d))$, $O(d \log(N))$	this paper
Logspace streaming (one pass over the rows)	$\Omega(d^2/\log d)$	this paper
Deterministic $O(\log d \log N)$ space	$\Omega(d^2/\log d)$	this paper
Sparse signal recovery		
Mallory	Num. Measurements	Reference
Adversarial	$\Omega(n)$	[12]
Adversarial, but restricted so that $\|x - x_k\|_1 \leq \sqrt{k}\|x - x_k\|_2$	$O(k \log(N/k)$	[8, 14]
Information-Theoretically bounded (logspace)	$O(k \log(N/k))$	this paper
Logspace streaming (one pass over the rows)	$O(k \log(N/k))$	this paper
Oblivious	$O(k \log(N/k))$	[23]

TABLE 1. Results on the number of measurements in combinatorial group testing and sparse recovery various access models. For sparse recovery, all results obtain an ℓ_2/ℓ_2 error bound, that is, $\|x - \hat{x}\|_2 < C\|x - x_k\|_2$, where $\hat{x}$ is the recovered signal and x_k is the best k-term approximation.

We begin with our results on the "for each" or probabilistic models for group testing. We show that $O(d \log N)$ measurements suffice to recover a d-sparse binary vector x in the "for each" model as well as the case when x is generated by the well-known binary symmetric channel (each bit is one independently with probability d/N). This shows that (a) we gain in the number of measurements for group testing when we go from the "for all" to "for each" model and (b) group testing and (both ℓ_1/ℓ_1 and ℓ_2/ℓ_2) compressed sensing have similar complexities in the "for each" model. A natural follow-up question is whether we can distinguish between group testing and ℓ_2/ℓ_2 compressed sensing by an adversary that is weaker than the "for all" model (where we know there is a gap).

Somewhat surprisingly we are able to show that a randomized one pass $O(\log N)$ space streaming adversary suffices to distinguish between group testing and ℓ_2/ℓ_2 compressed sensing. In particular, we are able to show that $O(k \log(n/k))$ measurements suffice against such an adversary for ℓ_2/ℓ_2 compressed sensing, i.e., in some sense such an adversary is just as weak as a "for each" adversary. We use lower bounds in communication complexity to show upper bounds for recovery against a streaming adversary. On the other hand, we show that for group testing a randomized one pass log space streaming adversary is almost as powerful as a "for all" adversary by showing that $\Omega(d^2/\log d)$ measurement are needed to perform group testing against such an adversary. Using limited-wise hash functions, we can de-randomize this result to show that $\Omega(d^2/\log d)$ measurements are necessary to recover a d sparse binary vector against a deterministic $O(\log d \log N)$ space bounded adversary (with no a priori limit on the number of passes).

Finally, we show that for information-theoretically bounded adversaries with $O(\log N)$ bits of mutual information do no better than the oblivious model for both compressed sensing and group testing.

2. Preliminaries

2.1. **Notation.** There are three key components to both group testing and sparse signal recovery designs: the measurement matrix Φ, the unknown sample or signal x, and the observations y. We assume that x is of length N and that the matrix Φ has dimensions $m \times N$. In the combinatorial group testing literature, the number of rows m is typically denoted by t for tests but we will stick with the sparse signal recovery notation. The final parameter of interest in either problem is the number of defectives d that we seek (for group testing) or the sparsity k (for signal recovery). For this parameter, we use the convention of the two separate bodies of literature and express the number of measurements m in terms of d and N for group testing and k and N for sparse signal

recovery. One important distinction between group testing and sparse recovery signals is that for group testing, the signal x is a binary vector with d 1s, while for sparse recovery, the signal $x \in \mathbb{R}^N$ consists of two pieces: x_k, the k-largest entries in absolute value (called the "head" of the vector), and $x - x_k$, the remaining $N - k$ entries (called the "tail" of the vector).

Since we are considering these problems in an adversarial context, we will name the process generating inputs x Mallory. She will take on various powers throughout the paper.

2.2. **Models for Mallory.** We consider several models for Mallory.

- **Binary symmetric channel:** Entries in the signal x are 1 with probability p and 0 with probability $1 - p$, independent of the matrix Φ (analogous to the error correcting code setting in which bits in the encoded message are flipped with probability p). We consider this model for combinatorial group testing only and note that in this model, the number of defectives present in the signal is a random variable.
- **Oblivious:** Mallory cannot see the matrix Φ and generates the signal x independent from Φ. In the combinatorial group testing problem, Mallory chooses a set of size d to be the defective set. For sparse signal recovery, this model is equivalent to the "for each" signal model.
- **Information-Theoretic:** Mallory's output has bounded mutual information with the matrix. To cast this in a computational light, we say that an algorithm M is *(log-)information-theoretically-bounded* if $M(x) = M_2(M_1(x))$, where the output of M_1 consists of at most $O(\log(|x|))$ bits. Lemma 1 shows that this requirement can be expressed as a bound on the success probability of an oblivious adversary. We distinguish this case from the oblivious case because it provides a natural abstraction in a number of settings. As mentioned in the introduction, in a situation with feedback, it is reasonable to assume that an adversary (powerful or not) might only see a few bits about Φ based on our actions. This might also arise in a situation where a space-bounded streaming adversary does not start to generate x until it has seen all of Φ. This is obviously a much weaker adversary that a general information-theoretic bounded one, but it comes up naturally. For example, suppose measurements must be coordinated between several sensors, and Φ (or a seed for Φ) must be passed between them, through an environment which then generates the signals. The sensors do not begin to take measurements until after Φ has been initialized. The signals generated then depend on Φ, and hence an algorithm designed for an oblivious adversary would not give guaranteed results. However, the assumption is that the environment can be modeled as a simple process, perhaps with logarithmic space. A recovery algorithm designed for an information-theoretic bounded adversary would apply.
- **Streaming log-space:** Mallory can stream over Φ but only has log space with which to store information about Φ and to compute an error message. In contrast to the information theoretic model above, which puts restrictions on the amount of mutual information between the signal and Φ, this is a computational restriction. A logspace streaming adversary is the natural "weakest" computationally bounded adversary, and thus is a reasonable place to start our analysis. Additionally, it has practical applications: both the sensor example and the error correction example above could fit into this model as well.
- **Adversarial:** Mallory has full computational power.

Before we dive into technical results, we give a general statement about randomized algorithms and information theoretically bounded adversaries that we will use in the proceeding sections.[9] For a randomized algorithm A with failure probability $\epsilon > 0$, if an omnipotent adversary sees the seed r for A before choosing the input x, then A will fail. However, if the adversary has limited space, success is still possible:

[9]Despite the simplicity of this observation, or, perhaps because of its simplicity, we cannot find an appropriate reference for this result.

Lemma 1. *Pick $\ell = \ell(N)$, and fix $0 < \alpha < 1$. Let A be any randomized algorithm which takes input $x \in \{0,1\}^N$, $r \in \{0,1\}^m$, which "succeeds" with probability $1-\epsilon$. Then for any information theoretically bounded algorithm M with space ℓ, $A(M(r), r)$ succeeds with probability at least $\min\{1-\alpha,\ 1-\ell/\log(\alpha/\epsilon)\}$ over the choice of r.*

Proof. Let R be a uniform random variable over $\{0,1\}^m$, and let $X = M(R)$. Let $\delta = \Pr(A(X,R) \text{ fails })$. (Abusing notation, by $A(X,R)$, I mean that both $X = M(R)$ and R have the same draw from R). Then

$$\begin{aligned} I(X:R) &= \Pr(A(X,R) \text{ fails })I(X:R|A(X,R) \text{ fails }) + Pr(A(X,R) \text{ succeeds })I(X:R|A(X,R) \text{ succeeds}) \\ &\geq \delta I(X:R|A(X,R) \text{ fails }). \end{aligned}$$

Let $B = \{r \in \{0,1\}^m \ : \ A(M(r),r) \text{ fails }\}$. If $|B| < \alpha \cdot 2^m$, then by definition, A succeeds with probability at least $1-\alpha$ over the choice of r. So suppose that $|B| \geq \alpha \cdot 2^m$. Given that $A(X,R)$ fails, we may specify r from $M(r)$ as a member of a set of at most $\epsilon 2^m$ bad seeds for $M(r)$, using at most $\log(\epsilon 2^m)$ bits. On the other hand, to specify r should take at least

$$H(R \mid A(X,R) \text{ fails }) - I(R:X \mid A(X,R) \text{ fails })$$

bits. Altogether,

$$\begin{aligned} I(X:R) \mid A(X,R) \text{ fails}) &\geq H(R \mid A(X,R) \text{ fails }) - \log(\epsilon 2^m) \\ &\geq \log(\alpha \cdot 2^m) - \log(\epsilon \cdot 2^m) \\ &= \log(\alpha/\epsilon). \end{aligned}$$

Then

$$\ell \geq I(X:R) \geq \delta \log(\alpha/\epsilon),$$

so

$$\delta \leq \frac{\ell}{\log(\alpha/\epsilon)}.$$

□

3. Combinatorial group testing

The goal of group testing is to identify a set of d "defective" elements from a population of size N. This is accomplished by creating a pooling design, where each element is placed in multiple "pools" and each pool is tested separately. A pool will fail if it contains at least one defective element. The primary goal in this scenario is to minimize the number of pools, which is equivalent to minimizing the number of tests needed. It is customary to identify the set of defectives as a vector, x, of weight d in $\{0,1\}^N$, and a pool as a vector in $\{0,1\}^N$ identifying which elements are in the pool. The entire design, consisting of m pools, can be written concisely as a matrix $\Phi \in \{0,1\}^{m \times N}$. The outcome of the m tests then corresponds to the matrix product Φx, where the matrix product is computed using boolean AND an OR, since each test fails if *any* element in that pool is defective.

Combinatorial group testing schemes have focused on developing pooling designs that are robust against "worst-case" (adversarial) distribution of defectives. In particular, this means, that the group testing scheme must be able to correctly identify any subset of size d from a population of size N. In this worst-case adversarial model, explicit schemes are known which make $m = O(d^2 \log(N))$ tests [43], and it is known that any scheme must have $m = \Omega(d^2 \log(N)/\log(d))$ [11].

Because we are not aware of any results about group testing in the random model (where the signal is composed of N independent Bernoulli random variables) or in the oblivious model, we first consider these models and note that $O(d \log(N))$ measurements suffice in either case. We give an explicit scheme that achieves this in the random model.

The rather large gap between the upper bound of $d\log(N)$ measurements in the random and oblivious cases, and the lower bound of $d^2\log(N)$ measurements in the "for all" case indicates a need for intermediate models. As a step towards filling this gap, we consider the similar case where x is generated by a information-theoretically bounded adversary, and by computationally bounded adversaries.

Our results show that moving from the restrictive "for each" model to the more general space bounded adversary does not result in a decrease in efficiency, and is below the lower bound in the "for all" (adversarial) setting.

On the other hand, we show lower bounds even for very weak computationally bounded adversaries. We show that any scheme where Φ has $m = O(d^2/\log(d))$ rows will fail against a randomized log-space adversary with one-pass streaming access to Φ, or by a deterministic adversary with $O(\log N \log m)$ space. As we will see in Section 4, this contrasts with the sparse recovery case, where a logspace streaming adversary can be defeated with the optimal number of measurements.

3.1. **Binary symmetric channel.** In this section, we give a concrete construction of a group testing design which recovers a random set of defectives with high probability over the choice of defectives. We consider a model where each item in the population is defective with probability d/N, independent of all other items. Thus, the expected number of defective items is d. In the setting of error-correcting codes, this corresponds to switching from Hamming's adversarial model of noise to Shannon's binary symmetric channel.

Proposition 2. *For any $0 < \delta < 1$, there is a group testing scheme with $O\left(d\log\frac{d}{\delta}\log\left(\frac{N}{d}\log\frac{d}{\delta}\right)\right)$ tests that succeeds with probability $1-\delta$ against input distributions where each item is defective independently with probability d/N.*

Proof. The idea is very simple: divide up domain (length N) into N/t chunks of size t each. If t is not too large, in each chunk the expected number of ones is small. Thus with high probability, each chunk will have very few ones (call that number x).

Our idea is then to create a block a block diagonal matrix, where we use a standard x-disjunct matrix "for each" of the N/t chunks. We now show that with high probability this construction succeeds.

The construction of [43] gives an x-disjunct matrix over a population of size t with $O(x^2\log(t))$ rows. Our test is simply a block diagonal matrix consisting of d copies of this x-disjunct matrix, so our scheme requires $(N/t)x^2\log(t)$ tests.

It remains to choose x so that the probability of error is bounded by δ. If X is the number of defectives in a block, then

$$\begin{aligned}
\Pr[X > x] &= \sum_{i=x}^{t}\binom{t}{i}\left(\frac{d}{N}\right)^i\left(\frac{N-d}{N}\right)^{t-i} \\
&\le \binom{t}{x}\left(\frac{d}{N}\right)^x\left(\frac{N-d}{N}\right)^{t-x}\frac{x\left(\frac{N-d}{N}\right)}{x-td/N} \qquad \text{([21], Equation 3.4)} \\
&\le \binom{t}{x}\left(\frac{d}{N}\right)^x \le \left(\frac{te}{x}\right)^x\left(\frac{d}{N}\right)^x = \left(\frac{edt}{Nx}\right)^x \\
&< 2^{-x} \qquad (\text{for } x > 6\tfrac{dt}{N})
\end{aligned}$$

If we choose $x > \max\left(6\frac{dt}{N}, \log\frac{N}{t\delta}\right)$, then $\Pr[X > x] < \frac{\delta t}{N}$. Taking a union bound over the N/t blocks gives a probability of at most δ that any block contains more than x defectives.

We must now choose t to minimize the total number of measurements, $N/tx^2\log(t)$, subject to the constraint $x > \max\left(6\frac{dt}{N}, \log\frac{N}{t\delta}\right)$. Setting $t = \frac{N}{d}\log\frac{d}{\delta}$, the total number of measurements made is $O\left(d\log\frac{d}{\delta}\log\left(\frac{N}{d}\log\frac{d}{\delta}\right)\right)$. □

There are, in fact, explicit constructions of x-disjunct matrices that can be constructed in polynomial time [43]; furthermore, these can be decoded in sublinear time [41].

This scheme is nearly optimal:

Lemma 3. *Any (non-adaptive) group testing scheme to recover from error rate d/N in the random model with probability $(1-\delta) = \Omega(1)$ requires at least $NH(d/N) = d\log\left(\frac{N}{d}\right) + (N-d)\log\left(\frac{N}{N-d}\right)$ tests.*

Proof. This follows immediately from Shannon's Noiseless Coding Theorem. A group testing scheme making m tests on a population of size N can be regarded as a compression scheme from N bits to m bits. If the scheme recovers with a probability that does not approach zero, Shannon's Noiseless Coding Theorem tells us that the average codeword length must be $NH(d/N)$ giving the claimed bound. □

While Lemma 3 relies on the fact that every group testing scheme is a compression algorithm, the converse is not true. Group testing schemes are compression algorithms that can be represented as a matrix product Φx, where the arithmetic is boolean AND and OR. This is similar to using the parity-check matrix of binary error-correcting code for noiseless compression [6] except the matrix multiplication is done using OR instead of XOR.

3.2. **Oblivious adversary.** We also consider the oblivious model, in which an adversary Mallory, with no knowledge of the design matrix Φ, chooses a pattern of defectives to create $x \in \{0,1\}^N$. For any x she chooses, we demand that with high probability over the construction of Φ, x is recoverable from Φx and Φ. The construction in Proposition 2, post-multiplied by a random permutation matrix, works in the oblivious setting with the same argument and the same number of measurements. We observe that, using a random matrix, we may correct the $\log(d/\delta)$ factor at the cost of switching from $\log(N/d)$ to $\log N$:

Proposition 4. *For any distribution of vectors $x \in \{0,1\}^N$, where $\Pr[|\mathrm{Supp}(x)| > d] < \delta$, there is a distribution of design matrices Φ with $O(d\log(N))$ rows such that with probability at least $1-2\delta$ over the choice of Φ, and the choice of x, the signal x can be exactly recovered from Φx and Φ, using boolean arithmetic.*

Proof. Consider an $m \times N$ measurement matrix Φ, where each entry is chosen to be 1 with probability p, and 0 otherwise. Let v_j denote the jth column of Φ, and let $\mathrm{Supp}(v_j)$ denote the set of coordinates where $v_j \neq 0$. With probability $1-\delta$, the error pattern $x \in \{0,1\}^N$, has $|\mathrm{Supp}(x)| \leq d$. If this is not the case, we may fail to recover. In the case that $|\mathrm{Supp}(x)| \leq d$, we need to show that with high probability over the choice of Φ, for all $j \notin \mathrm{Supp}(x)$, there exists an $i \in [m]$, such that $v_j(i) = 1$, and $v_\ell(i) = 0$ for all $\ell \in \mathrm{Supp}(x)$. In other words, $v_j \not\subset \bigcup_{\ell \in \mathrm{Supp}(x)} v_\ell$.

We view x as fixed, and for any column, v_j ($j \notin \mathrm{Supp}(x)$), the probability that an index i has $v_\ell(i) = 0$ for all $\ell \in \mathrm{Supp}(x)$, and $v_j(i) = 1$, is exactly $p(1-p)^{|\mathrm{Supp}(x)|} \geq p(1-p)^d$. Since this probability is the same "for each" row index i, a column is "bad" with probability at most $(1-p(1-p)^d)^m$.

We would like to bound the probability that there exists a bad column by δ. There are $N-d$ columns not chosen by the support of x, so to take a union bound, we need $((1-p(1-p)^d)^m < \frac{\delta}{N-d}$. Solving for m yields $m > \frac{\log\left(\frac{N-d}{\delta}\right)}{\log(1-p(1-p)^d)}$. Setting $p = 1/d$, we find that $m > 4d\log\left(\frac{N}{\delta}\right)$ suffices. □

Corollary 5. *A $m \times n$ matrix where each entry is 1 with probability $1/d$, and 0 with probability $1-1/d$ represents a pooling design against an oblivious adversary which recovers any weight d input probability at least $1-\delta$ over the choice of the matrix, where $m = 4d\log\left(\frac{N}{\delta}\right)$.*

3.3. **Group testing against a information-theoretically bounded adversary.** The error probability in Proposition 4 along with Observation 1 immediately implies that a logspace information theoretically bounded adversary with access to the a matrix Φ drawn from the distribution described above will also not succeed:

Corollary 6. *Let Φ be a matrix so that each entry is 1 independently with probability $1/d$, with $m = O(d \log(N))$ rows. For any information-theoretically bounded algorithm M with $O(\log N)$ space which sees Φ and then generates x, x can be recovered exactly from Φx and Φ with probability at least $2/3$.*

Proof. By Proposition 4 an oblivious adversary succeeds against Φ with probability at most $O(1/N^3)$. Lemma 1 (with $\alpha = 1$, and adjusting the constants inside the $O(\cdot)$ appropriately) implies that if x is generated by a logspace adversary then recovery succeeds with probability at least $\min\left\{\frac{2}{3}, 1 - \frac{O(\log(N))}{O(\log(N^3))}\right\} = \frac{2}{3}$. □

3.4. **Lower bounds.** In this section, we show that the good news in group testing stops at oblivious or information-theoretically bounded adversaries. For even very simple computationally bounded adversaries, $O(d^2/\log d)$ measurements are needed. We begin with a randomized adversary and then later show it can be derandomized to obtain the second adversary.

3.4.1. *One pass log space randomized adversary.*

Theorem 7. *Suppose Φ is an $m \times N$ binary matrix with $m = O(d^2/\log d)$ rows, and additionally assume $m = o(N)$. There is a algorithm with $O(\log(N))$ space which streams over the rows of Φ and outputs a set $T \subset [N]$ so that $|T| \leq d$ and so that the characteristic vector x of T satisfies the following property: there is some $i \notin T$ so that if y is the characteristic vector of $T \cup \{i\}$, $\Phi x = \Phi y$. In particular, accurate recovery of x is impossible.*

To prove Theorem 7, we describe in Algorithm 1 a one pass low space adversary who streams over Φ row-by-row and outputs a list of indices which will determine the set T of defectives. Let x be an integer that will be fixed later.

Algorithm 1: A logspace streaming adversary which outputs a set T of defectives which will resist recovery.

(1) Independently at random output each of the N column indices with probability

$$p = \Theta\left(\frac{x \log m}{N}\right).$$

Note that the above is done even without looking at the matrix Φ.
(2) Pick a random $i \in [N]$ and remember it.
(3) For each row index $r \in [m]$:
  (i) Count the number of ones y in row r;
  (ii) Remember the first column $j \neq i$ such that $\Phi_{r,j} = 1$.
  (iii) Set $B = \Phi_{r,i}$.
  (iv) If ($B \wedge y \leq N/x \wedge j$ is defined), output j. Otherwise, do nothing.
  (v) Stop if the algorithm has output $d/2$ (not necessarily distinct) j indices in Step 3(iv).

It is easy to check that the above algorithm can be implemented in one pass and $O(\log N)$ space. Lemma 8 below shows that the adversary succeeds with high probability, which proves Theorem 7.

Lemma 8. *There exists an instantiation of x so that with high probability,*

(i) *Let T be the set of indices output in Step 1. Then*

$$|T| \leq \frac{d}{2}$$

(ii) *If S is the set of indices output then the ith column (which is chosen in Step 2) is contained in the union of the columns in S.*

Note that (i) and Step 4 ensure that $|S| \leq d$, as needed.

Proof. For (i), note that due to Chernoff, with probability $\exp(-\Omega(x \log m))$, in Step 1, $\Theta(x \log m)$ indices are output. Thus, the adversary outputs at most d nonzeros with probability at least $1 - \exp(-\Omega(d))$ as long as

$$x \leq O(d/\log m) \tag{1}$$

For (ii), call a row *heavy* if its weight is at least N/x. For a heavy row r, note that by step 1,

$$\Pr[|\{j|M_{r,j} = 1\} \cap T| \geq 1] \geq 1 - (1-p)^{N/x} \geq 1 - m^{-2},$$

where the last inequality follows by choosing the constants in the definition of p properly. Thus, by the union bound, with probability at least $1 - 1/m$, T will cover each of the N columns in the heavy rows. Let Φ' be the sub-matrix of Φ containing only the non-heavy rows. It now suffices to show that, with high probability, the indices output in step 3(iv) cover the ones in column i for all the rows included in Φ'. Note that the average column weight of Φ' is at most $N/x \cdot m \cdot \frac{1}{N} = \frac{m}{x}$. Pick δ small, to be determined later. As long as

$$x \geq \frac{2m}{\delta d} \tag{2}$$

we will have the average column weight at most $\delta \cdot d/2$. Thus, with probability at least $1 - \delta$, the column i chosen in Step 2 will have Hamming weight (in Φ') at most $d/2$.

There are two cases for such a column i:

(Case 1) column i has a "private" row r: i.e. $\Phi'_{r,i} = 1$ but for every $j \neq i$, $\Phi'_{r,j} = 0$.

(Case 2) Column i has no private row, i.e. for every row r in Φ' such that $\Phi'_{r,i} = 1$ there exist a $j \neq i$ such that $\Phi'_{r,j} = 1$.

The number of columns i that fall in Case 1 values is at most m, which by assumption is $o(N)$. Thus, column i chosen in Step 2 is in Case 2 with probability at least $1 - \delta - o(1)$. For such i's Step 3(iv) will output as many j's as the weight of i in Φ', which is at most $d/2$. (Note that the algorithm ignores all the heavy rows in Step 3.) Thus, with high probability $|S \setminus T| \leq d/2$, and step 4 will not be executed. Further, with high probability $S \setminus T$ covers all the non-heavy rows where i has a 1.

To complete the argument we only need to show that there exists a value of x that satisfies both (1) and (2). This is true if

$$m \log m \leq O(\delta d^2),$$

which is satisfied with $m \leq O(d^2/\log d)$, as desired. □

3.4.2. *Deterministic low space adversary.* In this section, we will derandomize Algorithm 1 and present an equivalent deterministic low space algorithm. We allow this adversary to make multiple passes over the input matrix Φ.

Theorem 9. *For any matrix Φ with $O(d^2/\log(d))$ rows, there is a deterministic algorithm with space $O(\log m \log N)$ which generates $x \in \{0,1\}^N$ with at most d nonzeroes so that for some $y \in \{0,1\}^N$ with at most d nonzeroes, $\Phi x = \Phi y$. In particular, recovery of x is impossible.*

Proof. We will show that the Algorithm 1 can be simulated with $O(\log m \log N)$ random bits and $O(\log N)$ space. (By simulated we mean that the algorithm either outputs "fail" or outputs a set $S \subseteq [N]$ such that $|S| \leq d$ and there exists an $i \in [N] \setminus S$ such that i lies in the union of the columns in S. Further, the new algorithm outputs "fail" with probability < 1.) Thus, by going

over all possible instantiations of the random bits, one obtains an $O(\log m \log N)$ deterministic space algorithm.

We now inspect the use of randomness in Algorithm 1. Note that Step 1 needs N bits of p-biased random bits and Step 2 needs $O(\log N)$ random bits. We will show that in Step 1, the p-biased random bits needs only be $O(\log m)$-wise independent while Step 2 does not need any randomness.

We start with the easier case of Step 2. One can verify that the proof of Lemma 8 works as long as the chosen column i has the following two properties:

(1) Column i does not have any private row and
(2) column i has at most $d/2$ ones in M'.

The proof of Lemma 8 shows that at least $1-\delta-o(1)$ fraction of $i \in [N]$ have this property. Given any $i \in [N]$, it is easy to check in $O(\log N)$ space if column i has the required property. (One can check that the following algorithm works: run steps 3(i)-(iii) of the algorithm and do the following instead of Step 3(iv): (1) Stop with "fail" if $B \wedge j$ is not defined and (2) Maintain a count C of how many j would have been output in Step 3(iv) of the algorithm. Finally, declare i to have the required property if the algorithm does not output "fail" and $C \leq d/2$.) Thus, one can compute the required $i \in [N]$ instead of the current Step 2 in the Algorithm 1.

We now move on to Step 1 of the algorithm. Step 1 needs N random bits (say $R = (R_1, \ldots, R_N) \in \{0,1\}^N$) such that each R_i is p-biased, i.e. $\Pr[R_i = 1] = p$. First, we argue that given R and M as input one can verify in $O(\log N)$ space whether $T = \{i | R_i = 1\}$ satisfies the following two properties:

(1) $|T| \leq d/2$ and
(2) T covers the heavy rows of M.

The adversary is done once she gets a hold of such a T. Verifying property (1) is easy as one can compute $\sum_{i=1}^{N} R_i$ in $O(\log N)$ space. Further, verifying (2) can also be done in $O(\log N)$ space as follows: Go over each row r of M and check if (a) row r is heavy and (b) if so, if there exists an $j \in [N]$ such that $M_{r,j} = 1$ and $R_j = 1$.

The proof of Lemma 8 shows that if each R_i is picked independently to be p-biased then with high probability $T = \{i | R_i = 1\}$ satisfies the properties (1) and (2) above. Next, we show that if the random bit vector R is chosen to be $O(\log m)$-wise independent p-biased bits in $\{0,1\}^N$, then there exists a T that satisfies properties (1) and (2). To this end, we recall the following Chernoff bound for limited independent sources (the statement below appear as part of Theorem 5 in [44]):

Theorem 10 ([44]). *Let $t \geq 1$ be an integer and let $Y_1, \ldots, Y_N$ be t-wise independent random variables taking values in $[0,1]$. Let $Y = \sum_{i=1}^{N} Y_i$ and μ be the mean of Y. Then for any $\gamma \leq 1$ and if*

$$t \leq \left\lfloor \gamma^2 \mu e^{-1/3} \right\rfloor,$$

then

$$\Pr\left[|Y - \mu| \geq \gamma\mu\right] \leq \exp\left(-\lfloor t/2 \rfloor\right).$$

For the rest of the discussion, let $t = 2\lceil \ln m \rceil + 4$. Now consider the random binary vector $R = (R_1, \ldots, R_N) \in \{0,1\}^N$, where $R_1, \ldots, R_N$ are t-wise independent bits where $\Pr[R_i = 1] = p$. Let T be the random set $\{i | R_i = 1\}$. We will show that with non-zero probability T satisfies properties (1) and (2) above.

By (a suitable) choice of x and p, we get that the expected value of $\sum_{i=1}^{N} R_i$ is $\mu = d/4$. Since $m = O(d^2/\log d)$, for large enough value of d, we have $t \leq \lfloor \mu e^{-1/3} \rfloor$. Thus by applying Theorem 10 (with $Y_i = R_i$ and $\gamma = 1$), we have that

$$\Pr[|T| \geq 2\mu = d/2] \leq \exp(-\ln m - 1) = \frac{1}{em} < \frac{1}{2}, \tag{3}$$

where $\mu = \mathbb{E}\left(\sum_{i=1}^{N} R_i\right)$ and the last inequality follows as $m \geq 1$.

We now verify property (2). Fix a heavy row $r \in [m]$ of M. For every $i \in [N]$, define $Y_i = 1$ if $\Phi_{r,i} = R_i = 1$ and otherwise define $Y_i = 0$. By definition of a heavy row and proper choice of x,

$$\mu \geq pN/x \geq \Omega(\log m).$$

By appropriate choice of the constant in the definition of p, we can ensure that $\mu \geq 2e^{1/3}(2\lceil \ln m \rceil + 4)$, which in turn implies that $t \leq \lfloor \gamma^2 \mu e^{-1/3} \rfloor$ for $\gamma = (\mu - 1)/\mu$. Thus by Theorem 10, we obtain that

$$\Pr\left[\sum_{i=1}^{N} Y_i = |\{i | M_{r,i} = 1\} \cap T| \leq 1\right] \leq \exp(-\ln m - 1) = \frac{1}{em}.$$

Thus, by the union bound over the at most m heavy rows, we get that T does not satisfy property (2) with probability at most $1/e < 1/2$. This along with (3) shows that with our choice of t-wise independent p-biased bits, there exists at least one T that satisfies both properties (1) and (2).

Finally, we need to determine how many purely random bits we need to obtain N p-biased random bits that are t-wise independent. For what follows we will assume without loss of generality that p is a power of 2. It is well-known that can one obtain N *unbiased* t-wise independent bits with $O(t \log N)$ random bits using BCH codes. To obtain p-biased t-wise independent bits we can consider a $t \log(1/p)$-wise independent $N \log(1/p)$ unbiased random bits. (Note that if we group $\log(1/p)$ bits together and output a 1 iff all these $\log(1/p)$ bits are 1 then the resulting N bits are t-wise independent p-biased bits.) This implies that one can get away with $O(\log m \log(N/d) \log N)$ purely random bits (as $p = \Theta(d/N)$ by our choice of p and x).

However, we can do slightly better. Consider a Reed-Solomon code of dimension t and block length N over an alphabet size $q = 2^{\lceil \log N \rceil}$. It is well-known that if one picks a random codeword from this code then this is a t-wise independent random vector in $\mathbb{F}_q^N$. To convert this to a t-wise p-biased random bits, replace each symbol in the codeword by a 1 iff the first $\log(1/p)$ bits of that symbol are all 1. Note that the probability that a random element in $\mathbb{F}_q$ leads to a 1 with the above conversion is $2^{\log q - \log(1/p)}/q = p$. Thus, the adversary now needs $O(t \log q) = O(\log m \log N)$ purely random bits (to pick a random message from $\mathbb{F}_q^t$), as desired.

The derandomization of Algorithm 1 is complete by noting that given a message in $\mathbb{F}_q^t$ and an index $i \in [N]$ one can compute the value of the ith symbol in the corresponding codeword in space $O(\log q) = O(\log N)$. □

4. Sparse signal recovery

In sparse signal recovery, we wish to recover a signal x from measurements Φx with as little error as possible. We will use the notation that $x = y + z$, where y is the head (the largest k terms) and z is the tail. In this section, we suppose an adversary who generates the tail z, while the head y is assumed to be worst-case.[10]

The strongest possible bound on error between the original signal x and the recovered approximation $\hat{x}$ is an ℓ_2/ℓ_2 requirement:

$$\|x - \hat{x}\|_2 \leq C\|x - x_k\|_2. \tag{4}$$

This bound is achievable with $m = O(k \log(N/k))$ (see [23]) in the oblivious model, meeting a lower bound [2]. On the other hand, [12] show that in the adversarial model, (4) is impossible unless $N = O(m)$.

[10]Note that, beyond the distiction between a statistical model and a computational model, this is more general than many random signal models which assume that *both* the head and the tail are random.

In this section, we show that even when relaxing the adversary to the logspace (information theoretically bounded) model or logspace streaming model, (4) is still attainable with an optimal number of rows, circumventing the lower bound in [12].

4.1. **Information-Theoretically Bounded Adversaries.** In this section we consider an adversary who must pass through a $\log(N)$-space information-theoretic bottleneck.

The Restricted Isometry Property (RIP) is a useful criterion for generating matrices for sparse recovery:

Definition 11. *A matrix Φ satisfies the **Restricted Isometry Property** with constant δ if for every k-sparse vector x,*

$$(1-\delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta)\|x\|_2^2.$$

In [12] it is shown that, when Φ has the Restricted Isometry Property, a sufficient condition for unique recovery against an oblivious adversary is that no "tail" z is very stretched by Φ.

Theorem 12 ([12])**.** *If Φ satisfies the Restricted Isometry Property with constant δ for $2k$-sparse vectors, and if Mallory cannot generate a tail z so that*

$$\|\Phi z\|_2 > \left(\frac{(C-1)(1+\delta)}{2}\right)\|z\|_2,$$

then for all k-sparse y, and for all z generated by Mallory, if $x = y + z$, any $\hat{x}$ which minimizes $\|\Phi\hat{x} - \Phi x\|_2$ satisfies

$$\|x - \hat{x}\|_2 \leq C\|x - x_k\|_2.$$

Theorem 12 implies that unique recovery is possible with ℓ_2/ℓ_2 error as long as Mallory cannot find z so that $\|\Phi z\|_2 \geq C\|z\|_2$. From the fact that for several standard ensembles (Gaussian, Bernoulli, etc) an oblivious adversary is very unlikely to find such z (see for example [1]), Lemma 1 implies that neither is Mallory. This does not imply that *efficient* recovery is possible, but in fact several existing (efficient) algorithms will work. Many algorithms ([39], [38]) recover an exactly k-sparse x in the adversarial setting which are stable in the ℓ_2/ℓ_2 sense against some post-measurement noise. That is, if x is k-sparse, given $\Phi x + e$, such algorithms recover $\hat{x}$ so that

$$\|x - \hat{x}\|_2 \leq C\|e\|_2. \tag{5}$$

This immediately gives an algorithm which works against an information-theoretically bounded adversary with logarithmic space.

Proposition 13. *Suppose that Φ is chosen to have independent Bernoulli or Gaussian entries. Suppose A is an algorithm which recovers an exactly k-sparse vector x from $\Phi x + e$ so that (5) holds. Then A will succeed with high probability on any vector x generated by a logspace information theoretically bounded adversary with access to Φ.*

Proof. If Φ is chosen to have independent Bernoulli or Gaussian entries, then [1] shows that for a fixed z,

$$Pr(\|\Phi z\|_2 \geq (1+\epsilon)\|z\|_2) \leq e^{-c_0(\epsilon)m},$$

for $c_0(\epsilon) = \frac{1}{2}(\epsilon^2/2 - \epsilon^3/3)$. If Mallory generates z, Lemma 1 (with $\alpha = \frac{1}{3}$), implies that with probability at least $2/3$ or

$$1 - O\left(\frac{\log N}{m\log(c_0(\epsilon))}\right),$$

which tends to 0 for any reasonable value of m, $\|\Phi z\|_2 \leq (1+\epsilon)\|z\|_2$. Thus, A returns $\hat{x}$ so that

$$\|x - \hat{x}\|_2 \leq C\|\Phi z\|_2 \leq C(1+\epsilon)\|z\|_2.$$

□

A downside of Proposition 13 is that we would like to use a more combinatorial approach, for several reasons. First, these schemes tend to use sparser matrices and have faster recovery times. Secondly, and more importantly for the purposes of this paper, we will see in Section 4.2 that combinatorial algorithms will extend to work against a logspace streaming adversary as well as an information theoretically bounded adversary.

Our construction is based on several constructions in the literature, including [23] and [42]. Unlike those constructions, however, our algorithm will have superlinear runtime. It follows we can afford several simplifications. We do need strong guarantees of failure to take a union bound over all possible heads of signals.

At a high level, the idea is to create Φ out of $O(\log N)$ hash matrices with $O(k)$ buckets each. Additionally, each nonzero element in the hash matrix is subject to a random sigh flip. With high probability, at most one element of the head is hashed into each bucket, and the other (smaller) elements in the bucket are likely to cancel each other out. The savings in the number of rows and the error probability over other hashing-type algorithms comes from the recursive nature of the reconstruction algorithm. To recover, a block of hash matrices is used to identify the top half of the heavy hitters, which are then subtracted off. The process is repeated with the remaining hash matrices.

4.1.1. *A Hashing-Based Algorithm Effective Against Bounded Adversaries.* In this section, we give the details of the algorithm we will use for our proofs. We stress that the main ideas here are from [42], and that the contribution of this work is meant to be the theoretical implications of the algorithm, not the algorithm itself.

An inner loop (a "weak" recovery system) recovers half of the heavy hitters in each round. This is wrapped inside a "top level" algorithm which iteratively finds all of the heavy hitters.

Each iteration in the outer loop corresponds to a $(t \cdot b) \times N$ matrix Φ_j consisting of t hashing $b \times N$ hash matrices $\Phi_j^{(i)}$ (the parameters b, t are defined in the algorithm). If nonzero, the value of $\Phi_j^{(i)}(a, b)$ is ± 1 independently with probability $1/2$. Several copies of Φ_j, with different parameters, are stacked together to get Φ.

Algorithm 2: A Weak System ([42])

Input: N, sparsity s, error tolerance η, omission ζ, Φx, and Φ
Output: $\hat{x}$, an approximation of x
$b = \frac{8es}{\eta\zeta}$ // number of hash buckets ;
$t = 2(2\zeta^{-1} + 1)\log(Ne/s)$;
for $j \in \{1, 2, \ldots, t\}$ **do**
  for $i \in I$ **do**
    $\hat{x}_i^{(j)} = s_{h_j(i),i}(\Phi x)_{h_j(i)}$;
    // $s_{h_j(i),i}$ is sign flip of i'th position in $h_j(i)$'th bucket, implicit in Φ ;
for $i \in [N]$ **do**
  $\hat{x}_i = \text{median}_j(\hat{x}_i^{(j)})$;
zero out all but the top s elements of $\hat{x}$.;
return $\hat{x}$

Algorithm 3: A Toplevel algorithm ([42])

Input: N, sparsity k, error tolerance ϵ, Φx, Φ
Output: $\hat{x}$, an approximation of x
for $j \in \{1, 2, \ldots, \log(k)\}$ **do**
  $s \leftarrow k/2^j, \eta \leftarrow O(\epsilon(3/4)^j), \zeta \leftarrow 1/2$;
  $x' \leftarrow$ the output of Algorithm 2 with the above parameters;
  $\hat{x} = \hat{x} + x'$;
  $\mu = \mu - \Phi x'$;
return $\hat{x}$;

Proposition 14 shows that this algorithm works against an infomation-theoretically bounded adversary.

Proposition 14. *Algorithm 3 uses $O(\epsilon^{-1}k\log(N/k))$ measurements and runs in times $O(N\log(N/k))$. If $x = y + z$, where y is an* arbitrary *k-sparse vector and z is generated by an information theoretically bounded adversary with $(1/4)\log(N)$ space, Algorithm 3 returns $\hat{x}$ so that $\|x - \hat{x}\|_2 \leq (1 + O(\epsilon))\|x - x_k\|_2$ with probability $2/3$, where x_k is the best k-term approximation to x.*

The proof is similar to that in [42]. For completeness, we will repeat the parts that must be tweaked in this setting. We rely on the following lemma.

Lemma 15. *Without loss of generality, scale $x = y + z$ so that $\|x\|_\infty = 1$.For a* fixed *z, and for* any *y with $\|y\|_0 = s$, Algorithm 2 returns $\hat{x}$ so that $x = \hat{x} + \hat{y} + \hat{z}$, so that $\|\hat{y}\|_0 \leq \zeta s$, $\|\hat{x}\|_0 = s$, and $\|\hat{z}\|_2 \leq (1 + O(\sqrt{\eta}))\|z\|_2$, with probability at least*

$$1 - \left(\frac{s}{N}\right)^s$$

over the construction of Φ.

It uses

$$O\left(\frac{s\log(N/s)}{\zeta^2\eta}\right)$$

measurements, and has running time

$$O(\zeta^{-1}N\log(N/s)).$$

Proof. For $j \leq t$ and $i \leq N$ We say that a measurement $\hat{x}_i^{(j)}$ has **failed** for x_i if

$$|\hat{x}_i^{(j)} - x_i| > \sqrt{\frac{\eta}{s}}\|z\|_2.$$

Similarly, say that the final measurement $\hat{x}_i$ (before zeroing out the smallest $N-k$) for x_i has **failed** if

$$|\hat{x}_i - x_i| > \sqrt{\frac{\eta}{s}}\|z\|_2.$$

Fix $S \subset [N]$ to be the support set of y, and $Z \subset [N]$ to be any set of size ζs. We will show that with high probability, at least one measurement $\hat{x}_i$ for $i \in Z$ has not failed. Taking a union bound over S and Z, we will conclude that with high probability, there are at most ζs failed measurements.

First, consider how the elements outside $S \cup Z$ hash: Let $X_{j,i}$ denote the sum of elements outside $S \cup Z$ that hash into the same bucket as x_i on trial j:

$$X_{j,i} := \hat{x}_i^{(j)} - \sum_{i' \in S \cup Z; h(i') = h(i)} \Phi_i(h(i), i') = \sum_{i \notin S \cup Z} \Phi_j(h(i), i) z_i.$$

Then $X_{i,j}$ has variance

$$\mathrm{Var}(X_{i,j}) = \sum_{i \notin S \cap Z} z_i^2 \mathrm{Var}(\Phi_j(\ell, i)) \leq \frac{\|z\|_2^2}{b}.$$

By Chebyshev's inequality,

$$\Pr\left(|X_{i,j}| \geq \sqrt{\frac{\eta}{s}}\|z\|_2\right) \leq \frac{s}{b^2\eta} = \frac{\zeta}{8e}.$$

The probability that two elements of $S \cup Z$ collide is also $\frac{\zeta}{8e}$, for a total failure probability of $\frac{\zeta}{4e}$. There are $|S \cup Z| \cdot t$ (item, iteration) pairs, and by a Chernoff bound (okay to apply here because of negative association), the probability that more than a ζ fraction of them fail is

$$\Pr\left(\text{more than } \frac{\zeta}{2}|S \cup Z|t \text{ (item, iteration) pairs fail}\right) \leq \left(\frac{1}{2}\right)^{\zeta|S \cup Z|t/2} \leq \left(\frac{s}{eN}\right)^{2s+\zeta}.$$

In the favorable case, there are at most a $\zeta/2$ fraction of pairs (i, j) with $i \in S \cup Z$ and $j \leq t$ so that $\hat{x}_i^{(j)}$ has failed. In this case, there are at most $\zeta|S \cup Z|$ elements that have a majority of failed estimates.

Taking a union bound over the choice of S and Z, we obtain that

$$\begin{aligned}
\Pr(\text{more than } \zeta|S \cup Z| \text{ measurements } \hat{x}_i \text{ fail}) &\leq \binom{N}{s}\binom{N}{\zeta s}\left(\frac{s}{eN}\right)^{2s+\zeta} \\
&\leq \left(\frac{s}{eN}\right)^s \zeta^{-\zeta s} \\
&\leq \left(\frac{s}{N}\right)^s.
\end{aligned}$$

Next we define $\hat{y}$ and $\hat{z}$. Then the claim that $x = \hat{x} + \hat{y} + \hat{z}$ follows as in the statement of the Lemma follows as the proof in [42]. Let $J = \mathrm{Supp}(\hat{x})$ and let $T = \mathrm{Supp}(y)$. Let B be the set of indices in $J \cup T$ that have failed. Then:

- If $i \notin T \cup J$, then i contributes x_i to $\hat{z}$. Since $i \notin \mathrm{Supp}(y)$, these i's contribute at most $\|z\|_2^2$ to $\|\hat{z}\|_2^2$.
- Pick an arbitrary bijection $\psi : T \setminus J \to J \setminus T$, and let $\psi(i) = i$ for $i \in J \cap T$. Then:
  - If either of $i, \psi(i) \in B$ (i gets a failed estimate), then x_i contributes to $\hat{y}$. There are at most $O(\zeta s)$ of these.
  - If both $i, \psi(i)$ are not in B (neither estimate fails, but the algorithm chose $\psi(i)$ over the better choice i), then $\psi(i)$ contributes $x_{\psi(i)} - \hat{x}_{\psi(i)}$ to $\hat{z}$ (and $\hat{x}_{\psi(i)}$ to $\hat{x}$). Also, i contributes x_i to $\hat{z}$. There are at most $O(s)$ contributions of the form $x_{\psi(i)} - \hat{x}_{\psi(i)}$, each of which is at most $\sqrt{\frac{\eta}{s}}\|z\|_2$, for total $\sqrt{\eta}\|z\|_2$. The contribution x_i to $\hat{z}$ is nearly

offset by $x_{\psi(i)}$ that contributes to z but *not* to $\hat{z}$, so the the net contribution of i and $\psi(i)$ to $\|\hat{z}\|_2^2 - \|z\|_2^2$ is under control; specifically, it is

$$\begin{aligned}(x_{\psi(i)} - \hat{x}_{\psi(i)})^2 + x_i^2 - x_{\psi(i)}^2 &\leq (x_{\psi(i)} - \hat{x}_{\psi(i)})^2 + \hat{x}_i^2 - \hat{x}_{\psi(i)}^2 + (x_{\psi(i)} - \hat{x}_{\psi(i)})^2 + (x_i - \hat{x}_i)^2 \\ &\leq O\left(\sqrt{\frac{\eta}{s}}\|z\|_2\right),\end{aligned}$$

since the i and $\psi(i)$ estimates are at most $O\left(\sqrt{\frac{\eta}{s}}\|z\|_2\right)$ and $\hat{x}_i^2 \leq \hat{x}_{\psi(i)}^2$ since the algorithm chose $\psi(i)$ over i.

Thus, the total contribution to $\|\hat{z}\|_2$ is $(1 + O(\sqrt{\eta}))\|z\|_2$, and $\hat{y}$ is indeed $O(\zeta s)$-sparse. □

Given this Lemma, the proof of Proposition 14 amounts to showing that the error probability is small enough to apply Lemma 1.

Proof. The failure probability of each of the $\log(k)$ iterations is $(s/N)^s$, which (after adjusting constants) is enough to take a union bound over all $\binom{N}{s}$ possible heads. So the total failure probability, for a fixed tail vector z, is at most

$$\sum_{j=1}^{\log(k)} \left(\frac{k}{2^j N}\right)^{k/2^j}.$$

This rapidly converging series is bounded by the largest term, $O(1/N)$. By Observation 1, (choosing $\alpha = 1/4$), the probability of failure when Mallory is generating z is at most $1/4$ or $\ell/\log(N/2)$. If $\ell = (1/4)\log(N)$, then this probability is bounded by $1/3$ as N grows. □

4.2. **A Streaming Adversary.** In this section, we show that Algorithm 3 continues to work with a logspace streaming adversary, at the cost of some constant factors.[11]

The analysis uses communication complexity. Briefly, Alice holds an input in $a \in A$, and sends one message m to Bob, who holds $b \in B$. Bob uses b and m to compute a function $f(a, b)$ correctly with probability at least $1 - \delta$ for some δ. The *randomized one-way communication complexity* of f is minimum over all protocols of the the number of bits communicated by Alice in the worst case for that protocol. Let $\textsc{Ind}_q : \mathbb{F}_q^N \times (\mathbb{F}_q \times [N]) \to \{0, 1\}$ be given by

$$\textsc{Ind}_q((x_1, x_2, \ldots, x_N), (z, i)) = \begin{cases} 1 & x_i = z \\ 0 & x_i \neq z \end{cases}.$$

This is a tweak on the standard indexing problem, in which $q = 2$. The standard indexing problem is known to have randomized one way communication complexity $\Omega(N)$ [32, 26], and the same is true (by a straightforward extension of the proof in [26]) for larger q:

Theorem 16. *For all δ sufficiently small (bounded above by a small constant), the randomized communication cost of $\textsc{Ind}_q$ with failure probability δ is $\Omega(N \log(q))$.*

We will show that if a streaming adversary with log space could defeat Algorithm 3, then there would be a one way protocol for two players, Alice and Bob, which solves $\textsc{Ind}_q$, using only $\log N$ space, contradicting Theorem 16. However, this approach will only show that Mallory's success probability must be less than $1 - \delta$, for a small constant δ. In order to reduce this probability to something smaller than $1/2$, we simply run the algorithm $O(1/\delta)$ times, recovering vectors $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{O(1/\delta)}$. With probability at least $1/2$, $\hat{x}_i = x$ is correct for some i. This means that

[11] It seems likely that no "geometric" algorithm would be able to succeed against such an adversary, where "geometric" means based on the RIP and ℓ_2 distances. Indeed, a streaming adversary can generate a vector z in the row space of Φ via matrix multiplication, which would have $\|\Phi z\|$ much greater than $\|z\|$.

with overwhelming probability, for each iteration Φ_j, $\Phi_j \hat{x}_i = \Phi_i x$ for each j, and this will not be true for the incorrect $\hat{x}_j$. Thus, we may pick out the correct preimage.

Theorem 17. *Suppose that $k = o(N/(\log(N/k)\log N))$ and $k = \Omega(\log N)$. Consider the algorithm which runs Algorithm 3 a constant number of times, as described above. This algorithm uses $O(\epsilon^{-1}k\log(N/k))$ measurements and runs in time $O(Nk\log(N/k))$. If $x = y + z$, where y is an arbitrary k-sparse vector and z is generated synchronously by an adversary with $O(\log(N))$ space streaming over the rows of Φ, then the algorithm returns $\hat{x}$ so that $\|x - \hat{x}\|_2 \leq O(\epsilon)\|x - x_k\|_2$ with probability at least $2/3$.*

Proof. As above, it suffices to show that Mallory cannot succeed except with probability $1 - \delta$ for δ bounded away from 0. The proof proceeds in two parts: When Mallory outputs an element z_i of z, she is looking at some row ϕ. We first control the entries of ϕ that are directly impacted by z_i. Second, we control all the other entries, using a communication complexity argument.

First, we control the worst-case entries. An inspection of the proof of Proposition 14 shows the algorithm can recover against a constant fraction of worst case errors, in the following sense: For each hash block of size b, if only a constant fraction αb of the measurements are ruined, the algorithm will still work (after adjusting constants). Similarly, in Algorithm 3, the hash matrices are organized into blocks of size $t = O(\log(N/k))$. If only a constant fraction βt of the hashes in any block are ruined, then (again after adjusting constants) the algorithm will still work. We will show that the number of the measurements that Mallory might ruin in a worst case way fall into these categories, and summarily ignore them. Let A_i be the set of indices j so that Mallory outputs z_j while looking at the i^{th} row of Φ. There are at most $\alpha\log(N/k)$ rows i which have $|A_i| > N/(\alpha\log(N/k))$. Since each block has $t \geq O(\log(N/k))$ rows, at most βt of the rows in any one block are ruined this way—we will give up on these rows. Fix one of the $O(\log N \log k)$ hash matrices which make up Φ, and suppose it has b buckets. A series of Chernoff bounds shows that of the rows with $|A_i| < N/(\alpha\log(N/k))$, then at least αb of the rows i in any hash matrix with b buckets have only a constant number of nonzero entries in A_i, with probability at least $O\left(2^{-k}\right)$. A union bound over the $O(\log N \log k)$ hash matrices shows that at most αb rows in each hash matrix have any entries that Mallory can control in a worst-case way. We also give up on these rows.

Next, we control the rest of the entries. Let S_j denote Mallory's state when she outputs z_j, and suppose she is on row i when this occurs. Let s_r^j be the sign associated with z_j in the r^{th} hash matrix, and let $s^j = (s_1^j, \ldots, s_B^j)$. We will show that the distribution on s_j, conditional on S_j, is close to uniform.

Choose constants δ_1 and δ_2 so that $(1 - \delta_1)(1/2 + \delta_2) = 1 - \delta$ for δ small enough for Theorem 16 to apply. Consider the game of $\textsc{Ind}_{2^{i-1}}$ where Alice gets $s^1, s^2, \ldots, s^N$. Alice's strategy will be to send Bob $S_1, S_2, \ldots, S_m$, where S_i is Mallory's state when she is on row i. As long as $k = o(N/(\log(N)\log(N/k))$, the space this takes is $O(k\log N) = o(N)$. Suppose that with probability at least $1 - \delta_2$ over Φ, $\Pr(\Phi|(S_1, \ldots, S_m)) > \Pr(\Phi) + \delta_1$. If this is the case, then Bob, who gets $S_1, \ldots, S_m$, i, and $z \in \{-1, 1\}^N$, can determine the signs in the i^{th} column with advantage δ_1, and thus can succeed with probability $(1 - \delta_2)(1/2 + \delta_1)$, a contradiction of Theorem 16. Thus, with probability at least δ_2 over Φ, $\Pr(\Phi(S_1, \ldots, S_m)) < \Pr(\Phi) + \delta_1$. Let A be the set of all matrices Φ that are bad for x, given the worst case errors handled in the previous part. Since Algorithm 3 succeeds with probability $1 - o(1)$ against an oblivious adversary with a constant fraction of worst-case errors, the probability of A is $o(1)$. Thus, in the favorable case (which happens with probability at least δ_2),

$$\Pr(\Phi \in A \mid S_1, \ldots, S_m) < \Pr(\Phi \in A) + \delta_1 = o(1) + \delta_1.$$

Since the output of Mallory depends only on the states, the probability that the algorithm fails against Mallory in this favorable case is also $o(1) + \delta_1$, so the probability that Mallory succeeds is

at most

$$(1 - \delta_2) + \delta_2(o(1) + \delta_1) = 1 - \delta_3$$

for some constant δ_3.

By repeating the algorithm $O(1/\delta_3)$ times and testing the answers, as discussed above, the probability of failure can be made $> 2/3$.

□

5. Conclusions and Future Work

We present several new models for signal generation in combinatorial group testing and sparse recovery. These models capture the many natural situations in which the signal has a weak dependence on the measurement matrix, or when subsequent signals one observes have some weak dependence on the measurement matrix or upon the measurements obtained. It is often more natural to assume that the process generating this signal is either computationally bounded, or has limited information about the measurement matrix Φ than to assume that this process conforms to particular geometric requirements or follows a certain distribution. We show that there are significant gains (in the minimum number of measurements required) to be had by relaxing the model from adversarial to computationally or information-theoretically bounded, and not too much (in some cases, nothing at all) is lost by assuming these models over oblivious or statistical models. We also show that in the group testing case, there is a difference between information-theoretically bounded and computationally bounded (streaming) models, which contrasts the situation in sparse recovery.

One model we have not discussed is that of a polynomial-time bounded adversary, with cryptographic assumptions, which is a natural next step. It is perhaps of more practical use to consider sparse recovery or group testing against such an adversary.

Acknowledgements

The authors thank Hung Ngo for many helpful discussions.

References

[1] D Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 66(4):671–687, 2003.

[2] K Do Ba, P Indyk, E Price, and D.P Woodruff. Lower bounds for sparse recovery. *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1190–1197, 2010.

[3] R Baraniuk, V Cevher, M Duarte, and C Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, Jan 2008.

[4] R Baraniuk and P Steeghs. Compressive radar imaging. *IEEE Radar Conference 2007*, pages 128—133, Jan 2007.

[5] T. Berger, N. Mehravari, D. Towsley, and J. Wolf. Random multiple-access communications and group testing. *IEEE Trans. Commun.*, 32(7):769–779, 1984.

[6] G. Caire, S. Shamai, and S. Verd'u. Noiseless data compression with low-density parity-check codes. In *Advances in Network Information Theory (DIMACS '04)*, pages 263–284, 2004.

[7] Robert Calderbank, Stephen Howard, and Sina Jafarpour. A sublinear algorithm for sparse reconstruction with l2/l2 recovery guarantees. *arXiv*, cs.IT, Jun 2008.

[8] Candes, Romberg, and Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2), 2006.

[9] E Candes, M Rudelson, T Tao, and R Vershynin. Error correction via linear programming. 2005.

[10] V. Cevher, M.F. Duarte, C. Hegde, and R.G. Baraniuk. Sparse signal recovery using markov random fields. In *Proc. Workshop on Neural Info. Proc. Sys.(NIPS)*. Citeseer, 2008.

[11] Shiva Chaudhuri and Jaikumar Radhakrishnan. Deterministic restrictions in circuit complex- ity. pages 30–36, 1996.

[12] A Cohen, W Dahmen, and R DeVore. Compressed sensing and best k-term approximation. *American Mathematical Society*, 22(1):211–231, 2009.

[13] Graham Cormode and S. Muthukrishnan. What's hot and what's not: tracking most frequent items dynamically. *ACM Trans. Database Syst.*, 30(1):249–278, 2005.

[14] D Donoho. Compressed sensing. *Information Theory*, 52(4):1289–1306, Jan 2006.

[15] R. Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943.

[16] Ding-Zhu Du and Frank K. Hwang. *Combinatorial group testing and its applications*, volume 12 of *Series on Applied Mathematics*. World Scientific Publishing Co. Inc., River Edge, NJ, second edition, 2000.

[17] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 2008.

[18] A. G. Dýachkov and V. V. Rykov. Bounds on the length of disjunctive codes. *Problemy Peredachi Informatsii*, 18(3):7–13, 1982.

[19] A. G. Dýachkov, V. V. Rykov, and A. M. Rashad. Superimposed distance codes. *Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform.*, 18(4):237–250, 1989.

[20] Yaniv Erlich, Kenneth Chang, Assaf Gordon, Roy Ronen, Oron Navon, Michelle Rooks, and Gregory J. Hannon. Dna sudoku—harnessing high-throughput sequencing for multiplexed specimen analysis. *Genome Research*, 19:1243—1253, 2009.

[21] William Feller. *An Introduction to Probability Theory and Its Applications*. Wiley, 3 edition, 1967.

[22] Zoltán Füredi. On r-cover-free families. *J. Comb. Theory, Ser. A*, 73(1):172–173, 1996.

[23] A.C Gilbert, Y Li, E Porat, and M.J Strauss. Approximate sparse recovery: Optimizing time and measurements. *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 475–484, 2010.

[24] Michael T. Goodrich, Mikhail J. Atallah, and Roberto Tamassia. Indexing information for data forensics. In *Third International Conference on Applied Cryptography and Network Security (ANCS)*, pages 206–221, 2005.

[25] V Guruswami and A Smith. Codes for computationally simple channels: Explicit constructions with optimal rate. *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 723–732, 2010.

[26] TS Jayram, R. Kumar, and D. Sivakumar. The one-way communication complexity of hamming distance. *Theory OF Computing*, 4:129–135, 2008.

[27] Kainkaryam. Pooling in high-throughput drug screening. *Current Opinion in Drug Discovery & Development*, 12(3):339–350, May 2009.

[28] Raghunandan Kainkaryam and Peter Woolf. poolhits: A shifted transversal design based pooling strategy for high-throughput drug screening. *BMC Bioinformatics*, 9(1), 2008.

[29] Sherif M. Khattab, Sameh Gobriel, Rami G. Melhem, and Daniel Mossé. Live baiting for service-level dos attackers. In *INFOCOM*, pages 171–175, 2008.

[30] Sami Kirolos, Jason Laska, Michael Wakin, Marco Duarte, Dror Baron, Tamer Ragheb, Yehia Massoud, and Richard Baraniuk. Analog-to-information conversion via random demodulation. In *IEEE Dallas Circuits and Systems Workshop (DCAS)*, Dallas, Texas, Oct. 2006.

[31] E. Knill. Lower bounds for identifying subset members with subset queries. In *Proceedings of the sixth annual ACM-SIAM symposium on Discrete algorithms*, SODA '95, pages 369–377, Philadelphia, PA, USA, 1995. Society for Industrial and Applied Mathematics.

[32] I. Kremer, N. Nisan, and D. Ron. On randomized one-round communication complexity. *Computational Complexity*, 8(1):21–49, 1999.

[33] Amos Lapidoth and Prakash Narayan. Reliable communication under channel uncertainty. *IEEE Transactions on Information Theory*, 44:2148–2177, 1998.

[34] Jason Laska, Sami Kirolos, Yehia Massoud, Richard Baraniuk, Anna Gilbert, Mark Iwen, and Martin Strauss. Random sampling for analog-to-information conversion of wideband signals. In *IEEE Dallas Circuits and Systems Workshop (DCAS)*, Dallas, Texas, Oct. 2006.

[35] Richard J. Lipton. A new approach to information theory. In *STACS '94: Proceedings of the 11th Annual Symposium on Theoretical Aspects of Computer Science*, pages 699–708, London, UK, 1994. Springer-Verlag.

[36] Marc Mézard and Cristina Toninelli. Group testing with random pools: optimal two-stage algorithms. *CoRR*, abs/0706.3104, 2007.

[37] Silvio Micali, Chris Peikert, Madhu Sudan, and David A. Wilson. Optimal error correction against computationally bounded noise. In *TCC*, pages 1–16, 2005.

[38] D Needell and J.A Tropp. Cosamp: iterative signal recovery from incomplete and inaccurate samples. *Communications of the ACM*, 53(12):93–100, 2010.

[39] D Needell and R Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):310–316, 2010.

[40] Hung Q. Ngo and Ding-Zhu Du. A survey on combinatorial group testing algorithms with applications to DNA library screening. In *Discrete mathematical problems with medical applications (New Brunswick, NJ, 1999)*, volume 55 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 171–182. Amer. Math. Soc., Providence, RI, 2000.

[41] Hung Q. Ngo, Ely Porat, and Atri Rudra. Efficiently decodable error-correcting list disjunct matrices and applications. In *Proceedings of the 38th International Colloquium on Automata, Languages and Programming (ICALP)*, 2011. To appear.

[42] E Porat and M J Strauss. Sublinear time, measurement-optimal, sparse recovery for all. *submitted to SODA 2012*, 2011.

[43] Ely Porat and Amir Rothschild. Explicit non-adaptive combinatorial group testing schemes. In *ICALP '08*, volume 5125, pages 748–759, 2008.

[44] Jeanette P. Schmidt, Alan Siegel, and Aravind Srinivasan. Chernoff-Hoeffding bounds for applications with limited independence. *SIAM J. Discrete Math.*, 8(2):223–250, 1995.

[45] Milton Sobel and Phyllis A. Groll. Binomial group-testing with an unknown proportion of defectives. *Technometrics*, 8(4):pp. 631–656, 1966.

[46] Dharmpal Takhar, Jason Laska, Michael B. Wakin, Marco F. Duarte, Dror Baron, Shriram Sarvotham, Kevin Kelly, and Richard G. Baraniuk. A new compressive imaging camera architecture using optical-domain compression. In *Proc. IS&T/SPIE Symposium on Electronic Imaging*, 2006.

[47] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Trans. Signal Proc.*, 50(6), June 2002.

[48] Michael Wakin, Jason Laska, Marco Duarte, Dror Baron, Shriram Sarvotham, Dharmpal Takhar, Kevin Kelly, and Richard Baraniuk. Compressive imaging for video representation and coding. In *Proc. Picture Coding Symposium 2006*, Beijing, China, Apr. 2006.

[49] J. K. Wolf. Born again group testing: multiaccess communications. *IEEE Transaction on Information Theory*, IT-31:185–191, 1985.

[50] Xiaofeng Xin, Jean-François F. Rual, Tomoko Hirozane-Kishikawa, David E. Hill, Marc Vidal, Charles Boone, and Nicolas Thierry-Mieg. Shifted transversal design smart-pooling for high coverage interactome mapping. *Genome research*, 19(7):1262–1269, July 2009.

[51] Y.H. Zheng, D. J. Brady, M. E. Sullivan, and B. D. Guenther. Fiber-optic localization by geometric space coding with a two-dimensional gray code. *Applied Optics*, 44(20):4306–4314, 2005.