Optimal Group Testing using Left and Right regular sparse-graph codes

Avinash Vem, Nagaraj T. Janakiraman, Krishna R. Narayanan
Department of Electrical and Computer Engineering
Texas A&M University
{vemavinash,tjnagaraj,krn}@tamu.edu

***Abstract*—To be added.**

I. Introduction

The problem of Group Testing refers to testing a large population for sick/defective individual people when the fraction of sick people is known to be small. This problem was first introduced to the literature of statistics by Dorfman [1] during World War II for testing the soldiers for syphilis without having to test each soldier individually. Since then many schemes and algorithms were designed for this problem.

In [2] Lee, Pedarsani, Ramchandran applied the sparse-graph codes and a simple peeling decoder, which are popular tools in the error control coding community, to the non-adaptive group testing problem. [3], [4], [2]

Note that the computational complexity is order optimal for both the noiseless and noisy settings as mentioned in [2]. Regarding the optimality of the number of tests for the noiseless setting where both K and N scale satisfying $K = o(N)$, it was shown [3] that the number of tests need to be atleast as large as $CK\log(\frac{N}{K})$ for some constant C such that the probability of error approaches zero. As far as we are aware this is the tightest lower bound. In the same work it is shown that $CK\log N$ is the sufficient number of tests. In our work we show that in fact $C(\epsilon)K\log(\frac{N}{K})$ tests is sufficient to recover $(1-\epsilon)$ fraction of the defective items with high probability. More survey needs to be done regarding the lower and upper bounds for the number of tests in noiseless and noisy settings especially under different performance evaluation criteria. For e.g., in [3] the upper bound(achievable) on the minimal number of tests $O(K\log N)$ is when the performance metric considered is the average probability of error that the decoded support set is not exactly equal to the original support set. But for the framework where ϵ-fraction of the defective items are allowed to be missed, only the lower bound on the number of tests required is given.

II. Problem Statement

Formally the group testing problem can be stated as following. Given a total number of N items out of which K are defective, the objective is to perform m different tests and identify the location of the K defective items from the test outputs. For now we consider only the noiseless group testing problem i.e., the result of each test is exactly equal to the boolean OR of all the items participating in the test.

Let the support vector $\mathbf{x} \in \{0,1\}^N$ denote the list of items where the indices with non-zero values correspond to the defective items. A non-adaptive testing scheme consisting of m tests can be represented by a matrix $\mathbf{A} \in \{0,1\}^{m\times N}$ where each row $\mathbf{a}_i$ corresponds to a test. The non-zero indices in row $\mathbf{a}_i$ correspond to the items that participate in i^{th} test. The output corresponding to vector $\mathbf{x}$ and the testing scheme $\mathbf{A}$ and can be expressed in matrix form as:

$$\mathbf{y} = \mathbf{A} \odot \mathbf{x}$$

where $\odot$ is the usual matrix multiplication in which the arithmetic multiplications are replaced by the boolean AND operation and the arithmetic additions are replaced by the boolean OR operation.

III. Review: Prior Work

In [2] Lee, Pedarsani and Ramchandran introduced a framework, referred to as SAFFRON, based on left-regular sparse graph codes for non-adaptive group testing problem. We will briefly review their SAFFRON testing scheme, decoding scheme (reconstruction of $\mathbf{x}$ given $\mathbf{y}$) and their main results in this section. The SAFFRON testing scheme consists of two stages: the first stage is based on a left-regular sparse graph code which pools the N items into non-disjoint M_1 bins where each item belongs to exactly l bins. The second stage comprises of producing h testing outputs at each bin where the h different combinations of the pooled items (from the first stage) at the respective bin are defined according to a universal signature matrix. For the first stage the authors consider a bipartite graph with N variable nodes (corresponding to the N items) and M_1 bin nodes. Each variable node is connected to l bin nodes chosen uniformly at random from the M_1 available bin nodes. All the variable nodes (historically depicted on the left side of the graph in coding theory) have a degree l, hence the left-regular, whereas the degree of a bin node on the right is a random variable ranging from $[1:n]$.

Definition 1 (Left-regular sparse graph ensemble)**.** We define $\mathcal{G}_\ell(N, M_1)$ to be the ensemble of left-regular bipartite graphs where, for each variable node, the l right node connections are chosen uniformly at random from the M_1 right nodes.

Let $\mathbf{T}_G \in \{0,1\}^{M_1 \times N}$ be the adjacency matrix corresponding to a graph $G \in \mathcal{G}_\ell(N, M_1)$ i.e., each column in $\mathbf{T}_G$ corresponds to a variable node and has exactly l ones. Let the rows in matrix $\mathbf{T}_G$ be given by $\mathbf{T}_G = [\mathbf{t}_1^T, \mathbf{t}_2^T, \ldots, \mathbf{t}_{M_1}^T]^T$. For the second stage let the universal signature matrix defining the h tests at each bin be $\mathbf{U} \in \{0,1\}^{h \times N}$. Then the overall testing matrix $\mathbf{A} := [\mathbf{A}_1^T, \ldots, \mathbf{A}_{M_1}^T]^T$ where $\mathbf{A}_i = \mathbf{U}\,\mathrm{diag}(\mathbf{t}_i)$ of size $h \times N$ defines the h tests at i^{th} bin. Thus the total number of tests is $M = M_1 \times h$.

The signature matrix $\mathbf{U}$ in a more general setting with parameters r and p can be given by

$$\mathbf{U}_{r,p} = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \cdots & \mathbf{b}_r \\ \overline{\mathbf{b}}_1 & \overline{\mathbf{b}}_2 & \cdots & \overline{\mathbf{b}}_r \\ \mathbf{b}_{\pi_1^1} & \mathbf{b}_{\pi_2^1} & \cdots & \mathbf{b}_{\pi_r^1} \\ \overline{\mathbf{b}}_{\pi_1^1} & \overline{\mathbf{b}}_{\pi_2^1} & \cdots & \overline{\mathbf{b}}_{\pi_r^1} \\ \cdots & & \vdots & \\ \mathbf{b}_{\pi_1^p} & \mathbf{b}_{\pi_2^p} & \cdots & \mathbf{b}_{\pi_r^p} \\ \overline{\mathbf{b}}_{\pi_1^p} & \overline{\mathbf{b}}_{\pi_2^p} & \cdots & \overline{\mathbf{b}}_{\pi_r^p} \end{bmatrix} \tag{1}$$

where $\mathbf{b}_i \in \{0,1\}^{\lceil \log_2 r \rceil}$ is the binary expansion vector for i and $\overline{\mathbf{b}}_i$ is the complement of $\mathbf{b}_i$. $\pi^k = [\pi_1^k, \pi_2^k, \ldots, \pi_r^k]$ denotes a permutation chosen at random from symmetric group S_r. $\mathbf{U}_{r,p}$ will be referred to either as the ensemble of matrices generated over the choices of the permutations π^k for $k \in [1:p]$ or as a matrix picked uniformly at the random from the said ensemble. The reference should be sufficiently clear from the context. In the SAFFRON scheme the authors employed a signature matrix that is equivalent to $\mathbf{U}_{r,p}$ with $r = N$ and $p = 2$ thus resulting in a $\mathbf{U}$ of size $h \times N$ with $h = 6 \log_2 N$.

Decoding

Before describing the decoding process let us review some terminology. A bin is referred to as a *singleton* if there is exactly one non-zero variable node connected to the bin and similarly referred to as a *double-ton* in case of two non-zero variable nodes. In the case where we know the identity of one of them leaving the decoder to decode the identity of the other one, the bin is referred to as a *resolvable double-ton*. And if the bin has more than two non-zero variable nodes attached we refer to it as a *multi-ton*. First part of the decoder which is referred to as bin decoder will be able to detect and decode exactly the identity of the non-zero variable nodes connected to the bin if and only if the bin is a singleton or a resolvable double-ton. If the bin is a multi-ton the bin decoder will detect it as a multi-ton, i.e., the bin decoder output is not a singleton or a resolvable double-ton. The second part of the decoder which is commonly referred to as peeling decoder [5], when given the identities of some of the non-zero variable nodes by the bin decoder, identifies the bins connected to the recovered variable nodes and looks for newly uncovered resolvable double-ton in these bins. This process of recovering new non-zero variable nodes from already discovered non-zero variable nodes proceeds in an iterative manner (referred to as peeling off from the graph historically). For details of the decoder we refer the reader to [2].

The overall group testing decoder comprises of these two decoders working in conjunction as follows. In the first and foremost step, given the M tests output, the bin decoder is applied on the M_1 bins and the set of singletons i.e., the set of decoded non-zero variable nodes denoted as $\mathcal{D}$ is output. Now in an iterative manner, at each iteration, a variable node from $\mathcal{D}$ is considered and the bin decoder is applied on the bins connected to this variable node again but now with the knowledge of some recovered variable nodes. The idea is that hopefully one of these bins is detected as a resolvable double-ton thus resulting in decoding one more non-zero variable node. The considered variable node in the previous iteration is moved from $\mathcal{D}$ to a set of peeled off variable nodes $\mathcal{P}$ and the newly decoded non-zero variable node in the previous iteration, if any, will be placed in $\mathcal{D}$ and continue to the next iteration. The decoder is terminated when $\mathcal{D}$ is empty and is declared successful if the set $\mathcal{P}$ equals the set of defective items.

Remark 2. Note that we are not literally peeling off the decoded nodes from the graph because of the *non-linear* OR operation on the non-zero variable nodes at each bin thus preventing us in subtracting the effect of the non-zero node from the measurements of the bin node unlike in the problems of compressed sensing or LDPC codes on binary erasure channel.

Now we state the series of lemmas and theorems, without proofs, from [2] that enabled the authors Lee, Pedarsani and Ramchandran to show that this left-regular sparse-graph code based SAFFRON scheme with the described peeling decoder solves the group testing problem with $\Omega(K \log N)$ tests and $O(K \log N)$ computational complexity.

Lemma 3 (Bin decoder analysis)**.** For a signature matrix $\mathbf{U}_{r,p}$ as described in (1), the bin decoder successfully detects and resolves if the bin is either a singleton or a resolvable double-ton. In the case of the bin being a multi-ton, the bin decoder declares a wrong hypothesis of either a singleton or a resolvable double-ton with a probability no greater than $\frac{1}{r^p}$.

For the purpose of analysis, the error probability performance of the peeling decoder is analyzed independently of the bin decoder i.e., a peeling decoder is considered which assumes that the bin decoder is working accurately which will be referred to as *oracle based peeling decoder*. Another simplification considered is that the oracle based peeling decoder decodes a variable node if it is connected to a bin-node with degree one or a bin-node with degree two with the other variable node being already decoded, in an iterative fashion. Any right node with more than degree two is untouched by this oracle based peeling decoder. To simplify further, a pruned graph is considered where all the zero variable nodes and their respective edges are removed from the graph. It is easy to

ϵ	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}
$c_1(\epsilon)$	6.13	7.88	9.63	11.36	13.10	14.84	16.57
ℓ	7	9	10	12	14	15	17

TABLE I
CONSTANTS FOR VARIOUS ERROR FLOOR VALUES

verify that the original decoder with accurate bin decoding is equivalent to this simplified oracle based peeling decoder on a pruned graph.

Definition 4 (Pruned graph ensemble)**.** We will define the pruned graph ensemble $\tilde{\mathcal{G}}_l(N, K, M_1)$ as the set of all bipartite graphs obtained from removing a random $N-K$ subset of variable nodes from a graph from the ensemble $\mathcal{G}_\ell(N, M_1)$. Note that graphs from the pruned ensemble have K variable nodes.

Before we analyze the pruned graph ensemble let us define the right-node degree distribution (d.d) of an ensemble as $R(x) = \sum_i R_i x^i$ where R_i is the probability that a right-node has degree i. Similarly the edge d.d $\rho(x) = \sum_i \rho_i x^{i-1}$ is defined where ρ_i is the probability that a random edge in the graph is connected to a right-node of degree i. Note that the left-degree distribution is regular even for the pruned graph ensemble and hence is not specifically mentioned.

Lemma 5 (Edge d.d of Pruned graph)**.** For the pruned ensemble $\tilde{\mathcal{G}}_\ell(N, K, M_1)$, it can be shown in the limit $K, N \to \infty$ that $\rho_1 = e^{-\lambda}$ and $\rho_2 = \lambda e^{-\lambda}$ where $\lambda = l/c_1$ if $M_1 = c_1 K$ for some constant c_1.

Lemma 6. For the pruned graph ensemble $\tilde{\mathcal{G}}_\ell(N, K, M_1)$ the oracle-based peeling decoder fails to peel off atleast $(1-\epsilon)$ fraction of the variable nodes with exponentially decaying probability if $M_1 \geq c_1(\epsilon)K$ where $c_1(\epsilon)$ for various ϵ is given in Table. I.

Proof. Instead of reworking the whole proof here from [2], we will list the main steps involved in the proof which we will use further along. If we let p_j be the probability that a random defective item is not identified at iteration j, in the limit N and $K \to \infty$ we can write density evolution (DE) equation relating p_{j+1} to p_j as

$$p_{j+1} = \left[1 - (\rho_1 + \rho_2(1-p_j))\right]^{l-1}.$$

For this DE, we can see that 0 is not a fixed point and hence $p_j \nrightarrow 0$ as $j \to \infty$. Therefore numerically optimizing the values of c_1 and l such that $\lim_{j\to\infty} p_j \leq \epsilon$ gives the optimal values for $c_1(\epsilon)$ and l given in Table. I. $\square$

Combining the lemmas and remarks above, the main result from [2] can be summarized as follows.

Theorem 7. For any arbitrarily-small $\epsilon > 0$ the SAFFRON framework, performing $m = 6c_1(\epsilon)K \log_2 N$tests, recovers atleast a $(1-\epsilon)$ fraction of the defective items with a high probability of atleast $1 - O(\frac{K}{N^2})$. And the computational complexity of the decoding scheme is $O(K \log N)$. The constant $c_1(\epsilon)$ is given in Table. I for some values of ϵ.

IV. PROPOSED SCHEME

The main difference between the SAFFRON scheme and our proposed scheme is that we use a left-and-right-regular sparse-graph in the first stage for the binning operation.

Definition 8 (Left-and-right-regular sparse graph ensemble)**.** We define $\mathcal{G}_{\ell,r}(N, M_1)$ to be the ensemble of left-and-right-regular graphs where the $N\ell$ edge connections from the left and $M_1 r (= N\ell)$ edge connections from the right are paired up according to a permutation $\pi_{N\ell}$ chosen at random.

Let $\mathbf{T}_G \in \{0,1\}^{M_1 \times N}$ be the adjacency matrix corresponding to a graph $G \in \mathcal{G}_{l,r}(N, M_1)$ i.e., each column in $\mathbf{T}_G$ corresponding to a variable node has exactly l ones and each row corresponding to a bin node has exactly r ones. And let the universal signature matrix be $\mathbf{U} \in \{0,1\}^{h \times r}$ chosen from the $\mathbf{U}_{r,p}$ ensemble. Then the overall testing matrix $\mathbf{A} := [\mathbf{A}_1^T, \ldots, \mathbf{A}_{M_1}^T]^T$ where $\mathbf{A}_i \in \{0,1\}^{h \times N}$ defining the h tests at i^{th} bin is given by

$$\begin{aligned} \mathbf{A}_i &= [\mathbf{0}, \ldots, \mathbf{0}, \mathbf{u}_1, \mathbf{0}, \ldots, \mathbf{u}_2, \mathbf{0}, \ldots, \mathbf{u}_r], \quad \text{where} \\ \mathbf{t}_i &= [0, \ldots, 0,\ 1,\ 0, \ldots,\ 1,\ 0, \ldots,\ 1]. \end{aligned} \tag{2}$$

Note that $\mathbf{A}_i$ is defined by placing the r columns of $\mathbf{U}$ at the r non-zero indices of $\mathbf{t}_i$ and the remaining are padded with zero columns. We can observe that the total number of tests for this scheme is $M = M_1 \times h$ where $h = (2p+2)\log r$.

Definition 9 (Regular SAFFRON)**.** We define the ensemble of testing matrices for our scheme to be $\mathcal{G}_{\ell,r}(N, M_1) \times \mathbf{U}_{r,p}$ where a graph G is chosen from $\mathcal{G}_{\ell,r}(N, M_1)$, a signature matrix $\mathbf{U}$ is chosen from $\mathbf{U}_{r,p}$ and the testing matrix is defined according to Eq. (2). Note that the total number of tests for this testing scheme is $(2p+2)M_1 \log r$ where $r = \frac{N\ell}{M_1}$.

For the regular SAFFRON testing ensemble defined in Def. 9, we employ the same peeling based decoder described in Sec. III.

Now we consider the performance analysis of the regular SAFFRON scheme under the peeling based decoder. Similar to the SAFFRON scheme we will analyze the peeling decoder and the bin decoder separately and union bound the total error probability of the decoding scheme. As we have already mentioned the analysis of the peeling decoder part alone can be carried out by considering a *simplified oracle-based peeling decoder* on a pruned graph with only the non-zero variable nodes remaining.

Definition 10 (Pruned graph ensemble)**.** We will define the pruned graph ensemble $\tilde{\mathcal{G}}_{\ell,r}(N, K, M_1)$ as the set of all graphs obtained from removing a random $N-K$ subset of variable nodes from a graph from the ensemble $\mathcal{G}_{\ell,r}(N, M_1)$.

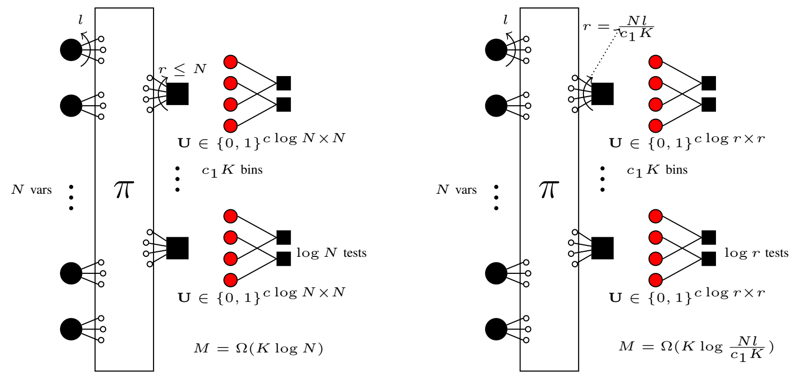

Fig. 1. Illustration of the main differences between SAFFRON [2] on the left and our regular-SAFFRON scheme on the right. In both the schemes the peeling decoder on sparse graph requires $\Omega(K)$ bins. But for the bin decoder part, in SAFFRON scheme the right degree is a random variable with a maximum value of N and thus requires $\Omega(\log N)$ tests at each bin. Whereas our scheme based on right-regular sparse graph has a constant right degree of $\Omega(\frac{N}{K})$ and thus requires only $\Omega(\log \frac{N}{K})$ tests at each bin. Thus we can improve the number of tests from $\Omega(K \log N)$ to order optimal $\Omega(K \log \frac{N}{K})$.

Note that graphs from the pruned ensemble have K variable nodes with a degree ℓ whereas the right degree is not regular anymore.

Lemma 11 (Edge d.d of pruned graph)**.** For the pruned graph ensemble $\tilde{\mathcal{G}}_{\ell,r}(N, K, M_1)$ it can be shown in the limit $K, N \to \infty$ that edge d.d coefficients $\rho_1 = e^{-\lambda}$ and $\rho_2 = \lambda e^{-\lambda}$ where $\lambda = \ell / c_1$ for the choice of $M_1 = c_1 K$, c_1 being some constant.

Note that even if our initial ensemble is left-and-right-regular the pruned graph has asymptotically same degree distribution as in the SAFFRON scheme where the initial graph is from left-regular ensemble.

Lemma 12. For the pruned graph ensemble $\tilde{\mathcal{G}}_{\ell,r}(N, K, M_1)$ the oracle-based peeling decoder fails to peel off atleast $(1-\epsilon)$ fraction of the variable nodes with exponentially decaying probability for $M_1 = c_1(\epsilon)K$ where $c_1(\epsilon)$ for various ϵ is given in Table. I.

Proof. From Lemma. 11 we know that the edge degree distribution coefficients ρ_1 and ρ_2 are identical to that of the SAFFRON scheme and hence the same DE equations can be used here. Therefore the exact same proof as the proof of Lemma. 6 can be employed here. □

Theorem 13. Let $p \in \mathbb{Z}$ such that K and N scale as $K \in o(N^{\frac{p}{p+1}})$. For $M = (2p+2)c_1(\epsilon)K \log_2 \frac{N}{K}$, the regular SAFFRON framework we proposed, asymptotically, recovers atleast a $(1-\epsilon)$ fraction of the defective items for arbitrarily-small ϵ with high probability $1 - O\left(\frac{K^{p+1}}{N^p}\right)$. Note that computational complexity of the decoding scheme is $O(K \log \frac{N}{K})$. The constant $c_1(\epsilon)$ is given in Table. I.

Proof. It remains to be shown that the total probability of error decays asymptotically in K and N. Let E_1 be the event of oracle-based peeling decoder terminating without recovering atleast $(1-\epsilon)K$ variable nodes. Let E_2 be the event of the bin decoder making an error during the entirety of the peeling process and E_{bin} be the event of one instance of bin decoder making an error. The total probability of error P_e can be upper bounded by

$$
\begin{aligned}
P_e &\leq \Pr(E_1) + \Pr(E_2) \\
&\leq \Pr(E_1) + K\ell \Pr(E_{\text{bin}}) \\
&\in O\left(\frac{K^{p+1}}{N^p}\right)
\end{aligned}
$$

where the second inequality is due to the union bound over a maximum of Kl instances of bin decoding. The third line is due to the fact that $\Pr(E_1)$ is exponentially decaying in K (see Lemma. 12) and $\Pr(E_{\text{bin}}) = (\frac{c_1 K}{N\ell})^p$ (see Lemma. 3 and Def. 9) □

Proof of Lem. 11. We will first derive $R(x)$ for the pruned graph ensemble and then use the relation[6] $\rho(x) = \frac{R'(x)}{R'(1)}$ to derive the edge d.d . Note that all the check nodes have a uniform degree r before pruning. When pruning we are removing a $N-K$ subset of variable nodes at random i.e., asymptotically this is equivalent to removing each edge from the graph with a probability $1-\epsilon$ where $\epsilon := \frac{K}{N}$. Under this process the right-node d.d can be written as

$$
\begin{aligned}
R_1 &= r\epsilon(1-\epsilon)^{r-1}, \quad \text{and similarly} \\
R_i &= \binom{r}{i} \epsilon^i (1-\epsilon)^{r-i},
\end{aligned}
$$

thus giving us $R(x) = (\epsilon x + (1-\epsilon))^r$. This gives us

$$
\begin{aligned}
\rho(x) &= \frac{r\epsilon(\epsilon x + (1-\epsilon))^{r-1}}{r\epsilon} \\
&= (\epsilon x + (1-\epsilon))^{r-1}.
\end{aligned}
$$

Thus we can compute that $\rho_1 = (1-\epsilon)^{r-1}$ and $\rho_2 = (r-1)\epsilon(1-\epsilon)^{r-2}$. We evaluate these quantities asymptotically as

$K, N \to \infty$ and $M_1 = CK$.

$$\lim_{K,N\to\infty} \rho_1 = \lim_{K,N\to\infty} (1 - \frac{K}{N})^{\frac{Nl}{CK}-1}$$
$$= e^{-\lambda} \qquad \text{where } \lambda = \frac{l}{C}$$

Similarly we can show $\lim_{K,N\to\infty} \rho_2 = \lambda e^{-\lambda}$. $\square$

V. Robust Group Testing

In this section we extend our scheme to the group testing problem where the test results can be noisy. To be formal, the signal model looks like

$$\mathbf{y} = \mathbf{A} \odot \mathbf{x} + \mathbf{w},$$

where the addition is over binary field and $\mathbf{w} \in \{0,1\}^N$ is an i.i.d noise vector distributed according to Bernoulli distribution with parameter $0 < q < \frac{1}{2}$.

Testing Scheme

In [2] for the robust group testing problem, the signature matrix used for noiseless group testing problem is modified using a error control coding such that it can handle singletons and resolvable doubletons in the presence of noise. The binning operation as defined by the bipartite graph is exactly identical to that of noiseless case. We describe the modifications to the signature matrix and the bin detection decoding scheme as given in [2] for the sake of completeness and then state the performance bounds for our scheme for the noisy group testing problem.

Let $\mathcal{C}_n$ be a binary error-correcting code with the following definition:

- Let the encoder and decoder functions be $f : \{0,1\}^n \to \{0,1\}^{\frac{n}{R}}$ and $g : \{0,1\}^{\frac{n}{R}} \to \{0,1\}^n$ respectively where R is the rate of the code.

We can use any error-correcting code but for ease of analysis and tight upper bound for the number of tests we will use spatially-coupled LDPC codes which are known to be capacity achieving [7], [8]. For spatially-coupled LDPC codes, being capacity achieving is equivalent to:

- There exists a sequence of codes $\{\mathcal{C}_n\}$ with the rate of each code being R satisfying

$$R < 1 - H(q) - \delta = 1 + q\log_2 q + \overline{q}\log_2 \overline{q} - \delta \quad (3)$$

for any arbitrary small constant δ such that the probability of error $\Pr(g(\mathbf{x}+\mathbf{w}) \neq \mathbf{x}) < 2^{-\kappa n}$ for some $\kappa > 0$. In Eqn. 3, $\overline{q} := 1 - q$.

The modified signature matrix $\mathbf{U}'_{r,p}$ can be described via $\mathbf{U}_{r,p}$ given in Eq. (1), and encoding function f for a code $\mathcal{C}_{\log_2 r}$ as follows:

$$\mathbf{U}'_{r,p} := \begin{bmatrix} \frac{f(\mathbf{b}_1)}{\overline{f(\mathbf{b}_1)}} & \frac{f(\mathbf{b}_2)}{\overline{f(\mathbf{b}_2)}} & \cdots & \frac{f(\mathbf{b}_r)}{\overline{f(\mathbf{b}_r)}} \\ \frac{f(\mathbf{b}_{\pi_1^1})}{\overline{f(\mathbf{b}_{\pi_1^1})}} & \frac{f(\mathbf{b}_{\pi_2^1})}{\overline{f(\mathbf{b}_{\pi_2^1})}} & \cdots & \frac{f(\mathbf{b}_{\pi_r^1})}{\overline{f(\mathbf{b}_{\pi_r^1})}} \\ \cdots & & \vdots & \\ \frac{f(\mathbf{b}_{\pi_1^p})}{\overline{f(\mathbf{b}_{\pi_1^p})}} & \frac{f(\mathbf{b}_{\pi_2^p})}{\overline{f(\mathbf{b}_{\pi_2^p})}} & \cdots & \frac{f(\mathbf{b}_{\pi_r^p})}{\overline{f(\mathbf{b}_{\pi_r^p})}} \end{bmatrix} \quad (4)$$

Then the overall testing matrix $\mathbf{A}$ is defined in identical fashion to the definition in Sec. III for the case of noiseless case except that $\mathbf{U}$ will be replaced by $\mathbf{U}'$ in Eqn. (4). Formally it can be defined as $\mathbf{A} := [\mathbf{A}_1^T, \ldots, \mathbf{A}_{M_1}^T]^T$ where $\mathbf{A}_i = \mathbf{U}' \operatorname{diag}(\mathbf{t}_i)$ where the binary vectors $\mathbf{t}_i$ are defined in Sec. III.

Decoding

The decoding scheme for the robust group testing, similar to the case of noiseless case, has two parts with the iterative peeling part of the decoder identical to that of the noiseless case whereas the bin detection part differs slightly with an extra step of decoding for the error control code involved.

Given the test output vector at a bin $\mathbf{y} = [\mathbf{y}_{01}^T, \mathbf{y}_{02}^T, \mathbf{y}_{11}^T, \ldots, \mathbf{y}_{p2}^T]^T$, the bin detection for the noisy case can be summarized as following: The decoder first applies the decoding function $g(\cdot)$ to the first segments in each section $\mathbf{y}_{i1} \forall i \in [0:p]$ then obtains the locations $l_0, l_1, \ldots l_p$ whose binary expansions are equal to the error-correcting decoder outputs $g(\mathbf{y}_{i1})$. The decoder declares the bin as a singleton if $\pi_{l_0}^i = l_i \ \forall i$.

Similarly given that one of the variable nodes connected to the bin is already decoded to be non-zero, the resolvable double-ton decoding can be summarized as following. Let the location of the already recovered variable node in the bin be l_0 and let it be originally a double-ton, then the test output can be given as

$$\begin{bmatrix} \mathbf{y}_{01} \\ \mathbf{y}_{02} \\ \mathbf{y}_{11} \\ \vdots \\ \mathbf{y}_{p2} \end{bmatrix} = \mathbf{u}_{l_0} \vee \mathbf{u}_{l_1} + \mathbf{w} = \begin{bmatrix} \frac{f(\mathbf{b}_{l_0})}{\overline{f(\mathbf{b}_{l_0})}} \\ f(\mathbf{b}_{\pi_{l_0}^1}) \\ \vdots \\ \overline{f(\mathbf{b}_{\pi_{l_0}^p})} \end{bmatrix} \vee \begin{bmatrix} \frac{f(\mathbf{b}_{l_1})}{\overline{f(\mathbf{b}_{l_1})}} \\ f(\mathbf{b}_{\pi_{l_1}^1}) \\ \vdots \\ \overline{f(\mathbf{b}_{\pi_{l_1}^p})} \end{bmatrix} + \mathbf{w}$$

where the location of the second non-zero variable node l_1 needs to be recovered. Given $\mathbf{y} = \mathbf{u}_{l_0} \vee \mathbf{u}_{l_1} + \mathbf{w}$ and $\mathbf{u}_{l_0}$, $\mathbf{u}_{l_1} + \mathbf{w}$ can be recovered since for each segment of $\mathbf{u}_{l_0}$ either the vector $f(\mathbf{b}_{\pi_{l_0}^k})$ or it's complement is available. Once $\mathbf{u}_{l_1} + \mathbf{w}$ is recovered, we singleton decoding procedure and rules as described above.

Lemma 14 (Robust Bin Decoder Analysis)**.** For a signature matrix $\mathbf{U}'_{r,p}$ as described in (4), the robust bin decoder misses

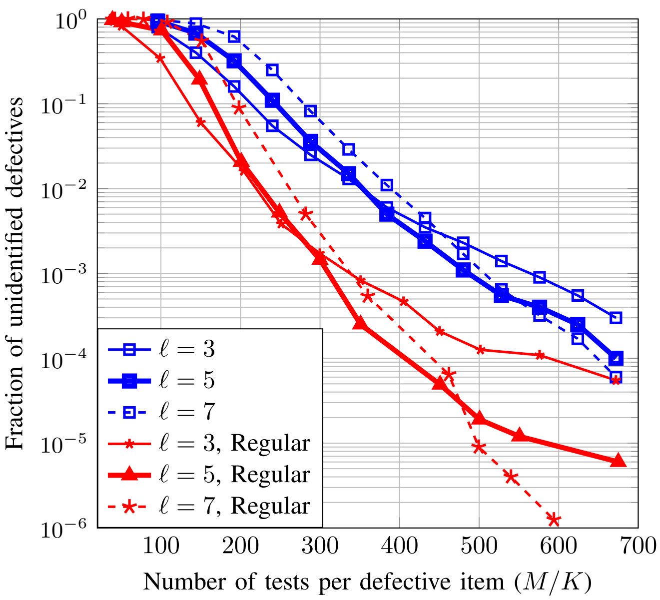

Fig. 2. MonteCarlo simulations for $K = 100, N = 2^{16}$. We compare the SAFFRON scheme with our regular SAFFRON scheme for various left degrees $\ell \in \{3, 5, 7\}$. For a given ℓ the bin detection size is fixed and we vary the number of bins. The plots in blue indicate the SAFFRON scheme and the plots in red indicate our regular SAFFRON scheme based on left-and-right-regular bipartite graphs.

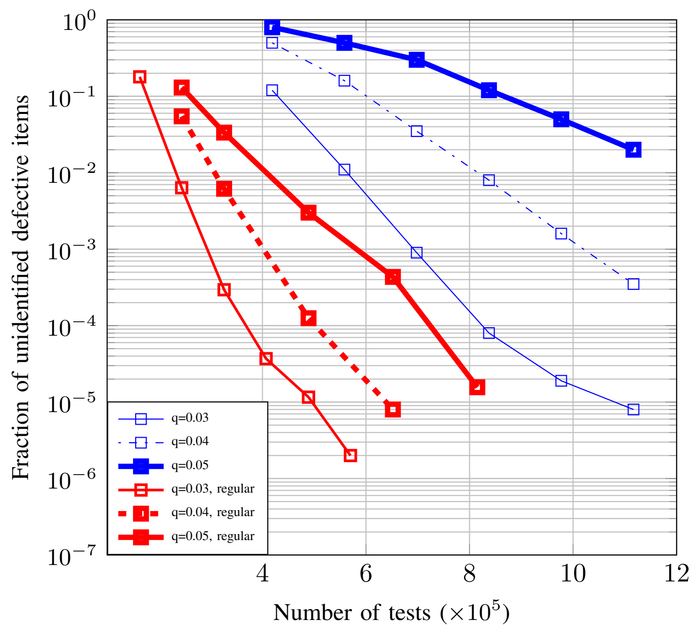

Fig. 3. MonteCarlo simulations for $K = 128, N = 2^{32}$. We compare the SAFFRON scheme with our regular SAFFRON scheme for a left degree $\ell = 12$. We fix the number of bins and vary the rate of the error control code used. The plots in blue indicate the SAFFRON scheme[2] and the plots in red indicate our regular SAFFRON scheme based on left-and-right-regular bipartite graphs.

a singleton with probability no greater than $\frac{3}{r^\kappa}$. The robust bin decoder wrongly declares a singleton with probability no greater than $\frac{1}{r^{p\kappa}}$.

The fraction of missed singeltons will be compensated by using $M(1 + \frac{3}{r^\kappa})$ instead of M such that the total number of singletons decoded will be $M(1 + \frac{3}{r^\kappa})(1 - \frac{3}{r^\kappa}) \approx M$.

Theorem 15. Let $p \in \mathbb{Z}$ such that K and N scale as $K \in o\left(N^{\frac{p\kappa}{p\kappa+1}}\right)$ where $\kappa > 0$ is defined in Eq. (3). For $M = 2(p+1)\beta(q)c_1(\epsilon)K \log_2 \frac{N}{K}$, the robust regular SAFFRON framework we proposed, asymptotically, recovers atleast a $(1-\epsilon)$ fraction of the defective items for arbitrarily-small ϵ with high probability $1 - O\left(\frac{K^{p\kappa+1}}{N^{p\kappa}}\right)$. Here $\beta(q) = \frac{1}{R} > \frac{1}{1-H(q)-\delta}$ for an arbitrary small constant δ and the constants $c_1(\epsilon)$ are given in Table. I. Note that computational complexity of the decoding scheme is $O(K \log \frac{N}{K})$.

VI. Simulation Results

In this section we will evaluate the performance of our proposed regular SAFFRON scheme via Monte Carlo simulations and compare it with the results for SAFFRON scheme provided in [2].

Noiseless Group Testing

As per Thm. 13 the regular SAFFRON scheme we proposed recovers $(1-\epsilon)$ fraction of defective items with a high probability for $M_1 > c_1 K$ where the pairs (c_1, ℓ) are given for various values of error floors ϵ in Table. I. We demonstrate this by simulating the performance for the system parameters summarized below.

- We fix $N = 2^{16}$ and $K = 100$
- In Eqn. 1 the parameter $p = 2$ is chosen i.e. $h = 6 \log_2 r$
- For $\ell \in \{3, 5, 7\}$ we vary the number of bins $M_1 = cK$.
- Hence the total number of tests $M = 6cK \log_2 \left(\frac{N\ell}{cK}\right)$

The results are shown in Fig. 2. We observe that there is clear improvement in performance for our regular SAFFRON scheme when compared to the SAFFRON scheme for each $\ell \in \{3, 5, 7\}$.

Noisy Group Testing

Similar to the noiseless group testing problem we simulate the performance of our regular SAFFRON scheme and compare it with that of the SAFFRON scheme. For convenience of comparison we choose our system parameters identical to the choices in [2]. The system parameters can be summarized as follows:

- $N = 2^{32}, K = 2^7$. We choose $\ell = 12, M_1 = 11.36K$
- Hence the total number of tests is $M = 6cK \log_2 \left(\frac{N\ell}{cK}\right)$
- We simulate for BSC noise parameter $q \in \{0.03, 0.04, 0.05\}$

For the signature matrix we choose $p = 1$ in Eqn. 1 i.e. $h = 4 \log_2 r$. Note that for the above set of parameters the right degree $r = \frac{N\ell}{M_1} \approx 26$. By choosing to operate in field $GF(2^7)$ gives us a message of length 4 symbols. Then we encode using a $(4+2e, 4)$ Reed-Solomon code for $e \in [0:8]$ thus giving us a column length of $4 \times 7(4+2e)$ bits.

References

[1] R. Dorfman, "The detection of defective members of large populations," *The Annals of Mathematical Statistics*, vol. 14, no. 4, pp. 436–440, 1943.

[2] K. Lee, R. Pedarsani, and K. Ramchandran, "Saffron: A fast, efficient, and robust framework for group testing based on sparse-graph codes," *arXiv preprint arXiv:1508.04485*, 2015.

[3] G. K. Atia and V. Saligrama, "Boolean compressed sensing and noisy group testing," *IEEE Transactions on Information Theory*, vol. 58, no. 3, pp. 1880–1901, 2012.

[4] A. Mazumdar, "Nonadaptive group testing with random set of defectives via constant-weight codes," *arXiv preprint arXiv:1503.03597*, 2015.

[5] X. Li, S. Pawar, and K. Ramchandran, "Sub-linear time compressed sensing using sparse-graph codes," in *Proceedings of the IEEE International Symposium on Information Theory*, pp. 1645–1649, 2015.

[6] T. Richardson and R. Urbanke, *Modern coding theory*. Cambridge University Press, 2008.

[7] S. Kumar, A. J. Young, N. Macris, and H. D. Pfister, "Threshold saturation for spatially coupled ldpc and ldgm codes on bms channels," *IEEE Transactions on Information Theory*, vol. 60, no. 12, pp. 7389–7415, 2014.

[8] S. Kudekar, T. Richardson, and R. L. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 7761–7813, 2013.