Peer reviewed version

University of Bristol - Explore Bristol Research
General rights

Performance of group testing algorithms
with constant tests-per-item

Oliver Johnson[*]   Matthew Aldridge[†]   Jonathan Scarlett[‡]

November 21, 2016

Abstract

We consider the nonadaptive group testing problem in the case that each item appears in a constant number of tests, chosen uniformly at random with replacement, so that the testing matrix has (almost) constant column weights. We analyse the performance of simple and practical algorithms in a range of sparsity regimes, showing that the performance is consistently improved in comparison with more standard Bernoulli designs. In particular, using a constant-column weight design, the DD algorithm is shown to outperform all possible algorithms for Bernoulli designs in a broad range of sparsity regimes, and to beat the best-known theoretical guarantees of existing practical algorithms in all sparsity regimes.

Keywords: algorithms, concentration-of-measure, group testing.
MSC: 60C05; 68W40

1   Introduction and notation

The group testing problem was introduced by Dorfman [12], as described in [13, Chapter 1.1]. While a large variety of problem setups have been considered, they all share common features, and can be considered in a wider class of sparse inference problems which includes compressed sensing [1]. Group testing has since been applied in a wide variety of contexts, for example including biology [11, 18, 31, 36, 39], anomaly detection in networks [20, 25], signal processing and data analysis [17, 19], and communications [8, 37, 40, 41], though this list is very far from being exhaustive.

[*]School of Mathematics, University of Bristol, University Walk, Bristol, BS8 1TW, UK. Email: `maotj@bristol.ac.uk`

[†]Department of Mathematical Sciences, University of Bath, Claverton Down, Bath, BA2 7AY, UK. Heilbronn Institute for Mathematical Research, Howard House, Queens Avenue, Bristol, BS8 1SN, UK. Email: `m.aldridge@bath.ac.uk`

[‡]Laboratory for Information and Inference Systems, École Polytechnique Fédérale de Lausanne, CH-1015, Switzerland. Email: `jonathan.scarlett@epfl.ch`

In this paper, we prove rigorous performance bounds on algorithms for the nonadaptive noiseless group testing problem. We consider a testing design in which each item is in a fixed number of tests, and study practical algorithms for detecting the defective items. We shall see that we can detect items with fewer tests compared to a more commonly-considered design in which items are placed in tests independently at random.

Let us fix some notation. Suppose we have a large number of items N, of which K are 'defective' in some sense. We assume that the defectives are rare, with $K = o(N)$ as $N \to \infty$; moreover, for concreteness we follow [6, 34, 35] by taking $K = \Theta(N^\theta)$ for some fixed parameter $\theta \in (0, 1)$. We follow the 'combinatorial model' and suppose that $\mathcal{K}$, the true set of defective items, is uniformly random from the $\binom{N}{K}$ sets of this size.

We perform a sequence of tests to form an estimate $\widehat{\mathcal{K}}$ of $\mathcal{K}$, and study the tradeoff between maximising the success probability $\mathbb{P}(\mathrm{suc}) := \mathbb{P}(\widehat{\mathcal{K}} = \mathcal{K})$ and minimising the number of tests T. We could simply take $T = N$, and test each item one by one. However, Dorfman's key insight [12] is that since the problem is sparse (in the sense that $K \ll N$), the outcomes of the tests are zero with high probability, so these tests are not optimally informative. A better procedure considers a series of subsets ('pools') of items which are tested together, where (in an idealized testing procedure) the outcome of each test is positive if and only if it contains at least one defective item. More formally:

Definition 1.1. We store the testing pools in a (possibly random) binary matrix $\mathsf{X} \in \{0, 1\}^{T \times N}$, where $x_{ti} = 1$ if test t includes item i and $x_{ti} = 0$ otherwise. The rows of X correspond to tests, and the columns correspond to items. The outcomes of each test are stored in a binary vector $\mathbf{y} = (y_t) \in \{0, 1\}^T$, where a positive outcome $y_t = 1$ occurs if and only if $x_{ti} = 1$ for some $i \in \mathcal{K}$.

We estimate the defective set by $\widehat{\mathcal{K}} = \widehat{\mathcal{K}}(\mathsf{X}, \mathbf{y})$, and write

$$\mathbb{P}(\mathrm{suc}) = \frac{1}{\binom{N}{K}} \sum_{|\mathcal{K}|=K} \mathbb{P}(\widehat{\mathcal{K}} = \mathcal{K}),$$

where the probability is over the random test design X.

We have freedom to design the test matrix, and in this paper, we focus on the non-adaptive case, where the entire matrix is fixed in advance of the tests. In the adaptive case (where the members of each test are chosen using the outcomes of the previous tests), Hwang's Generalized Binary Splitting Algorithm [21] recovers the defective set $\mathcal{K}$ using $\log_2 \binom{N}{K} + O(K)$ tests. This can be seen to be essentially optimal by a standard argument based on Fano's inequality (see for example [10]), a strengthened version of which (see [9]) implies that any algorithm using T tests has success probability bounded above by

$$\mathbb{P}(\mathrm{suc}) \leq \frac{2^T}{\binom{N}{K}}. \tag{1}$$

This means that any algorithm with success probability $\mathbb{P}(\mathrm{suc})$ tending to 1 requires at least

$$T = \log_2 \binom{N}{K} \sim (1 - \theta) K \log_2 N \ \text{ tests} \tag{2}$$

(see [5, Lemma 25] for details of the asymptotic behaviour of the binomial coefficient). This motivates the following definition [5] of the rate of an algorithm.

Definition 1.2. For any algorithm using T tests, we define the *rate* to be

$$\frac{\log_2 \binom{N}{K}}{T}. \tag{3}$$

Given a random matrix design, we say that R is an *achievable rate* if for any $\epsilon > 0$, there exists a group testing algorithm with rate at least R and success probability at least $1 - \epsilon$ for N sufficiently large. We define the capacity of a design to be the supremum of the achievable rates for that design.

Intuitively, one can think of R as being the number of bits of information learned per test when the recovery is successful.

In this language, Hwang's adaptive algorithm [21] has achievable rate $R = 1$ in the regime $K = \Theta(N^\theta)$ (and is therefore optimal, since no algorithm can learn more than 1 bit per test). It is an interesting question to consider whether there exists a matrix design and an algorithm with achievable rate $R = 1$ in the *nonadaptive* case. Indeed, it appears to be difficult even to design a class of matrices with non-zero achievable rate using combinatorial constructions (see [13, 27] for reviews of the extensive literature on this subject, with key early contributions coming from [14, 15, 28]). Hence, much recent work on nonadaptive group testing has considered Bernoulli random designs, where each item appears in each test independently with fixed probability; see for example [2, 3, 5, 7, 10, 34, 35]. In particular:

Theorem 1.3 ([2, 34]). *The capacity for Bernoulli nonadaptive group testing with $K = \Theta(N^\theta)$ defectives is*

$$C(\theta) = \max_{\nu > 0} \min \left\{ \frac{\nu e^{-\nu}}{\ln 2} \frac{1 - \theta}{\theta}, \ h(e^{-\nu}) \right\}, \tag{4}$$

where $h(t) = -t \log_2 t - (1 - t) \log_2(1 - t)$ is the binary entropy function. In particular, for $\theta \leq 1/3$, the capacity of Bernoulli designs is 1.

The curve (4) is illustrated in Figure 1 below. For $\theta \geq 1/2$, the paper [5] showed that the capacity (4) is achieved by a simple algorithm, which we refer to as DD (see Definition 3.3 for a description). However, for $\theta < 1/2$, the algorithms known to achieve the capacity (4) are based on maximising the likelihood or solving other difficult combinatorial problems, and cannot be considered as practical in a computational sense – see Section 3.4 for more detail. For example, we describe the SSS algorithm in Definition 3.4, which achieves the capacity but is impractical for large values of N and K.

The structure of the remainder of the paper is as follows. In Section 2, we summarise the main results of the paper, and provide simulation results to illustrate the performance of various algorithms. In Section 3, we describe the main algorithms used and introduce some key quantities that control their performance. In Section 4, we state some distributional results (proved in the appendices) and deduce the main theorem of the paper.

3

2   Main results

Our main result provides strict improvements on Theorem 1.3 in a broad range of sparsity regimes, as well as strict improvements over existing practical algorithms in all scaling regimes. To do this, we make use of the following class of test designs, following our initial work in [6].

Definition 2.1 (Constant column weight designs)**.** We define the constant-column weight testing design via a testing matrix X in which $L = \nu T/K$ entries of each column of are selected uniformly at random with replacement and set to 1 for some parameter $\nu > 0$, with independence between columns. The remaining entries of X are set to 0.

We refer to these as 'constant column weight' designs despite the fact that some columns will have weight slightly less than L. Since the weight of a column is the number of tests an item is in, these designs are also known as 'constant tests-per-item'.

In [6], we showed that $\nu = \ln 2$ is optimal with respect to all of the bounds derived therein, and we will also use this value throughout the present paper. This choice ensures that each test is equally likely to be positive or negative, and can thus be thought of as being maximally informative.

In our previous paper [6], we proved the two following results.

Theorem 2.2. *Using a constant column weight design in the regime where there are $K = \Theta(N^\theta)$ defectives:*

1. *([6, Theorem 1]) The COMP algorithm of Chan et al. [10] (see Section 3.1) has achievable rate $\ln 2\,(1 - \theta)$ (improving by 30.7% on the rate of $(1 - \theta)/(e \ln 2)$ proved by [5, Theorem 11] for COMP with Bernoulli designs).*

2. *([6, Theorem 2]) Regardless of the choice of $\nu > 0$, no algorithm can have a rate greater than*

$$\min\left\{1, \ln 2\, \frac{1 - \theta}{\theta}\right\}.$$

For completeness, we provide a proof of Theorem 2.2 in Section 4, albeit with $\nu = \ln 2$ in the second statement as opposed to a general choice.

The main result of this paper is the following theorem, which is proved in the same section.

Theorem 2.3. *For constant column weight designs in the regime where there are $K = \Theta(N^\theta)$ defectives, the DD algorithm has success probability tending to 1 if*

$$T \geq (1 + \epsilon) \frac{1}{(\ln 2)^2} \max\{\theta, 1 - \theta\} K \ln N,$$

and hence has achievable rate

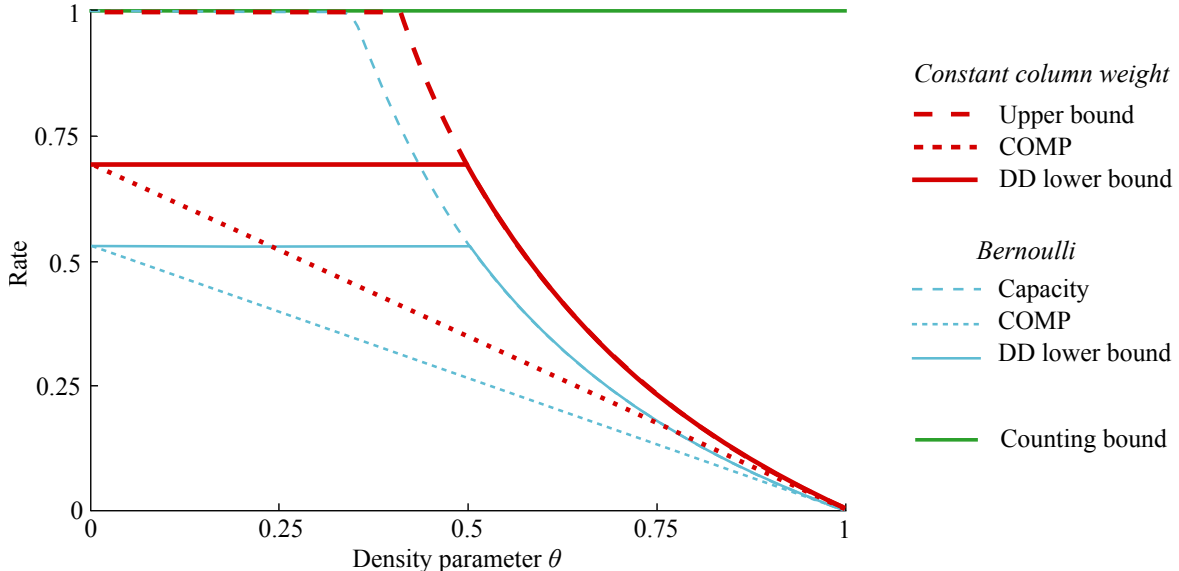$$R = \ln 2\, \min\left\{1, \frac{1 - \theta}{\theta}\right\}.$$

4

Figure 1: Rates and bounds for group testing algorithms with Bernoulli designs and constant column weight designs. In red, we plot the rate bounds for constant column weight designs from Theorems 2.2 and 2.3. In blue, we plot the rate bounds for Bernoulli designs from Theorem 1.3 and [5]. The green line represents the bound arising from (1).

We illustrate the above bounds in Figure 1, with the various rates for constant column weight designs marked in red, and corresponding rates for Bernoulli designs marked in blue. Note that in particular, a comparison of Theorems 1.3, 2.2 and 2.3 shows the following.

Remark 2.4. 1. DD with constant column weight designs offers the best rate currently proved for 'practical' algorithms for all θ.

2. DD with constant column weight designs outperforms any possible algorithm (practical or not) for Bernoulli designs, for all values of $\theta > 1/(1 + e(\ln 2)^2) \approx 0.434$.

3. If we use a constant column weight design, the DD algorithm gives the optimal performance for $\theta \geq 1/2$.

To complement our lower bound, in Theorem 4.7, we provide an upper bound for the rate of DD with constant column weight designs. Although our upper bound does not reveal whether the lower bound is tight at low values of θ, it does reveal that some amount of gap to the rate of 1 is unavoidable.

Although the rates are asymptotic as $N \to \infty$, in Figure 2 we illustrate the performance of these algorithms in a finite blocklength sense, in an illustrative sparse case ($N = 500$, $K = 10$) and a denser case ($N = 2000$, $K = 100$). For the sparse case, in addition to plotting performance of COMP and DD, we plot the performance of the SSS algorithm, which (see [3, Corollary 4]) achieves the capacity bounds of Theorem 1.3, though is not practical for larger problems. Because of this issue of practicality, we do not consider SSS

for the denser case. Instead, we plot the performance of a related algorithm called SCOMP, which is described in [5], so we omit a description in this paper for the sake of brevity. Essentially, it amouts to performing DD followed by greedy refinements.

Constant column weight designs are not new in the literature, as we briefly describe below. Our key contribution is a *rigorous* analysis of such designs in the regime $K = \Theta(N^\theta)$, including in particular the first such analysis for the DD algorithm, and the first results to attain the benefits described in Remark 2.4.

At an intuitive level, we believe the improvement over Bernoulli designs arises due to the fact that the latter can result in some items appearing in considerably fewer tests than average, meaning that it is harder for any algorithm to infer their defectivity status.

Related work:

Kautz and Singleton [22] observed that good group testing peformance is obtained by matrices corresponding to constant weight codes with a high minimum distance. However, ensuring the minimum distance is sufficiently large is not easy in practice. Mézard *et al.* [30] considered randomized designs with both fixed row and column weights, and with fixed column weights only. They suggested that such designs can beat Bernoulli designs; however, their 'short-loops' assumption is shown to be rigorous only for $\theta > 5/6$, and in fact fails for small θ. Chan *et al.* [10] considered constant row-weight designs and find no improvement over Bernoulli designs. Wadayama [38] analysed constant row and column weight designs in the $K = cN$ regime, and demonstrates close-to-optimal asymptotic performance for certain ratios of parameter sizes. D'yachkov *et al.* [16] used exactly-constant column weights, and setting their list size to one corresponds to insisting that COMP succeeds. However they only considered the case that $K = O(1)$; in the limit as K gets large, the rate $\ln 2$ obtained [16, Claim 2] matches the rate for COMP given in Theorem 2.2.1 as $\theta \to 0$.

3 Description of decoding algorithms used

Here we describe the COMP, DD and SSS algorithms, as introduced in [10] and [5], and discuss the conditions under which they succeed. We make one further key definition.

Definition 3.1. Given an item i and a set of items $\mathcal{L}$, we say that i is *masked* by $\mathcal{L}$ if every test which includes i also includes at least one member of $\mathcal{L}$.

3.1 COMP algorithm

The COMP algorithm [10] is based on a simple inference: Any negative test only contains non-defective items, so any item in a negative test can be marked as non-defective. Given enough negative tests, we might hope to correctly infer every member of $\mathcal{K}^c$ in this way. Formally speaking, the COMP algorithm proceeds as follows:
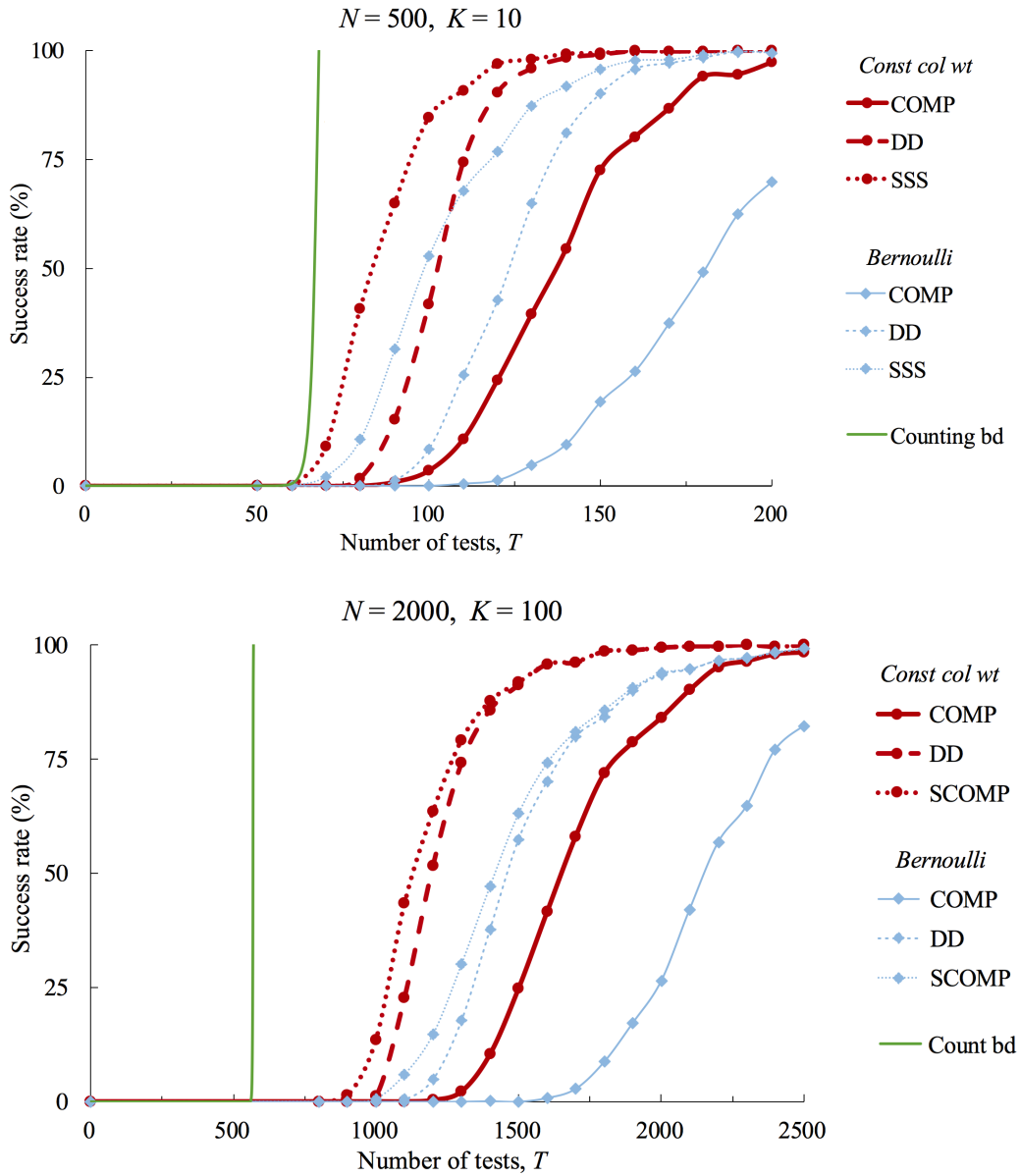
6

Figure 2: Empirical performance (based on 1000 experiments) of various algorithms for both constant column weight and Bernoulli designs, in the cases $N = 500$, $K = 10$ and $N = 2000$, $K = 100$.

Definition 3.2 (COMP algorithm, [10])**.**

1. Mark each item which appears in a negative test as non-defective, and refer to every other item as a Possible Defective (PD) – we write $\mathcal{PD}$ for the set of such items.

2. Mark every item in $\mathcal{PD}$ as defective.

Clearly, the first step will not make any mistakes (every item marked as non-defective will indeed be non-defective), so errors will only occur in the second step. As a result COMP will always estimate $\mathcal{K}$ by a set $\widehat{\mathcal{K}}_{\text{COMP}}$ with $\mathcal{K} \subseteq \widehat{\mathcal{K}}_{\text{COMP}}$.

As in [5], a particular quantity of interest is $G := |\mathcal{PD} \setminus \mathcal{K}| = |\mathcal{PD}| - K$, the number of non-defective items masked by the defective set $\mathcal{K}$; that is, non-defective items which do not appear in any negative test. It is clear that COMP succeeds (recovers the defective set exactly) if and only if $G = 0$, so that

$$\mathbb{P}^{\text{COMP}}(\text{suc}) = \mathbb{P}(G = 0). \tag{5}$$

We use this in the proof of Theorem 2.2.1 in Section 4.

3.2 DD algorithm

The DD ('Definite Defectives') algorithm builds on the first step of COMP, as follows:

Definition 3.3 (DD algorithm [5])**.**

1. Mark each item which appears in a negative test as non-defective, and refer to every other item as a Possible Defective ($\mathcal{PD}$).

2. For each positive test which contains a single Possible Defective item, mark that item as defective.

3. Mark all remaining items as non-defective.

Again, the first step will not make any mistakes, and since every positive test must contain at least one defective item, the second step is also certainly correct. Hence, any errors due to DD come from marking a true defective as non-defective in the third step, meaning that the estimate $\widehat{\mathcal{K}}_{\text{DD}} \subseteq \mathcal{K}$. The choice to mark all remaining items as non-defective is motivated by the sparsity of the problem, since *a priori* an item is much less likely to be defective than non-defective.

We analyse DD rigorously in Section 4, using the following notation, used in [5] and illustrated in Figure 3. For each $i \in \mathcal{K}$, we write:

- M_i for the number of tests containing defective item i and no other defective;

- L_i for the number of tests containing defective item i and no other possible defective item (no other member of $\mathcal{PD}$).

8

In the terminology of Definition 3.1, we see that DD succeeds if and only if no defective item $i \in \mathcal{K}$ is masked by $\mathcal{PD} \setminus \{i\}$. Further, since item i is masked by $\mathcal{PD} \setminus \{i\}$ if and only if $L_i = 0$, we can write:

$$\mathbb{P}^{\mathrm{DD}}(\mathrm{suc}) = 1 - \mathbb{P}\left(\bigcup_{i \in \mathcal{K}} \{L_i \neq 0\}\right). \tag{6}$$

For a given defective item $i \in \mathcal{K}$, we write $\mathcal{K}^{(i)} = \mathcal{K} \setminus \{i\}$ for the set of defectives with i removed. For a given set $\mathcal{M}$, we write $W^{(\mathcal{M})}$ for the total number of tests containing at least one item from $\mathcal{M}$. The random variable $W^{\mathcal{K} \setminus \{i\}}$ (the total number of tests containing at least one item in $\mathcal{K}^{(i)}$), henceforth denoted by $W^{(\mathcal{K} \setminus i)}$, will be of particular interest.

To understand the distributions of the quantities illustrated in Figure 3, as in [6], it is helpful to think of the process by which elements of the columns are sampled as a *coupon collector* problem, where each coupon corresponds to one of the T tests. Recall also that that distinct columns of X are sampled independently. Hence, for a single item, $W^{(\{i\})}$ is the number of distinct coupons selected when L coupons are chosen uniformly at random from a population of T coupons. In general, for a set $\mathcal{M}$ of size M, the independence of distinct columns means that $W^{(\mathcal{M})}$ is the number of distinct coupons chosen when choosing ML coupons uniformly at random from a population of T coupons.

Hence, as described in more detail in Section 4, we can first give a concentration of measure result for $W^{(\mathcal{K} \setminus i)}$ (see Lemma 4.1), then characterize the distribution of M_i given $W^{(\mathcal{K} \setminus i)}$ (see Proposition 4.2). Following this, we can state the distribution of G conditioned on $W^{(\mathcal{K})} = W^{(\mathcal{K} \setminus i)} + M_i$ (see Lemma 4.3), and finally deduce the distribution of L_i conditioned on G and $W^{(\mathcal{K})}$ (see Lemma 4.4). This allows us to deduce bounds on (6).

3.3 SSS algorithm

We describe one more algorithm, which we refer to as SSS, in terminology taken from [5].

Definition 3.4 (SSS algorithm [5])**.** Say that a putative defective set $\mathcal{J}$ is 'satisfying' if:

1. No negative test contains a member of $\mathcal{J}$.

2. Every positive test contains at least one member of $\mathcal{J}$.

The SSS algorithm simply finds the Smallest Satisfying Set (breaking ties at random), and takes that as an estimate $\widehat{\mathcal{K}}_{\mathrm{SSS}}$.

Note that the true defective set $\mathcal{K}$ is certainly a satisfying set, hence SSS is guaranteed to return a set of no larger size (that is $|\widehat{\mathcal{K}}_{\mathrm{SSS}}| \leq |\mathcal{K}|$). However, it may well not be the case that $\widehat{\mathcal{K}}_{\mathrm{SSS}} \subseteq \mathcal{K}$. We can, however, see a possible failure event for SSS: If a defective item $i \in \mathcal{K}$ is masked by the other defective items $\mathcal{K} \setminus \{i\}$ (in the sense of Definition 3.1) then $\mathcal{K} \setminus \{i\}$ will be a satisfying set in the sense above, so SSS is certain to fail.
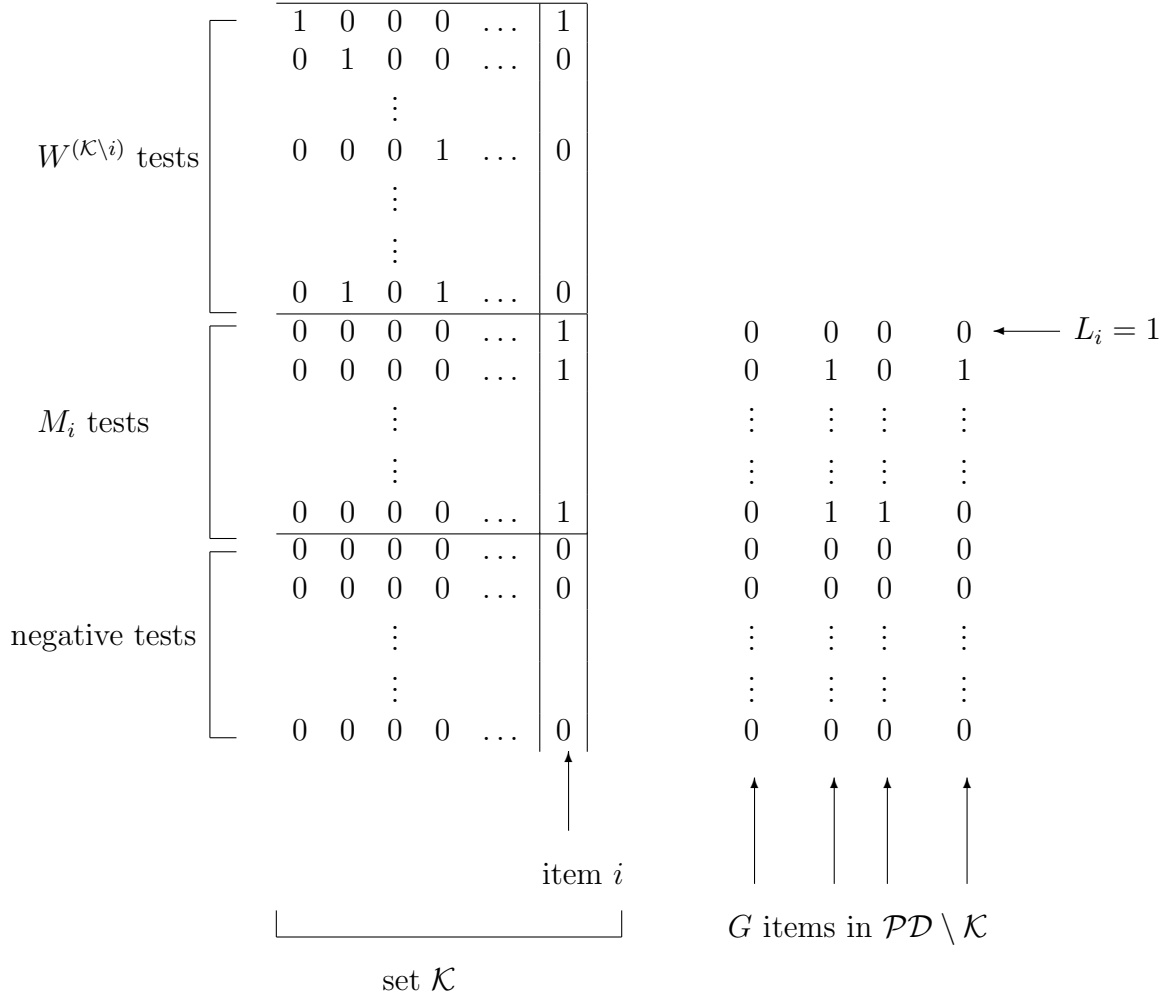
$$
\begin{array}{c}
W^{(\mathcal{K}\backslash i)} \text{ tests} \\
\\
M_i \text{ tests} \\
\\
\text{negative tests}
\end{array}
\qquad
\begin{array}{ccccc|c}
1 & 0 & 0 & 0 & \ldots & 1 \\
0 & 1 & 0 & 0 & \ldots & 0 \\
  &   & \vdots &   &   &   \\
0 & 0 & 0 & 1 & \ldots & 0 \\
  &   & \vdots &   &   &   \\
  &   & \vdots &   &   &   \\
0 & 1 & 0 & 1 & \ldots & 0 \\
\hline
0 & 0 & 0 & 0 & \ldots & 1 \\
0 & 0 & 0 & 0 & \ldots & 1 \\
  &   & \vdots &   &   &   \\
  &   & \vdots &   &   &   \\
0 & 0 & 0 & 0 & \ldots & 1 \\
\hline
0 & 0 & 0 & 0 & \ldots & 0 \\
0 & 0 & 0 & 0 & \ldots & 0 \\
  &   & \vdots &   &   &   \\
  &   & \vdots &   &   &   \\
0 & 0 & 0 & 0 & \ldots & 0
\end{array}
\qquad
\begin{array}{cccc}
0 & 0 & 0 & 0 \quad\longleftarrow L_i = 1\\
0 & 1 & 0 & 1 \\
\vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots \\
0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0
\end{array}
$$

item i

set $\mathcal{K}$ $\qquad\qquad$ G items in $\mathcal{PD} \setminus \mathcal{K}$

Figure 3: Illustration of random variables M_i, L_i, $W^{(\mathcal{K}\backslash i)}$ and G. For simplicity of presentation, and without loss of generality, we illustrate the case where $\mathcal{K} = \{1, \ldots, K\}$ and $i = K$. We present a version of X with rows sorted so that the tests containing no defectives appear at the bottom, and the $W^{(\mathcal{K}\backslash i)}$ tests containing at least one item of $\mathcal{K}^{(i)} = \{1, \ldots, K-1\}$ appear at the top.

Hence, writing A_i for the event that item i is masked by $\mathcal{K} \setminus \{i\}$, we can use the Bonferroni inequality to obtain a lower bound on the SSS error probability $\mathbb{P}^{\text{SSS}}(\text{err})$ of the form

$$\mathbb{P}^{\text{SSS}}(\text{err}) \geq \mathbb{P}\left(\bigcup_{i \in \mathcal{K}} A_i\right) \geq \sum_{i \in \mathcal{K}} \mathbb{P}(A_i) - \frac{1}{2}\sum_{i \neq j \in \mathcal{K}} \mathbb{P}\left(A_i \cap A_j\right). \tag{7}$$

This serves as a starting point for upper bounding the rate of the SSS algorithm.

3.4 Note on practical feasibility

We refer to COMP and DD as practical algorithms, since they can be implemented with low run-time. For example, COMP simply requires us to take one pass through the test matrix and outcomes, requiring no more than $O(N)$ storage beyond the matrix itself, and $O(TN)$ runtime. Similarly, DD builds on COMP, requiring two passes through the test matrix and outcomes and can be performed with the same amount of storage and runtime.

In contrast, we can think of SSS as a 0-1 linear programming problem, meaning that it is unlikely to be practical to run in practice for large problems. However, we think of it as a 'best possible' algorithm (this is made more rigorous in [2]), to deduce overall performance bounds. Note that although the SSS algorithm may be considered to be infeasible in practice, the paper [26] shows that a relaxation of the 0-1 linear programming problem to the reals can give good performance.

Further, the decoding algorithms we consider here do not require exact, or even approximate, knowledge of K. This compares to the optimal maximum likelihood decoder of [34], which requires the exact value of K. Note, however, that optimal choice of the parameter $\nu = (\ln 2)T/K$ in the *design* stage does require K.

4 Proofs of bounds on the rate

The ultimate goal of this section is to prove our achievable rate for the DD algorithm, but along the way, we will also prove the COMP rate and the SSS upper bound, since they will essentially come 'for free'.

4.1 Concentration of $W^{(\mathcal{M})}$

Recall that $W^{(\mathcal{M})}$ corresponds to total number of tests in which items from $\mathcal{M}$ are placed. The following lemma shows that this quantity is concentrated around its mean.

Lemma 4.1. *When making $LM = \alpha T$ draws with replacement from a total of T coupons, the total number of distinct coupons $W^{(\mathcal{M})}$ satisfies*

$$\mathbb{P}\left(\left|W^{(\mathcal{M})} - (1 - \exp(-\alpha))T\right| \geq \epsilon T\right) \leq 2\exp\left(-\frac{\epsilon^2 T}{\alpha}\right). \tag{8}$$

Proof. This result was stated as [6, Lemma 1]. The key insight is that the value of $W^{(\mathcal{M})}$ changes by at most 1 when the value of any particular coupon is changed, so the result follows using McDiarmid's inequality [29]. □

4.2 Upper bound on SSS rate

For completeness, we give a proof of Theorem 2.2.2 (SSS rate) using the above concentration result, albeit for $\nu = \ln 2$ instead of general $\nu > 0$ in Definition 2.1 (the latter is proved similarly).

Proof of Theorem 2.2.2. The upper bound of 1 is well-known for arbitrary test designs, following for example from (1), so we choose T according to the other term, setting $T = \gamma_{\mathrm{SSS}} \theta K \ln N$ with $\gamma_{\mathrm{SSS}} = (1 - \epsilon)/(\ln 2)^2$. Hence, $L = K \ln 2/T = (1 - \epsilon)\theta \ln N / \ln 2$. The key is to observe that Lemma 4.1 above shows that for $M = K - 2$ or $M = K - 1$, choosing $\alpha = LM/T = \ln 2 M/K \sim \ln 2$ reveals that $W^{(\mathcal{M})}$ is exponentially concentrated around $1/2$.

We consider the two terms of the RHS of (7) separately. Fixing i, conditional on $W^{(\mathcal{K}\backslash i)} = w$, the event A_i occurs if each test that item i occurs in is contained in the w 'already hit' tests. Hence, for any $c_1 > 0$, we can write

$$K\mathbb{P}(A_i) = K \sum_w \mathbb{P}\big(A_i \mid W^{(\mathcal{K}\backslash i)} = w\big)\mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)$$

$$= K \sum_w \left(\frac{w}{T}\right)^L \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)$$

$$\geq K \sum_w \left(\frac{w}{T}\right)^L \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)\mathbb{I}(w \geq Tc_1)$$

$$\geq Kc_1^L \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} \geq Tc_1\big) \tag{9}$$

Similarly, for any $i \neq j$, we can write the following for any $c_2 > 0$:

$$\binom{K}{2}\mathbb{P}(A_i \cap A_j)$$

$$= \binom{K}{2} \sum_w \mathbb{P}\big(A_i \cap A_j \mid W^{(\mathcal{K}\backslash i,j)} = w\big)\, \mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} = w\big)$$

$$= \binom{K}{2} \sum_w \left(\frac{w}{T}\right)^{2L} \mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} = w\big)$$

$$\leq \binom{K}{2} \sum_w \left(\frac{w}{T}\right)^{2L} \mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} = w\big)\, \mathbb{I}(w \leq Tc_2) + \binom{K}{2}\mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} \geq Tc_2\big)$$

$$\leq \frac{K^2}{2} c_2^{2L}\, \mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} \leq Tc_2\big) + \binom{K}{2}\mathbb{P}\big(W^{(\mathcal{K}\backslash i,j)} \geq Tc_2\big). \tag{10}$$

Combining (9) and (10), we obtain that (7) is at least

$$K c_1^L \, \mathbb{P}\big(W^{(\mathcal{K}\setminus i)} \geq T c_1\big) \left(1 - \frac{K c_2^{2L} \, \mathbb{P}(W^{(\mathcal{K}\setminus i,j)} \leq T c_2)}{c_1^L \, \mathbb{P}(W^{(\mathcal{K}\setminus i)} \geq T c_1)}\right) - \binom{K}{2} \mathbb{P}\big(W^{(\mathcal{K}\setminus i,j)} \geq T c_2\big),$$

which we can bound away from zero using Lemma 4.1 by taking $c_1 \sim 2^{1/(\theta(1-\epsilon))}$ and taking $c_2 = 1/2(1 + \epsilon')$ for ϵ' sufficiently small. $\qquad\square$

4.3 Conditional distributions of M_i and G

For any integers (n, k), define

$$\begin{Bmatrix} n \\ k \end{Bmatrix} := \frac{1}{k!} \sum_{j=0}^{k} (-1)^{k-j} \binom{k}{j} j^n \tag{11}$$

for the Stirling number of the second kind (see for example [24, Eq. (8)]).

Recall that M_i denotes the number of tests containing defective item i but no other defectives items. The following proposition gives the distribution of this quantity conditioned on $W^{(\mathcal{K}\setminus i)}$, the number of tests covered by those other defectives.

Proposition 4.2.

1. *We can write the distribution of $M_i \mid W^{(\mathcal{K}\setminus i)}$ explicitly as*

$$\mathbb{P}\left(M_i = j \mid W^{(\mathcal{K}\setminus i)} = w\right) = \frac{(T-w)_{(j)}}{T^L} \sum_{s=0}^{L-j} \binom{L}{s} \begin{Bmatrix} L-s \\ j \end{Bmatrix} w^s, \tag{12}$$

   where $(T-w)_{(j)} := (T-w)!/(T-w-j)!$ denotes the falling factorial.

2. *There exists an explicit constant $C := C(L, w) = \exp(L^2/(4w))$ such that*

$$\mathbb{P}\left(M_i = j \mid W^{(\mathcal{K}\setminus i)} = w\right) \leq C \binom{L}{j} \left(1 - \frac{w}{T}\right)^j \left(\frac{w}{T}\right)^{L-j},$$

   that is, a multiple of the $\mathrm{Bin}(L, 1 - w/T)$ mass function.

Proof. See Appendix A. $\qquad\square$

Next, we observe that (see Figure 3) we can write $W^{(\mathcal{K})} = W^{(\mathcal{K}\setminus i)} + M_i$. Recall that G is the number of non-defectives masked by the defective set $\mathcal{K}$, and observe that since an item is only counted in G if each of the tests appearing in the corresponding column are in the set of size $W^{(\mathcal{K})}$, we have the following.

Lemma 4.3. *Conditional on $W^{(\mathcal{K})} = x$ we have*

$$G \mid \{W^{(\mathcal{K})} = x\} \sim \mathrm{Bin}\big(N - K, (x/T)^L\big).$$

13

4.4 Proof of COMP achievable rate

For completeness, we give a proof of Theorem 2.2.1 using the above results.

Proof of Theorem 2.2.1. We consider the regime where $T = \gamma_{\text{COMP}} K \ln N$, where $\gamma_{\text{COMP}} = (1 + \epsilon)/(\ln 2)^2$. As mentioned in (6), COMP succeeds if and only if $G = 0$. Using Lemma 4.3, we know that

$$\mathbb{P}(G = 0 \mid W^{(\mathcal{K})} = x) = \left(1 - \left(\frac{x}{T}\right)^L\right)^{N-K},$$

which is a decreasing function in x. Hence, given δ, for all $x \leq (1/2 + \delta)T$, we have

$$\mathbb{P}(G = 0 \mid W^{(\mathcal{K})} = x) \geq \mathbb{P}(G = 0 \mid W^{(\mathcal{K})} = (1/2 + \delta)T)$$
$$= \left(1 - (1/2 + \delta)^L\right)^{N-K}.$$

Next, using the fact that

$$L = T \ln 2 / K = \gamma_{\text{COMP}} \ln 2 \ln N = (1 + \epsilon) \ln N / \ln 2,$$

we find that for any ϵ, we can choose δ sufficiently small that $(1/2 + \delta)^L \leq N^{-(1+\epsilon/2)}$, and hence

$$\mathbb{P}(G = 0 \mid W^{(\mathcal{K})} = x) \geq \left(1 - N^{-(1+\epsilon/2)}\right)^{N-K}.$$

We deduce that the success probability of COMP satisfies

$$\mathbb{P}^{\text{COMP}}(\text{suc}) = \sum_x \mathbb{P}\left(W^{(\mathcal{K})} = x\right) \mathbb{P}\left(G = 0 \mid W^{(\mathcal{K})} = x\right)$$

$$\geq \sum_{x \leq (1/2+\delta)T} \mathbb{P}\left(W^{(\mathcal{K})} = x\right) \left(1 - N^{-(1+\epsilon/2)}\right)^{N-K}$$

$$= \left(1 - N^{-(1+\epsilon/2)}\right)^{N-K} \left[1 - \mathbb{P}\left(W^{(\mathcal{K})} \geq (1/2 + \delta)T\right)\right],$$

which is seen to tend to 1 by taking $\alpha = \ln 2$ in Lemma 4.1 (since we collect a total of $KL = T \ln 2$ coupons). $\qquad\square$

4.5 Conditional distribution of L_i

Recalling that L_i denotes the number of tests containing defective item i and no other "possible defective" (item from $\mathcal{PD}$), we have the following.

Lemma 4.4. *For any g, w, j, we have*

$$\mathbb{P}\left(L_i = 0 \mid G = g, W^{(\mathcal{K} \setminus i)} = w, M_i = j\right) = \phi_j\left(\frac{1}{w+j}, gL\right),$$

where

$$\phi_j(s, V) = \sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} (1 - \ell s)^V. \qquad (13)$$

14

Proof. See Appendix A. □

Note that the function $\phi_j(s, V)$ also appeared in [5]; however, our analysis here requires using it very differently. In particular, we make use of the following properties, the proofs of which are deferred to Appendix B.

Lemma 4.5. *For all values of j, s and V, the function $\phi_j(s, V)$ introduced in (13) has the properties that:*

1. *$\phi_j(s, V)$ is increasing in s,*

2. *$\phi_j(s, V)$ is increasing in V,*

3. *$\phi_j(s, V)$ is decreasing in j.*

Proposition 4.6. *If $sVj \leq 2$, then*

$$\phi_j(s, V) \leq \frac{V! s^j}{(V-j)!} \leq \exp\left(j \ln(Vs)\right).$$

4.6 Proof of the DD achievable rate

We put the above results together to prove Theorem 2.3, giving a lower bound on the achievable rate of the DD algorithm. As in [5], the key is to express the success probability $\mathbb{P}^{\mathrm{DD}}(\mathrm{suc})$ in terms of an expectation of the function ϕ, and to show that this expectation is concentrated in a regime where ϕ takes favourable values.

Proof of Theorem 2.3. Recall that we consider the regime where $T = \gamma_{\mathrm{DD}} mK \ln N$, with $\gamma_{\mathrm{DD}} = (1+\epsilon)/(\ln 2)^2$ and $m = \max(\theta, 1-\theta)$. It is useful to know that in this regime, L satisfies $L \ln 2 = T(\ln 2)^2/K = m(1+\epsilon) \ln N$.

As in [5], writing $\mathbb{P}^{\mathrm{DD}}(\mathrm{suc})$ for the success probability of DD and applying the union bound to (6) we know that

$$\mathbb{P}^{\mathrm{DD}}(\mathrm{suc}) = 1 - \mathbb{P}\left(\bigcup_{i \in \mathcal{K}} \{L_i = 0\}\right) \geq 1 - \sum_{i \in \mathcal{K}} \mathbb{P}(L_i = 0), \tag{14}$$

so that $\mathbb{P}^{\mathrm{DD}}(\mathrm{suc})$ will tend to 1 (as required) if, for a particular defective item $i \in \mathcal{K}$,

$$K\mathbb{P}(L_i = 0) \to 0, \tag{15}$$

since symmetry means that $\mathbb{P}(L_i = 0)$ is equal for each $i \in \mathcal{K}$. The stated value for the rate then follows on substituting the value of T and (2) into (3).

In order to characterize $\mathbb{P}(L_i = 0)$, we define $A = \{w_- \leq W^{(\mathcal{K}\backslash i)} \leq w_+\}$ and $B = \{G \leq g^*\}$, for some w_-, w_+ and g^* to be chosen shortly. Using Lemma 4.4, we can write

$$\mathbb{P}(L_i = 0)$$

$$= \sum_{w,j,g} \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w, M_i = j, G = g\big)\big(\mathbb{I}(A \cap B) + \mathbb{I}((A \cap B)^c)\big)\phi_j\left(\frac{1}{w+j}, gL\right)$$

$$\leq \sum_{w \in (w_-, w_+)} \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)\sum_{j=0}^{L}\mathbb{P}\big(M_i = j \mid W^{(\mathcal{K}\backslash i)} = w\big)\phi_j(1/w_-, g^*L) + \mathbb{P}\big((A \cap B)^c\big)$$

$$\tag{16}$$

$$= \sum_{w \in (w_-, w_+)} \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)\sum_{j=0}^{L}\mathbb{P}\big(M_i = j \mid W^{(\mathcal{K}\backslash i)} = w\big)\phi_j(1/w_-, g^*L)$$

$$+ \mathbb{P}(A^c) + \mathbb{P}(A \cap B^c)$$

$$\leq \sum_{w \in (w_-, w_+)} \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} = w\big)\sum_{j=0}^{L}\mathbb{P}\big(M_i = j \mid W^{(\mathcal{K}\backslash i)} = w\big)\phi_j(1/w_-, g^*L)$$

$$+ \mathbb{P}\big(W^{(\mathcal{K}\backslash i)} \notin (w_-, w_+)\big) + \mathbb{P}\big(G > g^* \mid W^{(\mathcal{K}\backslash i)} \in (w_-, w_+)\big), \tag{17}$$

where:

- (16) follows because, by Lemma 4.5, on the event $\{A \cap B\}$ the bound $\phi_j(1/w, gL) \leq \phi_j(1/w_-, g^*L)$ holds and everywhere else $\phi \leq 1$ (since ϕ represents a probability);

- (17) folllows since $\mathbb{P}(A \cap B^c) = \mathbb{P}(B^c \mid A)\mathbb{P}(A) \leq \mathbb{P}(B^c \mid A)$.

We consider the terms of (17) separately, taking $w_- = T(1-\delta)/2$, $w_+ = T(1+\delta)/2$, $g^* = N(1/2 + \delta)^L$, where $\delta = (\epsilon \ln 2)/4(1 + \epsilon)$.

1. The first term of (17) can be bounded as follows. Combining $L = \ln 2T/K$ and $w_- = T(1-\delta)/2$ gives $L/w_- = 2\ln 2/(K(1-\delta))$, and recalling that $g^* = N(1/2+\delta)^L$ and $m = \max(\theta, 1 - \theta)$, it follows that

$$\beta := \ln\left(\frac{g^*L}{w_-}\right) = (1 - \theta)\ln N + L\ln(1/2 + \delta) + \ln\left(\frac{2\ln 2}{1 - \delta}\right)$$

$$\leq m\left(1 + (1 + \epsilon)\left(-1 + \frac{2\delta}{\ln 2}\right)\right)\ln N + \ln\left(\frac{2\ln 2}{1 - \delta}\right)$$

$$\leq m\left(-\frac{\epsilon}{2}\right)\ln N + \ln\left(\frac{2\ln 2}{1 - \delta}\right), \tag{18}$$

where the second line follows by combining $L = m(1+\epsilon)\ln N/\ln 2$ and $\ln(1/2 + \delta) \leq -\ln 2 + 2\delta$, and the third line follows since $1 + (1 + \epsilon)(-1 + 2\delta/\ln 2) \leq -\epsilon/2$ under the above choice $\delta = (\epsilon \ln 2)/4(1 + \epsilon)$.

We claim that (18) implies $jg^*L/w_- \leq 2$ for all $j \leq L$. Indeed, we have $T = \Theta(K \log N)$ and $L = \Theta(T/K)$, so that $L = \Theta(\log N)$, whereas (18) implies that g^*L/w_- decays to zero strictly faster than $1/\log N$. This implies that

$$\phi_j(1/w_-, g^*L) \leq \exp\left(j \ln \left(\frac{g^*L}{w_-} \right) \right),$$

since the conditions of Proposition 4.6 are satisfied under these arguments.

Writing $\phi(j) = \phi_j(1/w_-, g^*L)$ (which is decreasing in j by Lemma 4.5.3), we can bound K times the inner sum of (17) using (18), as follows:

$$K \sum_{j=0}^{L} \mathbb{P}\big(M_k = j \mid W^{(\mathcal{K} \setminus i)} = w\big) \phi(j)$$

$$\leq KC(L, w) \sum_{j=0}^{L} \mathbb{P}\big(\mathrm{Bin}(L, 1 - w/T) = j\big) \phi(j) \tag{19}$$

$$= KC(L, w) \sum_{j=0}^{L} \mathbb{P}\big(\mathrm{Bin}(L, 1 - w_+/T) = j\big) \left(\frac{\mathbb{P}(\mathrm{Bin}(L, 1 - w/T) = j)}{\mathbb{P}(\mathrm{Bin}(L, 1 - w_+/T) = j)} \right) \phi(j) \tag{20}$$

$$\leq KC(L, w) \sum_{j=0}^{L} \mathbb{P}\big(\mathrm{Bin}(L, 1 - w_+/T) = j\big) \phi(j) \tag{21}$$

$$\leq KC(L, w_-) \left(\frac{w_+}{T} + \frac{T - w_+}{T} \exp(\beta) \right)^L \tag{22}$$

$$= C(L, w_-) \frac{K}{2^L} \big(1 + \delta + \exp(\beta)(1 - \delta)\big)^L \tag{23}$$

$$\leq C(L, w_-) c \exp\left(-\theta\epsilon \ln N\right) \exp\big(L(\delta + \exp(\beta)(1 - \delta))\big) \tag{24}$$

$$\leq C(L, w_-) c \exp\left(\left(-\theta\epsilon + \frac{m(1 + \epsilon)}{\ln 2} (\delta + \exp(\beta)(1 - \delta)) \right) \ln N \right). \tag{25}$$

Here:

- (19) follows from the second part of Proposition 4.2.
- (21) uses the following argument: The bracketed term in (20) is easily verified to be increasing in j by substituting the Binomial mass function and noting $1 - w/T \geq 1 - w_+/T$, and we already know from Lemma 4.5 that $\phi(j)$ is decreasing. Hence, (20) is the expectation of the product of an increasing and decreasing function, and is therefore bounded above by the product of the expectations of those functions. (This uses 'Chebyshev's other inequality' – see [23, eq. (1.7)]). In fact, [23, eq. (1.7)] concerns $\mathbb{E}[f(X)g(X)]$ for two *increasing* functions, but we can transform this to $\mathbb{E}[f(X)h(X)]$ for decreasing h by simply defining $h(\cdot) = L - g(\cdot)$.)

- (22) follows by upper bounding $C(L, w) \le C(L, w_-)$ and $\phi(j) \le e^{j\beta}$ (as established above), and then evaluating the summation explicitly.
- (23) follows by substituting $w_+ = T(1 + \delta)/2$.
- (24) follows from $1 + \zeta \le e^\zeta$, along with the fact that

$$\frac{K}{2^L} \exp\left((\theta - m(1 + \epsilon)) \ln N\right) \le c \exp(-\theta\epsilon \ln N)$$

by $L = m(1 + \epsilon) \ln N / \ln 2$ and $K = \Theta(N^\theta)$ (and hence $K \le cN^\theta$ for some $c = \Theta(1)$).
- (25) follows by again using $L = m(1 + \epsilon) \ln N / \ln 2$.

We conclude that (25) acts as an upper bound on K times the first term of (17). Overall (25) tends to zero for δ sufficiently small, since $C(L, w_-) = \exp(L^2/(4w_-))$ tends to 1 in this regime.

2. The second term of (17) decays to zero exponentially fast in N, using Lemma 4.1. In this case, we make $(K - 1)L$ draws with replacement, so that $\alpha = (K - 1) \ln 2 / K \to \ln 2$, meaning that we can take $\epsilon = \delta/3$ in Lemma 4.1 to obtain

$$\limsup_{N\to\infty} K \, \mathbb{P}(W^{(\mathcal{K}\backslash i)} \notin (w_-, w_+)) \le \limsup_{N\to\infty} K \, \mathbb{P}\left(\left|W^{(\mathcal{K}\backslash i)} - (1 - \exp(-\alpha))T\right| \ge \epsilon T\right)$$

$$\le 2 \limsup_{N\to\infty} K \exp\left(-\frac{\epsilon^2 T}{\alpha}\right)$$

$$= 2c \limsup_{N\to\infty} \exp\left(\ln N\left(\theta - \frac{\epsilon^2 \gamma_{\mathrm{DD}} mK}{\alpha}\right)\right),$$

since $T = \gamma_{\mathrm{DD}} mK \ln N$ and $K = \Theta(N^\theta)$ (and hence $K \le cN^\theta$ for some $c = \Theta(1)$). We conclude that this term tends to zero, since the exponent behaves as $-K \ln N$.

3. To control the third term in (17), observe that if $W^{(\mathcal{K}\backslash i)} \le w_+$, then

$$\frac{W^{(\mathcal{K})}}{T} \le \frac{1 + \delta}{2} + \frac{L}{T} = \frac{1 + \delta}{2} + \frac{\ln 2}{K} \le \frac{1}{2} + \frac{3\delta}{4},$$

where the first inequality holds since $W^{(\mathcal{K})} \le W^{(\mathcal{K}\backslash i)} + L$ and $w_+ = T(1 + \delta)/2$, the equality holds since $L = \ln 2 \, T/K$, and the final inequality holds for K sufficiently large. Hence, and defining $p = (1/2 + 3\delta/4)^L$, Lemma 4.3 gives that

$$\mathbb{P}\left(G > g^* \mid W^{(\mathcal{K}\backslash i)} \in (w_-, w_+)\right) \le \mathbb{P}\left(\mathrm{Bin}(N, p) > g^*\right)$$

$$\le \exp\left(-\frac{(g^*)^2}{2(Np + g^*/3)}\right) \tag{26}$$

$$= \exp\left(-N\frac{(1/2 + \delta)^{2L}}{2\left((1/2 + 3\delta/4)^L + (1/2 + \delta)^L/3\right)}\right)$$

$$= \exp\left(-N\frac{(1/2 + \delta)^L}{2(1/3 + o(1))}\right), \tag{27}$$

18

where the second line follows from Bernstein's inequality [**?**, Eq. (2.10)], the third line follows from $p = (1/2 + 3\delta/4)^L$ and $g^* = N(1/2 + \delta)^L$, and the final line follows since the ratio of $(1/2 + 3\delta/4)^L$ to $(1/2 + \delta)^L$ tends to zero as $N \to \infty$ (and hence $L \to \infty$, since $L = \Theta(\log N)$).

Finally, since $L = m(1 + \epsilon) \log N / \log 2$, we find that $(1/2 + \delta)^L$ behaves as N^c for some c that can be made arbitrarily close to $m = \max(\theta, 1 - \theta)$ by choosing δ and ϵ sufficiently small. By definition, $m < 1$, and the bound in (27) is exponential in N^{1-c}, yielding the desired result. $\qquad \square$

4.7   Upper bound on DD rate

The following theorem provides an upper bound on the rate of DD with constant column weight designs.

Theorem 4.7. *The maximum achievable rate of DD with a constant column weight design and $K = \Theta(N^\theta)$ defectives is upper bounded by*

$$R \le \max_{\nu > 0} \min \left\{ e^{-\nu}(1 + \nu)\, h\!\left(\frac{1}{1 + \nu}\right),\ -\nu \log(1 - e^{-\nu}) \frac{1 - \theta}{\theta} \right\}. \tag{28}$$

This bound is illustrated in Figure 4.

Remark 4.8. More simply, and only very slightly worse, we have

$$R \le \min \left\{ 0.839,\ \ln 2 \, \frac{1 - \theta}{\theta} \right\}. \tag{29}$$

We get (29) from (28) by optimizing the first term (numerically) at $\nu \approx 0.563$, and optimizing the second term (by differentiating) at $\nu = \ln 2$.

The message here is that while is it is unclear whether the low-sparsity rates of DD with constant column weight designs can be improved via different analysis techniques, we know that some amount of gap to the counting bound is unavoidable. More specifically, while we don't know the maximum achievable rate of DD for $\theta < 1/2$, (29) shows the rate is always below 0.839, so we cannot achieve the counting bound of 1. For the fixed choice $\nu = \ln 2$, one can slightly tighten this upper bound to 0.826.

Proof. The second term in (28) is the algorithm-independent converse of [6] with ν unoptimised. (Specialising to $\nu = \ln 2$ gives the bound of [6] and Theorem 2.2 part 2 of this paper, and is the second term in the simpler form (29).) It remains to prove the first term, for fixed ν. Then optimising over ν gives the result.

We follow a similar proof for the Bernoulli case in [4]. The idea is that the group testing rate is bounded by the entropy rate of the outcomes, $R \le H(\mathbf{Y})/T$. A simple bound would be to write $H(\mathbf{Y}) \le \sum_{t=1}^{T} H(Y_t)$ and note that $H(Y_t) \le 1$, to get the counting bound. However, a key point is that DD makes no use of tests containing two or more defectives – these necessarily have two or more Possible Defectives, so cannot be used to 'convert' a
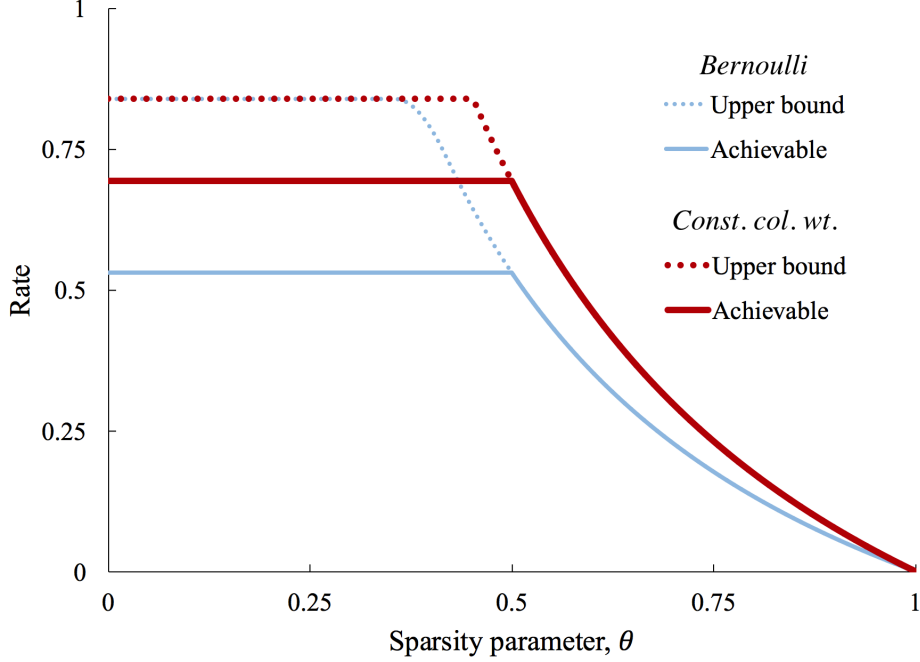
Figure 4: Achievable rates and upper bounds for the DD algorithm with Bernoulli and constant column weight designs. The Bernoulli upper bound is from [4], the Bernoulli achievability is from [5], the constant column weight upper bound is Theorem 4.7, and the constant column weight achievability is Theorem 2.3.

possible defective to a definite defective – so we can bound the group testing rate by the entropy rate of the outcomes with only zero or one defectives.

Write π_0, π_1, π_{2+} for the probability a given test contains, respectively zero, one, and at least two defectives, and write D_0, D_1, and D_{2+} for the number of such tests. Then

$$\pi_0 = \left(1 - \frac{1}{T}\right)^{KL} = \left(1 - \frac{1}{T}\right)^{\nu T} \to \mathrm{e}^{-\nu},$$

$$\pi_1 = K\left(1 - \left(1 - \frac{1}{T}\right)^{L}\right)\left(1 - \frac{1}{T}\right)^{(K-1)L}$$

$$= K\left(1 - \left(1 - \frac{1}{T}\right)^{\nu T/K}\right)\left(1 - \frac{1}{T}\right)^{\nu T(1-1/K)}$$

$$\sim K\left(1 - \mathrm{e}^{-\nu/K}\right)\mathrm{e}^{-\nu}$$

$$\to \nu\mathrm{e}^{-\nu}$$

$$\pi_{2+} = 1 - \pi_0 - \pi_1 \to 1 - \mathrm{e}^{-\nu} - \nu\mathrm{e}^{-\nu}.$$

Here, the expression for π_1 comes about since we need a test to be 'hit' by one or more of the L coupons corresponding to one of the K defectives, and missed by the $(K-1)L$ coupons corresponding to the other $K - 1$ defectives. Further, by linearity of expectation,

we have

$$\mathbb{E}D_0 = \pi_0 T, \qquad \mathbb{E}D_1 = \pi_1 T, \qquad \mathbb{E}D_{2+} = \pi_{2+}T,$$

and we have exponential concentration of D_0, D_1, D_{2+} around these means. (For D_0 this follows immediately form Lemma 4.1; the argument for D_1 and D_{2+} is identical, so we do not repeat it here.)

We fix $\epsilon > 0$. The event that $D_{2+} \geq (1 - \mathrm{e}^{-\nu} - \nu\mathrm{e}^{-\nu} - \epsilon)T$ has probability at least $1 - \epsilon$ for N (and hence T) sufficiently large. Thus, by conditioning on this event, and recalling the above discussion that DD only uses tests with 0 or 1 defectives, we have for sufficiently large N that

$$R \leq \epsilon + \frac{1}{T} H\big(\mathbf{Y}_{0,1} \mid D_{2+} \geq (1 - \mathrm{e}^{-\nu} - \nu\mathrm{e}^{-\nu} - \epsilon)T\big) \tag{30}$$

$$\leq 2\epsilon + \frac{1}{T} H\big(\mathbf{Y}_{0,1} \mid D_{2+} = (1 - \mathrm{e}^{-\nu} - \nu\mathrm{e}^{-\nu} - \epsilon)T\big) \tag{31}$$

$$\leq 2\epsilon + \frac{1}{T} (\mathrm{e}^{-\nu} + \nu\mathrm{e}^{-\nu} + \epsilon)T\, h\left(\frac{\pi_0'}{\pi_0' + \pi_1'}\right) \tag{32}$$

$$\leq 3\epsilon + \frac{1}{T} (\mathrm{e}^{-\nu} + \nu\mathrm{e}^{-\nu} + \epsilon)T\, h\left(\frac{\mathrm{e}^{-\nu}}{\mathrm{e}^{-\nu} + \nu\mathrm{e}^{-\nu}}\right) \tag{33}$$

$$= 3\epsilon + (\mathrm{e}^{-\nu} + \nu\mathrm{e}^{-\nu} + \epsilon)\, h\left(\frac{1}{1+\nu}\right),$$

where:

- In (30), $\mathbf{Y}_{0,1}$ is the vector of outcomes of the tests containing 0 or 1 defectives, and H is the entropy function.

- In (31), we have used that the entropy of $\mathbf{Y}_{0,1}$ given D_{2+} is decreasing in D_{2+}. Moreover, we have added ϵ to account for the uncertainty in the length of $\mathbf{Y}_{0,1}$ itself, which behaves as $o(T)$ since the length concentrates exponentially around its mean.

- In (32), we have used the inequality $H(\mathbf{Y} \mid \cdot) \leq \sum_i H(Y_i \mid \cdot)$, and we have defined π_0' and π_1' to be defined in the same way as π_0 and π_1 but conditioned on $D_{2+} = (1 - \mathrm{e}^{-\nu} - \nu\mathrm{e}^{-\nu} - \epsilon)T$, as well as letting h denote the binary entropy function.

- In (33), we have used continuity of h to replace $\pi_0'/(\pi_0' + \pi_1')$ with its limit, at the cost of an extra ϵ. Specifically, we have $\pi_0' \to \pi_0$ and $\pi_1' \to \pi_1$ as $\epsilon \to 0$, and these in turn have limiting values as characterized above.

Taking ϵ arbitrarily small gives the result. $\qquad\square$

5 Conclusions

We have shown the simple and practical DD algorithm outperforms the best possible group testing algorithms (practical or not) for Bernoulli designs for a broad range of (dense)

sparsity levels, as well as showing that it beats the best known performance bounds for practical algorithms with Bernoulli designs at all sparsity levels.

We briefly mention four interesting open problems connected with this paper, which we hope to address in future work:

1. It remains open to determine the capacity of constant column weight designs for $\theta \leq 1/2$, in the spirit of Theorem 1.3. We conjecture that the value is given by a similar expression, corresponding to the performance of a maximum likelihood algorithm (or equivalently the SSS algorithm of [5]), and that Part 2 of Theorem 2.2 is sharp.

2. It is an important open problem to decide whether 'practical' algorithms can improve on the performance of DD. For example, the SCOMP algorithm of [5] and approaches based on linear programming both have a rate at least as large as DD [4]. However, we do not know how to determine whether these algorithms or others can have a higher rate than DD.

3. It remains of great interest to determine whether a capacity of 1 can be achieved for values of θ beyond 1/3 using constant column weights or some other design, as well as determining whether there exists an *adaptivity gap*: Does there exist $\theta < 1$ such that any non-adaptive design must have rate less than 1, despite the rate of 1 being achievable when adaptivity is allowed?

4. A design of potential interest is that with both constant column weights (tests-per-item) and constant row weights (items-per-test). On one hand, the nonrigorous work of Mézard, Tarzia and Toninelli [30] suggests that for $\theta < 1$ there may be no gain over constant column weight designs considered here. On the other hand, in the light of Wadayama's work [38] on constant row-and-column designs in the regime where K grows linearly with N, and the performance of LDPC codes, this does seem like a natural place to look for improvements. Either way, the independence of columns of X plays a significant role in the proofs of this paper, so new arguments would be required to investigate this.

Acknowledgments

A   Properties of the distribution of M_i

A.1   Proof of Proposition 4.2

Proof of Proposition 4.2. The two parts of the proposition are proved as follows:

1. We prove this part directly. Suppose that we pick L coupons from a population of T coupons, w of which were previously chosen. Clearly, the probability of the event that exactly s of the coupons picked were previously chosen is $\mathbb{P}(\text{Bin}(L, w/T) = s)$.

   Conditioning on this event, we calculate the probability that we pick $L - s$ coupons out of a population of $T - w$ coupons and obtain exactly j distinct new coupons. Clearly we require $L - s \geq j$, or $s \leq L - j$. By a standard counting argument, we can choose these j coupons $\binom{T-w}{j}$ ways, then $\left\{{L-s \atop j}\right\}$ ways of placing the $L - s$ coupons into j unlabelled bins such that none of them are empty (see [24, P.204]), and finally $j!$ different labellings of the bins. Moreover, overall there are $(T - w)^{L-s}$ assignments of the coupons.

   Putting this all together and recalling the definition

   $$(T - w)_{(j)} = \frac{(T - w)!}{(T - w - j)!} = \binom{T - w}{j} j!,$$

   we have

   $$\mathbb{P}\left(M_i = j \mid W^{(\mathcal{K}\setminus i)} = w\right)$$

   $$= \sum_{s=0}^{L-j} \binom{L}{s} \left(\frac{w}{T}\right)^s \left(1 - \frac{w}{T}\right)^{L-s} \left(\binom{T - w}{j} \left\{{L - s \atop j}\right\} j! \frac{1}{(T - w)^{L-s}}\right)$$

   $$= \sum_{s=0}^{L-j} \binom{L}{s} \frac{w^s}{T^L} (T - w)_{(j)} \left\{{L - s \atop j}\right\},$$

   as required.

2. Relabelling $t = L - j - s \geq 0$ and using the fact that $\left\{{t+j \atop j}\right\} \leq \binom{t+j}{j} j^t$ (see [33]), we obtain that the inner sum of (12) is:

   $$\sum_{t=0}^{L-j} \binom{L}{L - j - t} \left\{{t + j \atop j}\right\} w^{L-j-t} \leq w^{L-j} \sum_{t=0}^{L-j} \binom{L}{L - j - t} \binom{t + j}{j} \left(\frac{j}{w}\right)^t$$

   $$= w^{L-j} \binom{L}{j} \sum_{t=0}^{L-j} \binom{L - j}{t} \left(\frac{j}{w}\right)^t$$

   $$= w^{L-j} \binom{L}{j} \left(1 + \frac{j}{w}\right)^{L-j}$$

   $$\leq w^{L-j} \binom{L}{j} C,$$

   where the third line follows by explicitly evaluating the summation, and the final line holds with $C = \exp(L^2/(4w))$ since

   $$(1 + j/w)^{L-j} \leq \exp(j(L - j)/w) \leq \exp(L^2/(4w)).$$

This allows us to deduce that the whole of (12) satisfies

$$\mathbb{P}\big(M_i = j \mid W^{(\mathcal{K}\setminus i)} = w\big) \leq C\frac{(T-w)_{(j)}}{T^L} \, w^{L-j} \binom{L}{j} \leq C\frac{(T-w)^j}{T^L} \, w^{L-j} \binom{L}{j},$$

as required. □

A.2   Proof of Lemma 4.4

Proof of Lemma 4.4. The case $j = 0$ is trivial, so we assume here that $j \geq 1$.

We have conditioned on $W^{(\mathcal{K}\setminus i)} = w$ (i.e., there are w tests containing one or more item from $\mathcal{K} \setminus i$), $M_i = j$ (i.e., there are j tests that contain item i and no member of $\mathcal{K} \setminus i$), and $G = g$ (i.e., there are g items labelled as possibly defective but not in $\mathcal{K}$).

By relabelling, without loss of generality, we can assume that tests $1, \ldots, j$ are the ones that contain defective item i and no other defective item. We write A_s for the event that test s does not have any element of $\mathcal{PD} \setminus \mathcal{K}$ in it.

If an item is in $\mathcal{PD} \setminus \mathcal{K}$, then the tests that it appears in are chosen uniformly among those which already contain a defective. Hence, for any set $S \subseteq \{1, \ldots, j\}$ of size ℓ, we have $\mathbb{P}(\bigcap_{r \in S} A_r) = (1 - \ell/(w+j))^{gL}$, since we require that the L coupons of each of g items in $\mathcal{PD} \setminus \mathcal{K}$ take values in the set of positive tests ($W^{(\mathcal{K}\setminus i)} + M_i = w + j$ in total), but avoid the specified ℓ tests. Thus,

$$\mathbb{P}\big(L_i = 0 \mid G = g, W^{(\mathcal{K}\setminus i)} = w, M_i = j\big) = \mathbb{P}\left(\bigcap_{s=1}^{j} A_s^c\right)$$

$$= 1 - \mathbb{P}\left(\bigcup_{s=1}^{j} A_s\right)$$

$$= \sum_{\ell=0}^{j} (-1)^\ell \sum_{\substack{S \subseteq \{1,\ldots,j\} \\ |S|=\ell}} \mathbb{P}\left(\bigcap_{j \in S} A_j\right)$$

$$= \sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} \left(1 - \frac{\ell}{w+j}\right)^{gL}, \qquad (34)$$

and the result follows. □

B   Properties of the ϕ function

B.1   Proof of Lemma 4.5

Proof of Lemma 4.5. Using the expression

$$\phi_j(s, V) = \sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} (1 - \ell s)^V$$

24

from (13), we deduce the results as follows:

1. As in [5, Lemma 32], a direct calculation using the fact that $\ell\binom{j}{\ell} = j\binom{j-1}{\ell-1}$ gives

$$\frac{\partial}{\partial s}\phi_j(s, V) = Vj\sum_{\ell=1}^{j}(-1)^{\ell-1}\binom{j-1}{\ell-1}(1 - \ell s)^{V-1} \tag{35}$$

$$= (1 - s)^{V-1}Vj\sum_{\ell=1}^{j}(-1)^{\ell-1}\binom{j-1}{\ell-1}\left(1 - \frac{(\ell-1)s}{1-s}\right)^{V-1}$$

$$= (1 - s)^{V-1}Vj\phi_{j-1}\left(\frac{s}{1-s}, V - 1\right) \tag{36}$$

$$\geq 0,$$

where the second line uses the fact that

$$(1 - \ell s) = (1 - s)\left(1 - \frac{(\ell-1)s}{1-s}\right),$$

and the third line above follows by relabelling $\ell' = \ell - 1$.

2. Again using $\ell\binom{j}{\ell} = j\binom{j-1}{\ell-1}$, we can write

$$\phi_j(s, V) - \phi_j(s, V - 1) = \sum_{\ell=0}^{j}(-1)^{\ell}\binom{j}{l}(1 - \ell s)^{V-1}((1 - \ell s) - 1)$$

$$= sj\sum_{\ell=1}^{j}(-1)^{\ell-1}\binom{j-1}{\ell-1}(1 - \ell s)^{V-1}$$

$$= \frac{s}{V}\frac{\partial}{\partial s}\phi_j(s, V)$$

$$\geq 0,$$

where the third line follows from (35).

3. Finally, by expanding $\binom{j}{\ell} = \binom{j-1}{\ell} + \binom{j-1}{\ell-1}$, we can write

$$\phi_j(s, V) = \sum_{\ell=0}^{j}(-1)^{\ell}\left(\binom{j-1}{\ell} + \binom{j-1}{\ell-1}\right)(1 - \ell s)^{V}$$

$$= \phi_{j-1}(s, V) - \frac{1}{(V+1)j}\frac{\partial}{\partial s}\phi_j(s, V + 1)$$

$$\leq \phi_{j-1}(s, V),$$

again using (35).

$$\square$$

25

B.2 Proof of Proposition 4.6

We now prove Proposition 4.6, first giving two preliminary lemmas.

Lemma B.1. *We can expand $\phi_j(s, V)$ (as defined in (13)) as a polynomial in s of degree V as follows:*

$$\phi_j(s, V) = \frac{s^j V!}{(V-j)!} \sum_{u=0}^{V-j} (-1)^u s^u \frac{j!(V-j)!}{(u+j)!(V-u-j)!} \left\{ {j+u \atop j} \right\}, \tag{37}$$

where we again write $\left\{ {j+u \atop j} \right\}$ for the Stirling number of the second kind.

Proof. We can expand

$$
\begin{aligned}
\phi_j(s, V) &= \sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} (1 - \ell s)^V \\
&= \sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} \sum_{t=0}^{V} \binom{V}{t} (-s)^t \ell^t \\
&= \sum_{t=0}^{V} \binom{V}{t} (-s)^t \left(\sum_{\ell=0}^{j} (-1)^\ell \binom{j}{\ell} \ell^t \right) \\
&= \sum_{t=0}^{V} \binom{V}{t} (-s)^t \left(\left\{ {t \atop j} \right\} j! (-1)^j \right) \\
&= \sum_{u=0}^{V-j} \binom{V}{j+u} (-s)^{j+u} \left(\left\{ {j+u \atop j} \right\} j! (-1)^j \right),
\end{aligned} \tag{38}
$$

where the second line can be seen by directly evaluating the summation, the forth line follows by recognising the bracketed inner sum in (38) as a multiple of the Stirling number using (11), and the last line follows since by relabelling $t = j + u$ and noting that $\left\{ {t \atop j} \right\}$ is non-zero only when $t \geq j$. The result now follows by writing

$$\binom{V}{j+u} = \frac{V!}{(V-u-j)!(u+j)!} \cdot \frac{(V-j)!}{(V-j)!}$$

and $(-s)^{j+u}(-1)^j = (-1)^u s^j s^u$. □

We also use the following result from [32, Theorem 4.4].

Lemma B.2. *The Stirling numbers of the second kind are log-concave in their first argument, that is for any $j, u \in \mathbb{Z}_+$:*

$$\left\{ {j+u+1 \atop j} \right\}^2 \geq \left\{ {j+u \atop j} \right\} \left\{ {j+u+2 \atop j} \right\}.$$

We deduce Proposition 4.6 as follows:

Proof of Proposition 4.6. Using Lemma B.1, we consider $\phi_j(s, V)$ as a sum of the form

$$\phi_j(s, V) = \frac{s^j V!}{(V-j)!} \sum_{u=0}^{V-j} (-1)^u a_u,$$

where

$$a_u = s^u \frac{j!(V-j)!}{(u+j)!(V-u-j)!} \begin{Bmatrix} j+u \\ j \end{Bmatrix}.$$

By the alternating series test, if a_u is a monotonically decreasing sequence, we can bound $\sum_{u=0}^{V-j}(-1)^u a_u \le a_0 = 1$, and the result follows. We can verify that a_u is indeed monotonically decreasing by considering the ratio:

$$\frac{a_{u+1}}{a_u} = s \left(\frac{V-j-u}{j+u+1} \right) \left(\frac{\begin{Bmatrix} j+u+1 \\ j \end{Bmatrix}}{\begin{Bmatrix} j+u \\ j \end{Bmatrix}} \right). \tag{39}$$

The first bracketed term in (39) is trivially decreasing in u, and the second bracketed term in (39) is decreasing in u by Lemma B.2. Hence, since the ratio (39) is decreasing in u, it is sufficient to verify that $a_1/a_0 \le 1$. Since $\begin{Bmatrix} j \\ j \end{Bmatrix} = 1$ and $\begin{Bmatrix} j+1 \\ j \end{Bmatrix} = j(j+1)/2$, direct substitution in (39) gives that $a_1/a_0 = s(V-j)j/2$, so it is sufficient to assume that $s(V-j)j/2 \le 1$. $\qquad\square$

References

[1] C. Aksoylar, G. Atia, and V. Saligrama. Sparse signal processing with linear and non-linear observations: A unified Shannon theoretic approach. In *Proceedings of the 2013 IEEE Information Theory Workshop*, pages 1–5, Sept 2013.

[2] M. P. Aldridge. The capacity of Bernoulli nonadaptive group testing. See `arxiv:1511:05201`, 2015.

[3] M. P. Aldridge, L. Baldassini, and K. Gunderson. Almost separable matrices. *Journal of Combinatorial Optimization (to appear)*, 2015. `doi:10.1007/s10878-015-9951-1`.

[4] M. Aldridge. On the optimality of some group testing algorithms. Preprint, 2016.

[5] M. P. Aldridge, L. Baldassini, and O. T. Johnson. Group testing algorithms: bounds and simulations. *IEEE Trans. Inform. Theory*, 60(6):3671–3687, 2014.

[6] M. P. Aldridge, O. T. Johnson, and J. Scarlett. Improved group testing rates with constant column weight designs. In *Proceedings of the 2016 IEEE International Symposium on Information Theory, Barcelona Spain, July 2016*, pages 1381–1385, 2016.

[7] G. Atia and V. Saligrama. Boolean compressed sensing and noisy group testing. *IEEE Trans. Inform. Theory*, 58(3):1880 –1901, March 2012.

[8] W. U. Bajwa, J. D. Haupt, A. M. Sayeed, and R. D. Nowak. Joint source–channel communication for distributed estimation in sensor networks. *IEEE Transactions on Information Theory*, 53(10):3629–3653, 2007.

[9] L. Baldassini, O. T. Johnson, and M. P. Aldridge. The capacity of adaptive group testing. In *Proceedings of the 2013 IEEE International Symposium on Information Theory, Istanbul Turkey, July 2013*, pages 2676–2680, 2013.

[10] C. L. Chan, P. H. Che, S. Jaggi, and V. Saligrama. Non-adaptive probabilistic group testing with noisy measurements: Near-optimal bounds with efficient algorithms. In *Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing*, pages 1832 –1839, Sept. 2011.

[11] H.-B. Chen and F. K. Hwang. A survey on nonadaptive group testing algorithms through the angle of decoding. *Journal of Combinatorial Optimization*, 15(1):49–59, 2008.

[12] R. Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, pages 436–440, 1943.

[13] D. Du and F. Hwang. *Combinatorial Group Testing and Its Applications*. Series on Applied Mathematics. World Scientific, 1993.

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

[15] A. G. D'yachkov, V. V. Rykov, and A. M. Rashad. Superimposed distance codes. *Problems Control and Information Theory*, 18(4):237–250, 1989.

[16] A. G. Dyachkov, I. V. Vorobev, N. Polyansky, and V. Y. Shchukin. Almost disjunctive list-decoding codes. *Problems of Information Transmission*, 51(2):110–131, 2015.

[17] A. Emad and O. Milenkovic. Poisson group testing: A probabilistic model for nonadaptive streaming Boolean compressed sensing. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3335–3339. IEEE, 2014.

[18] J. L. Gastwirth and P. A. Hammick. Estimation of the prevalence of a rare disease, preserving the anonymity of the subjects by group testing: Application to estimating the prevalence of AIDS antibodies in blood donors. *Journal of statistical planning and inference*, 22(1):15–27, 1989.

[19] A. C. Gilbert, M. A. Iwen, and M. J. Strauss. Group testing and sparse signal recovery. In *2008 42nd Asilomar Conference on Signals, Systems and Computers*, pages 1059–1063. IEEE, 2008.

[20] M. T. Goodrich and D. S. Hirschberg. Improved adaptive group testing algorithms with applications to multiple access channels and dead sensor diagnosis. *Journal of Combinatorial Optimization*, 15(1):95–121, 2008.

[21] F. K. Hwang. A method for detecting all defective members in a population by group testing. *Journal of the American Statistical Association*, 67(339):605–608, 1972.

[22] W. H. Kautz and R. C. Singleton. Nonrandom binary superimposed codes. *IEEE Transactions on Information Theory*, 10(4):363–377, 1964.

[23] J. F. C. Kingman. Uses of exchangeability. *Ann. Probability*, 6(2):183–197, 1978.

[24] E. H. Lieb. Concavity properties and a generating function for Stirling numbers. *J. Combinatorial Theory*, 5:203–206, 1968.

[25] C. Lo, M. Liu, J. P. Lynch, and A. C. Gilbert. Efficient sensor fault detection using combinatorial group testing. In *2013 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS),*, pages 199–206. IEEE, 2013.

[26] D. Malioutov and M. Malyutov. Boolean compressed sensing: Lp relaxation for group testing. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3305–3308. IEEE, 2012.

[27] M. Malyutov. Search for sparse active inputs: a review. In *Information Theory, Combinatorics and Search Theory*, volume 7777 of *Lecture notes in Computer Science*, pages 609–647. Springer, London, 2013.

[28] M. B. Malyutov. The separating property of random matrices. *Mathematical Notes*, 23(1):84–91, 1978.

[29] C. McDiarmid. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

[30] M. Mézard, M. Tarzia, and C. Toninelli. Group testing with random pools: Phase transitions and optimal strategy. *Journal of Statistical Physics*, 131(5):783–801, 2008.

[31] R. Mourad, Z. Dawy, and F. Morcos. Designing pooling systems for noisy high-throughput protein-protein interaction experiments using boolean compressed sensing. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 10(6):1478–1490, 2013.

[32] E. Neuman. On generalized symmetric means and Stirling numbers of the second kind. *Applicationes Mathematicae*, 4(18):645–656, 1985.

[33] B. Rennie and A. Dobson. On Stirling numbers of the second kind. *Journal of Combinatorial Theory*, 7(2):116 – 121, 1969.

[34] J. Scarlett and V. Cevher. Limits on support recovery with probabilistic models: An information-theoretic framework. *IEEE Trans. Inform. Thy. (to appear)*, 2015. See `arXiv:1501.07440`.

[35] J. Scarlett and V. Cevher. Phase transitions in group testing. In *27th ACM-SIAM Symposium on Discrete Algorithms (SODA), Arlington, Virginia, USA, January 10-12, 2016*, pages 40–53, 2016.

[36] K. H. Thompson. Estimation of the proportion of vectors in a natural population of insects. *Biometrics*, 18(4):568–578, 1962.

[37] M. K. Varanasi. Group detection for synchronous Gaussian code-division multiple-access channels. *IEEE Transactions on Information Theory*, 41(4):1083–1096, 1995.

[38] T. Wadayama. An analysis on non-adaptive group testing based on sparse pooling graphs. In *Proceedings of the 2013 IEEE International Symposium on Information Theory*, pages 2681–2685, 2013.

[39] S. D. Walter, S. W. Hildreth, and B. J. Beaty. Estimation of infection rates in populations of organisms using pools of variable size. *American Journal of Epidemiology*, 112(1):124–128, 1980.

[40] S. Wu, S. Wei, Y. Wang, R. Vaidyanathan, and J. Yuan. Achievable partition information rate over noisy multi-access boolean channel. In *2014 IEEE International Symposium on Information Theory*, pages 1206–1210. IEEE, 2014.

[41] S. Wu, S. Wei, Y. Wang, R. Vaidyanathan, and J. Yuan. Partition information and its transmission over boolean multi-access channels. *IEEE Transactions on Information Theory*, 61(2):1010–1027, 2015.