The capacity of non-identical adaptive group testing

Tom Kealy
CDT in Communications
MVB, School of Engineeering
University of Bristol, UK
Email: tom.kealy.kealy@bristol.ac.uk

Oliver Johnson
School of Mathematics
University Walk, Bristol
University of Bristol, UK
Email: O.Johnson@bristol.ac.uk

Robert Piechocki
CSN Group
MVB, School of Engineeering
University of Bristol, UK
Email: r.j.piechocki@bristol.ac.uk

***Abstract*—We consider the group testing problem, in the case where the items are defective independently but with non-constant probability. We introduce and analyse an algorithm to solve this problem by grouping items together appropriately. We give conditions under which the algorithm performs essentially optimally in the sense of information-theoretic capacity. This has applications to the allocation of spectrum to cognitive radios, in the case where a database gives prior information that a particular band will be occupied.**

I. Introduction and notation

A. The Probabilistic group testing problem

Group testing is a sparse inference problem, first introduced by Dorfman [**?**] in the context of testing for rare diseases. Given a large population of items $\mathcal{P}$, indexed by $\{1, \dots N\}$, where some small fraction of the items are interesting in some way, how can we find the interesting items efficiently?

We perform a sequence of T pooled tests defined by test sets $\mathcal{X}_1, \dots, \mathcal{X}_T$, where each $\mathcal{X}_i \subseteq \mathcal{P}$. We represent the interesting ('defective') items by a random vector $\mathbf{U} = (U_1, \dots, U_N)$, where U_i is the indicator of the event that item i is defective. For each test i, we jointly test all the items in $\mathcal{X}_i$, and the outcome y_i is 'positive' ($y_i = 1$) if and only if any item in $\mathcal{X}_i$ is defective. In other words, $y_i = \mathbb{I}\left(\sum_{j \in \mathcal{X}_i} U_j\right)$, since for simplicity we are considering the noiseless case. Further, in this paper, we restrict our attention to the adaptive case, where we choose test set $\mathcal{X}_i$ based on a knowledge of sets $\mathcal{X}_1, \dots, \mathcal{X}_{i-1}$ and outcomes $y_1, \dots, y_{i-1}$. The group testing problem requires us to infer $\mathbf{U}$ with high probability given a low number of tests T.

Since Dorfman's paper [**?**], there has been considerable work on the question of how to design the sets $\mathcal{X}_i$ in order to minimise the number of tests T required. In this context, we briefly mention so-called combinatorial designs (see [**?**], [**?**] for a summary, with [**?**] giving invaluable references to an extensive body of Russian work in the 1970s and 1980s). Such designs typically aim to ensure that set-theoretic properties known as disjunctness and separability occur. In contrast, for simplicity of analysis, as well as performance of optimal order, it is possible to consider random designs. Here sets $\mathcal{X}_i$ are chosen at random, either using constructions such as independent Bernoulli designs [**?**], [**?**], [**?**] or more sophisticated random designs based on LDPC codes [**?**].

Much previous work has focussed on the Combinatorial group testing problem, where there are a fixed number of defectives K, and the defectivity vector $\mathbf{U}$ is chosen uniformly among all binary vectors of weight K. In contrast, in this paper we study a Probabilistic group testing problem as formulated for example in the work of Li et al. [**?**], in that we suppose each item is defective independently with probability p_i, or equivalently take U_i to be independent Bernoulli(p_i).

This Probabilistic framework, including non-uniform priors, is natural for many applications of group testing. For example, see [**?**], the cognitive radio problem can be formulated in terms of a population of communication bands in frequency spectra with some (unknown) occupied bands you must not utilise. Here, the values of p_i may be chosen based on some database of past spectrum measurements or other prior information. Similarly, as in Dorfman's original work [**?**] or more recent research [**?**] involving screening for genetic conditions, values of p_i might summarise prior information based on a risk profile or family history.

B. Group testing capacity

It is possible to characterize performance tradeoffs in group testing from an information-theoretic point of view – see for example [**?**], [**?**], [**?**], [**?**]. These papers have focussed on group testing as a channel coding problem, with [**?**], [**?**] explicitly calculating the mutual information. The paper [**?**] defined the capacity of a Combinatorial group testing procedure, which characterizes the number of bits of information about the defective set which we can learn per test. We give a more general definition here, which covers both the Combinatorial and Probabilistic cases.

Definition 1.1: Consider a sequence of group testing problems where the ith problem has defectivity vector $\mathbf{U}^{(i)}$, and consider algorithms which are given $T(i)$ tests. We refer to a constant C as the (weak) group testing capacity if for any $\epsilon > 0$:

1) any sequence of algorithms with
$$\liminf_{i \to \infty} \frac{H(\mathbf{U}^{(i)})}{T(i)} \geq C + \epsilon, \tag{1}$$
has success probability $\mathbb{P}(\mathrm{suc})$ bounded away from 1,
2) and there exists a sequence of algorithms with
$$\liminf_{i \to \infty} \frac{H(\mathbf{U}^{(i)})}{T(i)} \geq C - \epsilon \tag{2}$$
with success probability $\mathbb{P}(\mathrm{suc}) \to 1$.

Remark 1.2: In the Combinatorial case of K defective items with all defective sets equally likely, $H(\mathbf{U}) = \log_2 \binom{N}{K}$, which is the term found in the denominator in [**?**, Eq. (1) and (2)]. In the Probabilistic case (as in [**?**]) we know $H(\mathbf{U}) = -\sum_{i=1}^{N} h(p_i)$ where $h(t) = -t \log_2 t - (1-t)\log_2(1-t)$ is the binary entropy function.

Remark 1.3: If for $\liminf_{i\to\infty} \frac{H(\mathbf{U}^{(i)})}{T(i)} \geq C + \epsilon$, the success probability $\mathbb{P}(\text{suc}) \to 0$ we say that C is the strong group testing capacity, following standard terminology in information theory. Such a result is referred to as a strong converse.

C. Main results

The principal contribution of [**?**, Theorem 1.2] was the following result:

Theorem 1.4 ([?]): The strong capacity of the adaptive noiseless Combinatorial group testing problem is $C = 1$, in any regime such that $K/N \to 0$.

This argument came in two parts. First, in [**?**, Theorem 3.1] the authors proved a new upper bound on success probability

$$\mathbb{P}(\text{suc}) \leq \frac{2^T}{\binom{N}{K}}, \tag{3}$$

which implied a strong converse ($C \leq 1$). This was complemented by showing that, in the Combinatorial case, an algorithm based on Hwang's Generalized Binary Splitting Algorithm (HGBSA) [**?**], [**?**] is essentially optimal in the required sense, showing that $C = 1$ is achievable.

It may be useful to characterize the Probabilistic group testing problem in terms of the effective sparsity $\mu^{(N)} := \sum_{i=1}^{N} p_i$. In particular, if the p_i are (close to) identical, we would expect performance similar to that in the Combinatorial case with $K = \mu^{(N)}$ defectives. As in [**?**], we focus on asymptotically sparse cases, where $\mu^{(N)}/N \to 0$ (in contrast, Wadayama [**?**] considered a model where p_i are identical and fixed). The main result of the present paper is Theorem 3.9, stated and proved in Section III-E below, which implies the following Probabilistic group testing version of Theorem 1.4.

Corollary 1.5: In the case where $p_i \equiv p$, the weak capacity of the adaptive noiseless Probabilistic group testing problem is $C = 1$, in any regime such that $\mu^{(N)}/N \to 0$ and $\mu^{(N)} \to \infty$.

Again we prove our main result Theorem 3.9 using complementary bounds on both sides. First in Section II-A we recall a universal upper bound on success probability, Theorem 2.1, taken from [**?**], which implies a weak converse. In [**?**], Li et al. introduce the Laminar Algorithm for Probabilistic group testing. In Section II-C we propose a refined version of this Laminar Algorithm, based on Hwang's HGBSA [**?**], which is analysed in Section III-E, and shown to imply performance close to optimal in the sense of capacity.

II. Algorithms and existing results

A. Upper bounds on success probability

Firstly [**?**, Theorem 1] can be restated to give the following upper bound on success probability:

Theorem 2.1: Any Probabilistic group testing algorithm using T tests with noiseless measurements has success probability satisfying

$$\mathbb{P}(\text{suc}) \leq \frac{T}{H(\mathbf{U})}.$$

Rephrased in terms of Definition 1.1, this tells us that the weak capacity of noiseless Probabilistic group testing is ≤ 1. The logic is as follows; if the capacity were $1 + 2\epsilon$ for some $\epsilon > 0$, then there would exist a sequence of algorithms with $H(\mathbf{U}^{(i)})/T(i) \geq 1 + \epsilon$ with success probability tending to 1. However, by Theorem 2.1, any such algorithms have $\mathbb{P}(\text{suc}) \leq 1/(1+\epsilon)$, meaning that we have established that a weak converse holds.

Remark 2.2: It remains an open and interesting problem to prove an equivalent of (3) as in [**?**, Theorem 3.1]. That is we hope to find an upper bound on success probability in a form which implies a strong converse, and hence that the strong capacity of Probabilistic group testing is equal to 1.

B. Binary search algorithms

The main contribution of this work is to describe and analyse algorithms that will find the defective items. In brief, we can think of Hwang's HGBSA algorithm as dividing the population $\mathcal{P}$ into search sets $\mathcal{S}$. First, all the items in a search set $\mathcal{S}$ are tested together, using a test set $\mathcal{X}_1 = \mathcal{S}$. If the result is negative ($y_1 = 0$), we can be certain that $\mathcal{S}$ contains no defectives. However, if the result is positive ($y_1 = 1$), $\mathcal{S}$ must contain at least one defective.

If $y_i = 1$, we can be guaranteed to find at least one defective, using the following binary search strategy. We split the set $\mathcal{S}$ in two, and test the 'left-hand' set, say $\mathcal{X}_2$. If $y_2 = 1$, then we know that $\mathcal{X}_2$ contains at least one defective. If $y_2 = 0$, then $\mathcal{X}_2$ contains no defective, so we can deduce that $\mathcal{S} \setminus \mathcal{X}_2$ contains at least one defective. By repeated use of this strategy, we are guaranteed to find a succession of nested sets which contain at least one defective, until $\mathcal{X}_i$ is of size 1, and we have isolated a single defective item.

However this strategy may not find every defective item in $\mathcal{S}$. To be specific, it is possible that at some stage both the left-hand and right-hand sets contain a defective. The Laminar Algorithm of [**?**] essentially deals with this by testing both sets. However, we believe that this is inefficient, since typically both sets will not contain a defective. Nonetheless, the Laminar Algorithm satisfies the following performance guarantees proved in [**?**, Theorem 2]:

Theorem 2.3: The expected number of tests required by the Laminar Algorithm [**?**] is $\leq 2H(\mathbf{U}) + 2\mu$. Under a technical condition (referred to as non-skewedness), the success probability can be bounded by $\mathbb{P}(\text{suc}) \geq 1 - \epsilon$ using $T = (1+\delta)(2^{\Gamma + \log_2 3} + 2)H(\mathbf{U})$ tests, where Γ is defined implicitly in terms of ϵ, and $\delta \geq 2e - 1$.

Ignoring the Γ term, and assuming the non-skewedness condition holds, this implies that (using the methods of [**?**]) $T = 2e(3+2)H(\mathbf{U}) = 10eH(\mathbf{U})$ tests are required to guarantee convergence to 1 of the success probability. In our language, this implies a lower bound of $C \geq 1/(10e) =$

0.0368. Even ignoring the analysis of error probability, the fact that the expected number of tests is $\leq 2H(\mathbf{U}) + 2\mu$ suggests that we cannot hope to achieve $C > 1/2$ using the Laminar Algorithm.

C. Summary of our contribution

Algorithm 1: Algorithm for the non-iid group testing problem

Data: A Set S of $|S| = n$ items, μ of which are actually defective in expectation, a probability vector $\vec{p}^{(n)}$ describing each item's independent probability of being defective, and a cutoff θ
Result: The set of defective items
Discard items with $p_i \leq \theta$
Sort the remaining items into B bins, collecting items together with $p_i \in \left[1/2C^r, 1/2C^{r-1}\right)$ in bin r.
Sort the items in each bin into sets s.t. the (normalised) probability of each set is less than $1/2$.
Test each set in turn
if *The test is positive* **then**
  Arrange the items in the set on a Shannon-Fano/Huffman Tree and search the set for all the defectives it contains
end

The main contribution of our paper is a refined version of the Laminar Algorithm, summarised above, and an analysis resulting in tighter error bounds as formulated in Proposition 3.7 (in terms of expected number of tests) and Theorem 3.9 (in terms of error probabilities). The key ideas are:

1) To partition the population $\mathcal{P}$ into search sets $\mathcal{S}$ containing items which have similar probabilities, expressed through the Bounded Ratio Condition 1. This is discussed in Section III-A, and optimised in the proof of Proposition 3.7.
2) The way in which we deal with sets $\mathcal{S}$ which contain more than one defective, as discussed in Remark 3.2 below. Essentially we do not backtrack after each test by testing both left- and right-hand sets, but only backtrack after each defective is found.
3) To discard items which have probability below a certain threshold, since with high probability none of them will be defective. This is an idea introduced in [**?**] and discussed in Section III-B, with a new bound given in Lemma 3.4.
4) Careful analysis in Section III-D of the properties of search sets $\mathcal{S}$ gives Proposition 3.7, which shows that the expected number of tests required can be expressed as $H(\mathbf{U})$ plus an error term. In Section III-E, we give an analysis of the error probability using Bernstein's inequality, Theorem 3.8, allowing us to prove Theorem 3.9.

D. Wider context: sparse inference problems

Recent work [**?**], [**?**] has shown that many arguments and bounds hold in a common framework of sparse inference which includes group testing and compressive sensing.

Digital communications, audio, images, and text are examples of data sources we can compress. We can do this, because these data sources are sparse: they have fewer degrees of freedom than the space they are defined upon. For example, images have a well known expansion in either the Fourier or Wavelet bases. The text of an English document will only be comprised of words from the English dictionary, and not all the possible strings from the space of strings made up from the characters $\{a, \ldots, z\}$.

Often, once a signal has been acquired it will be compressed. However, the compressive sensing paradigm introduced by [**?**], [**?**] shows that this isn't necessary. In those papers it was shown that a 'compressed' representation of a signal could be obtained from random linear projections of the signal and some other basis (for example White Gaussian Noise). The question remains, given this representation how do we recover the original signal? For real signals, a simple linear programme suffices. Much of the work in this area has been couched in terms of the sparsity of the signal and the various bases the signal can be represented in (see for example [**?**], [**?**]).

III. Analysis and new bounds

A. Searching a set of bounded ratio

Recall that we have a population $\mathcal{P}$ of items to test, each with associated probability of defectiveness p_i. The strategy of the proof is to partition $\mathcal{P}$ into search sets $\mathcal{S}_1, \ldots, \mathcal{S}_G$, each of which contains items which have comparable values of p_i.

Condition 1 (Bounded Ratio Condition): Given $C \geq 1$, say that a set $\mathcal{S}$ satisfies the Bounded Ratio Condition with constant C if

$$\max_{i,j \in \mathcal{S}} \frac{p_j}{p_i} \leq C. \tag{4}$$

(For example clearly if $p_i \equiv p$, any set $\mathcal{S}$ satisfies the condition for any $C \geq 1$).

Lemma 3.1: Consider a set $\mathcal{S}$ satisfying the Bounded Ratio Condition with constant C and write $P_{\mathcal{S}} = \sum_{j \in \mathcal{S}} p_j$. In a Shannon–Fano tree for the probability distribution $\overline{p}_i := p_i / P_{\mathcal{S}}$, each item has length $\ell_i^{(\mathcal{S})}$ bounded by

$$\ell_i^{(\mathcal{S})} \leq \ell_{\max}^{(\mathcal{S})} := \frac{h(\mathcal{S})}{P_{\mathcal{S}}} + \log_2 C + \log_2 P_{\mathcal{S}} + 1, \tag{5}$$

where we write $h(\mathcal{S}) := -\sum_{j \in \mathcal{S}} p_j \log_2 p_j$.

Proof: Under the Bounded Ratio Condition, for any i and j, we know that by taking logs of (4)

$$-\log_2 p_i \leq -\log_2 p_j + \log_2 C.$$

Multiplying by p_j and summing over all $j \in \mathcal{S}$, we obtain that

$$-P_{\mathcal{S}} \log_2 p_i \leq h(\mathcal{S}) + P_{\mathcal{S}} \log_2 C. \tag{6}$$

Now, the Shannon–Fano length of the ith item is

$$\begin{aligned} \ell_i^{(\mathcal{S})} = \lceil -\log_2 \overline{p}_i \rceil &\leq -\log_2 p_i + \log_2 P_{\mathcal{S}} + 1 \qquad (7) \\ &\leq \left(\frac{h(\mathcal{S})}{P_{\mathcal{S}}} + \log_2 C \right) + \log_2 P_{\mathcal{S}} + 1. \end{aligned}$$

and the result follows by (6). ∎

Next we describe our search strategy:

Remark 3.2: Our version of the algorithm will find every defective in a set $\mathcal{S}$. We start as before by testing every item in $\mathcal{S}$ together. If this test is negative, we are done. Otherwise, if it is positive, we can perform binary search as above to find one defective item, say d_1. Now, test every item in $\mathcal{S} \setminus \{d_1\}$ together. If this test is negative, we are done, otherwise we repeat the search step on this smaller set, to find another defective item d_2, then we test $\mathcal{S} \setminus \{d_1, d_2\}$ and so on.

We think of the algorithm as repeatedly searching a binary tree. Clearly, if the tree has depth bounded by ℓ, then the search will take $\leq \ell$ tests to find one defective. In total, if the set contains U defectives, we need to repeat U rounds of searching, plus the final test to guarantee that the set contains no more defectives, so will use $\leq \ell U + 1$ tests.

Lemma 3.3: Consider a search set $\mathcal{S}$ satisfying the Bounded Ratio Condition and write $P_{\mathcal{S}} = \sum_{j \in \mathcal{S}} p_j$. If (independently) item i is defective with probability p_i, we can recover all defective items in the set using $T_{\mathcal{S}}$ tests, where $\mathbb{E}T_{\mathcal{S}} \leq T_{\mathrm{bd}}(\mathcal{S})$ for

$$T_{\mathrm{bd}}(\mathcal{S}) := h(\mathcal{S}) + P_{\mathcal{S}} \log_2 C + P_{\mathcal{S}} \log_2 P_{\mathcal{S}} + P_{\mathcal{S}} + 1. \quad (8)$$

Proof: Using the algorithm of Remark 3.2, laid out on the Shannon-Fano tree constructed in Lemma 3.1, we are guaranteed to find every defective. The number of tests to find one defective thus corresponds to the depth of the tree, which is bounded by $\ell_{\max}^{(\mathcal{S})}$ given in (5).

Recall that we write U_i for the indicator of the event that the ith item is defective, $U_{\mathcal{S}} = \sum_{i \in \mathcal{S}} U_i$ and $l_i^{(\mathcal{S})}$ for the length of the word in the Shannon Fano tree. As discussed in Remark 3.2 this search procedure will take

$$\begin{aligned} T_{\mathcal{S}} &= 1 + \sum_{i \in \mathcal{S}} U_i \ell_i^{(\mathcal{S})} \\ &= \sum_{i \in \mathcal{S}} p_i \ell_i^{(\mathcal{S})} + 1 + \sum_{i \in \mathcal{S}} \ell_i^{(\mathcal{S})} (U_i - p_i) \\ &\leq \sum_{i \in \mathcal{S}} p_i \ell_{\max}^{(\mathcal{S})} + 1 + \sum_{i \in \mathcal{S}} V_i^{(\mathcal{S})} \\ &= P_{\mathcal{S}} \ell_{\max}^{(\mathcal{S})} + 1 + \sum_{i \in S} V_i^{(\mathcal{S})} \\ &\leq T_{\mathrm{bd}}(\mathcal{S}) + \sum_{i \in S} V_i^{(\mathcal{S})} \quad \text{tests.} \qquad (9) \end{aligned}$$

Here we write $V_i^{(\mathcal{S})} = \ell_i^{(\mathcal{S})}(U_i - p_i)$, which has expectation zero, and (9) follows using the expression for $\ell_{\max}^{(\mathcal{S})}$ given in Lemma 3.1. ∎

B. Discarding low probability items

As in [**?**], we use a probability threshold θ, and write $\mathcal{P}^*$ for the population having removed items with $p_i \leq \theta$. If an item lies in $\mathcal{P} \setminus \mathcal{P}^*$ we do not test it, and simply mark it as non-defective. This truncation operation gives an error if and only if some item in $\mathcal{P} \setminus \mathcal{P}^*$ is defective. By the union bound, this truncation operation contributes a total of $\mathbb{P}(\mathcal{P} \setminus \mathcal{P}^* \text{ contains a defective}) \leq \rho := \sum_{i=1}^n p_i \mathbb{I}(p_i \leq \theta)$ to the error probability.

Lemma 3.4: Choosing $\theta(P_e)$ such that

$$-\log_2 \theta(P_e) = \min \left(\log_2 \left(\frac{2n}{P_e} \right), \frac{2H(\mathbf{U})}{P_e} \right) \qquad (10)$$

ensures that

$$\mathbb{P}(\ \mathcal{P} \setminus \mathcal{P}^* \text{ contains a defective}) \leq P_e/2. \qquad (11)$$

Proof: The approach of [**?**] is essentially to bound $\mathbb{I}(p_i \leq \theta) \leq \theta/p_i$ so that $\rho = \sum_{i=1}^n p_i \mathbb{I}(p_i \leq \theta) \leq \sum_{i=1}^n p_i(\theta/p_i) = n\theta$. Hence, choosing a threshold of $\theta = P_e/(2n)$ guarantees the required bound on ρ.

We combine this with another bound, constructed using a different function: $\mathbb{I}(p_i \leq \theta) \leq (-\log_2 p_i)/(-\log_2 \theta)$, so that

$$\rho = \sum_{i=1}^n p_i \mathbb{I}(p_i \leq \theta) \leq \sum_{i=1}^n p_i \left(\frac{-\log_2 p_i}{-\log_2 \theta} \right) \leq \frac{H(\mathbf{U})}{-\log_2 \theta},$$

so we deduce the result. ∎

C. Searching the entire set

Having discarded items with p_i below this probability threshold θ and given bounding ratio C, we create a series of bins. We collect together items with probabilities $p \in [1/2, 1]$ in bin 0, $p \in [1/(2C), 1/2)$ in bin 1, items with probabilities $p \in [1/(2C^2), 1/(2C))$ in bin 2, …, and items with probabilities $p \in [1/(2C^B), 1/(2C^{B-1}))$ in bin B.

The probability threshold θ means that there will be a finite number of such bins, with the index B of the last bin defined by the fact that $1/(2C^B) \leq \theta < 1/(2C^{B-1})$, meaning that $(B-1)\log_2 C < -\log_2(2\theta)$, so

$$B \leq \frac{-\log_2(2\theta)}{\log_2 C} + 1. \qquad (12)$$

We split the items in each bin into search sets $\mathcal{S}_i$, motivated by the following definition:

Definition 3.5: A set of items $\mathcal{S}$ is said to be full if $P_{\mathcal{S}} = \sum_{i \in \mathcal{S}} p_i \geq \frac{1}{2}$.

Our splitting procedure is as follows: we create a list of possible sets $\mathcal{S}_1, \mathcal{S}_2, \ldots$. For i increasing from 0 to B, we place items from bin i into sets $\mathcal{S}_{b_i+1}, \ldots, \mathcal{S}_{b_{i+1}}$, for some b_i, where $b_0 = 0$. Taking the items from bin i, while $\mathcal{S}_{b_i+1}$ is not full (has total probability $< \frac{1}{2}$) we will place items into it. Once enough items have been added to fill $\mathcal{S}_{b_i+1}$, we will proceed in the same way to fill $\mathcal{S}_{b_i+2}$, and so on until all the items in bin i have been assigned to sets $\mathcal{S}_{b_i+1}, \ldots, \mathcal{S}_{b_{i+1}}$, where $\mathcal{S}_{b_{i+1}}$ may remain not full.

Proposition 3.6: This splitting procedure will divide $\mathcal{P}^*$ into search sets $\mathcal{S}_1, \ldots, \mathcal{S}_G$, where the total number of sets is

$$G \leq 2\mu + B \leq 2\mu + \left(\frac{-\log_2(2\theta)}{\log_2 C} + 1 \right).$$

Each set $\mathcal{S}_j$ satisfies the Bounded Ratio Condition and has total probability $P_j := P_{\mathcal{S}_j} \leq 1$.

Proof: First, note that the items from bin 0 each lie in a set $\mathcal{S}$ on their own. These sets will be full, trivially satisfy the Bounded Ratio Condition 1 with constant C, and have probability satisfying $P_j \leq 1$. For each of bins $1, \ldots, B$:

1) For each bin i, it is possible that the last set $\mathcal{S}_{b_i+1}$ will not be full, but every other set corresponding to that bin will be full. Hence, there are no more than B sets which are not full.
2) For each resulting set $\mathcal{S}_j$, the total probability $P_j \leq 1$ (since just before we add the final item, $\mathcal{S}_j$ is not full, so at that stage has total probability $\leq 1/2$, and each element in bins $1, \ldots, B$ has probability $\leq 1/2$).
3) Since each set $\mathcal{S}_j$ contains items taken from the same bin, it will satisfy the Bounded Ratio Condition with constant C.

Note that the number of full sets is $\leq 2\mu$, since

$$\mu = \sum_{i \in \mathcal{P}} p_i \geq \sum_{i \in \mathcal{P}^*} p_i = \sum_{j=1}^{G} P_j \geq \sum_{j:\ \mathcal{S}_j \text{ full}} P_j \geq |\mathcal{S}_j \text{ full}| \, \frac{1}{2}. \tag{13}$$

Since, as discussed in point 1) above, the total number of sets is bounded by the number of full sets plus B, the result follows using Equation (12). ∎

D. Bounding the expected number of tests

We allow the algorithm to work until all defectives in $\mathcal{P}^*$ are found, and write T for the (random) number of tests this takes.

Proposition 3.7: Given a population $\mathcal{P}$ where (independently) item i is defective with probability p_i, we recover all defective items in $\mathcal{P}^*$ in T tests with $\mathbb{E}T \leq T_{\rm bd}$, where

$$T_{\rm bd} := (H(\mathbf{U}) + 3\mu + 1) + 2\sqrt{\mu\,(-\log_2(2\theta))}. \tag{14}$$

Proof: Given a value of C, Proposition 3.6 shows that our splitting procedure divides $\mathcal{P}^*$ into G sets $\mathcal{S}_1, \ldots, \mathcal{S}_G$, such that each set $\mathcal{S}_j$ satisfies the Bounded Ratio Condition with constant C and has total probability $P_j \leq 1$. Using the notation of Lemma 3.3, $T = \sum_{j=1}^{G} T_{\mathcal{S}_j}$, where $\mathbb{E}T_{\mathcal{S}_j} \leq T_{\rm bd}(\mathcal{S}_j)$.

Adding this bound over the different sets, since $P_j \leq 1$ means that $P_j \log_2 P_j \leq 0$, we obtain

$$\begin{aligned}
&\sum_{j=1}^{G} T_{\rm bd}(\mathcal{S}_j) \\
&\quad\leq \sum_{j=1}^{G} (h(\mathcal{S}_j) + P_j(\log_2 C + 1) + 1) \\
&\quad= \sum_{j \in \mathcal{P}^*} -p_j \log_2 p_j + \mu(\log_2 C + 1) + G \\
&\quad\leq \sum_{j \in \mathcal{P}^*} h(p_j) + 3\mu + 1 + \left(\frac{-\log_2(2\theta)}{\log_2 C} + \mu \log_2 C \right) \\
&\quad\leq (H(\mathbf{U}) + 3\mu + 1) + \left(\frac{-\log_2(2\theta)}{\log_2 C} + \mu \log_2 C \right). \qquad (15)
\end{aligned}$$

This follows by the bound on G in Proposition 3.6, as well as the fact that $0 \leq p_j \leq 1$ means that for any i, $-p_j \log_2 p_j = (1-p_j)\log_2(1-p_j) + h(p_j) \leq h(p_j)$.

Finally, we choose $C > 1$ to optimize the second bracketed term in Equation (15). Differentiation shows that the optimal C satisfies $\log_2 C = \sqrt{-\log_2(2\theta)/\mu}$, meaning that the bracketed term

$$\frac{-\log_2(2\theta)}{\log_2 C} + \mu \log_2 C = 2\sqrt{\mu\,(-\log_2(2\theta))},$$

and the result follows. ∎

E. Controlling the error probabilities

Although Section III-D proves that $\mathbb{E}T \leq T_{\rm bd}$, to bound the capacity, we need to prove that with high probability T is not significantly larger than $T_{\rm bd}$. This can be done using Bernstein's inequality (see for example Theorem 2.8 of [**?**]):

Theorem 3.8 (Bernstein): For zero-mean random variables V_i which are uniformly bounded by $|V_i| \leq M$, if we write $L := \sum_{j=1}^{n} \mathbb{E}V_j^2$ then

$$\mathbb{P}\left(\sum_{j=1}^{n} V_j \geq t \right) \leq \exp\left(-\frac{t^2}{4L} \right), \text{ for any } 0 \leq t \leq \tfrac{L}{M}. \tag{16}$$

We deduce the following result:

Theorem 3.9: Write $L = \sum_{j \in \mathcal{P}^*} l_j^2 p_j (1-p_j)$, $M = -\log_2 \theta + 1$ and $\psi = (L/(4M^2))^{-1/3}$. Define

$$T_{\rm nec} = T_{\rm bd} + \psi H(\mathbf{U}), \tag{17}$$

where $T_{\rm bd}$ is given in (14).

1) If we terminate our group testing algorithm after $T_{\rm nec}$ tests, the success probability

$$\mathbb{P}(\text{suc}) \geq 1 - \frac{1}{2}\sqrt{\frac{\mu}{H(\mathbf{U})}} - \exp\left(-\left(\frac{L}{4M^2} \right)^{1/3} \right). \tag{18}$$

2) Hence in any regime where $\mu \to \infty$ with $\mu/H(\mathbf{U}) \to 0$ and $L/M^2 \to \infty$, our group testing algorithm has (a) $\liminf H(\mathbf{U})/T_{\rm nec} \geq 1/(1+\epsilon)$ for any ϵ and (b) $\mathbb{P}(\text{suc}) \to 1$, so the capacity $C = 1$.

Proof: We first prove the success probability bound (18). Recall that our algorithm searches the reduced population set $\mathcal{P}^*$ for defectives. This gives two error events – either there are defective items in the set $\mathcal{P}\setminus\mathcal{P}^*$, or the algorithm does not find all the defectives in $\mathcal{P}^*$ using $T_{\rm nec}$ tests. We consider them separately, and control the probability of either happening using the union bound.

Writing $H = H(\mathbf{U})$ for brevity and choosing $P_e = \sqrt{\mu/H}$ ensures that (by Lemma 3.4) the first event has probability $\leq P_e/2$, contributing $\frac{1}{2}\sqrt{\mu/H(\mathbf{U})}$ to (18).

Our analysis of the second error event is based on the random term from Equation (9), which we previously averaged over but now wish to bound. There will be an error if $T_{\rm nec} \leq T$, or (rearranging) if

$$\psi H \leq T - T_{\rm bd} \leq \sum_{j=1}^{G} \left(T_{\mathcal{S}_j} - T_{\rm bd}(\mathcal{S}_j)\right) = \sum_{i\in\mathcal{P}^*} V_i.$$

For brevity, for $i \in \mathcal{S}$, we write $V_i = V_i^{(\mathcal{S})} = \ell_i^{(\mathcal{S})}(U_i - p_i)$ and $\ell_i = \ell_i^{(\mathcal{S})}$, where V_i has expectation zero.

We have discarded elements with probability below θ, as given by (10), and by design all the sets $\mathcal{S}$ have total probability $P_{\mathcal{S}} \leq 1$. Using (7) we know that the V_i are bounded by

$$|V_i| \leq \ell_i \leq -\log_2 p_i + \log_2 P_{\mathcal{S}} + 1 \leq -\log_2\theta + 1. \quad (19)$$

Hence, the conditions of Bernstein's inequality, Theorem 3.8, are satisfied. Observe that since all $l_j \leq M$,

$$\frac{L}{HM} = \frac{\sum_{j\in\mathcal{P}^*} l_j^2 p_j(1-p_j)}{HM} \leq \frac{\sum_{j\in\mathcal{P}^*} l_j p_j}{H} \leq 1.$$

Hence Theorem 3.8 gives that

$$\begin{aligned}\mathbb{P}\left(\sum_{j\in\mathcal{P}^*} V_j \geq \psi H\right) &\leq \mathbb{P}\left(\sum_{j\in\mathcal{P}^*} V_j \geq \psi L/M\right)\\ &\leq \exp\left(-\frac{L\psi^2}{4M^2}\right)\\ &= \exp\left(-\left(\frac{L}{4M^2}\right)^{1/3}\right).\end{aligned}$$

Using the union bound, the probability bound (18) follows.

We next consider the capacity bound of 2). Since $-\log_2\theta \leq 2H/P_e$, using (14) and (17)

$$\begin{aligned}\frac{T_{\rm nec}}{H} &= \frac{T_{\rm bd}}{H} + \psi\\ &= 1 + 3\frac{\mu}{H} + \frac{1}{H} + 2\sqrt{\frac{\mu}{HP_e}} + \psi\\ &= 1 + 3\frac{\mu}{H} + \frac{1}{H} + 2\left(\frac{\mu}{H}\right)^{1/4} + \psi, \quad (20)\end{aligned}$$

which in our regime of interest is $\leq 1+\epsilon$ in the limit. ∎

Proof of Corollary 1.5: In the case where all p are identical, $\mu = Np$, $H = Np(-\log p)$, so $\mu/H = 1/(-\log p) \to 0$. Similarly, $L = Np(-\log_2 p)^2$ and $M = (-\log_2 p)$ so that $L/M^2 = Np \to \infty$ as required. ∎

IV. Results

The performance of the Algorithm 1 (in terms of the sample complexity) was analysed by simulating 500 items, with a mean number of defectives equal to 8. I.e. $N = 500$ and $\mu^{(N)} = 8$.

The probability distribution $\mathbf{p}$ was generated by a Dirichlet distribution with parameter α. This produces output which can be made more or less uniform, as opposed to simply choosing a set of random numbers and normalise by the sum. Consider the case of two random numbers, (x, y), distributed uniformly on the square $[0, 1]^2$. Normalising by the sum $(x + y)$ projects the point (x, y) onto the line $x + y = 1$ and so favours points closer to $(0.5, 0.5)$ than the endpoints. The Dirichlet distribution avoids this by generating points directly on the simplex.

We then chose values of the cutoff parameter θ from 0.0001 to 0.01, and for each θ_i simulated the algorithm 1000 times. We plot the empirical distribution of tests, varying theta as well as the uniformity/concentration of the probability distribution (via the parameter α of the Dirichlet distribution). We also plot, the theoretical lower and upper bounds on the number of Tests required for successful recovery alongside the empirical number tests (all as a function of θ).

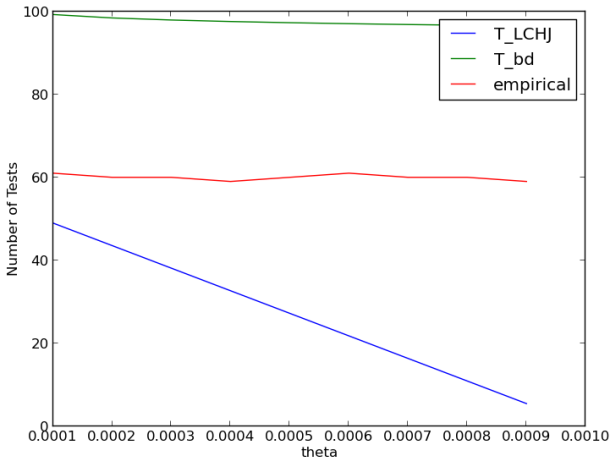

Fig. 1. Theoretical lower and upper bounds and empirical Test frequencies as functions of θ

Note that the Upper bound is not optimal and there still is some room for improvement. Note also that the lower bound degrades with θ_i. The lower bound (T_{LCHJ}) was generated according to theorem (2.1).

Figures (2) and (3) show that the performance is relatively insensitive to the cut-off θ, and more sensitive to the uniformity (or otherwise) of the probability distribution $\mathbf{p}$. Heuristically, this is for because distributions which are highly concentrated on a few items algorithms can make substantial savings on the testing budget by testing those highly likely items first (which is captured in the bin structure of the above algorithm).

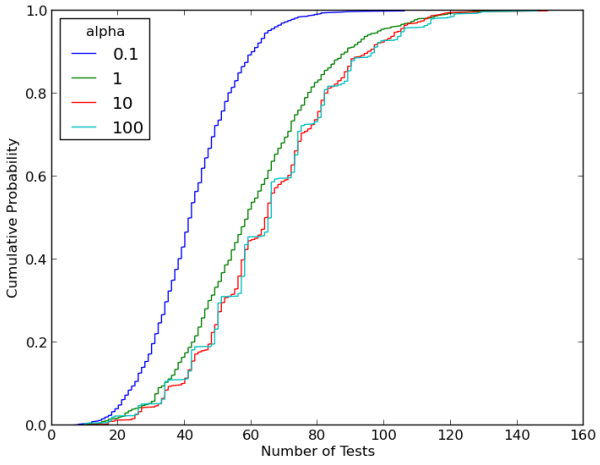

Fig. 2. Cumulative distribution curves of the modified Hwang algorithm with fixed $\theta = 0.0001$ and α varying

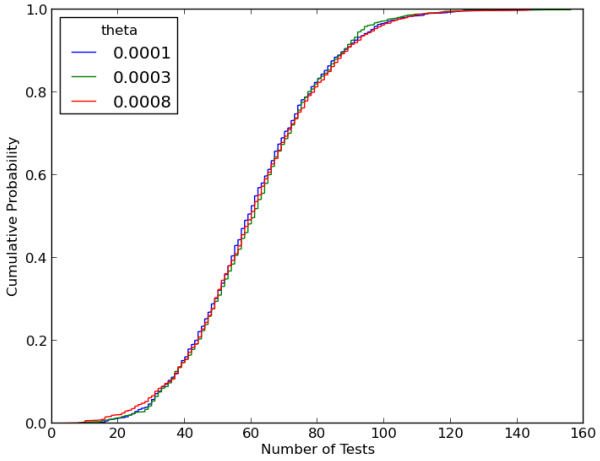

Fig. 3. Cumulative distribution curves for fixed $\alpha = 1$ and varying θ

The insensitivity to the cutoff θ is due to items below θ being overwhelmingly unlikely to be defective - which for small θ means that few items (relative to the size of the problem) get discarded.

V. Discussion

We have introduced and analysed an algorithm for Probabilistic group testing which uses 'just over' $H(\mathbf{U})$ tests to recover all the defectives with high probability. Combined with a weak converse taken from [**?**], this allows us to deduce that the weak capacity of Probabilistic group testing is $C = 1$. These results are illustrated by simulation.

For simplicity, this work has concentrated on establishing a bound T_{bd} in (14) which has leading term $H(\mathbf{U})$, and not on tightening bounds on the coefficient of μ in (14). For completeness, we mention that this coefficient can be reduced from 3, under a simple further condition:

Remark 5.1: For some $c \leq 1/2$, we assume that all the $p_i \leq c$, and we alter the definition of 'fullness' to assume that a set is full if it has total probability less than α. In this case, the term $P_{\mathcal{S}} \log_2 P_{\mathcal{S}}$ in (8) becomes $P_{\mathcal{S}} \log_2(\alpha + c)$, the bound in (13) becomes μ/α, and since $((1-p)\log_2(1-p))/p$ is decreasing in p, we can add a term $(1-c)\log_2(1-c)$ to (15). Overall, the coefficient of μ becomes $f(a,c) := \log_2(\alpha + c) + 1 + 1/\alpha + (1-c)\log_2(1-c)$, which we can optimize over α. For example, if $c = 1/4$, taking $\alpha = 0.88824$, we obtain $f(a,c) = 2.00135$.

It remains of interest to tighten the upper bound of Theorem 2.1, in order prove a strong converse, and hence confirm that the strong capacity is also equal to 1.

In future work, we hope to explore more realistic models of defectivity, such as those where the defectivity of U_i are not necessarily independent, for example by imposing a Markov neighbourhood structure.

Acknowledgments

This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/I028153/1]; Ofcom; and the University of Bristol. The authors would particularly like to thank Gary Clemo of Ofcom for useful discussions.