Electronic Colloquium on Computational Complexity, Report No. 53 (2018)

Lower Bound for Non-Adaptive Estimate the Number of Defective Items

Nader H. Bshouty
Dept. of Computer Science
Technion, Haifa, 32000
bshouty@cs.technion.ac.il

Abstract. We prove that to estimate within a constant factor the number of defective items in a non-adaptive group testing algorithm we need at least $\tilde{\Omega}((\log n)(\log(1/\delta)))$ tests. This solves the open problem posed by Damaschke and Sheikh Muhammad in [9, 10].

1 Introduction

Let X be a set of *items* that contains *defective items* $I \subseteq X$. In Group testing, we *test (query)* a subset $Q \subset X$ of items and the answer to the query is 1 if Q contains at least one defective item, i.e., $Q \cap I \neq \emptyset$, and 0 otherwise. Group testing was originally introduced as a potential approach to the economical mass blood testing, [11]. However it has been proven to be applicable in a variety of problems, including DNA library screening, [21], quality control in product testing, [25], searching files in storage systems, [17], sequential screening of experimental variables, [19], efficient contention resolution algorithms for multiple-access communication, [17, 29], data compression, [15], and computation in the data stream model, [8]. See a brief history and other applications in [7, 12, 13, 16, 20, 21] and references therein.

Estimating the number of defective items to within a constant factor λ is the problem of finding an integer D that satisfies $|I| \leq D \leq \lambda|I|$. This problem is extensively used in biological and medical applications [3, 26]. It is used to estimate the proportion of organisms capable of transmitting the aster-yellows virus in a natural population of leafhoppers [27], estimating the infection rate of yellow-fever virus in a mosquito population [28] and estimating the prevalence of a rare disease using grouped samples to preserve individual anonymity [18].

In *adaptive algorithms*, the queries can depend on the answers to the previous ones. In the *non-adaptive algorithms* they are independent of the previous one and; therefore, one can ask all the queries in one parallel step. In many applications in group testing non-adaptive algorithms are most desirable.

Estimating the number of defective items to within a constant factor with an *adaptive* deterministic, Las Vegas and Monte Carlo algorithms is studied in [2, 6, 9, 10, 14, 23]. For $|X| = n$ items and $|I| = d$ defective items the bounds are $\Theta(d \log(n/d))$ queries for Las Vegas and Deterministic algorithms and $\Theta(\log\log d + \log(1/\delta))$ queries for Monte Carlo algorithm [2, 14]. There are also polynomial time algorithms that achieve such bounds [2, 14].

ISSN 1433-8092

In this paper we study this problem in the non-adaptive setting. We first show that any deterministic and Las Vegas algorithm must ask at least $\Omega(n)$ queries. For randomized algorithm with any *constant* failure probability δ, Damaschke and Sheikh Muhammad give in [10] a non-adaptive randomized algorithm that asks $O(\log n)$ queries and with probability at least $1-\delta$ returns an integer D such that $D \geq d$ and $\mathbf{E}[D] = O(d)$. In this paper we give a polynomial time Monte Carlo algorithm that asks $O(\log(1/\delta)\log n)$ queries and with probability at least $1-\delta$ estimates the number of defective items to within a constant factor. They then prove in [9] the lower bound $\Omega(\log n)$ queries, but only for algorithms that chooses each item in each query randomly and independently with some fixed probability. They conjecture that $\Omega(\log n)$ queries are needed for *any* randomized algorithm with constant failure probability. In this paper we prove this conjecture. We give two results. The first result shows that for any $\delta > 1/poly(n)$, any non-adaptive randomized algorithm that with probability at least $1-\delta$ estimates the number of defective items to within a constant factor must ask at least

$$s = \Omega\left(\frac{\log \frac{1}{\delta}\log n}{\log\log n + \log\log \frac{1}{\delta}}\right) = \tilde{\Omega}\left(\log\frac{1}{\delta}\log n\right)$$

queries. The second result shows that for any fixed δ and large enough n any non-adaptive randomized algorithm that with probability at least $1-\delta$ estimates the number of defective items to within a constant factor must ask at least

$$\Omega\left(\frac{\log \frac{1}{\delta}\log n}{(c\log^* n)^{(\log^* n)+1}}\right)$$

queries. Here $\log^* n$ is equal to the minimum k such that $\log\log \overset{k}{\ldots} \log n < 2$. In particular, the lower bound is

$$\Omega\left(\frac{\log \frac{1}{\delta}\log n}{\log\log \overset{k}{\ldots} \log n}\right)$$

for any constant k.

This paper is organised as follows: In Section 2 we give some preliminary results. In Section 3 we give the proof of the above two lower bounds and the lower bound $\Omega(n)$ for the deterministic algorithm. Then in Section 4 we give the upper bound. The technique for the upper bound is standard and is given for completeness.

2 Preliminary Results

In this section we give some definitions and then prove some preliminary results.

We will consider the set of *items* $X = [n] = \{1, 2, \ldots, n\}$ and the set of *defective items* $I \subseteq X$. The algorithm knows n and has an access to an oracle $\mathcal{O}_I$. The algorithm can ask the oracle $\mathcal{O}_I$ a *query* $Q \subset X$ and the oracle answers

$\mathcal{O}_I(Q) := 1$ if $Q \cap I \neq \emptyset$ and $\mathcal{O}_I(Q) := 0$ otherwise. We say that algorithm A λ-*estimates* the number of defective items if for every $I \subseteq X$ it runs in polynomial time in n, asks queries to the oracle $\mathcal{O}_I$ and returns an integer D such that $|I| \leq D \leq \lambda |I|$. If λ is constant then we say that the algorithm *estimates the number of defective items to within a constant factor* . Our goal is to find such an algorithm that asks minimum number of queries.

We now prove some preliminary results. The next three results are proved for adaptive algorithms and therefore they are also valid for non-adaptive algorithms.

The following lemma follows by elementary information theory [1].

Lemma 1. *Let $d < n$ be a fixed integer. Let A be an adaptive Monte Carlo randomized algorithm that takes the number of items n as an input and, if the number of defective items is $|I| = d$, outputs the defective items I. Then A must ask at least $d \log(n/d)$ queries.*

We denote by $\mathcal{Q}(A, I)$ the set of all queries that A asks when the oracle is $\mathcal{O}_I$. Obviously, in non-adaptive algorithms $\mathcal{Q}(A, I)$ depends only on A.

Lemma 2. *Let A be a deterministic adaptive algorithm that asks queries and satisfies the following: For every two disjoint sets $I, J \subseteq X$ of size d there is a query $Q \in \mathcal{Q}(A, I)$ such that $Q \cap I = \emptyset$ and $Q \cap J \neq \emptyset$. Then A asks at least $d \log(n/d) - 2d + 1$ queries.*

Proof. We change algorithm A to a deterministic algorithm B that asks at most $2d - 1$ more queries and if the number of defective items is $|I| = d$ then it finds the defective items I. Assuming this is true, if algorithm A asks q queries then algorithm B asks at most $q + (2d - 1)$ queries. Then by Lemma 1, $q + 2d - 1 \geq d \log(n/d)$ and the result follows.

The following is algorithm B

1. Run A. Define $Y = X$ and for every query Q (to $\mathcal{O}_I$) that A asks that has answer 0, eliminate all the elements of Q from Y. Notice here that when the answer is 0, all the items in Q are not defective.
2. Exhaustively asks the queries $\{u\}$ for all $u \in Y$ and if the answer is 0 remove u from Y.
3. Output Y.

Let Y' be the set Y after step 1 and let Y'' be the output of the algorithm. Since algorithm B only removes items that are not defective, it is clear that $I \subseteq Y'$. Now in step 2, algorithm B removes from Y' all the elements that are not in I and therefore $I = Y''$. This proves the correctness of the algorithm.

We now show that, when A halts in step 1, the number of elements that remain in Y' is at most $2d - 1$. Suppose, for the contrary, this is not true. That is, Y' contains at least $2d$ items. Let $J \subseteq Y' \backslash I$ be a set of size d. There is a query $Q' \in \mathcal{Q}(A, I)$ that is asked by A such that $Q' \cap I = \emptyset$ and $Q' \cap J \neq \emptyset$. Then the answer to this query is 0 and $Y' \cap Q' \neq \emptyset$. So there is $j \in Y'$ that is not eliminated by the query $Q' \in \mathcal{Q}(A, I)$ which has an answer 0. A contradiction. □

For an algorithm A that asks queries we denote by $A(I)$ the output of A when it runs with the oracle $\mathcal{O}_I$. When the algorithm is randomized then we write $A(s, I)$ where s is the random seed of the algorithm.

Lemma 3. *Let $d \leq n/64$ be an integer and $\delta \geq (d/n)^d$. Let A be an adaptive randomized Monte Carlo algorithm that satisfies:*

1. *For a random uniform set $I \subseteq X$ of size d, with probability at least $1 - \delta$, $A(s, I) = 0$.*
2. *For a random uniform set $I \subseteq X$ of size $2d$, with probability at least $1 - \delta$, $A(s, I) = 1$.*

Then A must ask at least

$$\frac{1}{4} \log \frac{1}{\delta} - \frac{1}{4}$$

queries.

In particular, for $\delta < 1/16$, if A asks $(1/16)\log(1/\delta)$ queries then its failure probability is at least 4δ.

When $\delta \leq (d/n)^d$ then the the number of queries is $\Theta(d \log(n/d))$.

Proof. Let M be the query complexity of $A(s, I)$. For every two sets I_1 and I_2 of size d define a random variable $X_A(s, I_1, I_2) = 0$ if $A(s, I_1) = 0$ and $A(s, I_1 \cup I_2) = 1$, and $X_A(s, I_1, I_2) = 1$ otherwise. Then for random uniform disjoint sets I_1 and I_2 of size d we have

$$\mathbf{E}_s[\mathbf{E}_{I_1,I_2}[X_A(s, I_1, I_2)]] = \mathbf{E}_{I_1,I_2,s}[X_A(s, I_1, I_2)] \leq 2\delta.$$

Therefore, there is a seed s_0 such that $\mathbf{E}_{I_1,I_2}[X_A(s_0, I_1, I_2)] \leq 2\delta$.

Consider q random, uniform and independent permutations $\phi'_i : [n] \to [n]$, $i = 1, \ldots, q$ where

$$q = \frac{2d \log \frac{en}{d}}{\log \frac{1}{2\delta}}.$$

Notice here that q can be chosen to be an integer only when $\delta > (d/n)^d$. It is easy to see that $(\phi'_i I_1, \phi'_i I_2)$, $i = 1, \ldots, q$ are random, uniform and independent. Therefore

$$\mathbf{E}_{\phi'_1,\ldots,\phi'_q}\left[\mathbf{E}_{I_1,I_2}\left[\prod_{i=1}^{q} X_A(s_0, \phi'_i I_1, \phi'_i I_2)\right]\right] \leq (2\delta)^q.$$

This implies that there are $\phi_1, \ldots, \phi_q$ such that

$$\mathbf{E}_{I_1,I_2}\left[\prod_{i=1}^{q} X_A(s_0, \phi_i I_1, \phi_i I_2)\right] \leq (2\delta)^q.$$

Define an algorithm $A_{\phi_i}(s_0, I)$ that runs $A(s_0, I)$ and for each query Q in A it asks the query $\phi_i(Q) = \{\phi_i(x) | x \in Q\}$. Then $X_A(s_0, \phi_i I_1, \phi_i I_2) = X_{A_{\phi_i}}(s_0, I_1, I_2)$ and

$$\mathbf{E}_{I_1,I_2}\left[\prod_{i=1}^{q} X_{A_{\phi_i}}(s_0, I_1, I_2)\right] \leq (2\delta)^q.$$

Thus

$$\sum_{I_1,I_2}\prod_{i=1}^{q} X_{A_{\phi_i}}(s_0,I_1,I_2) \le (2\delta)^q \binom{n}{d\ \ d\ \ n-2d} \le (2\delta)^q \left(\frac{en}{d}\right)^{2d} < 1.$$

Since $\prod_{i=1}^{q} X_{A_{\phi_i}}(s_0,I_1,I_2) \in \{0,1\}$ we have $\prod_{i=1}^{q} X_{A_{\phi_i}}(s_0,I_1,I_2) = 0$ for every two disjoint set I_1 and I_2 of size d. This implies that for every disjoint sets I_1 and I_2 of size d there is ϕ_i such that $X_{A_{\phi_i}}(s_0,I_1,I_2) = 0$. Therefore, For every two disjoint sets I_1 and I_2 of size d there is ϕ_i such that $A_{\phi_i}(s_0,I_1) = 0$ and $A_{\phi_i}(s_0,I_1 \cup I_2) = 1$. Since $A_{\phi_i}(s_0,I_1)$ is deterministic algorithm, this implies that there is a query Q in A_{ϕ_i} where $Q \cap I_1 = \emptyset$ and $Q \cap (I_1 \cup I_2) \neq \emptyset$. Otherwise, $A_{\phi_i}(s_0,I_1) = A_{\phi_i}(s_0,I_1 \cup I_2)$. Let B be the algorithm that runs all A_{ϕ_i}. By Lemma 3, the query complexity of B is at least $d\log(n/d) - 2d + 1$ and therefore $Mq \ge d\log(n/d) - 2d + 1$ and

$$M \ge \frac{(d\log\frac{n}{d} - 2d + 1)\log(1/2\delta)}{2d\log\frac{en}{d}} \ge \frac{1}{4}\log\frac{1}{2\delta}.$$

This proves the case when $\delta \ge (d/n)^d$.

Now when $\delta < (d/n)^d$ then the above lower bound is $\Omega(d\log(n/d))$ (take $\delta = (d/n)^d$) and the upper bound follows from the algorithm that finds the defective items and asks $O(d\log(n/d))$ queries, [4, 5, 24]. □

We now prove two results that will be used for the lower bound

Lemma 4. *Let N' be a finite set of elements and s be an integer. Let S be a probability space of s-tuples $W = (w_1, w_2, \ldots, w_s) \in N'^s$. Let $N \subseteq N'$ and $N = N_1 \cup N_2 \cup \cdots \cup N_r$ be a partition of N to r disjoint sets. There is i_0 such that for a random $W \in S$, the probability that at least k of the elements (coordinates) of W are in N_{i_0}, is at most $s/(kr)$. Equivalently, there is i_0 such that with probability at least $1 - s/(kr)$, the number of elements in W that are in N_{i_0} is at most k.*

Proof. Define the random variables X_i, $i = 1, \ldots, r$, where $X_i(W) = 1$ if at least k of the elements of W are in N_i and 0 otherwise. Obviously, $k(X_1 + \ldots + X_r) \le s$ and therefore

$$\mathbf{E}[X_1] + \cdots + \mathbf{E}[X_r] = \mathbf{E}[X_1 + \cdots + X_r] \le \frac{s}{k}.$$

Therefore there is i_0 such that $\mathbf{Pr}[X_{i_0} = 1] = \mathbf{E}[X_{i_0}] \le s/(kr)$. □

Lemma 5. *Let $X' \subseteq X = [n]$. Let D be the probability space of random uniform subsets $I \subset X'$ of size d and D' be the probability space of random uniform and independent d elements $I = \{x_1, \ldots, x_d\} \subseteq X'$. Let A be any event in D and D'. Let B be the event that $I \in D'$ has size d, i.e., $x_1, \ldots, x_d$ are distinct. Then*

$$\mathbf{Pr}_{D'}[A] + \mathbf{Pr}_{D'}[\bar{B}] \ge \mathbf{Pr}_{D}[A] \ge \mathbf{Pr}_{D'}[A] - \mathbf{Pr}_{D'}[\bar{B}].$$

Proof. Since

$$\begin{aligned}\mathbf{Pr}_{D'}[A] &= \mathbf{Pr}_{D'}[A|B]\mathbf{Pr}_{D'}[B] + \mathbf{Pr}_{D'}[A|\bar{B}]\mathbf{Pr}_{D'}[\bar{B}] \\ &\leq \mathbf{Pr}_{D'}[A|B] + \mathbf{Pr}_{D'}[\bar{B}] = \mathbf{Pr}_{D}[A] + \mathbf{Pr}_{D'}[\bar{B}],\end{aligned}$$

Therefore $\mathbf{Pr}_D[A] \geq \mathbf{Pr}_{D'}[A] - \mathbf{Pr}_{D'}[\bar{B}]$. In the same way we have $\mathbf{Pr}_D[\bar{A}] \geq \mathbf{Pr}_{D'}[\bar{A}] - \mathbf{Pr}_{D'}[\bar{B}]$ which implies the left-hand side inequality. □

3 Lower Bound

In this section we prove two lower bound for the number of queries in any non-adaptive randomized algorithm that λ-estimates the number of defective items. We give the proof for $\lambda = 1.5$. The proof for any other constant is similar. We then prove the lower bound $\Omega(n)$ for any deterministic algorithm.

3.1 Lower Bound for Randomized Algorithm

We first prove

Theorem 1. *Let $\delta > 1/poly(n)$. Any non-adaptive Monte Carlo randomized algorithm that with probability at least $1-\delta$, 1.5-estimates the number of defective items must ask at least*

$$s = \Omega\left(\frac{\log \frac{1}{\delta} \log n}{\log\log n + \log\log \frac{1}{\delta}}\right)$$

queries.

In particular, when $\delta = 1/poly(n)$ then

$$s = \Omega\left(\frac{\log^2 n}{\log\log n}\right).$$

Proof. Let c be a large enough constant. Suppose, for the contrary, there is a non-adaptive Monte Carlo algorithm $A(s, I)$ that chooses a random sequence of queries $M := Q_1, \ldots, Q_s \subseteq X = [n]$ from some probability space where $s = \Delta/(c \log \Delta)$ and $\Delta = (\log n)(\log(1/\delta))$, asks queries to $\mathcal{O}_I$ and with probability at least $1 - \delta$, 1.5-estimates the number of defective items $|I|$. For $r = \log n/(16 \log \Delta)$ let $N_i = [n/\Delta^{4i+4}, n/\Delta^{4i}]$, $i = 0, 1, \ldots, r-1$, be a partition of $N = [n^{3/4}, n]$. By Lemma 4, for $k = (1/16)\log(1/\delta)$ and the s-tuple $W = (|Q_1|, \ldots, |Q_s|)$, there is i_0 such that, with probability at least

$$1 - \frac{s}{kr} = 1 - \frac{256}{c} \geq \frac{15}{16}$$

the number of queries Q in M where $|Q| \in N_{i_0}$ is at most k. Let C the event that the number of queries Q in M where $|Q| \in N_{i_0}$ is at most k. Then

$$\mathbf{Pr}[\bar{C}] \leq \frac{1}{16}.$$

Let $d' = \Delta^{4i_0+2}$. For a random uniform set $I \subset X$ of size $d = d'$, with probability at least $1-\delta$, $A(s,I)$ returns an integer in the interval $[d', 1.5d']$. For a random uniform set $I \subset X$ of size $d = 2d'$, with probability at least $1-\delta$, $A(s,I)$ returns an integer in the interval $[2d', 3d']$. Since both intervals are disjoint, algorithm A can distinguish between defective sets of size d' and $2d'$, with probability at least $1-\delta$. We have constructed an algorithm, call it A', that satisfies the conditions in Lemma 3. The probability that A' fails is at most δ.

Let D, D' and $\{x_1, \ldots, x_d\}$ be as in Lemma 5. Here $d \in \{d', 2d'\}$. Let B be the event that $x_1, \ldots, x_d$ are distinct. Since $i_0 \leq r$ we have $d < n^{1/4}$ and therefore

$$\mathbf{Pr}_{D'}[\bar{B}] = 1 - \prod_{i=1}^{d-1}\left(1 - \frac{i}{n}\right) \leq \frac{d(d-1)}{2n} \leq \frac{1}{n^{1/2}} \leq \frac{1}{16}.$$

Now partition the queries in M to three sets of queries $M_1 \cup M_2 \cup M_3$ where M_1 are the queries that contains at most n/Δ^{4i_0+4} items, M_2 are the queries that contains at least n/Δ^{4i_0} items and $M_3 = M \backslash (M_1 \cup M_2)$, i.e., M_3 are the queries Q that satisfies $|Q| \in N_{i_0}$. Let $A_1(I)$ be the event that for $I \subseteq X$ all the queries in M_1 give answer 0. Then

$$\begin{aligned}
\mathbf{Pr}_{D'}[\bar{A}_1] &= \mathbf{Pr}[(\exists Q \in M_1) Q \cap I \neq \emptyset] \\
&\leq s\mathbf{Pr}[Q \cap I \neq \emptyset | Q \in M_1] \\
&= s(1 - \mathbf{Pr}[Q \cap I = \emptyset | Q \in M_1]) \\
&\leq s\left(1 - \left(1 - \frac{1}{\Delta^{4i_0+4}}\right)^d\right) \\
&\leq \frac{sd}{\Delta^{4i_0+4}} = \frac{2}{c\Delta \log \Delta} \leq \frac{1}{16}.
\end{aligned} \tag{1}$$

Then by Lemma 5, $\mathbf{Pr}_D[\bar{A}_1] \leq 2/16$. Let $A_2(I)$ be the event that for $I \subseteq X$ all the queries in M_2 give answer 1. Then

$$\begin{aligned}
\mathbf{Pr}_{D'}[\bar{A}_2] &= \mathbf{Pr}[(\exists Q \in M_2) Q \cap I = \emptyset] \\
&\leq s\mathbf{Pr}[Q \cap I = \emptyset | Q \in M_2] \\
&\leq s\left(1 - \frac{1}{\Delta^{4i_0}}\right)^d \\
&\leq se^{-\frac{d}{\Delta^{4i_0}}} = \frac{\Delta}{ce^{\Delta^2} \log \Delta} \leq \frac{1}{16}.
\end{aligned}$$

Thus, by Lemma 5, $\mathbf{Pr}_D[\bar{A}_2] \leq 2/16$.

Now

$$\mathbf{Pr}[A' \text{ fails}] \geq \mathbf{Pr}[A_1 \wedge A_2 \wedge C] \cdot \mathbf{Pr}[A' \text{ fails} \mid A_1 \wedge A_2 \wedge C]$$

When events A_1 and A_2 happen then the only useful queries for A' are the one in M_3. If, in addition, C happens then by Lemma 3,

$$\mathbf{Pr}[A' \text{ fails} | A_1 \wedge A_2 \wedge C] \geq 4\delta.$$

Since

$$\begin{aligned}\mathbf{Pr}[A_1 \wedge A_2 \wedge C] &= 1 - \mathbf{Pr}[\bar{A}_1 \vee \bar{A}_2 \vee \bar{C}] \\ &\geq 1 - \mathbf{Pr}[\bar{A}_1] - \mathbf{Pr}[\bar{A}_2] - \mathbf{Pr}[\bar{C}] \\ &\geq \frac{1}{2},\end{aligned}$$

we get $\mathbf{Pr}[A' \text{ fails}] \geq 2\delta$ which gives a contradiction. □

Note that by choosing $N = [n^\epsilon, n^{2\epsilon}]$ in the above proof, where ϵ is small constant, we make the result in Theorem 1 also valid for any $\delta < 2^{-n^c}$ where $c < 1$ is any constant. In particular, when $\delta = 2^{-n^c}$ the lower and upper bound is $\tilde{\Theta}(n^c)$. For $\delta < 2^{-n}$ one can ask n queries and finds the number of defective items exactly and therefore the bound is $\Theta(n)$.

In the proof of Theorem 1, one cannot take smaller intervals for N_i (for example $[n/2^{4i+4}, n/2^{4i}]$). This is because, with the multiplicand s for the union bound in (1), the probability of $\bar{A}_1$ cannot then be bounded by $1/16$. In the next theorem we overcome this problem, but only for fixed δ and large enough n. The idea is the following. We define the sets $N_{1,i}$ as we do with N_i in the proof of Theorem 1. One of the intervals N_{1,i_1} contains a "few" query sizes. The event A_1 then happens with high probability as before. We then recursively partition N_{1,i_1} to smaller intervals $N_{2,i}$. One of the smaller interval N_{2,i_2} contains a "very few" query sizes. But now the event A_1 should happen only to the query sizes that are outside N_{2,i_2} and inside N_{1,i_1} that are "few" which makes the union bound argument work again. The details are in the proof of the following

Theorem 2. *Let c be a large enough constant. Fix any δ. Any non-adaptive algorithm that 1.5-estimates the number of defective items must ask at least*

$$\Omega\left(\frac{\log \frac{1}{\delta} \log n}{(c \log^* n)^{(\log^* n)+1}}\right)$$

queries.

In particular, the lower bound is

$$\Omega\left(\frac{\log \frac{1}{\delta} \log n}{\log\log \overset{k}{\ldots} \log n}\right)$$

for any constant k.

Proof. We will denote $\log^{[k]} n = \log\log \overset{k}{\ldots} \log n$, $\log^{[0]} n = n$ and $\tau = \log^* n$. Let, for $i = 1, 2, \cdots, \tau$,

$$s_i = \frac{\log \frac{1}{\delta} \log^{[i]} n}{(c\tau)^{\tau-i+2}}, \qquad k_i = s_{i+1} = \frac{\log \frac{1}{\delta} \log^{[i+1]} n}{(c\tau)^{\tau-i+1}}, \qquad r_i = \frac{\log^{[i]} n}{16 \log^{[i+1]} n},$$

$$N_i = \left[\frac{n_i}{(\log^{[i-1]} n)^{1/4}}, n_i\right] \subset N_{i-1,j_{i-1}}$$

where $n_1 = n$ and for $i > 1$, n_i, $N_{i-1,j_{i-1}}$ and j_{i-1} will be defined later in the proof.

Let A be any non-adaptive algorithm that generates a random sequence of queries $M := Q_1, \ldots, Q_{s_1} \subseteq X = [n]$ from some probability space and with probability at least $1-\delta$, 1.5-estimates the number of defective items. Let C_i be the event that at most $s_i (= k_{i-1})$ of the sizes in $|Q_1|, \ldots, |Q_{s_1}|$ are in $N_{i-1,j_{i-1}}$ (and therefore in N_i). Now partition N_i into r_i intervals

$$N_{i,j} = \left[\frac{n_i}{(\log^{[i]} n)^{4j+4}}, \frac{n_i}{(\log^{[i]} n)^{4j}} \right]$$

where $j = 0, \ldots, r_i - 1$.

By Lemma 4, if event C_i happens then: for k_i and the sizes in $W = (|Q_1|, \ldots, |Q_{s_1}|)$ that are in N_i, there is j_i such that, with probability at least

$$1 - \frac{s_i}{k_i r_i} = 1 - \frac{16}{c\tau} \geq 1 - \frac{1}{16\tau}$$

at most k_i of the sizes in W are in N_{i,j_i}. We now define

$$n_{i+1} = \frac{n_i}{(\log^{[i]} n)^{4j_i+2}}.$$

Then

$$N_{i+1} = \left[\frac{n_{i+1}}{(\log^{[i]} n)^{1/4}}, n_{i+1} \right] = \left[\frac{n_i}{(\log^{[i]} n)^{4j_i+2\frac{1}{4}}}, \frac{n_i}{(\log^{[i]} n)^{4j_i+2}} \right] \subseteq N_{i,j_i}.$$

Therefore, with probability at least $1 - 1/(16\tau)$ the event C_{i+1} happens. The number of defective items d will be chosen to be in $[d_1, d_2]$ where $N_\tau = [n/d_2, n/d_1]$ and since $N_\tau \subseteq N_{i+1}$ for all i, we know at this stage that

$$\frac{n(\log^{[i]} n)^{4j_i+2\frac{1}{4}}}{n_i} \geq d \geq \frac{n(\log^{[i]} n)^{4j_i+2}}{n_i}.$$

Let M_i be the set of queries in M that have sizes in $N_{i-1,j_{i-1}}$. Now partition M_i into three sets of queries $M_{1,i} \cup M_{2,i} \cup M_{3,i}$ where $M_{1,i}$ are the queries in M_i that contains at most $n_i/(\log^{[i]} n)^{4j_i+4}$ items, $M_{2,i}$ are the queries in M_i that contains at least $n/(\log^{[i]} n)^{4j_j}$ items and $M_{3,i} = M_{i+1} = M_i \backslash (M_{1,i} \cup M_{2,i})$ are the queries that have sizes in N_{i,j_i}. Let $A_{1,i}(I)$ be the event that all the queries in $M_{1,i}$ give answers 0 in the oracle $\mathcal{O}_I$ and $A_{2,i}(I)$ be the event that all the queries in $M_{2,i}$ give answer 1. Let H_i be the *history* event $H_1 = C_1$ and $H_i = H_{i-1} \wedge A_{1,i-1} \wedge A_{2,i-1} \wedge C_i$. Let D and D' be as in Lemma 5 and B as in

Theorem 1. Then as in the proof of Theorem 1, $\mathbf{Pr}_{D'}[\bar{B}] \leq 1/(16\tau)$. Then

$$\begin{aligned}
\mathbf{Pr}_{D'}[\bar{A}_{1,i}|H_i] &= \mathbf{Pr}[(\exists Q \in M_{1,i})Q \cap I \neq \emptyset|H_i] \\
&\leq s_i \mathbf{Pr}[Q \cap I \neq \emptyset|Q \in M_{1,i}] \\
&= s_i(1 - \mathbf{Pr}[Q \cap I = \emptyset|Q \in M_{1,i}]) \\
&\leq s_i \left(1 - \left(1 - \frac{n_i}{n(\log^{[i]} n)^{4j_i+4}}\right)^d\right) \\
&\leq \frac{n_i s_i d}{n(\log^{[i]} n)^{4j_i+4}} \leq \frac{\log \frac{1}{\delta}}{(c\tau)^{\tau-i+2}(\log^{[i]} n)^{3/4}} \leq \frac{1}{16\tau}.
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{Pr}_{D'}[\bar{A}_{2,i}|H_i] &= \mathbf{Pr}[(\exists Q \in M_{2,i})Q \cap I = \emptyset|H_i] \\
&\leq s_i \mathbf{Pr}[Q \cap I = \emptyset|Q \in M_{2,i}] \\
&\leq s_i \left(1 - \frac{n_i}{n(\log^{[i]})^{4j_i}}\right)^d \\
&\leq s_i e^{-(\log^{[i]} n)^2} = \frac{\log \frac{1}{\delta} \log^{[i]} n}{(c\tau)^{\tau-i+2} e^{(\log^{[i]} n)^2}} \leq \frac{1}{16\tau}.
\end{aligned}$$

Then as in Theorem 1, $\mathbf{Pr}_D[\bar{A}_{1,i}|H_i] \leq 1/(8\tau)$ and $\mathbf{Pr}_D[\bar{A}_{2,i}|H_i] \leq 1/(8\tau)$. Now

$$\mathbf{Pr}[\bar{A}_{1,i} \vee \bar{A}_{2,i} \vee \bar{C}_{i+1}|H_i] \leq \frac{1}{2\tau}$$

and

$$\mathbf{Pr}[\bar{H}_{i+1}] \leq \mathbf{Pr}[\bar{H}_i] + \mathbf{Pr}[\bar{A}_{1,i} \vee \bar{A}_{2,i} \vee \bar{C}_{i+1}|H_i] \leq \mathbf{Pr}[\bar{H}_i] + \frac{1}{2\tau}$$

which implies that $\mathbf{Pr}[\bar{H}_\tau] \leq 1/2$. We now proceed as in Theorem 1. Let A' be as in the proof of Theorem 1. Then since $k_\tau < (1/16)\log(1/\delta)$ we have $\mathbf{Pr}[A' \text{ fails}|H_\tau] \geq 4\delta$ and therefore $\mathbf{Pr}[A' \text{ fails}] \geq 2\delta$ which gives a contradiction. □

3.2 Lower Bound for Deterministic Algorithm

In this section we prove

Theorem 3. *Let c be any constant. Any non-adaptive deterministic algorithm that c-estimates the number of defective items must ask at least $\Omega(n)$ queries.*

Proof. Let A be a non-adaptive deterministic algorithm that c-estimates the number of defective items. Let $Q_1, \ldots, Q_s$ be the queries that A asks. Let $d = n/2c$. For possible answers $a_1, \ldots, a_s \in \{0, 1\}$ to the queries we define $S_{(a_1,\ldots,a_s)}$, the set of all defective sets of size d that gives the answers $a_1, \ldots, a_s$ to the queries $Q_1, \ldots, Q_s$, respectively. That is, for every $I \in S_{(a_1,\ldots,a_s)}$ we have $|I| = d$

and for every $i = 1, \ldots, s$ we have $Q_i \cap I \neq \emptyset$ if $a_i = 1$ and $Q_i \cap I = \emptyset$ if $a_i = 0$. For $a = (a_1, \ldots, a_s) \in \{0,1\}^s$ let $I_a = \cup_{s \in S_a} s$. We now prove two claims

Claim 1. If the defective set is I_a then the algorithm gets the answers a to the queries.

If $Q_i \cap I_a \neq \emptyset$ then there is $I \in S_a$ such that $Q_i \cap I \neq \emptyset$ and then $a_i = 1$. If $Q_i \cap I_a = \emptyset$ then for every $I \in S_a$ we have $Q_i \cap I = \emptyset$ and then $a_i = 0$.

Claim 2. $|I_a| \leq cd$.

If $|I_a| > cd$ then the algorithm returns a value in $[cd+1, c^2 d]$ and then for the sets in S_a, that are of size d, this answer is not a c-estimation. A contradiction.

Since each $I \in S_a$ is of size d and is a subset of I_a we have

$$|S_a| \leq \binom{cd}{d}.$$

Since there are $\binom{n}{d}$ sets of size d we get

$$\binom{n}{d} = \sum_{a \in \{0,1\}^s} |S_a| \leq 2^s \binom{cd}{d}.$$

Since $d = n/(2c)$,

$$s \geq \log \binom{n}{\frac{n}{2c}} - \log \binom{\frac{n}{2}}{\frac{n}{2c}} = \Omega(n). \square$$

4 Upper Bounds

In this section we use techniques similar to the ones in [10, 14] to prove

Theorem 4. *Let c be any constant. There is a non-adaptive Monte Carlo randomized algorithm that asks*

$$s = O\left(\log \frac{1}{\delta} \log n\right)$$

queries and with probability at least $1 - \delta$, c-estimates the number of defective items.

We recall Chernoff Bound

Lemma 6. (Chernoff Bound). *Let $X_1, \ldots, X_t$ be independent random variables that takes values in $\{0,1\}$. Let $X = (X_1 + \cdots + X_t)/t$ and $\mathbf{E}[X] \leq \mu$. Then for any $\Delta \geq \mu$*

$$\mathbf{Pr}[X \geq \Delta] \leq \left(\frac{e^{1-\frac{\mu}{\Delta}}\mu}{\Delta}\right)^{\Delta t} \tag{2}$$

$$\leq \left(\frac{e\mu}{\Delta}\right)^{\Delta t}. \tag{3}$$

We will assume that $d \geq 6$. Otherwise, d can be estimated exactly in $O(\log n)$ more queries. Just run the algorithm that finds the defective items that asks $O(\log n)$ queries [22]. Here we give a 2-estimation algorithm. This can be extended in a straightforward manner to c-estimation for any constant c.

A p-query is a query Q that contains each item $i \in [n]$ randomly and independently with probability p. In the algorithm, $\mathcal{O}_I(Q) = 1$ if $Q \cap I \neq \emptyset$ and 0 otherwise.

Consider the following algorithm

Estimate(u, w, δ)

1. For each $p_i = 1/(u \cdot 2^{i/4})$, $i = 0, 1, 2, 3, \cdots, 8\log(w/u)$,
2. For $t = O(\log(1/\delta))$ independent p_i-queries $Q_{i,1}, \ldots, Q_{i,t}$ do:
3. $q_i = (\mathcal{O}_I(Q_{i,1}) + \cdots + \mathcal{O}_I(Q_{i,t}))/t$.
4. Choose the first i_0 such that $q_{i_0} < 0.83$.
5. If no such i_0 exists then output("$d > w$").
6. Otherwise output($D := 2/p_{i_0}$).

We now prove

Lemma 7. *Let $|I| = d \geq 6$. If $u \leq d \leq w$ then with probability at least $1 - \delta$, $d \leq D \leq 2d$. The algorithm asks $O(\log(1/\delta)\log(w/u))$ queries.*

In particular, for $u = 1$ and $w = n$, the algorithm asks

$$O\left(\log\frac{1}{\delta}\log n\right)$$

queries.

Proof. Let i_1 be such that $p_{i_1-1} > 2/d$ and $p_{i_1} \leq 2/d$. Then for $j = 0, 1, \cdots,$

$$2^{j/4}/d < p_{i_1+3-j} \leq 2^{(j+1)/4}/d.$$

For every i, j we have

$$\mu_i := \mathbf{E}[q_i] = \mathbf{E}[\mathcal{O}_I(Q_{i,j})] = \mathbf{Pr}[I \cap Q_{i,j} \neq \emptyset] = 1 - (1 - p_i)^d.$$

Since $d \geq 6$ we have $\mathbf{E}[q_{i_1+3}] = \mu_{i_1+3} \leq 1 - (1 - 2^{1/4}/d)^d \leq 0.74$ and

$$\begin{aligned}\mathbf{Pr}[D > 2d] &= \mathbf{Pr}[p_{i_0} < 1/d] = \mathbf{Pr}[i_0 > i_1 + 3] \\ &\leq \mathbf{Pr}[q_{i_1+3} \geq 0.83] \leq \delta/2. \end{aligned} \tag{4}$$

The first inequality in (4) follows from the fact that if $i_0 > i_1 + 3$ then $q_{i_1+3} \geq 0.83$. The second inequality follows from Chernoff bound (2) with $\mu = 0.74$ and $\Delta = 0.83$.

Now, since

$$\begin{aligned}\mathbf{E}[1 - q_{i_1+3-j}] &= 1 - \mu_{i_1+3-j} = (1 - p_{i_1+3-j})^d \\ &\leq e^{-p_{i_1+3-j}d} < e^{-2^{j/4}},\end{aligned}$$

we have $\mathbf{E}[1 - q_{i_1-2}] \leq \mathbf{E}[1 - q_{i_1-1}] \leq 0.136$ and

$$\begin{aligned}
\mathbf{Pr}[D < d] &= \mathbf{Pr}[p_{i_0} > 2/d] = \mathbf{Pr}[i_0 \leq i_1 - 1] \\
&= \sum_{i=0}^{i_1-1} \mathbf{Pr}[i_0 = i] \leq \sum_{i=0}^{i_1-1} \mathbf{Pr}[q_i < 0.83] \\
&= \sum_{i=0}^{i_1-3} \mathbf{Pr}[1 - q_i > 0.17] + \sum_{i=i_1-2}^{i_1-1} \mathbf{Pr}[1 - q_i > 0.17] \\
&\leq \sum_{i=0}^{i_1-3} \left(\frac{e \cdot e^{-2^{(i_1-i+3)/4}}}{0.17} \right)^{0.17 \cdot t} + \frac{\delta}{4} \qquad (5) \\
&\leq \sum_{k=0}^{\infty} \left(0.95 \cdot e^{-2^{k/4}} \right)^{0.17 \cdot t} + \frac{\delta}{4} \leq \frac{\delta}{4} + \frac{\delta}{4} = \frac{\delta}{2}.
\end{aligned}$$

In the first summand of (5) we use Chernoff bound (3). In the second summand we use Chernoff bound (2) for $\mu = 0.136$ and $\Delta = 0.17$. □

References

1. Hasan Abasi, Nader H. Bshouty, and Hanna Mazzawi. On exact learning monotone DNF from membership queries. In *Algorithmic Learning Theory - 25th International Conference, ALT 2014, Bled, Slovenia, October 8-10, 2014. Proceedings*, pages 111–124, 2014.
2. Nader H. Bshouty, Vivian E. Bshouty-Hurani, George Haddad, Thomas Hashem, Fadi Khoury, and Omar Sharafy. Adaptive group testing algorithms to estimate the number of defectives. *ALT*, 2017.
3. Chao L. Chen and William H. Swallow. Using group testing to estimate a proportion, and to test the binomial model. *Biometrics.*, 46(4):1035–1046, 1990.
4. Yongxi Cheng, Ding-Zhu Du, and Yinfeng Xu. A zig-zag approach for competitive group testing. *INFORMS Journal on Computing*, 26(4):677–689, 2014.
5. Yongxi Cheng, Ding-Zhu Du, and Feifeng Zheng. A new strongly competitive group testing algorithm with small sequentiality. *Annals OR*, 229(1):265–286, 2015.
6. Yongxi Cheng and Yinfeng Xu. An efficient FPRAS type group testing procedure to approximate the number of defectives. *J. Comb. Optim.*, 27(2):302–314, 2014.
7. Ferdinando Cicalese. *Fault-Tolerant Search Algorithms - Reliable Computation with Unreliable Information.* Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2013.
8. Graham Cormode and S. Muthukrishnan. What's hot and what's not: tracking most frequent items dynamically. *ACM Trans. Database Syst.*, 30(1):249–278, 2005.
9. Peter Damaschke and Azam Sheikh Muhammad. Bounds for nonadaptive group tests to estimate the amount of defectives. In *Combinatorial Optimization and Applications - 4th International Conference, COCOA 2010, Kailua-Kona, HI, USA, December 18-20, 2010, Proceedings, Part II*, pages 117–130, 2010.
10. Peter Damaschke and Azam Sheikh Muhammad. Competitive group testing and learning hidden vertex covers with minimum adaptivity. *Discrete Math., Alg. and Appl.*, 2(3):291–312, 2010.

11. R. Dorfman. The detection of defective members of large populations. *Ann. Math. Statist.*, pages 436–440, 1943.
12. D. Du and F. K Hwang. Combinatorial group testing and its applications. *World Scientific Publishing Company.*, 2000.
13. D. Du and F. K Hwang. Pooling design and nonadaptive group testing: important tools for dna sequencing. *World Scientific Publishing Company.*, 2006.
14. Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Estimating the number of defectives with group testing. In *IEEE International Symposium on Information Theory, ISIT 2016, Barcelona, Spain, July 10-15, 2016*, pages 1376–1380, 2016.
15. Edwin S. Hong and Richard E. Ladner. Group testing for image compression. *IEEE Trans. Image Processing*, 11(8):901–911, 2002.
16. F. K. Hwang. A method for detecting all defective members in a population by group testing. *Journal of the American Statistical Association*, 67:605–608, 1972.
17. William H. Kautz and Richard C. Singleton. Nonrandom binary superimposed codes. *IEEE Trans. Information Theory*, 10(4):363–377, 1964.
18. Joseph L.Gastwirth and Patricia 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. C. H. Li. A sequential method for screening experimental variables. *J. Amer. Statist. Assoc.*, 57:455–477, 1962.
20. Anthony J. Macula and Leonard J. Popyack. A group testing method for finding patterns in data. *Discrete Applied Mathematics*, 144(1-2):149–157, 2004.
21. Hung Q. Ngo and Ding-Zhu Du. A survey on combinatorial group testing algorithms with applications to DNA library screening. In *Discrete Mathematical Problems with Medical Applications, Proceedings of a DIMACS Workshop, December 8-10, 1999*, pages 171–182, 1999.
22. Ely Porat and Amir Rothschild. Explicit nonadaptive combinatorial group testing schemes. *IEEE Trans. Information Theory*, 57(12):7982–7989, 2011.
23. Dana Ron and Gilad Tsur. The power of an example: Hidden set size approximation using group queries and conditional sampling. *CoRR*, abs/1404.5568, 2014.
24. Jens Schlaghoff and Eberhard Triesch. Improved results for competitive group testing. *Combinatorics, Probability & Computing*, 14(1-2):191–202, 2005.
25. M. Sobel and P. A. Groll. Group testing to eliminate efficiently all defectives in a binomial sample. *Bell System Tech. J.*, 38:1179–1252, 1959.
26. William H. Swallow. Group testing for estimating infection rates and probabilities of disease transmission. *Phytopathology*, 1985.
27. Keith H. Thompson. Estimation of the proportion of vectors in a natural population of insects. *Biometrics*, 18(4):568–578, 1962.
28. S. D. Walter, S. W. Hildreth, and B. J. Beaty. Estimation of infection rates in population of organisms using pools of variable size. *Am J Epidemiol.*, 112(1):124–128, 1980.
29. Jack K. Wolf. Born again group testing: Multiaccess communications. *IEEE Trans. Information Theory*, 31(2):185–191, 1985.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il