Optimal Deterministic Group Testing Algorithms to Estimate the Number of Defectives

Nader H. Bshouty and Catherine A. Haddad-Zaknoon$^{(\boxtimes)}$

Technion - Israel Institute of Technology, Haifa, Israel
{bshouty,catherine}@cs.technion.ac.il

Abstract. We study the problem of estimating the number of defective items d within a pile of n elements up to a multiplicative factor of $\Delta > 1$, using deterministic group testing algorithms. We bring lower and upper bounds on the number of tests required in both the adaptive and the non-adaptive deterministic settings given an upper bound D on the defectives number. For the adaptive deterministic settings, our results show that, any algorithm for estimating the defectives number up to a multiplicative factor of Δ must make at least $\Omega\left((D/\Delta^2)\log(n/D)\right)$ tests. This extends the same lower bound achieved in [1] for non-adaptive algorithms. Moreover, we give a polynomial time adaptive algorithm that shows that our bound is tight up to a small additive term.

For non-adaptive algorithms, an upper bound of $O((D/\Delta^2)$ $(\log(n/D) + \log \Delta))$ is achieved by means of non-constructive proof. This improves the lower bound $\Omega((\log D)/(\log \Delta))D\log n)$ from [1] and matches the lower bound up to a small additive term.

In addition, we study polynomial time constructive algorithms. We use existing polynomial time constructible *expander regular bipartite graphs*, *extractors* and *condensers* to construct two polynomial time algorithms. The first algorithm makes $O((D^{1+o(1)}/\Delta^2) \cdot \log n)$ tests, and the second makes $(D/\Delta^2) \cdot Quazipoly\,(\log n)$ tests. This is the first explicit construction with an almost optimal test complexity.

Keywords: Group testing · Pooling design · Deterministic group testing

1 Introduction

The problem of *group testing* is the problem of identifying or, in some cases, examining the properties of a small amount of items known as *defective* items within a pile of elements using *group tests*. Let X be a set of n items, and let $I \subseteq X$ be the set of defective items. A *group test* is a subset $Q \subseteq X$ of items. The result of the test Q with respect to I is defined by $Q(I) := 1$ if $Q \cap I \neq \emptyset$ and $Q(I) := 0$ otherwise. While the defective set I is unknown to the algorithm, in many cases we might be interested in finding the size of the defective set $|I|$, or at least an estimation of that value with a minimum number of tests.

Group testing was originally proposed as a potential solution for economising mass blood testing during WWII [11]. Since then, group testing approach has been diversely applied in a wide area of practical applications including DNA library screening [20], product testing quality control [22], file searching in storage systems [17], sequential screening of experimental variables [18], efficient contention algorithms for MAC [17, 26], data compression [16], and computations in data stream model [8]. Recently, during the COVID-19 pandemic outbreak, a number of researches adopted the group testing paradigm not only to accelerate mass testing process, but also to dramatically reduce the number of kits required for testing due to severe shortages in the testing kits supply [14, 19, 27].

While an up-front knowledge of the value of d or at least an upper bound on it is required in many of the algorithms aimed at identifying the defective items, estimating or finding the number of defectives is an interesting problem on its own as well. Defectives estimation via group testing has been applied vastly in biological and medical applications [7, 13, 23–25]. In [24], for example, group testing algorithms are used to estimate aster-yellow virus transmitters proportion over the organisms in a natural population of leafhoppers. Similarly, in [25], the authors estimate the infection rate of the yellow-fever virus in mosquito population using group testing methods. On the other hand, in [13], group-testing-based estimation of rare diseases prevalence is employed not only for its effectiveness but also because it naturally preserves individual anonymity of the subjects.

Algorithms dedicated for this task might operate in *stages* or *rounds*. In each round, the tests are defined in advance and tested in a single parallel step. Tests on some round might depend on the test results of the preceding rounds. A single round algorithm is called *non-adaptive* algorithm, while a multi-round algorithm is called *adaptive algorithm*.

In recent years, there has been an increasing interest in the problem of estimating the number of defective items via group testing [2, 3, 5, 7, 9, 10, 12, 21]. The target in some of these papers is to find an estimation $\hat{d}$ within an additive factor of $\epsilon < 1$ such that $(1 - \epsilon)d \leq \hat{d} \leq (1 + \epsilon)d$. For randomized adaptive algorithms we have the following results. Falhatgar et.al. [12] give a randomised adaptive algorithm that estimates d using $2 \log \log d + O(1/\epsilon^2 \log 1/\delta)$ queries in expectation where δ is the failure probability of the algorithm. Bshouty et. al. [3] modified this result and gave an algorithm that uses $(1 - \delta) \log \log d + O((1/\epsilon^2) \log 1/\delta)$ expected number of queries. Moreover, they proved a lower bound of $(1 - \delta) \log \log d + \Omega((1/\epsilon) \log(1/\delta))$ queries.

For randomized non-adaptive algorithms with constant estimation, Damaschke and Sheikh Muhammad give in [10] a randomized non-adaptive algorithm that makes $O((\log(1/\delta)) \log n)$ tests and in [2], Bshouty gives the lower bound $\Omega(\log n/ \log \log n)$.

In this paper, we are interested in *d*eterministic adaptive and non-adaptive algorithms that estimate the defective items set size d up to a multiplicative factor of $\Delta > 1$. Formally, let $|I| := d$ and let $D \geq d$. We say that a deterministic algorithm $\mathcal{A}$ estimates d up to a multiplicative factor of Δ if, given D as an

Table 1. Upper and lower bounds on the number of tests required for estimating defectives in deterministic group testing.

Bounds	Adaptive/Non-adapt.	Result	Explicit/Non-expl.	Ref.
Lower B.	Non-Adapt.	$\frac{D}{\Delta^2} \log \frac{n}{D}$	–	[1]
Lower B.	Adaptive	$\frac{D}{\Delta^2} \log \frac{n}{D}$	–	Ours
Upper B.	Adaptive	$\frac{D}{\Delta^2} \left(\log \frac{n}{D} + \log \Delta \right)$	Explicit	Ours
Upper B.	Non-Adapt.	$\frac{\log D}{\log \Delta} D \log n$	Non-Expl.	[1]
Upper B.	Non-Adapt.	$\frac{D}{\Delta^2} \left(\log \frac{n}{D} + \log \Delta \right)$	Non-expl.	Ours
Upper B.	Non-Adapt.	$\frac{D^{1+o(1)}}{\Delta^2} \log n$	Explicit[a]	Ours
Upper B.	Non-Adapt.	$\frac{D}{\Delta^2} \cdot \text{Quazipoly}(\log n)$	Explicit	Ours

[a] This result is true for $\Delta > C$ for some constant C. See Sect. 6.2.

input to the algorithm, it evaluates an estimation $\hat{d}$ such that $d/\Delta \leq \hat{d} \leq d\Delta$. Bshouty et al. show in [3] that, if no upper bound D is given to the algorithm, then any deterministic adaptive algorithm (and therefore also non-deterministic algorithm) for this problem must make at least $\Omega(n)$ tests. This is equivalent to testing all the items. This justifies the fact that any non-trivial efficient algorithm must have some upper bound D for d.

Agarwal et.al. [1] consider this problem. They first give the lower bound of $\Omega((D/\Delta^2) \log(n/D))$ queries for any non-adaptive deterministic algorithm. Moreover, using a non-constructive proof, they give an upper bound of $O(((\log D)/(\log \Delta))D \log n)$ queries.

We further investigate this problem. We bring new lower and upper bounds on the number of tests required both in adaptive and non-adaptive deterministic algorithms. For the adaptive deterministic settings, our results show that, any algorithm for estimating the defectives number up to a multiplicative factor of Δ must make at least $\Omega\left((D/\Delta^2)\log(n/D)\right)$ tests. This extends the same lower bound achieved in [1] for non-adaptive algorithms. Furthermore, we give a polynomial time adaptive algorithm that shows that our bound is tight up to a small additive term.

For non-adaptive algorithms, we achieve an upper bound of $O((D/\Delta^2)(\log(n/D)+\log \Delta))$ by means of non-constructive proof. This improves the lower bound $O((\log D)/(\log \Delta))D \log n)$ from [1], and matches the lower bound up to a small additive term.

We then study polynomial time constructive algorithms. For this task, we use existing polynomial time constructible *expander regular bipartite graphs*, *extractors* and *condensers* to construct two polynomial time algorithms. The first algorithm makes $O((D^{1+o(1)}/\Delta^2) \cdot \log n)$ tests, and the second makes $(D/\Delta^2) \cdot quazipoly (\log n)$ tests. To the best of our knowledge, this is the first explicit construction with an almost optimal test complexity. Our results are summarised in Table 1.

2   Definitions and Preliminary Results

In this section, we give some notations and definition that will be used in this paper.

Let $X = [n] := \{1, \cdots, n\}$ be a set of *items*. Let $I \subseteq X$ be a set of *defective* items, and let d denote its size, i.e. $d = |I|$. In the group testing settings, a *test* is a subset $Q \subseteq X$ of items. An answer to a test Q with respect to the defective items set I, is denoted by $Q(I)$, such that $Q(I) := 1$ if $Q \cap I \neq \emptyset$ and 0 otherwise. We denote by $\mathcal{O}_I$ an *oracle* that for a test Q returns $Q(I)$.

Let $\mathcal{A}$ be an algorithm with an access to $\mathcal{O}_I$, and let $d = |I|$. We say that the algorithm $\mathcal{A}$ *estimates d up to a multiplicative factor of Δ*, if $\mathcal{A}$ gets as an input an upper bound $D \geq d$ and a parameter $\Delta > 1$, and outputs $\hat{d}$ such that $d/\Delta \leq \hat{d} \leq d\Delta$. We say that $\mathcal{A}$ is an *adaptive* algorithm, if its queries depend on the result of previous queries, and *non-adaptive* if its queries are independent of previous ones and therefore, can be executed in a single parallel step. We may assume that $D \geq \Delta^2$, otherwise, the algorithm trivially outputs $\hat{d} = D/\Delta$. We note here that $\Delta \geq 1 + \Omega(1)$, that is, it is greater than a constant that is greater than 1 and it may depend[1] on n and/or D. This is implicit in [1] and is also constrained in this paper. It is also interesting to investigate this problem when $\Delta = 1 + o(n)$ where $o()$ (small o) is with respect to D and/or n.

We will use the following

Lemma 1. Chernoff's Bound. *Let $X_1, \ldots, X_m$ be independent random variables taking values in $\{0, 1\}$. Let $X = \sum_{i=1}^{m} X_i$ denotes their sum and $\mu = \mathbf{E}[X]$ denotes the sum's expected value. Then*

$$\Pr[X > (1 + \lambda)\mu] \leq \left(\frac{e^\lambda}{(1 + \lambda)^{(1+\lambda)}} \right)^\mu \leq e^{-\frac{\lambda^2 \mu}{2+\lambda}} \leq \begin{cases} e^{-\frac{\lambda^2 \mu}{3}} & \text{if } 0 < \lambda \leq 1 \\ e^{-\frac{\lambda \mu}{3}} & \text{if } \lambda > 1 \end{cases} . \tag{1}$$

In particular,

$$\Pr[X > \Lambda] \leq \left(\frac{e\mu}{\Lambda} \right)^\Lambda . \tag{2}$$

For $0 \leq \lambda \leq 1$ we have

$$\Pr[X < (1 - \lambda)\mu] \leq \left(\frac{e^{-\lambda}}{(1 - \lambda)^{(1-\lambda)}} \right)^\mu \leq e^{-\frac{\lambda^2 \mu}{2}} . \tag{3}$$

Moreover, we will often use the inequality

$$\left(\frac{n}{k} \right)^k \leq \binom{n}{k} \leq \sum_{i=0}^{k} \binom{n}{i} \leq \left(\frac{en}{k} \right)^k . \tag{4}$$

---

[1] For example $\Delta = \log \log n + \log D$.

3   Upper Bound for Non-adaptive Deterministic Algorithms

In this section, we give the upper bound for deterministic non-adaptive algorithm that estimates d up to a multiplicative factor of Δ. We prove:

Theorem 1. *Let D be some upper bound on the number of defective items d and $\Delta > 1$. Then, there is a deterministic non-adaptive algorithm that makes*

$$O\left(\frac{D}{\Delta^2}\left(\log\frac{n}{D} + \log\Delta\right)\right)$$

tests and outputs $\hat{d}$ such that $\frac{d}{\Delta} \leq \hat{d} \leq d\Delta$.

To prove the Theorem we need the following:

Lemma 2. *Let $\Delta > 1$ and $\ell \geq 2\Delta^2$. There is a non-adaptive deterministic algorithm that makes*

$$t = O\left(\frac{\ell}{\Delta^2}\left(\log\frac{n}{\ell} + \log\Delta\right)\right)$$

tests such that,

1. *If the number of defectives d is less than ℓ/Δ^2, it outputs 0.*
2. *If it is greater than ℓ/Δ, it outputs 1.*

Proof. We choose a constant c such that $(1 - \Delta^2/(c\ell))^{\ell/\Delta^2} = 1/e$. Note that

$$\left(1 - \frac{\Delta^2}{2\ell}\right)^{\ell/\Delta^2} \geq 1 - \frac{\Delta^2\ell}{2\ell\Delta^2} = \frac{1}{2} > \frac{1}{e}$$

and

$$\left(1 - \frac{2\Delta^2}{\ell}\right)^{\ell/\Delta^2} = \left(\left(1 - \frac{2\Delta^2}{\ell}\right)^{\frac{\ell}{2\Delta^2}}\right)^2 \leq \frac{1}{e^2} < \frac{1}{e}.$$

Therefore, such c exists and we have $1/2 \leq c \leq 2$.

Consider a test $Q \subseteq [n]$ chosen at random where each item $i \in [n]$ is chosen to be in Q with probability $\Delta^2/(c\ell)$. Let I be the set of defective items such that $|I| = d$, and let $Q(I)$ be the result of the test Q with respect to the set I. Then,

$$\Pr[Q(I) = 0] = \left(1 - \frac{\Delta^2}{c\ell}\right)^d. \tag{5}$$

If $d \leq \ell/\Delta^2$,

$$\Pr[Q(I) = 0] \geq \left(1 - \frac{\Delta^2}{c\ell}\right)^{\ell/\Delta^2} = e^{-1}, \tag{6}$$

if $d = 2\ell/\Delta^2$,

$$\Pr[Q(I) = 0] = \left(1 - \frac{\Delta^2}{c\ell}\right)^{2\ell/\Delta^2} = e^{-2}, \tag{7}$$

and if $d = \ell/\Delta$, we get:

$$\Pr[Q(I) = 0] = \left(\left(1 - \frac{\Delta^2}{c\ell}\right)^{\frac{\ell}{\Delta^2}}\right)^{\Delta} = e^{-\Delta}. \tag{8}$$

Let $Q_1, Q_2, \ldots, Q_t$ be a sequence of t i.i.d tests such that

$$t = \frac{c'\ell}{(\Delta - 1)^2} \ln \frac{c''\Delta^2 n}{\ell}$$

where $c' = 54e^2$ and $c'' = 4e$.

Let

$$\eta = e^{-1}\left(\frac{1}{2} + \frac{1}{2\Delta}\right).$$

Consider the following two events:

1. A: There is a set of defectives I of size $|I| \leq \ell/\Delta^2$ such that the number of tests with 0 answer is less than ηt.
2. B: There is a set of defectives J of size $|J| > \ell/\Delta$ such that the number of tests with 0 answer is at least ηt.

Notice that, to prove the lemma it is enough to prove that $\Pr[A \vee B] < 1$. We will show that $\Pr[A], \Pr[B] < 1/2$.

Let $X_1, \ldots, X_t$ be random variables such that $X_i = 1$ if and only if $Q_i(I) = 0$. Let X be the number of tests that yield the result 0. Therefore, $X = \sum_{i=1}^{t} X_i$ and define $\mu := \mathbf{E}[X]$.

If $|I| = d \leq \ell/\Delta^2$, then $\mu = t \cdot \mathbf{E}[X_i] = t \cdot \Pr[X_i = 1]$. By (6) we have

$$\mu = \mathbf{E}[X] \geq t \cdot e^{-1}. \tag{9}$$

By (3) in Lemma 1, for $\lambda = 1/2 - 1/(2\Delta)$ we have

$$\Pr[X \leq \eta t] = \Pr[X \leq (1 - \lambda)te^{-1}] \leq \Pr[X \leq (1 - \lambda)\mu] \leq e^{-\frac{\lambda^2 \mu}{2}} \leq e^{-\frac{(1-\Delta^{-1})^2 t}{8e}}.$$

Using this result, Eq. (4) and the union bound, we can conclude that

$$\Pr[A] \leq \left(\sum_{i=0}^{\ell/\Delta^2} \binom{n}{i}\right) e^{-\frac{(1-\Delta^{-1})^2 t}{8e}} \leq \left(\frac{e\Delta^2 n}{\ell}\right)^{\frac{\ell}{\Delta^2}} e^{-\frac{(1-\Delta^{-1})^2 t}{8e}}$$

$$= \left(\frac{e\Delta^2 n}{\ell}\right)^{\frac{\ell}{\Delta^2}} e^{-\frac{c'\ell}{8e\Delta^2} \ln \frac{c''\Delta^2 n}{\ell}} = \left(\frac{e\Delta^2 n}{\ell}\right)^{\frac{\ell}{\Delta^2}} \left(\frac{c''\Delta^2 n}{\ell}\right)^{-\frac{c'\ell}{\Delta^2}} < \frac{1}{2}.$$

On the other hand, for the event B, we have two cases.

Case I. $1 < \Delta \le 2$.

If there is a set of defectives J of size $|J| > \ell/\Delta$ such that more than ηt of the tests yield the answer 0, then there is a set of defectives J' of size $|J'| = \ell/\Delta$ such that more than ηt of the tests answers are 0. Denote by B' the latter event. Then, by (8) we have $\mu = \mathbf{E}[X] = e^{-\Delta}t$ and for $\lambda = (e^{\Delta-1} - 1)/2 \ge (\Delta - 1)/2$, $\eta' = (e^{-1} + e^{-\Delta})/2 \le \eta$ we get

$$\Pr[B] \le \Pr[B'] \le \binom{n}{\ell/\Delta} \Pr[X \ge \eta t] \le \binom{n}{\ell/\Delta} \Pr[X \ge \eta' t]$$

$$= \binom{n}{\ell/\Delta} \Pr[X \ge (1 + \lambda)\mu]$$

$$\le \left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} \Pr[X \ge (1 + \lambda)\mu]$$

If $1 < \Delta \le 2$ then $0 \le \lambda \le 1$ and then by (1) in Lemma 1, we have

$$\left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} \Pr[X \ge (1 + \lambda)\mu] \le \left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} e^{-\lambda^2\mu/3}$$

$$\le \left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} e^{-(\Delta-1)^2\mu/12} \quad \text{since } \lambda \ge (\Delta - 1)/2$$

$$= \left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} e^{-(\Delta-1)^2 e^{-\Delta}t/12}$$

$$= \left(\frac{e\Delta n}{\ell}\right)^{\frac{\ell}{\Delta}} \left(\frac{c''\Delta^2 n}{\ell}\right)^{-c'\ell e^{-\Delta}/12}$$

$$\le \left(\frac{2en}{\ell}\right)^{\ell} \left(\frac{c''n}{\ell}\right)^{-(c'e^{-2}/12)\ell} < \frac{1}{2} \qquad 1 \le \Delta < 2$$

Case II. $\Delta > 2$.

In this case we have $\ell/\Delta > 2\ell/\Delta^2$. Therefore, if there is a set of defectives J of size $|J| > \ell/\Delta$ such that more than ηt of the tests yield the answer 0, then there is a set of defectives J' of size $|J'| = 2\ell/\Delta^2$ such that more than ηt of the tests answers are 0. Denote by B'' the latter event. By (7), $\mu = \mathbf{E}[X] = e^{-2}t$. Let $\lambda = 1/3 - 1/(3\Delta) < 1$. Then $\eta t > (1 + \lambda)\mu$. By (1) in Lemma 1, we have

$$\Pr[X \ge \eta t] \le \Pr[X \ge (1 + \lambda)\mu] \le e^{-\frac{\lambda^2\mu}{3}} \le e^{-\frac{(1-\Delta^{-1})^2 t}{27e^2}}.$$

Then

$$\Pr[A] \le \Pr[B''] \le \binom{n}{2\ell/\Delta^2} e^{-\frac{(1-\Delta^{-1})^2 t}{27e^2}} \le \left(\frac{e\Delta^2 n}{2\ell}\right)^{\frac{2\ell}{\Delta^2}} e^{-\frac{(1-\Delta^{-1})^2 t}{27e^2}}$$

$$= \left(\frac{e\Delta^2 n}{2\ell}\right)^{\frac{2\ell}{\Delta^2}} e^{-\frac{c'\ell}{27e^2\Delta^2} \ln \frac{c''\Delta^2 n}{\ell}} = \left(\frac{e\Delta^2 n}{2\ell}\right)^{\frac{2\ell}{\Delta^2}} \left(\frac{c''\Delta^2 n}{\ell}\right)^{-\frac{c'\ell}{27e^2\Delta^2}} < \frac{1}{2}.$$

We are now ready to prove Theorem 1.

Let $\mathcal{A}(\ell, \Delta)$ be the algorithm from Lemma 2. Then, $\mathcal{A}(\ell, \Delta)$ makes at most

$$\frac{c\ell}{\Delta^2} \log \frac{\Delta n}{\ell} \tag{10}$$

queries for some constant c, and

1. If $\mathcal{A}(\ell, \Delta) = 1$, then $d \geq \frac{\ell}{\Delta^2}$.
2. If $\mathcal{A}(\ell, \Delta) = 0$, then $d \leq \frac{\ell}{\Delta}$.

Consider the algorithm $\mathcal{T}(n, D, \Delta)$ that runs $\mathcal{A}(D/\Delta^i, \Delta)$ for all $i = 0, \ldots, \lceil \log D / \log \Delta \rceil$. Let r be the minimum integer such that $\mathcal{A}(D/\Delta^r, \Delta) = 1$. Algorithm $\mathcal{T}(n, D, \Delta)$ then outputs $\hat{d} = D/\Delta^{r+1}$. See algorithm $\mathcal{T}$ in Fig. 1.
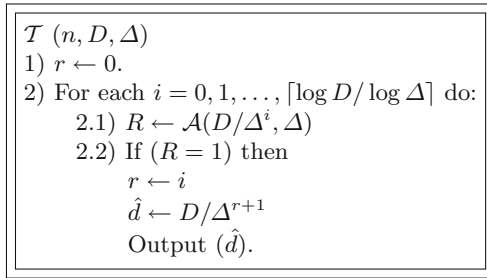
$\mathcal{T}\ (n, D, \Delta)$
1) $r \leftarrow 0$.
2) For each $i = 0, 1, \ldots, \lceil \log D / \log \Delta \rceil$ do:
    2.1) $R \leftarrow \mathcal{A}(D/\Delta^i, \Delta)$
    2.2) If $(R = 1)$ then
        $r \leftarrow i$
        $\hat{d} \leftarrow D/\Delta^{r+1}$
        Output $(\hat{d})$.

Fig. 1. Algorithm $\mathcal{T}$

We now prove:

Lemma 3. *Algorithm $\mathcal{T}(n, D, \Delta)$ is deterministic non-adaptive that makes*

$$O\left(\frac{D}{\Delta^2} \log\left(\frac{\Delta n}{D}\right)\right)$$

tests and outputs $\hat{d}$ that satisfies

$$\frac{d}{\Delta} \leq \hat{d} \leq \Delta d.$$

Proof. For $i = 0$, if $A(D/\Delta^i, \Delta) = 1$ then $d \geq D/\Delta^2$. Then $\hat{d} = D/\Delta \leq \Delta d$ and since $D \geq d$ we also have $\hat{d} = D/\Delta \geq d/\Delta$.

For $i > 0$, if $A(D/\Delta^{i-1}, \Delta) = 0$ and $A(D/\Delta^i, \Delta) = 1$ then $d \leq D/\Delta^i$ and $d \geq D/\Delta^{i+2}$. Then $\hat{d} = D/\Delta^{i+1} \leq \Delta d$ and $\hat{d} \geq d/\Delta$.

Let $q = \lceil \log D / \log \Delta \rceil$. Let t denote the number of queries performed by algorithm $\mathcal{T}(n, D, \Delta)$. By (10), the number of tests is at most

$$\sum_{i=0}^{q} \frac{cD}{\Delta^i \Delta^2} \log \frac{n \Delta^{i+1}}{D} \leq \frac{cD}{\Delta^2} \sum_{i=0}^{\infty} \frac{1}{\Delta^i} \log \frac{n \Delta^{i+1}}{D}$$

$$= \frac{cD}{\Delta^2} \left(\left(\log \frac{n}{D} \right) \sum_{i=0}^{\infty} \frac{1}{\Delta^i} + (\log \Delta) \sum_{i=0}^{\infty} \frac{i+1}{\Delta^i} \right)$$

$$\leq \frac{cD}{\Delta^2} \left(\frac{\Delta}{\Delta - 1} \log \frac{n}{D} + \frac{\Delta^2}{(\Delta - 1)^2} \log \Delta \right).$$

For the case when $\Delta = 1 + \Theta(1)$ we get

$$t = O \left(D \log \frac{n}{D} \right)$$

and for the case when $\Delta = \omega(1)$ we get

$$t = O \left(\frac{D}{\Delta^2} \left(\log \frac{n}{D} + \log \Delta \right) \right).$$

4   Lower Bound for Adaptive Deterministic Algorithm

In this section, we prove the following lower bound.

Theorem 2. *Any deterministic adaptive group testing algorithm that given $D > d$, outputs $\hat{d}$ that satisfies $d/\Delta \leq \hat{d} \leq \Delta d$ must make at least*

$$\Omega \left(\frac{D}{\Delta^2} \log \frac{n}{D} \right)$$

queries.

For the proof, we use the following from [3].

Lemma 4. *Let A be a deterministic adaptive algorithm that for a defective sets $I \subset [n]$ makes the tests $T_1^I, T_2^I \ldots, T_{w(I)}^I$ and let $s(I)$ be the sequence of answers to these tests. If $M = |\{s(I)|I \subseteq [n]\}|$ then the test complexity of A is $\max_I w(I) \geq \log M$.*

The following Lemma assists us to prove the result declared by Theorem 2.

Lemma 5. *Any deterministic adaptive algorithm such that, if the number of defectives d is less than or equal d_1 it outputs 0 and if it is greater than d_2 it outputs 1, must make*

$$\Omega \left(d_1 \log \frac{n}{d_2} \right)$$

tests.

In particular, when $d_1 = \ell/\Delta^2$ and $d_2 = \ell/\Delta$ we get

$$\Omega\left(\frac{\ell}{\Delta^2}\left(\log\frac{n}{\ell} + \log\Delta\right)\right)$$

tests.

Proof. Let A be such algorithm. Let $s(I)$ be the sequence of answers to the tests of A when the set of defective items is I. Consider a set I of size d_1 and let $\mathcal{J} = \{J \subseteq [n] : |J| = d_1, s(J) = s(I)\}$. Let $I' = \cup_{J \in \mathcal{J}} J$. We claim that $s(I') = s(I)$. Suppose for the contrary, $s(I') \neq s(I)$. Then, since $I \subseteq I'$, there is a test $Q \subseteq [n]$ that is asked by A that gives answer 0 to I and 1 to I'. Since $I' \cap Q \neq \emptyset$, there is a subset $J' \in \mathcal{J}$ such that $J' \cap Q \neq \emptyset$ and therefore Q gives answer 1 to J'. Then $s(J') \neq s(I)$ and we get a contradiction.

Since $s(I') = s(I)$ and algorithm A outputs 0 to I, it also outputs 0 to I'. Therefore, $|I'| \leq d_2$. Therefore $|\mathcal{J}| \leq N := \binom{d_2}{d_1}$. That is, for every possible sequence of answers s' of the algorithm A, there is at most N sets of size d_1 that get the same sequence of answers. Since there are $L := \binom{n}{d_1}$ such sets, the number of different sequences of answers that A might have must be at least L/N. By Lemma 4, the number of tests that the algorithm makes is at least

$$\log\frac{\binom{n}{d_1}}{\binom{d_2}{d_1}} \geq \log\left(\frac{n}{ed_2}\right)^{d_1} = \Omega\left(d_1\log\frac{n}{d_2}\right).$$

The conclusions established by Lemma 5 show that the upper bound from Lemma 2 is tight. Moreover, using these results, we provide the following proof for Theorem 2.

Proof. Let $d_1 = D/\Delta^2 - 1$ and $d_2 = D$. For sets of size less than or equal d_1 the algorithm returns $d_1/\Delta \leq \hat{d} \leq \Delta d_1$ and for sets of equal to d_2 the algorithm returns $d_2/\Delta < \hat{d} \leq \Delta d_2$. Since $\Delta d_1 < d_2/\Delta$, the above intervals are disjoint. So, the algorithm can distinguish between sets of size less that or equal to d_1 and sets of size greater than d_2. By Lemma 5 the algorithm must make at least

$$\Omega\left(\frac{D}{\Delta^2}\log\frac{n}{D}\right)$$

tests.

5   Polynomial Time Adaptive Algorithm

In this section, we prove:

Theorem 3. *Let D be some upper bound on the number of defective items d and $\Delta > 1$. Then, there is a linear time deterministic adaptive algorithm that makes*

$$O\left(\frac{D}{\Delta^2}\left(\log\frac{n}{D} + \log\Delta\right)\right)$$

tests and outputs $\hat{d}$ such that $\frac{d}{\Delta} \leq \hat{d} \leq d\Delta$.

We first describe the algorithm. The algorithm gets as an input the set of items $X = [n]$ and splits it into two equally-sized disjoint sets Q_1 and Q_2. The algorithm asks the queries defined by Q_1 and Q_2 and proceeds in the splitting process on the sets that yielded positive answers only. We call these sets *defective sets*. As long as the algorithm gets less than D/Δ^2 distinct defective sets, it continues to split and test. Two cases can happen. Either it gets D/Δ^2 defective sets and then the algorithm outputs $\hat{d} = D/\Delta$, or the number of the defective sets is always less than D/Δ^2 and then, the algorithm finds all the defective items and returns their exact number. The algorithm is given in Fig. 2. The algorithm invokes the procedure **Split**(X) that on an input $X = \{a_1, a_2, \ldots, a_n\}$, it returns the set W where $W := \{X_1, X_2\}$ such that $X_i \subseteq X$, $X_1 = \{a_1, a_2, \ldots, a_{\lfloor n/2 \rfloor}\}$, $X_2 = \{a_{\lfloor n/2 \rfloor + 1}, \ldots, a_n\}$ if $|X| \geq 2$, and $W := \{X\}$ otherwise.

Adaptive-dEstimate $(\mathcal{O}_I, X, \Delta, D)$
1) $Q \leftarrow X, S \leftarrow \emptyset$
2) While $(|Q| \leq D/\Delta^2)$ do:
     2.1) For each $Q_i \in Q$
        $\left\{Q_i^{(1)}, Q_i^{(2)}\right\} \leftarrow$ **Split**(Q_i)
        If $(Q_i^{(1)}(I) = 1)$ then $S \leftarrow S \cup \{Q_i^{(1)}\}$
        If $(Q_i^{(2)}(I) = 1)$ then $S \leftarrow S \cup \{Q_i^{(2)}\}$
     2.2) If $\forall S_i \in S, |S_i| = 1$
        $\hat{d} \leftarrow |S|$
        Output $(\hat{d})$
    Else
        $Q \leftarrow S, S \leftarrow \emptyset$.
3) $\hat{d} \leftarrow |Q| \cdot \Delta$.
4) Output $(\hat{d})$

Fig. 2. Algorithm **Adaptive-dEstimate** to estimate the number of defective items.

Lemma 6. *Algorithm* **Adaptive-dEstimate** *is a deterministic adaptive algorithm that makes*

$$2\frac{D}{\Delta^2} \log \frac{n\Delta^2}{D} = O\left(\frac{D}{\Delta^2}\left(\log \frac{n}{D} + \log \Delta\right)\right)$$

tests and outputs an estimation $\hat{d}$ such that:

$$\frac{d}{\Delta} \leq \hat{d} \leq d\Delta.$$

Proof. If $d \leq \frac{D}{\Delta^2}$, then the splitting process in step 2 of the algorithm proceeds until each defective item belongs to a distinct set. Eventually, the condition in

step 2.2 is met and the algorithm outputs the exact value of d. If $d > D/\Delta^2$, then the splitting process stops when the number of defective sets $|Q|$ exceeds D/Δ^2. The algorithm halts and outputs $\hat{d} = |Q|\Delta$. Obviously, $|Q| \leq d$. Therefore, $\hat{d} = |Q|\Delta \leq d\Delta$. Moreover, $|Q| > D/\Delta^2 \geq d/\Delta^2$ which implies that $\hat{d} \geq d/\Delta$.

The number of iterations cannot exceed $\log n$ iterations. In the first $\log(D/\Delta^2)$ iterations, in the worst case scenario, the algorithm splits its current set Q_i on each iteration into two sets $Q_i^{(1)}$ and $Q_i^{(2)}$ such that $Q_i^{(1)}(I) = Q_i^{(2)}(I) = 1$. Therefore, the number of tests that the algorithm asks over all the first $\log(D/\Delta^2)$ iterations is at most

$$\sum_{i=1}^{\log(D/\Delta^2)} 2^i \leq 2\frac{D}{\Delta^2}.$$

Since $|Q| \leq D/\Delta^2$, in the other $\log n - \log(D/\Delta^2)$ iterations, the algorithm makes at most $2D/\Delta^2$ tests each iteration. So, the total number of tests is at most

$$2\frac{D}{\Delta^2}\left(\log n - \log\frac{D}{\Delta^2}\right) + 2\frac{D}{\Delta^2} = O\left(\frac{D}{\Delta^2}\log\frac{n\Delta^2}{D}\right)$$

.

6   Polynomial Time Non-adaptive Algorithm

In this section, we show how to use expanders, condensers and extractors to construct deterministic non-adaptive algorithms for defectives number estimation. We prove:

Theorem 4. *Let D be some upper bound on the number of defective items d and $\Delta > 1$. Then, there is a polynomial time deterministic non-adaptive algorithm that makes*

$$\min\left(D^{o(1)}, 2^{O(\log^3(\log n))}\right) \cdot \frac{D}{\Delta^2}\log n$$

tests and outputs $\hat{d}$ such that $\frac{d}{\Delta} \leq \hat{d} \leq d\Delta$.

6.1   Algorithms Using Expanders

Let G be a bipartite graph $G = G(L, R, E)$ with left vertices $L = [n]$, right vertices $R = [m]$ and edges $E \subseteq L \times R$. For each edge $(i, j) \in E$, it holds that the endpoint $i \in L$ and $j \in R$. For a vertex $v \in L$, define $\Gamma(v)$ to be the set of the neighbours of v in G i.e. $\Gamma(v) := \{u \in R | (v, u) \in E\}$. For a subset $S \subseteq L$, we define $\Gamma(S)$ to be the set of neighbours of S, meaning $\Gamma(S) := \cup_{v \in S} \Gamma(v)$. For a vertex $v \in L$, the *degree* of v is defined as $deg(v) := |\Gamma(v)|$. We say that a bipartite graph $G = G(L, R, E)$ is a (k, a)-*expander δ-regular bipartite graph* if, the degree of every vertex in L is δ, and for every left-subset $S \subseteq L$ of size at most k, we have $|\Gamma(S)| \geq a|S|$.

Lemma 7. *Let $X = [n]$ be a set of items and $I \subseteq [n]$ is the set of defective items such that $|I| = d$ is unknown to the algorithm. Let $G = G(L, R, E)$ be a (k, a)-expander δ-regular bipartite graph with $|L| = n$ and $|R| = m$. Then, there is a deterministic non-adaptive algorithm A, such that for n items, it makes m tests and satisfies:*

1. If $|I| < ak/\delta$, then A outputs 0.
2. If $|I| \geq k$, then A outputs 1.

Proof. For every $j \in R$, we define the test $T^{(j)} = \{i | (i, j) \in E\}$. The number of tests is $|R| = m$. If $|I| \geq k$, then $|\Gamma(I)| \geq ak$. Therefore, at least ak tests will give positive answer 1. If $|I| < ak/\delta$, then, since the degree of every vertex in L is δ, we have $|\Gamma(I)| \leq \delta|I| < ak$. This shows that, for this case, at most $ak - 1$ tests give the answer 1. Hence, we can distinguish between the two cases.

Following the same proof of Lemma 3 with algorithm $\mathcal{T}$ in Fig. 1, we have:

Lemma 8. *Let $A(\ell, \Delta)$ be a deterministic non-adaptive algorithm such that, for n items, it makes $m(\ell, \Delta)$ tests and satisfies:*

1. If $|I| < \ell/\Delta^2$, then A outputs 0.
2. If $|I| \geq \ell/\Delta$, then A outputs 1.

Then, there is a deterministic non-adaptive algorithm $\mathcal{T}$ such that, given $D > d$, for n items it makes

$$\sum_{i=0}^{\lceil \log D / \log \Delta \rceil} m \left(\frac{D}{\Delta^i}, \Delta \right)$$

tests and outputs $\hat{d}$ that satisfies $d/\Delta \leq \hat{d} \leq \Delta d$.

The parameters of the explicit construction of a (k, a)-expander δ-regular bipartite graph from [4] are summarised in the following lemma.

Lemma 9. *For any $k > 0$ and $0 < \epsilon < 1$, there is an explicit construction of a (k, a)-expander δ-regular bipartite graph with*

$$m = O(k\delta/\epsilon), \quad \delta = 2^{O(\log^3(\log n/\epsilon))}, \quad a = (1 - \epsilon)\delta.$$

We now prove:

Lemma 10. *There is a polynomial time deterministic non-adaptive algorithm that makes*

$$\frac{D}{\Delta^2} \cdot 2^{O(\log^3(\log n))} = \frac{D}{\Delta^2} \cdot \text{quasipoly}(\log n)$$

tests and outputs $\hat{d}$ that satisfies

$$\frac{d}{\Delta} \leq \hat{d} \leq \Delta d.$$

Proof. We use the expander in Lemma 9. Recall that $\Delta = 1 + \Omega(1)$. Let $r = \min(\Delta, 2)$, $\epsilon = 1 - 1/r$ and $k = r\ell/\Delta^2$. Then $a = \delta/r = 2^{O(\log^3 \log n)}$ and $m = m(\ell, \Delta) = (\ell/\Delta^2)2^{O(\log^3 \log n)}$. By Lemma 7, there is a deterministic non-adaptive algorithm A such that for n items, it makes $m(\ell, \Delta)$ tests and

1. If $|I| < ak/\delta = \ell/\Delta^2$ then A outputs 0.
2. If $|I| \geq k = r\ell/\Delta^2$ then A outputs 1.

Algorithm A trivially satisfies the first condition required by Lemma 8. Consider item 2. If $\Delta < 2$ then $r = \Delta$ and then if $|I| \geq k = \ell/\Delta$ then A outputs 1. If $\Delta > 2$ then $r = 2$ and then if $|I| \geq k = 2\ell/\Delta^2$ then A outputs 1. Since $2\ell/\Delta^2 < \ell/\Delta$, if $|I| \geq \ell/\Delta$ then A outputs 1.

Now by Lemma 8, there is a deterministic non-adaptive algorithm $\mathcal{T}$ such that, given $D > d$, for n items, it makes

$$\sum_{i=0}^{\lceil \log D/ \log \Delta \rceil} m\left(\frac{D}{\Delta^i}, \Delta\right) = \frac{D}{\Delta^2} \cdot 2^{O(\log^3(\log n))}$$

tests and outputs $\hat{d}$ that satisfies $d/\Delta \leq \hat{d} \leq \Delta d$.

6.2   Algorithms Using Extractors and Condensers

Extractors are functions that convert weak random sources into almost-perfect random sources. We use these objects to construct a non-adaptive algorithm for estimating d. We start with some definitions.

Definition 1. *Let X be a random variable over a finite set S. We say that X has min-entropy at least k if $Pr[X = x] \leq 2^{-k}$ for all $x \in S$.*

Definition 2. *Let X and Y be random variables over a finite set S. We say that X and Y are $\epsilon - close$ if $\max_{P \subseteq S} |\Pr[X \in P] - \Pr[Y \in P]| \leq \epsilon$.*

We denote by U_ℓ the uniform distribution on $\{0,1\}^\ell$. The notations $\Pr_{x \in B}$ or $\mathbf{E}_{x \in B}$ stand for the fact that the probability and the expectation are taken when x is chosen randomly uniformly from B.

Definition 3. *A function $F : \{0,1\}^{\hat{n}} \times \{0,1\}^{\hat{t}} \rightarrow \{0,1\}^{\hat{m}}$ is a $k \rightarrow_\epsilon k'$ condenser if for every X with min-entropy at least k and Y uniformly distributed on $\{0,1\}^{\hat{t}}$, the distribution of $(Y, F(X, Y))$ is ϵ-close to a distribution $(U_{\hat{t}}, Z)$ with min-entropy $\hat{t} + k'$. A condenser is called (k, ϵ)-lossless condenser if $k' = k$. A condenser is called (k, ϵ)-extractor if $\hat{m} = k'$.*

Let $\hat{N} = \{0,1\}^{\hat{n}}, \hat{T} = \{0,1\}^{\hat{t}}$ and $\hat{M} = \{0,1\}^{\hat{m}}$, and let $F : \hat{N} \times \hat{T} \rightarrow \hat{M}$ be a $k \rightarrow_\epsilon k'$ condenser. Consider the $2^{\hat{t}} \times 2^{\hat{n}}$ matrix $\mathcal{M}$ induced by F. That is, for $r \in \hat{T}$ and $s \in \hat{N}$, the entry $\mathcal{M}_{r,s}$ is equal to $F(s,r)$. For $s \in \hat{N}$, let $\mathcal{M}^{(s)}$ be the sth column of $\mathcal{M}$. Then, $\mathcal{M}_r^{(s)} = \mathcal{M}_{r,s} = F(s,r)$.

Definition 4. *Let Σ be a finite set. An n-mixture over Σ is an $n-$tuple $\mathcal{S} := (S_1, \cdots, S_n)$ such that for all $i \in [n]$, $S_i \subseteq \Sigma$.*

Using these definitions and notations, we restate the result proved by Cheraghchi [6] (Theorem 9) in the following lemma.

Lemma 11. *Let $F : \{0,1\}^{\hat{n}} \times \{0,1\}^{\hat{t}} \to \{0,1\}^{\hat{m}}$ be a $k \to_\epsilon k'$ condenser. Let $\mathcal{M}$ be the matrix induced by F. Then, for any $2^{\hat{t}}-$mixture $\mathcal{S} = (S_1, \cdots, S_{2^{\hat{t}}})$ over $\hat{M} := \{0,1\}^{\hat{m}}$, the number of columns s in $\mathcal{M}$ that satisfies*

$$\Pr_{r \in \hat{T}}[\mathcal{M}_r^{(s)} \in S_r] > \frac{\mathbf{E}_{r \in \hat{T}}[|S_r|]}{2^{k'}} + \epsilon$$

is less than 2^k.

Equipped with Lemma 11, we prove:

Lemma 12. *If there is a $k \to_\epsilon k'$ condenser $F : \{0,1\}^{\hat{n}} \times \{0,1\}^{\hat{t}} \to \{0,1\}^{\hat{m}}$ then, there is a deterministic non-adaptive algorithm $\mathcal{A}$ for $n = 2^{\hat{n}}$ items that makes $m = 2^{\hat{t}+\hat{m}}$ tests and satisfies the following.*

1. *If the number of defectives is less than $(1 - \epsilon)2^{k'}$ then $\mathcal{A}$ outputs 0.*
2. *If the number of defectives is greater than or equal $2^k + 1$ then $\mathcal{A}$ outputs 1.*

Proof. Consider the matrix $\mathcal{M}$ induced by the condenser F as explained above. We define the test matrix $\mathcal{T}$ from $\mathcal{M}$ as follows. Let $x \in \{0,1\}^{\hat{m}}$. Define $e(x) \in \{0,1\}^{2^{\hat{m}}}$ such that $e(x)_y = 1$ if and only if $x = y$, where the bits in $e(x)$ are indexed by the elements of $\{0,1\}^{2^{\hat{m}}}$. Each row r in the matrix $\mathcal{M}$ is replaced by $2^{\hat{m}}$ rows (in $\mathcal{T}$) such that in each entry $\mathcal{M}_{r,s} \in \{0,1\}^{\hat{m}}$ is replaced by the column vector $e(\mathcal{M}_{r,s})^T \in \{0,1\}^{2^{\hat{m}}}$. The rows of the matrix $\mathcal{T}$ are indexed by $\hat{T} \times \hat{M}$. Let $\mathcal{T}^{(i)}$ denote the ith column of $\mathcal{T}$. Therefore, for $r \in \hat{T}$ and $j \in \hat{M}$, the row (r, j) in the matrix $\mathcal{T}$ is denoted by $\mathcal{T}_{(r,j)}$. Moreover, the ith entry of the row $\mathcal{T}_{(r,j)}$ is denoted by $\mathcal{T}_{(r,j),i}$ and $\mathcal{T}_{(r,j),i} = \mathcal{T}_{(r,j)}^{(i)} = 1$ if and only if $\mathcal{M}_{r,i} = j$. The size of the test matrix $\mathcal{T}$ is $m \times n$.

Let the defective elements be $s_{i_1}, \ldots, s_{i_\ell}$ and let $y \in \{0,1\}^m$ indicate the tests result. Then, y is equal to $\mathcal{T}^{(s_{i_1})} \vee \cdots \vee \mathcal{T}^{(s_{i_\ell})}$. Let $\mathcal{S} = (S_r)_{r \in \hat{T}}$ be a $2^{\hat{t}}-$mixture over $\{0,1\}^{\hat{m}}$ where for all $r \in \hat{T}$, $S_r = \{j \in \{0,1\}^{\hat{m}} | y_{(r,j)} = \mathcal{T}_{(r,j)}^{(s_{i_1})} \vee \cdots \vee \mathcal{T}_{(r,j)}^{(s_{i_\ell})} = 1\}$. It is easy to see that:

1. $|S_r| \le \ell$. This is because, by the definition of S_r, $j \in S_r$ if and only if $y_{(r,j)} = 1$. The entry $y_{(r,j)}$ gets the value 1 if at least one of the entries $\mathcal{T}_{(r,j)}^{(s_{i_1})}, \cdots, \mathcal{T}_{(r,j)}^{(s_{i_\ell})}$ is 1. Any row in $\mathcal{T}^{(s_{i_1})}, \cdots, \mathcal{T}^{(s_{i_\ell})}$ has exactly one entry that is equal to 1 in all the $2^{\hat{m}}$ rows indexed by r. Hence, each row can cause one item to be inserted to S_r.
2. For any $s_{i_j} \in \{s_{i_1}, \ldots, s_{i_\ell}\}$, we have $\Pr_{r \in \hat{T}}[\mathcal{M}_r^{(s_{i_j})} \in S_r] = 1$

3. Given the matrix $\mathcal{M}$, its test matrix $\mathcal{T}$ and the observed result y, for any column s the probability $\Pr_{r\in\hat{T}}[\mathcal{M}_r^{(s)} \in S_r]$ can be easily computed.

If the number of defectives is less than $(1 - \epsilon)2^{k'}$ then, by Lemma 11, all columns, except for at most 2^k columns, satisfy

$$\Pr_{r\in\hat{T}}[\mathcal{M}_r^{(s)} \in S_r] \leq \frac{\mathbf{E}_{r\in\hat{T}}[|S_r|]}{2^{k'}} + \epsilon < \frac{\mathbf{E}_{r\in\hat{T}}[(1 - \epsilon)2^{k'}]}{2^{k'}} + \epsilon = 1.$$

So for less than $2^k + 1$ columns we have $\Pr_{r\in\hat{T}}[\mathcal{M}_r^{(s)} \in S_r] = 1$. If the number of defectives is greater than or equal $2^k + 1$, then for the columns of the defectives we have $\Pr_{r\in\hat{T}}[\mathcal{M}_r^{(s)} \in S_r] = 1$. So for more than 2^k columns we have $\Pr_{r\in\hat{T}}[\mathcal{M}_r^{(s)} \in S_r] = 1$.

The following Lemma summarises the state of the art result due to Guruswami et. al. [15] on explicit construction of expanders.

Lemma 13. *For all positive integers $\hat{n}, k$ such that $\hat{n} \geq k$, and all $\epsilon > 0$, there is an explicit (k, ϵ) extractor $F : \{0,1\}^{\hat{n}} \times \{0,1\}^{\hat{t}} \rightarrow \{0,1\}^{\hat{m}}$ with $\hat{t} = \log \hat{n} + O(\log k \log (k/\epsilon))$ and $\hat{m} = k' = k - 2\log 1/\epsilon - c$ for some constant c.*

We now prove:

Lemma 14. *There is a constant C such that for every $\Delta > C$, there is a polynomial deterministic non-adaptive algorithm that estimates the number of defective items in a set of n items up to a multiplicative factor of Δ and asks*

$$O\left(\frac{D^{1+o(1)}}{\Delta^2} \log n\right)$$

queries.

Proof. We use the notations from Lemma 13. Let $C = 27 \cdot 2^{c-2}$. We choose $\epsilon = 2/3$ and k' such that $(1 - \epsilon)2^{k'} = \ell/\Delta^2$. Then

$$2^k = 2^{k'+2\log(1/\epsilon)+c} = 27 \cdot 2^{c-2}\frac{\ell}{\Delta^2} < \frac{\ell}{\Delta}$$

By Lemma 12, there is a deterministic non-adaptive algorithm $\mathcal{A}$ for $n = 2^{\hat{n}}$ items that makes

$$m = 2^{\hat{t}+\hat{m}} = \hat{n}2^{O(\log k \log(k/\epsilon))}\frac{\ell}{(1 - \epsilon)\Delta^2} = 2^{\log^2 \log(\ell/\Delta)}\frac{\ell}{\Delta^2}\log n$$

tests that satisfies the following:

1. If the number of defectives is less than $(1 - \epsilon)2^{k'} = \ell/\Delta^2$ then $\mathcal{A}$ outputs 0.
2. If the number of defectives is greater than or equal $2^k + 1$ then $\mathcal{A}$ outputs 1 and, since $2^k < \ell/\Delta$, in particular, if the number of defectives is greater than or equal ℓ/Δ then $\mathcal{A}$ outputs 1.

By Lemma 8, the result follows.

A similar work by Capalbo et al. [4] gives an explicit construction of a lossless condenser is summarised in the following lemma:

Lemma 15. *For all positive integers $\hat{n}, k$ and all $\epsilon > 0$, there is an explicit lossless condenser $F : \{0,1\}^{\hat{n}} \times \{0,1\}^{\hat{t}} \to \{0,1\}^{\hat{m}}$ with $\hat{t} = O(\log^3(\hat{n}/\epsilon))$ and $\hat{m} = k + \log(1/\epsilon) + O(1)$.*

The construction from Lemma 15 yields a result that is similar to the one established in Lemma 14.

References

1. Agarwal, A., Flodin, L., Mazumdar, A.: Estimation of sparsity via simple measurements. In: 2017 IEEE International Symposium on Information Theory (ISIT), pp. 456–460. IEEE (2017)
2. Bshouty, N.H.: Lower bound for non-adaptive estimation of the number of defective items. In: 30th International Symposium on Algorithms and Computation, ISAAC 2019, 8–11 December 2019, Shanghai University of Finance and Economics, Shanghai, China, pp. 2:1–2:9 (2019). https://doi.org/10.4230/LIPIcs.ISAAC.2019.2
3. Bshouty, N.H., Bshouty-Hurani, V.E., Haddad, G., Hashem, T., Khoury, F., Sharafy, O.: Adaptive group testing algorithms to estimate the number of defectives. CoRR abs/1712.00615 (2017). http://arxiv.org/abs/1712.00615
4. Capalbo, M., Reingold, O., Vadhan, S., Wigderson, A.: Randomness conductors and constant-degree expansion beyond the degree/2 barrier, January 2002
5. Cheng, Y., Xu, Y.F.: An efficient FPRAS type group testing procedure to approximate the number of defectives. J. Combin. Optim. **27**, 302–314 (2014)
6. Cheraghchi, M.: Noise-resilient group testing: limitations and constructions. CoRR abs/0811.2609 (2008). http://arxiv.org/abs/0811.2609
7. Chen, C.L., Swallow, W.H.: Using group testing to estimate a proportion, and to test the binomial model. Biometrics **46**(4), 1035–1046 (1990)
8. Cormode, G., Muthukrishnan, S.: What's hot and what's not: tracking most frequent items dynamically. ACM Trans. Database Syst. **30**(1), 249–278 (2005). https://doi.org/10.1145/1061318.1061325
9. Damaschke, P., Muhammad, A.S.: Bounds for nonadaptive group tests to estimate the amount of defectives. In: Wu, W., Daescu, O. (eds.) COCOA 2010. LNCS, vol. 6509, pp. 117–130. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17461-2_10
10. Damaschke, P., Sheikh Muhammad, A.: Competitive group testing and learning hidden vertex covers with minimum adaptivity. In: Kutyłowski, M., Charatonik, W., Gębala, M. (eds.) FCT 2009. LNCS, vol. 5699, pp. 84–95. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03409-1_9
11. Dorfman, R.: The detection of defective members of large populations. Ann. Math. Stat. **14**(4), 436–440 (1943)
12. Falahatgar, M., Jafarpour, A., Orlitsky, A., Pichapati, V., Suresh, A.: Estimating the number of defectives with group testing, pp. 1376–1380, July 2016. https://doi.org/10.1109/ISIT.2016.7541524

13. Gastwirth, J.L., Hammick, P.A.: 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. J. Stat. Plann. Inference **22**(1), 15–27 (1989). https://doi.org/10.1016/0378-3758(89)90061-X. http://www.sciencedirect.com/science/article/pii/037837588990061X
14. Gollier, C., Gossner, O.: Group testing against Covid-19. Covid Econ. 32–42 (2020)
15. Guruswami, V., Umans, C., Vadhan, S.: Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes. In: Twenty-Second Annual IEEE Conference on Computational Complexity (CCC 2007), pp. 96–108 (2007)
16. Hong, E.S., Ladner, R.E.: Group testing for image compression. IEEE Trans. Image Process. **11**(8), 901–911 (2002)
17. Kautz, W., Singleton, R.: Nonrandom binary superimposed codes. IEEE Trans. Inf. Theory **10**(4), 363–377 (1964)
18. Li, C.: A sequential method for screening experimental variables. J. Am. Stat. Assoc. - J AMER STATIST ASSN **57**, 455–477 (1962). https://doi.org/10.1080/01621459.1962.10480672
19. Mentus, C., Romeo, M., DiPaola, C.: Analysis and applications of adaptive group testing methods for Covid-19. medRxiv (2020). https://doi.org/10.1101/2020.04.05.20050245. https://www.medrxiv.org/content/early/2020/04/16/2020.04.05.20050245
20. Ngo, H., Du, D.Z.: A survey on combinatorial group testing algorithms with applications to DNA library screening. DIMACS Ser. Discrete Math. Theor. Comput. Sci. **55**, 171–182 (2000). https://doi.org/10.1090/dimacs/055/13
21. Ron, D., Tsur, G.: The power of an example: hidden set size approximation using group queries and conditional sampling. CoRR abs/1404.5568 (2014). http://arxiv.org/abs/1404.5568
22. Sobel, M., Groll, P.A.: Group testing to eliminate efficiently all defectives in a binomial sample. Bell Syst. Tech. J. **38**(5), 1179–1252 (1959)
23. Swallow, W.H.: Group testing for estimating infection rates and probabilities of disease transmission. Phytopathology (USA) (1985)
24. Thompson, K.H.: Estimation of the proportion of vectors in a natural population of insects. Biometrics **18**(4), 568–578 (1962)
25. Walter, S.D., Hilderth, S.W., Beaty, B.J.: Estimation of infection rates in populations of organisms using pools of variable size. Am. J. Epidemiol. **112**(1), 124–128 (1980). https://doi.org/10.1093/oxfordjournals.aje.a112961
26. Wolf, J.: Born again group testing: multiaccess communications. IEEE Trans. Inf. Theory **31**(2), 185–191 (1985)
27. Yelin, I., et al.: Evaluation of Covid-19 RT-qPCR test in multi-sample pools. medRxiv (2020). https://doi.org/10.1101/2020.03.26.20039438. https://www.medrxiv.org/content/early/2020/03/27/2020.03.26.20039438