# GT Lecture 3

Oct. 7

**Adaptive Algorithm for $d=2$ case** (Idea from Jinn Lu)
呂晉

Let $F_n = \lceil \log_2 \binom{n}{2} \rceil$ (Informatic lower bound).

(•) By Frank K. Hwang's Generalized algorithm, we have

$$M(2,n) = F_n \text{ or } F_n + 1.$$

(••) We can use Lu's algorithm to determine $M(2,n)$ for more values $n$.

(•••) Let $n$ be a positive integer. Then, there exists a $k$ such that $\underset{①}{n \le 10 \cdot 2^k}$ or $\underset{②}{10 \cdot 2^k < n \le 14 \cdot 2^k}$. For example, if $n = 100$, $n \le 14 \cdot 2^3$; $n = 75$, $n \le 10 \cdot 2^3$.

**Fact 1** If $n_1 \le n_2$, then $M(d,n_1) \le M(d,n_2)$.

(*) For ①, $M(2,n) \le M(2, 10\cdot 2^k)$, and for ②, $M(2,n) \le M(2, 14 \cdot 2^k)$.

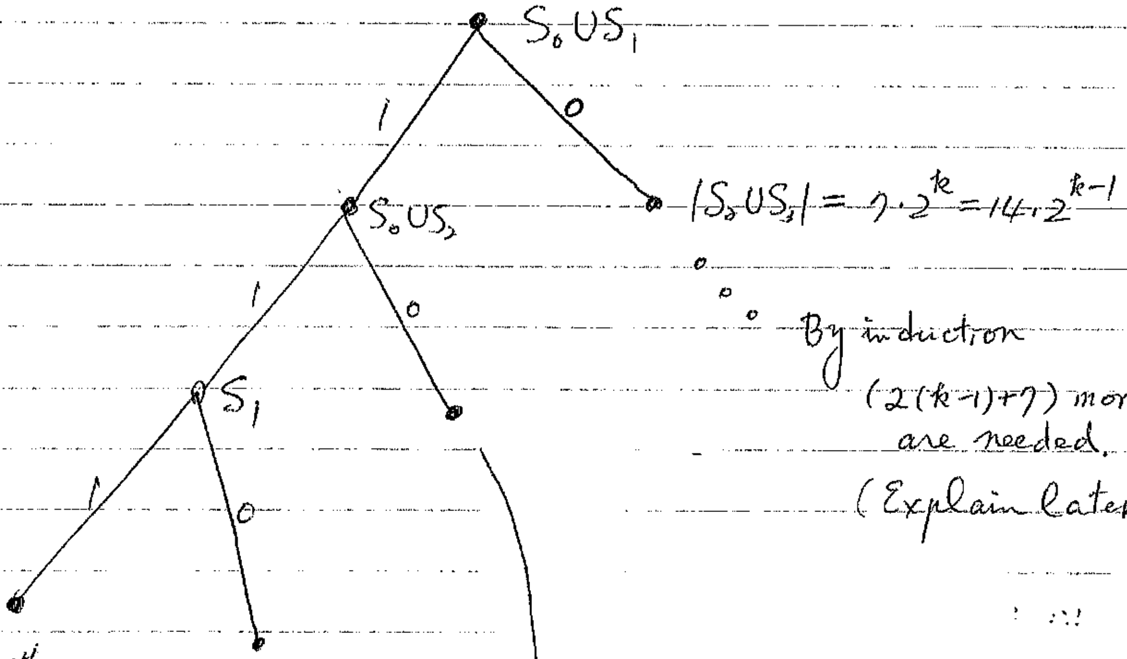
**Fact 2.** $M(2, 10\cdot 2^k) \le 2k+6$ and $M(2, 14\cdot 2^k) \le 2k+7$.

**Step 1.** Partition $[1, 10\cdot 2^k]$ into four sets $S_0, S_1, S_2$ and $S_3$ such that $|S_0| = 2^k$, $|S_1| = |S_2| = 2 \cdot 2^k$ and $|S_3| = 5 \cdot 2^k$.

**Step 2.** We use the following testing tree to run the algorithm.

## $n = 10 \cdot 2^k$ case

(Initial cases ~~are in 2'~~)
$n = 10, 14$



~~By induction~~
$(2(k-1)+7)$ more tests are needed.
(Explain later).

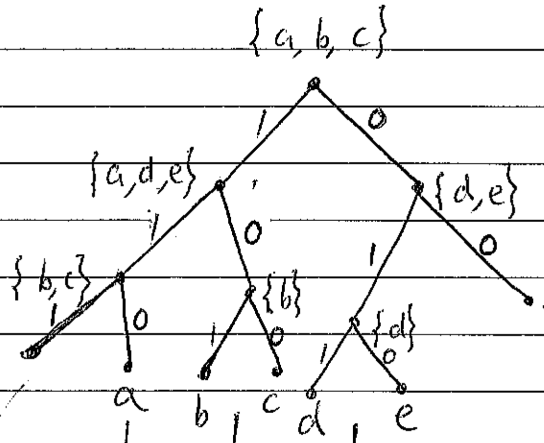
Since both $S_1$ and $S_0 \cup S_2$ are positive, find one for each, It takes $k+1$ and $k+2$ tests respectively.

$S_1$ is positive, it takes $k+1$ tests to find the first positive and $\lceil \log_2 7 \cdot 2^k \rceil = k+3$ tests to find the second positive. ($2k+4$ in total.)

Find one positive in $S_0$, it takes $k$ tests. Then, find another one in $S_0 \cup S_2 \cup S_3$, it takes $\lceil \log_2 8 \cdot 2^k \rceil = k+3$ tests to get the job done. ($2k+3$ in total.)

As a conclusion, the worst case would take $2k+6$ tests to find the two positives. For example, $k=3$, $\lceil \log_2 \binom{80}{2} \rceil = 12$. This implies that $M(2,80) = 12$.

$n=10, \quad N=\{\overbrace{a}^{S_0},\overbrace{b,c}^{S_1},\overbrace{d,e}^{S_2},\overbrace{f,g,h,i,j}^{S_3}\}$



→ 4 tests to find 2 defectives from {f, g, h, i, j}. (Individual tests).

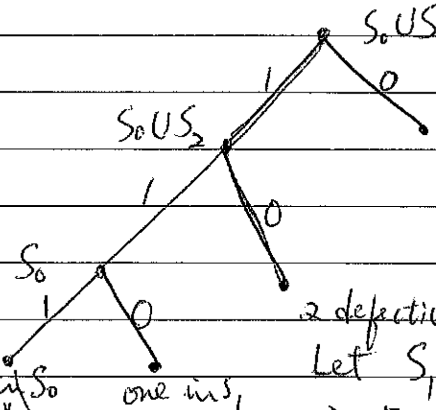
a → 3 tests to find one from {d, e, ⋯, i, j}.

b, c, d, e → 3 tests to find another 1 (from 6 items)

→ Find one in {b, c} (1 test) and one in {a, d, e} (2 tests)

$\left\lceil \log_2 \binom{10}{2} \right\rceil = 6 \Rightarrow M(2,10)=6.$

$n=14 \qquad N=\{\overbrace{a}^{S_0},\overbrace{b,c,d}^{S_1},\overbrace{e,f,g}^{S_2},\overbrace{h,i,j,k,l,m,n}^{S_3}\}$



find two defectives in 6 more tests by $n=10$

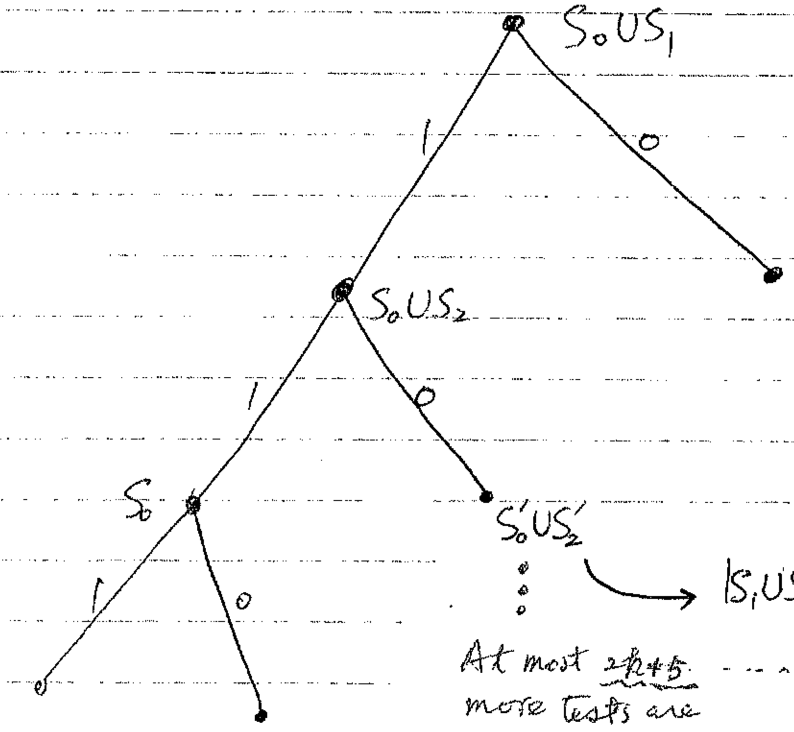
2 defectives are in $S_1 \cup S_3$. (*)
Let $S_1 = S_0' \cup S_1'$ and $S_3 = S_2' \cup S_3'$. $S_1$ is positive.
⇒ 5 more tests are needed in above case.

↓ The other one in $N \setminus \{a\}$, (4 more tests)

$\left\lceil \log_2 \binom{14}{2} \right\rceil = \lceil \log_2 91 \rceil = 7, \Rightarrow M_2(2,14)=7.$

**Fact 2.** $n = 14 \cdot 2^k$ Case
(Continued)

$|S_0| = 2^k$, $|S_1| = |S_2| = 3 \cdot 2^k$ and $|S_3| = 7 \cdot 2^k$



The two defectives are in $S_2 U S_3$. $|S_2 U S_3| = 10 \cdot 2^k$. By induction, it takes $2k+6$ more Tests.

$|S_1 U S_3| \doteqdot 10 \cdot 2^k$

$S_1 U S_3 \Rightarrow \underline{S_0' U S_1'} U S_2' U S_3'$ positive $(S_1)$

$|S_0'| = 2^k$
$|S_1'| = 2 \cdot 2^k$
$|S_2'| = 2 \cdot 2^k$
$|S_3'| = 5 \cdot 2^k$

At most $\underline{2k+5}$ more tests are needed.
(The 1st test of $10 \cdot 2^k$ case has been done)

one in $S_1$ and one in $S_2$ (已經知道是 "positive.)

one in $S_0$ and one in $S_0 U S_1 U S_2 U S_3$.
$k$ $k+4$

The worst case would take $2k+7$ tests to find the two positives. $k = 3$, $M(2, 112) = 13$.

$$M(2, n) \leq \begin{cases} 2k+6, & \text{if } 14 \times 2^{k-1} < n \leq 10 \times 2^k, \text{ and} \\ 2k+7, & \text{if } 10 \times 2^k < n \leq 14 \times 2^k. \end{cases}$$

## Examples (Use $10 \cdot 2^k$)

Coro. $\left\lceil \log_2 \binom{10 \cdot 2^k}{2} \right\rceil \geq 2k+6 \Rightarrow M(2, 10 \cdot 2^k) = 2k+6.$

Coro. For $7 \cdot 2^k < n \leq 10 \cdot 2^k$, $\left\lceil \log_2 \binom{n}{2} \right\rceil \geq 2k+6 \Rightarrow M(2,n) = 2k+6.$

Proof. $M(2,n) \leq M(2, 10 \cdot 2^k) = 2k+6.$

$$\left\lceil \log_2 \binom{80}{2} \right\rceil = 12.$$

$$\left\lceil \log_2 \binom{79}{2} \right\rceil = \lceil \log_2 79 \times 39 \rceil = \lceil \log_2 3071 \rceil = \lceil \log_2 4096 \rceil = 12$$

$$\left\lceil \log_2 - \right\rceil = 12 \quad \left\{ \begin{array}{l} \binom{80}{2} = 3160 \\ \binom{79}{2} = 3071 \\ \binom{78}{2} = 3003 \\ \binom{77}{2} = 2926 \\ \vdots \\ \binom{70}{2} = 2415 \\ \vdots \\ \binom{65}{2} = 2080 \end{array} \right.$$

Hence, for $n = 65, 66, \cdots, 80$, $M(2,n) = 12.$

## Examples ( Use $14\cdot 2^k$)

$\lceil \log_2 \binom{112}{2} \rceil = 13 \Rightarrow M(2,112) = 13$

$\lceil \log_2 \binom{100}{2} \rceil = \lceil \log_2 4950 \rceil = 13$

$\lceil \log_2 \binom{92}{2} \rceil = 13$

$\Rightarrow$ For $n = 92, 93, \cdots, 112$, $M(2,n) = 13$

Solution to $n = 1,000$. —

$$14\cdot 2^6 < 1,000 \le 10\cdot 2^7, \quad k = 7, \quad \overset{\overset{2\times 6+7}{\nearrow}}{19 \le} M(2,1,000) \le \overset{2\times 7+6}{20''}$$

$\lceil \log_2 \binom{1000}{2} \rceil = 19$.

So, $19 \le M(2,1000) \le 20$.

(○) Let $i_t$ denote the integer such that

$$\binom{i_t}{2} < 2^t < \binom{i_t+1}{2}.$$

( $i_t$ is the largest number of items with 2 defectives which can be solved in $t$ tests.)

Then, $i_t = \lfloor 2^{\frac{t+1}{2}} - \frac{1}{2} \rfloor + 1$.

Proof. $\lfloor 2^{\frac{t+1}{2}} - \frac{1}{2} \rfloor + 1 < 2^{\frac{t+1}{2}} + \frac{1}{2} < \lfloor 2^{\frac{t+1}{2}} - \frac{1}{2} \rfloor + 2$

$$\binom{2^{\frac{t+1}{2}} + \frac{1}{2}}{2} = \frac{(2^{\frac{t+1}{2}} + \frac{1}{2})(2^{\frac{t+1}{2}} - \frac{1}{2})}{2} = \frac{2^{t+1} - \frac{1}{4}}{2} = 2^t - \frac{1}{8}$$

The proof follows by the fact that $\binom{i}{2} = 2^t$ has no solution for $i \in Z$.

e.g. $t = 9$

$i_t = 32$ ( $\binom{32}{2} = 496$, $\binom{33}{2} = 528$, $2^9 = 512$ )

(*) $M(2,33) > 9$ or $M(2,33) \ge \lceil \log_2 \binom{33}{2} \rceil = 10$.

從固定 "test"$_{(t)}$ 數量來看最多可以解決的 $(d,n)$-problem 中的 $n$,

$n_t$. 則 $i_t \ge n_t$ ( $(2,n)$-problem )。

(**) $n_t$ 不可能大到與 $i_t$ 相等, $i_t - 1 \ge n_t$ ( $d=2$ case ).

## $d \doteq 3$ Case (Three possible outcomes)

(•) ∀ $n$, ∃ $k$, s.t. either $n \underset{\textcircled{1}}{\le} 19\cdot 2^k$ or $n \underset{\textcircled{2}}{\le} 24\cdot 2^k$ or $n \underset{\textcircled{3}}{\le} 30\cdot 2^k$.

(••) It takes $3k+11$, $3k+12$ and $3k+13$ tests to find the three positives respectively in ①, ② and ③.

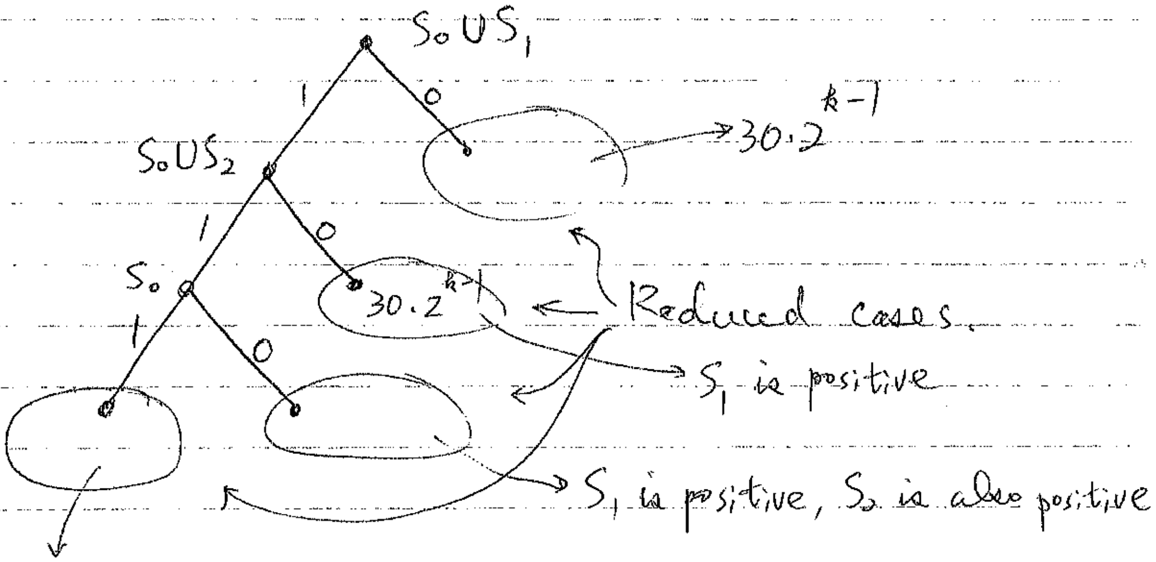
✓ Initial cases $\qquad S_0 \quad S_1 \quad S_2 \quad S_3$.

$n \le 19\cdot 2^k$, Use $19\cdot 2^k = \underline{1}\cdot 2^k + \underline{3}\cdot 2^k + \underline{3}\cdot 2^k + \underset{3(}{\underline{\underline{12}}}\cdot 2^k$,

$n \le 24\cdot 2^k$, Use $24\cdot 2^k = \underline{1}\cdot 2^k + \underline{4}\cdot 2^k + \underline{4}\cdot 2^k + \underset{4(}{\underline{\underline{15}}}\cdot 2^k$,

$n \le 30\cdot 2^k$, Use $30\cdot 2^k = \underline{1}\cdot 2^k + \underline{5}\cdot 2^k + \underline{5}\cdot 2^k + \underline{19}\cdot 2^k$.

The test-procedure is similar.



$S_0 \longrightarrow$ find one positive

$\underline{S_0 \cup S_1 \cup S_2 \cup S_3} \longrightarrow$ find two positives

$$M(3,n) \le \left\{ \begin{array}{ll} 3k+11, & \text{if } 30\times 2^{k-1} < n \le 19\times 2^{k}, \\ 3k+12, & \text{if } 19\times 2^{k} < n \le 24\times 2^{k}, \text{ and} \\ 3k+13, & \text{if } 24\times 2^{k} < n \le 30\times 2^{k}. \end{array} \right) \text{3 intervals.}$$

$$\Uparrow$$

$$M_L(3,n)$$

(*) For $n > 19$, $\lceil \log_2 \binom{n}{3} \rceil \le M_L(3,n) \le \lceil \log_2 \binom{n}{3} \rceil + 1$.

(**) $M(d,n) \le M_L(d,n) \le \lceil \log_2 \binom{n}{d} \rceil + d-2$.

$$\uparrow$$

$d-1$ in Hwang's Algorithm.

<u>Remark</u> In general $d$ case, we consider.. $n$ in $d$ intervals,

Forexample, $k=1$, $M(3,38) = 14$ $\left( \lceil \log_2 \binom{38}{3} \rceil = \lceil \log_2 8436 \rceil = 14 \right)$.

$M(3,48) = 15$ $\left( \lceil \log_2 \binom{48}{3} \rceil = \lceil \log_2 16{,}364 \rceil = 15 \right)$.

$M(3,60) = 16$ $\left( \lceil \log_2 \binom{60}{2} \rceil = \lceil \log_2 34{,}220 \rceil = 16 \right)$.

(*) For $d=4$, the intervals are

$$\begin{tikzcd}
(52\times 2^{k-1}, 31\times 2^{k}], \arrow[d] & (31\times 2^{k}, 37\times 2^{k}], \arrow[d] & (37\times 2^{k}, 44\times 2^{k}], \arrow[d] & (44\times 2^{k}, 52\times 2^{k}]. \arrow[d] \\
1,\underline{7,7},37 \arrow[rrr, dash, bend right, "6"'] & 1,\underline{4,4},22 \arrow[r, dash, bend right, "4"'] & 1,\underline{5,5},26 \arrow[r, dash, bend right, "5"'] & 1,\underline{6,6},31
\end{tikzcd}$$

Journal of Combinatorial Optimization (2021) 41:73–79
https://doi.org/10.1007/s10878-020-00661-6

# Quaternary splitting algorithm in group testing

Jinn Lu[1] · Hung-Lin Fu[1]

Accepted: 14 October 2020 / Published online: 4 November 2020

## Abstract

In Classical group testing, one is given a population of $n$ items $N$ which contains some defective $d$ items inside. A group test (pool) is a test on a subset of $N$. Under the circumstance of no errors, a test is negative if the testing pool contains no defective items and the test is positive if the testing pool contains at least one defective item but we don't know which one. The goal is to find all defectives by using as less tests as possible, mainly to minimize the number of tests (in the worst case situation). Let $M(d, n)$ denote the minimum number of tests in the worst case situation where $|N| = n$ and $d$ is the number of defectives. In this paper, we focus on estimating $M(d, n)$ and obtain a better result than known ones in various cases of $d$ and $n$.

**Keywords** Group testing · Adaptive algorithm · Quaternary splitting

## 1 Introduction

The group testing problem has been developed since 1943 during the World War II (Dorfman 1943). The motivation was to detect syphilis among large number of blood samples via blood tests. Even the idea was not actually carried out, nowadays, it has been applied to many fields such as chemical leaking testing, codes, multi-access channel communication (Wolf 1985), data compression (Hong and Ladner 2002), compressed sensing (Donoho 2006), network security (Thai 2012), multimedia fingerprinting (Cheng and Miao 2011), and DNA library screening (Bruno et al. 1995; Farach et al. 1997). Various revised models are also considered in Du and Hwang (2000), Hu et al. (1981), Torney (1999) and Weng and Hwang (1993) to fit the real situation of applications, such as complex models, inhibitor models, and threshold models.

---

✉ Jinn Lu
jinnlu.am05g@g2.nctu.edu.tw

Hung-Lin Fu
hlfu@math.nctu.edu.tw

[1] Department of Applied Mathematics, National Chiao Tung University, 30010 HsinChu, Taiwan

Given a set $N$ of $n$ items and each item can be either defective or pure. A group test (or test in short) will be on a set $S \subset N$, denoted by $S$ for convenience. Under the circumstance of no errors, the outcome of $S$ is positive (denoted by $P$) if it contains at least one defective item and the outcome is negative ($N$) if it contains all pure items. The classical group testing problem is to use less tests as possible to find all defective items. Mainly, for all algorithms, the goal is to minimize the number of tests in the worst case situation.

Let $M_G(d, n)$ denote the minimum number of tests which are used in algorithm "G" on a set $N$ of $n$ items and there are $d$ defective items. Moreover, let $M(d, n) = \min\{M_G(d, n) | G$ is an algorithm of group testing$\}$. So, as a consequence, $M_G(d, n) \geq M(d, n)$.

In general, there are two basic algorithms to tackle the problem: adaptive and non-adaptive algorithms. The tests designed in an adaptive algorithm may depend on the outcome of previous tests. But, in a non-adaptive algorithm all tests are carried out simultaneously. Therefore, an adaptive algorithm takes less tests in determining all the defectives in most of the cases. In this paper, we shall focus on applying adaptive algorithms.

The following facts are well-known.

**Proposition 1.1** $M(d, n) \geq \left\lceil \log_2 \binom{n}{d} \right\rceil$.

This is known as the information lower bound. By using binary search, we have

**Proposition 1.2** $M(1, n) = \lceil \log_2 n \rceil$.

Surprisingly, determining $M(2, n)$ for general $n$ remains unsettled at this moment (Chang and Hwang 1981; Chang et al. 1989; Du and Hwang 2000). Based on the generalized binary splitting algorithm developed by in Hwang (1972), we can prove the following result.

**Proposition 1.3** *(Hwang 1972) Let $H$ be the generalized binary splitting algorithm. Then, $M_H(d, n) - \left\lceil \log_2 \binom{n}{d} \right\rceil \leq d - 1$ for $d \geq 2$.*

Indeed, this result provides a good estimation of $M(d, n)$ by way of the information lower bound.

In this paper, we propose an algorithm to estimate $M(d, n)$. Mainly, we prove that $M(d, n) - \left\lceil \log_2 \binom{n}{d} \right\rceil \leq d - 2$ for $3 \leq d \leq 5$. For larger $d$, a similar idea can be utilized, but it is getting more tedious even it can be done.

## 2 Results

We shall use $M_L(d, n)$ to denote the number of tests needed in algorithm "L" (depending on $d$) for finding $d$ defectives among $n$ items. Since we need $d = 2$ case for finding small cases, $M_L(2, n)$ will be introduced first.

The following results are easy to see.

**Lemma 2.1** $M(2, 5) = M_L(2, 5) = 4$ *and* $M(2, 7) = M_L(2, 7) = 5$.

**Proposition 2.2** $M_L(2, 10 \times 2^k) = 2k + 6$ *and* $M_L(2, 14 \times 2^k) = 2k + 7$, $k \geq 0$.

***Proof*** We shall prove these two facts by induction alternatively on $k \geq -1$. By Lemma 2.1, the assertion is true for $k = -1$. Now, consider $k \geq 0$ and assume the inductive hypothesis holds for small $k$, i.e., $M_L(2, 10 \times 2^{k-1}) = 2k+4$ and $M_L(2, 14 \times 2^{k-1}) = 2k + 5$.

First, we partition $N$ into 4 subsets respectively, $N = S_0 \cup S_1 \cup S_2 \cup S_3$. If $N = 10 \times 2^k$, then let $|S_0| = 2^k$, $|S_1| = |S_2| = 2 \cdot 2^k$ and $|S_3| = 5 \cdot 2^k$, and if $N = 14 \times 2^k$, then $|S_0| = 2^k$, $|S_1| = |S_2| = 3 \cdot 2^k$ and $|S_3| = 7 \cdot 2^k$. Let $T_i$ denote the $i^{\text{th}}$ test and $L(T_i)$ be its outcome.

Let $T_1 = S_0 \cup S_1$ be the first test for $N = 10 \times 2^k$. If $L(T_1) = N$, then the defectives are in $S_2 \cup S_3$. Since $|S_2 \cup S_3| = 7 \cdot 2^k = 14 \cdot 2^{k-1}$, the proof follows by induction, $1 + 2(k - 1) + 7 = 2k + 6$. On the other hand, if $L(T_1) = P$, then let $T_2 = S_0 \cup S_2$. Now, if $L(T_2) = N$, then all the defectives are in $S_1 \cup S_3$. By finding one defective in $S_1$ (using binary splitting test) and one defective in $S_1 \cup S_3$ ($k + 3$ tests) we use at most $2k + 6$ tests. So, consider $L(T_2) = P$. Let $T_3 = S_1$. If $L(T_3) = N$, then find one defective in $S_0$ first by using $k$ tests and find the other one in $S_0 \cup S_2 \cup S_3$ by at most $k + 4$ tests. In total, at most $2k + 6$ tests are used. On the other hand, if $L(T_3) = P$, then find one defective in $S_1$ by using at most $k + 1$ tests and on defective in $S_0 \cup S_2$ with at most $k + 3$ tests. Hence, at most $2k + 6$ tests are needed.

Consider $N = 14 \times 2^k$. Now, the process can be carried out by a similar argument. First if $L(S_0 \cup S_1) = N$, then we use $2k + 6$ tests for finding two defectives in $S_2 \cup S_3$. If $L(S_0 \cup S_1) = P$, then test $S_0 \cup S_2$. Now, in case that $L(S_0 \cup S_2) = N$, then all defectives are in $S_1 \cup S_3$, moreover $L(S_1) = P$. Hence, let $S_1 = S'_0 \cup S'_1$ and $S_3 = S'_2 \cup S'_3$ such that $|S'_0| = 2^k$, $|S'_1| = |S'_2| = 2 \times 2^k$ and $S'_3 = 5 \times 2^k$. By previous argument, we start from testing $|S'_0 \cup S'_2|$ to find all the defectives in at most $2k + 5$ tests. If $L(S_0 \cup S_2) = P$, then let $T_3 = S_1$. Again, if $L(T_3) = N$, then all defectives are in $S_0 \cup S_2 \cup S_3$. By finding one in $S_0$ and one in $S_0 \cup S_2 \cup S_3$, we have the proof. On the other hand, if $L(T_3) = P$, then, one in $S_1$ and the other one in $S_0 \cup S_2$. This concludes the proof. □

In summary, we have

**Proposition 2.3** $M(2, n) \leq \begin{cases} 2k + 6, & \textit{if } 14 \times 2^{k-1} < n \leq 10 \times 2^k, \textit{ and} \\ 2k + 7, & \textit{if } 10 \times 2^k < n \leq 14 \times 2^k. \end{cases}$

So, if $14 \times 2^{k-1} < n \leq 10 \times 2^k$, $\lceil \log_2 \binom{n}{2} \rceil \geq \left\lceil \log_2 \binom{14 \times 2^{k-1}+1}{2} \right\rceil \geq \lceil \log_2 (7 \times 2^{k-1} \cdot (14 \times 2^{k-1} + 1)) \rceil \geq \lceil \log_2 98 \cdot 2^{2k-2} \rceil = \lceil \log_2 98 \rceil + (2k - 2) = 2k + 5$. On the other hand, if $10 \times 2^k < n \leq 14 \times 2^k$, then $\lceil \log_2 \binom{n}{2} \rceil \geq \left\lceil \log_2 \binom{10 \times 2^k+1}{2} \right\rceil \geq \lceil \log_2 50 \rceil + 2k = 2k + 6$. In any case, $\lceil \log_2 \binom{n}{2} \rceil \leq |M(2, n)| \leq \lceil \log_2 \binom{n}{2} \rceil + 1$.

Now, consider $d = 3$. The following results are the initial cases of our algorithm.

**Lemma 2.4** $M_L(3, 12) = 9$ *and* $M_L(3, 15) = 10$.

***Proof*** For $n = 12$, test two items first ($T_1$). If $L(T_1) = P$, then it takes one more test to find one defective, and seven other tests to find the others since $M_L(2, 11) \leq 7$. On the other hand, if $L(T_1) = N$, then take another two items for $T_2$ to start. Again, if $L(T_2) = P$, we need one more test for one defective and six more tests for the others since $M_L(2, 9) \leq 6$. Finally, if $L(T_2) = N$, then we can use seven individual tests to find all three defectives. Hence, $M_L(3, 12) = 9$.

For $n = 15$, partition the set of 15 items into four subsets $S_0$, $S_1$, $S_2$ and $S_3$ such that $|S_0| = 1$, $|S_1| = |S_2| = 2$ and $|S_3| = 12$. First, test $S_0 \cup S_1$ ($T_1$). If $L(T_1) = N$, then go to the case $n = 12$, and thus $M_L(3, 15) = 10$. On the other hand, if $L(T_1) = P$, then take $T_2 = S_0 \cup S_2$. If $L(T_2) = N$, then there exists one defective in $S_1$. So, we can find this defective first and the others by $M_L(2, 14)$. Otherwise, if $L(T_2) = P$, then take $T_3 = S_0$. Again, if $L(T_3) = P$, then the proof follows by using $M_L(2, 14) \leq 7$. Finally, if $L(T_3) = N$, then find one defective in $S_1$ and one defective in $S_2$ with two more tests, and four more tests for the last defective. In total, we use at most 10 tests, thus $M_L(3, 15) = 10$. □

**Lemma 2.5** $M_L(3, 19) = 11$.

***Proof*** Partition the set of 19 items into four subsets, $S_0$, $S_1$, $S_2$ and $S_3$ such that $|S_0| = 1$, $|S_1| = |S_2| = 3$ and $|S_3| = 12$. Let $T_1 = S_0 \cup S_1$. If $L(T_1) = N$, then by $M_L(3, 15) = 10$, we have the proof. On the other hand, if $L(T_1) = P$, then take $T_2 = S_0 \cup S_2$. If $L(T_2) = N$, then find one defective in $S_1$ and two defectives in 14 items (exclude $S_0 \cup S_2$). Now, if $L(T_2) = P$, then let $T_3 = S_0$ and consider whether $T_3$ is $N$ or $P$.

If $L(T_3) = N$, then there is at least one defective in $S_1$ and at least one defective in $S_2$. Now, test two items in $S_1$ and $S_2$ respectively. Following the outcomes, we need at $2 + 2 + 4$ tests in $S_1$, $S_2$ and $S_1 \cup S_2 \cup S_3$ (16 items), respectively. Hence, in total, 11 tests are needed.

Finally, if $L(T_3) = P$, then there are two defectives in the set of the other 18 items. By Proposition 2.3, we have the desired number of tests. □

**Proposition 2.6** *For* $k \geq 0$, $M_L(3, 19 \times 2^k) = 3k + 11$, *and for* $k \geq -1$, $M_L(3, 24 \times 2^k) = 3k + 12$ *and* $M_L(3, 30 \times 2^k) = 3k + 13$.

***Proof*** By induction on $k$, and the assertion is true for small cases by Lemmas 2.4 and 2.5. Now, we claim $M_L(3, 19 \times 2^k) = 3k + 11$ and the other two equalities can be obtained by similar arguments.

The proof follows by partitioning the set of items into four subsets $S_0$, $S_1$, $S_2$ and $S_3$ such that $|S_0| = 2^k$, $|S_1| = |S_2| = 3 \cdot 2^k$ and $|S_3| = 12 \cdot 2^k$. Then, following the same argument used in Lemma 2.5, we conclude the proof of $M_L(19 \times 2^k) = 3k + 11$. For the case of $M_L(3, 24 \times 2^k)$, we partition the set of items into subsets $S_0$, $S_1$, $S_2$ and $S_3$ such that $|S_0| = 2^k$, $|S_1| = |S_2| = 4 \times 2^k$ and $|S_3| = 15 \times 2^k$. Finally, if $N = 30 \times 2^k$, then the four subsets $S_0$, $S_1$, $S_2$ and $S_3$ are of sizes $2^k$, $5 \times 2^k$, $5 \times 2^k$ and $19 \times 2^k$ respectively. □

A corollary of Proposition 2.6 shows that $M(3, n) \leq M_L(3, n)$ for respective cases. More precisely we have

**Proposition 2.7** $M(3, n) \leq \begin{cases} 3k + 11, & \textit{if } 30 \times 2^{k-1} < n \leq 19 \times 2^k, \\ 3k + 12, & \textit{if } 19 \times 2^k < n \leq 24 \times 2^k, \textit{ and} \\ 3k + 13, & \textit{if } 24 \times 2^k < n \leq 30 \times 2^k. \end{cases}$

This result enables us to provide a better estimation of $M(3,n)$.

**Proposition 2.8** *For* $n \geq 19$, $\lceil \log_2 \binom{n}{3} \rceil \leq M(3,d) \leq \lceil \log_2 \binom{n}{3} \rceil + 1$.

***Proof*** It suffices to consider $n = 19 \times 2^k + 1$, $n = 24 \times 2^k + 1$ and $n = 30 \times 2^k + 1$. By direct calculation, $\lceil \log_2 \binom{n}{3} \rceil$ is at least $3k + 11$, $3k + 12$ and $3k + 13$ respectively. Hence, the proof follows by Proposition 2.6. □

Next, we consider $d = 4$. Instead of using three algorithms, we need four algorithms, that is, to estimate $M_L(4, 31 \times 2^k)$, $M_L(4, 27 \times 2^k)$, $M_L(4, 44 \times 2^k)$ and $M_L(4, 52 \times 2^k)$ first, and then find $M_L(4, n)$ in a sequel. In this case, we partition $N$ into four subsets $S_0$, $S_1$, $S_2$ and $S_3$ with the ratio of sizes $|S_0| : |S_1| : |S_2| : |S_3| = 1:4:4:22$ for the estimation of $M_L(4, 31 \times 2^k)$ and $1:5:5:26$, $1:6:6:31$, $1:7:7:37$ for the other three cases respectively.

First, by using the results obtained in the cases $d = 2$ and $d = 3$, we have the followings.

**Lemma 2.9** $M_L(4, 26) = 16$, $M_L(4, 31) = 17$, $M_L(4, 37) = 18$, $M_L(4, 44) = 19$ *and* $M_L(4, 52) = 20$.

***Proof*** We verify the first two result and the others can be obtained similarly.

For $n = 26$, test four items first $(T_1)$. If $L(T_1) = P$, then it takes two more tests to find one defective, and 13 more tests to find the others since $M_L(3, 25) \leq 13$. On the other hand, if $L(T_1) = N$, then take another four items for $T_2$. Again, if $L(T_2) = P$, we need two more tests to find one defective, and 12 more tests to find the others since $M_L(3, 21) \leq 12$. If $L(T_2) = N$, then take another four items for $T_3$. Again, if $L(T_3) = P$, we need two more tests to find one defective, and 11 tests to find the others since $M_L(3, 17) \leq 11$. Finally, if $L(T_3) = N$, then we can use 13 individual tests to find all four defectives. Hence, $M(4, 26) = 16$.

For $n = 31$, partition the set of 31 items into four subsets $S_0$, $S_1$, $S_2$ and $S_3$ such that $|S_0| = 1$, $|S_1| = |S_2| = 4$ and $|S_3| = 22$. First test $S_0 \cup S_1$ $(T_1)$. If $L(T_1) = N$, then go to the case $n = 22$, and thus $M(4, 31) = 17$. On the other hand, if $L(T_1) = P$, then take $T_2 = S_0 \cup S_2$. If $L(T_2) = N$, then there exist one defective in $S_1$. So, we can find this defective by two tests first and the others by 13 tests since $M_L(3, 25) \leq 13$. Otherwise, if $L(T_2) = P$, then take $T_3 = S_0$. Again, if $L(T_3) = P$, then the proof follows by using $M_L(3, 30) \leq 14$. Finally, if $L(T_3) = N$, then find one defective in $S_1$ and one defective in $S_2$ with four more tests, and 12 more tests for the other defectives since $M(2, 28) \leq 10$. □

By the same technique we use in the case $d = 3$, we have the result for $d = 4$ and 5.

**Proposition 2.10** $M_L(4, 31 \times 2^k) = 4k + 17$, $M_L(4, 37 \times 2^k) = 4k + 18$, $M_L(4, 44 \times 2^k) = 4k + 19$ *and* $M_L(4, 52 \times 2^k) = 4k + 20$.

**Proposition 2.11** $M_L(5, 46 \times 2^k) = 5k + 23$, $M_L(5, 53 \times 2^k) = 5k + 24$, $M_L(5, 61 \times 2^k) = 5k + 25$, $M_L(5, 70 \times 2^k) = 5k + 26$ *and* $M_L(5, 80 \times 2^k) = 5k + 27$.

***Proof*** For $d = 5$, $n = 46 \times 2^k, 53 \times 2^k, 61 \times 2^k, 70 \times 2^k$ and $80 \times 2^k$, we partition $n$ items into four subsets $S_0$, $S_1$, $S_2$ and $S_3$ with ratio $1 : 5 : 5 : 35$, $1 : 6 : 6 : 40$, $1 : 7 : 7 : 46$, $1 : 8 : 8 : 53$ and $1 : 9 : 9 : 61$, respectively. We take $T_1 = S_0 \cup S_1$ and $T_2 = S_0 \cup S_2$ if $L(T_1) = P$. Moreover, if $L(T_2) = P$, then take $T_3 = S_0$.

For $n = 46 \times 2^k$ and $L(T_3) = P$, we need $3 + k + M_L(4, 46 \times 2^k - 1) \leq 5k + 23$. If $L(T_3) = N$, we need $3 + (2k + 5) + M_L(3, 45 \times 2^k - 2) \leq 5k + 23$.

For $n = 53 \times 2^k$ and $L(T_3) = P$, we need $3 + k + M_L(4, 53 \times 2^k - 1) \leq 5k + 24$. If $L(T_3) = N$, we need $3 + 2 \times (k + 3) + M_L(3, (52 - 4) \times 2^k - 2) = 3 + 2 \times (k + 3) + M_L(3, 48 \times 2^k - 2) \leq 5k + 24$.

For $n = 61 \times 2^k$ and $L(T_3) = P$, we need $3 + k + M_L(4, 61 \times 2^k - 1) \leq 5k + 24$. If $L(T_3) = N$, we need $3 + 2 \times (k + 3) + M_L(3, (60 - 2) \times 2^k - 2) = 3 + 2 \times (k + 3) + M_L(3, 58 \times 2^k - 2) \leq 5k + 25$.

For $n = 70 \times 2^k$ and $L(T_3) = P$, we need $3 + k + M_L(4, 70 \times 2^k - 1) \leq 5k + 25$. If $L(T_3) = N$, we need $3 + 2 \times (k + 3) + M_L(3, 69 \times 2^k - 2) \leq 5k + 26$.

For $n = 80 \times 2^k$ and $L(T_3) = P$, we need $3 + k + M_L(4, 80 \times 2^k - 1) \leq 5k + 26$. If $L(T_3) = N$, we need $3 + (2k + 7) + M_L(3, (79 - 4) \times 2^k - 2) = 3 + (2k + 7) + M_L(3, 75 \times 2^k - 2) \leq 5k + 27$ or $3 + (2k + 6) + M_L(3, 79 \times 2^k - 2) \leq 5k + 27$. □

Now, by calculating $\lceil \log_2 \binom{n}{4} \rceil$ and $\lceil \log_2 \binom{n}{5} \rceil$, we have the following

**Corollary 2.12** $\lceil \log_2 \binom{n}{d} \rceil \leq M(d, n) \leq \lceil \log_2 \binom{n}{d} \rceil + d - 2 for 3 \leq d \leq 5$.

Finally, we note that if $d > 5$ is considered, then $d$ algorithms are needed to determine $M_L(d, n)$. Moreover, the sizes of four sets $S_0$, $S_1$, $S_2$ and $S_3$ are of sizes $2^k$, $(d + t) \cdot 2^k$, $(d + t) \cdot 2^k$, and $(\frac{(3d - 1)d}{2} + td + \frac{t(t - 1)}{2}) \cdot 2^k$ respectively provided $n = (1 + 2(d + t) + \frac{(3d - 1)d}{2} + td + \frac{t(t - 1)}{2}) \cdot 2^k$ where $t \in \{0, 1, 2, \cdots, d - 1\}$.

## 3 Conclusions

By using quaternary splitting on the testing set $N$, we are able to obtain a more accurate estimation of $M(d, n)$. Though not match the information lower bound $\lceil \log_2 \binom{n}{d} \rceil$, we believe that the results obtained in this paper are very close to the exact answer of $M(d, n)$. This is due to the fact that for larger $d$, binary informatic partition is in general not possible.

**Acknowledgements** The authors are grateful to the reviewers for carefully reading the manuscripts and making helpful comments. Also, we appreciate the suggestions of Kuo-Ching Hwang and Wu-Hsiung Lin in writing this article

## References

Bruno WJ, Knill E, Balding DJ, Bruce DC, Doggett NA, Sawhill WW, Stallings RL, Whittaker CC, Torney DC (1995) Efficient pooling designs for library screening. Genomics 26:21–30

Chang G, Hwang FK (1981) A group testing problem on two disjoint sets. SIAM J Algebraic Discrete Methods 2:35–38

Chang XM, Hwang FK, Weng JF (1989) Group testing with two and three defectives. Ann N Y Acad Sci 576:86–96

Cheng M, Miao Y (2011) On anti-collusion codes and detection algorithms for multimedia fingerprinting. IEEE Trans Inf Theory 57:4843–4851

Donoho DL (2006) Compressed sensing. IEEE Trans Inf Theory 52:1289–1306

Dorfman R (1943) The detection of defective members of large populations. Ann Math Stat 14:436–440

Du DZ, Hwang FK (2000) Combinatorial group testing and its applications, 2nd edn. World Scientific, Singapore

Farach M, Kannan S, Knill E, Muthukrichnan S (1997) Compression and complexity of sequences In: Proceedings, pp 357–367

Hong ES, Ladner RE (2002) Group testing for image compression. IEEE Trans Image Process 11:901–911

Hu MC, Hwang FK, Wang JK (1981) A boundary problem for group testing. SIAM J Algebraic Discrete Methods 2:81–87

Hwang FK (1972) A method for detecting all defective members in a population by group testing. J Am Stat Assoc 67:605–608

Thai MT (2012) Group testing theory in network security: an advanced solution. Springer, New York

Torney DC (1999) Sets pooling designs. Ann Comb 3:95–101

Weng JF, Hwang FK (1993) An optimal group testing algorithm on k disjoint sets. Op Res Lett 13:43–44

Wolf J (1985) Born again group testing: multiaccess communications. IEEE Trans Inf Theory 31:185–191