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

Let $F_n = \left\lceil \log_2 \binom{n}{2} \right\rceil$ (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 $M(d,n_1) \le M(d,n_2)$.

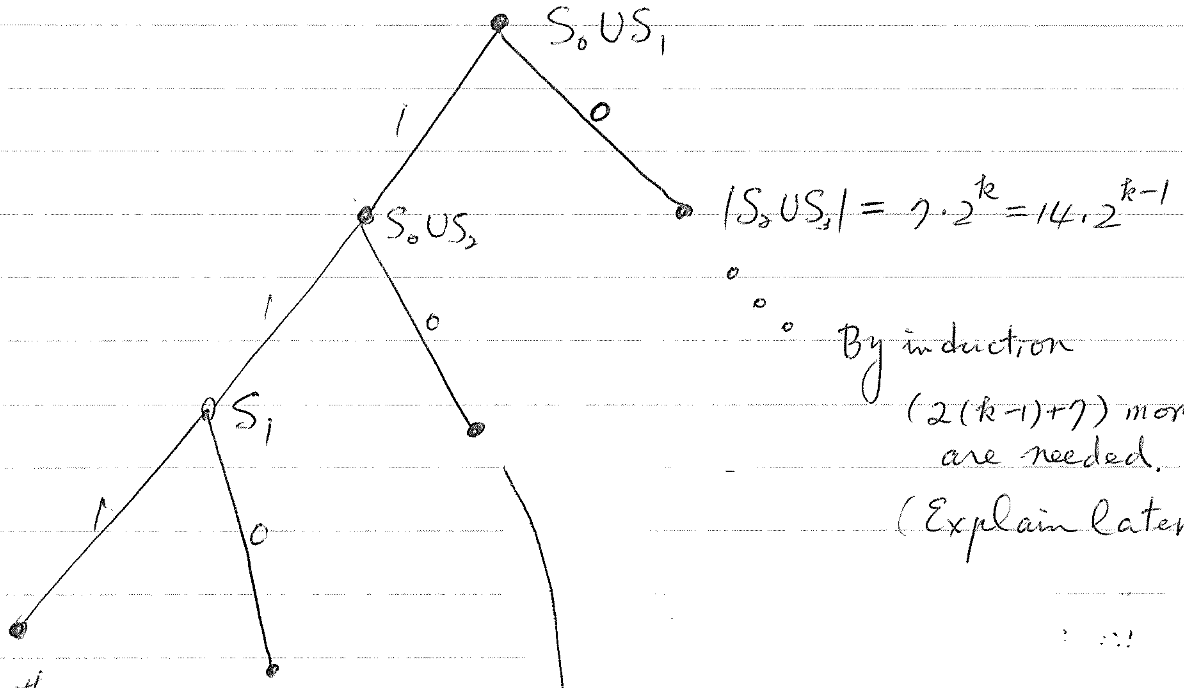
(*) For ①, $M(2,n) \le M(2, 10\cdot 2^k)$, and for ②, $M(2,n) \le M(2, 14 \cdot 2^k)$.

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



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.

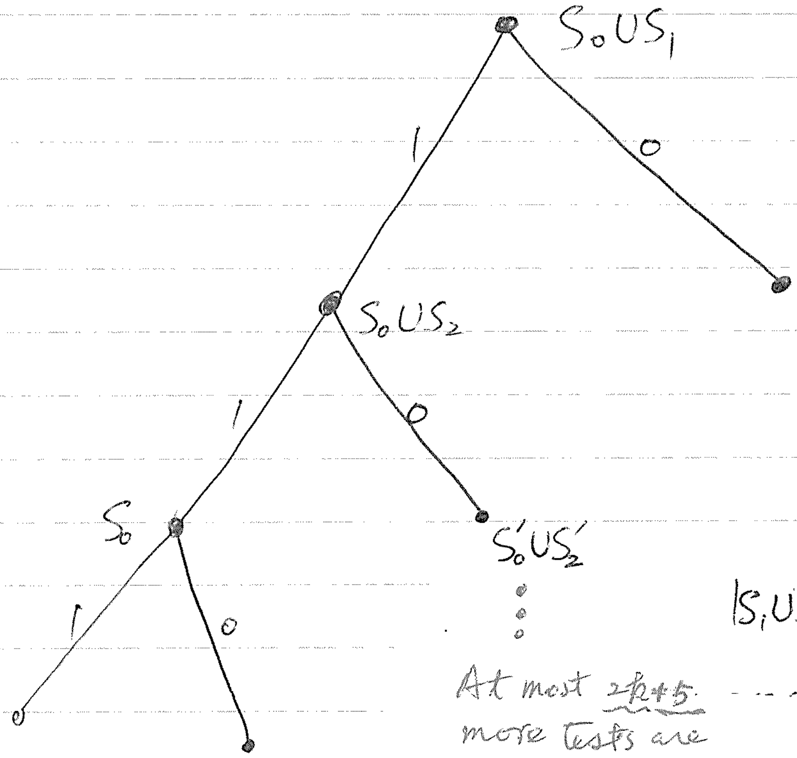
S_1 is positive, it takes $k+1$ tests to find the first positive and $\lceil \log_2 7 \cdot 2^k \rceil = k+3$ tests to find the second positive. ($2k+4$ in total.)

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.

$n = 14 \cdot 2^k$ Case

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



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

$|S_1 \cup S_3| = 10 \cdot 2^k$

$S_1 \cup S_3 \Rightarrow S_0' \cup S_1' \cup S_2' \cup S_3'$ 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 $2k+5$ 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

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

The worst case would take $2k+7$ tests to find the two positives.

$d=3$ Case

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

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

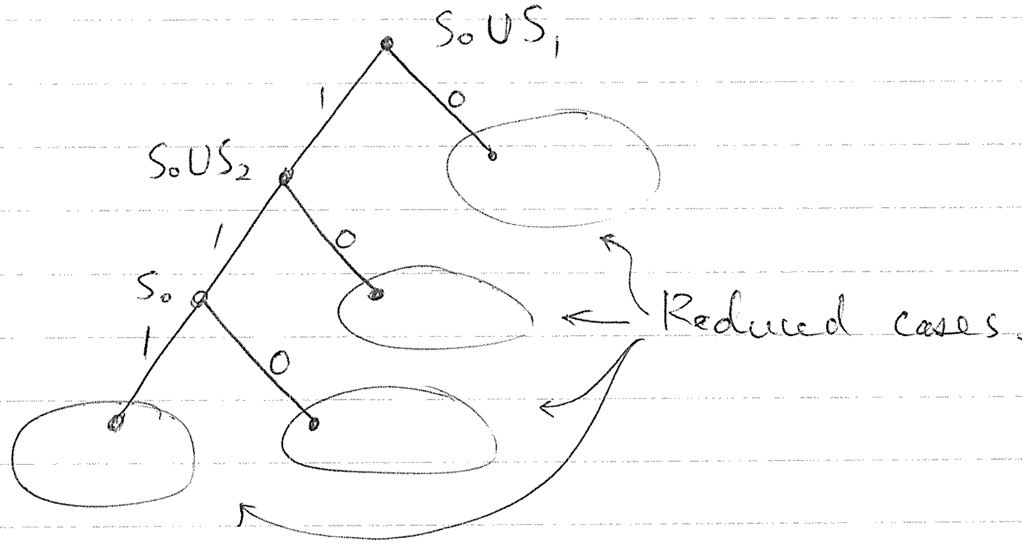
✓ Initial cases $\quad S_0 \quad S_1 \quad S_2 \quad S_3$.

$n \le 19\cdot 2^k$, Use $19\cdot 2^k = 2^k + 3\cdot 2^k + 3\cdot 2^k + 12\cdot 2^k$,

$n \le 24\cdot 2^k$, Use $24\cdot 2^k = 2^k + 4\cdot 2^k + 4\cdot 2^k + 15\cdot 2^k$,

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

The test-procedure is similar.



Construction of (d,r)-disjunct matrices

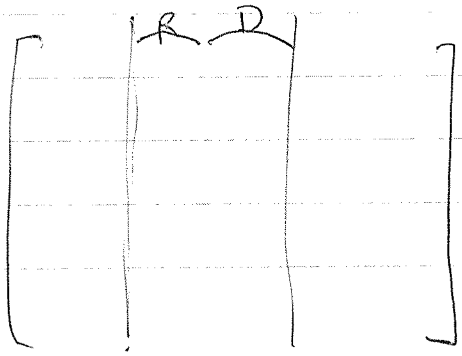
Definition (Dependency graph)

Let $A_1, A_2, \cdots, A_n$ be events in an arbitrary probability space. A graph $G=(V,E)$ where $V=\{1,2,\cdots,n\}$ is said to be a dependency graph for events $A_1, A_2, \cdots, A_n$ if for each i, $1 \le i \le n$, the event A_i is mutually independent of a set of all the other events except for those A_j with $\{i,j\} \in E$.

Theorem (LLL, Symmetric case)

Let $A_1, A_2, \cdots, A_m$ be events in an arbitrary probability space. Suppose that each event A_i is mutually independent of a set of all the other events A_j but at most μ of them, and that $P_r(A_i) \le p$ for all $1 \le i \le m$. If $e \cdot p \cdot (\mu+1) \le 1$, then $P_r\left(\bigcap_{i=1}^{m} \overline{A_i}\right) > 0$.

Construction for (d,r)-disjunct matrices



$t \times n$.

$|D| = d$ $\quad D \in \binom{N}{d}$

$|R| = r$ $\quad R \in \binom{N}{r}$

$R \cap D = \emptyset$

(d,r)-disjunct

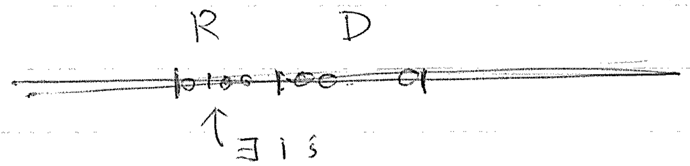
$\cup R \nsubseteq \cup D$

Let $M = (m_{i,j})$ be a $t \times n$ random binary matrix with $P(m_{i,j} = 1) = p$ and $P(m_{i,j} = 0) = 1 - p$. Notice that all $m_{i,j}$ are chosen independently.

Let $A_{D,R}$ be the event that $UR \subseteq UD$. Then

$$\Pr(A_{D,R}) = \left(1 - (1-p)^d (1 - (1-p)^r)\right)^t$$

每一列都不發生以下的情況



By LLL, it suffices to show that $e \cdot p \cdot (\mu + 1) \le 1$.

Here, $m = \binom{n}{d} \cdot \binom{n-d}{r}$ and $m' = \binom{n-(d+r)}{d}\binom{n-(d+r)-d}{r}$ is the number of independent events. By the way, the maximum degree of the dependency graph is $m - \cancel{1.}\ m'$

$\Pr(A_{D,R})$.

Hence, $(\mu + 1) = (m - m')$. So, we need $e\,p\,(m - m') \le 1$.

This inequality is equivalent to

$$t \ge \frac{1 + \ln(m - m')}{-\ln\left(1 - (1-p)^d (1 - (1-p)^r)\right)}.$$

Since $-\ln(1-x) \ge x$ for $0 \le x < 1$

$$t \ge \frac{1 + \ln(m - m')}{(1-p)^d (1 - (1-p)^r)}$$

$p = 1 - \left(\frac{d}{d+r}\right)^{\frac{1}{r}}$ gives the minimum after calculation.

(*) t 夠大即可得到 (d,r)-disjunct matrices;

因此取右边的最小值可以作為 t 的 upper bound!

(前值分母最大).

(**) Let $f(p) = (1-p)^{d}\cdot(1-(1-p)^{r})$

$$f'(p) = d(1-p)^{d-1}\cdot(-1) + (d+r)(1-p)^{d+r-1} = 0$$

$$\Rightarrow \quad (1-p)^{d-1}\left[(d+r)(1-p)^{r} - d\right] = 0$$

$$\Rightarrow \quad (1-p)^{r} = \frac{d}{d+r}, \quad 1-p = \left(\frac{d}{d+r}\right)^{\frac{1}{r}}$$

$$\Rightarrow \quad p = 1 - \left(\frac{d}{d+r}\right)^{\frac{1}{r}}.$$

$$f''(p) = d(d-1)(1-p)^{d-2} - (d+r)(d+r-1)(1-p)^{d+r-2}$$

$$= (1-p)^{d-2}\left[d(d-1) - (d+r)(d+r-1)\underline{(1-p)^{r}}\right]$$

$$\geq 0$$

代入之後,

$$(\star)\ t(n,d,r) \leq \left(1+\frac{d}{r}\right)\cdot\left(1+\frac{r}{d}\right)^{\frac{d}{r}}\cdot\left[1+\ln(m-m')\right] \text{ where}$$

$$\Uparrow$$

n 行所需要的最少列數

$$m = \binom{n}{d}\binom{n-d}{r}, \quad m' = \binom{n-d-r}{d}\binom{n-2d-r}{r}.$$

Thanks for your patience in this semester!

Happy New Year of "Dogs"!

2018.1.5