Lec11/12 $\boxed{\text{Question 5}}$

For any d-regular graph G, $\Phi(G) \leq \frac{1}{2} + o(1)$

<u>Proof:</u> Recall

$$\phi(S) = \frac{e(S, S^c)}{|S| \cdot d} \qquad S \neq \emptyset,\ S \neq V$$

and $$\phi(G) = \min_{1 \leq |S| \leq n/2} \frac{e(S, S^c)}{|S| \cdot d}$$

G connected $\Leftrightarrow \phi(G) \neq 0 \Leftrightarrow \lambda_2(G) \neq 0$

<u>easier</u> inequality: $\phi(G) \leq 1$, since

$$e(S, S^c) \leq |S| \cdot d$$

Back to proof:

We need to show that for any graph $G = (V, E)$, there is $S \subseteq V$ with $1 \leq |S| \leq n/2$ and

$$\phi(S) \leq \tfrac{1}{2} + o(1)$$

<u>Hint:</u> Use probabilistic method

The randomised MAX-CUT Algorithm achieves

$$E[e(S, S^c)] = |E|/2,$$ where

$S := \{v \in V : X_v = 1\}$, X_v's independent Bernoulli

Question 5 (continuation)

What about the size of S?

$$E[|S|] = E[\sum_{v \in V} X_v] = \sum_{v \in V} E[X_v]$$
$$= n \cdot 1/2$$

We want $|S|$ to be close to $n \cdot 1/2$

and $e(S, S^c)$ to be at most $|E|/2$

$\Rightarrow$ use concentration & union bound

Step 1: Size of S

Chernoff-Bound ("nice" form, slide 16)

$$P[|X - E[X]| \geq t] \leq 2 \cdot e^{-2t^2/n}$$

$$\Rightarrow P[\,|\,|S| - n/2\,| \geq \sqrt{n \log n}\,] \leq 2 \cdot e^{-2 \log n}$$
$$= 2n^{-2}.$$

Step 2: Cut edges between S and S^c

Is this a sum of independent r.v.'s?

No! (unfortunately...)

But: $e(S, S^c) = f(X_{v_1}, X_{v_2}, \ldots, X_{v_n})$,

where $X_{v_1}, X_{v_2}, \ldots, X_{v_n}$ are mutually independent

$\Rightarrow$ Method of Bounded Differences

$\boxed{\text{Question 5}}$ (continuation)

If X_{v_i} changes, how much does $e(S,S^c)$ change?

$\Rightarrow$ by at most $\deg(v_i) = d =: c_i$

MOBD (McDiarmid's Inequality)

$$\hookrightarrow P[e(S,S^c) \geq |E|/2 + t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n} c_i^2}\right)$$

we choose $t = \sqrt{n \cdot \log n} \cdot d$

$$\Rightarrow P[e(S,S^c) \geq |E|/2 + \sqrt{n \log n} \cdot d] \leq n^{-2}.$$

Union Bound gives:

$$P\left[\{||S| - n/2| \leq \sqrt{n \log n}\} \cap \{e(S,S^c) \leq |E|/2 + \sqrt{n \log n} \cdot d\}\right]$$

$$\geq 1 - 2n^{-2} - n^{-2} \geq 1 - n^{-1} \underline{\underline{> 0}}$$

For any graph G, there exists $S \subseteq V$ with:

i) $||S| - n/2| \leq \sqrt{n \log n}$, and

ii) $e(S,S^c) \leq |E|/2 + \sqrt{n \log n} \cdot d$

By possibly switching $|S|$ with $|S^c|$, we get

i) $\Rightarrow |S| \in [n/2 - \sqrt{n \log n}, n/2]$

$$\Rightarrow \phi(G) \leq \phi(S) \leq \frac{|E|/2 + \sqrt{n \log n} \cdot d}{(n/2 - \sqrt{n \log n}) \cdot d} = \frac{nd/4 + \sqrt{n \log n} \cdot d}{(n/2 - \sqrt{n \log n}) \cdot d}$$

$$\leq \frac{1}{2} + o(1) \qquad \blacksquare$$

Lec 13 $\boxed{\text{Question 1}}$

Let's consider case where <u>exactly one</u> packet is missing

$x_1, x_2, \dots, x_{n-1}$ be $n-1$ different IDs from $\{1, \dots, n\}$

<u>Naive Solution:</u> Create array of size n
$\Rightarrow$ space complexity $\Omega(n)$

<u>Idea:</u> use that $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

Algorithm:
$S_1 \leftarrow 0$
For each packet with label x
$S_1 \leftarrow S_1 + x$
Return $\frac{n(n+1)}{2} - S_1$

$\rightarrow$ does the job and uses only $O(\log n)$ bits of space (if all packets are sent, we return 0)

<u>Extension:</u> What if <u>two</u> packets are missing?

Let $S_1 = \sum_{i=1}^{n-2} x_i \Rightarrow x_{n-1} + x_{n-2} = \frac{n(n+1)}{2} - S_1$

and $S_2 = \sum_{i=1}^{n-2} x_i^2 \qquad x_{n-1}^2 + x_{n-2}^2 = \frac{(2n+1)n(n+1)}{6} - S_2$

$\rightarrow$ solve quadratic formula

Lec 13 $\boxed{\text{Question 1}}$ (continuation)

As suggested by one student (thank you!), we can also take the XOR:

$$x_1 \oplus x_2 \oplus \dots \oplus x_{n-1} := S$$

and then return

$$1 \oplus 2 \oplus \dots \oplus n \oplus S$$

which is equal to x_n (missing element).

- this uses only $\log_2 n + 1$ space
- to cover the case where all n packets are sent, we should encode packets starting from 1 (and <u>not</u> 0!)

Lec 14 $\boxed{\text{Question 1}}$

Note: If we run a <u>deterministic</u> algorithm, the adversary specifying the data set can simulate our algorithm, in particular, can make all predictions <u>wrong</u>!

(Compare to pivot selection in Quick-Sort! Choosing pivot randomly is only so effective, since we assume the input is specified without knowing the random decisions / random bits)

$n=2$, T arbitrary

t	Exp. 1	Exp. 2	Algo Prediction	Actual Result
1	0	1	1	0
2	1	0	1	0
3	1	0	0	1
4	0	1	1	0
5	1	0	0	1
⋮	⋮	⋮	⋮	⋮
T				

$$M^{(T)} = T, \quad m_1^{(T)} + m_2^{(T)} = T$$

$$\Rightarrow \min\left(m_1^{(T)}, m_2^{(T)}\right) \leq T/2$$

Lec 14 $\boxed{\text{Question 3}}$

For each expert i, $i=1,2,\dots,n$ create another "opposing" expert $i+n$ which <u>always</u> predicts the opposite of expert i.

Performance-Guarantee of RWMA:

old: $M^{(T)} \leq 1 \cdot (1+\delta) \cdot \min_{i \in [n]} m_i^{(T)} + \frac{\ln n}{\delta}$

$$\Downarrow$$

new: $M^{(T)} \leq 1 \cdot (1+\delta) \cdot \min_{i \in [n]} \left(m_i^{(T)}, T - m_i^{(T)} \right) + \frac{\ln (2n)}{\delta}$

<u>General Approach:</u> can build large set of additional experts which are logical functions or even algorithms based on the original experts

$$\Downarrow$$

- "Baseline" of mistakes by best expert improves
- Convergence becomes a bit slower due to $\ln(.)$ term

Lec 14 | Question 4

Naive Greedy: Always follow prediction of the <u>best</u> expert so far (break ties in favour of smallest ID)

Example for $n=3$:

t	$m_1^{(t-1)}$	$m_2^{(t-1)}$	$m_3^{(t-1)}$	Predictions E1	E2	E3	Greedy	Result
1	0	0	0	1	0	0	1	0
2	1	0	0	0	1	0	1	0
3	1	1	0	0	0	1	1	0
4	1	1	1	1	0	0	1	0
5	2	1	1	0	1	0	1	0
6	2	2	1	0	0	1	1	
7	2	2	2	...				

⋮

For $T = n \cdot K = 3 \cdot k$, $K \in \mathbb{N}$:

$$m_i^{(T)} = \frac{1}{3} \cdot T = \underline{\frac{T}{n}}, \quad \text{but } M^{(T)} = \underline{T}$$

Large gap!

A better approach is called "Follow-the-Perturbed-Leader", which adds some random noise to each $m_i^{(T)}$ ⤳ similar to UCB-Algo.

Lec 14 $\boxed{\text{Question 4}}$ (continuation)

We used "worst-case" tie breaking before!
What if we randomly choose one of the leaders in case of a tie?

In round 1, we have $\frac{1}{n}$ prob. of making error
" 2, " $\frac{1}{n-1}$ prob. of making error
⋮
" n, we have 1 prob. of making error

$\Rightarrow$ Every n consecutive rounds, we make in expectation

$$\sum_{i=1}^{n} \frac{1}{n-(i-1)} \approx \ln(n)$$

many mistakes.

$\Rightarrow$ $E[M^{(T)}] = T \cdot \frac{\ln(n)}{n}$, where $n \mid T$.

as before: $m_i^{(T)} = \frac{T}{n}$ $\Rightarrow$ better performance,
but there is still a large gap between $M^{(T)}$ and $\min_{i \in [n]} m_i^{(T)}$.