LCA for Maximal Independent Set (cont.)

Recall from last time:

Maximal Independent Set:

$X =$ description of graph
$Y = (y_1 \dots y_n)$ s.t. $y_i = \begin{cases} 1 & \text{if node } i \text{ in MIS} \\ 0 & \text{o.w} \end{cases}$

<u>def</u> $U \subseteq V$ is a "maximal independent set" (MIS) if

(1) $\forall\, u_1, u_2 \in U \quad (u_1, u_2) \notin E$ "independent"

(2) $\nexists\, w \in V \setminus U$ s.t. $U \cup \{w\}$ is independent "maximal"

G has max degree Δ

Recall from last time:

def $\mathcal{A}$ is "Local computation algorithm" (LCA) for a problem Π if

- given:
  - probe access to input X
  - random bits r
  - local memory
- answers queries to bits/words of

$$y = \cancel{\Pi(x)}$$
$$y \in \Pi(x)$$

what if more than one solution?, must be consistent with one of them

maybe $y = \Pi_r(x)$? determined by random bits

alternatively many copies, each gets one query

- memory of LCA "wiped" clean between queries (keep the random bits)

- $\forall$ queries i,

  let $y_i \leftarrow$ output of LCA on i

  note: can't depend on other queries since "wiped"

  $\forall x$, whp over r, $\forall$ queries y

  then y_i's must be consistent with a legal solution for input X. (e.g. $\in \Pi(x)$)

Recall from last time

Distributed Algorithm for MIS:

"Luby's Algorithm" (actually a variant...)

- $MIS \leftarrow \phi$
- all nodes set to "live"
- repeat k times:
  - $\forall$ nodes v, (in parallel) color self "red" with prob $\frac{1}{2\Delta}$ else "blue"
    send color to all nbrs
  - if v colors self red, & no other nbr of v colors self red, then
    - add v to MIS
    - remove v & all nbrs from graph (set to "inactive")

(For purposes of analysis, continue to select colors after removed but don't send to nbrs)

Thm $\Pr[\#\text{ phases til graph empty} \geq 8\Delta \log n] \leq \frac{1}{n}$

Corr $E[\#\text{ phases}]$ is $O(\Delta \log n)$ ← can improve!

Recall from last time:

Main Lemma: For live v, $\Pr[v \text{ added to MIS in a round}] \geq \frac{1}{4\Delta}$

Lemma $\Rightarrow$ $\Pr[v \text{ live after } \underbrace{4 \cdot k' \cdot \Delta}_{= k} \text{ rounds}] \leq \left(1 - \frac{1}{4\Delta}\right)^{4\Delta \cdot k'}$

$\leq e^{-k'}$

LCA will sequentially simulate v's view of computation

in $\Delta^{O(\# \text{rounds})} = \Delta^{O(k)}$ time

Recall: Problem: to decide all nodes, need $k' = O(\log n)$ $(k = O(\Delta \log n))$

to sequentially simulate $\Theta(\Delta \log n)$ rounds,

need $\Delta^{\Theta(\Delta \log n)}$ complexity .... not sublinear in n ☹

$= n^{\Theta(\Delta \log \Delta)}$

- define

"Luby-status":

sequentially simulate Luby for $K = \Theta(\Delta \log \Delta)$ rounds

at end, each $v \in V$ is one of

$\begin{cases} \text{live} \\ \text{in MIS} \\ \underline{\text{not}} \text{ in MIS} \end{cases}$

in MIS ← set self to red & no nbrs did

not in MIS ← taken out by nbr in MIS

- if v is in/not in then done

  else v is alive ← what do we do now?

Questions:

- what is probability that v still alive?

$$\Pr[v \text{ inactive}] \le \Pr[\text{survives } \Theta(\Delta \log \Delta) \text{ rounds}]$$

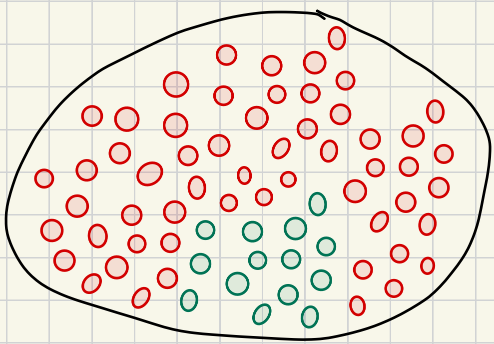
$$\le e^{-\Theta(\Delta \log \Delta)} \le \frac{1}{\Delta^c}$$

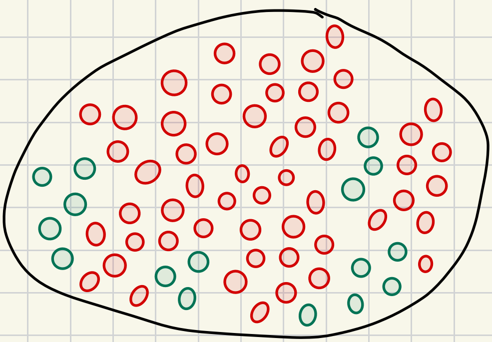
for any c & for proper choice of constant in # rounds

Very few stay active!

- how much work to "finish off" v?

Key insight distribution of live nodes!





big clumps of surviving nodes?

NO!

surviving nodes in small connected components

relies on degree bound of graph

$\Longrightarrow$ survival of components $\approx$ independent

LCA for MIS(v):

runtime

- run sequential version of Luby status(v) — $\Delta^{O(\Delta \log \Delta)}$
- if it is "in"/"out", output answer + halt
- else, (1) do BFS to find v's connected component of live nodes — $\Delta^{O(\Delta \log \Delta)} \times$ size of component

  (2) Compute lexicographically 1st MIS M′ for that connected component — size of component

  (note that nbrs outside connected component are all "out" of MIS)

  (3) Output whether v "in"/"out" of M′

What is size of component?

Bounding size of connected components:

Claim After $O(\Delta \log \Delta)$ rounds, connected components of survivors of size $\leq O(\text{polylog}(\Delta) \cdot \log n)$

$\Longrightarrow$ can find whole component via BFS "brute force"

total runtime is $O(\Delta^{O(\Delta \log \Delta)} \cdot \log n)$

Main difficulty:

- survival of v & neighbors not independent in a round
- not independent of previous rounds

will show: (1) any big set has lots of independent events

(2)

Bounding survivors: independence between rounds

Some interesting variables...

$$A_v = \begin{cases} 1 & \text{if } v \text{ survives all rounds} \\ 0 & \text{o.w.} \end{cases}$$

$$B_v = \begin{cases} 1 & \text{if } \nexists \text{ round s.t. } v \text{ colors self \& no } w \in N(v) \text{ colors self} \\ 0 & \text{o.w.} \end{cases}$$

$B_v = 0$ might not mean that $v \in MIS$

e.g. if v inactive due to nbr being put in MIS in earlier round

Note: $A_v = 1 \Rightarrow B_v = 1$

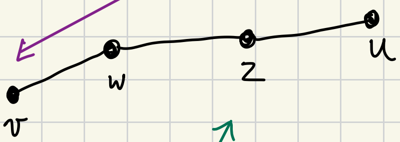
e.g. v survives $\Rightarrow \nexists$ round s.t. v colors self & no $w \in N(v)$ colors self

these events are independent!

$$\Pr[B_v = 1] \leq \Big(\underbrace{1 - \tfrac{1}{4\Delta}}_{\text{prob survive one round}}\Big)^{c \cdot \Delta \log \Delta} \leq \frac{1}{8\Delta^3}$$

for $c \geq 20$

We care about the A_v's, but the B_v's have nice independence properties



distance 3:
B_v depends on B_w
B_u depends on B_z
but B_u & B_v are independent!

distance 2:
B_v & B_z depend on w's coins,
so <u>not</u> independent

$\deg \le \Delta \Rightarrow$ each B_u depends on $\le \Delta^2$ other B_w's

Plan to bound size of connected component:

- any large conn component has lots of nodes that are $\underbrace{\text{dist} \geq 3}_{\text{so independent!}}$

size $\geq w$

- these independent nodes are unlikely to simultaneously survive

BUT there are lots of large conn comps!! do we need to union bound over all sets of size w?

NO

Define new graph: the hope... nodes in new graph are independent?

Let $G^{(3)} \leftarrow$ graph with

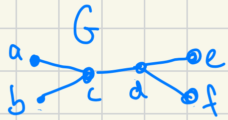
nodes $\sim B_v \quad \forall v \in G$

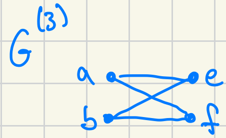
edges $\sim B_v + B_w$ distance $= 3$ in G

$\deg(G^{(3)}) \leq \Delta^3$

hope: represent independent events!! (but not there yet)

What is $G^{(3)}$?





$G^{(3)}$ might not even be connected?

or worse...

nodes in $G^{(3)}$ could be nbrs in G + thus not independent?

First,

Let's simplify conn. comp. to trees:

Observe: # conn comp in $G^{(3)}$ of size w

$$\leq$$

size w subtrees in $G^{(3)}$

why? map each component C to arbitrary spanning tree of C

mapping is 1-1 (but could have many spanning trees per component)

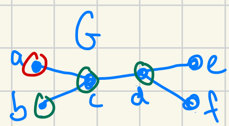
great! we are good at counting trees!

Claim For (live) connected component S in G, $G^{(3)}$ contains tree with vertex set T as subgraph +

(1) $|T| \geq \dfrac{|S|}{\Delta^2+1}$ big!

(2) $\text{dist}_G(u,v) \geq 3 \quad \forall\, u,v \in T$ } so independent!!

Proof Pick T greedily



1. pick arbitrary $v \in S$
2. repeat until S empty
3. move v from S to T
   remove all u with $\text{dist}_G(u,v) < 3$ from S
4. pick new node $v \in S$ s.t. $\text{dist}_G(u,v) = 3$ for some $u \in T$

so $(u,v) \in G^{(3)}$

Note: for each v put in T, remove $\leq \Delta^2$ nodes from S

$+$ remove v

total $\leq \Delta^2 + 1$

How big are the remaining components?

Let $s \leftarrow \log \frac{n}{3}$

Let $\Upsilon_s \leftarrow \{ T \subseteq V \mid |T| = s,$ all $u,v \in T$ have dist ≥ 3
in G & T connected in $G^{(3)} \}$

$\Pr[\exists T \in \Upsilon_s$ s.t. all nodes in T survive$]$

$\leq \sum_{T \in \Upsilon_s} \Pr[$ all nodes in T survive$]$ union bnd

$\leq |\Upsilon_s| \cdot \left(\frac{1}{8\Delta^3}\right)^s$

$\leq \left(N \cdot (4\Delta^3)^s\right) \cdot \left(\frac{1}{8\Delta^3}\right)^s \leq \frac{N}{2^s} = \frac{1}{3}$ see next page for bound on # trees

(so, unlikely any size s tree survives)

$\Rightarrow$ with prob $\geq 2/3$ all surviving conn. comp.
have size $\leq (\Delta^2+1) \cdot \log \frac{n}{3}$
$= O(\Delta^2 \log n)$

How many subtrees in a degree bounded graph?

Known Thm # non isomorphic trees on w nodes $\le 4^w$

ignores names of nodes & root (just shape)

Corr # size w subtrees in N-node graph of degree $\le D$

is $\le N \cdot 4^w \cdot D^w = N(4D)^w$

considers names of nodes & root

why?

	# choices
• choose root in H	N
• choose size w tree (shape) from "known thm"	4^w
• choose placement in H via DFS sequence	$\le D$ choices for 1st child $\le$ " " " 2nd step ⋮

total # choices: $N \cdot 4^w \cdot D^w$

$= N(4D)^w$