Lecture 19

Local Computation Algorithms:

Maximal Independent Set

Maximal Independent Set:

not maximum

not NP-complete

<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) $\exists \ w \in V \setminus U$ s.t. $U \cup \{w\}$ is independent "maximal"

Today's assumption:

G has max degree d

Note: MIS can be solved via greedy (not NP-Complete)

Distributed Algorithm for MIS:
"Luby's Algorithm" (actually a variant)

Maximal Independent Set:

<u>def</u> $U \subseteq V$ is a "Maximal Independent Set" (MIS) if

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

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

- MIS $\leftarrow \varphi$
- all nodes set to "live"
- repeat K times in parallel:
  - $\forall$ nodes v, color self "red" with prob $= \frac{1}{2d}$, 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 "dead")

red ≈ "I volunteer to be in MIS"

(for purposes of analyses, continue to select selves after die, but don't do anything else)

<u>Thm</u> $\Pr[\#\text{ rounds til graph empty} \geq 8d \log n] \leq \frac{1}{n}$

<u>Corr</u> $E[\#\text{ rounds}]$ is $O(d \log n)$ $\Longleftarrow$ can improve!

- MIS $\leftarrow \varphi$
- all nodes set to "live"
- repeat K times in parallel:
  - $\forall$ nodes v, color self "red" with prob $\geq \frac{1}{2d}$, 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 "dead")

Main Lemma

$$\Pr[v \text{ live + added to MIS in round}] \geq \frac{1}{4d}$$

} then v "dies"

Proof

$$\Pr[v \text{ colors self red}] = \frac{1}{2d}$$

$$\Pr[\text{any } w \in N(v) \text{ colors self red}] \leq \sum_{w \in N(v)} \frac{1}{2d}$$ (union bnd)

$$\leq 1/2$$ (bound on degree)

$$\therefore \Pr[v \text{ colors self red + no other nbr colors self red}] \geq \frac{1}{2d}\left(1 - \frac{1}{2}\right) = \frac{1}{4d} \quad \blacksquare$$

$\Rightarrow$ **Corr** $\Pr[v \text{ live after } \underline{4K'd} \text{ rounds}] \leq \left(1 - \frac{1}{4d}\right)^{4K'd} = e^{-K'}$

$$\boxed{K' = \log n}$$

Setting K:

if $K = O(d \log n)$, $\Pr[v \text{ live at end}] \leq e^{-O(\log n)} = \frac{1}{n^c}$
of $O(d \log n)$ rounds

(can do better)

See slides for
Local Computation Algorithm (LCA)
model

Problem when sequentially simulate k-round algorithm
get d^k complexity

d ← degree, k ← #rounds

$k = o(\log n) \Rightarrow d^k$ not sublinear

What to do? run fewer rounds
many nodes will not be decided yet ← is it ok?

Local Computation Algorithm to compute Luby's answer:

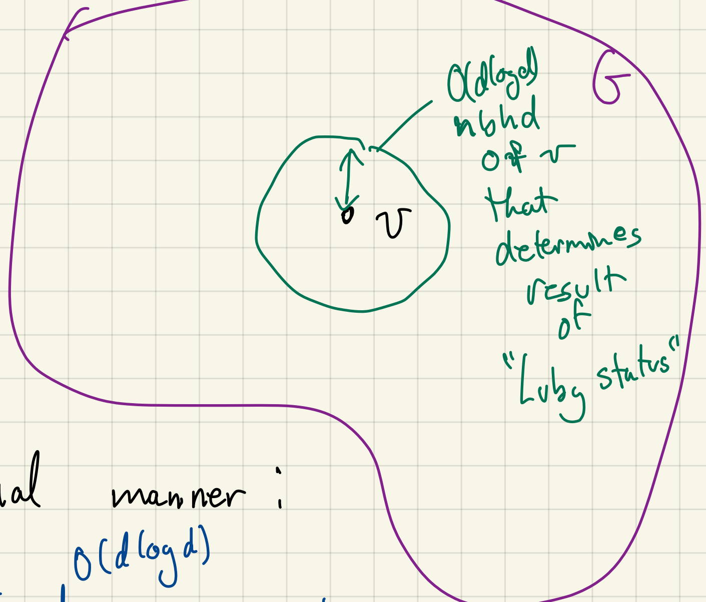
"Luby status"

- Run $\boxed{\text{Luby with } K = O(d \log d) \text{ rounds}}$

at end, each node v is one of:

- live
- in MIS ← set self to red + no nbrs red
- not in MIS ← taken out by nbr who is in MIS

```
Luby:   set K = O(d log d)        "Luby Status"
  · MIS ← φ
  · all nodes set to "live"
  · repeat K times in parallel:
      · ∀ nodes v, color self "red" with prob ≥ $\frac{1}{2d}$, 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 "dead")
```



- Use "Parnas-Ron" reduction:

simulate v's view of computation in sequential manner:

+ determine whether v is live/in/not in

$d^K = d^{O(d \log d)}$ queries

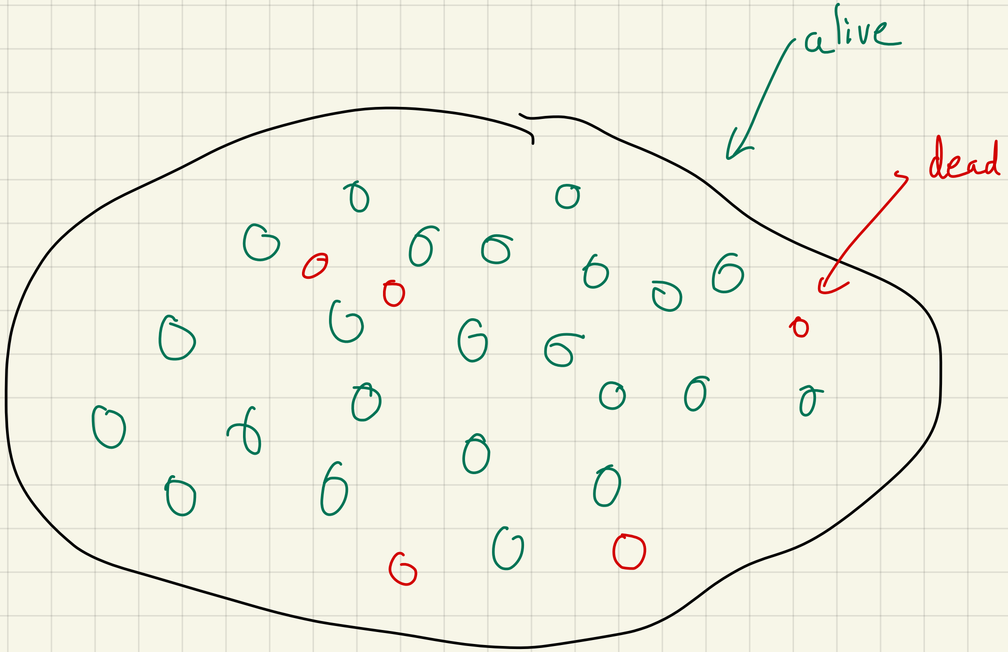
- if v is in/not in then done

else v is alive ← what do we do now?

Questions:

What is prob v is alive?

How are live nodes distributed after $O(d \log d)$ rounds?

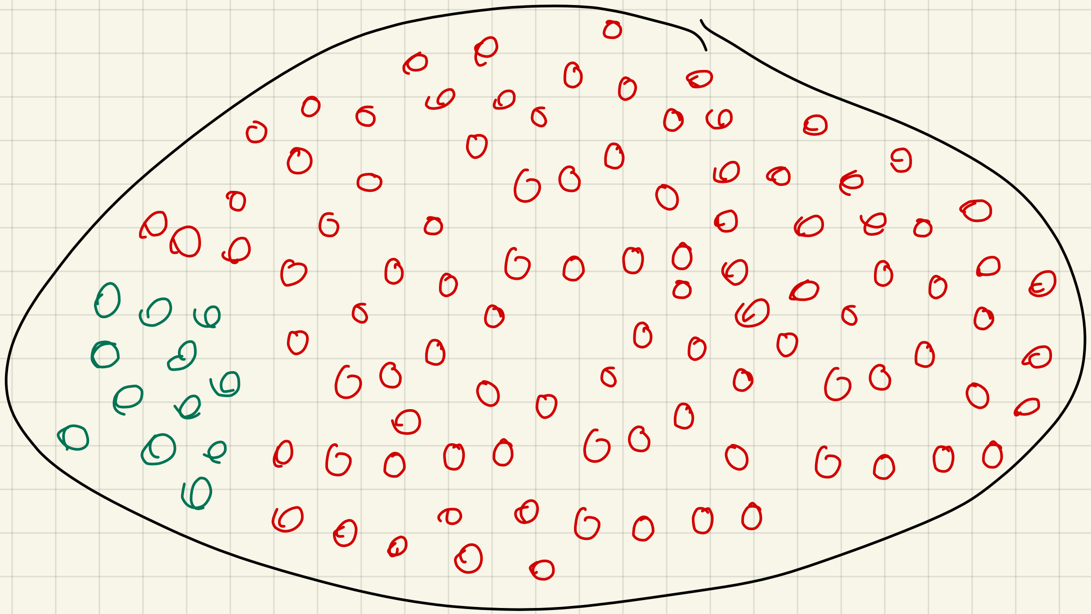


Most "die"

lots of live, few dead?

NO

$$\Pr[v \text{ survives } O(d \log d) \text{ rounds}] \le e^{-O(\log d)} \le \frac{1}{d^c}$$



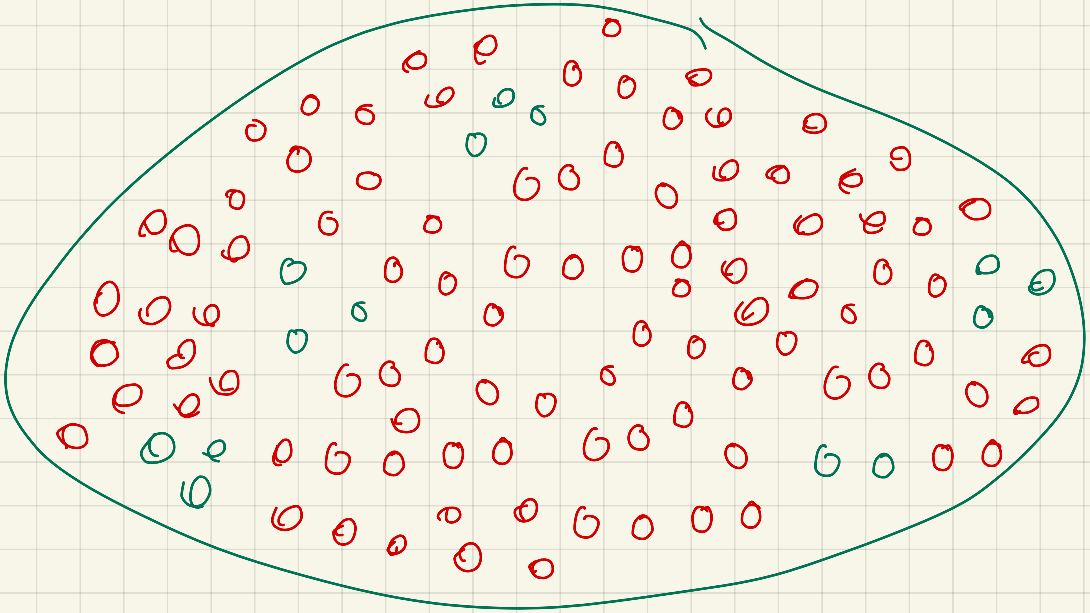
few live but clumped together?

NO!

surviving nodes will be in small connected components

Surviving nodes will be in small connected components

"Shattered".



This relies heavily on degree bound of graph

- \# conn comp subgraphs small
- survival of components ≈ independent

"Luby status" Luby with $k = O(d \log d)$:

given v, is it:

- live
- in MIS ← set self to red + no nbrs did
- ~~not~~ in MIS ← taken out by nbr

Luby:

```
• MIS ← $\varphi$
• all nodes set to "live"
• repeat K times in parallel:
    • $\forall$ nodes $v$, color self "red" with prob $\geq \frac{1}{2d}$, 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 "dead")
```

LCA for MIS(v):

- run sequential version of Luby status (v)
- if it is in/out output answer + halt
- else, (1) do BFS to find v's connected component of live nodes

  (2) Compute lexicographically 1st MIS M' for that connected component (consistent with nbrs that are decided)

  (3) Output whether v in/out of M'

Runtime

$d^{O(d \log d)}$

$$d^{O(d \log d)} \times \underbrace{\text{size of component}}_{\text{what is this?}}$$

size of of component

Bounding size of connected components:

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

$\Rightarrow$ runtime of above procedure is $\sim d^{O(d \log d)} \times \text{polylog}(d) \cdot \log n$

Main difficulty: survival of v + neighbors not independent

Bounding survivors:

Luby:

```
• MIS ← φ
• all nodes set to "live"
• repeat K times in parallel:
    • ∀ nodes v, color self "red" with prob ≥ $\frac{1}{2d}$, 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 "dead")
```

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

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

might not mean v alive since at some pt some nbr w of v could have gone into MIS & removed v

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

$$\Pr[B_v = 1] \leq \left(\underbrace{1-\frac{1}{4d}}_{\text{prob survive one round}}\right)^{cd\log d} \leq \frac{1}{8d^3} \quad \text{for } c \geq 20$$

Bounding size of connected components:

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

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

might not mean that $v \in MIS$

e.g. if v died due to nbr being put in MIS

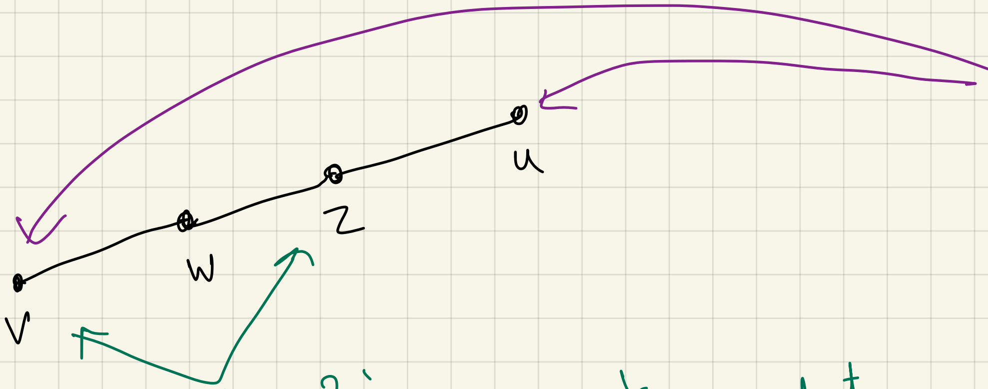
Luby:

- $MIS \leftarrow \varphi$
- all nodes set to "live"
- repeat K times in parallel:
  - $\forall$ nodes v, color self "red" with prob $\geq \frac{1}{2d}$, 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 "dead")

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

eg. v survives $\Rightarrow \nexists$ round s.t. v colors self + no $w \in N(v)$ colors self

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



distance 2: B_v + B_z depend on w's coins so not independent

distance 3: B_v depends on B_w

B_u depends on B_z

but B_u + B_v are indep

$\deg \leq d \Rightarrow$ each B_u depends on $\leq d^2$ other B_w's

Bounding size of connected components:

Claim After $O(d \log d)$ rounds, connected components of survivors of size $\leq O(\mathrm{poly}(d) \cdot \log n)$

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

Proof idea:

- any large conn component has lots of nodes that are independent (distance ≥ 3)
- these indep nodes unlikely to simultaneously survive

do we need union bnd over all sets of size w?

NO

Claim After $O(d \log d)$ rounds, connected components of survivors of size $\le O(\text{poly}(d) \cdot \log n)$

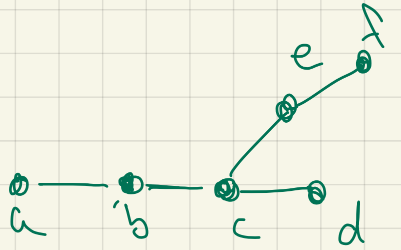
Proof

Let $H \leftarrow$ graph s.t. nodes $\sim B_v$

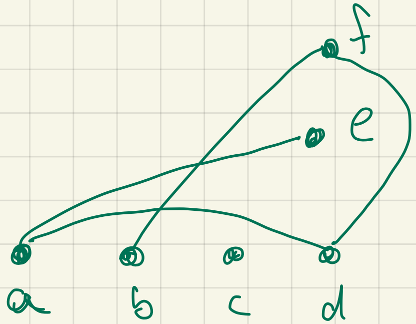
edges $\sim B_v$ & B_w distance 3 in G

represent indep events

$\deg(H) \le d^3$



G



H

not even connected!

great! we are good at counting subtrees

Observe: # components in H of size w $\le$ # size w subtrees in H

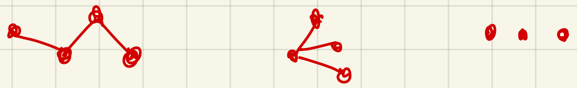
why? map each component C in H to arbitrary spanning tree of C

mapping is 1-1

but could have many spanning trees per component

How many subtrees in a degree bounded graph?

Known Thm # non isomorphic trees on w nodes $\leq 4^w$ ← ignores "names" of nodes + root (just shape)



Corr # size w subtrees in N-node graph of degree $\leq D$ ← consider names of nodes + root

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

why?

	# Choices
• choose root in H	N
• choose size w tree shape from known thm	4^w
• choose placement in H	D choices for 1st child " " " 2nd ⋮

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

Claim After $O(d \log d)$ rounds, connected components of survivors of size $\le O(\text{poly}(d) \cdot \log n)$

Proof

- Let $H \leftarrow$ graph s.t. nodes $\sim B_v$

  edges $\sim B_v$ & B_w distance 3 in G

  represent independent events!!

- $\deg(H) \le d^3$

- # components in H of size w $\le$ # size w subtrees in H $\le n \cdot (4d^3)^w$

- $\Pr[\text{node } u \text{ survives}] \le \frac{1}{8d^3}$

$\Pr[\text{component of size } w \text{ in } H \text{ survives}] \le \left(\frac{1}{8d^3}\right)^w$ since indep

$\Pr[\underline{\text{any}} \text{ component of size } w \text{ survives in } H] \le n(4d^3)^w \cdot \left(\frac{1}{8d^3}\right)^w = \frac{n}{2^w}$

$\Rightarrow$ for $w = \Omega(\log n)$,

$\Pr[\exists \text{ surviving component of size } \underline{w \text{ in } H}] \le \frac{1}{n}$

what about G??

Component of size $\le w$ in H $\Rightarrow$ Component of size $\le w \cdot d^2$ in G

So unlikely to have $\underline{\text{any}}$ surviving component of size $\Omega(d^2 \log n)$