Local Computation Algorithms:
Maximal Independent Set

Maximal Independent Set:

\[ \text{def } u \subseteq V \text{ is a "Maximal Independent Set" (MIS) if} \]
\[ \begin{align*}
(1) & \quad \forall u, u' \in U, \ (u, u') \notin E \\
(2) & \quad \exists w \in V \setminus U \text{ s.t. } U \cup \{w\} \text{ is independent} \]

Today's assumption:

\[ G \text{ has max degree } d \]

Note: MIS can be solved via greedy (not NP-complete)
Distributed Algorithm for Mls:

"Luby's Algorithm" (actually a variant)

1. MIS = ∅
2. all nodes set to "live"
3. repeat K times in parallel:
   - A 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")
4. (for purposes of analyses, continue to select selves after die, but don't send color to nbrs)

Thm \( \Pr \left[ \text{# phases til graph empty} \geq 8d \log n \right] \leq \frac{1}{n} \)

Corr \( E[\text{# phases}] \) is \( O(d \log n) \) \( \leftarrow \) can improve!
Main Lemma

\[ \Pr[v \text{ live } \& \text{ added to MIS in round}] = \frac{1}{y} \]

Proof

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

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

\[ \Pr[v \text{ colors self red } \& \text{ no other nbr colors self red}] = \frac{1}{2d} \left(1 - \frac{1}{2d}\right) = \frac{1}{4d} \quad \text{(bound on degree)} \]

\[ \Rightarrow \quad \Pr[v \text{ live after } 4kd \text{ rounds}] \leq \left(1 - \frac{1}{4d}\right)^{4kd} = e^{-K} \]

Setting K:

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

(can do better)
See slides for Local Computation Algorithm (LCA) model.

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

$k = O(\log n) \implies$ 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:

- Run \( Luby \) with \( K = O(d \log d) \) rounds at end, each node \( V \) is one of:
  - live in MIS \( \leftarrow \) set self to red + no nbrs did
  - not in MIS \( \leftarrow \) taken out by nbr

- Use "Parnas - Ron" reduction:
  - simulate \( V \)'s view of computation in sequential manner: \( d^k = O(d \log d) \) queries
  - degree \( \uparrow \) rounds
  - determine whether \( V \) is live/in/not-in
    - if \( V \) is in/not-in then done
    - else \( V \) is alive \( \leftarrow \) What do we do?
Questions:

What is \( \text{prob } \nu \text{ is alive} \)?

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

Lots of live, few dead?

\[ \NO! \]

\[ \Pr [\nu \text{ survives } O(d \log d) \text{ rounds}] \leq e^{-O(d \log d)} \leq \frac{1}{d^2} \]

Most will “die”

\[ \text{don’t worry it won’t be painful!} \]
surviving nodes will be in small connected components

few live but clumped together?

NO!
Surviving nodes will be in small connected components.

This relies heavily on degree bound of graph
- # conn subgraphs small
- Survival of components ≈ independent
"Luby status" Luby with $k = O(d \log d)$:

1. Given $v$, is it:
   - live in MIS $\Leftarrow$ set self to red if no nbrs did
   - not in MIS $\Leftarrow$ taken out by nbr

Luby:
- MIS $\leftarrow \emptyset$
- all nodes set to "live"
- repeat $K$ times in parallel:
  - A 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

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

What is size of component?
Bounding size of connected components:

Claim. After $O(d \log d)$ rounds, connected components of survivors of size $\leq O(p \log d \cdot \log n)$ can find whole component via BFS.

Main difficulty: survival of vertex and neighbors are not independent.
Bounding survivors:

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

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

Note: \( A_v = 1 \Rightarrow B_v = 1 \)

\( \text{eg. } v \text{ survives } \Rightarrow \# \text{round s.t. } v \text{ colors self} \) \( \text{no } \text{w}(\text{N}(v)) \text{ colors self} \)

\[ \Pr [ B_v = 1 ] \leq (1 - \frac{1}{\mu d})^c \cdot \log d \]

\[ \leq \frac{1}{8d^3} \text{ for } c \geq 20 \]

\[ \text{prob survive one round} \]
Bounding size of connected components:

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

\[ B_v = \begin{cases} 1 & \text{if } \exists \text{ round } s.t. v \text{ colors self } \text{red} \text{ and no } w(\text{N}(v)) \text{ colors self } \text{red} \\ 0 & \text{o.w.} \end{cases} \]

Note: \( A_v = 1 \Rightarrow B_v = 1 \)

e.g., \( v \) survives \( \Rightarrow \exists \) round s.t. \( v \) colors self \( \text{red} \) and no \( w(\text{N}(v)) \) colors self \( \text{red} \)

We care about \( A_v \)'s, but \( B_v \)'s have nice independence properties.

Luby:

- MIS \( \rightarrow \) all nodes set to "live"
- Repeat \( K \) times in parallel:
  - If 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")

Might not mean that \( v \in \text{MIS} \)
e.g., if \( v \) died due to nbr being put in MIS

\[ \text{distance 3:} \]

\[ B_v \text{ depends on } B_w \]

\[ B_w \text{ depends on } B_z \]

But \( B_w + B_v \) independent!

\[ \text{distance 2:} \]

\[ B_v + B_z \text{ depend on } \]

\( w \)'s coins so not independent

\[ \text{degree} \leq d \Rightarrow \text{each } B_w \text{ depends on } \leq d^2 \]

\( \text{other } B_w \)'s
Bounding size of connected components:

Claim: After $O(d \log d)$ rounds, connected components of survivors of size $\leq O(d \log d \cdot \log n)$.

⇒ can find whole component via BFS

"brute force"

Proof idea:
- Any large connected component has lots of nodes that are independent (distance $\geq 3$).
- These independent nodes are unlikely to simultaneously survive all sets of size $\frac{1}{2d} n$.
- Do we need to union bound over all sets of size $\frac{1}{2d} n$?
Claim: After $O(d \log d)$ rounds, connected components of survivors of size $\leq O(ply \log d \log n)$.

Proof:

Let $H \leftarrow \text{graph s.t. nodes } \sim B_v$
edges $\sim B_v \cup B_w$ distance 3 in $G$
represent independent events!!

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

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

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!!
How many subtrees in a degree bounded graph?

Known Thm: # non isomorphic trees on w nodes ≤ 4^w

Corr: # size w subtrees in N-node graph of degree ≤ D is ≤ N · 4^w · D^w = N (4D)^w

Why?
• Choose Root in H
• Choose size w tree (shape) from known thm
• Choose placement in H

Total # choices: N · 4^w · D^w
Claim: After $O(d \log d)$ rounds, connected components of survivors of size $\leq O(poly \log(n) \cdot \log(n))$

Proof:

Let $H$ ← graph st. nodes ~ $B_v$

- $\deg(H) \leq d^3$
- $\#$ components in $H \leq \#$ size $w$ subtrees in $H$ of size $w \leq n \cdot (4d^3)^w$
- $\Pr[\text{node $u$ survives}] \geq \frac{1}{8d^3}$
- $\Pr[\text{component of size $w$ in $H$ survives}] \geq \left(\frac{1}{8d^3}\right)^w < \text{since independent!}$

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

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

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

Component of size $\leq w$ in $H$ $\Rightarrow$

Component of size $\leq w \cdot d^3$ in $G$

So unlikely to have any surviving component of size $O(d^3 \log n)$.