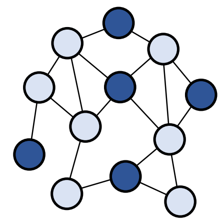
Local Computation Algorithms for the Maximal Independent Set Problem

Maximal Independent Set



Let $G = (V, E)$ be a simple undirected graph

- A set $I \subseteq V$ is an **independent set** if no two nodes $u, v \in I$ are adjacent.
- A set I is a **maximal independent set** if no other nodes can be added.
  - Not **maximum**: we're trying to satisfy a **local** condition here

Local Computation Algorithms (for graphs under adjacency list model)

- Given
  - the input - via an oracle access
  - a query - location on the output
  - (for randomized LCAs: also a random tape T)
- Compute an answer of a computational problem for the input on that query.

MIS: given an oracle access to a graph G and a query vertex v, answer "Is v in the MIS?"

- Answers from all v must form a valid MIS (with high probability over T)
  - Notice: cannot store any previous answers. Order of queries must not matter.

Other example -- maximal matching: Is (u, v) in the approximate MM?

Probe complexity – the **maximum** number of probes made to the oracle to answer any single query.

Old lecture: estimating the size of the maximal matching (or VC) by creating an **oracle**

- Same goal; somewhat different guarantee. Must provide an answer for every node: can't just skip if it's talking too long – an expected probe complexity is not good enough.

Adjacency List Oracle (how LCA accesses G)

- Assume each node v has a unique ID from $\{1, \dots, n\}$.
- Given a node v (by ID) and an integer i, returns the i^{th} neighbor of v if $\deg v \le i$ (or $\perp$ otherwise).
- Also assume a known bound Δ on the maximum degree.

Today: 2 LCAs for MIS:

- A **deterministic** LCA for MIS with probe complexity $\Delta^{O(6^\Delta)} \log^* n$
  Note: many (early) papers assume constant Δ
- A **randomized** LCA for MIS with probe complexity $\Delta^{O(\log^2 \Delta)} \log n$

Recall: **Distributed LOCAL model**

- G is both the **input graph** and the **network structure**.
- Each communication round: nodes send (unlimited-sized) messages to neighbors.
- Computation done between rounds (no communication
- Goal = minimize #rounds until all nodes have an answer
- PR reduction: an r-round distributed algorithm can be simulated using $\Delta^{O(r)}$ queries
  - probe for all nodes at distance up to r away, then simulate distributed LOCAL

A **deterministic** LCA for MIS with probe complexity $\Delta^{O(6^\Delta)} \log^* n$

version presented here simplified from [Even Medina Ron '14]

Claim: given a c-coloring ϕ, can compute an MIS in c (distributed) rounds.

- $\phi: V \to \{1, \dots, c\}$ such that for $(u, v) \in E$, $\phi_u \neq \phi_v$

Coloring-to-MIS
maintain an independent set I (initially empty)
for each color $i = 1, \dots, c$
each node v with $\phi_v = i$ joins I if no neighbor is already in I

Claim: The algorithm computes a MIS in c rounds.

- Output is an independent set: $v \in I$ cannot be adjacent to u of lower color since we check explicitly whether $u \in I$; v cannot be adjacent to node of the same color since ϕ is a valid coloring.
- Output is maximal: If v could have been added, it would have been added during ϕ_v^{th} round.

Note: think of this as a simulation of greedy algorithm

- rather than using a random ranking idea, here colors are used instead
- adjacent nodes are of different colors so there is no need for tie-breaking
- can bound query tree depth by #colors

Parnas-Ron reduction: an LCA can simulate (distributed) **Coloring-to-MIS** with $\Delta^{O(c)}$ probes.

⇒ If we have a **Coloring-LCA** for 6^Δ-coloring, then we can create an LCA for MIS:

- for a query v, first call **Coloring-LCA** on every node at distance $\sim 6^\Delta$ from v
- apply PR: simulate **Coloring-to-MIS** on these $\Delta^{O(6^\Delta)}$ nodes using the obtained colors

Claim: There exists a deterministic **Coloring-LCA** for 6^Δ-coloring with $O(\Delta \log^* n)$ probes.

By above, this implies the desired $\Delta^{O(6^\Delta)} \log^* n$-probe LCA for computing MIS.

- must **multiply** the probe complexities (unlike distributed, we cannot just add them)
  - **Coloring-LCA** takes $O(\Delta \log^* n)$ probes to compute the color of a node
  - to compute a MIS we need the colors of up to $\Delta^{O(6^\Delta)}$ nodes: $\Delta^{O(6^\Delta)}$ calls to **Coloring-LCA**

Constructing Coloring-LCA

Step 1 **Decompose** the graph into Δ different **oriented forests**

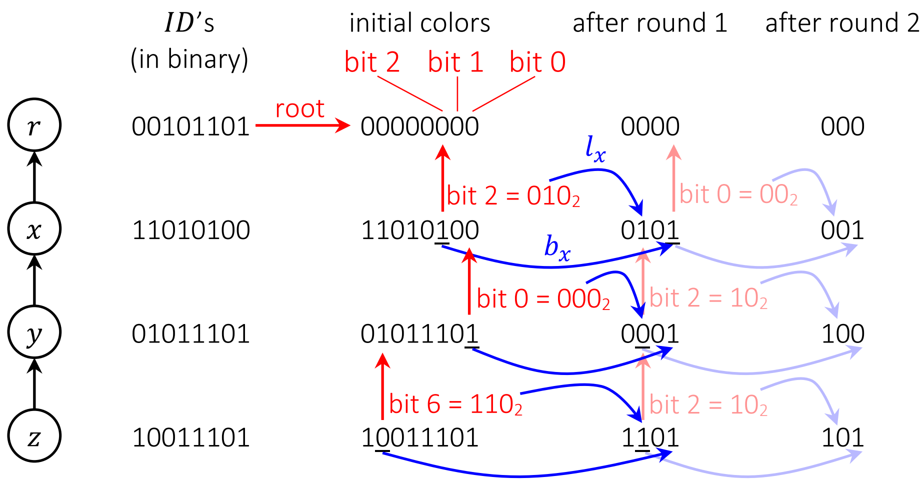
- Let $E^i = \{(u, v): ID(u) < ID(v)$, v is the i^{th} neighbor of $u\}$ (each (u, v) is in a unique E^i)
- $G^i = (V, E^i)$ has maximum out-degree 1; form trees (roots = whose without an out-neighbor)
- Can find the out-neighbor v of u in one probe by looking at i^{th} neighbor of u and compare IDs

Step 2 **6-color oriented trees** in each G^i in $O(\log^* n)$ rounds/probes [ColeVishkin'86]

$\phi_u \leftarrow 0$ if u is a root (i^{th} neighbor of u has lower ID than u)
$ID(u)$ if u is not a root ($\in \{1, \dots, c\}$) ($\log n$ bits)
Repeat $\Theta(\log^* n)$ rounds compute a new color, ensure valid coloring, hopefully reduce #colors
if u is a root, $c_u \leftarrow 0$
else let v be u's "parent" (i^{th} neighbor of u, so edge $u \to v$)
$l_u \leftarrow$ index of the least significant bit (little endian) where ϕ_v differs from ϕ_u (0-based)
$b_u \leftarrow$ the value of u's l_u^{th} bit
$\phi_u \leftarrow (l_u, b_u)$ (new color: just concatenate l_u and b_u together)

Example 6-coloring

compare with parent, look for first different bit from right to left, put index l followed by that bit itself b



- **Claim:** always maintains a valid coloring: (say consider edge $u \to v$)
  - initial colors: begin with unique ID's; roots cannot be adjacent on E^i
  - induction: after each iteration, need to show $\phi_u \neq \phi_v$ (see example, round 2)
    - if v is a root, $b_u = 1$ while $\phi_v = 0$'s, so $\phi_u \neq \phi_v$
    - else, suppose $l_u = l_v$, then the l^{th} bits b_u and b_v must be different, so $b_u \neq b_v$
- **Claim:** computes a valid 6-coloring in $O(\log^* n)$ rounds
  - In one round, K bits → $\lceil \log K \rceil + 1$ bits, so takes $\Theta(\log^* n)$ (induction: just $1 + \log^* n$)
  - cannot go below 6: $l_u \in \{0, 1, 2\}$, $b_u \in \{0, 1\}$, stuck; needs a different algorithm

Step 3 **Combine** into a 6^Δ-coloring over G

- formed by the vector of c_u^i's: length Δ, each entry is one of the 6 possible colors

Claim: Can implement **Coloring-LCA** using $O(\Delta \log^* n)$ probes

- For each G^i, must follow i^{th} out-neighbors from v for at most $O(\log^* n)$ steps, learn all the ID's, then apply (simulate) the procedure for 6-coloring.
- Aside, the distributed version takes $O(\log^* n)$ rounds since can do all Δ graphs in parallel.

Note: [EMR] can get $\Delta^{O(\Delta^2)} \log^* n$ with more black-box distributed coloring
best known: $\Delta^{\tilde{O}(\sqrt{\Delta})} \log^* n$ [Fraigniaud Heinrich Kosowski '16]

A **randomized** LCA for MIS with probe complexity $\Delta^{O(\log^2 \Delta)} \log n$

version presented here based on distributed algorithm by [Barenboim Elkin Pettie Schneider '15]

Lemma There exist a $O(\log^2 \Delta)$-round distributed algorithm **Shattering** that computes an independent set I such that with high probability, the graph induced by $V \setminus N^+(I)$ contains no connected component of size $\geq \Delta^4 \log n$. [BEPS'15]

This implies the desired LCA:

- By PR, we have **Shattering-LCA** that computes whether $v \in I$ (in the lemma) in $\Delta^{O(\log^2 \Delta)}$ probes.
- If $v \in I$ then YES, v is in the MIS
- Else ($v \notin I$), check v's neighbors: if there's a $u \in N(v) \cap I$ then NO, v is not in the MIS.
- Else ($v \notin N^+(I)$)
  - DFS from v, call **Shattering-LCA** on reached nodes to identify the entire component $C_v \subseteq V \setminus N^+(I)$ containing v. ($|C_v| \leq \Delta^4 \log n$, so need $\mathrm{poly}\,\Delta \cdot \log n$ calls to **Shattering-LCA**.)
  - Solve MIS of C_v **deterministically** in **consistent** way, answer YES or NO for v accordingly.
  - queries anywhere on this component must give consistent answers; e.g., compute lexicographically first MIS = greedy via ID order

Strategy: Define **base sets** S of size t (later will pick $t = \log n$)

- **(1)** Construct an algorithm that any base set **survives** ($S \subseteq V \setminus N^+(I)$) with small prob ($\Delta^{-\Omega(t)}$).
- **(2)** Show that any connected component of size $t\Delta^4$ must contain a base set of size t.
- **(3)** Show that there are not too many base sets ($n(4\Delta^5)^t$).
- (Prob **(1)** × # base sets **(3)**) small ⇒ no base set exists ⇒ by **(2)**, no large component exists

Base sets

- Let H be the distance-5 graph of G. Namely, $E(H) = \{(u,v): \mathrm{dist}_G(u,v) = 5\}$.
- A set S is a base set if
  - S is the vertex set $V(T)$ of a tree T on H, and
  - for any $u, v \in S$, $\mathrm{dist}_G(u,v) \geq 5$.

<u>Constructing the Distributed Algorithm</u>

<u>Luby's Step</u> [Luby '85] – building block for **Shattering**

each node v selects itself with probability $\frac{1}{\Delta+1}$ (there are many other variations on selection condition)
if v is the only node in $N^+(v)$ that selects itself, add v to I and remove $N^+(v)$ from G

Claim: Each node v with $\deg v \geq \Delta/2$ is removed with **constant** probability $p > 0$. "v is **vulnerable**."

- $\Pr[\text{no } u \in N^+(v) \text{ selects itself}] \geq 1 - \prod_{u \in N^+(v)} \left(1 - \frac{1}{\Delta+1}\right) \geq 1 - \left(1 - \frac{1}{\Delta+1}\right)^{\Delta/2+1/2} > 1 - \frac{1}{\sqrt{e}}$
- Let u = **lowest *ID*** node in $N^+(v)$ that selects itself
  - $\Pr[u \text{ joins } I] \geq \prod_{w \in N(u)} \left(1 - \frac{1}{\Delta+1}\right) \geq \left(1 - \frac{1}{\Delta+1}\right)^{\Delta} > \frac{1}{e}$.
  - enforce "lowest ID" because when we consider $w \in N(u)$, we cannot condition on any other nodes that already select themselves (else probability of u joining I will be 0); more precisely, $\Pr[u \text{ joins } I] = \prod_{w \in N(u) \setminus \{u' \in N^+(v), ID(u') < ID(u)\}} \left(1 - \frac{1}{\Delta+1}\right)$
- Above argument works for any u, so overall, v is removed with prob $\geq \left(1 - \frac{1}{\sqrt{e}}\right)\left(\frac{1}{e}\right) > 0.14 = p$.
  - Note: this already gives $O(\log^2 n)$-round distributed algorithm (Δ halved whp after $O(\log n)$ rounds); $O(\log n)$ under careful analysis.

<u>Shattering</u> ($O(\log^2 \Delta)$ distributed rounds – putting **Luby's steps** together)

for $k = \lceil \log \Delta \rceil, \dots, 1$ "**iteration**"
  // maximum degree $\leq 2^k$ at the beginning of each iteration
  perform $c_1 \log \Delta$ rounds **Luby's Step** using probability $\frac{1}{2^{k+1}}$
  for each v with $\deg v \geq 2^{k-1}$, remove v from G, and put v in L (v is lucky; deal with it later)
add all remaining (isolated) nodes to I

⇒ After **Shattering**, surviving nodes are $L \setminus N^+(I)$. We will bound max connected component size in L.

Observation

For each node $v \in L$, in the iteration that it finally joins L, it was **vulnerable** throughout the entire **iteration**. (The exact iteration is not known in advance.)

Claim: For any node v, $\Pr[v \in L] \leq p^{c_1 \log \Delta} = \Delta^{-c_2}$.

- It must survive all $\Theta(\log \Delta)$ **vulnerable** rounds of the entire **iteration**.

Claim (1) For any set S of t nodes such that $\text{dist}_G(u, v) \geq 5$ for every $u, v \in S$, $\Pr[S \subseteq L] \leq \Delta^{-c_2 t}$.

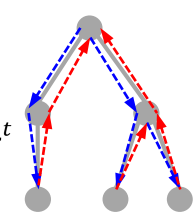
- Event that v joins I in each iteration only depends on coin tosses at distance ≤ 2 away, so these events are independent for nodes at distance ≥ 5 away.
  - Probability that S survives a particular iteration $\leq p^{\text{\# vulnerable nodes}}$.
- Outcomes are independent between any **Luby's Step**.
  - Imagine the whole coin tosses "table" being fixed in advance; only revealed row-by-row.

Claim (2) Any connected component size $t\Delta^4$ must contain a base set of size t.

- pick an initial node $v \in S$, remove the ball $\{u: \text{dist}_G(u, v) \leq 4\}$ from S
- continue picking a node at distance exactly 5 from some removed node
  - must be adjacent on H to a picked node
  - cannot have distance < 5 to any picked node since balls around picked nodes removed
- each removed ball has $\leq \Delta^4$ nodes ⇒ t nodes can be picked ⇒ form a base set

Claim (3) There are at most $n(4\Delta^5)^t$ possible base sets.

We show there are $\leq n(4\Delta^5)^t$ possible trees on H. (over-count since ignoring distance ≥ 5 condition)



- Structure-wise, there are $\leq 4^t$ **plane trees** with t nodes
  - plane trees: the subtrees of each node are linearly ordered
  - # plane trees $=$ # DFS sequences defining the tree structure
    $\leq$ # sequences with $(t-1)$ ↓'s and $(t-1)$ ↑'s $< 2^{2(t-1)} < 4^t$
  - actually # plane trees with $t-1$ edges $= C_{t-1}$ (Catalan number)
- choose the first node in n ways (arbitrary node in H)
- for each subsequent node (child), its parent is already determined by the tree structure, so can choose each child in $\leq \Delta^5$ ways (max degree in H)
- total $4^t \cdot (n \cdot (\Delta^5)^{t-1}) \leq n(4\Delta^5)^t$

Lemma With high probability, L contains no connected component of size $\Delta^4 \log n$.

Set $t = \log n$, sufficiently large constant c_2; probability that there exists a large connected component

$$\leq n \cdot (4\Delta^5)^t \cdot \Delta^{-c_2 t} = n^{1 + \log 4 + 5\Delta - c_2 \Delta} = n^{-c}.$$

Note: Best known **Shattering** $O(\log \Delta)$ distributed rounds ⇒ $\Delta^{O(\log \Delta)} \log n$ LCA probes [Ghaffari '16]