D S

(Par

Outline

- Independent Set (IS)

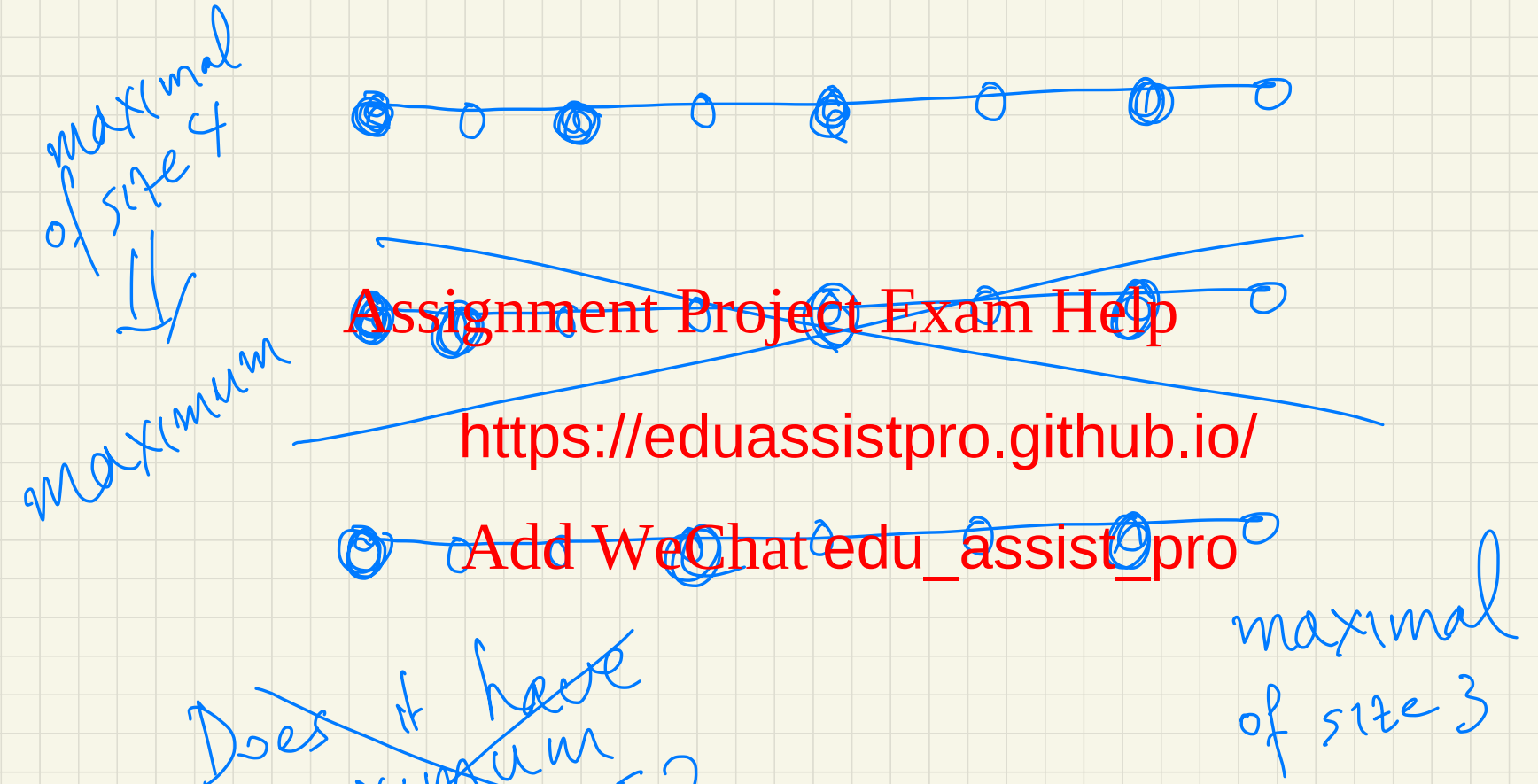
- Distributed IS
  1. Distributed Slow MIS
  2. Distributed Fast MIS

Inependent Sets

- **IS:** Given an undirected Gra $V, E)$ an independent set is a subset of nodes $U \subseteq$ o nodes in U are adjacent.

- **MIS:** An independent set is maximal if no node can be added without violating independence.

- **MaxIS:** An ind ity is called maximum.



Independent Set (IS)

- An IS is a set of nodes of the graph suc em are not adjacent.

- We also have maximal and maximum independent sets.

maximal maximum -al

-al -um -al

- Every MIS (Maximal Independent Set) is a dominating set.
- In general, the size of every MIS can be larger than the size of an optimal minimum dominating set by a factor of $\Omega(n)$.[a]

[a] We won't prove this here.

A set D is a dominating set, if every of the graph is adjacent to some vertex in D

Coloring and Independent Sets

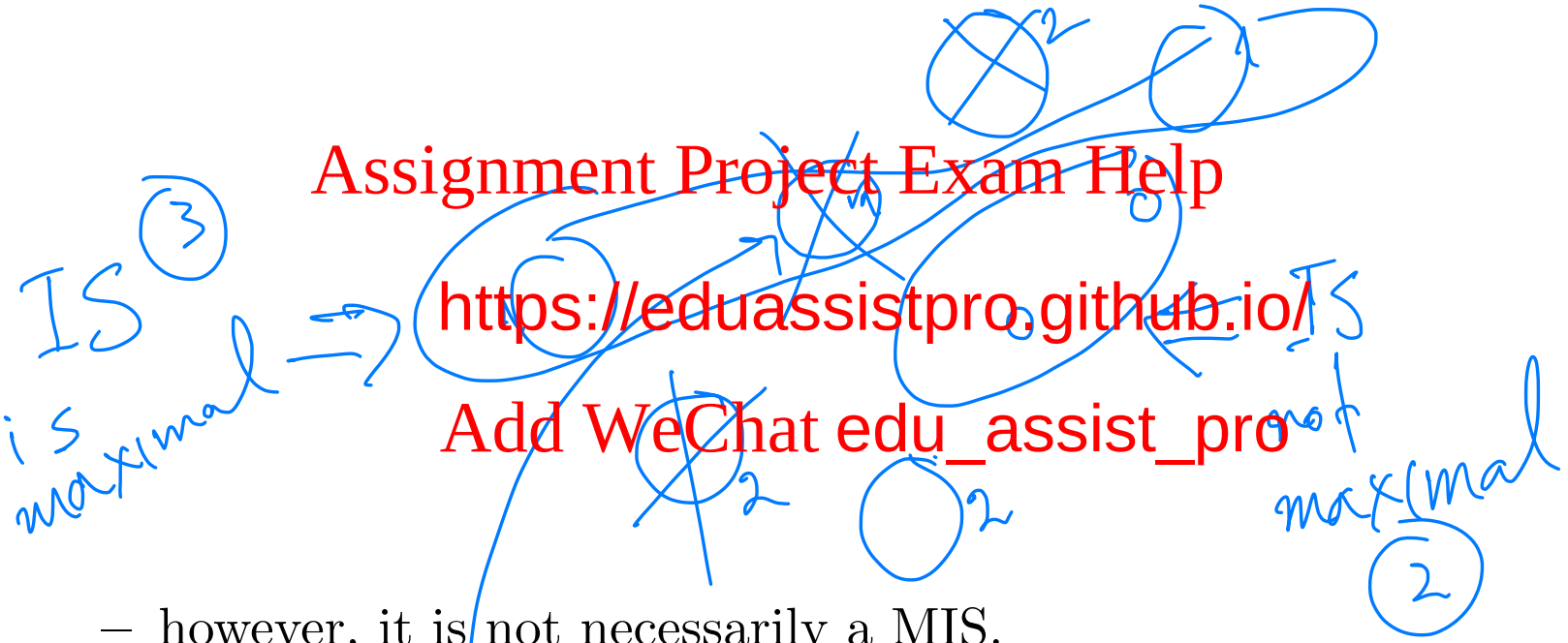
- **Example 1** Graph has two endent sets (MIS), but only one is a maximum indep IS).

In a colored graph, the vertices of a color form an IS.

- **Example 2**

Coloring and Independent Sets

- There is a relation between inde d node coloring:
  - each color class is an indepen

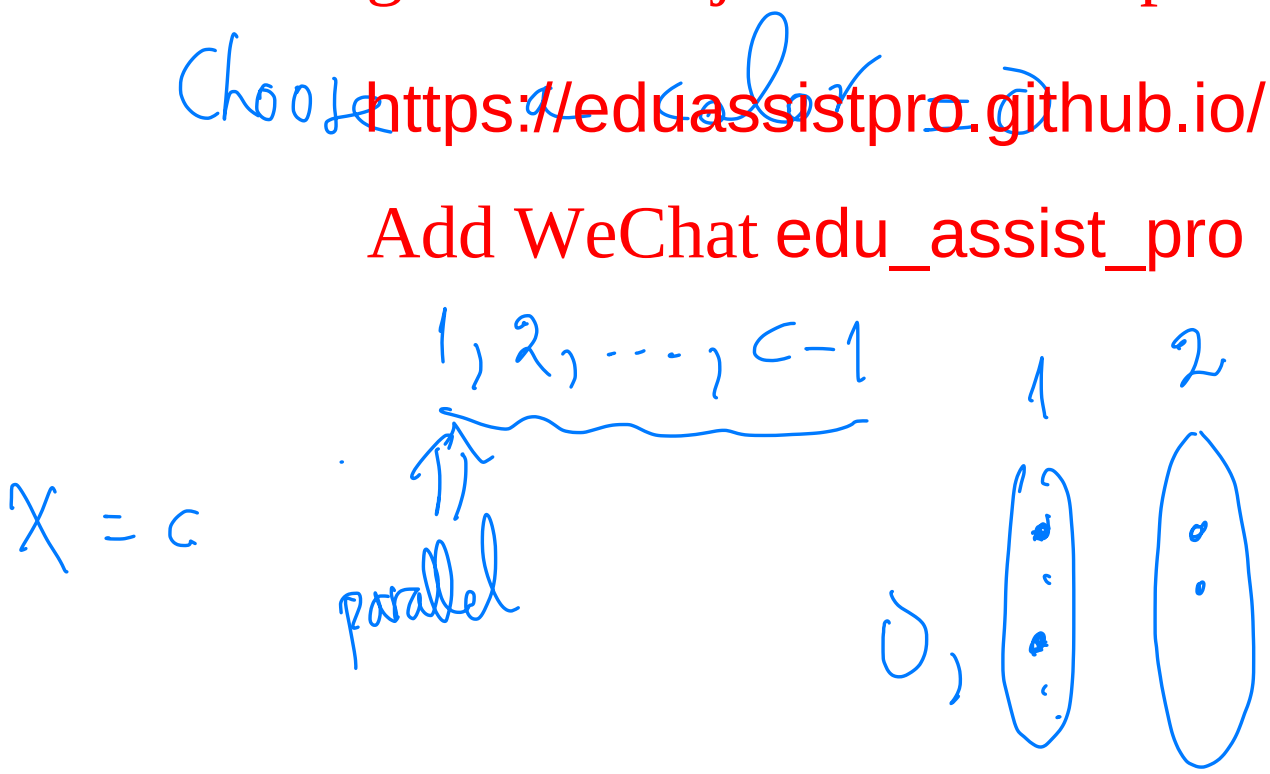


  - however, it is not necessarily a MIS.

From Coloring to Independent Sets

- Starting with a coloring, one ca rithm:

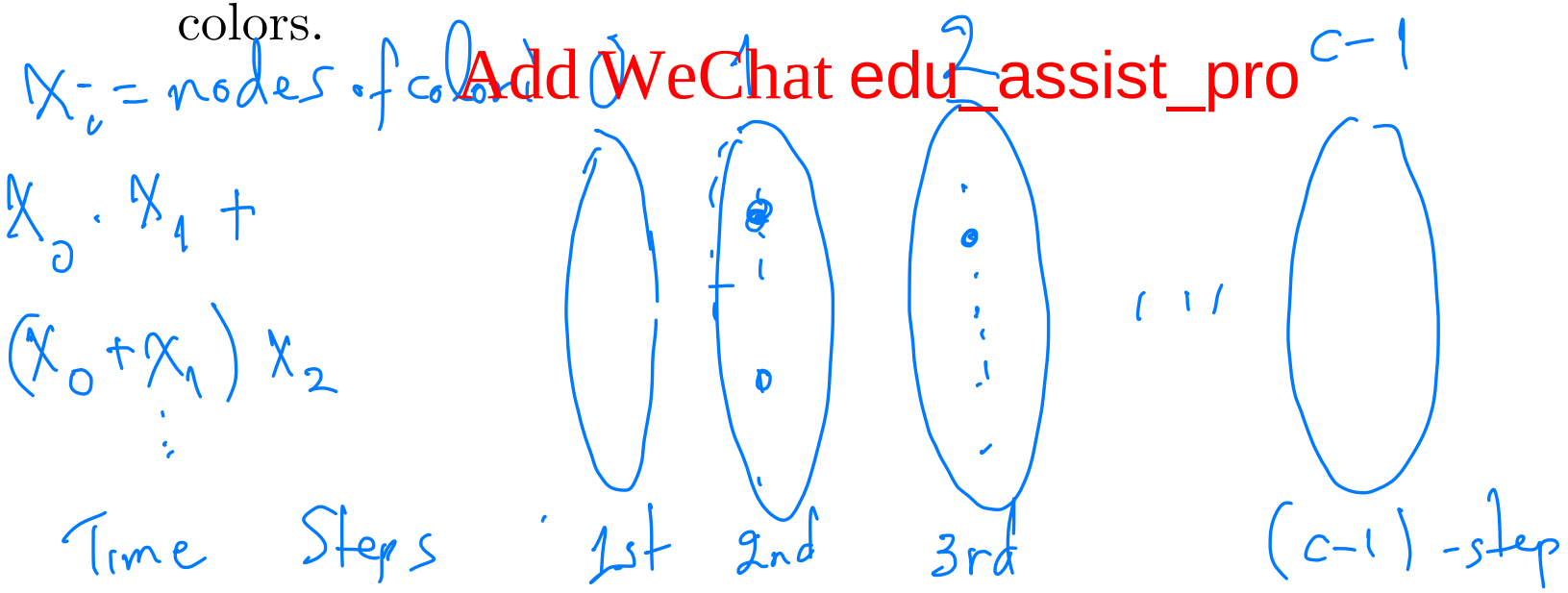
1. We first choose all nodes of the fir
2. Then, for each additional color we add "in parallel" (without conflict) as many nodes as possible.



From Coloring to Independent Sets: Analysis

- **Theorem 1** *Given a colori* *t needs C colors*
  and runs in time T, we can c *ime C + T.*

- Time complexity:
  - the T in the time complexity comes from the coloring algorithm, and
  - the C in the ti ber of colors.



Time Complexity is C + T

Message Complexity:

X_i = # of nodes of color i

$X_0 X_1 + (X_0 + X_1) X_2 + (X_0 + X_1 + X_2) X_3$

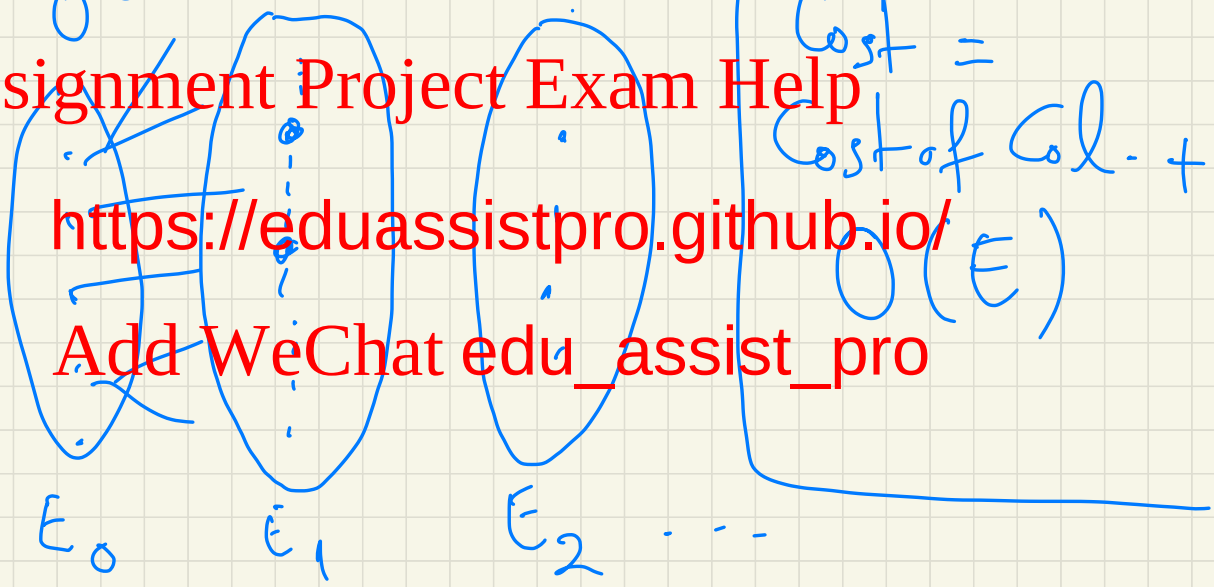
$\cdots + (\underbrace{X_0 + \cdots + X_{c-2}}_{n^2}) X_c$

Message Complexity is $O(c \cdot n^2)$, $c = \chi$ = chromatic number

In addition we need the cost of the coloring alg.

{Message Complexity of IS algorithm} ≤ {Message Compl. of Coloring Algorithm} + {above calculation}

Joyce: Can't we do it in $O(E)$ messages. YES



It needs: $E_0 \cup E_1 \cup \cdots \subseteq E$

Related Topic: Set Cover (SC)

- Given a set of elements { called the universe) and a collection S of m sets whe e universe, the set cover problem is to identify the smallest sub-collection of S whose union equals the universe.

$S = \{S_1, \ldots, S_m\}$

1. For example, consider the universe $U = \{1, 2, 3, 4, 5\}$ and the collection o 4},{3,4},{4,5}}. Clearly the uni an cover all of the elements with the following, smalle $\{\{1, 2, 3\}, \{4, 5\}\}$.

2. A company needs to buy a certain amount of varied supplies and there are suppliers that offer various deals for different combinations of materials (Supplier A: 2 tons of steel + 500 tiles for $\$x$; Supplier B: 1 ton of steel + 2000 tiles for $\$y$; etc.). You could use set covering to find the best way to get all the materials while minimizing cost.

Issues

- Computing a maximum inde xIS) is a notoriously difficult proble

  - Equivalent to maximum clique on the complementary graph.
  - Both problems are NP-hard, in fact not approximable within $n^{1/2-\epsilon}$.

- MIS and MaxIS ca

  - On a star graph MIS is $\Theta(n)$ s axIS.

{c} is MIS

{1,2,...,7} MaxIS

MIS : size 1

MaxIS : size n-1

n leaves

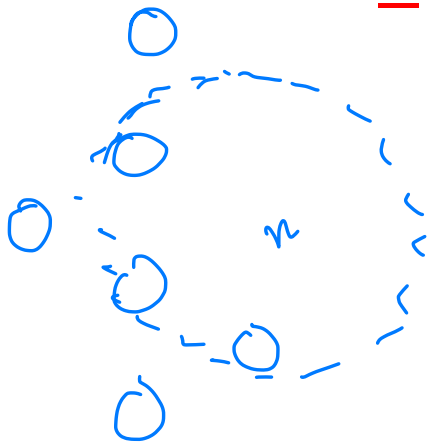
1 central node

Star with n nodes

Examples: MIS

- Example 1

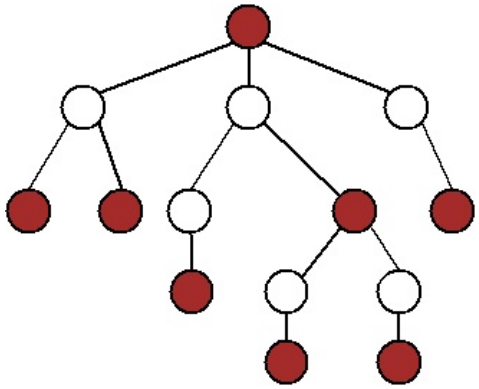
- Example 2

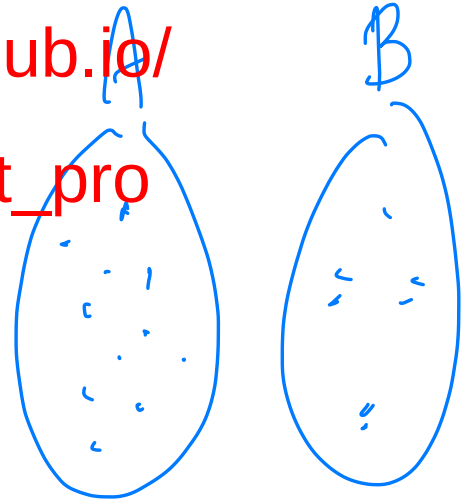


Examples: MIS

- Example 3

- Example 4



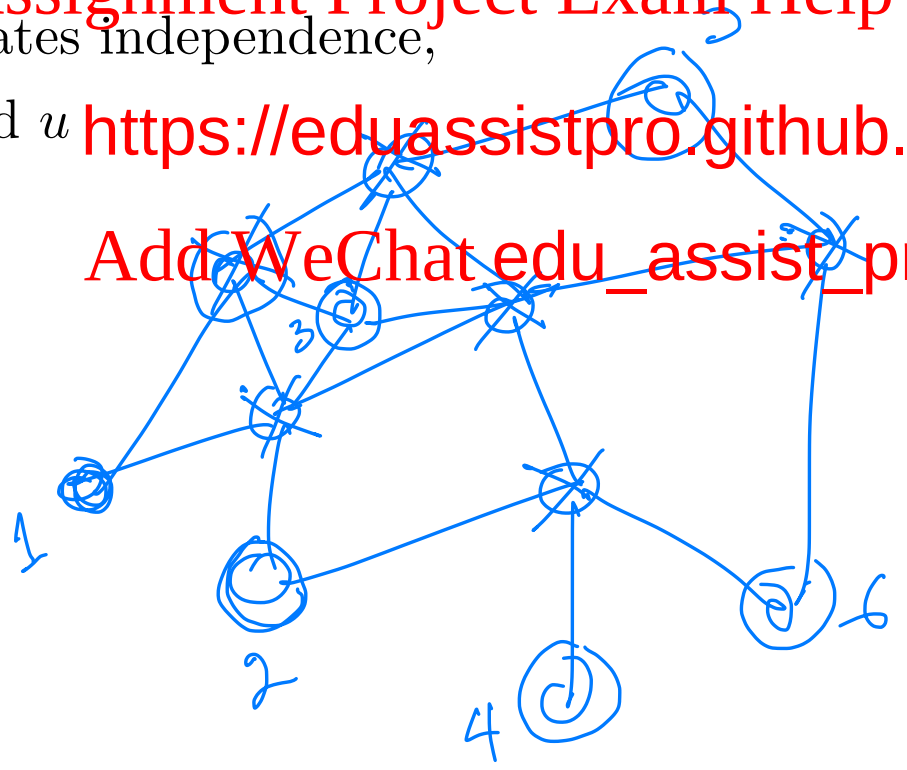


Computing MIS

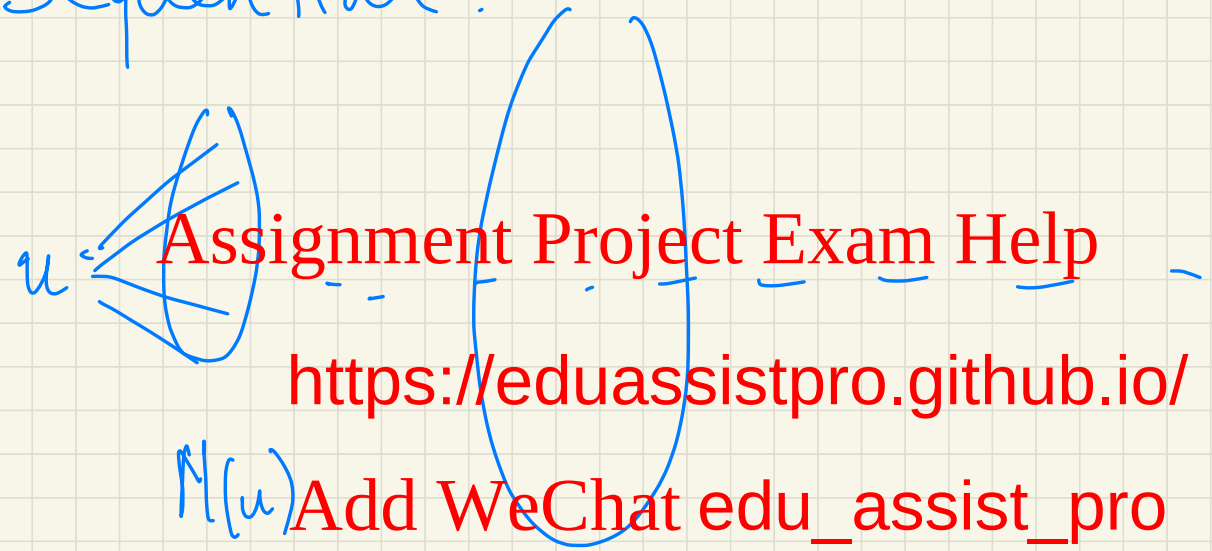
- Computing a MIS sequentiall

1. Scan the nodes in arbitrary ord
2. If a node u does not violate independence,
   - add u to the MIS.

3. If u violates independence,
   - discard u



Sequential:



Algorithm: Lexicographic $MIS(G)$

- Previous algorithm sometim ws. Consider a graph $G=(V,E)$ in which th icographically ordered.

1: $I = \emptyset, V' = V$
2: **while** $V' \neq \emptyset$ **do**
3: Choose
4: $I \leftarrow I \cup$
5: $V' \leftarrow V' \setminus (\{v\} \cup N(v))$
6: Return I;

- With this simple greedy algorithm, we can find a MIS in $O(|V| + |E|)$ time.
- The main question is how to compute a MIS in a distributed manner.

Distributed Slow MIS

- Main idea is to give priority to nod .
- **Slow MIS**
- Requires Node IDs
- Every node v executes the following code:

1. **if** all neighbo ave decided not to join the MIS t
2. v decides to join the MIS
3. **end if**

v queries its neighborset N(v)

Complexity

- **Theorem 2** *Algorithm Sl* *mplexity of* $O(n)$ *and a message complexity of*

- Slow MIS is not better than the sequential algorithm in the worst case, because there might be one single point of activity at any time.

Issues

- Using Theorems 1 and 2 we get a di ministic MIS algorithm for cycles (also for tre degree graphs) with time complexity $O(\log^* n)$ (will cover this later in class).
  - First do the colouring in $O(\log^* n)$ rounds.
  - Choose all nodes of the first color.
  - For each addit ut conflict) as ma

- With a lower bound argument one can sho deterministic MIS algorithm for rings is asymptotically optimal.
  - Because in the ring MIS is "essentially" the same as coloring.
- There have been attempts to extend the 6-Color Algorithm to more general graphs, however, so far without much success.

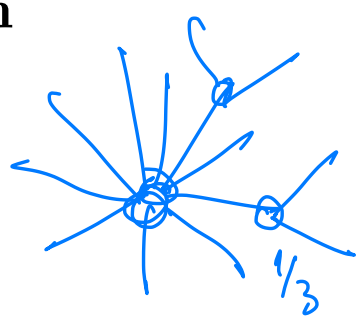
Is There a Faster Algorithm?

- Given that "Slow MIS" is not bet ential algorithm in the worst case

  - Is there a faster MIS?

- In the sequel we give a probabilistic algorithm with $O(\log n)$ expected termination time.

Goal: Find a parallel MIS algorithm



- Consider algorithms of the for

  1. $I = \emptyset, V' = V$
  2. While G' is not the empty graph
     (a) Choose a random set of vertices $S \subseteq V$ by selecting each vertex v independently with probability $\frac{1}{d_v}$, where d_v is the degree of

     (b) For every edge (oints are in S, then remove the vertex of lower de S (break ties). Denote the set after this step as S'.

     (c) Remove S' and $Neighbor(S')$ and all adjacent edges from G'.
     (d) $I \leftarrow I \cup S'$

Distributed Fast MIS[a]

- Algorithm operates in synchr rouped in phases.

- A single phase is as follows:

not required

1. Each node v marks itself with probability $\frac{1}{2d(v)}$, where $d(v)$ is the current degree of v.

2(a) If no higher degree neighbor of v is also marked, node v joins the MI e */

(b) If a higher degre v unmarks itself again

/* If neighbors have same degree, ties broken by ID */

3. Delete all nodes that joined the MIS and their neighbors /* as they cannot join the MIS anymore. */

[a] A more general form of this algorithm assigns real numbers (in the range $[0, 1]$) as weights at the nodes. An alternative version is to label the vertices with a random permutation.

Issues

- Correctness in the sense that th uces an independent set is relatively si

  - Steps 1 and 2 make sure that if a node v joins the MIS, then v's neighbors do not join the MIS at the same time.
  - Step 3 makes sure that v's neighbors will never join the MIS.

- The algorithm eve e node with the highest degree will mark itself at s .

- The only remaining question is how fast t terminates.
  - This is not easy to figure out!

Exercises[a]

1. Show that any maximal matchin ation of a maximum matching.

2. Let $G = (V, E)$ be the graph for which we want to construct the matching. Define the auxiliary graph G' as follows:

   - for every edge in G there is a node in G';
   - two nodes in G' are adjacent if their respective edges in G are adjacent.

   Show that a (maximal) independent set in G' is a (maximal) matching in G, and vice versa.

[a] Do not submit!