Lecture Notes CS:5350 Luby's MIS Algorithm
Lecture 11: Oct 1, 2019
Scribe: Mohammad Shafkat Islam

In this lecture we develop a distributed deterministic algorithm of Luby for finding a maximal independent set in an undirected graph.A maximal independent set is one contained in no larger independent set. A maximal independent set need not be of maximum cardinality among all independent sets in the graph.

Theorem. *There is a distributed deterministic $\mathcal{O}(n)$ polynomial time algorithm for finding a maximal independent set in a graph starting with an arbitrary vertex and adding vertices until all remaining vertices are connected to at least one vertex already taken.*

Using randomization we can do much better. MIS can be computed in $\mathcal{O}(logn)$ rounds.

Idea 1. *Each node picks a random ID. Then we run deterministic MIS algorithm using these IDs*

Intuition. *The algorithm proceeds in rounds; in every round it finds an independent set S, and adds S to I (initially I is empty) and deletes $S \cup \Gamma(S)$ from the graph. Note that it is very easy to implement this algorithm in a linear (i.e., $\mathcal{O}(n)$) rounds.*

The randomization runs faster. Using randomization we can avoid the situation shown in Figure 1.

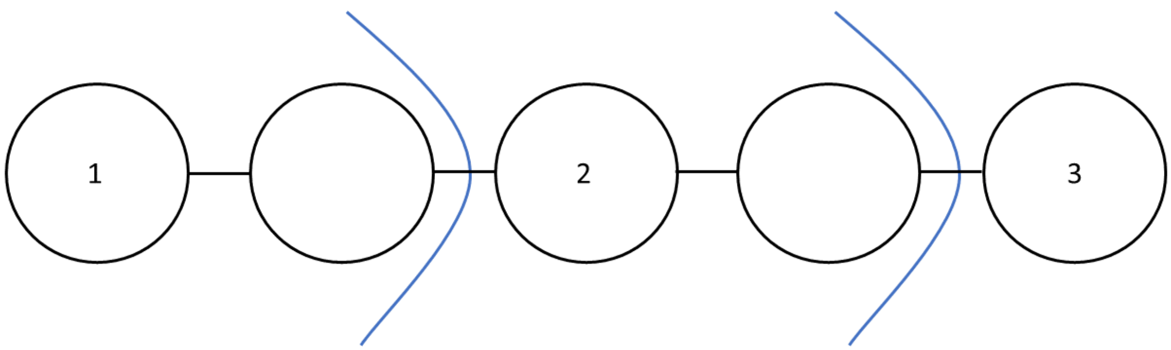

Figure 1: Randomized algorithm allows to avoid this situation.

Idea 2. *Each node attempts to join the MIS with some probability. There will be conflicts when two neighbors both express desire to join the MIS. The conflicts are resolved by some simple deterministic tie breaking rule. Some nodes will end up joining MIS. These nodes and neighbors are deactivated. Then we repeat.*

MIS Algorithm

We will only discuss algorithm based on Idea 2 because it is easier to analyze. Idea 1 also leads to an $\mathcal{O}(\log n)$ round algorithm, but we will not discuss it because it is harder to analyze. This is also called "Luby's Algorithm".

Each node v chooses itself to be in the MIS with probability $1/(2d(v))$ (this is slightly smaller than $1/(d(v)+1)$ — this "slack" is useful for analysis). Where $d(v)$ is not just v's degree in the original graph, but v's degree in the graph induced by active nodes- the so called active graph. To handle the scenario, that two neighboring nodes choose themselves, ties are broken based on the degrees: the tie is broken in favor of the higher degree node. If the degrees or the same, then tie is broken based on IDs.This is slightly counter-intuitive, but makes sense: if a higher degree node is favored in tie breaking, it will eliminate more (of its) neighbors.

Psuedocode for Luby's Algorithm

$status(v)$ = undecided
while $status(v)$ = undecided **do**
  if $d(v) = 0$ **then**
    $status(v)$ = yes // belongs to MIS
  else
    v marks itself with probability 1/(2d(v)) // marking step (refers to degree(v) in active graph
Round 1
    v notifies neighbors that it is marked and also sends its current degree
    if v receives a message from a marked neighbor of higher degree (or equal degree but higher ID) **th**
      v unmarks itself // tie-breaking step
    if v is still marked **then**
      $status(v)$ = yes
Round 2
      v notifies all neighbors its status
    if v receives a message from a neighbor that is in MIS **then**
      $status(v)$ = no

Analysis of Luby's Algorithm

Theorem. *This algorithm terminates in $\mathcal{O}(\log n)$ with probability $\geq (1-\frac{1}{n})$.*

Definition 1. *With high probability for graph algorithms refers to probability $\geq (1-\frac{1}{n^c})$ for $c \geq 1$*

We will present a simpler analysis of the algorithm that shows a weaker time bound of $\mathcal{O}(\log n \log \Delta)$ rounds, where Δ is the maximum node degree.

Theorem. *MIS Algorithm 1 runs in $\mathcal{O}(\log n \log \Delta)$ rounds with high probability.*

Proof. As stated, in the notes on "Distributed Network Algorithms" by Gopal Pandurangan, posted on the course website, the main idea of the analysis is as follows. We divide the algorithm into phases (for the sake of analysis): in the first phase, we will consider nodes having degree between ($\Delta/2$, Δ]; in phase i the nodes considered will have degree between ($\Delta/2^i$, $\Delta/2^{i-1}$]. Thus there will be $\mathcal{O}(\log \Delta)$ phases. At the end of phase i, the status of all nodes of degree higher than $\Delta/2^i$ would have been decided. We will show that each phase lasts for $O(logn)$ rounds with high probability. [?]

Let high degree refer to nodes with initial degree in $(\Delta/2\ \Delta\ ]$. Let v be a high degree node. We lower bound the probability that the status of v will be determined in one iteration. This can be done in two ways in a iteration: (1) v enters MIS or (2) a neighbor of v enters MIS. [**?**]

We lower bound the probability that a neighbor of v enters MIS as follows in two steps:

(Event A):A neighbor of v, say w, marks itself; and

(Event B) At least one of v's marked neighbors remains marked after the tie-breaking step.

In next class we will lower bound the probability that both Event A and Event B will happen is constant.

$\square$