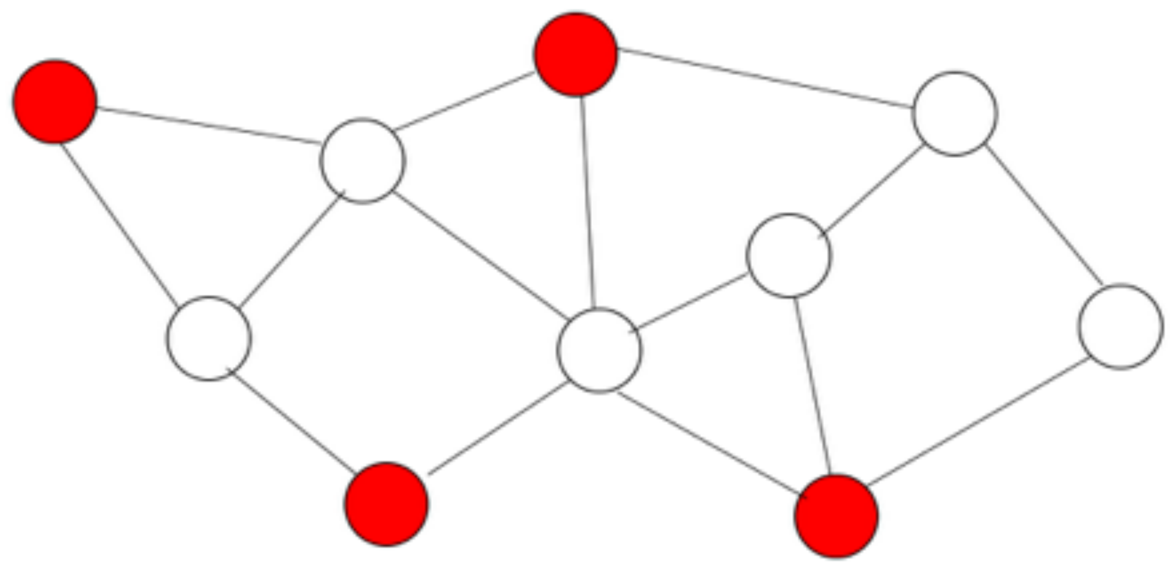
Luby's Algorithm

Ciccolone Sara **D'Andrea Gabriella**

A.A. 2015/2016

Remark MIS



A maximal Indipendent Set **(MIS)** in an undirected graph is a maximal collection of vertices *I*, subject to the restriction that no pair of vertices in *I* are adjacent.

The MIS problem is to find a MIS.

Applications of MIS Algorithms

A growing number of parallel algorithms use the MIS algorithms as a subroutine.

Luby show that the problem is in random NC(RNC) which means that there is a parallel algorithm using polynomially many processors that can make calls on a random number generator such that the expected running time is polylogarithmic in the size of the input.
Luby also gives a deterministic NC algorithms, in fact we use this strategy to convert a specific parallel Monte Carlo algorithms into a deterministic algorithm.

Monte Carlo MIS Algorithm for Maximal Matching Problem

- High level description of the algorithms

Input: G = (V,E) is an undirected graph

Output: MIS I ⊆ V.

begin
$I \leftarrow \emptyset$
$G' = (V', E') \leftarrow G = (V, E)$
while $G' \neq \emptyset$ **do**
  begin
    select a set $I' \subseteq V'$ which is independent in G'
    $I \leftarrow I \cup I'$
    $Y \leftarrow I' \cup N(I')$
    $G' = (V', E')$ is the induced subgraph on $V' - Y$.
  end
end

Luby's Algorithm

All of randomization is incorporated in one step of the algorithm called the **Choice Step**:

If v is a vertex and A is a set of vertices define:

$$\begin{aligned} N(v) &= \{u \mid (u,v) \in E\} = \{\mathit{neighbors} \text{ of } v\} \\ N(A) &= \bigcup_{u \in A} N(u) = \{\mathit{neighbors} \text{ of } A\} \\ d(v) &= \text{the } \mathit{degree} \text{ of } v = |N(v)| \ . \end{aligned}$$

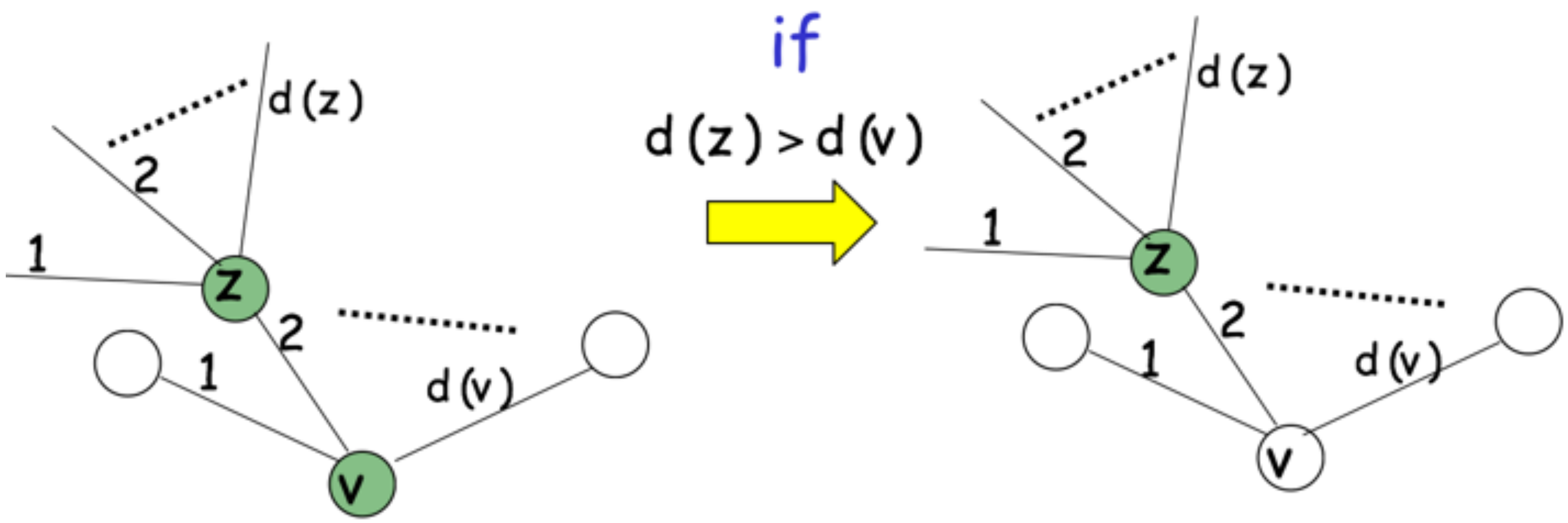
1. Create a set *S* of candidates for *I* as follows. For each vertex v in a parallel include v ∊ *S,* with probability 1/ 2d(v), where d(v) is the degree of v.

2. For each edge in E, if both its end point are in *S,* discard the one of the lower degree; ties are resolved arbitrarily. The resulting set is *I.*

Luby's Algorithm

Example:

If two neighbors are elected simultaneously, then the algorithms discard the node with lower degree.



Luby's Algorithm

In the choice step , value for random variables X_0 , …, X_{n-1} are chosen mutually independently, such that on that average a set of value for this random variable is **good**.

A vertex is **good** if

$$\sum_{u \in N(v)} \frac{1}{2d(u)} \geq \frac{1}{6} .$$

Define an *edge to be good* if at least one of its endpoint is **good.**

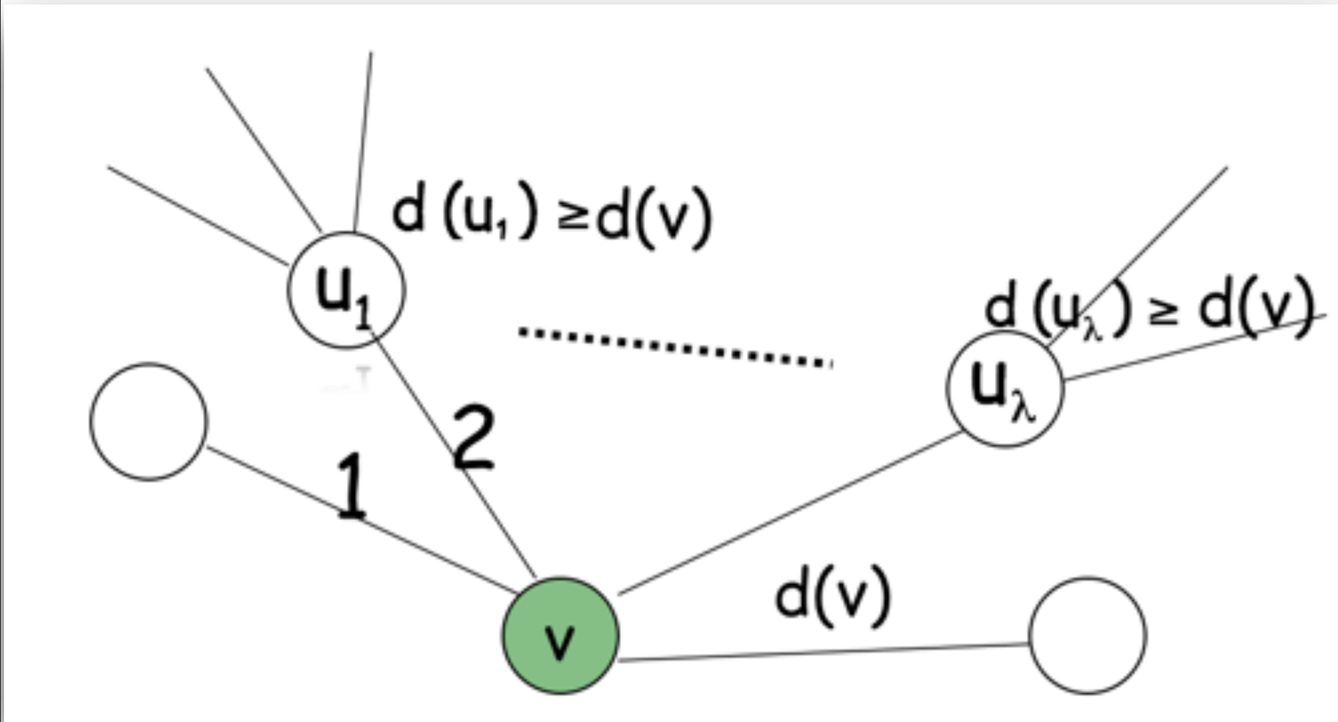
Luby's Algorithm

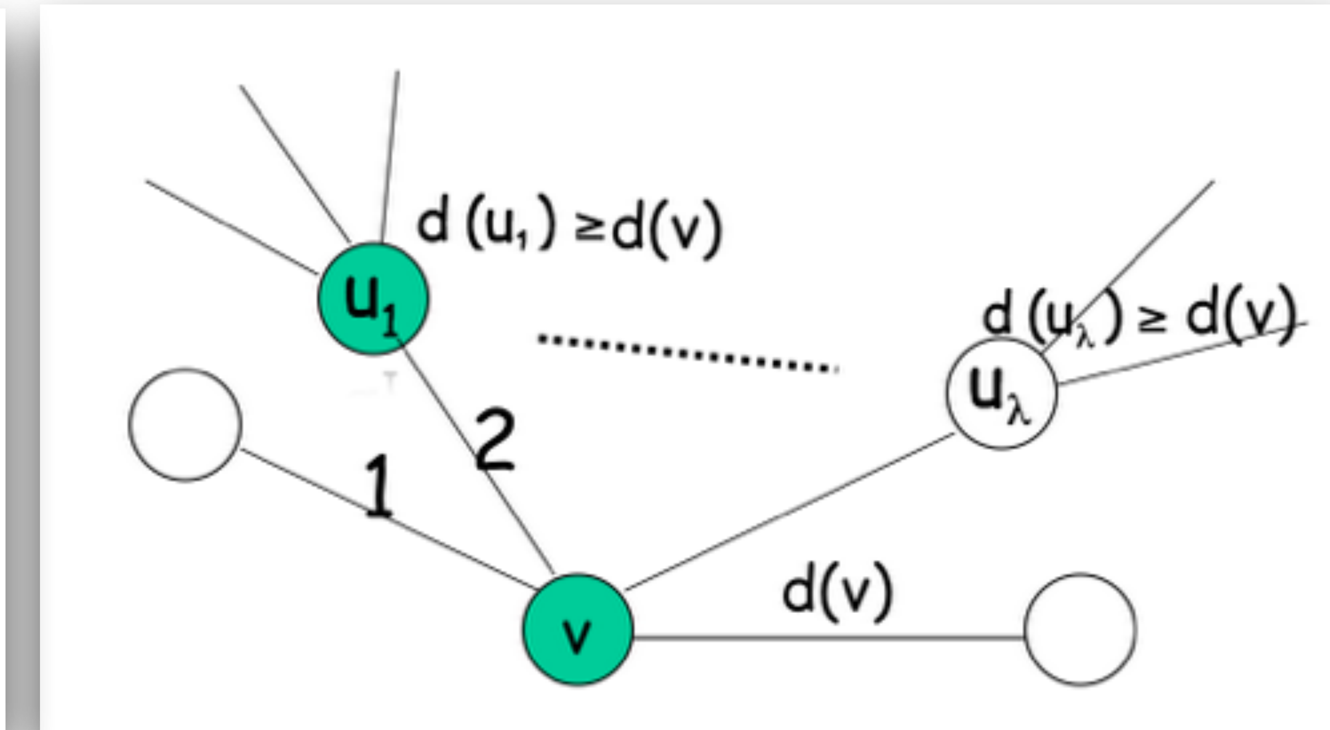
LEMMA 1: For all v Pr (v ∈ I) ≥ 1/4d(v).

Proof: Let L(v) = {u ∈ N(v)| d(u) ≥ d(v)}. If v ∈ *S* then v does not make it into *I* only if some element of L(v) is also in S. Then

$$\begin{aligned}
\Pr(v \notin I \mid v \in S) &\leq \Pr(\exists u \in L(v) \cap S \mid v \in S) \\
&\leq \sum_{u \in L(v)} \Pr(u \in S \mid v \in S) \\
&= \sum_{u \in L(v)} \Pr(u \in S) \quad \text{(by pairwise independence)} \\
&\leq \sum_{u \in L(v)} \frac{1}{2d(u)} \\
&\leq \sum_{u \in L(v)} \frac{1}{2d(v)} \quad (\text{since } d(u) \geq d(v)) \\
&\leq \frac{d(v)}{2d(v)} = \frac{1}{2}\,.
\end{aligned}$$

Luby's Algorithm





The probability that some neighbor **(λ)** of v with same or higher degree elects its self

And then

$$\begin{aligned} \Pr(v \in I) &= \Pr(v \in I \mid v \in S) \cdot \Pr(v \in S) \\ &\geq \frac{1}{2} \cdot \frac{1}{2d(v)} = \frac{1}{4d(v)} . \end{aligned}$$

□

Luby's Algorithm

LEMMA 2: If v is good then Pr (v ∈ N(I)) ≥ 1/36.

Proof: If v as a neighbor u of degree 2 or less , then

$$\begin{aligned} \Pr(v \in N(I)) &\geq \Pr(u \in I) \\ &\geq \frac{1}{4d(u)} \quad \text{by LEMMA 1} \\ &\geq \frac{1}{8}\,. \end{aligned}$$

Otherwise d(u) ≥ 3 for all u ∈ N(v). Then for all u ∈ N(v), 1/2d(u)≤ 1/6 and since v is good,

$$\sum_{u \in N(v)} \frac{1}{2d(u)} \geq \frac{1}{6}\,.$$

Luby's Algorithm

There must exist a subset M(v) ⊆ N(v) such that $\frac{1}{6} \leq \sum_{u \in M(v)} \frac{1}{2d(u)} \leq \frac{1}{3}$.

Then

$$\begin{aligned}
\Pr(v \in N(I)) &\geq \Pr(\exists u \in M(v) \cap I) \\
&\geq \sum_{u \in M(v)} \Pr(u \in I) - \sum_{\substack{u, w \in M(v) \\ u \neq w}} \Pr(u \in I \wedge w \in I) \\
&\geq \sum_{u \in M(v)} \frac{1}{4d(u)} - \sum_{\substack{u, w \in M(v) \\ u \neq w}} \Pr(u \in S \wedge w \in S) \\
&\geq \sum_{u \in M(v)} \frac{1}{4d(u)} - \sum_{\substack{u, w \in M(v) \\ u \neq w}} \Pr(u \in S) \cdot \Pr(w \in S) \\
&= \sum_{u \in M(v)} \frac{1}{4d(u)} - \sum_{u \in M(v)} \sum_{w \in M(v)} \frac{1}{2d(u)} \cdot \frac{1}{2d(w)} \\
&= \Big(\sum_{u \in M(v)} \frac{1}{2d(u)}\Big) \cdot \Big(\frac{1}{2} - \sum_{w \in M(v)} \frac{1}{2d(w)}\Big) \\
&\geq \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}
\end{aligned}$$

□

Luby’s Algorithm

The Luby’s MIS Distributed Algorithm runs :

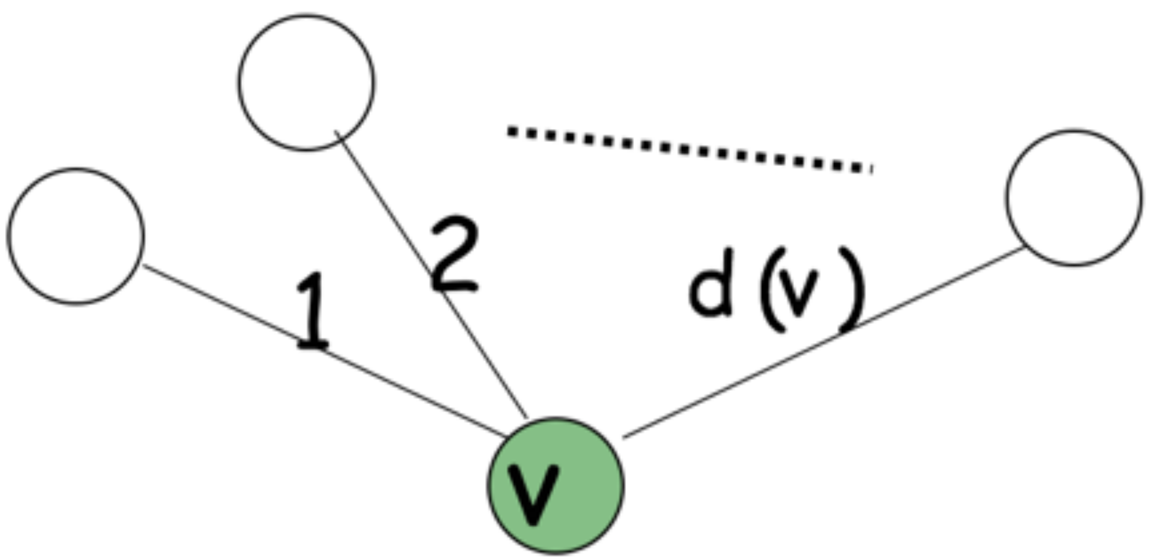
in time **O(log n)** in expected case

O(logd · logn) with high probability

Luby's Algorithm

At each phase *k* :

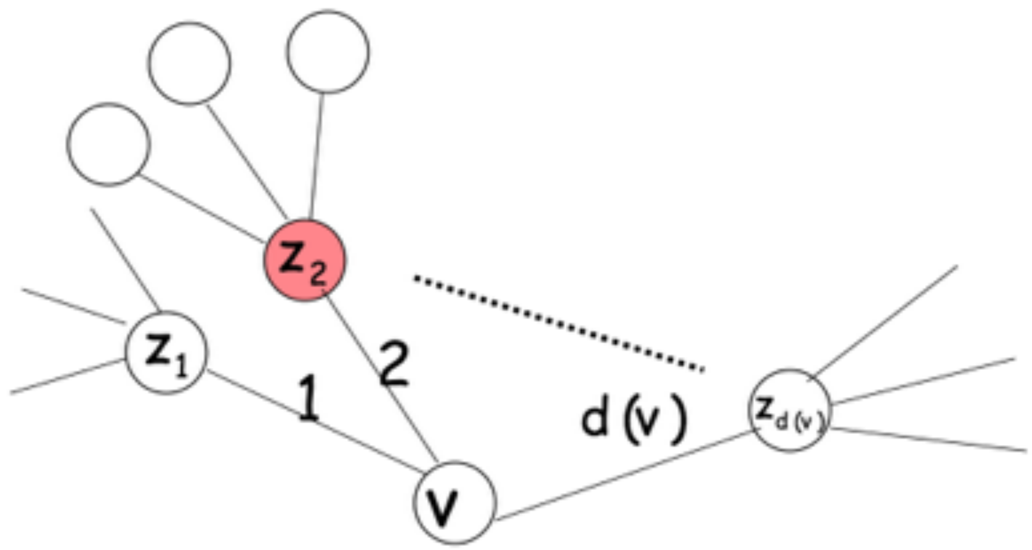
Each node $v \in G_k$ elects itself with probability $p(v) = 1/2d(v)$.



Elected node are candidates for the independent set I_k

Luby's Algorithm: Analysis

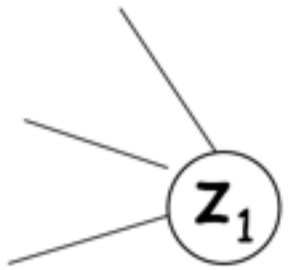
Consider the fase k :

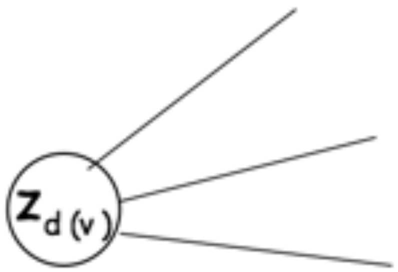


A good event for node v

H_v : at least one neighbor enters in I_k

If H_v is true , then $v \in I_k$ and v will disappear at the end of the current phase
At end of phase k





Luby's Algorithm: Analysis

LEMMA 3: At least one neighbor of V is elected with probability at least

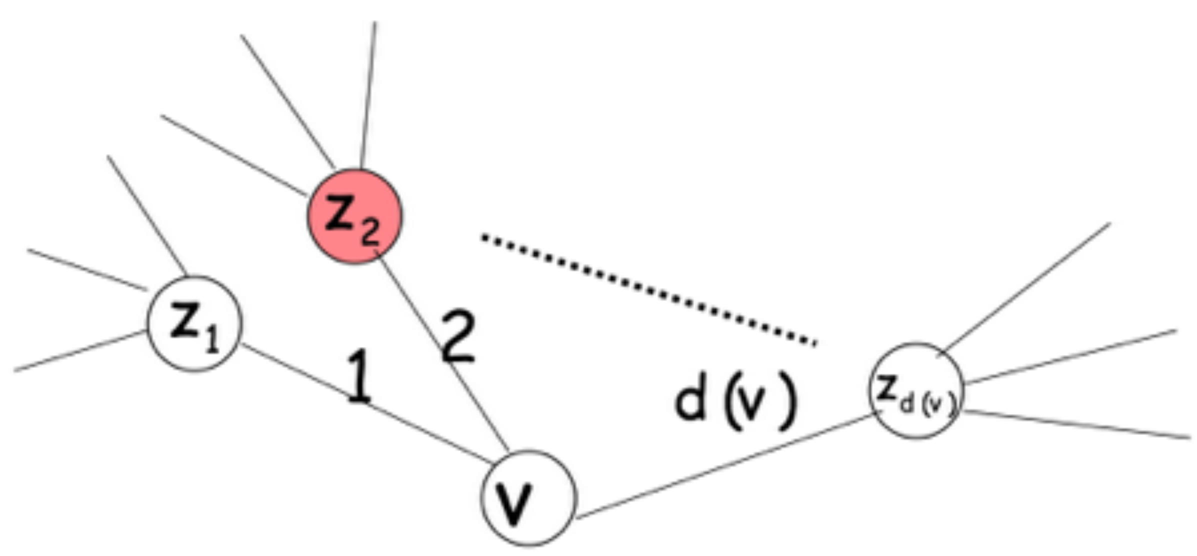
$$1 - e^{-\frac{d(v)}{2\tilde{d}(v)}}$$

$$\tilde{d}(v) = \max_{z \in N(v)} d(z)$$ maximum neighbor degree

LEMMA 4:

$$P[H_v] \geq \frac{1}{2}\left(1 - e^{-\frac{d(v)}{2\tilde{d}(v)}}\right)$$

H_v : at least one neighbor of v enters in I_k



Luby's Algorithm: Analysis

Let d_k be the maximum of degree in the graph G_k.

Suppose that in G_k: $d(v) \geq \frac{d_k}{2}$

Then, $\tilde{d}(v) \leq 2d(v)$

$$P[H_v] \geq \frac{1}{2}\left(1 - e^{-\frac{d(v)}{2\tilde{d}(v)}}\right) \geq \frac{1}{2}\left(1 - e^{-\frac{1}{4}}\right) = c$$

And in phase k a node with degree $d(v) \geq \frac{d_k}{2}$ disappears with

probability at least **c,** obviously the node with high degree will disappear fast

Luby's Algorithm: Analysis

Suppose that the degree of v remains at least d/2 for the next Φ phases.

Node v does not disappear within Φ phases with probability at most $(1\text{-}c)^{\Phi}$

Take $\Phi = 3 \log_{1-c} 1/n$

Node v does not disappear within Φ phases with probability at most

$$(1-c)^{\phi} = (1-c)^{3\log_{1-c}\frac{1}{n}} = \frac{1}{n^3}$$

And with Φ phases v either disappears or its degree gets lets that d/2 with probability at least 1- $\frac{1}{n^3}$

Luby's Algorithm: Analysis

For this by the end of 3 log $_{1-c}$1/n phases there is non node of degree higher than d/2 with probability at **least** 1- $\frac{1}{n^2}$

And in every Φ phases the maximum degree of the graph reduces by at least half , with probability at least 1- $\frac{1}{n^2}$

Luby's Algorithm: Analysis

Maximum number of phases until degree drops to 0 (MIS has formed)

$$\log d \cdot 3\log_{1-c}\frac{1}{n} = O(\log d \cdot \log n)$$

with probability at least

$$\left(1-\frac{1}{n^2}\right)^{\log d} \geq 1-\frac{1}{n}$$

References

- https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0ahUKEwiR-OSNxP7JAhUHaxQKHXfGALYQFggtMAI&url=http%3A%2F%2Fcourses.csail.mit.edu%2F6.852%2F08%2Fpapers%2FLuby.pdf&usg=AFQjCNFBo5Ko_Majfmgo1fLmwr5Io7K_rw&cad=rja

- http://www.di.univaq.it/~proietti/slide_algdist2008/MIS.ppt

- http://link.springer.com/chapter/10.1007%2F978-1-4612-4400-4_36#page-1