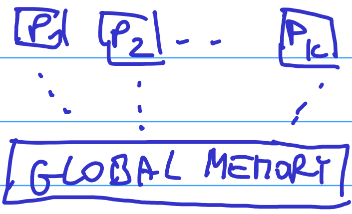
LECTURE 10 3/12/2020

MAXIMAL INDEPENDENT SET ON PRAM

Task: given a graph $G=(V,E)$, find an independent set that cannot be extended

Model: Parallel RAM

processors $P_1, \dots, P_k$
each local memory
global memory



synchronized computation:
each step: global read
local computation
global write (no conflicts)

RANDOMIZED ALGORITHM

PARAL - MAXIMAL IS

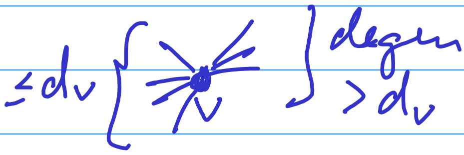
1. $I := \emptyset$; $V' := V$; $G' := G$
2. while $(V' \neq \emptyset)$ do
3. in parallel for $\forall v \in V'$: if $d_v = 0$, then $I := I \cup \{v\}$
   $V' := V' \setminus \{v\}$
4. in parallel for $\forall v \in V'$ mark v with prob $\frac{1}{2d_v}$
   (independ. for different vertices & iterations)
5. in parallel for $\forall \{u,v\} \in E(G')$: if both u,v marked,
   then unmark the lower degree vertex of u,v
   (break ties arbitrarily) .. e.g. unmark the the lower
   let S be the set of marked vertices and
   $N(S)$ be their neighbours
6. $I := I \cup S$; $V' := V' \setminus (S \cup N(S))$; $G' := G[V']$
   induced by V'
7. OUTPUT I

NOTATION

Let $G_j = (V_j, E_j)$ --- the graph G' after iteration j.

Correctness: OK ✓

For a graph G_j: $v \in V_j$ is **bad** if



$$|\{u \in N(v) \mid d_u > d_v\}| > \frac{2}{3} d_v$$

"likely to get deleted"

$v \in V_j$ is **good** if $|\{u \in N(v) \mid d_u \leq d_v\}| \geq \frac{d_v}{3}$

an edge

$\{u,v\} \in E$ is bad if both u, v are bad

$\{u,v\} \in E$ is good if at least one of u, v is good

Lemma 1: At least half of the edges of G_j are good, for each G_j.

Proof: orient the edges in G_j as follows:

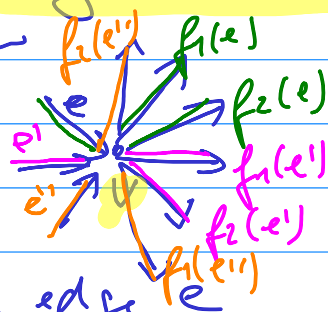
an $\{u,v\}$ is directed to the higher degree vertex; break ties as in the algorithm

Consider a bad vertex v

by definition of bad vertices:

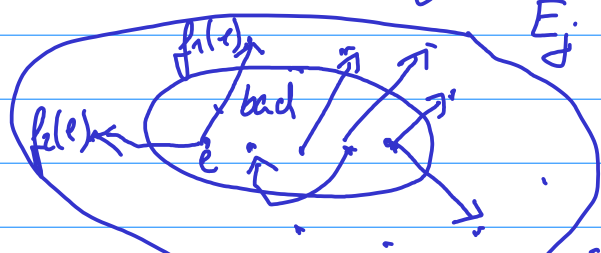
$$d^{out}(v) \geq 2\, d^{in}(v)$$

it is possible to assign to each bad edge e incoming to v two outgoing edges $f_1(e), f_2(e)$ in such a way that no outgoing edge is assigned to more than one incoming edge



$\Rightarrow$ $2 \cdot \#\text{bad edges} \leq |E_j|$

$\Rightarrow$ $\#\text{good edges} \geq \frac{|E_j|}{2}$

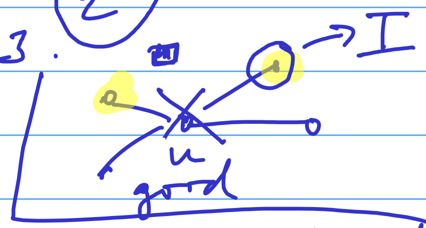


Lemma 2: A good vertex v with $d_v > 0$ has marked neighbour with probability $> (1 - e^{-1/6}) =: \alpha$

Lemma 3: If a vertex is marked, then it remains marked after step 5 with probability $\ge \frac{1}{2}$.

Lemma 4: The probability that a good vertex gets deleted, is at least $\ge \frac{1}{2}\alpha$.

Proof of Lemma 4: by Lemmas 2 and 3.



Proof of Lemma 2: v ... good

$L(v) = \{ w \in N(v) \mid d(w) \le d(v) \}$ $\qquad d_v = d(v)$

by def. of good vertices

$$|L(v)| \ge \frac{d(v)}{3}$$

each vertex $w \in L(v)$ is marked with prob.

$$\frac{1}{2d(w)} \ge \frac{1}{2d(v)}$$

as the marking was done independently

$\Rightarrow$ the probability that no $w \in L(v)$ is marked

is $\le \underbrace{\prod_{w \in L(v)} \left(1 - \frac{1}{2d(v)}\right)} = \left(1 - \frac{1}{2d(v)}\right)^{\frac{d(v)}{3}} \underset{1-x \le e^{-x}}{\le} e^{-\frac{1}{6}}$

$\Rightarrow$ the probability that a (low degree) neighbour is marked is $\ge 1 - e^{-1/6}$ ☒

Proof of Lemma 3:

consider a fixed marked vertex v $\qquad v \text{ --- } w$

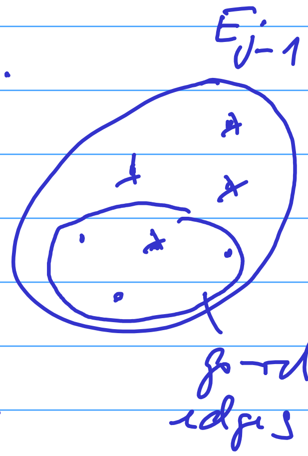
v can get unmarked only if $\exists\, w \in N(v)$ s.t. $d(w) \ge d(v)$ & w is marked

w is marked with probability $\frac{1}{2d(w)}$

$$\Rightarrow \Pr[v \text{ gets unmarked}] \le \sum_{\substack{w \in N(v):\\ d(w) \ge d(v)}} \frac{1}{2d(w)} \le \frac{d(v)}{2d(v)} = \frac{1}{2}$$ ∎

Main Lemma: If E_{j-1} is the set of edges at the beginning of iteration j, then

$$\mathbb{E}[|E_j|] \le \left(1 - \frac{\alpha}{4}\right)|E_{j-1}| .$$



Proof:

$$\mathbb{E}[|E_j|] = \sum_{e \in E_{j-1}} \left(1 - \Pr[e \text{ gets deleted}]\right)$$

by Lemma 4 $$\le |E_{j-1}| - \frac{1}{2}\alpha \cdot \#\text{good edges in } E_{j-1}$$

by Lemma 1 $$\le |E_{j-1}| - \frac{1}{4}\alpha |E_{j-1}| = \left(1 - \frac{\alpha}{4}\right)|E_{j-1}|$$ ∎

Corollary: $$\mathbb{E}[|E_j|] \le \left(1 - \frac{\alpha}{4}\right)\mathbb{E}[|E_{j-1}|] .$$

Proof: $$\mathbb{E}[|E_j|] = \sum_{\ell=1}^{|E|} \mathbb{E}\left[|E_j| \,\middle|\, |E_{j-1}| = \ell\right] \cdot \Pr\left[|E_{j-1}| = \ell\right]$$

by Main Lemma $$\le \left(1 - \frac{\alpha}{4}\right) \underbrace{\sum_{\ell=1}^{|E|} \ell \cdot \Pr\left[|E_{j-1}| = \ell\right]}_{\mathbb{E}[|E_{j-1}|]}$$ ∎

$$\Rightarrow \mathbb{E}[|E_j|] \le |E|\left(1 - \frac{\alpha}{4}\right)^j \underset{1-x \le e^{-x}}{\le} |E| \cdot e^{-\frac{\alpha \cdot j}{4}} =$$

$$= |E| \cdot e^{-2\ln|E|} = \frac{1}{|E|} < 1$$

for $j = \frac{8 \cdot \ln|E|}{\alpha}$.

Markov inequality: $$\Pr[X \ge a] \le \frac{\mathbb{E}[X]}{a}$$

$\underbrace{\text{for } j = \frac{8 \ln |E|}{\alpha}}$ if $|E| \geq 4$

Thus, by Markov: $\Pr[|E_j| \geq 1] \leq \frac{1}{|E|} \stackrel{\text{if } |E|\geq 4}{\leq} \frac{1}{4}$

$\left(a = 1,\ \mathbb{E}[X] = \frac{1}{|E|}\right)$

Theorem: With probability $\geq \frac{3}{4}$, the algorithm finds a maximal independent set in

$\rightarrow O(\log |E|)$ iterations with polynomial number of processors.