BFS: Proof of shortest path thm

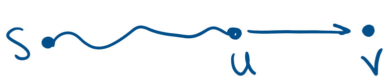
Thm: $\forall u \in V$: $u.d = \delta(s,u)$ ← length of shortest path from s to u

Lemma 1: u enqueued before v $\Rightarrow$ $u.d \le v.d$

Lemma 2: $\forall u \in V$: $u.d \ge \delta(s,u)$

Proof: (Smallest counterexample) Assume otherwise.

- $\exists v \quad v.d \ne \delta(s,v)$
  By Lemma 2, $v.d > \delta(s,v)$
  Take such a v that minimizes $\delta(s,v)$
- Then, consider preceding node

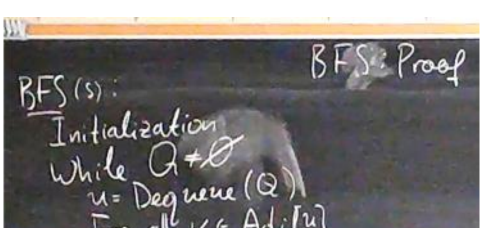


u: preceding node before v on a shortest path from s to v

- Observation:

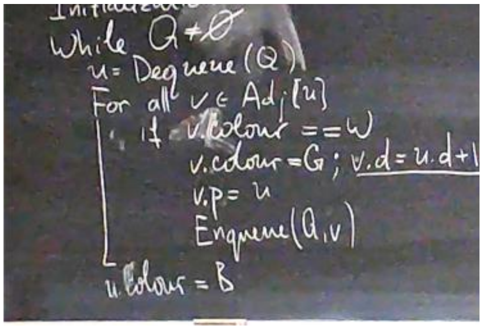
$$\left.\begin{aligned} &\delta(s,u) = u.d \quad \text{(by hypothesis)} \\ &u.d = \delta(s,u) = \delta(s,v) - 1 \\ &\qquad\qquad\qquad = v.d - 1 \\ &\quad \text{(by shortest path)} \end{aligned}\right\} \Leftrightarrow \quad v.d > u.d + 1$$

Suppose when u is explored, v was:



a) White $\Rightarrow$ $v.d = u.d + 1$

b) Black $\Rightarrow$ v enqueued before u
$\Rightarrow$ $v.d \le u.d$ (Lemma 1)



b) Black $\Rightarrow$ v enqueued before u

$\Rightarrow$ $v.d \le u.d$ (Lemma 1)

c) Grey $\Rightarrow$ w discovered v

$\Rightarrow$ w enqueued before u

$\Rightarrow$ $w.d \le u.d$

$\Rightarrow$ $v.d = w.d + 1 \le u.d + 1$

Depth-First Search

DFS(G):
- For all $u \in V$: $u.color = W$, $u.d = Null$
- $time = 0$ ← Global
- For all $u \in V$:
  - If $u.color == W$:
    - DFS-Visit(G, u)

DFS-Visit(G, u):
- $time = time + 1$
- $u.d = time$
- $u.color = G$
- For all $v \in Adj[u]$:
  - if $v.color = W$:
    - $v.p = u$
    - DFS-Visit(G, v)
- $u.color = B$
- $time = time + 1$
- $u.f = time$

Running time $O(|V| + |E|)$

Init: $O(|V|)$

DFS-Visit: called $|V|$ times (at most one time per node)

Loop in DFS-Visit(G, v) has $|Adj[u]|$ iterations

total $\sum_{u \in V} |Adj[u]| \le 2|E|$ (or at most $|E|$ if graph directed)

Properties of DFS:

(Parenthesis Property) For any u and v exactly one holds:

- $[u.d, u.f]$ and $[v.d, v.f]$ disjoint and neither u nor v is a descendant of the other.
- $u.d < v.d < v.f < u.f$ and v is a descendant of u
- $v.d < u.d < u.f < v.f$ and u is a descendant of v

Observation: v descendant of u $\Leftrightarrow$ u was grey when v was discovered

WLOG assume $u.d < v.d$:

Case 1: $u.d < v.d < u.f$ $\Rightarrow$ u was G when v discovered

$\Rightarrow$ v descendant of u

by LIFO order $\Rightarrow$ $v.f < u.f$

Case 2: $u.d < u.f < v.d < v.f$ $\Rightarrow$ u was B when v discovered

$\Rightarrow$ v was W when u discovered

$\Rightarrow$ neither is descendent of the other.

(*) v is a descendant of u $\Leftrightarrow$ $u.d < v.d < v.f < u.f$

(**) $(u,v) \in E$ $\Rightarrow$ $v.d < u.f$

Either $v.d < u.d < v.f$ or $u.d < v.d$ and v descendant of u

$\Rightarrow u.d < v.d < v.f < u.f$

$\square$

(White Path theorem) v descendant of u $\Leftrightarrow$ at time $u.d$ there is a path of white vertices from u to v

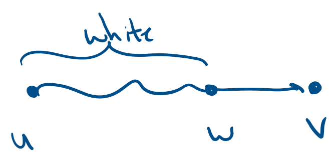
Proof: ($\Rightarrow$) Let $u, v_1, \ldots, v_k = v$ in the DFS tree path from u to v $\Rightarrow$ $u.d < v_1.d < \ldots < v_k.d$

This is a white path.

($\Leftarrow$) Assume otherwise, that

$\exists v$ such that at time $u.d$ $\exists$ white path from u to v but v is not descendant of u.

Let v be the first such vertex on a given white path



w is descendant of u $\Rightarrow$ $u.d < w.d < w.f < u.f$

$u.d < v.d$ and by (**) $v.d < u.f$.

$$u.d < v.d < v.f < u.f$$

Only possible position

then v is descendant of u. ∎

Application of White Path theorem:

Test if a directed graph $G = (V, E)$ has a directed cycle.

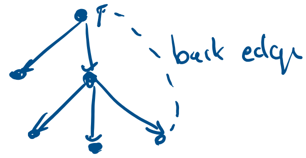
Run DFS(G)

Test if $\exists$a back edge: $\begin{cases} \text{back edge} \Rightarrow \text{cycle} \\ \text{no back edge} \Rightarrow \text{acyclic} \end{cases}$

For edge (u,v) back edge

$\Leftrightarrow$ $v.d < u.d < u.f < v.f$ (u desc. of v)

Proof: (Back edge $\Rightarrow$ Cycle)



Easy.

(Cycle $\Rightarrow$ Back edge)

Assume there is a cycle

$$C = v_1 \to v_2 \to \dots \to v_k$$

renumber this cycle such that v_1 is discovered first

$(v_1.d < v_2.d \dots v_k.d)$

All other nodes are white when v_1 discovered

$\Rightarrow$ white path from $v_1 \dots v_k$.

$\Rightarrow$ By WPT, v_k descendant of v_1

the edge (v_k, v_1) is a back edge.

$\boxtimes$

Next tutorial: if graph acyclic, then sorting by finish time will yield a forward graph: