EE360C ALGORITHMS - FALL 2018 - October 11 - 11-12:30

Dijkstra's Algorithm

1. INITIALIZE (G, s)

> for each vertex v:
> $v.d := \infty$
> $v.\pi := nil$
> $s.d := 0$

2. Q := all vertices
3. while Q ≠ empty:
4. $\quad v := \underbrace{\text{EXTRACT\_MIN}(Q)}_{\text{find min } v.d \text{ and remove } v \text{ from } Q}$
5. $\quad$ for each edge (v,u)
6. $\quad\quad$ RELAX (v,u)

> if $u.d > v.d + w(v,u)$:
> $u.d := v.d + w(v,u)$
> $u.\pi = v$

Linked list

1. $\Theta(n)$
2. $\Theta(n)$
3. for every node v
4. $\quad \Theta(n)$
5. $\quad$ for every edge outgoing from v (m_v)
6. $\quad\quad O(1)$

Right now we've proved it is $O(n^2+m)$, but it can be proven that it actually is $\Theta(n^2+m)$.

Heap

1. $O(n)$
2. $O(n)$
3. for every node v
4. $\quad O(\log n)$
5. $\quad$ for every edge m_v outgoing from v
6. $\quad\quad O(\log n)$

? $\boxed{\text{TOTAL}}$

$$O(n) + \underbrace{\sum_{\text{nodes } v} (O(n) + m_v \cdot O(1))}$$

$$\underbrace{\sum_{\text{nodes } v} O(n) + \sum_{\text{nodes } v} m_v \cdot O(1)}$$

$$O(n^2) + O(m)$$

$\boxed{\text{Overall: } \Theta(n^2+m)}$

? $\boxed{\text{TOTAL}}$

$$O(n) + \sum_{\text{nodes } v} (O(\log n) + m_v \cdot O(\log n))$$

$$O(n) + O(n\log n) + O(m \log n)$$

$\boxed{\text{Overall: } O((n+m)\log n)}$

1 Note: Usually $m > n$, in which case we have $O(m \log n)$.

		SPARSE	DENSE
Linked List:	$O(n^2 + m)$	$O(n^2)$	$O(n^2)$
Heap:	$O(m \log n)$	$O(n \log n)$	$O(n^2 \log n)$

Two Cases

1) Sparse Graph: $m \approx n$, heap is better!
2) Dense Graph: $m \approx n^2$. linked list is better!

Claim All nodes removed from Q (in step 4) have $v.d = \delta(s,v)$

↑ actual shortest path distance

Proof Induction on K^{th} node removed from Q.

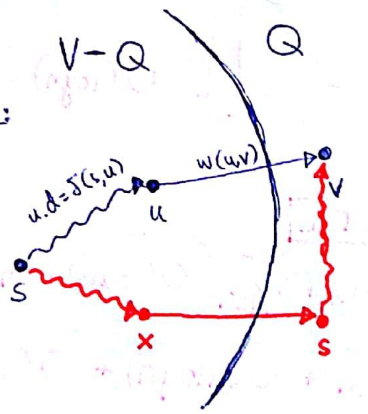
<u>Base Case:</u> $K = 1$

$v = s$ and $s.d = 0 = \delta(s,s)$.

<u>Induction Hypothesis:</u> First K nodes removed from Q have $v.d = \delta(s,v)$.

Want to show that $K+1$ st node removed from Q has $v.d = \delta(s,v)$.

<u>By ind. hyp.:</u>



Let u be the node s.t. v.d was last changed when relaxing on (u,v).
Then:

$v.\pi = u$

$v.d = u.d + w(u,v)$

$u \notin Q$ since we only relax on edges from a node once that node is removed from Q.

By ind. hyp. $u.d = \delta(s,u)$ so $v.d = \delta(s,u) + w(u,v)$.

cont. on next page

Towards a contradiction, suppose ∃ path (in red) from s to v that is shorter than v.d.

When x was removed from Q, we relaxed on (x,y), so

$$y.d \le \underbrace{x.d}_{=\,\delta(s,x) \text{ by ind. hyp.}} + w(x,y)$$

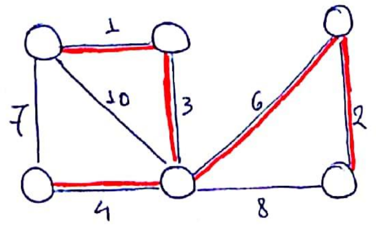
Since we chose v as the next node to extract from Q and not y, it must be $v.d \le y.d$

$$\underbrace{\delta(s,u) + w(u,v)}_{\text{our path}} = v.d \le y.d \le \underbrace{\delta(s,x) + w(x,y)}_{\text{part of the other (red) path from } s \text{ to } y}$$

So red path is not shorter. □

MINIMUM SPANNING TREES - GREEDY ALGORITHMS

Context: Undirected, connected, weighted graph



Definition

Spanning Tree is an acyclic subset of edges that connects all the nodes.

Definition

Minimum Spanning Tree: a spanning tree of minimum total weight.