9/12/14

Minimum Spanning Tree

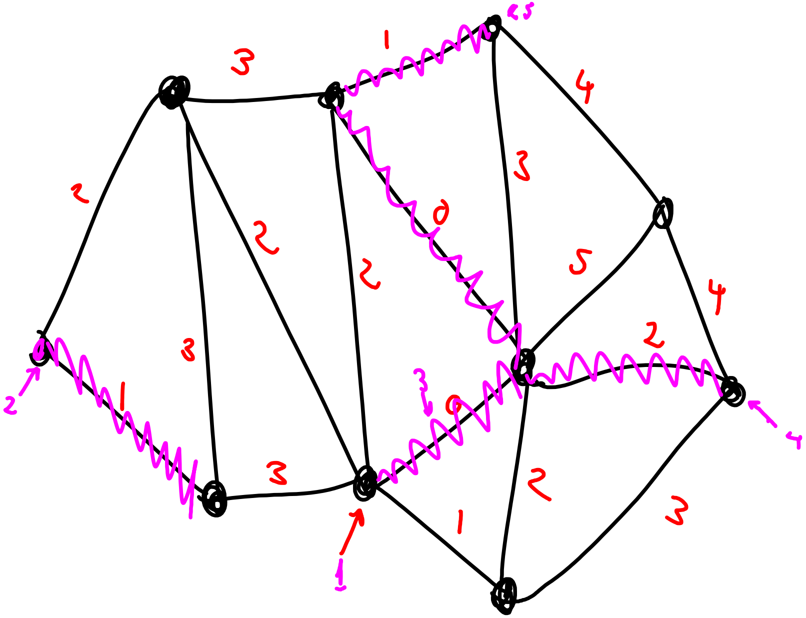
Given a connected graph G with edge weights

Problem: find spanning tree of minimum total weight

More general algorithm: Having chosen

$F_t = \{e_1, e_2, \ldots, e_t\}$ = Collection of disjoint subtrees $T_1, T_2, \ldots, T_k$

Choose any T_i and add the shortest edge from T_i to $\overline{T_i}$.



3
1
2.5
4
2
2
3
2
0
3
5
4
2
3
1
3
0
2
4
2
1
3
1

Claim: at all times $\exists$ a minimum length tree that contains all of the edges chosen so far. $\Rightarrow$ correctness of algorithm.

<u>Proof</u> by induction on number of edges k, chosen.

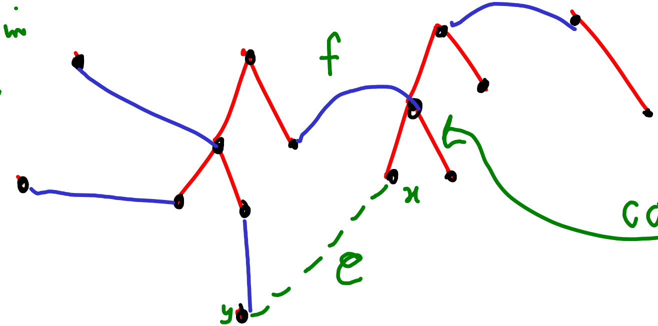
$k=0$: trivial because any minimum tree contains the empty set $\emptyset$ edges

Inductive step

Edges chosen so far, in red.

T = Red + Blue edges :

If added edge is in T, nothing more to do.



$\ell(e) \leq \ell(f)$

So, $T' = T + e - f$ is also shortest.

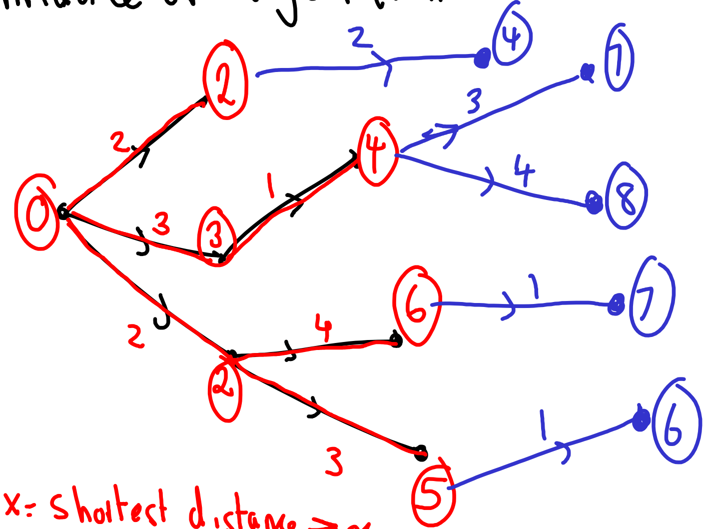
Suppose we added the green edge $e = (x, y)$: min. length edge leaving

Shortest Paths:

Case 1: $\ell(e) \geq 0$

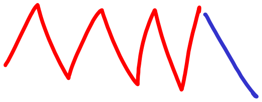
Dijkstr algorithm: find a shortest path from s to every other vertex.

- In middle of algorithm:

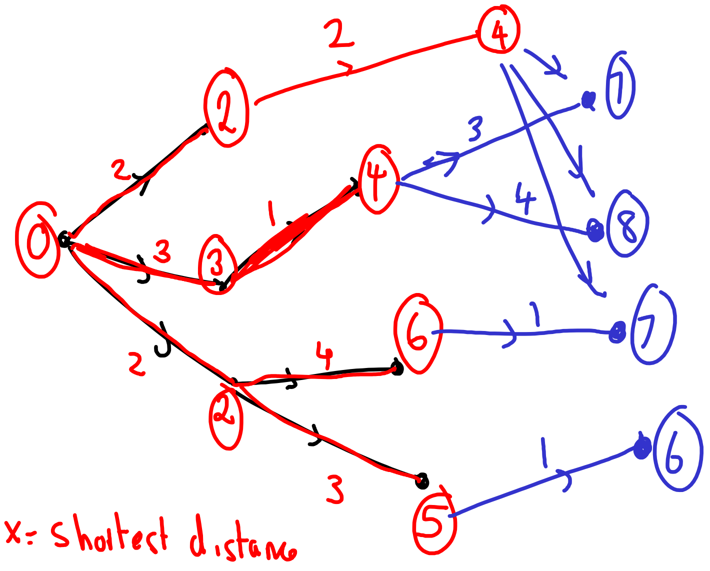


(x) : x = shortest distance → x

(x) : shortest path of form

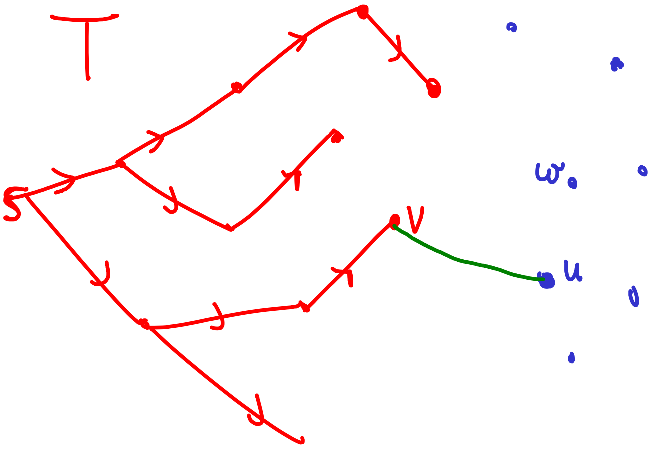


choose minimum (x) and add to tree



Then
we
update
blue
values

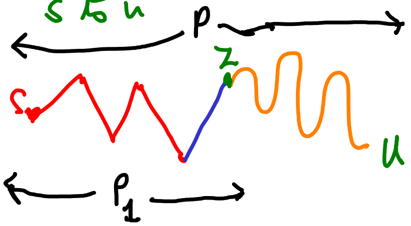
Correctness of algorithm: induction on size of tree



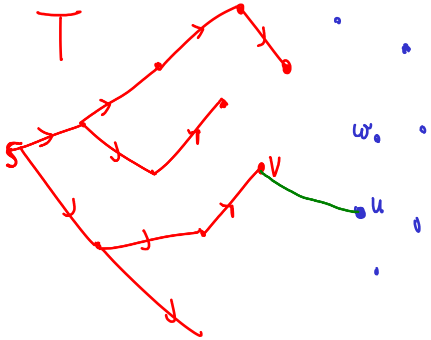
We have $d(w)$

Let $d(u) := \min_{w \notin T} [d(w)]$

Take any path from s to u



We know that $s \to v$ path is shortest.

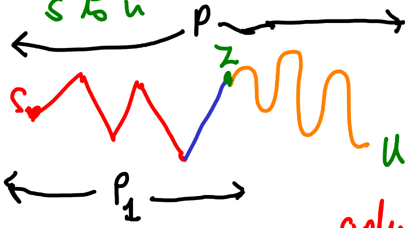


I have to show that

$\ell(P) \geq d(u)$.

Let $d(u) := \min_{w \notin T} [d(w)]$

Take any path from s to u



① $\ell(P) \geq \ell(P_1)$ ← only place I use $\ell(e) \geq 0$.

② $\ell(P_1) \geq d(z) \geq d(u)$

Conclude: if we have a function $\ell(P)$ s.t.

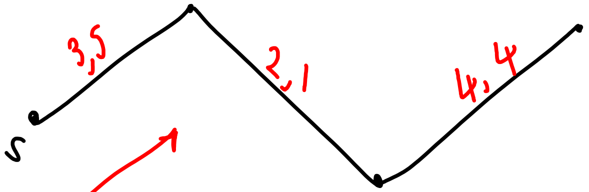
always

then Dijkstra algorithm works.

Example: time dependent edge length. Suppose $\ell(e) = a_e + b_e t$

where t = 'time' we reach x.

$e = (x,y)$, $\quad a_e \geq 0$, $\quad b_e \geq 0$



(a, b)

$$\ell(P) = 3 + 5 + 36 = 44$$