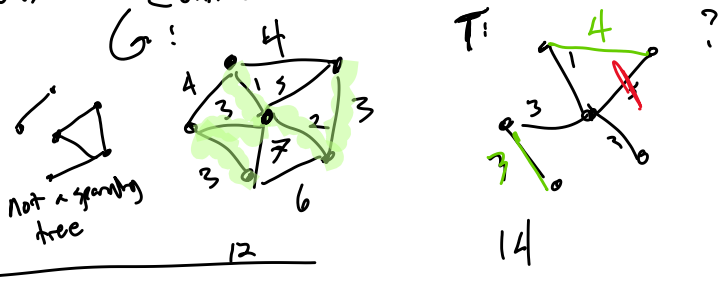
Minimum Spanning Trees

Thursday, November 12, 2020 1:48 PM

Minimum Spanning Tree for Undirected graph

- sum of edge weights
- all vertices connected
- unique path between any (u, v)



MST applications:

- installing a network connecting all rooms minimizing total cost of wires.
- image registration and mosaic alignment

Algorithms to find MST for given graph.

- 3 algs, all variations on a template.

Basic template:

Init: let T := an empty graph. (T has all vertices V but no edges)

E := all edges in G

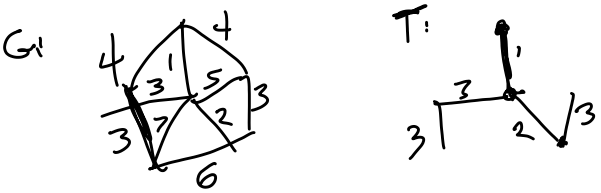
Disjoint forest

Loop: while E is not empty:

- pop some e = (u, v) from E

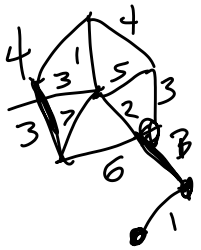
if e is "helpful" add E to T

- Prim's
  - T initially have no vertices.
  - Pop the smallest edge in E, connected to some node in T, add V\T.

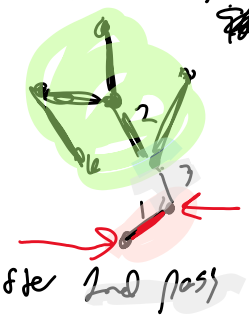


- Boruvka's

- T starts as distinct forest
- For each connected component S of T, add the smallest edge adjacent to some node in S, and T\S



T after 1 pass...

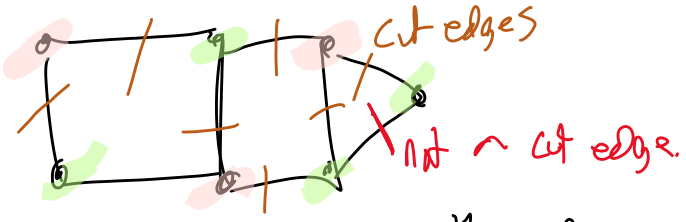


T is not yet a spanning tree. not yet fully connected.

- Kruskal's alg:
  - T is initially disjoint forest
  - Pick e as smallest edge that does not create a cycle.

How to prove this is correct?

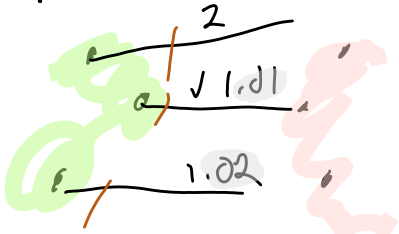
- "Safe edges"
  - Cut: a cut of graph G, is a partition of vertices, S ⊆ G, and V\S



- Cut edges: cut edges for S are the edges (u, v) where u∈S, v∉S.

- Safe edges: e=(u,v) is safe if

it's the unique minimum cut edge for some cut.



$P(V)$ cuts,

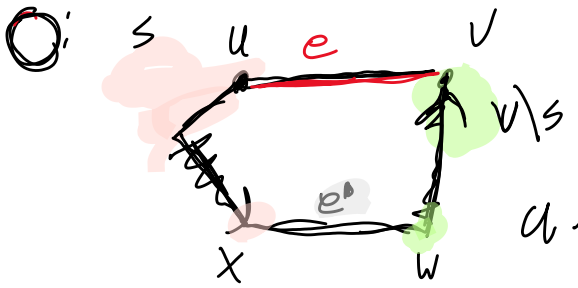
- If edges have unique weights... every cut includes some safe edge

- Math lemma: if $e=(u,v)$ is safe,
  then it's a member of *every* MST of G.

Suppose $e=(u,v)$ is a safe edge in G,
and let O be some MST for G.

e safe means there's some cut S for which
e is the min edge with $u \in S$, $v \in V \setminus S$.

O spanning tree means $u \rightsquigarrow v$ is a unique
path from u to v in O.



$u \rightsquigarrow x \to w \rightsquigarrow v$

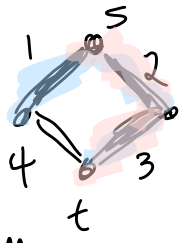
must be some *cut edge* e' on the path $u \rightsquigarrow v$.

But, safe edge e is unique min. of the set of
cut edges including e': $w(e) < w(e')$

- Substitute e for e' in O, resulting O':
  This still a spanning tree:
  look at subtree containing u when e' is removed
  and containing v when e' is removed.

mean u, v, $u \rightsquigarrow v$ say P. is the unique path in T

- If edge weights in G are unique,
then there is a unique MST.



not true for SP

Consider some MST T. For every edge $(u,v) \in T$, consider subtree of v and subtree excluding v

	data structure	running time	$E = O(V)$	$E = O(V^2)$
Boruvka	just connected components,	~~$\log V$~~ passes, each $V \log V$ $(E+V)\log V$	$V \log V$	$V^2 \log n$
Prim	same as dijkstra's	$E + V \log V$	$V \log V$	n^2
Kruskal	union find data structures	$(E+V)\log V$	$V \log V$	$n^2 \log n$

For each algorithm
we add a safe edge each time we add one.