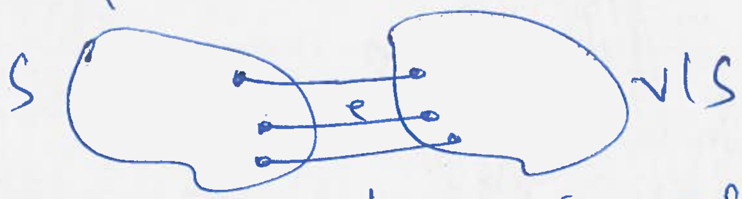
Oct 30

Cut Property lemma

Assume all c_e's are distinct. ← Assume this

Then for all cuts $(S, V \setminus S)$ s.t. $S \neq \phi$, $S \neq V$, $S \subseteq V$



If e is the cheapest crossing edge $\Rightarrow$ e is in ALL MSTs.

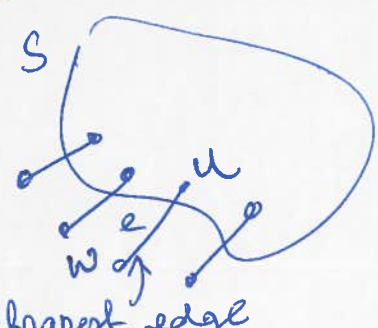
(Assume true for now)

THM 1: Prim's algo is correct.

Pf idea: Consider the point when Prim's is about to add an edge e to T.

(Goal: e is the ~~chea~~ cheapest crossing edge for some cut $(S, V \setminus S)$.)

Idea: Apply Cut Property Lemma, the S in the lemma is exactly ~~S~~ the same as S in algo.



By defn c_e is the cheapest crossing edge.

Claim 1: $S \neq \phi$ (as $s \in S$)

Claim 2: $S \neq V$ (as $w \notin S$)

$\Rightarrow$ adding e is "safe" as e is in ALL MSTs.

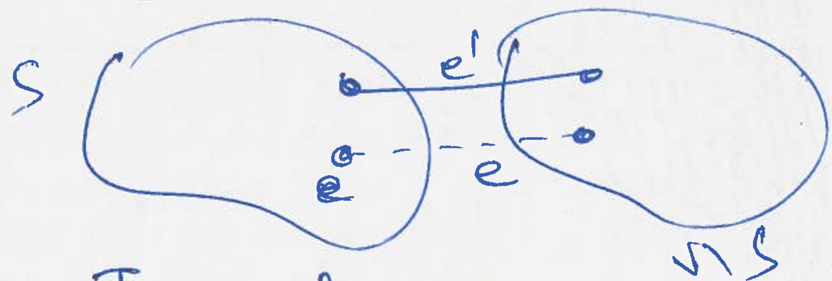
$\rightarrow$ Need to argue that the final T is a spanning tree.

Claim 3: At the end of each iteration, (S, T) is connected

($\Rightarrow$ Prims is correct when $S = V$). ↑ (Pf: Ex)

Pf (idea) of Cut Property Lemma:

For contradiction, assume $\exists$ an $\emptyset \neq S \neq V$ & $\exists$ an MST T s.t. the cheapest crossing edge for $(S, V \setminus S)$ is not in T.



Goal: ~~Construct~~ Construct a spanning tree T' s.t. $c(T') < c(T)$

As T connects all V, $\exists$ a crossing edge $e' \neq e$ that is in T

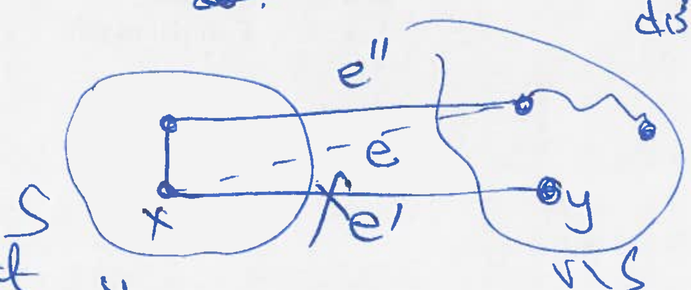
$\Rightarrow$ Consider $T' = (T \setminus \{e'\}) \cup \{e\}$

$$c(T') = c(T) - c_{e'} + c_e < c(T)$$

[as $c_{e'} > c_e$ as e is the cheapest crossing edge + edge costs are distinct.]

Q: Done?

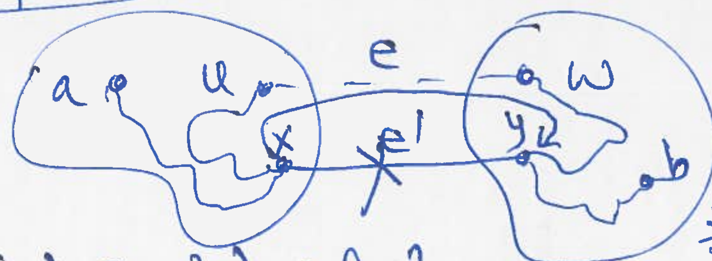
A: NO!



Dropping e' would disconnect y.

(fixed) Pf idea:

$e = (u, w) \notin T$



Since T is a spanning tree, $\exists$ u-w path without e

$\Rightarrow \exists$ a crossing edge $e' = (x, y)$ on this path

$\rightarrow T' = (T \setminus \{e'\}) \cup \{e\}$

$c(T') < c(T)$ as before

Claim: T' is connected.

Case 1: a-b path doesn't use e'.

Case 2: a-b path uses e'.

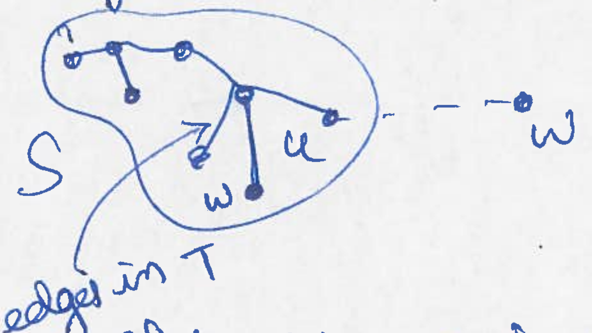
$\Rightarrow$ contradiction to fact that T is an MST ∎.

THM 2: Kruskal's algo is correct

(Recall: Consider all edges in increasing order of costs & add it to T if it doesn't introduce a cycle.)

Pf (idea): Consider the time algo is about to add e to T.
$= (u, w)$

Q: What is S?
A: Let S be set of vertices connected to u using ONLY edges in T.



Claim 1: $S \neq \emptyset$ (as $u \in S$)

Claim 2: $S \neq V$

Pf idea: $w \notin S$
(as otherwise adding e would create a cycle).

Claim 3: e is the cheapest crossing edge for $(S, V \setminus S)$
($\Leftarrow$ e is the first crossing edge considered by Kruskal's)

Claims $1+2+3 \Rightarrow$ THM.