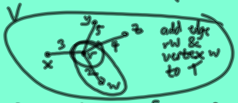
Prim's Algorithm

Finds a MST & is greedy

Start vertex r (arbitrary)



Grow MST out from r

Initially, r is only vx & no edges in tree.

$T.V = \{r\}$

$T.E = \emptyset$

while $T.V \neq G.V$

find a min wt edge with one endpoint in $T.V$ and the other in $G.V - T.V$.

call this edge (u,v) where $u \in T.V$ & $v \in G.V - T.V$

$T.V := T.V \cup \{v\}$

$T.E := T.E \cup \{(u,v)\}$.

Kruskal vs. Prim

Kruskal	Prim
T is an acyclic graph at any time (not necessarily connected)	T is always connected, but does not include all the vertices until the end.

Prim is an example of graph search.

Implementation

Use a min heap of vertices H

Initially: each vx u has key (numerical) and π (vertex)

$r.key := 0$

$u.key := \infty$ for all $u \neq r$

$u.\pi := nil$ for all u

H initially contains all vertices

while $H \neq \emptyset$

$u :=$ ExtractMin(H) // $u = r$ on 1st iteration

u is not in T.V

for each v adjacent to u

if $v \notin H$ // $v \in T.V$

go on to the next v

// $v \in H$

let w be the weight of the edge (u,v)

if $w < v.key$ then

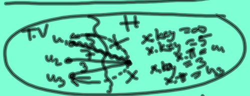
$v.key := w$ // decrease key

$v.\pi := u$

end foreach

add u to $T.V$

add $(u, u.\pi)$ to $T.E$, provided $u.\pi \neq nil$



Min key in H corresponds to the lightest edge between $T.V$ and H.

Extracting it executes Prim's algo.

Running times of Prim & Kruskal

Kruskal:

sort edges: time $|E| \lg |E|$

[note: $|V|-1 \leq |E| \leq |V|^2$

$|V|-1 \leq |E|$: G is connected; $|E| \leq |V|^2$: simple graph

so $\lg |E| = \Theta(\lg |V|)$

$\therefore$ sorting time $\Theta(|E| \lg |V|)$]

Disjoint set system of vertices:

$|V|$ many MakeSet ops

for each edge, 2 Finds & ≤ 1 merge (Union)

$\Theta(|E|)$ ops — total time is $\Theta(|E| \alpha(|V|))$

$= o(|E| \lg |V|)$

$\therefore$ time of Kruskal is dominated by the initial sort

$\therefore$ total time is $\Theta(|E| \lg |V|)$

Nov 17-12:46 PM

Running time for Prim:

H — insert $|V|$ many vertices into H

time $\Theta(H)$

(H is a binary heap)

$|V|$ many extract mins

(each takes $\lg|V|$)

total time is $\Theta(|V|\lg|V|)$

$|E|$ many decrease key ops (worst case)

time $\Theta(|E|\lg|V|)$

(if H is binary heap)

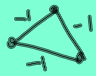
bottleneck time

If H is a Fib heap, get better run time:

DecreaseKey is $O(1)$ amortized time

$\therefore$ total decrease key time is $\Theta(|E|)$

$\therefore$ total over all is $\Theta(|E| + |V|\lg|V|)$

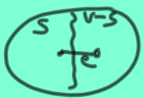


Correctness

Def: Let $G = (V,E)$ be a graph (undir.)

A cut of G is a partition of V into two disjoint sets S & $V-S$ for some $S \subseteq V$

An edge e crosses cut $(S, V-S)$ if one endpt of e is in S & the other is in $V-S$

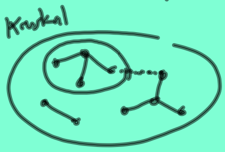


Let A be a set of edges. Say that A respects cut $(S, V-S)$ if no edge in A crosses the cut.

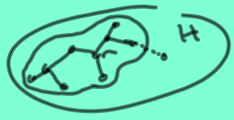
Lemma: Suppose A is a set of edges that respects some cut $(S, V-S)$, and e is some minimum wt edge crossing this cut (among all edges crossing the same cut). If A is a subset of some MST, then $A \cup \{e\}$ is also a subset of some MST.

Assume lemma (proof later)

Kruskal



Prim



Nov 17-1:22 PM