Minimum Spanning Trees

Thursday, April 16, 2020 3:25 PM

Algorithms for MST:

- Kruskal's
  - Process edges \( e \in E \) in order by cost.
  - Add if new edge does not form a cycle in \( T \).
  - Start with \( T \) as all dummy edges (disjoint set)

- Prim's
  - Add the smallest edge adjacent to some node in \( T \).
  - \( T \) starts as empty graph.

- Boruvka's
  - For each component \( S \) in \( T \),
    add the smallest edge adjacent to some node in \( S \).
- Trees as disjoint forests.

Cuts: a partition into two of $V$ 
such that
$$S \subseteq V$$ 
and
$$V \setminus S = \emptyset$$

Cut edges:
$$(u, v)$$

Safe edges:

$e$ is safe if $e$ is the smallest
edge of some cut

Cut property: If $e$ is safe then it
is in every MST for $G$.

Proof: Assume $e$ is safe.

Suppose $T$ is an MST.
and for contradiction, $e \notin T$.

$T$: 

$e$ is safe reen. If some $S$, $$V \setminus S \text{ cut,}$$ 
s.t. $e$ is min cut edge

$T$ is a tree, so => unique path from
$u \to w$ in $T$. 

https://onedrive.live.com/redir?resid=D2ECBD89F9DF77E%21110&par...
Let \( e' = (v, w') \) be the first edge on this path, \( v \sim w \).

- \( e' \) is safe; \( e \prec e' \).
- Look \( T' = T \cup \{e'\} \).

- \( T' \) is still a spanning tree.
- For \( u, x \in G \), path \( u \rightarrow x \in T' \).
- \( u \rightarrow w' \rightarrow x \in T' \) is a spanning tree.

Reducing non-unique to unique.

Some edges could neither be safe nor unsafe.

- \( e \) safe \(\implies\) every MST has \( e \).
- \( e \) unsafe \(\implies\) no MST has \( e \).

Claim: if edge weights are unique, every edge is either safe or unsafe.

Consider: unique edge weights \(\implies\) exactly one MST.

Ex. 1 2 3 3 4

1.001 2.002 3.003 3.004 4.005...

Running Time:

Borůvka's alg:

- In each iteration:
  - merge each CC with one neighbor,
  - \( \Rightarrow \) \# of CC's cut in half each time.
$\log_2 n \leq \log m$ in each step.

- Find all edges $O(mn)$
- Sort all edges $O(n \log m) \log n$

Prim's: just like Dijkstra's

$\mathcal{O}$(sort) $O(m + n \log n)$

Kruskal's: union-find $O((n + m) \log m)$

<table>
<thead>
<tr>
<th>Prim</th>
<th>Dijkstra</th>
<th>Kruskal</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n \log n$</td>
<td>$n \log n$</td>
<td>$n \log n$</td>
</tr>
<tr>
<td>$m \log n$</td>
<td>$m \log n$</td>
<td>$m \log n$</td>
</tr>
<tr>
<td>$(mn) \log n$</td>
<td>$(mn) \log n$</td>
<td>$(mn) \log n$</td>
</tr>
</tbody>
</table>

$m = O(n)$

$m = O(n^2)$

- $n^2$
- $n \log n$
- $n \log n$