Minimum Spanning Tree (MST)

Hengfeng Wei

hfwei@nju.edu.cn

June 12, 2019

Cut Property

$$G = (V, E, w)$$

Cut Property (I)

$$\begin{aligned} X &: \text{ A part of some MST } T \text{ of } G \\ (S, V \setminus S) &: \text{ A } \textit{cut} \text{ such that } X \text{ does } \textit{not} \text{ cross } (S, V \setminus S) \\ e &: \text{ } A \text{ lightest edge across } (S, V \setminus S) \end{aligned}$$

Cut Property (I)

X : A part of some MST T of G

$(S, V \setminus S)$: A *cut* such that X does *not* cross $(S, V \setminus S)$

e : *A* lightest edge across $(S, V \setminus S)$

Then $X \cup \{e\}$ is a part of some MST T' of G.

Cut Property (I)

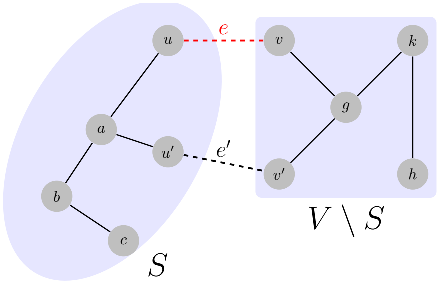
$$
\begin{aligned}
X &: \text{ A part of some MST } T \text{ of } G \\
(S, V \setminus S) &: \text{ A } \textit{cut} \text{ such that } X \text{ does } \textit{not} \text{ cross } (S, V \setminus S) \\
e &: \textit{ A} \text{ lightest edge across } (S, V \setminus S)
\end{aligned}
$$

Then $X \cup \{e\}$ is a part of some MST T' of G.

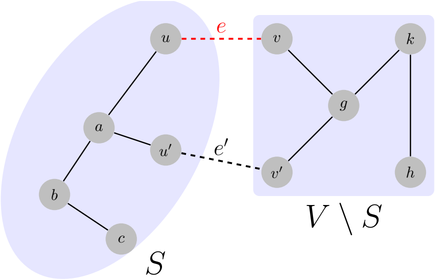
Correctness of Prim's and Kruskal's algorithms.

By Exchange Argument.

By Exchange Argument.

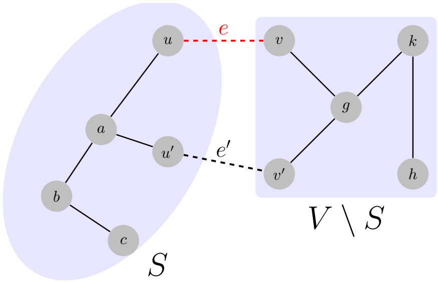


By Exchange Argument.



$$T' = \underbrace{T + \{e\}}_{\text{if } e \notin T} - \{e'\}$$

By Exchange Argument.



$$T' = \underbrace{T + \{e\}}_{\text{if } e \notin T} - \{e'\}$$

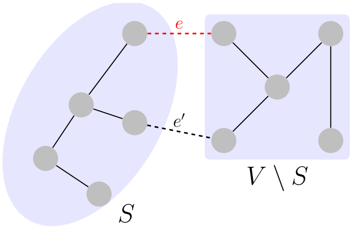
"a" → "the" $\Longrightarrow$ "some" → "all"

Cut Property (II)

A cut $(S, V \setminus S)$

Let $e = (u, v)$ be *a* lightest edge across $(S, V \setminus S)$

$\exists$ MST T of $G : e \in T$

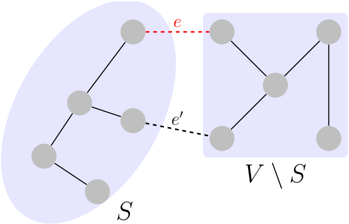


Cut Property (II)

A cut $(S, V \setminus S)$

Let $e = (u, v)$ be *a* lightest edge across $(S, V \setminus S)$

$\exists$ MST T of $G : e \in T$



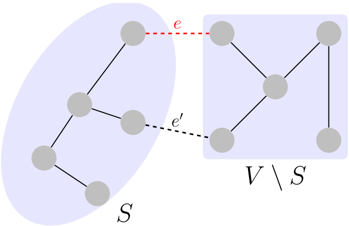
$$T' = \underbrace{T + \{e\}}_{\text{if } e \notin T} - \{e'\}$$

Cut Property (II)

A cut $(S, V \setminus S)$

Let $e = (u, v)$ be *a* lightest edge across $(S, V \setminus S)$

$\exists$ MST T of $G : e \in T$



$$T' = \underbrace{T + \{e\}}_{\text{if } e \notin T} - \{e'\}$$

"a" $\rightarrow$ "the" $\Longrightarrow$ "$\exists$" $\rightarrow$ "$\forall$"

Application of Cut Property [Problem: 10.15 (3)]

$$e = (u, v) \in G \text{ is a lightest edge } \implies e \in \exists \text{ MST of } G$$

Application of Cut Property [Problem: 10.15 (4)]

$$e = (u, v) \in G \text{ is the unique lightest edge } \implies e \in \forall \text{ MST}$$

Application of Cut Property [Problem: 10.15 (3)]

$$e = (u, v) \in G \text{ is a lightest edge} \implies e \in \exists \text{ MST of } G$$

$$\Big(S = \{u\}, V \setminus S\Big)$$

Application of Cut Property [Problem: 10.15 (4)]

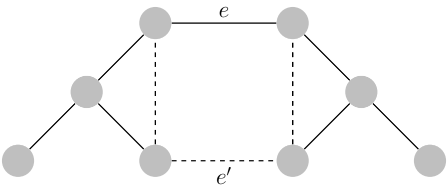
$$e = (u, v) \in G \text{ is the unique lightest edge} \implies e \in \forall \text{ MST}$$

Cycle Property

Cycle Property [Problem: 10.19(b)]

- Let C be any cycle in G
- Let $e = (u, v)$ be *a* maximum-weight edge in C

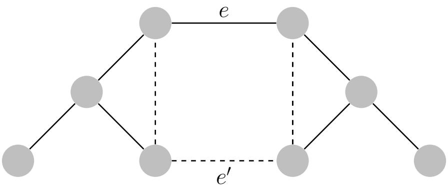
$$\text{Then } \exists \text{ MST } T \text{ of } G : e \notin T.$$



Cycle Property [Problem: 10.19(b)]

- Let C be any cycle in G
- Let $e = (u, v)$ be *a* maximum-weight edge in C

$$\text{Then } \exists \text{ MST } T \text{ of } G : e \notin T.$$

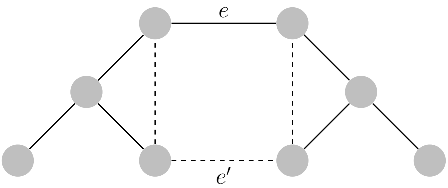


$$T' = \underbrace{T - \{e\}}_{\text{if } e \in T} + \{e'\}$$

Cycle Property [Problem: 10.19(b)]

- Let C be any cycle in G
- Let $e = (u, v)$ be *a* maximum-weight edge in C

Then $\exists$ MST T of $G : e \notin T$.



$$T' = \underbrace{T - \{e\}}_{\text{if } e \in T} + \{e'\}$$

"a" $\to$ "the" $\implies$ "$\exists$" $\to$ "$\forall$"

Anti-Kruskal algorithm [Problem: 10.19 (c)]

Reverse-delete algorithm (wiki; clickable)

Delete an edge if this does not disconnect the graph.

Anti-Kruskal algorithm [Problem: 10.19 (c)]

Reverse-delete algorithm (wiki; clickable)

Delete an edge if this does not disconnect the graph.

$$O\Big(m \log n \, (\log \log n)^3\Big)$$

Anti-Kruskal algorithm [Problem: 10.19 (c)]

Reverse-delete algorithm (wiki; clickable)

Delete an edge if this does not disconnect the graph.

$$O\Big(m \log n \; (\log \log n)^3\Big)$$

Proof.

Cycle Property

Anti-Kruskal algorithm [Problem: 10.19 (c)]

Reverse-delete algorithm (wiki; clickable)

Delete an edge if this does not disconnect the graph.

$$O\Big(m \log n \, (\log \log n)^3\Big)$$

Proof.

Cycle Property

$$T \subseteq F \implies \exists\, T' : T' \subseteq F - \{e\}$$

□

Anti-Kruskal algorithm [Problem: 10.19 (c)]

Reverse-delete algorithm (wiki; clickable)

Delete an edge if this does not disconnect the graph.

$$O\Big(m \log n \, (\log\log n)^3\Big)$$

Proof.

Cycle Property

$$T \subseteq F \implies \exists\, T' : T' \subseteq F - \{e\}$$

□

"On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem"

— Kruskal, 1956.

Application of Cycle Property [Problem: 10.15 (1)]

$$G = (V, E), \quad |E| > |V| - 1$$

e : the unique maximum-weighted edge of G

$$\overset{?}{\Longrightarrow}$$

$e \notin$ any MST

Application of Cycle Property [Problem: 10.15 (1)]

$$G = (V, E), \quad |E| > |V| - 1$$

e : the unique maximum-weighted edge of G

$$\overset{?}{\Longrightarrow}$$

$e \notin$ any MST

Bridge

Application of Cycle Property [Problem: 10.15 (2)]

$$C \subseteq G, \quad e \in C$$

$$e: \text{ the unique maximum-weighted edge of } G$$

$$\Longrightarrow$$

$$e \notin \text{ any MST}$$

Application of Cycle Property [Problem: 10.15 (2)]

$$C \subseteq G, \quad e \in C$$

$$e: \text{ the unique maximum-weighted edge of } G$$

$$\Longrightarrow$$

$$e \notin \text{ any MST}$$

Cycle Property

Application of Cycle Property [Problem: 10.15 (5)]

$$C \subseteq G, \quad e \in C$$

$$e: \text{ the unique lightest edge of } C$$

$$\overset{?}{\Longrightarrow}$$

$$e \in \forall \text{ MST}$$

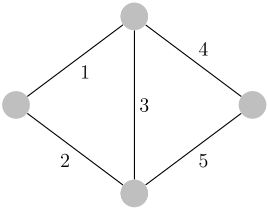
Application of Cycle Property [Problem: 10.15 (5)]

$$C \subseteq G, \quad e \in C$$

$$e: \text{ the unique lightest edge of } C$$

$$\overset{?}{\Longrightarrow}$$

$$e \in \forall \text{ MST}$$



Uniqueness of MST

Uniqueness of MST [Problem: 10.18 (1)]

$$\text{Distinct weights} \implies \text{Unique MST.}$$

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights $\implies$ Unique MST.

By Contradiction.

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights $\implies$ Unique MST.

By Contradiction.

$\exists$ MSTs $T_1 \neq T_2$

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights $\implies$ Unique MST.

By Contradiction.

$$\exists \text{ MSTs } T_1 \neq T_2$$

$$\Delta E = \Big\{ e \mid e \in T_1 \setminus T_2 \lor e \in T_2 \setminus T_1 \Big\}$$

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights $\implies$ Unique MST.

By Contradiction.

$$\exists \text{ MSTs } T_1 \neq T_2$$

$$\Delta E = \Big\{ e \mid e \in T_1 \setminus T_2 \lor e \in T_2 \setminus T_1 \Big\}$$

$$e = \min \Delta E$$

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights $\implies$ Unique MST.

By Contradiction.

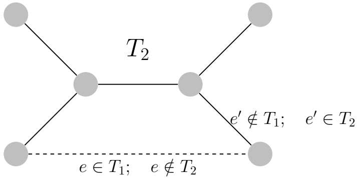
$$\exists \text{ MSTs } T_1 \neq T_2$$

$$\Delta E = \Big\{ e \mid e \in T_1 \setminus T_2 \lor e \in T_2 \setminus T_1 \Big\}$$

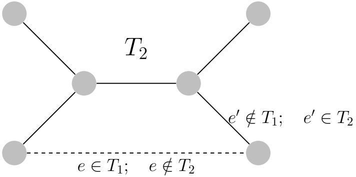
$$e = \min \Delta E$$

$$e \in T_1 \setminus T_2 \; (w.l.o.g)$$

$$e \in T_1 \setminus T_2$$

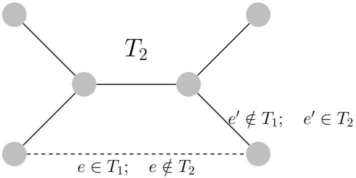


$$e \in T_1 \setminus T_2$$



$$T_2 + \{e\} \implies C$$

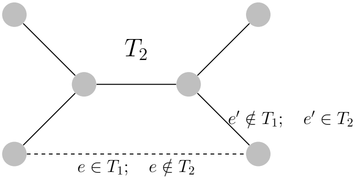
$$e \in T_1 \setminus T_2$$



$$T_2 + \{e\} \implies C$$

$\exists (e' \in C) \notin T_1$

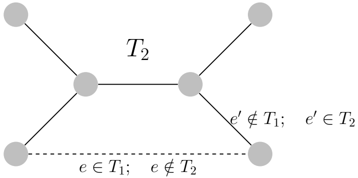
$$e \in T_1 \setminus T_2$$



$$T_2 + \{e\} \implies C$$

$$\exists (e' \in C) \notin T_1 \implies e' \in T_2 \setminus T_1 \implies e' \in \Delta E$$

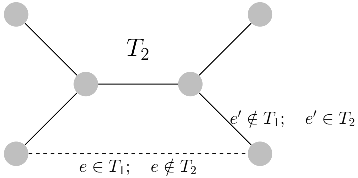
$$e \in T_1 \setminus T_2$$



$$T_2 + \{e\} \implies C$$

$$\exists (e' \in C) \notin T_1 \implies e' \in T_2 \setminus T_1 \implies e' \in \Delta E \implies w(e') > w(e)$$

$$e \in T_1 \setminus T_2$$



$$T_2 + \{e\} \implies C$$

$$\exists (e' \in C) \notin T_1 \implies e' \in T_2 \setminus T_1 \implies e' \in \Delta E \implies w(e') > w(e)$$

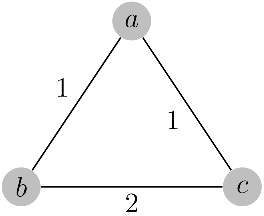
$$T' = T_2 + \{e\} - \{e'\} \implies w(T') < w(T_2)$$

Condition for Uniqueness of MST [Problem: 10.18 (2)]

$$\text{Unique MST} \not\Longrightarrow \text{Distinct weights.}$$

Condition for Uniqueness of MST [Problem: 10.18 (2)]

$$\text{Unique MST} \;\not\Longrightarrow\; \text{Distinct weights.}$$

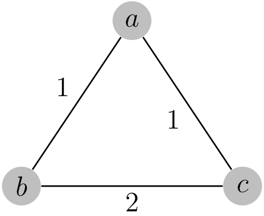


Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Minimum-weight edge across any cut is unique.

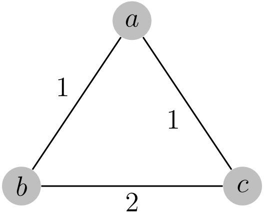
Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Minimum-weight edge across any cut is unique.



Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Longrightarrow$ Minimum-weight edge across any cut is unique.

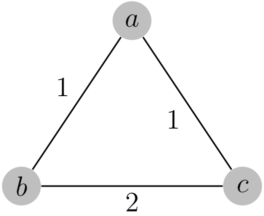


Theorem

Minimum-weight edge across any cut is unique $\Longrightarrow$ *Unique MST.*

Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Minimum-weight edge across any cut is unique.



Theorem

Minimum-weight edge across any cut is unique $\Longrightarrow$ *Unique MST.*

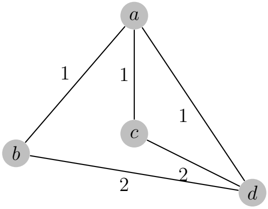
Construct T by adding all such edges.

Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Longrightarrow$ Maximum-weight edge in any cycle is unique.

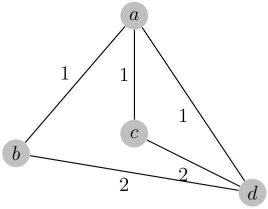
Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Maximum-weight edge in any cycle is unique.



Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Maximum-weight edge in any cycle is unique.

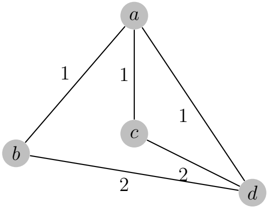


Theorem

Maximum-weight edge in any cycle is unique $\Longrightarrow$ *Unique MST.*

Unique MST [Problem: 10.18 (3)]

Unique MST $\not\Rightarrow$ Maximum-weight edge in any cycle is unique.



Theorem

Maximum-weight edge in any cycle is unique $\Longrightarrow$ *Unique MST.*

Construct T by deleting all such edges.

Unique MST [Problem: 10.18 (4)]

To decide whether a graph has a unique MST.

Unique MST [Problem: 10.18 (4)]

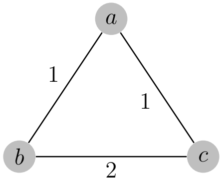
To decide whether a graph has a unique MST.

Ties in Prim's and Kruskal's algorithms

Unique MST [Problem: 10.18 (4)]

To decide whether a graph has a unique MST.

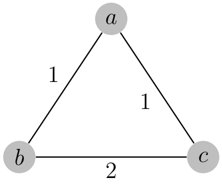
Ties in Prim's and Kruskal's algorithms



Unique MST [Problem: 10.18 (4)]

To decide whether a graph has a unique MST.

Ties in Prim's and Kruskal's algorithms

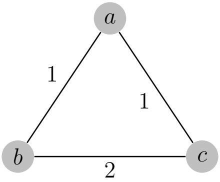


$$\underbrace{\color{red}{T}}_{\text{Any MST}} + \underbrace{\color{red}{\{e\}}, \forall e \notin T}_{\text{Cycle}}$$

Unique MST [Problem: 10.18 (4)]

To decide whether a graph has a unique MST.

Ties in Prim's and Kruskal's algorithms



$$\underbrace{T}_{\text{Any MST}} + \underbrace{\{e\}, \forall e \notin T}_{\text{Cycle}}$$

By Kruskal Algorithm.

Variants of MST

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

$$O\Big((m+n)\log n\Big) \quad \text{(recompute on } G'\text{)}$$

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

$$O\Big((m+n)\log n\Big) \quad \text{(recompute on } G'\text{)}$$

Theorem

There exists an MST of G' that includes no edges in $G \setminus T$.

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

$$O\Big((m+n)\log n\Big) \quad \text{(recompute on } G'\text{)}$$

Theorem

There exists an MST of G' that includes no edges in $G \setminus T$.

$$O(n \log n) \quad \text{(recompute on } G'' = (V + \{v\}, T + E_v)\text{)}$$

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

$$O\Big((m+n)\log n\Big) \quad \text{(recompute on } G'\text{)}$$

Theorem

There exists an MST of G' that includes no edges in $G \setminus T$.

$$O(n \log n) \quad \text{(recompute on } G'' = (V + \{v\}, T + E_v)\text{)}$$

$$O(n)$$

Adding a Vertex v to MST T [Problem: 10.7]

$$G' = (V', E') : V' = V + \{v\}, E' = E + E_v$$

To find an MST T' of G'.

$$O\Big((m + n) \log n\Big) \quad \text{(recompute on } G'\text{)}$$

Theorem

There exists an MST of G' that includes no edges in $G \setminus T$.

$$O(n \log n) \quad \text{(recompute on } G'' = (V + \{v\}, T + E_v)\text{)}$$

$$O(n)$$

"On Finding and Updating Spanning Tress and Shortest Paths", 1975

"Algorithms for Updating Minimum Spanning Trees", 1978

MST with Specified Leaves: [Problem: 10.11]

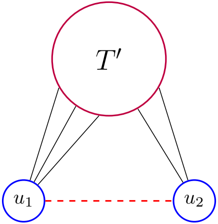
$$G = (V, E), \quad U \subset V$$

To find an MST with U as leaves.

MST with Specified Leaves: [Problem: 10.11]

$$G = (V, E), \quad U \subset V$$

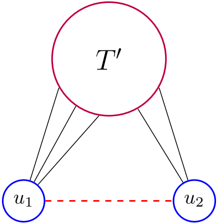
To find an MST with U as leaves.



MST with Specified Leaves: [Problem: 10.11]

$$G = (V, E), \quad U \subset V$$

To find an MST with U as leaves.

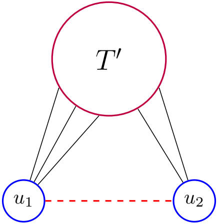


MST T' of $G' = G \setminus U$

MST with Specified Leaves: [Problem: 10.11]

$$G = (V, E), \quad U \subset V$$

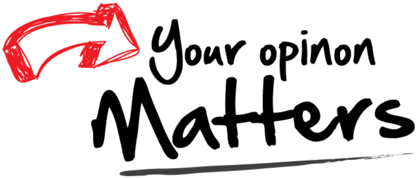
To find an MST with U as leaves.



MST T' of $G' = G \setminus U$

Attach $\forall u \in U$ to T' (with lightest edge)


THANK YOU



Office 302

Mailbox: H016

hfwei@nju.edu.cn