

Minimum Spanning Tree (MST)

Hengfeng Wei

hfwei@nju.edu.cn

June 19, 2018



Cut Property

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

Cut Property (I)

X : A part of some MST T of G

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

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

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)

X : A part of some MST T of G

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

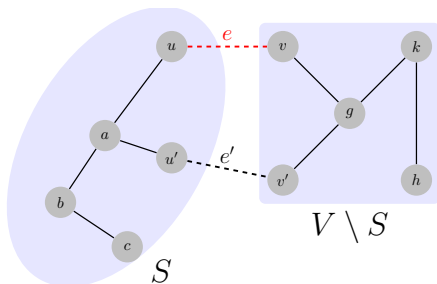
e : A lightest edge across $(S, V \setminus S)$

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

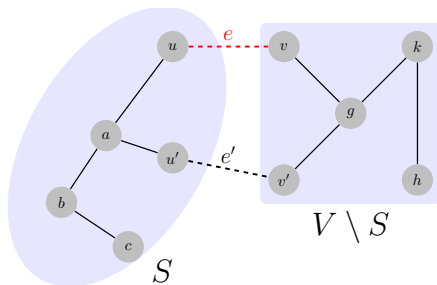
Correctness of Prim's and Kruskal's algorithms.

By Exchange Argument.

By Exchange Argument.

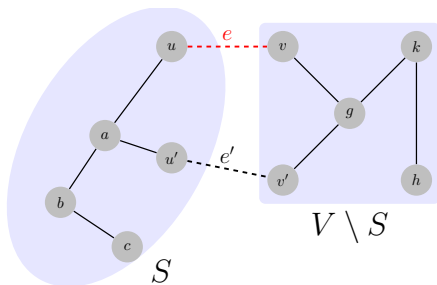


By Exchange Argument.



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

By Exchange Argument.



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

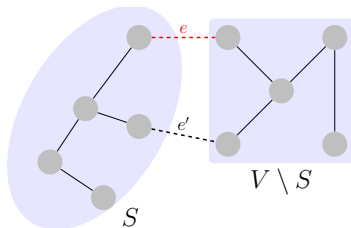
“a” \rightarrow “the” \Rightarrow “some” \rightarrow “all”

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 $G : e \in T$

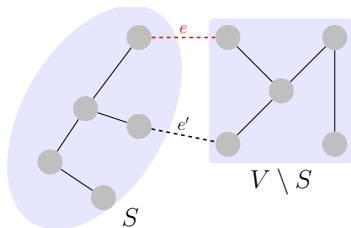


Cut Property (II)

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

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

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



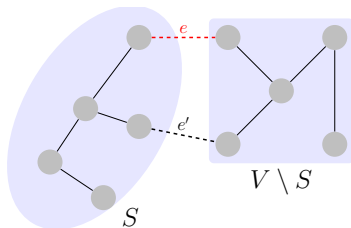
$$T' = T + \underbrace{\{e\}}_{\text{if } e \notin T} - \{e'\}$$

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 $G : e \in T$



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

“a” \rightarrow “the” \implies “ \exists ” \rightarrow “ \forall ”

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

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

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

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

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

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

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

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

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

Wrong Divide&Conquer Algorithm for MST [Problem: 10.21]

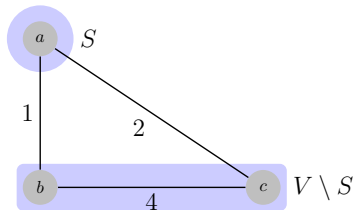
$$(V_1, V_2) : \left| |V_1| - |V_2| \right| \leq 1$$

$$T_1 + T_2 + \{e\} : e \text{ is a lightest edge across } (V_1, V_2)$$

Wrong Divide&Conquer Algorithm for MST [Problem: 10.21]

$$(V_1, V_2) : \left| |V_1| - |V_2| \right| \leq 1$$

$$T_1 + T_2 + \{e\} : e \text{ is a lightest edge across } (V_1, V_2)$$

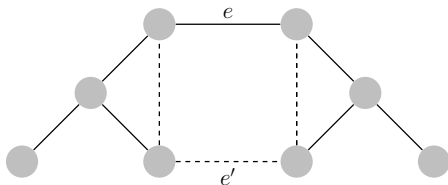


Cycle Property

Cycle Property [Problem: 10.19(b)]

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

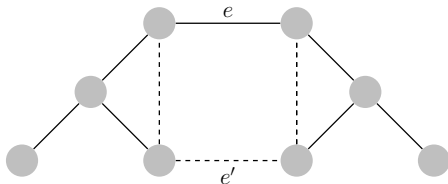
Then \exists MST T of $G : e \notin T$.



Cycle Property [Problem: 10.19(b)]

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

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

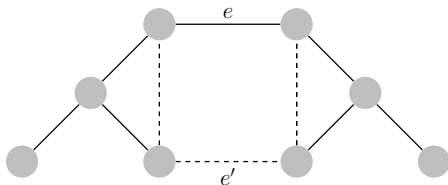


$$T' = T - \underbrace{\{e\}}_{\text{if } e \in T} + \{e'\}$$

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 $G : e \notin T$.



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

“a” \rightarrow “the” \Rightarrow “ \exists ” \rightarrow “ \forall ”

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

Reverse-delete algorithm ([wiki](#); [clickable](#))

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

Reverse-delete algorithm ([wiki](#); [clickable](#))

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

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

Reverse-delete algorithm ([wiki](#); [clickable](#))

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

Proof.

Cycle Property

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

Reverse-delete algorithm ([wiki](#); [clickable](#))

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

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](#))

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

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

$\implies ?$

$e \notin \text{any MST}$

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

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

e : the unique maximum-weighted edge of G

\Rightarrow ?

$e \notin$ any MST

Bridge

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

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

e : the unique maximum-weighted edge of C

\implies

$e \notin \text{any MST}$

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

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

e : the unique maximum-weighted edge of C

\implies

$e \notin \text{any MST}$

Cycle Property

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

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

e : the unique lightest edge of C

\Rightarrow ?

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

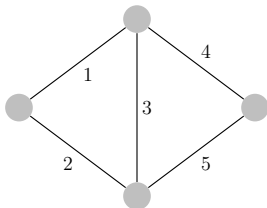
Application of Cycle Property [Problem: 10.15 (5)]

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

e : the unique lightest edge of C

\Rightarrow ?

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



Uniqueness of MST

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights \implies 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 MSTs $T_1 \neq T_2$

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

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights \implies Unique MST.

By Contradiction.

\exists MSTs $T_1 \neq T_2$

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

$$e = \min \Delta E$$

Uniqueness of MST [Problem: 10.18 (1)]

Distinct weights \implies Unique MST.

By Contradiction.

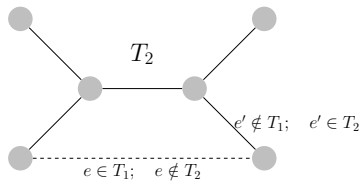
\exists MSTs $T_1 \neq T_2$

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

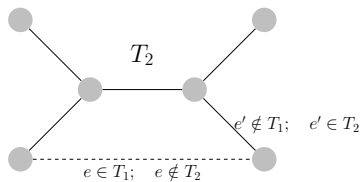
$$e = \min \Delta E$$

$$e \in T_1 \setminus T_2 \text{ (w.l.o.g.)}$$

$$e \in T_1 \setminus T_2$$

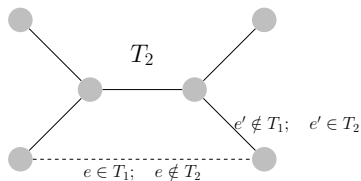


$$e \in T_1 \setminus T_2$$



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

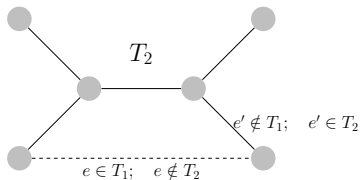
$$e \in T_1 \setminus T_2$$



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

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

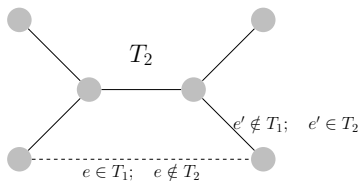
$$e \in T_1 \setminus T_2$$



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

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

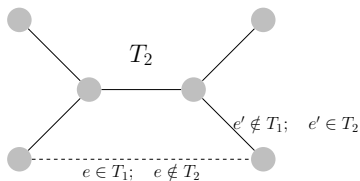
$$e \in T_1 \setminus T_2$$



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

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

$$e \in T_1 \setminus T_2$$



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

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

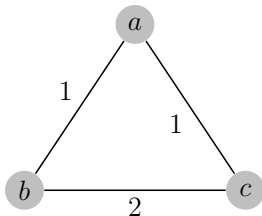
$$T' = T_2 + \{e\} - \{e'\} \implies w(T') < w(T_2)$$

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

Unique MST \nRightarrow Equal weights.

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

Unique MST $\not\Rightarrow$ Equal weights.

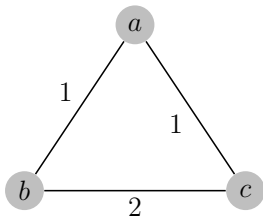


Unique MST [Problem: 10.21 (3)]

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

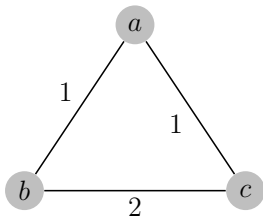
Unique MST [Problem: 10.21 (3)]

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



Unique MST [Problem: 10.21 (3)]

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



Theorem (After-class Exercise)

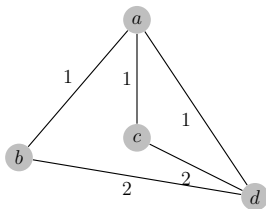
Minimum-weight edge across any cut is unique \Rightarrow Unique MST.

Unique MST [Problem: 10.21 (3)]

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

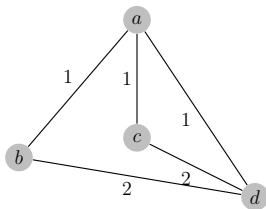
Unique MST [Problem: 10.21 (3)]

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



Unique MST [Problem: 10.21 (3)]

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



Theorem (After-class Exercise)

Maximum-weight edge in any cycle is unique \implies Unique MST.



Unique MST [Problem: 10.21 (4)]

To decide whether a graph has a unique MST.

Unique MST [Problem: 10.21 (4)]

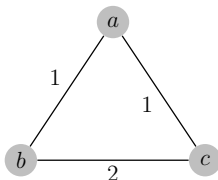
To decide whether a graph has a unique MST.

Ties in Prim's and Kruskal's algorithms

Unique MST [Problem: 10.21 (4)]

To decide whether a graph has a unique MST.

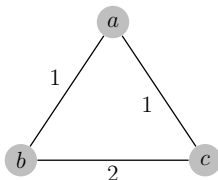
Ties in Prim's and Kruskal's algorithms



Unique MST [Problem: 10.21 (4)]

To decide whether a graph has a unique MST.

Ties in Prim's and Kruskal's algorithms

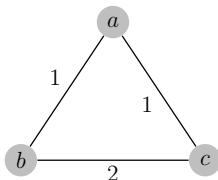


$$\underbrace{T}_{\text{Any MST}} + \underbrace{\{e\}, \forall e \notin T}_{\text{Cycle}}$$

Unique MST [Problem: 10.21 (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.

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$
$$w(T') > w(T)$$

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$

$$w(T') > w(T)$$

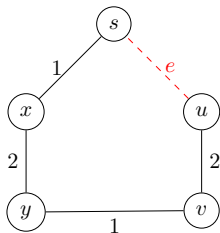
To find all critical edges in $O(m \log m)$ time.

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$

$$w(T') > w(T)$$

To find all critical edges in $O(m \log m)$ time.



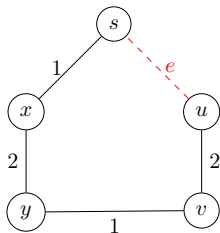
$$w(e) = 3 \quad w(e) = 2$$

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$

$$w(T') > w(T)$$

To find all critical edges in $O(m \log m)$ time.



By Kruskal Algorithm.

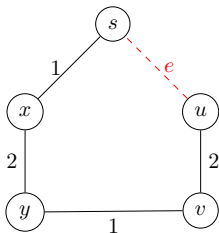
$$w(e) = 3 \quad w(e) = 2$$

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$

$$w(T') > w(T)$$

To find all critical edges in $O(m \log m)$ time.



$$w(e) = 3 \quad w(e) = 2$$

By Kruskal Algorithm.

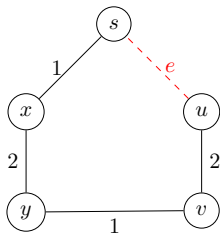
No missing: Check all cycles.

Critical Edges [Problem: 10.12]

$$G \rightarrow T, \quad G' \triangleq G \setminus \{e\} \rightarrow T'$$

$$w(T') > w(T)$$

To find all critical edges in $O(m \log m)$ time.



$$w(e) = 3 \quad w(e) = 2$$

By Kruskal Algorithm.

No missing: Check all cycles.

$$O(m \log m)$$

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' .

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

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' .

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

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' .

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

Theorem

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

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

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' .

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

Theorem

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

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

$$O(n)$$

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' .

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

Theorem

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

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

$$O(n)$$

“On Finding and Updating Spanning Trees and Shortest Paths”, 1975

“Algorithms for Updating Minimum Spanning Trees”, 1978

Feedback Edge Set (FES): [Problem: 10.8]

$\text{FES} \subseteq E : G' = (V, E \setminus \text{FES})$ is acyclic

To find a minimum FES.

Feedback Edge Set (FES): [Problem: 10.8]

$\text{FES} \subseteq E : G' = (V, E \setminus \text{FES})$ is acyclic

To find a minimum FES.

G is connected $\implies G'$ is connected

G' is connected + acyclic $\implies G'$ is an ST

Feedback Edge Set (FES): [Problem: 10.8]

$\text{FES} \subseteq E : G' = (V, E \setminus \text{FES})$ is acyclic

To find a minimum FES.

G is connected $\implies G'$ is connected

G' is connected + acyclic $\implies G'$ is an ST

$\text{FES} \iff G \setminus \text{Max-ST}$

MST with Specified Leaves: [Problem: 10.11]

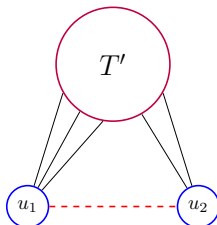
$$G = (V, E), \quad U \subset V$$

To find an MST with U as leaves.

MST with Specified Leaves: [Problem: 10.11]

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

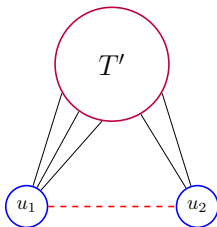
To find an MST with U as leaves.



MST with Specified Leaves: [Problem: 10.11]

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

To find an MST with U as leaves.

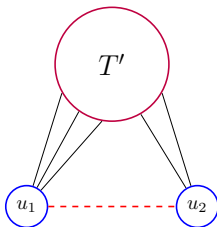


MST T' of $G' = G \setminus U$

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)

MST with Specified Edges: [Problem: 10.13]

$$G = (V, E), \quad S \subset E \text{ (no cycle in } S\text{)}$$

To find an MST with S as edges.

MST with Specified Edges: [Problem: 10.13]

$$G = (V, E), \quad S \subset E \text{ (no cycle in } S\text{)}$$

To find an MST with S as edges.

$G \rightarrow G'$: contract each component of S to a vertex

MST with Specified Edges: [Problem: 10.13]

$$G = (V, E), \quad S \subset E \text{ (no cycle in } S\text{)}$$

To find an MST with S as edges.

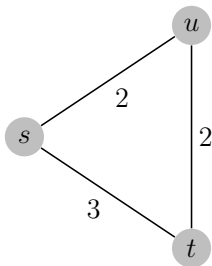
$G \rightarrow G'$: contract each component of S to a vertex

Compute MST on G'

MST v.s. Shortest Path

MST vs. Shortest Paths [Problem: 10.15 (6)]

✗ The shortest path between s and t is necessarily part of some MST.



Sharing Edges [Problem: 10.9]

$$G = (V, E, w), \quad w(e) > 0, \quad s \in V$$

All sssp trees from s must share some edge with **all** (some) MSTs of G .

Sharing Edges [Problem: 10.9]

$$G = (V, E, w), \quad w(e) > 0, \quad s \in V$$

All sssp trees from s must share some edge with **all** (some) MSTs of G .

$$E' \subseteq E : \text{lightest edges leaving } s$$

Sharing Edges [Problem: 10.9]

$$G = (V, E, w), \quad w(e) > 0, \quad s \in V$$

All sssp trees from s must share some edge with **all** (some) MSTs of G .

$$E' \subseteq E : \text{lightest edges leaving } s$$

$$E' \subseteq \forall \text{ sssp tree from } s$$

Sharing Edges [Problem: 10.9]

$$G = (V, E, w), \quad w(e) > 0, \quad s \in V$$

All sssp trees from s must share some edge with **all** (some) MSTs of G .

$$E' \subseteq E : \text{lightest edges leaving } s$$

$$E' \subseteq \forall \text{ sssp tree from } s$$

$$\forall \text{ MST } T \text{ of } G : T \cap E' \neq \emptyset$$

