

# Minimum Spanning Tree (MST)

Hengfeng Wei

hengxin0912@gmail.com

May 19, 2016

# Minimum Spanning Tree (MST)

1 Cut Property and Cycle Property

2 Updating MST

3 Variants of MST

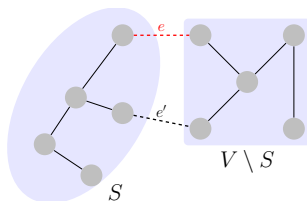
4 MST vs. Shortest Paths

# Cut property of MST

## Cut Property [Problem: 3.6.18 (a)]

- ▶ Graph  $G = (V, E)$  (undirected, connected, weighted)
- ▶ Weights are distinct
- ▶ A cut  $(S, V \setminus S)$  where  $S, V - S \neq \emptyset$
- ▶ Let  $e = (u, v)$  be a minimum-weight edge across  $(S, V \setminus S)$

Then  $e$  must be in *some* MST of  $G$ .



# Cut property of MST

Cut Property [Problem: 3.6.18 (a)]

Proof.

Basic idea:  $T$  is an MST of  $G$ .

- ▶  $e \in T$
- ▶  $e \notin T \Rightarrow e \in T'$ 
  - ▶  $T + \{e\}$  to construct a cycle  $C$
  - ▶  $\exists e' = (u', v') \in C$  ( $e' \in P_{u,v}$ ),  $e'$  crosses  $(S, V \setminus S)$
  - ▶  $T' = T + \{e\} - \{e'\}$ : spanning tree (connected, acyclic)
  - ▶  $w(e') \geq w(e) \Rightarrow w(T') \leq w(T) \Rightarrow w(T') = w(T)$



Remark.

- ▶ a minimum-weight edge;  $\in$  some MST
- ▶ exchange argument

# Cut property of MST

## Application of cycle property [Problem: 3.6.15]

- ▶ [Problem: 3.6.15 (3)]:  $e \in G$  is a lightest edge  $\Rightarrow e \in \exists$  MST of  $G$
- ▶ [Problem: 3.6.15 (4)]:  $e \in G$  is the unique lightest edge  $\Rightarrow e \in \forall$  MST
- ▶ [Problem: 3.6.15 (9)]:  $e = (u, v) \in \exists$  MST  $T$  of  $G \Rightarrow e$  is a lightest edge across some cut  $(S, V \setminus S)$  (*converse of cut property*)

## Solution.

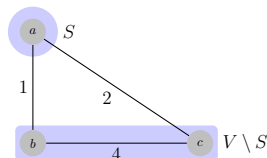
- ▶ [Problem: 3.6.15 (3)]:  $(S = \{u\}, V \setminus S)$
- ▶ [Problem: 3.6.15 (4)]: By contradiction.  $e \notin T$ ;  
 $T' = T + \{e\} - \{e'\} \Rightarrow w(T') < w(T)$
- ▶ [Problem: 3.6.15 (9)]:
  1. to find the cut  $(S, V \setminus S)$ 
    - ▶  $T - \{e\}$
  2. to prove that  $e$  is a lightest edge across  $(S, V \setminus S)$ 
    - ▶ by contradiction:  $T' = T - \{e\} + \{e'\}$

# Cut property of MST

Wrong divide-and-conquer algorithm for MST [Problem: 3.6.29]

- ▶  $G = (V, E, w)$
- ▶  $(V_1, V_2) : ||V_1| - |V_2|| \leq 1$
- ▶  $T_1 + T_2 + \{e\}$ :  $e$  is a lightest edge across  $(V_1, V_2)$

Solution.

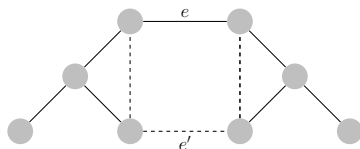


# Cycle property of MST

Cycle property [Problem: 3.6.18 (b)]

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

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



# Cycle property of MST

Cycle property [Problem: 3.6.18 (b)]

Proof.

Basic idea: pick any MST  $T$  of  $G$

- ▶  $e \notin T$
- ▶  $e \in T \Rightarrow e \notin T'$ 
  - ▶  $T - \{e\} \Rightarrow (S, V \setminus S)$
  - ▶  $\exists e' = (u', v') \in C$  ( $e' \in P_{u,v}$ ) across the cut
  - ▶  $T' = T - \{e\} + \{e'\}$ : spanning tree
  - ▶  $w(e') \leq w(e) \Rightarrow w(T') \leq w(T) \Rightarrow w(T') = w(T)$



Remark.

- ▶ Why don't we pick any  $e' \in C$ ?
- ▶ “Anti-Kruskal” (reverse-delete; also by Kruskal) [Problem: 3.6.20 (c)]



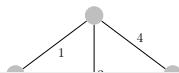
# Applications of cycle property

## Applications of cycle property [Problem: 3.6.15]

- ▶ [Problem: 3.6.15 (2)]:  $C \subseteq G, e \in C$ ,  $e$  is the unique maximum-weighted edge of  $C \Rightarrow e \notin \text{any MST of } G$
- ▶ [Problem: 3.6.15 (5); 3.6.18 (c)]:  $C \subseteq G, e \in C$ ,  $e$  is the unique lightest edge of  $C \Rightarrow e \in \forall \text{ MST}$
- ▶ [Problem: 3.6.15 (1)]:  $G = (V, E), |E| > |V| - 1$ ,  $e$  unique maximum-weighted edge  $\Rightarrow e \notin \text{any MST}$
- ▶ [Problem: 3.6.20 (a)]:  $e$  does not belong to any cycle  $\Rightarrow e \in \forall \text{ MST}$

## Solution.

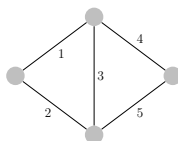
- ▶ [Problem: 3.6.15 (2)]: By contradiction.  $T' = T - \{e\} + \{e'\}$
- ▶ [Problem: 3.6.15 (5); 3.6.18 (c)]



# Properties of MST

✓ or ✗ [Problem: 3.6.15]

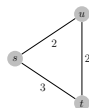
1. ✗  $|E| > |V| - 1$ ,  $e$  is the unique maximum edge  $\Rightarrow e$  does not belong to any MST.
2. ✓ If  $G$  has a cycle with a unique maximum edge  $e$ , then  $e$  cannot be part of any MST. (Prove: Cycle property)
3. ✓ Let  $e$  be any edge of minimum edge in  $G$ . Then  $e$  belongs to some MST. (Prove: Cut property)
4. ✓ If the minimum edge is unique, then it belongs to every MST.
5. ✗ If  $G$  has a cycle with a unique minimum edge  $e$ , then  $e$  belongs to every MST.



# Properties of MST

✓ or ✗ [Problem: 3.6.15]

6. ✗ The shortest-path tree computed by Dijkstra's algorithm is necessarily an MST.
7. ✗ The shortest path between two nodes is necessarily part of some MST.



8. ✓ Prim's algorithm works correctly when there are negative edges.
9. ✓ If  $e$  belongs to some MST, then  $e$  is a minimum edge across some cut.
10. ✓  $w > 0$ ; Vertex  $s$ ; shortest-path tree of  $s$  and some MST share a common edge [Problem: 6.1.5]
11. ✓  $w'(e) = (w(e))^2$  [Problem: 6.2.2]

# Uniqueness of MST

## Uniqueness of MST [Problem: 3.6.21]

Distinct weights  $\Rightarrow$  unique MST.

Solution.

Proof.

By contradiction: two MSTs  $T_1 \neq T_2$ .

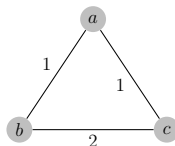
- ▶  $\Delta E = \{e \mid e \in T_1 \setminus T_2 \vee e \in T_2 \setminus T_1\}$
- ▶  $e = \min \Delta E$ . Suppose  $e \in T_1 \setminus T_2$
- ▶  $T_2 + \{e\} \Rightarrow C$
- ▶  $\exists (e' \in C) \notin T_1 \Rightarrow e' \in T_2 \setminus T_1 \Rightarrow e' \in \Delta E \Rightarrow w(e') > w(e)$
- ▶  $e' \in T_2 \setminus T_1 \Rightarrow e' \in \Delta E$
- ▶  $T' = T_2 + \{e\} - \{e'\} \Rightarrow w(T') < w(T_2)$



# Uniqueness of MST

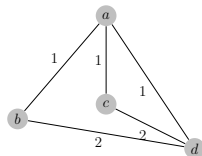
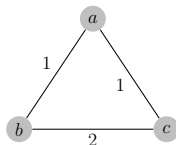
## Conditions for Uniqueness of MST [Problem: 3.6.19]

- ▶ [Problem: 3.6.19 (a)]: unique MST  $\nRightarrow$  equal weights



- ▶ [Problem: 3.6.19 (c)]: Counterexamples

- ▶ ~~X~~cut: minimum-weight edge across any cut is unique
- ▶ ~~X~~cycle: maximum-weight edge in any cycle is unique



# Minimum Spanning Tree (MST)

- 1 Cut Property and Cycle Property
- 2 Updating MST**
- 3 Variants of MST
- 4 MST vs. Shortest Paths

# Updating MST

Decreasing/increasing edge weight [Problem: 3.6.6]

$G$  and an MST  $T$

1.  $w(e)$  is decreased:  $w'(e) = w(e) - k$
2.  $w(e)$  is increased

Solution for (1).

- ▶  $e \in T$ : no need to update  $T' = T$ .

$$w'(T') = w(T) - k \Rightarrow w'(T') < w(T).$$

To prove that  $T'$  is an MST of  $G'$ :

Suppose  $\exists T'' : T''$  is an ST of  $G'$  and  $w'(T'') < w'(T')$ .

- ▶  $e \notin T''$ :  $w(T'') = w'(T'') < w'(T') < w(T)$
- ▶  $e \in T''$ :  $w(T'') = w'(T'') + k < w'(T') + k = w(T)$
- ▶  $e \notin T$ :  $T' = T + \{e\} - \{e'\}$ ;  $e'$  is the maximum-weight edge in cycle and  $w(e') > w(e)$ 
  - ▶  $e \notin T''$ :  $w(T'') = w'(T'') < w'(T') < w(T)$

# Updating MST

## Adding vertex to MST [Problem: 3.6.2]

- ▶  $G = (V, E)$ ; an MST  $T$
- ▶  $G' = (V', E')$ :  $V' = V + \{X\}$ ,  $E' = E + E_X$ ;  $E_X$ : incident edges to  $X$
- ▶ To find an MST  $T'$  of  $G'$

## Solution.

1. Recomputing  $O((m + n) \log n)$
2.
  - ▶ There exists an MST of  $G'$  that includes no edges in  $G \setminus T$
  - ▶ Run MST alg. on  $G'' = (V + \{X\}, T + E_X)$
  - ▶  $O(n \log n)$
3.  $O(n)$ 
  - ▶ “On Finding and Updating Spanning Trees and Shortest Paths”, 1975
  - ▶ “Algorithms for Updating Minimum Spanning Trees”, 1978



# Minimum Spanning Tree (MST)

- 1 Cut Property and Cycle Property
- 2 Updating MST
- 3 Variants of MST**
- 4 MST vs. Shortest Paths

# Variants of MST

Feedback edge set: [Problem: 3.6.4]

1. maximum spanning tree
2. (minimum) feedback edge set:
  - ▶ a set of edges which, when removed from the graph, leave an acyclic graph  $G'$
  - ▶ assuming  $G$  is connected  $\Rightarrow G'$  is connected
  - ▶ feedback *arc* set: “cycle”  $\Rightarrow$  circular dependency

Solution.

- ▶  $G'$  is connected + acyclic  $\Rightarrow G'$  is an ST
- ▶  $\text{FES} \Leftrightarrow G \setminus \text{Max-ST}$

# Variants of MST

Edge weights [Problem: 3.6.15 (8); 3.6.16]

- ▶ [Problem: 3.6.15 (8)]: negative edges for Prim algorithm
- ▶ [Problem: 3.6.16]:  $w'(e) = (w(e))^2$

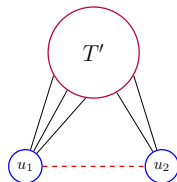
# Variants of MST

## MST with specified leaves: [Problem: 3.6.7]

- ▶  $G = (V, E), U \subset V$
- ▶ finding an MST with  $U$  as leaves

## Solution.

- ▶  $G' = G \setminus U$
- ▶ MST  $T'$  of  $G'$
- ▶ attach  $\forall u \in U$  to  $T'$  (lightest edge)



# Variants of MST

ST with specified edges: [Problem: 3.6.10]

- ▶  $G = (V, E), S \subset E$  (no cycle in  $S$ )
- ▶ finding an MST with  $E$  as edges

Solution.

- ▶ contract each isolated component of  $S$  to a *super-vertex*
- ▶  $G \rightarrow G'$
- ▶ find MST of  $G'$

# Minimum Spanning Tree (MST)

- 1 Cut Property and Cycle Property
- 2 Updating MST
- 3 Variants of MST
- 4 MST vs. Shortest Paths

# MST vs. shortest paths

[Problem: 3.6.15]

- ▶ [Problem: 3.6.15 (6)]: Dijkstra  $\Rightarrow$  SSSP tree  $\Rightarrow$ ? MST
- ▶ [Problem: 3.6.15 (7)]:  $s \rightarrow t$  shortest path  $\Rightarrow$ ?  $\subseteq \exists$  MST

# MST vs. shortest paths

## Sharing edges [Problem: 3.6.5]

- ▶  $G = (V, E), w(e) > 0$
- ▶ Given  $s$ : all sssp trees from  $s$  must share some edge with **all** (some) MSTs of  $G$

## Solution

$E'$ : lightest edges leaving  $s$

- ▶ any MST  $T$  of  $G$ :  $T \cap E' \neq \emptyset$
- ▶  $E' \subset \forall$  sssp trees



