CS5800 – ALGORITHMS

MODULE 7. GREEDY ALGORITHMS - II

Lesson 1: MST- Basics

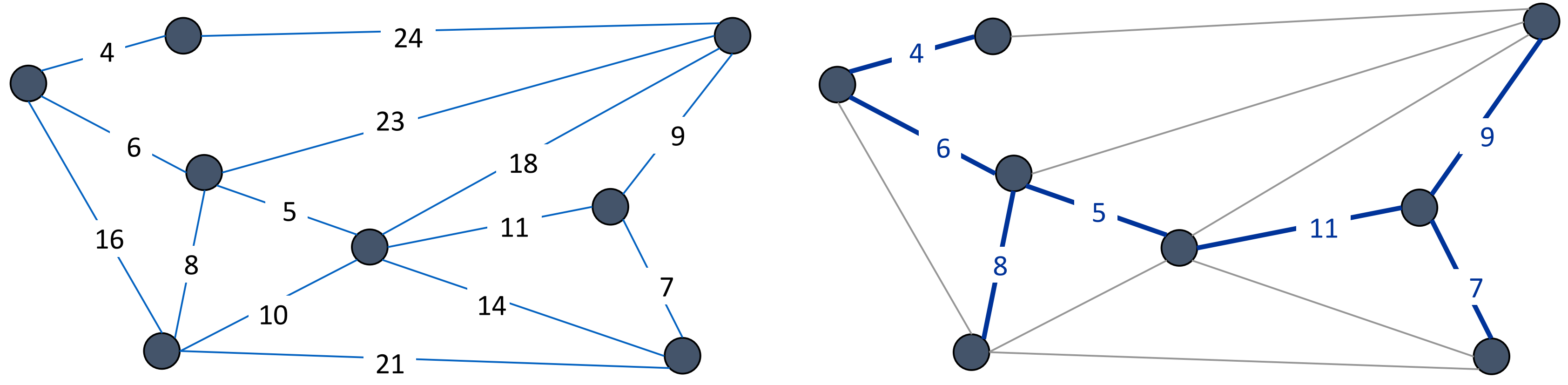
Ravi Sundaram

Topics

- Problem
- Applications
- Greedy Algorithms
- Key properties
- State of the art
- Summary

Minimum Spanning Tree

- Minimum spanning tree. Given a connected graph G = (V, E) with real-valued edge weights c_e, an MST is a subset of the edges $T \subseteq E$ such that T is a spanning tree whose sum of edge weights is minimized.



G = (V, E)

T, $\Sigma_{e \in T}\, c_e = 50$

- How many trees does a complete graph have?
  - $\Omega(n!)$ is easy to see, but actually
  - n^{n-2} spanning trees of K_n – Cayley's theorem
  - Hence can't solve by brute force.

Applications

- MST is a fundamental problem with diverse applications.
- Network design.
  - telephone, electrical, hydraulic, TV cable, computer, road
- Approximation algorithms for NP-hard problems.
  - traveling salesperson problem, Steiner tree
- Indirect applications.
  - max bottleneck paths
  - LDPC codes for error correction
  - image registration with Renyi entropy
  - learning salient features for real-time face verification
  - reducing data storage in sequencing amino acids in a protein
  - model locality of particle interactions in turbulent fluid flows
  - autoconfig protocol for Ethernet bridging to avoid cycles in a
  - network
- Cluster analysis.

Greedy Algorithms

- Kruskal's algorithm. Start with $T = \phi$. Consider edges in ascending order of cost. Insert edge e in T unless doing so would create a cycle.

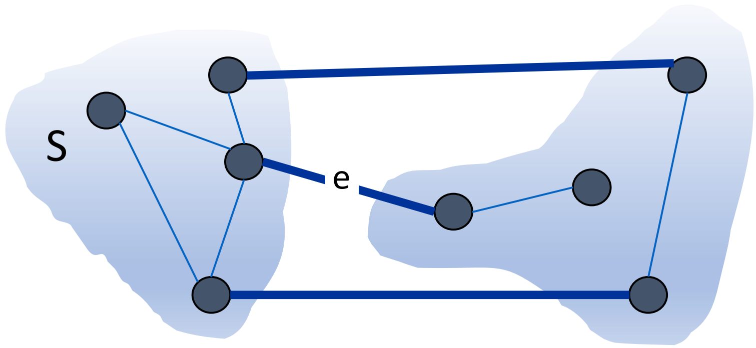
- Reverse-Delete algorithm. Start with T = E. Consider edges in descending order of cost. Delete edge e from T unless doing so would disconnect T.

- Prim's algorithm. Start with some root node s and greedily grow a tree T from s outward. At each step, add the cheapest edge e to T that has exactly one endpoint in T.

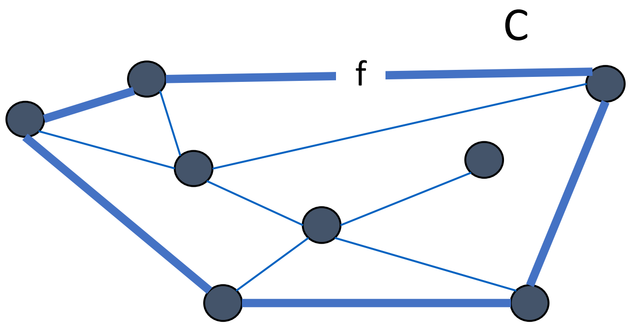
- All three algorithms produce an MST.

Key properties

- Simplifying assumption. All edge costs c_e are distinct.
- Cut property. Let S be any subset of nodes, and let e be the min cost edge with exactly one endpoint in S. Then the MST contains e.
- Cycle property. Let C be any cycle, and let f be the max cost edge belonging to C. Then the MST does not contain f.



e is in the MST



f is not in the MST

MST Algorithms: Theory

- Deterministic comparison-based algorithms.
  - $O(m \log n)$ [Jarník, Prim, Dijkstra, Kruskal, Boruvka]
  - $O(m \log \log n)$ [Cheriton-Tarjan 1976, Yao 1975]
  - $O(m\, \beta(m, n))$. [Fredman-Tarjan 1987]
  - $O(m \log \beta(m, n))$. [Gabow-Galil-Spencer-Tarjan 1986]
  - $O(m\, \alpha(m, n))$. [Chazelle 2000]

- Holy grail. $O(m)$.

- Notable.
  - $O(m)$ randomized. [Karger-Klein-Tarjan 1995]
  - $O(m)$ verification. [Dixon-Rauch-Tarjan 1992]

- Euclidean.
  - 2-d: $O(n \log n)$. compute MST of edges in Delaunay
  - k-d: $O(k\, n^2)$. dense Prim

Summary

- MST problem – given graph with edge weights find mimimum weight spanning tree
- Application – fundamental subroutine useful for network design, approximation algorithms, clustering etc.
- 3 basic algorithms – all greedy – Kruskal’s, reverse-delete and Prim’s
- Main Takeaway: MST is a fundamental problem that can be solved in near linear time. (Finding an exact linear-time algorithm is still open)