Greedy Algorithms Overview

“An algorithm that finds a solution by adding elements one by one, where each element that is added is the best current choice without regard to the future consequences of this choice.”

- Minimum Spanning Tree and Kruskal’s algorithm
- Matroids and Subset Systems
- Bipartite Matching and Intersections of Matroids
- Union-Find Data Structure
Minimum Spanning Tree and Kruskal’s Algorithm

**Problem:** Given an undirected, connected graph \((V, E)\) with edge weights find the min-weight subset \(E' \subset E\) such that the graph \((V, E')\) is acyclic and connected, i.e., a min-weight spanning tree (MST).

**Algorithm (Kruskal)**

1. *Sort edges by non-decreasing weight*
2. \(F = \emptyset\)
3. Until \(F\) is a spanning tree of \(G\)
   3.1 Get the next edge \(e\)
   3.2 If \(F + e\) is acyclic then \(F = F + e\)

The algorithm produces a tree because a) it never completes a cycle so the end result is acyclic and b) for any edge \((u, v)\) in a tree of the original graph, either \((u, v)\) is added or there is a path in \(F\) between \(u\) and \(v\).
Running Time of Kruskal’s Algorithm

**Implementation:** Maintain an array $A$ with an entry for each $v \in V$ that indicates which connected component it belongs to.

- **Sorting:** $O(|E| \log |E|)$
- **Checking if acyclic:** $|E|$ checks and each is $O(1)$ time.
- **Adding $e$ to $F$:** Updating array takes $O(|V|)$ time and array is updated exactly $|V| - 1$ times.

**Total Running Time:** $O(|E| \log |E| + |V|^2)$

Will make this $O(|E| \log |E|)$ later via the union-find data structure
Correctness of Kruskal’s Algorithm (1/2)

Let $S(F)$ be the set of spanning trees that extend a forest $F$.

**Lemma**

*If $S(F)$ contains an MST of $G$ and $e$ is the min-weight edge not in $F$ and not causing a cycle in $F$. Then $S(F + e)$ also contains an MST for $G$.***

**Proof.**

- Let $T' \in S(F)$ be an MST that doesn't include $e$
- Adding $e$ to $T'$ makes a cycle $C$
- Since $e$ doesn't make a cycle in $F$, there exists $e' \in C \setminus F$
- By greediness $w(e) \leq w(e')$ (note that $e'$ doesn't form a cycle with $F$ because $F + e' \in T'$ so the greedy algorithm could potentially add it to $F$)
- Then $T = T' + e - e'$ is a MST in $S(F + e)$:
  - $T$ is a spanning tree in $S(F + e)$
  - $T$ is a MST because $w(T) = w(T') + w(e) - w(e') \leq w(T')$

\[\square\]
Theorem
For every forest $F$ produced by Kruskal, $S(F)$ contains a MST of $G$.

Proof.
- Base Case $F = \emptyset$: $S(F)$ contains all spanning trees including MST
- Induction Step: By the previous lemma, if $S(F)$ contains a MST for $G$, then so does $S(F + e)$

Since the final forest is actually a tree $T$, $S(F) = T$ is a MST.