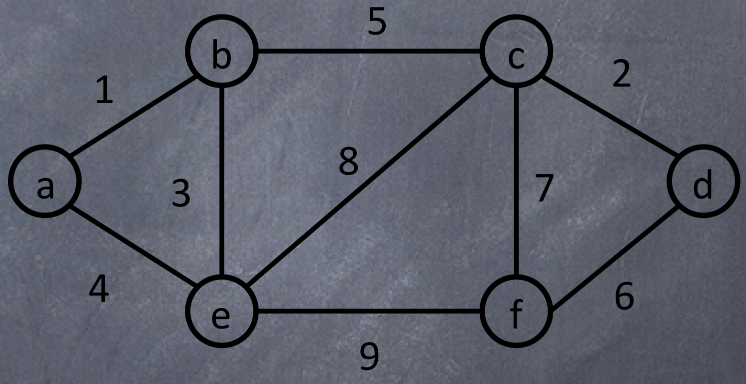
Minimum Spanning Trees

A Network Design Problem

Given: undirected graph $G = (V, E)$ with edge costs $c_e > 0$

Find: edge subset $T \subseteq E$ such that (V, T) is connected and total cost $\sum_{e \in T} c_e$ is as small as possible



Fundamental problem with many applications!

Example

Example on board: total cost of different subgraphs

Minimum Spanning Tree Problem

Lemma. Let T be a minimum-cost solution of the network design problem. Then (V, T) is a tree.

Proof on board

Definition. $T \subseteq E$ is a spanning tree if (V, T) is a tree

Network design problem is the **Minimum Spanning Tree (MST) Problem**

Greedy MST Template (Kruskal and Prim)

"Grow" a tree greedily

```
T = {}
While |T| < n-1 {    // (V, T) is not connected
   Pick "best" edge e that does not create a
      cycle when added to T
   T = T ∪ {e}
}
```

Kruskal's Algorithm

Grow many small trees

Sort edges by weight: $c_1 \leq c_2 \leq \ldots \leq c_m$

```
T = {}
for e = 1 to m {
    if adding e to T does not cause a cycle {
        T = T ∪ {e}
    }
}
```

Example on board

Prim's Algorithm

Grow a tree outward from starting node s

```
T = {}
S = {s}   // connected nodes
While |T| < n-1 {
    Let e = (u, v) be the minimum cost edge from
          S to V-S
    T = T ∪ {e}
    S = S ∪ {v}
}
```

Example on board

Analysis: Cut Property

Simplifying assumption. All edge weights are distinct.

Theorem (Cut Property). Assume edge weights are distinct, and let (S, V-S) be a partition of V into two nonempty sets. Let e = (v, w) be the minimum cost with $v \in S$ and $w \notin S$. Then every minimum spanning tree contains e.

Illustration and proof on board

Correctness of Prim's Algorithm

Theorem: the tree T returned by Prim's algorithm is a minimum spanning tree.

Proof? (Hint: maintain invariant that T is a subset of some MST, use the cut property)

Correctness of Kruskal's Algorithm

Theorem: the tree T returned by Kruskal's algorithm is a minimum spanning tree.

Proof? What cut can you use to prove that Kruskal's algorithm is correct?

Removing Distinctness Assumption

Idea: Break ties in weights by adding tiny amount to each edge weight so they become distinct.

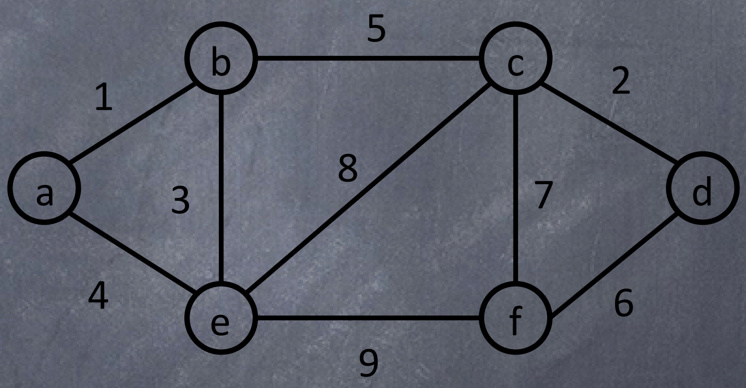
If perturbations are small enough then:
cost(T) > cost(T′) before ⇒ cost(T) > cost(T′) after

Implementation: break ties arbitrarily (e.g., lexicographically)

Network Design: Steiner Tree Problem

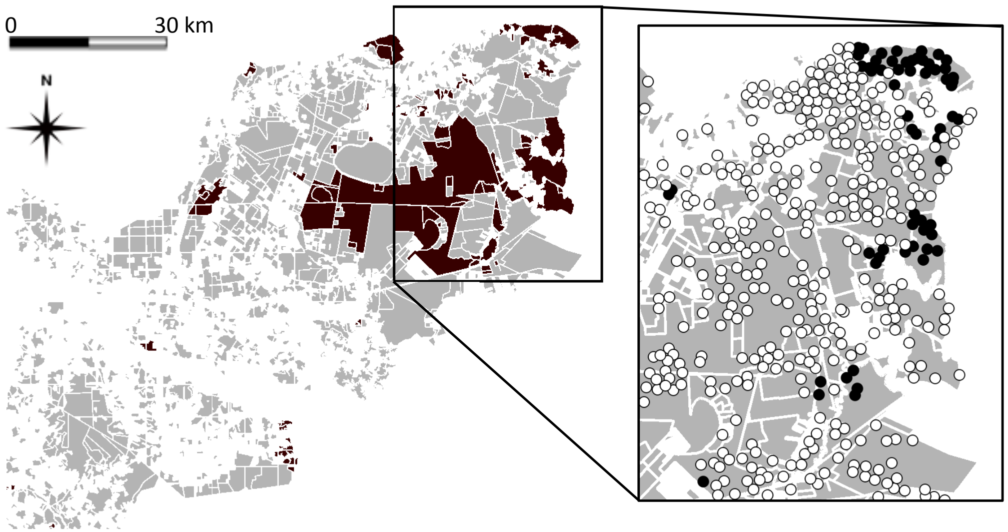
Given: undirected graph $G = (V, E)$ with edge costs $c_e > 0$ and terminals $X \subseteq V$

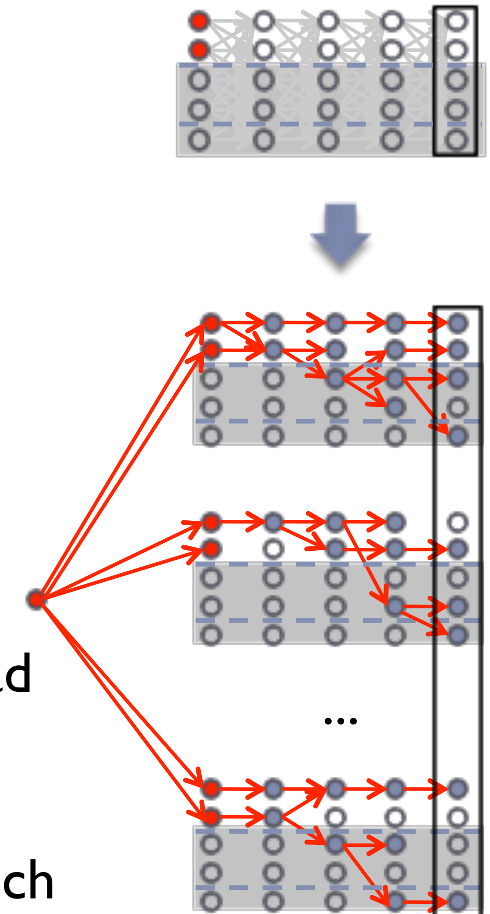
Find: edge subset $T \subseteq E$ such that (V, T) has a path between each pair of terminals and the total cost $\sum_{e \in T} c_e$ is as small as possible



Easier? Harder?

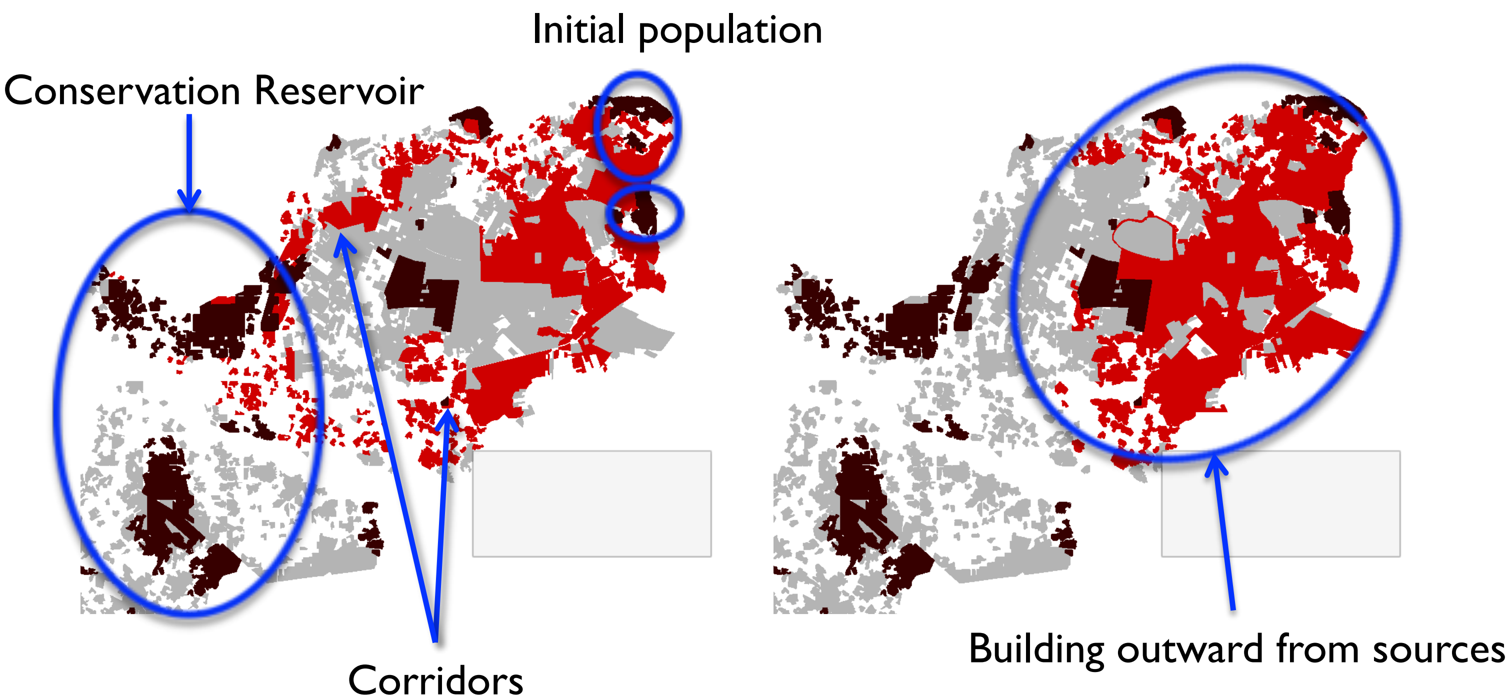
Spatial Conservation Planning





- Which land should I buy to maximize the spread of an endangered species?
- Optimization problem over a graph: decide which nodes to add to the graph, given a fixed budget

Conservation Strategies



Our solution

Greedy baseline

Formulate and solve network design problem as an integer program