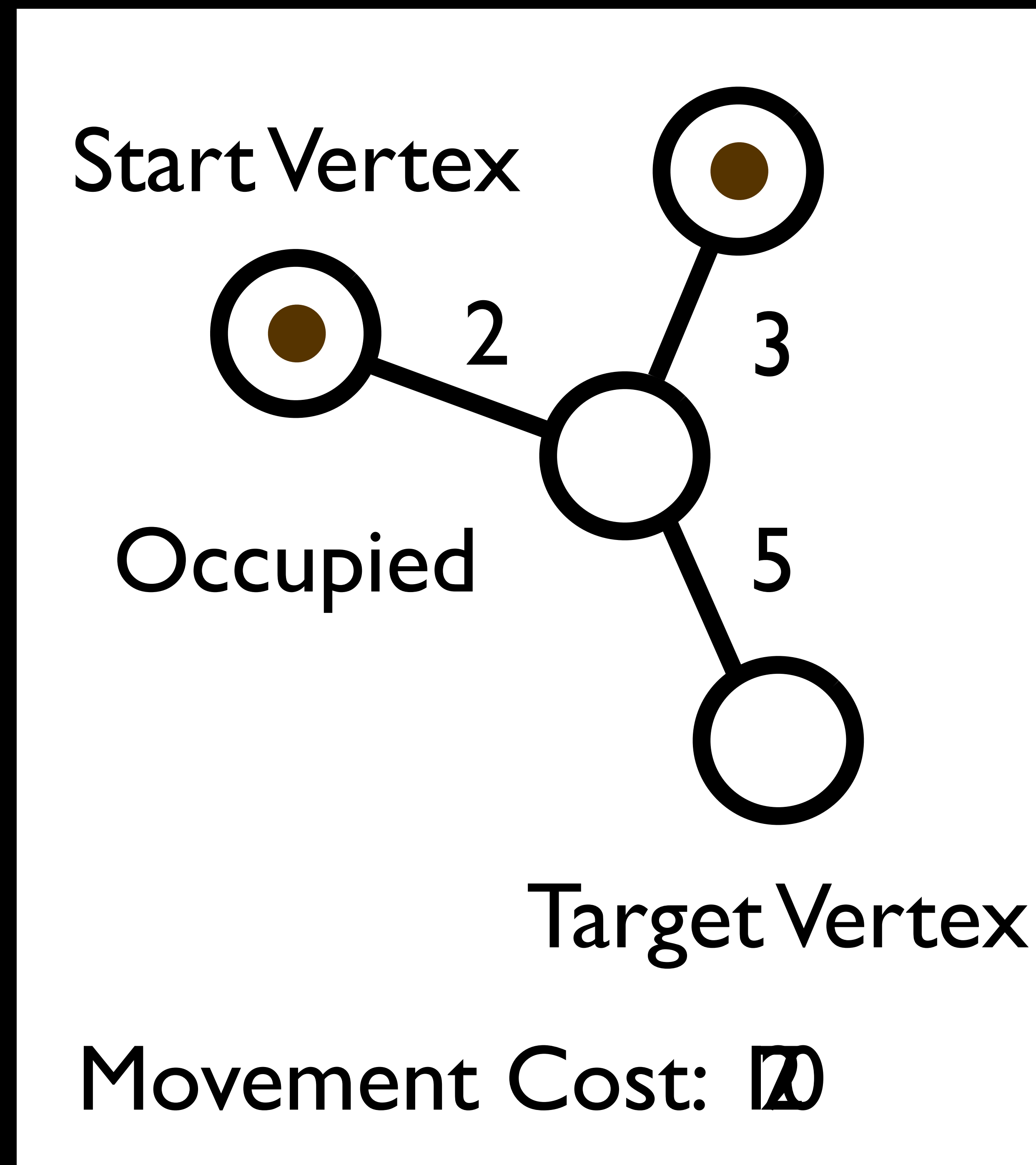
Clearing Paths with Minimum Movement

by Anu Bandi, Daniel Apon

Movement

- Graph (E, V)
- Pebbles
- Movement cost = edge weight
- $|\pi(p)| = \sum_{e \in p} w_e$;
- Total movement = $\sum_i |\pi(p_i)|$



Natural Meaning

- Pebbles = robots, firefighters, cars, etc
- Edges = roads, paths, etc
- Example:
  - Swarm robots need to communicate.

General Background

- Minimizing Movement (SODA 2007)
  - by Demaine, et al [2].
- Graph Problems
- Geometric Problems
- Movement introduces combinatorial explosion.

General Problem Statement

- Input:
  - Graph G = (V, E), |V| = n
  - m pebbles
    - assigned vertices (not necessarily distinct)
- Goal:
  - Achieve graph property
  - Minimizing complexity measure

Problems

Graph Property		Complexity Measure
Connectivity	⟷	Maximum Movement
Directed Connectivity		Sum Movement
s-t Path		Number of Movements
Independent Set		
Perfect Matching		

Results of Demaine, et al.

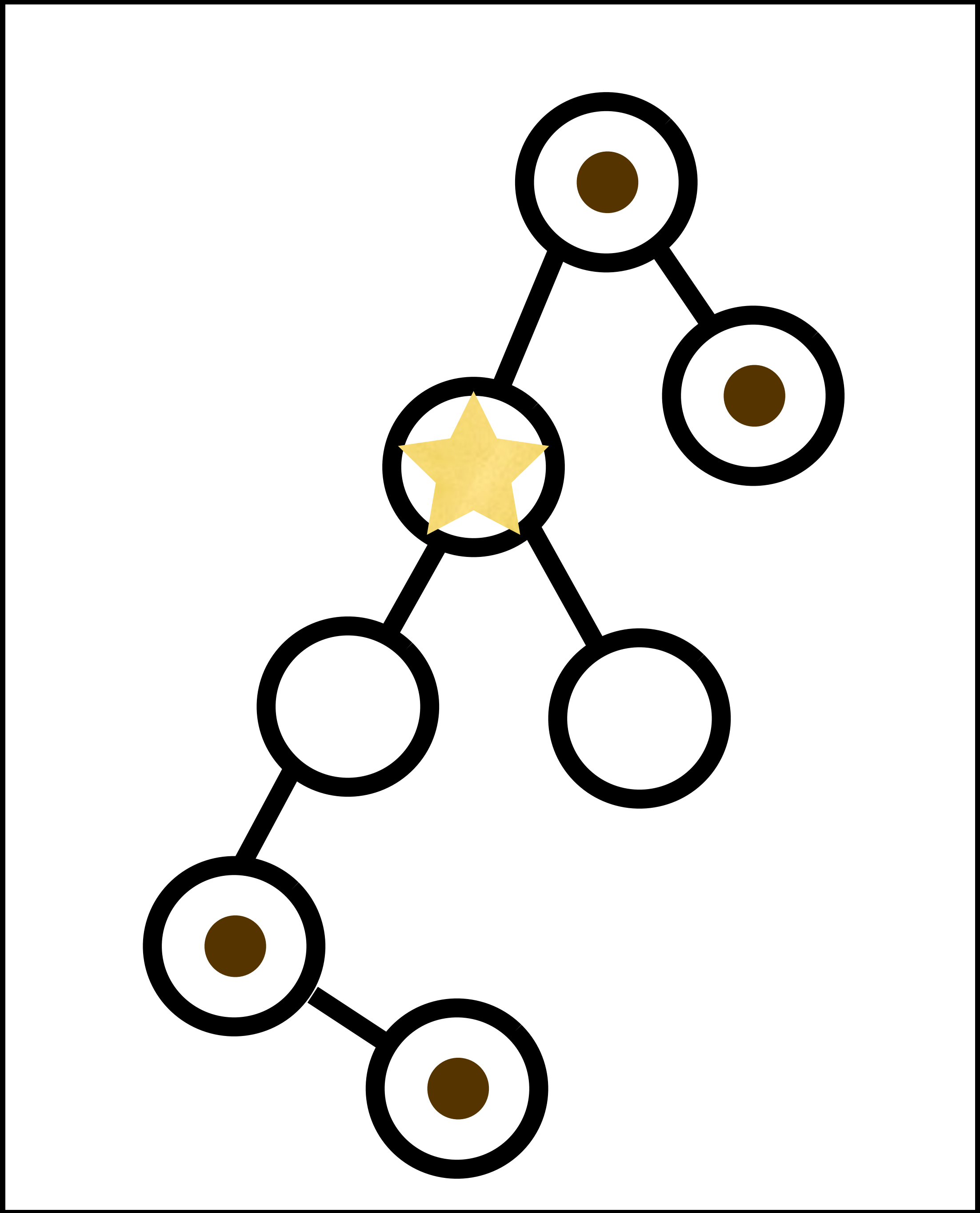
	MAX	SUM	NUM
CON	$O(\sqrt{m/OPT})$	$O(\min\{n, m\})$, $\Omega(n^{1-\epsilon})$	$O(m^{\epsilon})$, $\Omega(\log n)$
DIRCON	ϵm, $\Omega(n^{1-\epsilon})$	open	$O(m^{\epsilon})$, $\Omega(\log^2 n)$
PATH	$O(\sqrt{m/OPT})$	$O(n)$	polynomial
IND	$1+\frac{1}{\sqrt{3}}$ additive in $\mathbb{R}^2$	open	PTAS in $\mathbb{R}^2$
MATCH	polynomial	polynomial	polynomial

Theorem

Given a tree T and a configuration of k pebbles on T, ConMax can be solved in polynomial time.

Proof Sketch

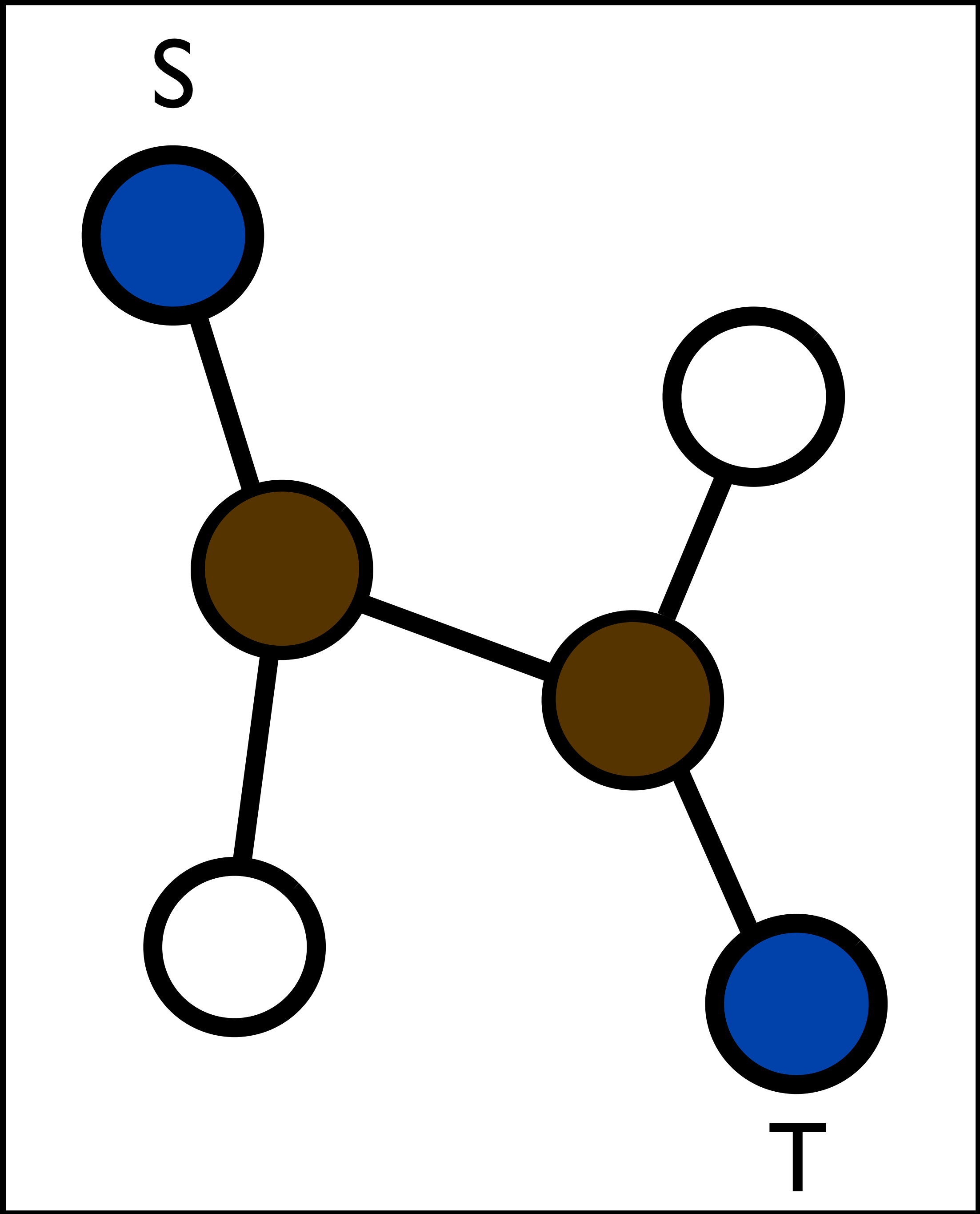
- Guess OPT = 1
- Select vertex v
- Move pebbles toward v
- Find forced vertices
- Bipartite Graph H = (U,V, E)
- Maximum-cardinality matching of H→ pebbles can be moved



Our Work

Clear Path Problem

- New graph property
- Create a pebble-less path between 2 vertices
- Minimize sum, max, or num

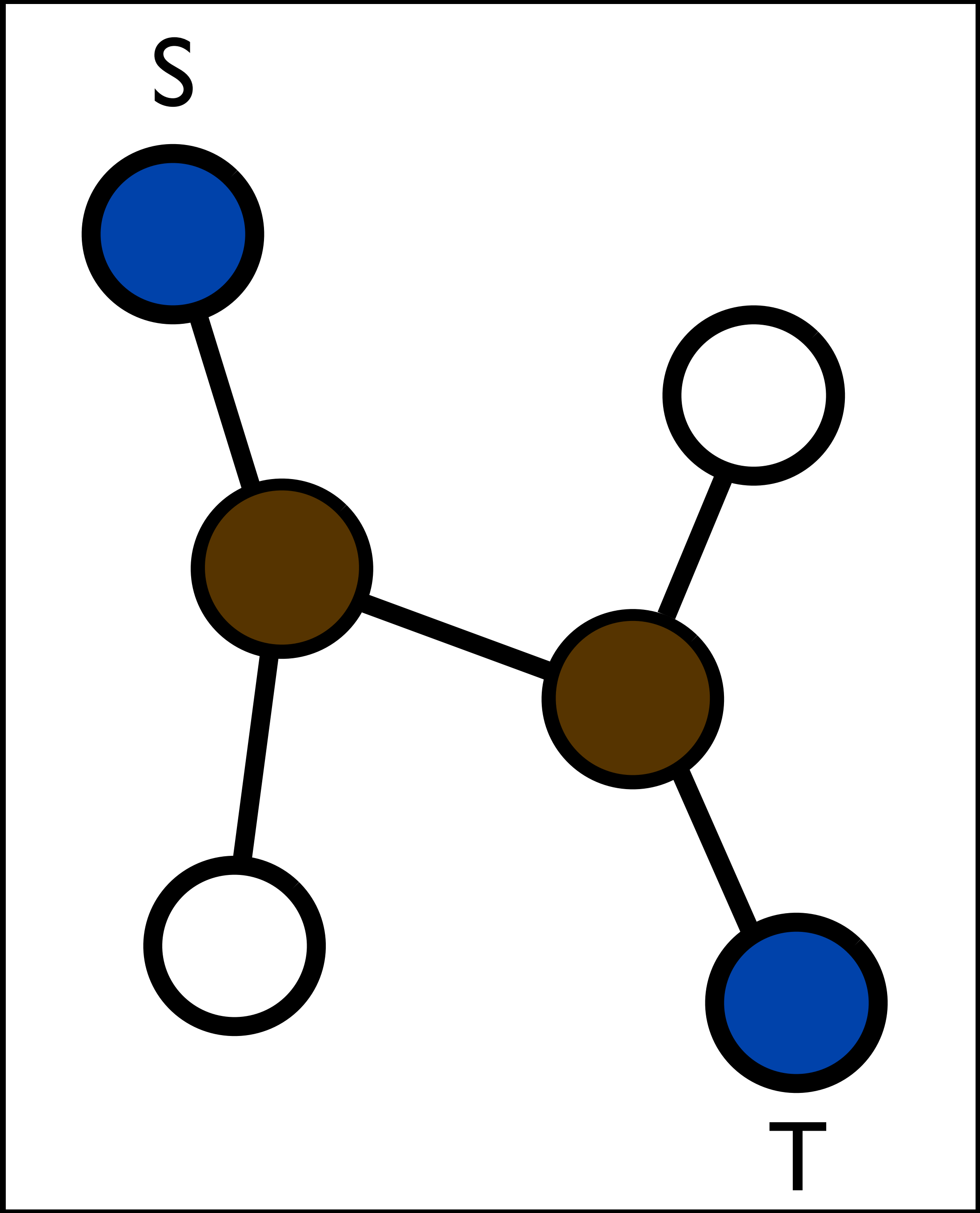


Real World Applications

- Clear path for emergency vehicle
- Store and retrieve goods in warehouse
- Robots clearing debris after disaster

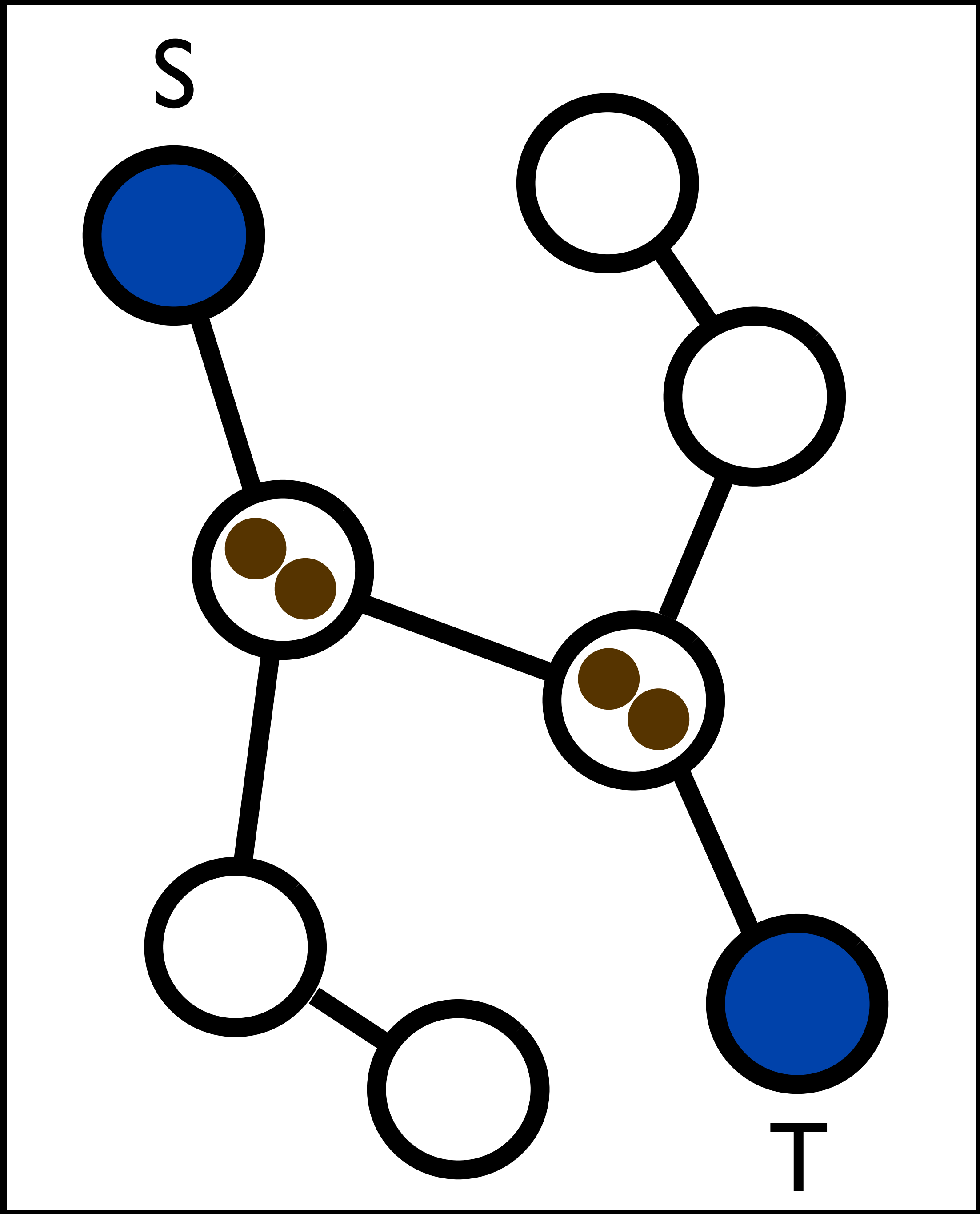
Similarity to Path Finding

- Invert vertices with pebbles
- Solve path finding
- Revert vertices



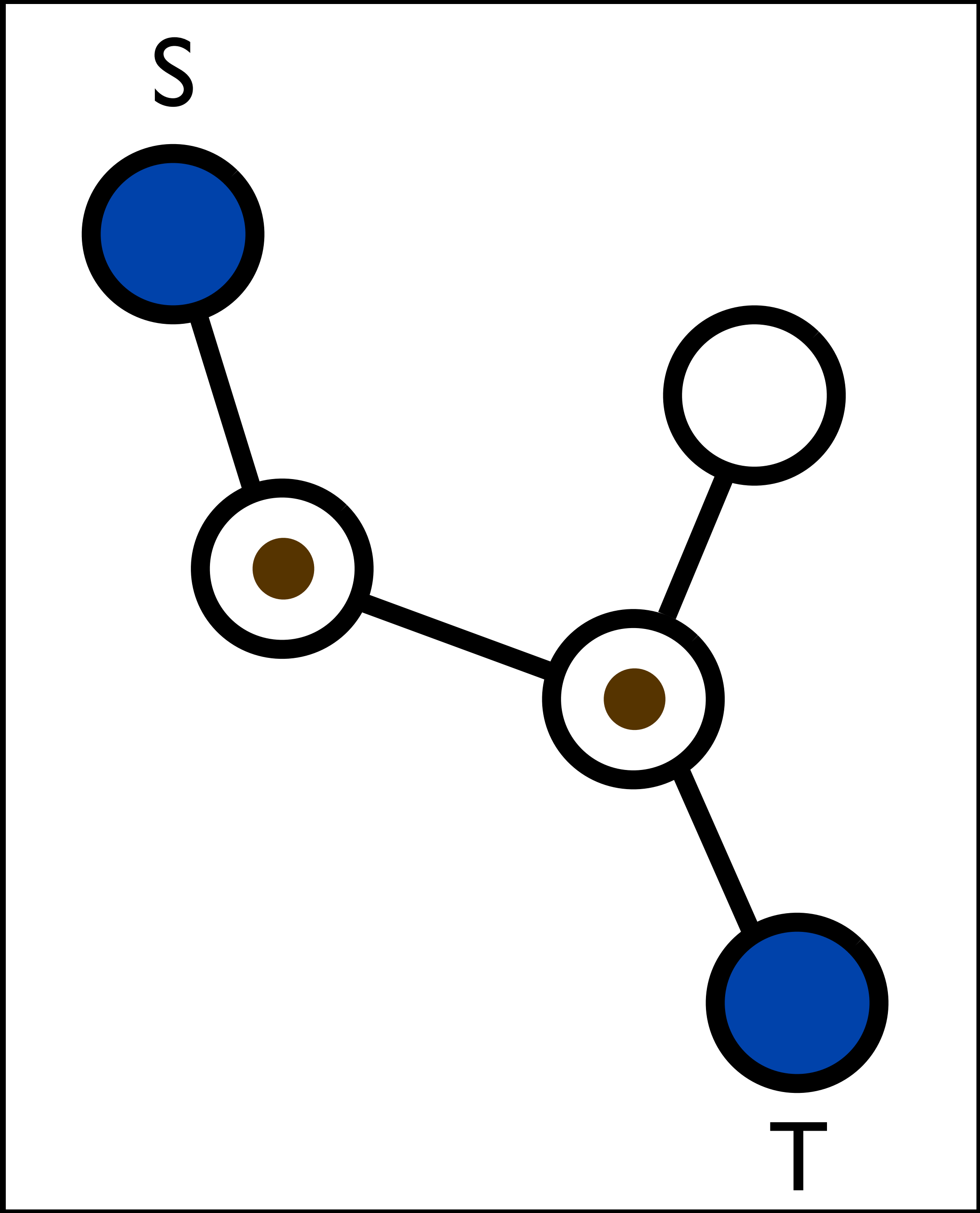
Differences from Path Finding

- Multiple pebbles



Differences from Path Finding 2

- Allow coincident pebbles



Fact 1

In the case of disallowed coincident pebbles, on a arbitrary graph G, the computational problems of clearing an $s - t$ path and creating an $s - t$ path such that we minimize (any measure of) movement are identical.

Problem Statement

- Input:
  - Graph $G = (V, E)$, $|V| = n$
  - m pebbles
    - assigned vertices (not necessarily distinct)
  - Coincident-cost function for $v \in G$
- Output:
  - A set of moves to clear pebbles from s-t path

Goals

- ClearMax

$$\min\left\{\max_{p_i : i\in[m]}\left\{\sum_{(u,v)\in E: u,v\in \mathbb{P}_i} w(u,v) + \mathrm{cost}_{v_i'}\left(i_{p_i,v}\right)\right\}\right\}$$

- ClearSum

$$\min\left\{\sum_{p_i : i\in[m]}\left(\sum_{(u,v)\in E: u,v\in \mathbb{P}_i} w(u,v) + \mathrm{cost}_{v_i'}\left(i_{p_i,v}\right)\right)\right\}$$

- ClearNum