

Section 23.1: Amassing Evidence of Intractability

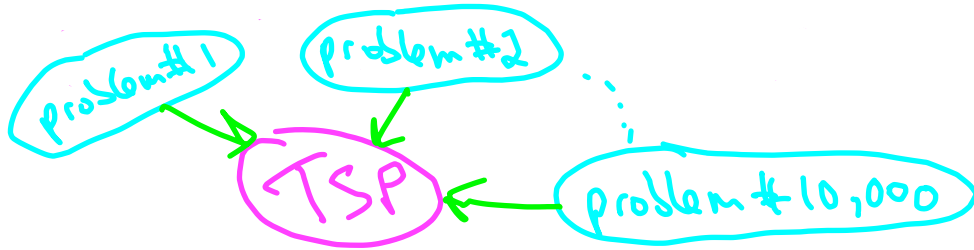
Building a Case with Reductions

Conjecture: (Edmonds '67) no polynomial-time algorithm for the TSP.

Question: how to amass evidence for this?

- ① Choose a really big collection \mathcal{C} of problems.
- ② Prove that every problem in \mathcal{C} reduces to the TSP.

via a Cook reduction



Note: if any problem in \mathcal{C} not polynomial-time solvable \Rightarrow neither is the TSP.

Choosing a Class for the TSP

Question: how to choose \mathcal{C} ? [bigger $\mathcal{C} \Rightarrow$ stronger evidence of TSP's intractability]

Idea: Let $\mathcal{C} =$ ALL computational problems.

Problem: Some problems (like the Halting Problem) can't reduce to the TSP.
[Turing 1936]

Revised idea: $\mathcal{C} =$ all problems that can be solved by naive exhaustive search.

Question: how to define?