Approximation algorithms

CS 146 - Spring 2017

Coping with NP-completeness

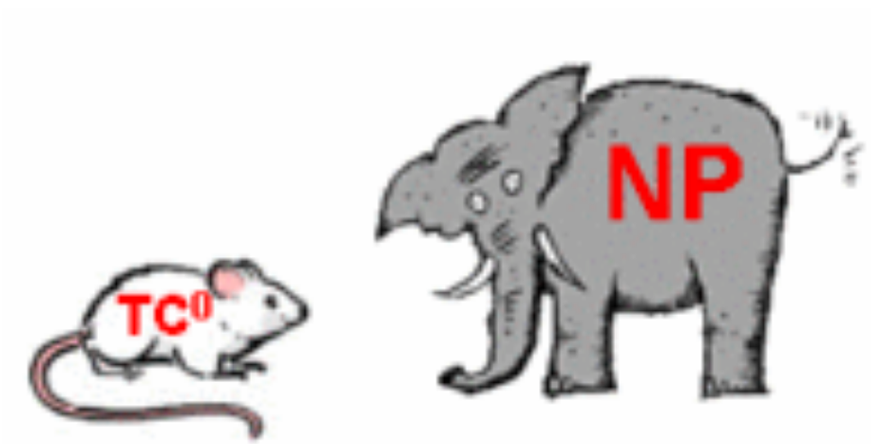
→ heuristic search

→ exponential-time algorithms

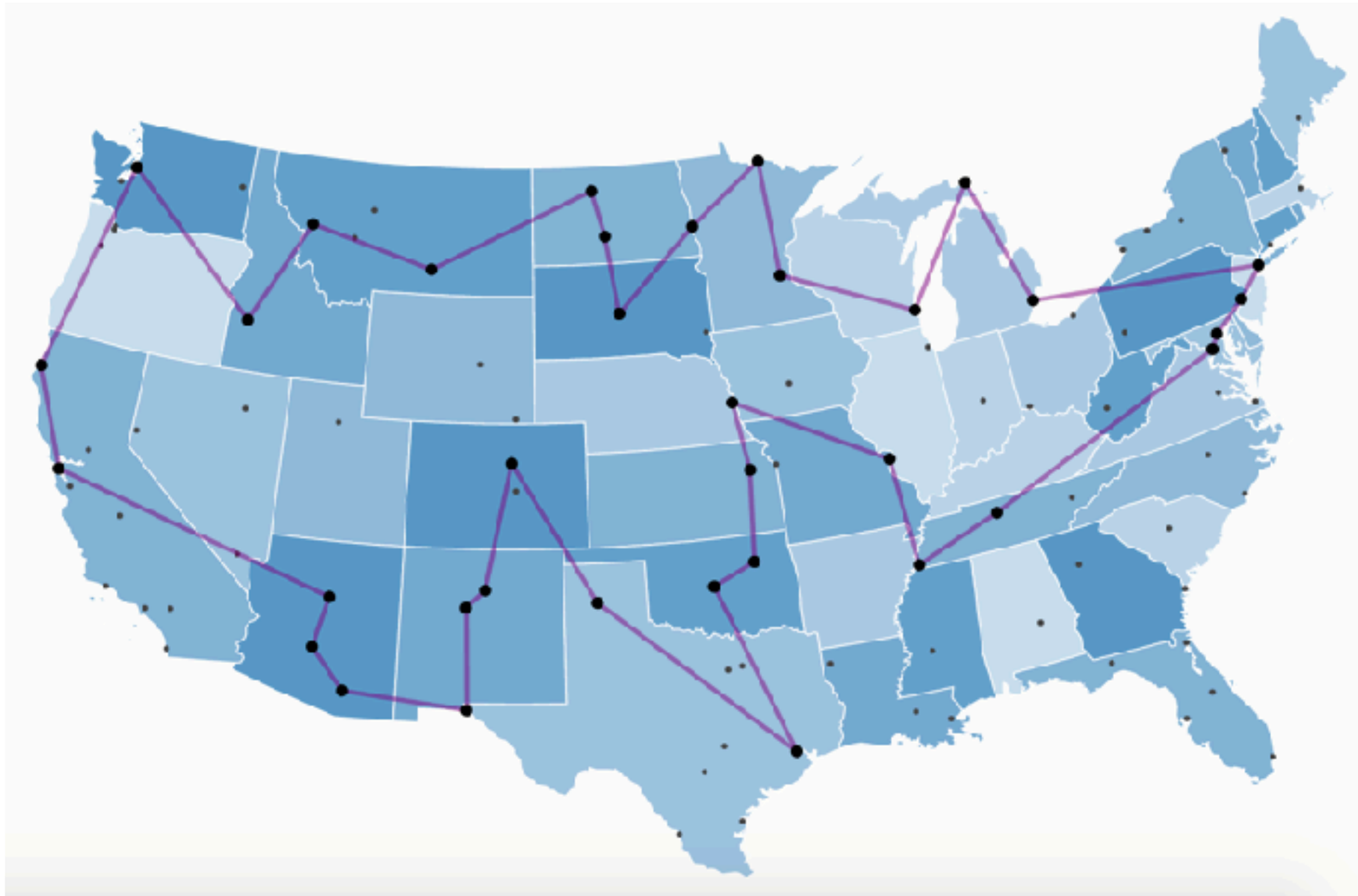
→ **approximation algorithms**

→ fixed-parameter tractable algorithms

http://complexityzoo.com



Traveling salesman problem



input: a connected undirected weighted graph

output: the minimal length cycle containing every vertex

http://examples.gurobi.com/traveling-salesman-problem/

Approximation ratio for minimization problems

approximation ratio of ALG

$$\alpha = \min_{I} \frac{\mathrm{ALG}(I)}{\mathrm{OPT}(I)}$$

value of solution given by algorithm ALG on I

correct value of solution for I

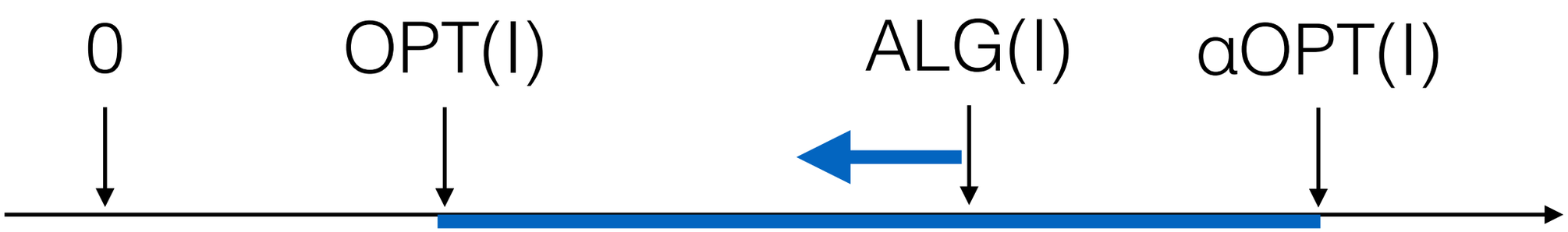
problem instances

- measures factor by which the output of alg ALG exceeds the optimal solution on worst-case input

Approximation algorithm

ALG is an **α-approximation algorithm** for a minimization problem if for all problem instances I

$$\mathrm{ALG}(I) \leq \alpha \mathrm{OPT}(I) \quad \text{or} \quad \alpha = \min_I \frac{\mathrm{ALG}(I)}{\mathrm{OPT}(I)}$$



2 approximation algorithms for TSP

1. MST-doubling heuristic

   - MST-doubling(G) ≤ **2** TSP(G)
     for all undirected weighted graphs G

2. Christofides

   - Christofides(G) ≤ **1.5** TSP(G)
     for all undirected weighted graphs G

MST-doubling heuristic

1. find MST using some MST-finding algorithm (eg Kruskal)

2. construct a cycle through every vertex by doubling every edge of the MST

3. optional shortcutting step: going along the cycle, skip a vertex that has already been visited

The doubling-heuristic is a 2-approximation of TSP

Proof

- TSP with 1 edge removed is a spanning tree, so:
  MST (G) ≤ TSP(G) with 1 edge removed

- so MST(G) ≤ TSP(G)

- multiply both sides by 2:
  MST-doubling(G) ≤ **2** TSP(G)