What to do with NP-hard optimization problems

- efficient exhaustive search (backtracking, branch & bound) - exponential time.

- heuristics
  - local search - start with some solution and try to improve it via small "local" changes.
  - "simulated annealing" overcomes local optima.

- approximation algs. - today's topic

Example

TSP for points in the plane w/ Euclidean distances

- complete graph

triangle inequality

\[ w(a, c) \leq w(a, b) + w(b, c) \]

Approx. Alg.

- compute MST (in black)

- take a tour by walking around it. (in blue)

  (we visit every vertex but maybe more than once)

- take shortcuts to avoid revisiting (in red)

Note: \( \Delta \) inequality \( \Rightarrow \) shortcuts shorter

We can do this in poly. time.
let $l$ be length of resulting tour $l_{\text{TSP}} = \min$ of min TSP tour

**Claim** $l \leq 2l_{\text{TSP}}$ [Note $l_{\text{TSP}} \leq l$]

So in poly. time we find a tour within $2 \times$ optimum. We call this approximation factor 2.

**Proof of Claim.**

$l_{\text{MST}} = \text{length of MST}

l_{\text{MST}} \leq l_{\text{TSP}}$ because deleting one edge of TSP gives a spanning tree.

$l \leq 2l_{\text{MST}}$ because we use every MST edge twice, then take short cuts. (Use Δ ineq.)

Putting these together:

$$l \leq 2l_{\text{TSP}}$$

We say this alg. has approximation factor 2.
Example: Vertex Cover

\[ G = (V, E) \]

want set \( C \subseteq V \) s.t.
\[ \forall (u, v) \in E, u \in C \text{ or } v \in C \text{ (or both)} \]

minimize \( |C| \)

Greedy approximation alg:

\[ C = \emptyset \quad F = E \quad \text{//F is uncovered edges} \]

while \( F \neq \emptyset \)

pick \( e = (u, v) \) from \( F \)

add \( u \) and \( v \) to \( C \)

remove edges incident to \( u \) from \( F \)

end

\[ \text{gives } |C| = 4 \quad (\text{min. is } 3) \]

Note that the alg. takes poly. time.

Let \( C = \text{vertex cover found by alg.} \)
\[ C_{opt} = \text{a min vertex cover} \]

claim \[ |C| \leq 2 \cdot |C_{opt}| \]
Pf. The set of edges you pick forms a matching $M$ (no 2 edges are incident).

$$|C| = 2|M|$$

Any vertex cover must have at least one vertex from each edge in a matching.

$$|M| \leq |C_{opt}|$$

Thus $|C| \leq 2|C_{opt}|$.

This alg. has approx. factor 2.
General TSP cannot be approximated to within constant factor in poly. time (unless P = NP).

Suppose we have a poly. time alg. for TSP that guarantees a tour of length \( \leq k \cdot \text{opt} \).

Claim: Then we can make a poly. time alg. for Hamiltonian cycle.

And hence P = NP.

Alg. for Ham. cycle:

Input: \( G = (V, E) \) \( |V| = n \)

Construct \( G' = (V, V \times V) \) — complete graph

\[ w(e) = \begin{cases} 1 & \text{if } e \in E \\ k \cdot n & \text{otherwise} \end{cases} \]

Run approx TSP alg. on \( G' \) to get tour, length \( l \)

if \( l \leq k \cdot n \) output YES (\( G \) has Ham. cycle)
else output NO

This is a poly. time alg.

Correctness:

In \( G' \), a tour that only uses edges of \( G \) has length \( n \).

a tour that uses at least one edge not in \( G \) has length \( \geq (n-1) + k \cdot n \) \( > k \cdot n \) (assuming \( n > 1 \))

Claim: \( l \leq k \cdot n \) iff \( G \) has Ham. cycle.

Pf. \( \Rightarrow \) \( l \leq k \cdot n \) \( \Rightarrow l = n \) so \( G \) has Ham. cycle

\( \Leftarrow \) \( G \) has Ham. cycle \( \Rightarrow G' \) has a tour of length \( n \)

\( \Rightarrow k \)-approx has length \( \leq k \cdot n \).