Approximation algorithms

**Def.** A factor $c$ approximation Algorithm for a minimization problem is a polynomial time alg. that computes, for any feasible input instance $I \in I$, a feasible solution with cost

$$\text{Alg}(I) \leq c \cdot \text{OPT}(I),$$

where $\text{OPT}(I)$ is the cost of an optimal solution to $I$.

We call $c$ the approximation factor.

**Theorem:** How to compare against an unknown opt. solution?
- Use lower bounds instead (LP relax., greedy…)

General approach: (min. problem)

1. Find $\text{LB}(I) \leq \text{OPT}(I), \forall I \in I$, and
2. Show $\text{Alg}(I) \leq c \cdot \text{LB}(I), \forall I \in I$.

Then Alg is a $c$-approx. Since $\text{Alg}(I) \leq c \cdot \text{OPT}(I), \forall I$.

**Outlook:**
- Combinatorial alg. analyzed using LP relaxation
- LP based algorithms + techniques, how to turn a fractional solution into good integral one.

But first: introductory example: TSP (purely combinatorial)
**Huristic TSP**

**Problem:** Given is a complete graph with non-negative edge cost, that satisfy the triangle inequality, i.e., for all \( u, v, w \in V \):

\[
C(u, v) \leq C(u, w) + C(w, v).
\]

Find a minimum cost cycle (tour) visiting each node exactly once.

**Algorithm [Christofides '76]**

1. Find a HST \( T \) in \( G \)
2. Double every edge of \( T \) to obtain a Eulerian graph
   
   **Recall:** \( G' \) is Eulerian (= every node in \( G' \) has node degree 2)
3. Find a Eulerian Tour \( \gamma \) on the doubled HST.
4. Output the tour that visits vertices of \( G \) in the order of their first appearance in \( \gamma \). (= take shortcuts)

**Ex.**

![Diagrams showing HST, doubled HST, Eulerian tour, and final tour.]

Thus, the algorithm is a factor 2 approximation for Huristic TSP.

**Pf:** Let \( \text{OPT}(I) \) be the length of an optimal tour for \( \text{HST}(I) \).

Removing one edge gives a spanning tree for \( I \).

\[
\text{OPT}(I) \geq \text{HST}(I) = \text{cost}(T).
\]

Obviously, \( \text{cost}(T) = 2 \cdot \text{cost}(T) \).

With the triangle inequality, the algorithm's final tour has length

\[
\text{Alg}(I) \leq \text{cost}(T) \leq 2 \cdot \text{OPT}(I).
\]
Analysis of this algorithm is tight (asymptotically).

**Example:**
- Complete graph $K_n$ with edge costs 1 and 2.
  - For $n = 6$:
    - Cost = 1
    - Cost = 2

- **Alg.**
  - Doubled HST
  - $\text{Alg} = (n-2) \cdot 2 + 2 \cdot 1 = 2n - 2$

- **Opt.**
  - $\text{Opt} = n - 1$

- $\frac{\text{Alg}(I)}{\text{Opt}(I)} \xrightarrow{m \to \infty} 2$

**Question:** Is there a better alg. for metric TSP? **Yes,**

- The above alg. finds Eulerian tour by doubling.
- Can we find a cheaper way from HST to Eulerian graph?
  - All nodes need even degree!
    - Worry only about nodes with odd degree in HST $= V'$.
  - Observe: $|V'|$ is even since the sum of all node degrees in HST is even. (Handshaking lemma)
  - Thus, HST + perfect matching on $V'$ give Eulerian graph.
Algorithm I [Christofides 76]

1. Find an MST $T$ on $G$
2. Compute a minimum cost perfect matching $H$ on the set of odd degree nodes of $T$. Add $H$ to $T$. $\rightarrow$ Eulerian
3. Find a Eulerian tour $T$ on this graph.
4. Output the tour that visits vertices of $G$ in the order of their first appearance in $T$.

The alg. is a factor $3/2$ approx. for the metric TSP.

Lemma (lower bound)

Let $V' \subseteq V$ s.t. $V'/V'$ is even and let $H$ be a min cost perfect matching on $V'$, Then $\text{cost}(H) \leq \text{OPT}/2$.

Proof.

Let $\text{OPT}' \leq \text{OPT}$ be the subtour of $\text{OPT}$ on $V'$ obtained by shrinking (Δ-inequality).

$\text{OPT}'$ is the union of 2 disjoint perfect matchings $H_1$ and $H_2$. Then $\min \left\{ \text{cost}(H_1), \text{cost}(H_2) \right\} \leq \frac{1}{2} \text{OPT}'$.

$\text{cost}(H) \leq \min \left\{ \text{cost}(H_1), \text{cost}(H_2) \right\} \leq \frac{1}{2} \text{OPT}.$

Proof of Thm.

$\text{cost}(T) \leq \text{cost}(T) + \text{cost}(H) \leq \frac{3}{2} \text{OPT}.$

Tight example: $n$ vertices, with $n$ being odd

This is the currently best known approx. alg. for metric TSP. It is conjectured -- but open -- if a $4/3$ exists.
**Question:** Has "good" can any poly. time alg. be? \\
\[\approx \text{lower bound on the approx. factor}\]

**Inapproximability**

- Example: gap reduction (general) TSP.

**Theorem.** [Gonzalez, Sahni '76]

For any polynomial-time computable function \(x(n)\), there is no \(x(n)\)-factor approx. alg., unless \(P = NP\).

**Pf.** by contradiction: Suppose there is an \(x(n)\)-approx. alg. \(A\). We show that it can be used to decide the Hamiltonian circuit problem, which is \(NP\)-hard. This would imply \(P = NP\).

| Given a graph, does there exist a tour that visits each node exactly once? | [NP-hard, Karp '72] |

**Ham. circuit problem**

\[G = (V, E)\] 

\[\text{construct} \quad G' \text{ is } G(V, E) \text{ with } \forall e \in E, \forall e \in E\]

\[\text{such that } G' \text{ is connected and } \sum_{e \in E} \text{ with cost } = x(n) \cdot n, \forall e \in E\]

**Obs.** If \(G\) has \(HC\), then \(\exists\) tour with cost \(= n\).

If \(G\) has no \(HC\), then cost of any tour \(> x(n) \cdot n\).

\[=\] An \(x(n)\)-approx. alg. \(A\) could decide \(HC\) problem in poly. time.

**Contradiction.**

\[\begin{array}{c|c|c}
\text{YES} & x(n)n & \text{NO} \\
\end{array}\]
Analysis based on LP relaxation:

The knapsack problem

\[
\begin{align*}
\max & \quad \sum_{j=1}^{n} c_j x_j \\
\text{subject to} & \quad \sum_{j=1}^{n} w_j x_j \leq K \\
& \quad x_j \in \{0,1\}
\end{align*}
\]

Alg:

0.) Remove all items with \( w_j > K \).
1.) Sort and remove items s.t. \( \frac{c_1}{w_1} \geq \frac{c_2}{w_2} \geq \ldots \geq \frac{c_n}{w_n} \).
2.) Let \( k = \max \{ j \mid j \in \{1,\ldots,n\}, \sum_{l=1}^{j} w_l \leq K^3 \} \).
3.) Choose the best of the two solutions

\( \{1,\ldots,k^3\} \) or \( \{k+1,\ldots,n\} \).

Thus, Alg. is a factor 2-approximation for knapsack.

Pf. Let \( \text{OPT} \) be cost of an optimal solution and \( 2^{\text{OPT}} \) the cost of the optimum to the LP-relaxation.

\[ \text{OPT} \leq 2^{\text{LP}}. \]

Prop. \[ 2^{\text{LP}} \leq \sum_{j=1}^{K^3} c_j. \] (Proved in homework.)

Then, our solution \[ \frac{1}{2} \left( \sum_{j=1}^{K^3} c_j + c_{K+1} \right) \geq \frac{1}{2} 2^{\text{LP}} \geq \frac{1}{2} \text{OPT}. \]