Approximate Solution for NP-hard problems

\[ \text{WANT} \rightarrow \text{HEURISTIC (Greedy)} \rightarrow \text{Solution} \rightarrow \text{How good is this solution?} \]

\[ \exists \text{bound} \leq \frac{|\text{HEURISTIC}|}{|\text{OPT}|} \leq \text{bound} \]

Ideally, too closely

Usually approximating an NP-hard problem in NP-hard!

Vertex Cover Problem — Stupid Algorithm!

\[ \Theta(n) \leq \frac{|\text{GREEDY}|}{|\text{OPT}|} \leq \Theta(n) \]
TSP Approximation

\[ \frac{|\text{Heuristic}|}{|\text{OPT}|} \leq \rho \]

[No]

TSP cannot be approximated to within any fixed ratio!

Then it is NP-hard to determine if TSP can be solved within \( \rho \) times optimal.

Proof:
Reduction from Hamiltonian Cycle.

H.C. problem \[\rightarrow\] approximate TSP with ratio \( \rho \)

(Greedy)

\[ \text{cost} \left( v_i \rightarrow v_j \right) = \begin{cases} 1 & v_i \rightarrow v_j \in E \\ \rho |V| + 1 & \text{o.w.} \end{cases} \]

Suppose \[ \frac{|\text{Heuristic}|}{|\text{OPT}|} \leq \rho \]

Then \( |V| \) edges in a H.C.

OPT

\[ |V| \]

\[ \rightarrow \]

\[ \frac{1}{\rho |V| + 1} \]

\[ |V| - 1 \] edges

\[ \geq (\rho + 1)|V| \]
Try to approximate the TSP w/ an inequality:

\[ \frac{|\text{Heuristic}|}{|\text{OPT}|} \leq \rho \]

\[ |\text{APPROX TOUR}| \leq 2 |\text{MST}| \]

\[ 2 |\text{MST}| \leq 2 |\text{OPT}| \]

\[ \frac{|\text{APPROX TOUR}|}{|\text{OPT}|} \leq 2 \]

**Christofides' Algorithm**

\[ \frac{|\text{Heuristic}|}{|\text{OPT}|} \leq \frac{3}{2} \]

\[ G = (V, E) \]

\[ \text{cost} \]

\[ \text{MST} \]

\[ \text{MIN of vertices of odd degree} \]

\[ \text{Eulerian Cycle} \]

\[ |\text{MST}| + |\text{MAT}| = |\text{MST}| + |\text{MAT}| \geq |\text{OPT}| + \frac{1}{2} |\text{OPT}| \]

\[ \Rightarrow \frac{1}{2} |\text{MST}| + \frac{1}{2} |\text{MST}| = \frac{1}{2} |\text{OPT}| \]
Nearest Neighbor

\[ \Theta(n^2) \leq \frac{|\text{NNT}|}{10^{\text{OPT}}} \leq \frac{1}{2} \frac{2^n}{n!} \]

\[ |\text{OPT}| \geq 2 \sum_{i=1}^{n} l_i \]

\[ |\text{OPT}| \geq 2 \sum_{i=h=1}^{n} l_i \]

\[ \geq 2 |\text{NNT}| \]
Let $t_j$ be any term of $x_i, x_{2} \ldots x_{j}$

$|O \cap T| \geq |T_j| \quad \forall j$

$|O \cap T| \geq |T_{2h}|$

(1) $\frac{k_i}{k_j} \leq c_j \in T_{2h}$

(2) $c_{i,j} \leq \min\{k_i, k_j\}$

$|O \cap T| \geq |T_{2h}| = \sum_{T_{2h}} c_{i,j} \geq \sum_{T_{2h}} \min\{k_i, k_j\} \geq 2\sum_{T_{2h}} c_{i,j}$
Sec. 4.3  Approximation to Exhaustive Search

(a) Initially $c_1$ is the only city in the closest insertion tour; the edges shown are those of $O_n - \{f_{\text{max}}\}$.

(b) Later in the algorithm the configuration consists of the closest insertion tour (in boldface) and certain edges from $O_n - \{f_{\text{max}}\}$ (in lightface).

Figure 4.14 Examples of the spiderlike configuration in the closest insertion algorithm.

Suppose that city $c_k$ is to be inserted in the closest insertion tour, $c_k$ being closest to city $c_m$ on the tour, as illustrated in Figure 4.15. The cost of inserting $c_k$ between $c_m$ and an adjacent city, say $c_l$, is $C_{cxm} + C_{cmx} - C_{cml}$. Let $c_{r}$ be the point on the tour to which the leg containing $c_k$ is attached and let $c_{r}c_{l}$ be the first edge of that leg (it is possible that $c_{l} = c_k$). Since the algorithm chooses the city closest to the tour to insert, we have

$$C_{cxm} \leq C_{cxr}, \quad (4.7)$$

and by the triangle inequality

$$C_{cxl} \leq C_{cxm} + C_{cmx}. \quad \text{Using symmetry, we can combine these to obtain} \quad C_{cxl} \leq C_{cxm} + C_{cmx}. \quad \text{(4.8)}$$

Adding (4.7) and (4.8) gives

$$C_{cxm} + C_{cmx} \leq C_{cxm} + 2C_{cxr},$$

which is equivalent to

$$C_{cxm} + C_{cmx} - C_{cxm} \leq 2C_{cxr}.$$
Figure 4.15 The spiderlike configuration as $c_k$ is about to be added to the closest insertion tour (shown in boldface). The cost of inserting $c_k$ between $c_i$ and $c_m$ is $C_{cm} + C_{cm} - C_{cm}$, which is at most $2C_{cm}$. After $c_k$ is inserted, the edge $c_{ck}$ is deleted.

Consequently, inserting $c_k$ between $c_i$ and $c_m$ costs at most $2C_{cm}$. After $c_k$ is inserted, the edge $c_{ci}$ is deleted from the configuration (which thus remains spiderlike) and the algorithm continues.

Thus each city $c_k$, $k = 2, \ldots, n$, corresponds to a unique edge $c_i - c_j$ of $O_n - \{l_{\text{max}}\}$ in such a way that the insertion of $c_k$ costs at most twice the length of the corresponding edge in $O_n - \{l_{\text{max}}\}$. Since the cost of $L_n$ is the sum of the costs of the insertions, we have

$$|L_n| \leq 2(|O_n| - |l_{\text{max}}|)$$

and the theorem follows.

So we know that the closest insertion algorithm always produces tours that cost at most twice the cost of an optimal tour. Moreover, we can show that there is an $n$-city problem, $n \geq 6$, for which the closest insertion algorithm produces a tour whose length is almost twice that of an optimal tour. Consider the cost matrix $C$ defined by

$$C_{ij} = C_{ji} = \min (j - i, n - j + i), \quad i \leq j.$$ 

This matrix corresponds to having $C_{ij} = C_{ji} = 1$ for $j = i + 1 \mod n$ and making $C_{ij} = C_{ji}$ the length of the shortest path from $i$ to $j$ following arcs of the type $k \to (k + 1) \mod n$. The configuration, and the closest insertion tour, is shown in Figure 4.15.