Exercises on

NP-Hardness and Approximation Algorithms

Exercise 1

Let $X$ be a set variables and $F \subseteq \{\{\lambda_1, \lambda_2\} | \lambda_i \in (X \times \{0, 1\})\}$ a set of clauses on $X$ with two literals each. Consider the following digraph $G(F) := (X \times \{0, 1\}, \{(\lambda_1, \lambda_2) | (\lambda_1, \lambda_2) \in F\})$.

(a) Show that $\bigwedge F$ is not satisfiable if and only if there is a variable $x \in X$, such that $x$ and $\bar{x}$ are in the same strongly connected component of $G(F)$.

(b) Determine the complexity status of the decision problem $2SAT$!

Exercise 2

The optimization problem MAXCUT reads as follows:
Given an undirected graph, $G$, with a cost function $c : E(G) \to \mathbb{N}$. Determine a cut of maximum weight.

Formulate a suitable decision problem and determine its complexity roughly. (Hint: Reduce 2PARTITION!)

Exercise 3

Show that 3COLOURING is NP-complete. The decision problem 3COLOURING is to decide for a given graph $G$, whether there is a three valued function $f : V(G) \to \{0, 1, 2\}$ for which $f(v) \neq f(w) \ \forall \{v, w\} \in E(G)$ holds.

Hint: Reduce 3SAT! Start with a central gadget consisting of one central vertex that for each variable $x$ forms a triangle with two other vertices, reasonably called $x$ and $\bar{x}$. Now, construct gadgets for each clause.

Exercise 4

Come up with an example, that fools the nearest neighbour heuristic for the metric TSP.

Exercise 5

In the metric TSP, the points to be visited are embedded in $\mathbb{R}^2$ and the distances are just the euclidean ones (with sufficient precision for our purpose).

(a) Show that a solution of the metric TSP can not be optimal if two chosen edges cross.

(b) How can you improve upon this solution?

(c) Does this yield a polynomial algorithm?

Exercise 6

Let $G$ be an undirected, connected graph with a weight function $c : E(G) \to \mathbb{R}^+$ on the edges, and $T \subseteq V(G)$. A Steiner Tree in $G$ for $T$ is a subgraph, $S$, of $G$ such that $S$ is a tree and $T \subseteq V(S)$, in other words: $S$ is a tree, such that $T \subseteq V(S) \subseteq V(G)$, $E(S) \subseteq E(G)$. For an instance of a Steiner Tree problem, $(G, c, T)$ we are looking for the minimum Steiner Tree in $G$ for $T$.

(a) In general this problem is NP-complete (even MAXSNP-hard). Which special cases are elements of P.
(b) Let \((\bar{G}, \bar{c})\) be the metric closure of \((G, c)\), i.e. \(\bar{G}\) is the complete graph on \(V(G)\) and \(\bar{c}(\{v, w\})\)
equals the lenght of a shortest path between \(v\) and \(w\) in \(G\). (Note that it is possible for an edge \(e \in E(G)\) that \(c(e) > \bar{c}(e)\).)

Let \(T \in V(G)\), \(S\) an optimal Steiner Tree for \((G, c, T)\) and \(M\) be a minimum spanning Tree in the subgraph of \(\bar{G}\) induced by \(T\) (where the weight function is the restriction of \(\bar{c}\)). Show that \(\bar{c}(E(M)) \geq 2c(E(S))\)!

(c) Derive a constant factor approximation for the Steiner Tree problem!

**Exercise 7**

Let \(G\) be a graph. We are looking for subsets, \(M\), of \(E(G)\) such that all distinct pairs of its elements intersect trivially, i.e. \(\forall a \neq b \in M : a \cap b = \emptyset\). As always such a set is called maximal, if it cannot be extended, whereas it is a maximum set of its kind, if no other such set has a strictly higher cardinality. Let \(M\) and \(M'\) both be maximal such sets.

(a) Show that \(|M| \leq 2 \cdot |M'|\).

(b) Develope an constant factor approximation algorithm for the problem of finding a maximum set of that kind.

**Exercise 8**

We are trying to find a maximum clique.

(a) Show that a Greedy Algorithm is arbitrarily bad.

(b) Can you observe a pattern in your examples that limits the badness of the Greedy Algorithm a priori?

(c) Show that the problem to decide whether a graph has clique of size \(k\) is \(\text{NP}\)-complete.

(d) What does it change, if \(k\) is not part of the input?

(e) Show that minimum vertex cover is \(\text{NP}\)-hard.

**Exercise 9**

Show that the constraint shortest path problem is \(\text{NP}\)-hard.