Prof. Dr. Sándor P. Fekete
Peter Kramer
Due June 10^{th}
Discussion June 17^{th}

Please submit your handwritten answers in pairs, *using the box in front of* `IZ338` *before 1pm on the due date above. Make sure to include your* full names, matriculation numbers, *and the* study programmes *that you are enrolled in (e.g., computer science master).*

This sheet concerns itself with the TRAVELING SALESPERSON PROBLEM, which asks for a cheapest tour that visits each vertex of a graph exactly once. In the lecture, we focused on complete graphs, that is, we were given the ability to directly move from any city to any other city without visiting any other city in between.

Exercise 1 (Graph TSP). (5 points)

In this exercise, we consider TSP on weighted graphs (GRAPH TSP), where a tour may not necessarily exist. We are given an n-vertex graph $G = (V, E)$ with non-negative edge costs $c(e) : E \to \mathbb{R}_{\geq 0}$ and have to compute a tour T, i.e., a sequence $T = v_1 v_2 \dots v_n v_{n+1}$ that contains all n vertices with $v_{n+1} = v_1$ and $\{v_i, v_{i+1}\} \in E$ for all $1 \leq i \leq n$, such that the sum of edge weights $\sum_{i=1}^{n} c(\{v_i, v_{i+1}\})$ is minimized.

Give a polynomial-time reduction from GRAPH TSP to TSP on complete graphs. That is, give a polynomial-time algorithm that transforms an instance $\mathcal{I}$ of GRAPH TSP to an instance $\mathcal{J}$ of the TSP such that you can transform the optimal solution of $\mathcal{J}$ to an optimal solution of $\mathcal{I}$ or decide that no such solution exists in polynomial time.

Exercise 2 (Hardness and inapproximability). (3+4+8 points)

In the previous exercise, we showed that the TSP is still hard even if the existence of a tour is guaranteed and finding a valid tour is trivial. GRAPH TSP is also still hard if the edge weights are set to $c(e) = 1$ for all $e \in E$; the problem is then also known as HAMILTONIAN CYCLE.

In this exercise, we consider the other assumption that we made in the lecture: we assumed that the edge weights satisfy the triangle inequality, i.e., that for any three vertices u, v, w, we have $c(\{u, v\}) + c(\{v, w\}) \geq c(\{u, w\})$. TSP on complete graphs with this additional restriction is also called METRIC TSP.

a) Show that the edge weights of any complete graph that only uses edge weights 1 and 2 satisfy the triangle inequality.

b) By a reduction from HAMILTONIAN CYCLE, show that the TSP is still NP-hard on complete graphs with edge weights that satisfy the triangle inequality. (Hint: Use your result from a))

c) Let $f(n) : \mathbb{N} \to \mathbb{Q}$ be any polynomial-time computable function (this also implies that the output of $f(n)$ can be encoded in poly(n) bits). Assuming $\mathsf{P} \neq \mathsf{NP}$, show that there is no polynomial-time $f(n)$-approximation algorithm for the TSP on complete graphs that do not have to satisfy the triangle inequality.

Exercise 3 (Christofides' Algorithm). **(3+7 points)**

In the lecture, we discussed a $3/2$-approximation algorithm for METRIC TSP due to Christofides. The algorithm computes an MST on V and a minimum-weight perfect matching (MWPM) M on the (even number of) vertices with odd degree in that MST. It then adds the matching edges to the MST, resulting in a Eulerian graph U, computes a Eulerian walk on U and constructs a tour by skipping repeated vertices in that walk using the triangle inequality.

We claimed that the total edge weight of U was at most $\frac{3}{2}$ OPT, where OPT denotes the weight of an optimal tour. In particular, we claimed that the weight of M is at most $\frac{1}{2}$ OPT.

Let $G = (V, E)$ be a graph with an even number of vertices and non-negative edge weights satisfying the triangle inequality.

a) Prove or disprove: For any $V' \subseteq V$ with even cardinality, the weight of a minimum-weight perfect matching on V' is upper-bounded by the weight of an MWPM on V.

b) Prove: For any $V' \subseteq V$ with evenly many vertices, the weight of a MWPM on V' is at most half the cost of an optimal TSP tour on V.