Discrete Optimization 2010

Hand-in Assignments (due Nov. 29, 2010)

Marc Uetz (m.uetz@utwente.nl), Applied Mathematics, University of Twente

The background literature on $\mathcal{NP}$-completeness is Chapter 9 of of [2] and Chapter 1 of [1], contained in the reader. The reader also contains a chapter on approximation algorithms.

1. (5 points) Which of the following is true or false, assuming that $\mathcal{P} \neq \mathcal{NP}$. Give succinct(!) but precise explanations, or (counter)examples.

   (a) Maximum Flow polynomially reduces to Shortest Path.

   (b) Let us assume $k \in \mathbb{N}$ is given and fixed. The existence of a polynomial time algorithm that decides if any given graph G has a clique of size k would imply $\mathcal{P}$=$\mathcal{NP}$.

   (c) Minimum Spanning Tree polynomially transforms to CLIQUE.

   (d) The class $\mathcal{NP}$ are exactly the problems that do not have polynomial time algorithms.

   (e) As Knapsack has a pseudo polynomial time algorithm, but SAT doesn't, there cannot be a polynomial time transformation from SAT to Knapsack.

2. (4 points) The symmetric TSP problem in its decision version is this: Given is a complete, edge-weighted graph on n nodes, and given a number k, does there exist a tour of total length at most k?

   Show that the symmetric TSP problem is strongly $\mathcal{NP}$-hard.

   (Hint: Use the fact that the problem Hamiltonian Cycle in undirected graphs is $\mathcal{NP}$-complete.)

3. (5 points) Recall the Knapsack problem: There are n items $V = \{1, \ldots, n\}$, with weights w_j and values v_j, and a knapsack of capacity W. Assume w.l.o.g. $w_j \leq W$ for all $j = 1, \ldots, n$, and assume w.l.o.g. the items are ordered such that

   $$\frac{v_1}{w_1} \geq \frac{v_2}{w_2} \geq \cdots \geq \frac{v_n}{w_n}\,.$$

   Consider the following algorithm H: Either take the item with maximal value $v_{\max} = \max_j v_j$, or take the maximal subset $[k] := \{1, \ldots, k\}$ that still fits in the knapsack (i.e., $\sum_{j=1}^{k} w_j \leq W$), whatever of the two solutions yields the maximal value.

   Show that algorithm H is a $\frac{1}{2}$-approximation algorithm for Knapsack.

References

[1] V. V. Vazirani, *Approximation Algorithms*, Springer, 2001

[2] Cook, W. J., W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*, Wiley-Interscience, 1998.