Exercise 2: June 02

Lecturer: Merav Parter

Distance Oracles

This section is devoted for the Thorup and Zwick distance oracles that we have studied in depth in class. We already saw that these oracles are quite "flexible" and can be extended to related settings of labeling and routing schemes. In this exercise, we will reveal another useful extension of these oracles.

Exercise 1. The distance oracle presented in class supports distance queries u, v for any $u, v \in V \times V$. Our goal is to construct a *smaller* oracle of size $o(n^{1+1/k})$ that would handle only a subset of query pairs. In particular, you are now given a subset of vertices, called hereafter, *sources*, $S \subseteq V$ and it is required to design a *source-wise* approximate distance oracle scheme. In such a scheme, the query algorithm receives only queries of the form $u, v \in S \times V$ (i.e., your oracle should only answer distance queries between one of the sources and some other vertex in the graph). Show that in such a case, the preprocessing algorithm can be adapted to yield a data structure of size $\widetilde{O}(k|S|^{1/k} \cdot n)$. Your answer should include a modified preprocessing algorithm, query algorithm and a correctness analysis.

Routing Schemes

Exercise 2. Describe an efficient routing scheme for the unweighted $\sqrt{n} \times \sqrt{n}$ 2-dimensional grid. The labels and the routing tables should be of size $O(\log n)$ bits. Bonus: extend it to the d-dimensional n-vertex hypercube for $n = 2^d$.

Labeling Schemes

Exercise 3. Let T be a weighted tree with weights in $[-W, W]$. Show an $O(\log^2 n + \log n \log W)$ labeling scheme for computing the largest edge weight on the tree path between u and v. I.e., such that given $L(u)$ and $L(v)$, one can compute the weight of the heaviest edge on the path between u and v in the tree.