Anna Adamaszek, Andreas Wiese
Tutorials: Marvin Künnemann

Summer 2014

Assignment 2 for
Approximation Algorithms and Hardness of Approximation

Discussion:
Thursday, 8 May 2014, 14 pm

Assignment 1 *(PTAS for Knapsack)*

In the last tutorial, we discussed the following two observations about GreedyKnapsack:

1. If $p_i \leq \varepsilon\mathrm{OPT}$ for all i, GreedyKnapsack gives a $(1-\varepsilon)$-approximation.
2. There are at most $\lceil \frac{1}{\varepsilon} \rceil$ items with profit at least $\varepsilon\mathrm{OPT}$ in any optimal solution.

Show how to compute a $(1-\varepsilon)$-approximation in time $O(n^{\lceil \frac{1}{\varepsilon} \rceil + 1})$.

Try out all possibilities for $\lceil \frac{1}{\varepsilon} \rceil$ "large" items and use GreedyKnapsack.

Assignment 2 *(Tightness of TSP approximation algorithms)*

Find examples for which

1. MetricTSPviaMST does not find a α-approximation for any constant $\alpha < 2$.
2. MetricTSPviaMSTandMatching does not find a α-approximation for any constant $\alpha < \frac{3}{2}$.

Assignment 3 *(Asymmetric TSP)*

We are given a directed graph G with edge costs $d : V \times V \to \mathbb{R}_{\geq 0}$ that satisfy the directed triangle inequality, i.e., $d(u,v) \leq d(u,w) + d(w,v)$ for all $u, v, w \in V$. Give a $O(\log n)$-approximation for the problem of finding a shortest tour visiting all vertices.

Use the minimum-cost cycle cover. Shrink the cycles and recurse.

Assignment 4 *(Approximating maximum compression)*

Let X be a set of strings and let $\|X\|$ denote the sum of lengths of the strings in X. Consider the *maximum compression* problem: find a superstring s of all strings in X that maximizes $\|X\| - |s|$. Call such an optimal string s^*. Show that in polynomial time, one can compute a $\frac{1}{2}$-approximation, i.e., a superstring s of all strings in X with compression

$$\|X\| - |s| \geq \frac{1}{2} \left(\|X\| - |s^*| \right).$$

Hint: Reduce this problem to finding a longest traveling salesman path in a suitably constructed graph. Then use an approximation algorithm for maxTSP.

Open problem

Consider the greedy algorithm for the shortest superstring problem: Take two strings with maximum overlap and replace them by overlapping them as much as possible. Repeat until a single string is left. Does this algorithm achieve a 2-approximation?