Thomas Rothvoß
Location: ELD120 Discussion: 24.03.10

Exercises

Approximation Algorithms

Spring 2010

Sheet 4

Exercise 1
Consider again the MINCONGESTION problem, where a directed graph $G = (V,E)$ with demand pairs (s_i,t_i) for $i = 1,\dots,k$ is given and one aims at finding s_i-t_i paths P_i that minimize the congestion $\max_{e\in E} |\{i : e \in P_i\}|$. Show that the algorithm presented in the lecture gives a $O(1)$-approximation (with high probability) if $k \geq (\log_2 |E|)\cdot|E|$ (and $s_i \neq t_i$ for all $i = 1,\dots,k$).
Hint: Which lower bound on the fractional congestion from the linear program do you obtain using the additional assumption $k \geq (\log_2 |E|)\cdot|E|$?

Exercise 2
Suppose we have 3 machines and n jobs. If we run job j on machine $i \in \{1,2,3\}$ this takes a *processing time* of $p_{ij} \in \mathbb{Q}_+$. The MINIMUMMAKESPAN problem is to find a way to assign the jobs to machines such that the load of the highest loaded machine (the *makespan*) is minimized. Formally

$$OPT = \min_{J_1 \dot\cup J_2 \dot\cup J_3 = \{1,\dots,n\}} \left\{ \max_{i=1,\dots,3} \left\{ \sum_{j\in J_i} p_{ij} \right\} \right\}$$

Design an FPTAS for this problem (and prove an approximation guarantee of $1+\varepsilon$).
Hint: Similar to the KNAPSACK FPTAS it is a good idea to first round the running times in a suitable way. Then apply dynamic programming.