Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Harald Räcke
Chintan Shah

Wintersemester 2011
Übungsblatt 11
26 January 2012

Effiziente Algorithmen und Datenstrukturen I

Aufgabe 1 (10 Punkte)

Give an upper bound on the number of distinct global minimum cuts a graph with n vertices could have. Prove that your bound is tight by demonstrating a family of graphs which realizes as many distinct minimum cuts as suggested by your bound.

Aufgabe 2 (10 Punkte)

Suppose we prematurely stop the randomized min-cut algorithm 'KargerMincut' after iteration $n - t$ i.e., when there are t vertices left. With some probability, a minimum cut survives these $n - t$ iterations (refer lecture slides). Now, we find the minimum cut in G_t in time $O(t^4)$ as suggested in the lecture. In our algorithm, we would then repeat this procedure (how many times?) to ensure that we find a global minimum cut with high probability. Find the optimal value of t so as to minimize the running time of this algorithm.

Aufgabe 3 (10 Punkte)

We know how to find the global minimum cut in a graph G with n vertices and m edges in $O(n^4)$ time by computing n minimum $s - t$ cuts (using the push-relabel algorithm). Observe that the minimum degree of any vertex in G is an upper bound on the value of a global minimum cut. Suggest how to find a global minimum cut for an unweighted graph in $O(m^2)$ time by computing n minimum $s - t$ cuts. Choose the source s carefully and use Ford-Fulkerson's algorithm instead of the push-relabel algorithm.

Aufgabe 4 (10 Punkte)

Give an example of a graph $G = (V, E, c)$ and a tree $T = (V, F, w)$ such that T is flow-equivalent to G, but does not satisfy the cut property.

Aufgabe 5 (5 Punkte)

(*Note:*Attempt this question iff your marks in previous assignments are below the required threshold of 40%)

The Randomized Mincut Algorithm returns a given (global) mincut with probability $\frac{1}{\binom{n}{2}}$. If there is more than one mincut in the graph, the probability that Randomized Mincut returns some mincut would increase. Keeping this in mind, suggest a family of graphs for which Randomized Mincut returns a mincut with probability 1.