Issue Date: 08.12.2014
Version: 2014-12-08 19:51

6th Exercise Sheet for Kombinatorische Algorithmen, WS 14/15

Hand In: *Until Monday, 15.12.2014, 12:00,*
deliver or email to Raphael (`reitzig@cs.uni-kl.de`).

Problem 10

2 + Bonus points

The Max-Flow-Min-Cut theorem relates the flow values of maximal flows and the capacities of minimal cuts; the optimal solution *values* of both problems coincide. However, additional work is needed to compute the actual *solutions* — and we only considered flow algorithms in class.

a) Design an algorithm for computing a minimum-capacity s-t-cut in the network $G = (V, E, c)$ from a maximum flow f^* in G.

b) Assume you are given a network $G = (V, E, c)$ and a minimum-capacity s-t-cut. Can you use the cut for determining a maximum flow f^* in G faster than solving Max-Flow from scratch?

Problem 11

5 points

The augmenting-path approach is related to the *primal* Simplex algorithm[1] for solving linear programs (LP): we start with and maintain a feasible solution, which is initially suboptimal. This solution is successively improved until we reach optimality.

In this exercise, we show that the preflow-push approach resembles the *dual* simplex method: we maintain a *dually* feasible solution, which in our case is an s-t-cut. This dual solution is modified until it becomes (primally) feasible; here until the preflow becomes a flow.

The original statement of the preflow-push algorithm does not explicitly maintain this cut, therefore we augment the algorithm as follows.

[1] For details on LPs and the Simplex algorithm(s) see any textbook on linear optimization, e. g. Hamacher and Klamroth [1]

We maintain a set S of nodes throughout the algorithm, which is initially $S := \{s\}$. The algorithm then invokes a sequence of relabel and push operations. Each push moves some amount of flow over an edge $(u, v) \in E_f$ of the *residual* network G_f. After each such push-operation, we now check whether $u \notin S$ and $v \in S$, i. e. whether the push was"into S". If yes, we add to S all nodes reachable from u in G_f.

Show the following invariants:

(i) At any time is $(S, V \setminus S)$ a s-t-cut.

(ii) The capacity of the cut $(S, V \setminus S)$ never increases during the run of the algorithm.

(iii) If f fulfills the flow conservation property then its value $v(f)$ equals the capacity $c(S, V \setminus S)$ of the cut.

References

[1] Horst W. Hamacher and Kathrin Klamroth. *Lineare und Netzwerk-Optimierung / Linear and Network-Optimization. Ein bilinguales Lehrbuch. A bilingual textbook.* Wiesbaden: Vieweg+Teubner Verlag, 2000. ISBN: 978-3-528-03155-8. DOI: 10.1007/978-3-322-91579-5.