Combinatorial optimization 1
Winter term 2016/2017
4th working sheet

29. Apply the Push-Relabel algorithm to determine a maximum s-t-flow in the network represented in Figure 1. The numbers next to the edges are the capacities. Apply the following two variants of the algorithm:

    (a) The active vertices are maintained in a FIFO list ("First In First Out"): new vertices which become active are added at the end of the list and the first vertex in the list is selected to perform the next push or relabel operation. For every active vertex v the algorithm maintains also the FIFO list A_v of the edges starting at v in G_f. The next push operation involving vertex v is performed on the first admissible edge from the list A_v.

    (b) The active vertex with the maximum distance label is selected for the next push or relabel operation. The choice of the edge on which the push operation for the selected active vertex v is performed is done as in (a).

    Compare the number of relabel operations, as well as saturating and non-saturating push operations in both variants of the algorithm, respectively.

30. Determine the MA order in the graph G obtained from the network in Figure 1 by ignoring the directions of the edges. Then use that order to determine a minimum capacity cut in G.

31. Let (G, u, s, t) be a network with integral capacities $u(e)$, for all $e \in E(G)$. We search for a minimum s-t-cut with the smallest number of edges in (G, u, s, t)- Show that a mimimum s-t-cut in $(G, \bar{u}, s, t)$ with $\bar{u}(e) = |E(G)|u(e) + 1$, $\forall e \in E(G)$, is the required minimum s-t-cut with the smallest number of edges in (G, u, s, t).

32. Let λ_{ij} with $1 \le i, j \le n$, $i \ne j$, be nonnegative numbers with $\lambda_{ij} = \lambda_{ji}$ and $\lambda_{ik} \ge \min\{\lambda_{ij}, \lambda_{jk}\}$ for pairwise different indices $i, j, k \in \{1, 2, \ldots, n\}$. Show that there exists a graph G with $V(G) = \{1, 2, \ldots, n\}$ and capacities $u \colon E(G) \to \mathbb{R}_+$, such that λ_{ij}, $i, j \in \{1, 2, \ldots, n\}$ are the local connectivity numbers in G.

    Hint: Consider a spanning tree with maximum weight in the complete graph K_n with edge weights c given by $c((i, j)) := \lambda_{ij}$, for all $1 \le i, j \le n$, $i \ne j$.

33. The manager of a restaurant faces the followig problem. He knows that d_i napkins wll be needed for day i of the next week ($i = 1, \ldots, 7$). Every morning he can buy new napkins at a Euro per piece. Moreover used napkins can be brought to the loundry for cleaning at the evening every day. There the fast service or the normal service can be chosen at the price of b or c Euro per napkin, respectively. In the case of the fast service the cleaned napkins are available at the mornig of the next day, and in the case of the normal service the cleaned napkins are available at the morning of the second next day. The manager wants to decide how many napkins to buy every morning and how many napkins to bring to the loundry for a fast and a normal service, respectively, such that the overall costs related to the napkins are minimized. Model this problem as a minimum cost flow problem.

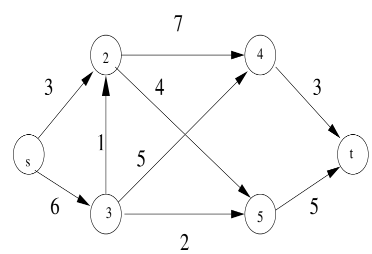

Figure 1: Network for task no. 29 and 30