Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

SS 2007 Prof. R. Wattenhofer / Prof. P. Widmayer / Prof. S. Suri / T. Locher / Y. A. Oswald

Principles of Distributed Computing
Exercise 10: Sample Solution

1 Minimum Cut with Fewest Arcs

Consider two mincuts in G, one with k_1 edges, the other with k_2 edges and both with capacity C^*. Using the transformed capacities, observe that the first cut has capacity $mC^* + k_1$, the second has capacity $mC^* + k_2$; so if $k_1 < k_2$, then k_1 wins.

Consequently the winninig cut among all min cuts in G is the one with fewest edges. Next, suppose there is a sub-optimal cut C, with $C > C^*$, but fewer edges $k < k_1$.

The value of this cut is

$$\begin{aligned} mC + k &\geq m(C^* + 1) + k && \text{/* because } C \geq C^* + 1 \\ &> mC^* + k1 && \text{/* because } k_1 < m \end{aligned}$$

Thus, no suboptimal cut beats any optimal cut.

2 Maximum Flow Reduction Algorithm

Compute the mincut C^* in G. If you delete k edges from this min cut, the flow shrinks by k units.

Proof: Focus on the cut C^*. Initially, this cut has capacity (=number of edges) $|C^*|$. If we delete k edges from it, the cut now has capacity $|C^*| - k$. By the flow capaciy lemma, the max permissible flow through this cut is at most $|C^*| - k$. Because each edge has capacity 1, the deletion of k edges can reduce the flow by at most k, so this is optimal.