University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
P. Schneider

Advanced Algorithms
Sample Solution Problem Set 4

Issued: Friday May 17, 2019

Exercise 1: Multicast Routing

For the *Multicast Routing Problem* we are given a graph $G = (V, E, c)$ with edge capacities $c : E \to \mathbb{R}_{\geq 0}$ and multi-cast groups $M_i \subseteq V$ with requirements r_i. We need to output a collection of trees $\mathcal{P} := \bigcup P_i$, where P_i is a tree which spans M_i whereas each edge has to reserve capacity r_i for each tree P_i that uses this edge. That means, we seek a set of trees $\bigcup P_i$, such that the maximal congestion: $\max_{e\in E} \frac{1}{c_i} \sum_{i:e\in P_i} r_i$ is minimized. Show that an $O(\log n)$ approximation to this problem can be computed efficiently and w.h.p.

Sample Solution

First we compute the tree decomposition $T_1, \dots, T_k$ from the lecture, i.e. for which there exist $\lambda_1, \dots, \lambda_k$ such that $\sum_{i=1}^k \lambda_i = 1$ and $\beta = \max_{e\in E} \left\{ \sum_{i=1}^k \lambda_i \cdot \text{rload}_{T_i}(e) \right\}$ is minimized. In the lecture we showed that when we are given multi commodity flows f_i on T_i, each of which causes congestion at most C on T_i, then mapping a linear combination $\sum_{i=1}^k \lambda_i f_i$ of commodity flows to G causes congestion at most $\beta C \in O(C \log n)$ on G.
Next, we sample a tree $T \in \{T_1, \dots, T_k\}$ according to the probability distribution $\lambda_1, \dots, \lambda_k$. From T we define a broadcast tree $\mathcal{P}$ that connects the terminals in M_i as follows

$$P_{i,T} := \bigcup_{(t_1,t_2)\in M_i^2} P_T(t_1, t_2), \quad P_i := \bigcup_{\{u,v\}\in P_i^T} P_G(u, v), \quad \mathcal{P}_T := \bigcup P_{i,T}, \quad \mathcal{P} := \bigcup P_i.$$

Note that $\mathcal{P}_T$ is an optimal solution for connecting all multicast groups M_i in the tree T in terms of congestion. Assume $\mathcal{P}^* = \bigcup P_i^*$ is the optimal solution in G, meaning that all terminals in the M_i are connected and the maximum congestion on the edges of G is minimized. We can project a tree $\mathcal{P}^*$ back to T as follows:

$$P_{i,T}^* := \bigcup_{\{u,v\}\in P_i^*} P_T(u, v), \quad \mathcal{P}_T^* := \bigcup_i P_{i,T}^*.$$

Let $C_T(\mathcal{P})$ and $C_G(\mathcal{P})$ be the maximum congestion for a set of solution trees $\mathcal{P} = \bigcup P_i$ (from T or G respectively). Then we have

$$C_T(\mathcal{P}_T) \overset{\mathcal{P}_T \text{ opt. on } T}{\leq} C_T(\mathcal{P}_T^*) \overset{\text{“}T\text{ has at least capacity of }G\text{”}}{\leq} C_G(\mathcal{P}^*).$$

The second inequality was shown in the context of flows in the lecture. We can consider the congestion caused by some set of trees $\mathcal{P} = \bigcup_i P$ in the context of flows as well, by defining a flow $f = \sum f_i$ where f_i is the edge-wise flow of size r_i on the edges of P_i. Figuratively speaking this is due to the fact that T has at least the same total capacity on each cut as G.
Now we define a multi commodity flow problem on G as follows. For each edge $\{u, v\} \in E$ define the edge-wise requirements

$$r(u, v) := \sum_i \sum_{\{u,v\}\in P_i^*} r_i.$$

The tree $\mathcal{P} = \bigcup_i P_i$ can be interpreted as a flow $f = \sum_i f_i$ on G with $f_i(u,v) = r_i$ for each $\{u,v\} \in P_i$. Analogously $\mathcal{P}_T = \bigcup_i P_{i,T}$ defines a flow $f_T = \sum_i f_{i,T}$. Then f_T gives a valid solution for the above commodity flow problem (projected to T). The congestion caused by the flow f_T (on T or G) equals the congestion of $\mathcal{P}$ (in terms of our multicast definition). And from the lecture we know that if this flow f_T is mapped back to G, the resulting congestion is (in expectation) at most $\beta \in O(\log n)$ times bigger then it was on T:

$$\mathbb{E}\big(C_G(\mathcal{P})\big) = \mathbb{E}\big(C_G(f_T)\big) \overset{\text{lecture}}{\leq} \beta \cdot C_T(f_T) = \beta \cdot C_T(\mathcal{P}_T) \overset{\text{from above}}{\leq} \beta \cdot C_G(\mathcal{P}^*).$$

With a Markov bound we get that the probability that our approximation factor is worse than 2β is at most $1/2$. Repeating the whole procedure above $\ln n$ times and taking best set of trees $\mathcal{P}$, we obtain a $2\beta \in O(\log n)$ approximation, w.h.p.

Exercise 2: Minimum Bisection Problem

Let $G = (V, E, c)$ be a graph with an even number of nodes $|V|$ and edge capacities $c : E \to \mathbb{R}_{\geq 0}$. In the *Minimum Bisection Problem* we are asking for a partition of vertices into two *equally* sized sets (B, W) (black and white) with minimal cut (sum of edge capacities between B and W). Give an efficient approximation algorithm for the problem, using the tree decomposition designed for multi commodity flow approximation.

Hint: You can use that the leaves of trees can be efficiently and optimally bisected.

Sample Solution

Let $T_1, \ldots, T_k$ be the tree decomposition from the lecture, i.e. for which there exist $\lambda_1, \ldots, \lambda_k$ such that $\sum_{i=1}^k \lambda_i = 1$ and $\beta = \max_{e \in E} \big\{ \sum_{i=1}^k \lambda_i \cdot \text{rload}_{T_i}(e) \big\}$ is minimized.
For some bisection (B, W) let $c_G(B, W)$ be the size of the cut in G and let $c_T(B, W)$ be the size of the minimal cut that separates the leaves B from the set of leaves W in the decomposition tree $T \in \{T_1, \ldots, T_k\}$. Let (B^*, W^*) and (B_i^*, W_i^*), $i \in \{1, \ldots, k\}$ be the solutions that minimize $c_G(B^*, W^*)$ and $c_T(B_i^*, W_i^*)$ respectively. We do not know the first but we can compute the latter according to the hint. Our solution strategy for G is to select the bisection (B_i^*, W_i^*) as solution for G, such that $c_{T_i}(B_i^*, W_i^*)$ is minimized among all T_i.
Due to the way we assigned edge capacities to the decomposition trees[1] we always have $c_T(B, W) \geq c_G(B, W)$. This is because if we send a a flow of value $c(e)$ over each edge of G that goes over the cut (U_t, V_t), then this incurs a multi commodity flow with congestion exactly $C_G = 1$ in G. In the lecture we learned that transferring this flow to T gives a congestion $C_T \leq C_G$. Such a small congestion in T would not be possible if there would be a cut smaller than $c_G(U_t, V_t)$ seperating U_t from V_t in T.
Assume that no (B_i^*, W_i^*) would give us a β-approximation of (B^*, W^*), that is $c_{T_i}(B_i^*, W_i^*) > \beta \cdot c_G(B^*, W^*)$ for all $i \in \{1, \ldots, k\}$. Then

$$c_{T_i}(B^*, W^*) \overset{(B_i^*, W_i^*) \text{ opt on } T_i}{\geq} c_{T_i}(B_i^*, W_i^*) \overset{\text{cuts in } G \text{ smaller}}{\geq} c_G(B_i^*, W_i^*) \overset{\text{assumption}}{>} \beta \cdot c_G(B^*, W^*).$$

That means we can route some multi commodity flow f_i with total demand of more than $\beta \cdot c_G(B^*, W^*)$ in T_i from B^* to W^* with congestion $C_{T_i} \leq 1$. At the same time we know that when we map the convex combination $f := \sum_{i=1}^k \lambda_i f_i$ to G, then f has a value of bigger than $\beta \cdot c_G(B^*, W^*)$, but at the same time this incurs only a congestion of at most $C_G \leq \beta$ on G, a contradiction to the fact that $c_G(B^*, W^*)$ is the size of the cut (B^*, W^*).

[1] Recall that any edge e_t of T induces a cut (U_t, V_t) of V and we assigned $c_T(e_t) := c_G(U_t, V_t)$, i.e. the sum over the capacities of edges in G that go over the cut (U_t, V_t).