Approximation algorithm for multiway-cut

We are given an undirected graph $G = (V, E)$ with costs $c_e \geq 0$ for all edges $e \in E$, and k distinguished vertices $s_1, ..., s_k$. The goal is to remove a minimum-cost set of edges F such that no pair of distinguished vertices s_i and s_j for $i = j$ are in the same connected component of $(V, E - F)$. We know that when $k = 2$ then the problem is just the min-cut problem, and it can be solved in polynomial time via max-flow. Consider the following algorithm for the problem with k vertices. For $i = 1, ..., k$ let F_i be the min-cut that separates vertex s_i and vertices $s_1, s_2, ..., s_{i-1}, s_{i+1}, ..., s_k$. Solution $i = 1, ..., k$ F_i is obviously a feasible solution, i.e. it separates all $s_1, ..., s_k$. Show that it is also a 2-approximation of the optimal solution.

Solution

Let be:

- E_{OPT} the set of the edges in the optimal solution;
- $OPT = \sum_{e \in E_{OPT}} c_e$ the value of the optimal solution;
- F_i is the min-cut that separates vertex s_i from other terminals;
- $F = \bigcup_{i=1}^{k} F_i$;
- E_i is the subset of E_{OPT} that it is a cut that separates vertex s_i from other terminals.

From the definition of min-cut we have:

$$cost(F_i) \leq cost(E_i) \Rightarrow \sum_{i=1}^{k} cost(F_i) \leq \sum_{i=1}^{k} cost(E_i).$$

Let be $G' = (V, E - E_{OPT})$. We know that G' has k disconnected components. Thus E_i is the set of edges that connect the component i to other ones. This implies that an edge $e \in E_i$ connects two components i, j thus the same edge $e \in E_j$. Remembering that $\bigcup_{i=1}^{k} = E_{OPT}$:

$$\Rightarrow \sum_{i=1}^{k} cost(E_i) = \sum_{i=1}^{k} \sum_{e \in E_i} c_e = \sum_{e \in E_{OPT}} 2c_e = 2 \cdot OPT.$$

Follows that:

$$\sum_{e \in F} c_e \leq \sum_{i=1}^{k} cost(F_i) \leq \sum_{i=1}^{k} cost(E_i) = 2 \cdot OPT.$$

$\square$