CMPUT 675: Approximation Algorithms	**Winter 2005**
Lecture 11: Feb 16	
Lecturer: Mohammad R. Salavatipour	*Scribe: Zhuang Guo*

Today and next lecture, we will be talking about another cut problem.

11.1 The Minimum Multicut Problem

The (integer) minimum multicut problem is defined as follows:

Input: an undirected graph $G = (V, E)$ with nonnegative weight/capacity c_e for each edge $e \in E$ and a set of source-sink pairs $S = \{(s_1, t_1), (s_2, t_2), \cdots, (s_k, t_k)\}$ of vertices, where each pair is distinct (but each vertex may appear in several pairs).

Definition 11.1 *A multicut is a set of edges $C \subseteq E$,whose removal separates each of the pairs. That is there is no $s_i \to t_i$ path in $G^{'}(V, E - C)$, for $1 \leq i \leq k$.*

Goal: to find a multicut with minimum total weight.

For $k = 1$, the problem becomes the minimum s-t cut problem, which can be solved in polynomial time using the maximum flow algorithms. For $k \geq 3$, the problem becomes NP-hard. In fact, the minimum multicut problem is APX-hard for any fixed $k \geq 3$. A multiway cut problem with a set of terminals $S = \{s_1, s_2, \cdots, s_k\}$ can be reduced to a minimum multicut problem by creating a pair (s_i, s_j) for any $1 \leq i, j \leq k$. This reduction implies that the minimum multicut problem is NP-hard even for $k = 3$, since the multiway cut problem is NP-hard for the case of 3 terminals.

In the next lecture, we will obtain an $O(\log k)$ factor approximation algorithm for the minimum multicut problem. In this lecture, we discuss a factor 2 approximation algorithm using LP-duality theory for the special case when G is restricted to be a tree.

11.2 The Primal-Dual Formulations

First, we obtain an integer programming formulation of the problem and then derive its LP-relaxations. Introduce a 0/1 variable x_e for each edge $e \in E$, which will be set to 1 iff e is picked in the multicut. Let p_i denote the unique path between s_i and t_i in the tree.

Integer programming formulation:

$$\begin{array}{lll} \textit{minimize} & \displaystyle\sum_{e \in E} c_e \cdot x_e & \\ \textit{subject to} & \displaystyle\sum_{e \in p_i} x_e \geq 1, & \forall p_i, i \in \{1, 2, \cdots, k\} \\ & x_e \in \{0, 1\}, & e \in E \end{array}$$

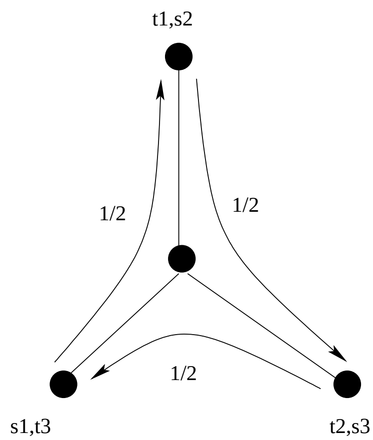


Figure 11.1: $IMCF \neq IMC$

The LP-relaxation is obtained by replacing the constraint $x_e \in \{0,1\}$ with $x_e \geq 0$:

$$\begin{array}{lll} \textit{minimize} & \sum_{e\in E} c_e \cdot x_e & \\ \textit{subject to} & \sum_{e \in p_i} x_e \geq 1, & \forall p_i, i \in \{1,2,\cdots,k\} \\ & x_e \geq 0, & e \in E \end{array}$$

Next, we derive the dual problem. We have already seen how to obtain he dual LP from the primal. The dual LP corresponds to a relaxation of another problem, called *maximum multicommodity flow*, in G where we have a separate commodity corresponding to each source-sink pair (s_i, t_i). Let dual variable f_i denote the amount of commodity routed along the unique path from s_i to t_i.

$$\begin{array}{lll} \textit{maximize} & \sum_{i=1}^{k} f_i & \\ \textit{subject to} & \sum_{i: e \in p_i} f_i \leq c_e, & e \in E \\ & f_i \geq 0, & i \in \{1,2,\cdots,k\} \end{array}$$

Definition 11.2 ***Integer multicommodity flow (IMCF)***
Given a graph G and source-sink pairs $(s_i, t_i), 1 \leq i \leq k$, all edge capacities are integral and we wish to route commodities from $s_i \to t_i$, such that the total flow routed along every edge (in both directions) is no more than the capacity of the edge and each flow is integral. The goal is to maximize the total flow.

By duality theory, the minimum fractional multicut is equal to the maximum fractional multicommodity flow. Also, the integer multicommodity flow (IMCF) is a lower bound for the integer multicut (IMC). However, the maximum IMCF is not necessarily equal to the minimum IMC. Consider the graph in Figure 11.1 with unit capacity edges and 3 vertex pairs. The maximum fractional multicommodity flow is $\frac{3}{2}$, which is obtained by routing $\frac{1}{2}$ unit along each source-sink pair. On the other hand, the minimum integral multicut is 2, because any integral multicut must pick at least two of the three edges in order to disconnect all three pairs.

Note that the minimum multicut problem is still NP-hard even if G is restricted to trees that are stars (of height 1) and all edge capacities are 1.

Theorem 11.3 *The minimum multicut problem on stars with unit capacities is equivalent to the vertex cover problem. In general, the minimum multicut problem on stars is equivalent to the (weighted) vertex cover problem.*

Corollary 11.4 *It is NP-hard to approximate the minimum multicut problem on stars with ratio less than* $\frac{7}{6}$.

11.3 A Primal-Dual schema based algorithm

We will use a primal-dual schema to obtain an algorithm that simultaneously solves IMC and IMCF on trees and the solutions are within a factor of 2 of each other. Hence, we get approximation algorithms for both problems. For IMC, the approximation factor is 2, and $\frac{1}{2}$ for IMCF.

Definition 11.5 *An edge e is tight if the total flow through it is equal to its capacity.*

By keeping the primal complementary slackness conditions ($\alpha = 1$) and relaxing the dual condition with $\beta = 2$, we get the following relaxed complementary slackness conditions:

Primal conditions ($\alpha = 1$): For each $e \in E$, $x_e \neq 0 \Rightarrow \sum_{i:\ e\in p_i} f_i = c_e$.
Equivalently, if e is picked, it is tight.

Relaxed dual conditions ($\beta = 2$): For each $i \in \{1, \cdots, k\}$, $f_i \neq 0 \Rightarrow \sum_{e\in p_i} x_e \leq 2$.
Equivalently, at most two edges are picked on every path with nonzero flow.

Given the input tree, consider its rooted version at some arbitrary vertex r.

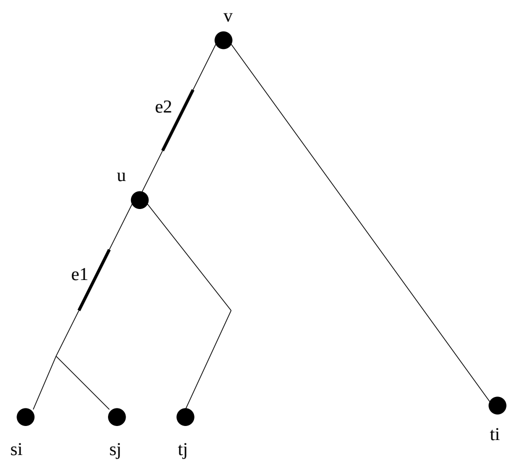
Definition 11.6 *The depth of a vertex v is the number of edges on the path $v \to r$. Root r has zero depth.*

Definition 11.7 *For two vertices $u, v \in V$, the lowest common ancestor of u and v, denoted $lca(u,v)$, is the minimum depth vertex on path $u \to v$.*

The idea of the algorithm is as follows. We start with the empty solution which is feasible for dual. As we go, we improve the feasibility of the primal and the optimality of the dual, integrally. At the end we will have an integral feasible primal and an integral feasible dual which are within a factor two of each other. By LP duality, we get the desired approximation factor. We start from the bottom of the tree and move up, checking vertices one by one. At each iteration, we look at one vertex v and greedily route all pairs (s_i, t_i) for which $lca(s_i, t_i) = v$. After this we will add all the edges the become tight to the set C. If we remove C from E, there cannot be any s_i, t_i-path, otherwise when we were considering $lca(s_i, t_i)$ we could have routed more flow and therefore the solution we had found was not maximal. This implies that:

Lemma 11.8 *Set C found is a multicut.*

However, we cannot guarantee that the size of C is small, as it may have many redundant edges. For this reason, the algorithm will have a clean-up phase. In this phase, we will delete the un-necessary edges from C. The following is the IMCF and IMC approximation algorithm:



Approximation algorithm for IMCF and IMC on trees:

1. Let $f \leftarrow 0$, $C \leftarrow \Phi$.
2. for each $v \in V$, in non-increasing order of depth, do:
   For each pair (s_i, t_i) such that $lca(s_i, t_i) = v$, greedily route maximum integral flow that you can from s_i to t_i.
   Add all edges that become tight in the current iteration to C in arbitrary order.
3. Let $e_1, e_2, \cdots, e_m$ be the edges in C in the order they were added.
4. for $j = m$ to 1 do:
   if $C - \{e_j\}$ is a multicut, delete e_j from C.
5. Output the flow and the multicut C.

Clearly, we only remove an edge e from C if $C - e$ is still a multicut. Therefore, using Lemma 11.8, it is clear that at the end of this algorithm the solution is indeed a multicut.

Lemma 11.9 *For every pair (s_i, t_i) with nonzero flow, at most two edges are picked.*

Proof: Let $lca(s_i, t_i) = v$. We prove that at most one edge is picked on each of the two paths: $s_i \to v$ and $v \to t_i$. We give the argument for the $s_i \to v$ path. The same argument works for the other path too. By way of contradiction, suppose that two edges e_1 and e_2 are both picked on path $s_i \to v$. Let's assume that e_1 is the deeper edge. Consider the time during the reverse removal when e_1 is tested. Since e_1 is not discarded, there must be a source-sink pair, say (s_j, t_j), such that e_1 is the only edge picked on the path $s_j \to t_j$. Let $u = lca(s_j, t_j)$. Since e_2 does not lie on path $s_j \to t_j$, u must be lower than e_2. After u was processed, C must contain an edge e_3 from the path $s_j \to t_j$. Since nonzero flow was routed between s_i and t_i, e_1 must be added during or after the iteration v was processed. Since v is an ancestor of u, e_1 must be added to C after e_3 was added. So $e_1 \neq e_1$ and e_3 must be in C when e_1 was being tested. This contradicts the fact that e_1 is the only edge picked on path $s_j \to t_j$. ∎

Theorem 11.10 *This algorithm has an approximation factor 2 for IMC and $\frac{1}{2}$ for IMCF.*

Proof: By Lemma 11.8, after Step 2, C is a multicut. Since the reverse removal step only deletes redundant edges, C is still a multicut after this step. Thus, a feasible solution has been found for both the IMCF and IMC problem.

Since each edge in multicut C is tight (we only pick tight edges), the primal conditions are satisfied. From Lemma 11.9, the relaxed dual conditions must also hold. From $\alpha \cdot \beta = 2$, we know that the approximation factor is 2 for IMC and $\frac{1}{2}$ for IMCF. ■