Lecture notes for Apr 5, 2023
Gomory-Hu trees

Chun-Hung Liu

April 5, 2023

We recall the terminologies. A *tree-cut decomposition* of (G, w) is a pair $(T, \mathcal{X})$, where T is a tree and $\mathcal{X}$ is a partition $\{X_t : t \in V(T)\}$ of $V(G)$ into (possibly empty) sets indexed by $V(T)$. For every edge xy of T,

- we know that $T - xy$ contains two components T_x and T_y of T, where T_x contains x and T_y contains y, so $[\bigcup_{t \in V(T_x)} X_t, \bigcup_{t \in V(T_y)} X_t]$ is an edge-cut of (G, w), and we call this edge-cut the *edge-cut given by* (x, y) *(with respect to* $(T, \mathcal{X})$*)*,
- the *adhesion of* xy *(with respect to* $(T, \mathcal{X})$*)* is defined to be the weight of the edge-cut given by (x, y),

A *Gomory-Hu tree* of a positive weighted graph (G, w) is a tree-cut decomposition $(T, \mathcal{X})$ of G such that

- $|X_t| = 1$ for every $t \in V(T)$, and
- for any distinct $x, y \in V(G)$, the minimum weight of an edge-cut of (G, w) separating x and y equals the minimum of the adhesion of e over all edges e in the unique path in T between x and y.

  (Note that it implies that if the edge e gives the minimum adhesion in the path between x and y, then the edge-cut given by e is an edge-cut of (G, w) separating x and y.)

Therefore, if a Gomory-Hu tree $(T, \mathcal{X})$ of a positive weighted graph (G, w) is given, we can find the edge-connectivity in linear time by simply finding the minimum adhesion of the edges of T, which only takes time $O(|V(G)|)$; and for any distinct vertices x, y of G, we can find an edge-cut separating x and y with minimum weight by simply finding the edge in T in the path between x and y giving the minimum adhesion, which only takes time $O(|V(G)|)$.

1 Nice tree-cut decompositions

A tree-cut decomposition $(T, \mathcal{X})$ of (G, w) is *nice* if for every $xy \in E(T)$, there exist $u \in X_x$ and $v \in X_y$ such that the edge-cut given by (x, y) is a minimum (weighted) (u, v)-edge-cut in (G, w). Notice that the tree-cut decomposition $(T, \mathcal{X})$ of (G, w) with $|V(T)| = 1$ is a nice tree-cut decomposition.

Lemma 1 *Let (G, w) be a weighted graph with positive w. Let $(T, \mathcal{X})$ be a nice tree-cut decomposition of G such that $|X_t| = 1$ for every $t \in V(T)$. Then $(T, \mathcal{X})$ is a Gomory-Hu tree.*

Proof. Since $|X_t| = 1$ for every $t \in V(T)$, we may assume $V(G) = V(T)$. Let a, b be distinct vertices of G. Let λ be the minimum weight of an edge-cut of (G, w) separating a and b. Let P be the path in T between a and b. For every $e \in E(P)$, let λ_e be the adhesion of e. For every $e \in E(P)$, since the edge-cut of G given by e separates a and b, we know $\lambda \leq \lambda_e$. So $\lambda \leq \min_{e \in E(P)} \lambda_e$.

Suppose to the contrary that $\lambda < \min_{e \in E(P)} \lambda_e$. Let $[A, B]$ be a minimum weighted (a, b)-edge-cut of G. So $w(A, B) = \lambda$. Since $a \in A$ and $b \in B$, there exists an edge uv of P such that $u \in A$ and $v \in B$. Then $[A, B]$ is an edge-cut separating u and v with weight $\lambda < \lambda_{uv}$. Since $(T, \mathcal{X})$ is nice, there exist $u' \in X_u$ and $v' \in X_v$ such that the edge-cut $[A', B']$ given by (u, v) is a minimum weighted (u', v')-edge-cut. Note that $w(A', B') = \lambda_{uv}$. But $|X_u| = |X_v| = 1$, so $u' = u$ and $v' = v$. Hence $[A, B]$ is an edge-cut separating u and v with weight smaller than $[A', B']$, a contradiction. ∎

2 Submodularity

The following property for edge-cuts is useful.

Proposition 2 (Submodularity of edge-cuts) *Let (G, w) be a weighted graph with positive w. Let $[A, B]$ and $[C, D]$ be edge-cuts of (G, w). Then $w(A, B) + w(C, D) \geq w(A \cap C, B \cup D) + w(A \cup C, B \cap D)$.*

Proof. It is straightforward to verify it by considering the contribution of $w(e)$ in the two sides of the inequality for each edge e of G. ∎

3 Torsos and splitting

In this subsection, (G, w) is a weighted graph, where w is a positive function.

Given a partition $\mathcal{P}$ of $V(G)$, we define $(G, w)/\mathcal{P}$ to be the weighted graph (G', w'), where

- G' is the graph obtained from G by for each nonempty member M of $\mathcal{P}$, identifying M into a single vertex v_M and deleting the resulting loops (but keeping parallel edges),

  (so there is a natural injection from $E(G')$ to $E(G)$, and we can treat each edge of G' an edge of G),

- $w' : E(G') \to \mathbb{R}$ is the function such that for every edge e of G', $w'(e) = w(e)$.

Let $(T, \mathcal{X})$ be a tree-cut decomposition of G. For every $t \in V(T)$,

- the set X_t is called the *bag* at t, and
- the *torso* at t is the weighted graph $(G, w)/\mathcal{P}_t$, where $\mathcal{P}_t$ is the partition $\{\{v\} : v \in X_t\} \cup \{\bigcup_{x \in V(C)} X_x : C$ is a component of $T - t\}$ of $V(G)$.

If $t \in V(T)$ and $[A, B]$ is an edge-cut of the torso at t, then

- the $[A, B]$-*extension* is the edge-cut $[A', B']$ of G, where
  - $A' = (X_t \cap A) \cup \bigcup\{X_s : s \in V(T) - \{t\}$ and X_s is contained in a part of $\mathcal{P}_t$ identified into a vertex in $A\}$, and
  - $B' = (X_t \cap B) \cup \bigcup\{X_s : s \in V(T) - \{t\}$ and X_s is contained in a part of $\mathcal{P}_t$ identified into a vertex in $B\}$,
- the $[A, B]$-*split* of $(T, \mathcal{X})$ is the tree-cut decomposition $(T', \mathcal{X}')$ such that
  - the vertex-set $V(T')$ is obtained from $V(T) - \{t\}$ by adding two new vertices t_A and t_B,

- the edge-set $E(T') = \{t_A t_B\} \cup (E(T) - \delta(t)) \cup \{t_A s : s \in N_T(t), X_s \subseteq A'\} \cup \{t_B s : s \in N_T(t), X_s \subseteq B'\}$, and
- $\mathcal{X}' = (X'_z : z \in V(T'))$ such that
  * $X'_{t_A} = X_t \cap A$,
  * $X'_{t_B} = X_t \cap B$, and
  * for every $z \in V(T') - \{t_A, t_B\} = V(T) - \{t\}$, $X'_z = X_z$.

Lemma 3 *Let (G, w) be a weighted graph with positive w. Let $(T, \mathcal{X})$ be a nice tree-cut decomposition of (G, w). Let $t \in V(T)$. Let $u, v \in X_t$. If $[A, B]$ is a minimum weighted (u, v)-edge-cut of the torso at t, then the $[A, B]$-extension is a minimum weighted (u, v)-edge-cut of (G, w).*

Proof. Clearly, the $[A, B]$-extension is an edge-cut of (G, w) with $u \in A$ and $v \in B$ such that it has weight equal to $w(A, B)$. So it suffices to show that no edge-cut of (G, w) separating u and v has weight smaller than $w(A, B)$.

For every edge-cut $[X, Y]$ of G, we define the *badness* of $[X, Y]$ to be the number of nodes t of T such that $X_t \cap X \neq \emptyset \neq X_t \cap Y$. Note that every edge-cut of G separating u and v has badness at least 1, and if it has badness 1, then it also gives an edge-cut of the torso at t with the same weight. So to prove this lemma, it suffices to show that there exists a minimum weighted (u, v)-edge-cut of (G, w) with badness 1.

Let $[C, D]$ be a minimum weighted (u, v)-edge-cut with minimum badness. Suppose to the contrary that $[C, D]$ has badness at least two. Then there exists $s \in V(T) - \{t\}$ such that $X_s \cap C \neq \emptyset \neq X_s \cap D$. Let s' be the neighbor of s in T contained in the path in T between s and t. Let $[A_s, B_s]$ be the edge-cut given by (s, s'). Since $(T, \mathcal{X})$ is a nice tree-cut decomposition, there exist $u_s \in A_s$ and $v_s \in B_s$ such that $[A_s, B_s]$ is a minimum weighted (u_s, v_s)-edge-cut of (G, w).

If $u_s \in C$, then let $[C^*, D^*] = [C, D]$; otherwise, let $[C^*, D^*] = [D, C]$. That is, $u_s \in C^*$.

Since $u_s \in A_s$ and $v_s \in B_s$, $[A_s \cap C^*, B_s \cup D^*]$ is an edge-cut of G separating u_s and v_s. Since $[A_s, B_s]$ is a minimum weighted (u_s, v_s)-edge-cut of G, we know $w(A_s \cap C^*, B_s \cup D^*) \geq w(A_s, B_s)$. Then by Proposition 2, $w(A_s \cup C^*, B_s \cap D^*) \leq w(C^*, D^*) = w(C, D)$.

Note that $\{u, v\} \subseteq X_t \subseteq B_s$, and exactly one of u, v is in D^*. So $[A_s \cup C^*, B_s \cap D^*]$ is an edge-cut of G separating u and v. Since $[C, D]$ is a minimum weighted (u, v)-edge-cut of G, we know $w(A_s \cup C^*, B_s \cap D^*) \geq w(C, D)$. Therefore, $w(A_s \cup C^*, B_s \cap D^*) = w(C, D)$.

Note that $X_s \cap (B_s \cap D) = \emptyset$. And for every $z \in V(T) - \{t, s\}$, if $X_z \cap (A_s \cup C) \neq \emptyset \neq X_z \cap (B_s \cap D)$, then z is contained in the component of $T - ss'$ containing t, so $X_z \cap C \neq \emptyset \neq X_z \cap D$. That is, the badness of $[C, D]$ is strictly bigger than $[A_s \cup C, B_s \cap D]$, a contradiction. ■

Lemma 4 *Let (G, w) be a weighted graph with positive w. Let $(T, \mathcal{X})$ be a nice tree-cut decomposition of (G, w). Let $t \in V(T)$. Let $u, v \in X_t$. If $[A, B]$ is a minimum weighted (u, v)-edge-cut of the torso at t, then the $[A, B]$-split of $(T, \mathcal{X})$ is a nice tree-cut decomposition of (G, w).*

Proof. Let $(T', \mathcal{X}')$ be the $[A, B]$-split of (T, X). Let t_A and t_B be the two new vertices in T. Since $[A, B]$ is a minimum weighted (u, v)-edge-cut of the torso at t, Lemma 3 implies that the edge-cut of (G, w) given by $t_A t_B$ is a minimum weighted (u, v)-edge-cut with $u \in X'_{t_A}$ and $v \in X'_{t_B}$. And note that for every edge e of T not incident with t, the edge-cut given by e with respect to $(T, \mathcal{X})$ and the edge-cut given by e with respect to $(T', \mathcal{X}')$ are the same.

For every $s \in N_T(t)$, let t_s be the vertex in $\{t_A, t_B\}$ such that $st_s \in E(T')$. Since $(T, \mathcal{X})$ is nice, to show that $(T', \mathcal{X}')$ is nice, it suffices to show that for every $s \in N_T(t)$, there exist $v_s \in X'_s$ and $v_{t_s} \in X'_{t_s}$ such that the edge-cut given by (s, t_s) is a minimum weighted (v_s, v_{t_s})-edge-cut of (G, w).

Let $s \in N_T(t)$. Since $(T, \mathcal{X})$ is nice and $st \in E(T)$, there exist $v_s \in X_s = X'_s$ and $v_t \in X_t$ such that the edge-cut $[A_{st}, B_{st}]$ given by (s, t) is a minimum weighted (v_s, v_t)-edge-cut. By the definition of the $[A, B]$-split, $[A_{st}, B_{st}]$ is also the edge-cut given by (s, t_s). If $v_t \in X'_{t_s}$, then we are done by choosing $v_{t_s} = v_t$. So we may assume $v_t \notin X'_{t_s}$.

Let $[A', B']$ be the $[A, B]$-extension. Note that $[A', B']$ separates v_s and v_t. So $w(A', B') \geq w(A_{st}, B_{st})$.

By symmetry, we may assume $t_s = t_A$, so $v_s \in A'$ and $v_t \in B'$. Let $[C, D]$ be a minimum weighted (v_s, u)-cut of (G, w). Note that $v_s \in X'_s$ and $u \in X'_{t_s}$. If $w(C, D) = w(A_{st}, B_{st})$, then we are done by choosing $v_{t_s} = u$. So we may assume $w(C, D) \neq w(A_{st}, B_{st})$. Since $v_s \in X_s \subseteq A_{st}$ and $u \in X_t \subseteq B_{st}$, $[A_{st}, B_{st}]$ separates v_s and u, so $w(A_{st}, B_{st}) \geq w(C, D)$. Hence $w(A_{st}, B_{st}) > w(C, D)$.

By Lemma 3, $[A', B']$ is a minimum weighted (u, v)-edge-cut. If $v \in C$, then $[C, D]$ separates v and u, so $w(C, D) \geq w(A', B') \geq w(A_{st}, B_{st})$, a contradiction. So $v \in D$.

Since $[A' \cap C, B' \cup D]$ separates v_s and v_t, $w(A' \cap C, B' \cup D) \geq w(A_{st}, B_{st}) > w(C, D)$. Since $[A' \cup C, B' \cap D]$ separates u and v, $w(A' \cup C, B' \cap D) \geq$

$w(A', B')$. But by Proposition 2, $w(A' \cap C, B' \cup D) + w(A' \cup C, B' \cap D) \leq w(A', B') + w(C, D)$, a contradiction. ■

4 Algorithm

==============================

An algorithm for finding a Gomory-Hu tree
Input: A weighted graph (G, w), where w is a positive function.
Output: A Gomory-Hu tree $(T, \mathcal{X})$ of (G, w).
Procedure:

Step 0: Delete all loops of G. And for any two distinct vertices u and v of G, if there exist more than one edge of G between u and v, then delete all those edges and add a single edge between u and v whose weight equals the sum of the weight of the original edges between u and v.

Step 1: Set $(T_0, \mathcal{X}_0)$ be a tree-cut decomposition of G such that T_0 is a tree with one vertex,

Step 2: For $i = 1, 2, ..., |V(G)| - 1$, do the following:

- Pick a node t of T_{i-1} with $|X_t| \geq 2$, pick two distinct vertices x, y in X_t, and find a minimum weighted (x, y)-edge-cut $[S_x, S_y]$ of the torso at t with respect to $(T_{i-1}, \mathcal{X}_{i-1})$.
- Define $(T_i, \mathcal{X}_i)$ to be the $[S_x, S_y]$-split of $(T_{i-1}, \mathcal{X}_{i-1})$.

Step 3: Output $(T_{|V(G)|-1}, \mathcal{X}_{|V(G)|-1})$.

==============================

Theorem 5 *The above algorithm outputs a Gomory-Hu tree in time* $O(|E(G)| + |V(G)|^3 \sqrt{|E(G)|})$.

Proof. Note that Gormory-Hu trees of the original graph G are exactly the Gomory-Hu tree of the graph G modified in Step 0.

We first show the correctness. Clearly, $(T_0, \mathcal{X}_0)$ is a nice tree-cut decomposition. By Lemma 4, $(T_i, \mathcal{X}_i)$ is nice for every $i \geq 1$. And clearly for every

$0 \leq i \leq |V(G)| - 1$, $|V(T_i)| = i + 1$ and the bag of $(T_i, \mathcal{X}_i)$ at any node contains at least one vertex. So $(T_{|V(G)|-1}, \mathcal{X}_{|V(G)|-1})$ is a nice tree-decomposition such that every bag has size one. Hence $(T_{|V(G)|-1}, \mathcal{X}_{|V(G)|-1})$ is a Gomory-Hu tree by Lemma 1.

Now we show the time complexity. Step 0 takes time $O(|E(G)|)$. Step 1 takes time $O(|V(G)|)$. For each round of Step 2, we can find a desired vertex t of T in linear time, construct the torso at t in linear time, find the edge-cut $[S_x, S_y]$ in time that runs a minimum cut algorithm (which can be done in $O(|V(G)|^2\sqrt{|E(G)|})$ time by a previous theorem), and find the $[S_x, S_y]$-split in linear time. So each round of Step 2 can be done in time $O(|V(G)|^2\sqrt{|E(G)|} + |E(G)|) = O(|V(G)|^2\sqrt{|E(G)|})$ since G is simple after Step 1. And we execute Step 2 $|V(G)|$ times. So the total running time is $O(|E(G)| + |V(G)|^3\sqrt{|E(G)|})$. ■