17.1 Multiway Cut: Improved LP

We're going to continue from last week, where we were trying to design a better-than-2-approximation for Multiway Cut by using a better LP relaxation. Let's review where we were, starting with the definition of Multiway Cut

$$
\begin{aligned}
\text{Input:} \quad & \text{Graph } G = (V, E) \\
& \text{Costs } c : E \to \mathbb{R}^+ \\
& T = \{s_1, s_2, ..., s_k\} \subseteq V \\
\text{Feasible:} \quad & A \subseteq E \text{ s.t. } G - A \text{ has no } s_i - s_j \text{ path } \forall i, j \in \{1, 2, ..., k\} \\
\text{Objective:} \quad & \min \sum_{e \in A} c(e)
\end{aligned}
$$

Consider some optimal solution F. Then let $C_i = \{v \in V : v \text{ reachable from } s_i \text{ in } G \setminus F\}$. Clearly since F is a feasible multiway cut, $C_i \cap C_j = \emptyset$ for all $i \neq j$. But it turns out that they also form a partition: every node is in some C_i. To see this, suppose that it is false, and let S be the set of nodes that unreachable from any terminal in $G \setminus F$. Add S to C_1 to get C_1', which together with the other C_i's now form a partition. Then any edge which is cut under the new sets was also cut by the old sets, so this new solution is just as good as the old solution.

Thus WLOG, we can assume that the optimal solution is actually a partition $C_1, C_2, \ldots C_k$ where $s_i \in C_i$ for all $i \in [k]$, and the edges we cut are the edges between the parts of the partition.

This point of view suggests the following, different LP relaxation. We'll have the following variables.

$$
x_u^i = \begin{cases} 1 & \text{if } u \in C_i \\ 0 & \text{else} \end{cases}
$$

$$
z_e^i = \begin{cases} 1 & \text{if } e \in \delta(C_i) \\ 0 & \text{else} \end{cases}
$$

We now define an LP using these indicator variables

$$\min \quad \frac{1}{2} \sum_{e \in E} \sum_{i=1}^{k} c(e) z_e^i$$

$$\text{subject to} \quad \sum_{i=1}^{k} x_u^i = 1 \qquad\qquad\qquad \forall u \in V$$

$$z_e^i \geq x_u^i - x_v^i \qquad \forall e = \{u, v\} \in E, \quad \forall i \in 1, 2, ..., k$$

$$z_e^i \geq x_v^i - x_u^i \qquad \forall e = \{u, v\} \in E, \quad \forall i \in 1, 2, ..., k$$

$$x_{s_i}^i = 1 \qquad\qquad\qquad\qquad \forall i \in 1, 2, ..., k$$

$$0 \leq x_u^i \leq 1 \qquad\qquad\qquad \forall u \in V, \quad \forall i \in [k]$$

$$0 \leq z_e^i \leq 1 \qquad\qquad\qquad \forall e \in E, \quad \forall i \in [k]$$

It is straightforward to verify that this is a valid relaxation of the multiway cut problem. We will give a more compact way of writing this LP which makes the connection to metrics clear.

Definition 17.1.1 *Let $x, y \in \mathbb{R}^k$ then their ℓ_1-distance is $\|x - y\|_1 = \sum_{i=1}^{k} |x^i - y^i|$ where x^i is the ith coordinate of the vector x.*

Definition 17.1.2 *The k-simplex is $\Delta_k = \{x \in \mathbb{R}^k : \sum_{i=1}^{k} x^i = 1 \wedge x^i \geq 0 \; \forall i\}$ where x^i is the ith coordinate of the vector x.*

Definition 17.1.3 *Let e_i be a vector with a 1 in the ith coordinate and zeros elsewhere*

Let $x_u = (x_u^1, x_u^2, ..., x_u^k)$ and $x_v = (x_v^1, x_v^2, ..., x_v^k)$. Note that $z_e^i = |x_u^i - x_v^i|$ in any optimal LP solution because of the constraints on z_e^i and because we are minimizing the objective function. So, $\|x_u - x_v\|_1 = \sum_{i=1}^{k} z_e^i$. Using all these definitions we can rewrite the new LP as follows

$$\min \quad \frac{1}{2} \sum_{e = \{u,v\} \in E} c(e) \|x_u - x_v\|_1$$

$$\text{subject to} \quad x_{s_i} = e_i \qquad\qquad \forall i \in 1, 2, ..., k$$

$$x_u \in \Delta_k$$

Note that $\|x_{s_i} - x_{s_j}\|_1 = \|e_i - e_j\|_1 = 2$ for all $i \neq j \in [k]$.

17.2 Improved Rounding for Multiway Cut

Now we need a way of rounding this LP. We're going to use a technique due to Calinescu, Karloff, and Rabani [CKR00], which might look familiar: it's a lot like one level of FRT! Formally, we'll use the following algorithm. We'll define balls as one would expect, i.e., we'll let $B(s_i, r) = \{v \in V : \|e_i - x_v\|_1 \leq r\}$.

1. Initially we set $C_i = \emptyset$ for all $i \in [k]$, and set another set $X = \emptyset$.

2. Pick $r \in (0, 2)$ uniformly at random, and pick a permutation π of $[k]$ uniformly at random.

3. For $i = 1$ to $k - 1$:

   (a) Set $C_{\pi(i)} \leftarrow B(s_{\pi(i)}, r) \setminus X$
   (b) $X \leftarrow X \cup C_{\pi(i)}$

4. Set $C_{\pi(k)} = V \setminus X$

5. Return $A = \cup_{i=1}^{k} \delta(C_i)$ (the edges cut by the partition that we built)

We're going to prove the following theorem.

Theorem 17.2.1 *This is a 3/2-approximation.*

To prove this, for every edge $\{u, v\}$ let $Z_{u,v}$ be an indicator random variable which will be 1 if this rounding algorithm separates u and v (i.e., includes e in the cut) and is 0 otherwise. We'll first state the following lemma.

Lemma 17.2.2 $\mathbf{Pr}[Z_{u,v} = 1] \leq \frac{3}{4}\|x_u - x_v\|_1$ *for all* $\{u, v\} \in E$.

It's straightforward to see that this lemma implies the theorem:

$$\mathbf{E}[ALG] = \mathbf{E}\left[\sum_{e=\{u,v\}\in E} c(e)Z_{u,v}\right] = \sum_{e=\{u,v\}\in E} c(e)\mathbf{E}[Z_{u,v}]$$

$$\leq \sum_{e=\{u,v\}\in E} c(e)\frac{3}{4}\|x_u - x_v\|_1 = \frac{3}{2} \cdot \frac{1}{2} \cdot \sum_{e=\{u,v\}\in E} c(e)\|x_u - x_v\|_1$$

$$\leq \frac{3}{2} \cdot OPT.$$

So we just need to prove Lemma 17.2.2. This is going to look an awful lot like a simplified version of FRT. Fix some $\{u, v\} \in E$. Define S_i to be 1 if i is the first index in π such that $|B(s_i, r) \cap \{u, v\}| \geq 1$ (i.e., s_i settles $\{u, v\}$) and 0 otherwise. Define X_i to be 1 if $|B(s_i, r) \cap \{u, v\}| = 1$ (i.e., s_i cuts $\{u, v\}$) and 0 otherwise.

Similar to FRT (but simpler since there's only one "level"), we can note that $Z_{u,v} = 1$ if and only if there is some $i \in [k]$ such that $S_i = X_i = 1$. And by the definition of S_i there is at most one i which settles $\{u, v\}$, and thus some $i \in [k]$ has $S_i = X_i = 1$ if and only if $\sum_{i=1}^{k} S_i X_i = 1$. Thus $Z_{u,v} = \sum_{i=1}^{k} S_i X_i$, so by linearity of expectations we get that $\mathbf{E}[Z_{u,v}] = \sum_{i=1}^{k} \mathbf{E}[S_i X_i]$, and thus $\mathbf{Pr}[Z_{u,v} = 1] = \sum_{i=1}^{k} \mathbf{Pr}[S_i = X_i = 1]$. Less formally, the probability that $\{u, v\}$ is cut by the algorithm is equal to the sum over all terminals of the probability that the algorithm cut the edge because of that terminal.

Let's start by analyzing the cut probabilities. To do this, let's prove another easy lemma.

Lemma 17.2.3 *For all $w \in V$ and $i \in [k]$, $w \in B(s_i, r)$ if and only if $1 - x_w^i \leq r/2$.*

Proof: We have the following series of if and only if statements:

$$w \in B(s_i, r) \Leftrightarrow \|e_i - x_w\|_1 \leq r$$

$$\Leftrightarrow \sum_{j=1}^{k} |e_i^j - x_w^j| \leq r$$

$$\Leftrightarrow 1 - x_w^i + \sum_{j \neq i} x_w^j \leq r$$

$$\Leftrightarrow 2(1 - x_w^i) \leq r$$

$$\Leftrightarrow 1 - x_w^i \leq r/2,$$

as claimed. ∎

To interpret this lemma, it says that we can think of the distance from w to s_i as being $2(1 - x_w^i)$. This lemma implies that

$$\mathbf{Pr}[X_i = 1] = \mathbf{Pr}\left[\min(1 - x_v^i, 1 - x_u^i) \leq r/2 < \max(1 - x_u^i, 1 - x_v^i)\right]$$

$$= |x_u^i - x_v^i|.$$

Let $\ell \in [k]$ be the index minimizing $\min_{i=1}^{k} \min(1 - x_u^i, 1 - x_v^i)$, i.e., s_ℓ is the terminal that is closest to $\{u, v\}$ in the ℓ_1-metric given by the LP. For this special index, we can bound the probability that the algorithm cuts $\{u, v\}$ in because of s_ℓ by

$$\mathbf{Pr}[S_\ell = X_\ell = 1] \leq \mathbf{Pr}[X_\ell = 1] \leq |x_u^\ell - x_v^\ell|$$

Now consider some $i \neq \ell$. If i cuts $\{u, v\}$, then $|B(s_\ell, r) \cap \{u, v\}| \geq 1$ since ℓ is closer to $\{u, v\}$ than i is. This if i cuts $\{u, v\}$, a necessary condition for it to settle $\{u, v\}$ is for it to be before ℓ in π. Since π is chosen uniformly at random, this happens with probability $1/2$. Thus we get that

$$\mathbf{Pr}[S_i = X_i = 1] = \mathbf{Pr}[S_i = 1 | X_i = 1]\mathbf{Pr}[X_i = 1] \leq \frac{1}{2}|x_u^i - x_v^i|.$$

Putting this all together we have

$$\mathbf{Pr}[Z_{u,v} = 1] = \sum_{i=1}^{k} \mathbf{Pr}[X_i = S_i = 1] \leq |x_u^\ell - x_v^\ell| + \frac{1}{2} \sum_{i \neq \ell} |x_u^i - x_v^i|$$

$$= \frac{1}{2}|x_u^\ell - x_v^\ell| + \frac{1}{2}\|x_u - x_v\|_1$$

$$\leq \frac{3}{4}\|x_u - x_v\|_1.$$

which completes the proof of Lemma 17.2.2 and thus the proof of Theorem 17.2.1. However, to prove the last inequality, we actually need one final lemma.

Lemma 17.2.4 *For any index j and any $u, v \in V$, $|x_u^j - x_v^j| \leq \frac{1}{2}\|x_u - x_v\|_1$.*

4

Proof: Without loss of generality, let's assume that $x_u^j \geq x_v^j$. Then

$$|x_u^j - x_v^j| = x_u^j - x_v^j = \left(1 - \sum_{i \neq j} x_u^i\right) - \left(1 - \sum_{i \neq j} x_v^i\right) = \sum_{i \neq j}\left(x_v^i - x_u^i\right) \leq \sum_{i \neq j}|x_v^i - x_u^i|.$$

Adding $|x_u^j - x_v^j|$ to both sides and then dividing by two gives the lemma. ∎

17.3 Multicut

Definition 17.3.1 *In the Multicut problem, we are given a graph $G = (V, E)$ with costs $c : E \rightarrow \mathbb{R}^+$, and k pairs $(s_1, t_1), \ldots, (s_k, t_k)$ of nodes. A feasible solution is a set $F \subseteq E$ such that s_i and t_i are not connected in $G \setminus F$ for all $i \in [k]$. The objective is to minimize $\sum_{e \in F} c(e)$.*

For the remainder of the day, we're going to prove the following theorem:

Theorem 17.3.2 *There is an $O(\log n)$-approximation algorithm for Multicut.*

We will use $\mathcal{P}_i$ to denote the set of all s_i-t_i paths. The problem admits the following LP relaxation:

$$\text{minimize:} \quad \sum_{e \in E} c(e) \cdot x_e \qquad\qquad\qquad \textbf{(MULTICUT-LP)}$$

$$\text{subject to:} \quad \sum_{e \in P} x_e \geq 1 \quad \forall i \in [k], \forall P \in \mathcal{P}_i \qquad (17.3.1)$$

$$0 \leq x_e \leq 1 \quad \text{for each edge } e \in E \qquad (17.3.2)$$

Note: As with multiway cut, we can solve this LP in polytime via ellipsoid, using shortest path (for each $\mathcal{P}_i$) to separate. For the remainder, we will use $\vec{x}$ to refer to the solution of the LP, and set $V^* = \sum_{e \in E} c(e) x_e$ as the value of the solution.

Definition 17.3.3 *Let $d : V \times V \rightarrow \mathbb{R}^+$ be the shortest path metric using the LP solution $\vec{x}$ for the edge lengths.*

Similarly to what we had for Multiway Cut, we know that $d(s_i, t_i) \geq 1$ for all $i \in [k]$. How should we round the LP solution? Before we say *how*, let's first define what we want. In particular, consider the following definition.

Definition 17.3.4 *Given a metric space (V, d), a* Low-Diameter Random Decomposition *with parameter δ is a randomized algorithm which creates a partition $C_1, C_2, \ldots, C_\ell$ of V with the following properties.*

1. *$diam(C_i) \leq \delta$ for all $i \in [\ell]$*

2. *$\textbf{Pr}[u, v \text{ in different clusters}] \leq \frac{d(u,v)O(\log n)}{\delta}$ for all $u, v \in V$.*

Suppose that we have a LDRD algorithm with parameter $1 - \epsilon$ for some $\epsilon > 0$. If we apply this to the metric space from the LP, we get a bunch of clusters and can define F to be the edges between

the clusters, i.e., set $F = \cup_{i=1}^{\ell} \delta(C_i)$. Since $d(s_i, t_i) \geq 1$, we know that s_i and t_i are in different clusters, and thus are not connected in $G \setminus F$. So this would give a feasible solution.

To analyze the expected cost, for every edge $\{u, v\}$ let $Z_{u,v}$ be an indicator random variable for u and v being in different clusters of the partition. Then the expected cost of our solution is

$$\mathbf{E}\left[\sum_{e \in E} c(e) Z_e\right] = \sum_{e \in E} c(e) \mathbf{E}[Z_e] \leq \sum_{e = \{u, v\} \in E} c(e) \frac{d(u, v) O(\log n)}{1 - \epsilon} \leq \frac{O(\log n)}{1 - \epsilon} \sum_{e \in E} c(e) x_e$$
$$\leq O(\log n) \cdot LP$$

So if we can design an algorithm for Low-Diameter Random Decompositions, we get an $O(\log n)$-approximation for Multicut. And, amazingly, we already know how to design an LDRD: it's basically a single level of FRT!

- Choose r uniformly at random in $[\delta/4, \delta/2]$

- Choose a random permutation $\pi : [n] \to V$

- Let $S \leftarrow V$

- For $j = 1$ to n

  - $P = B(\pi(j), r) \cap S$
  - If $P \neq \emptyset$
    - ∗ Add P as a cluster
    - ∗ $S \leftarrow S \setminus P$

Clearly the diameter of every cluster is at most δ, as desired. So we just need to analyze the probability of separating u and v. Let's just repeat what we did for FRT!

Definition 17.3.5 w settles u, v if w is the first vertex in π s.t. $B(w, r) \cap \{u, v\} \neq \emptyset$.

Definition 17.3.6 w cuts u, v if $|B(w, r) \cap \{u, v\}| = 1$.

Clearly u and v are in different clusters if and only if there is some $w \in V$ that both settles and cuts u, v. Let S_w be an indicator random variable for the event that w settles u, v, and let X_w be an indicator random variable for the event that w cuts u, v. As with FRT, since exactly one node settles u, v, there is some w which both settles and cuts u, v if and only if $\sum_{w \in V} S_w X_w = 1$. So we have that

$$\mathbf{Pr}[u, v \text{ in different clusters}] = \mathbf{Pr}\left[\sum_{w \in V} S_w X_w = 1\right] = \mathbf{E}\left[\sum_{w \in V} S_w X_w\right] = \sum_{w \in V} \mathbf{E}[S_w X_w]$$
$$= \sum_{w \in V} \mathbf{Pr}[S_w = X_w = 1] = \sum_{w \in V} \mathbf{Pr}[S_w = 1 | X_w = 1] \mathbf{Pr}[X_w = 1]$$

The following lemma is identical to the FRT setting, so we won't reprove it here.

Lemma 17.3.7 *We can assign a value b_w to each $w \in V$ such that:*

1. $\mathbf{Pr}[S_w = 1 | X_w = 1] \leq b_w$, *and*

2. $\sum_{w \in V} b_w \leq O(\log n)$

The next lemma is slightly different, since it's actually simpler than the FRT setting since there's only one level.

Lemma 17.3.8 $\mathbf{Pr}[X_w = 1] \leq 4d(u,v)/\delta$ *for all $w \in V$*

Proof: As in FRT, $\mathbf{Pr}[X_w = 1]$ depends only on the random choice of r, not on π. WLOG, assume that $d(w, u) \leq d(w, v)$. Then in order for w to cut u, v, it must be the case that $d(w, u) \leq r < d(w, v)$. Since r is distributed uniformly in $[\delta/4, \delta/2]$, this happens with probability at most

$$\frac{d(w, v) - d(w, u)}{\delta/4} \leq \frac{4d(u, v)}{\delta},$$

as claimed. ∎

Now we put these together, and we get that

$$
\begin{aligned}
\mathbf{Pr}[u, v \text{ in different clusters}] &= \sum_{w \in V} \mathbf{Pr}[S_w = 1 | X_w = 1]\mathbf{Pr}[X_w = 1] \\
&\leq \sum_{w \in V} b_w \frac{4d(u, v)}{\delta} \\
&\leq \frac{d(u, v)O(\log n)}{\delta}
\end{aligned}
$$

So we have our LDRD algorithm!

References

[CKR00] Gruia Călinescu, Howard Karloff, and Yuval Rabani. An improved approximation algorithm for multiway cut. *Journal of Computer and System Sciences*, 60(3):564 – 574, 2000.