Advanced Algorithms 2024	25.11, 2024
Sample Solutions 10	
Lecturer: Bernhard Haeupler	*Teaching Assistant: Andor Vári-Kakas*

1 L1 embedding of cycle

You are given an unweighted cycle C on n vertices.

1. Find a randomized algorithm that embeds the cycle C to $\mathbb{R}$ such that the expected stretch of every edge is constant. That is, your randomized algorithm maps each vertex $u \in C$ to some number $f(u)$. For every pair $u, v \in C$ it has to be the case that $d_C(u,v)/K \leq \mathbb{E}[|f(u) - f(v)|] \leq d_C(u,v)$ for some constant K.
2. Find a deterministic algorithm that embeds C to $\mathbb{R}^2$ with $L1$ norm such that the stretch of every edge is constant. That is, you should map each vertex $u \in C$ to some number $f(u)$. For every pair $u, v \in C$ it has to be the case that $d_C(u,v)/K \leq ||f(u) - f(v)||_1 \leq d_C(u,v)$ for some constant K.

Solution

1.1 Cycle embedding in $\mathbb{R}$ with constant expected stretch

We want to embed a cycle $C = (V, E)$ into $\mathbb{R}$ such that the expected stretch is constant, or more formally, define a mapping: $f : V \to \mathbb{R}$ and two constants $K_1, K_2 \geq 1$ such that:

$$\forall u, v \in V \ \frac{d_C(u,v)}{K_1} \leq \mathbb{E}[|f(u) - f(v)|] \leq K_2 \cdot d_C(u,v). \tag{1}$$

One possible way to do that is as follows. We choose one vertex of V at random and call it v_1. Then traverse the cycle in one direction and label each node with consecutive numbers $v_2, v_3, \ldots, v_n$. Having done that we can define f as $f(v_i) = i$. This construction is illustrated in Figure **??**. In the remainder of this section let “jump edge” be the edge $\{v_n, v_1\}$, i.e., the edge that is mapped to a “jump” of length $n-1$ in $\mathbb{R}$. Now we just need to prove the two inequalities from Equation 1.

Claim: $d_C(u,v) \leq |f(u) - f(v)|$

Proof: In a cycle with n nodes the highest possible distance between two nodes is $\lfloor n/2 \rfloor$. If we take any pair of nodes u and v in a cycle and consider their mappings in $\mathbb{R}$, we can analyze two cases. In the first case, the shortest path between u and v in C does not include the jump edge, in which case $|f(u) - f(v)| = d_C(u,v)$. In the second case, the shortest path includes the jump edge between v_1 and v_n, in which case $|f(u) - f(v)| \geq \lfloor n/2 \rfloor \geq d_C(u,v)$. That is, the distance along the $\mathbb{R}$ axis is at least as long as the shortest path in the cycle. This finishes the argument.

Claim: $\mathbb{E}[|f(u) - f(v)|] \leq 2 \cdot d_C(u,v)$

Proof: Let us fix a pair of vertices u and v on the cycle. We know that there are exactly $d_C(u,v)$ edges between them and exactly n edges in the cycle in total. The question is where is the jump edge. Any edge in the cycle can be mapped to the jump edge with uniform probability. Therefore the probability that it is on the shortest path between u and v is exactly $d_C(u,v)/n$ and the probability that it is outside that shortest path is $1 - d_C(u,v)/n$. If the jump edge is outside the shortest path then $|f(u) - f(v)| = d_C(u,v)$. If it is inside, we can still use the trivial upper bound $|f(u) - f(v)| \leq n$.

The full expression for the expected value is therefore:

$$\begin{aligned}\mathbb{E}[|f(u) - f(v)|] &\leq \frac{d_C(u,v)}{n} \cdot n + \left(1 - \frac{d_C(u,v)}{n}\right) \cdot d_C(u,v) \\ &\leq d_C(u,v) + d_C(u,v) \\ &= 2 \cdot d_C(u,v).\end{aligned}$$

1.2 Cycle embedding in $\mathbb{R}^2$ with constant deterministic stretch

In this task we are asked to define a mapping $f : V \to \mathbb{R}^2$ and two constants $K_1, K_2 \geq 1$ such that:

$$\forall u, v \in V : \frac{d_C(u,v)}{K_1} \leq \|f(u) - f(v)\|_1 \leq K_2 \cdot d_C(u,v).$$

Choose an arbitrary vertex of the cycle and call it v_1. Then traverse the cycle in one direction and label each node with consecutive numbers $v_2, v_3, \ldots, v_n$. With all nodes labelled, we will now describe the map f from V to a square in $\mathbb{R}^2$. Let $a = \lfloor n/4 \rfloor$ be the side of the square. Now we map v_1 to $(1, a)$ and in general for $i = \{1, \ldots, \lfloor n/4 \rfloor - 1\}$ we set $f(v_i) = (i, a)$. We proceed analogously for the remaining sides of the square. If $n \leq 4$ we just map nodes to corners (1, 2 or 3). If n is not divisible by 4, we make the last side "denser" such that for $i \in \{3 \cdot \lfloor n/4 \rfloor + 1, \ldots, n\}$ the $\|f(v_{i+1}) - f(v_i)\|_1 \leq 1$. This construction is illustrated in Figure **??**. Now we need to prove the two inequalities that define the constant stretch.

Claim: $d_C(u,v)/3 \leq \|f(u) - f(v)\|_1$

Proof: If we fix a pair of nodes u and v then we may consider exactly three cases:

- $f(u)$ and $f(v)$ are on the same side of the square. In this case $d_C(u,v) = \|f(u) - f(v)\|_1$.
- $f(u)$ and $f(v)$ are on neighbouring sides of the square: $d_C(u,v) = \|f(u) - f(v)\|_1$. Strictly speaking the L_1 norms on the last side of the square may be a bit smaller but still larger than $d_C(u,v)/2$. The smallest possible distance between two consecutive nodes in this construction comes with $n = 7$, where $a = 1$ but the last side has 3 nodes spaced at distances 1/3 from each other. In this case for u and v on the last side $d_C(u,v)/3 = \|f(u) - f(v)\|_1$
- $f(u)$ and $f(v)$ are on opposite sides of the square: the norm is at least the length of the side of the square $\|f(u) - f(v)\|_1 \geq \lfloor n/4 \rfloor \geq d_C(u,v)/2 \geq d_C(u,v)/3$. The penultimate step is correct due to the fact that in any cycle, for any pair of nodes $d_C(u,v) \leq \lfloor n/2 \rfloor$ which means that $d_C(u,v)/2 \leq \lfloor n/2 \rfloor/2 \leq \lfloor n/4 \rfloor$.

Claim: $\|f(u) - f(v)\|_1 \leq d_C(u,v)$ This holds for any two consecutive vertices u, v on the cycle and then can be extended to any two vertices via triangle inequality.

2 Minimum bisection cut

A *bisection cut* is a cut (S, S') such that $|S| = |S'| = n/2$. An *r-balanced cut* is a cut where $r \cdot n \leq |S| \leq (1-r) \cdot n$. A size of a cut is the number of edges that go across the cut.

Give a polynomial-time algorithm that, given a graph G as input, outputs a 1/3-balanced cut whose size is $\mathcal{O}(\log n)$ factor from the size of the smallest-size bisection cut of G.

Hint:

Find a black box reduction to the result you saw in the lecture via a greedy algorithm.

Solution

We will base the solution on the sparsest cut finding algorithm we have seen in the lecture notes. SparsestCut(G) finds a cut $S \in V(G)$ that gives $O(\log n)$ approximation for minimum value of the following:

$$\frac{|E(S, V \setminus S)|}{|S| \cdot |V \setminus S|}.$$

Let us first modify SparsestCut(G) algorithm slightly so that it always returns a cut S of the smaller size (i.e. if it would have returned S such that $|S| > \frac{|V(G)|}{2}$, let it return $V \setminus S$ instead).

Let's introduce the following sequence:

$$S_1 = \textsc{SparsestCut}(G[V])$$
$$S_2 = \textsc{SparsestCut}(G[V \setminus S_1]) \qquad G[V \setminus W] \textit{ is the induced subgraph of } G \textit{ restricted to } V \setminus W$$
$$S_3 = \textsc{SparsestCut}(G[V \setminus (S_1 \cup S_2)])$$
$$\ldots$$
$$S_{i+1} = \textsc{SparsestCut}(G[V \setminus \bigcup_{j=1}^{i} S_i])$$
$$\ldots$$

Claim 1. *There exists t such that $1 \leq t \leq n$ and $\frac{n}{3} \leq |\bigcup_{i=1}^{t} S_i| \leq \frac{2 \cdot n}{3}$.*

Proof. Note that $|S_{k+1}| > |S_k|$ for all k such that $|\bigcup_{i=1}^{k} S_i| \leq n - 2$. Now let j be the maximum index such that $|\bigcup_{i=1}^{j} S_i| < \frac{n}{3}$. We have

$$
\begin{aligned}
|\bigcup_{i=1}^{j+1} S_i| &= |(\bigcup_{i=1}^{j} S_i) \cup S_{j+1}| \\
&= |(\bigcup_{i=1}^{j} S_i) \cup \textsc{SparsestCut}(G[V \setminus \bigcup_{i=1}^{j} S_i])| \\
&= |(\bigcup_{i=1}^{j} S_i)| + |\textsc{SparsestCut}(G[V \setminus \bigcup_{i=1}^{j} S_i])| \\
&\leq |(\bigcup_{i=1}^{j} S_i)| + \frac{|V \setminus \bigcup_{i=1}^{j} S_i|}{2} && \textit{Because } \textsc{SparsestCut}(\bullet) \leq \frac{|V(\bullet)|}{2} \\
&= |(\bigcup_{i=1}^{j} S_i)| + \frac{|V|}{2} - \frac{|\bigcup_{i=1}^{j} S_i|}{2} \\
&= \frac{|V|}{2} + \frac{|(\bigcup_{i=1}^{j} S_i)|}{2} < \frac{n}{2} + \frac{\frac{n}{3}}{2} = \frac{2 \cdot n}{3}
\end{aligned}
$$

as needed. □

Claim 2. *Let t be the smallest integer such that $\frac{n}{3} \leq |\bigcup_{i=1}^{t} S_i| \leq \frac{2 \cdot n}{3}$. Then $\bigcup_{i=1}^{t} S_i$ is a 1/3-balanced cut whose size is $O(\log n)$ factor from the size of the smallest bisection cut of G. In other words, $\bigcup_{i=1}^{t} S_i$ is the 1/3-balanced cut we are looking for.*

Proof. Denote $(V \setminus \bigcup_{j=1}^{i-1} S_j)$ by R_i for $1 \leq i \leq t$. Let T be the minimum bisection cut of G, and C_i (for $1 \leq i \leq t$) be the sparsest cut in $G[R_i]$.

By the construction, S_i is $O(\log n)$ approximation of the sparsest cut in $G[R_i]$, hence:

$$
\frac{E(S_i, R_i \setminus S_i)}{|S_i| \cdot |R_i \setminus S_i|} \leq \frac{E(C_i, R_i \setminus C_i)}{|C_i| \cdot |R_i \setminus C_i|} \cdot O(\log n) \tag{2}
$$

Because C_i is a sparsest cut in $G[R_i]$, we have:

$$
\begin{aligned}
\frac{E(C_i, R_i \setminus C_i)}{|C_i| \cdot |R_i \setminus C_i|} \cdot O(\log n) &\leq \frac{E(T \cap R_i, (V \setminus T) \cap R_i)}{|T \cap R_i| \cdot |(V \setminus T) \cap R_i|} \cdot O(\log n) \\
&\leq \frac{E(T, V \setminus T)}{|T \cap R_i| \cdot |(V \setminus T) \cap R_i|} \cdot O(\log n)
\end{aligned} \tag{3}
$$

Recall t is the smallest integer such that $\frac{n}{3} \leq |\bigcup_{j=1}^{t} S_j| \leq \frac{2 \cdot n}{3}$. Hence, $|R_i| = |V \setminus \bigcup_{j=1}^{i-1} S_j| \geq |V \setminus \bigcup_{j=1}^{t-1} S_j| \geq \frac{2 \cdot n}{3}$ for all $1 \leq i \leq t$. So, for $1 \leq i \leq t$, $|T \cap R_i| \geq \frac{2 \cdot n}{3} - \frac{n}{2} = \frac{n}{6}$ and $|(V \setminus T) \cap R_i| \geq \frac{2 \cdot n}{3} - \frac{n}{2} = \frac{n}{6}$.

From eq. (2) and eq. (3) we get:

$$
\begin{aligned}
E(S_i, R_i \setminus S_i) &\le \frac{E(T, V\setminus T)}{|T \cap R_i| \cdot |(V\setminus T)\cap R_i|} \cdot O(\log n)\cdot |S_i| \cdot |R_i \setminus S_i| \\
&\le \frac{E(T, V\setminus T)}{\frac{n}{6}\cdot\frac{n}{6}} \cdot O(\log n) \cdot |S_i| \cdot n = \frac{E(T, V\setminus T)}{n} \cdot O(\log n) \cdot |S_i|
\end{aligned}
\tag{4}
$$

Finally we calculate the cut-size of $\bigcup_{i=1}^{t} S_i$.

$$
\begin{aligned}
E(\bigcup_{i=1}^{t} S_i, V \setminus \bigcup_{i=1}^{t} S_i) &= E(\bigcup_{i=1}^{t} S_i, R_t \setminus S_t) \\
&= \sum_{i=1}^{t} E(S_i, R_t \setminus S_t) \\
&\le \sum_{i=1}^{t} E(S_i, R_i \setminus S_i) && \textit{Because } (R_t \setminus S_t) \subseteq (R_i \setminus S_i) \\
&\le \sum_{i=1}^{t} (\frac{E(T, V\setminus T)}{n} \cdot O(\log n) \cdot |S_i|) && \textit{By eq. } (4) \\
&= (\sum_{i=1}^{t} |S_i|) \cdot \frac{E(T, V\setminus T)}{n} \cdot O(\log n) \\
&\le \frac{2\cdot n}{3} \cdot \frac{E(T, V\setminus T)}{n} \cdot O(\log n) \\
&= E(T, V\setminus T) \cdot O(\log n)
\end{aligned}
\tag{5}
$$

That is, the cut-size of $\bigcup_{i=1}^{t} S_i$ is $O(\log n)$ factor from the size of the smallest bisection cut T. □

Claim 3. *$\bigcup_{i=1}^{t} S_i$ can be computed in poly(n) time.*

Proof. In order to calculate $\bigcup_{i=1}^{t} S_i$, we run SparsestCut algorithm t times. Each run has time complexity $poly(n)$, thus the complete algorithm has time complexity $O(t \cdot poly(n)) = O(n \cdot poly(n)) = O(poly(n))$. □

To summarize, our algorithm is:

Find a cut S that is $O(\log n)$ approximation of the sparsest cut in G (s.t. $|S| \le \frac{|V|}{2}$). If S is 1/3-balanced cut, we have found the solution and stop here. Otherwise, find a sparsest cut in $G[V \setminus S]$ and add it to S. Repeat the procedure until a 1/3-balanced cut is found.