Problem Set 3 Solutions

姓名: 黄睿
学号: MP1933008

2019-11-16

1 Problem 1

1.1 Q1

Algorithm 1 Greedy algorithm for max k-cut

Input: G(V,E)

Output: $S_1, S_2, \cdots, S_k$

1: Init $S = \{S_1, S_2, \cdots, S_{|V|}\}$ with $S_i = \{v_i\}, i = 1, 2, \cdots, |V|$;

2: **while** $|S| \geq k$ **do**

3: pick S_i and S_j with edges between them have **smallest weight**, contract S_i and S_j into $S_{\min(i,j)}$.

4: **return** $S_1, S_2, \cdots, S_k$

To analyze the approximation ratio, we define that on each contraction the weight of edges is $W_i, i = 1, 2, \cdots, |V| - k$.As we take the greedy strategy, it's obvious that there is monitonicity, shows that: $W_1 \leq W_2 \leq \cdots \leq W_{|V|-k}$, we also have:

$$\begin{aligned} W_i &\leq \frac{\sum_{e \in E} W_e - \sum_{j=1}^{i-1} W_j}{\binom{|V|-i+1}{2}} \\ &\leq \frac{\sum_{e \in E} W_e - (i-1) W_i}{\binom{|V|-i+1}{2}} \end{aligned} \tag{1}$$

simplify the inequality above:

$$W_i \leq \frac{2 \sum_{e \in E} W_e}{(|V| - i)(|V| - i + 1)}$$

the term on the right side is the average weight, and we can make it sure that:$OPT \leq \sum_{e \in E} W_e$.

The greedy algorithm will produce:

$$
\begin{aligned}
SOL &= \sum_{e\in E} W_e - \sum_{i=1}^{|V|-k} W_i \\
&\geq \sum_{e\in E} W_e - \sum_{i=1}^{|V|-k} \frac{2\sum_{e\in E} W_e}{(|V|-i)(|V|-i+1)} \\
&= \sum_{e\in E} W_e \left[1 - 2\sum_{i=1}^{|V|-k}\left(\frac{1}{|V|-i} - \frac{1}{|V|-i+1}\right)\right] \\
&= \sum_{e\in E} W_e \left[1 - 2\left(\frac{1}{k} - \frac{1}{|V|}\right)\right] \\
&\geq \left[1 - 2\left(\frac{1}{k} - \frac{1}{|V|}\right)\right] \cdot OPT
\end{aligned} \tag{2}
$$

1.2 Q2

The missing code: move v to S_{1-i} will yield a larger weighted cut;

The time complexity of this algorithm is : $O(|V|^2 \sum_{e\in E} w_e)$, for each round of the while loop involves examining at most $|V|$ vertices and selecting one that increases the cut value when moved.This process takes $O(|V|^2)$ time. Since we assume that edges have positive integral weights, the cut value is increased by at least 1 after each iteration.The maximum possible cut value is $\sum_{e\in E} w_e$.Combining all of them, we get the total running time: $O(|V|^2 \sum_{e\in E} w_e)$.

To analyze the approximation ratio, we first notice that for any max 2-cut:

$$
w(S_0, S_1) = \sum_{\substack{uv\in E \\ u\in S_0, v\in S_1}} w(uv) \leq OPT \leq \sum_{e\in E} w(e)
$$

According to the local search algorithm, it will end with: for any $u \in S_0$, moving u to S_1 will not increase the cut value, thus:

$$
\sum_{\substack{u\in S_0 \\ uv\in E}} w(uv) \geq \sum_{\substack{u\in S_1 \\ uv\in E}} w(uv)
$$

$$
2\sum_{\substack{u\in S_0 \\ uv\in E}} w(uv) \geq \sum_{\substack{u\in S_0 \\ uv\in E}} w(uv) + \sum_{\substack{u\in S_1 \\ uv\in E}} w(uv) = \sum_{u:uv\in E} w(uv) \tag{3}
$$

By symmetry, we have:

$$2 \sum_{\substack{v \in S_1 \\ uv \in E}} w(uv) \geq \sum_{v : uv \in E} w(uv) \tag{4}$$

Combining 3 and 4, we have:

$$\begin{aligned} w(S_0, S_1) &= \left(\sum_{\substack{u \in S_0 \\ uv \in E}} w(uv) + \sum_{\substack{v \in S_1 \\ uv \in E}} w(uv) \right) \\ &\geq 0.5 \cdot \sum_{e \in E} w(e) \\ &\geq 0.5 \cdot OPT \end{aligned} \tag{5}$$

Thus, we proved the approximation ratio is 0.5.

2 Problem 2

Algorithm 2 Greedy algorithm for max k coverage

Input: $S_1, S_2, \cdots, S_m$
Output: C
1: Init $C = \emptyset$;
2: **while** $|C| < k$ **do**
3: add i with largest $|S_i \cap U|$ to C;
4: $U = U \backslash S_i$;
5: **return** C

Let C_i be the elements covered in i coverage, where $i = 1, 2, \cdots, k$.We can prove the approximation ratio by perform induction on i.

If $i = 1$, It's trivial that: $C_1 \geq 1 - \left(1 - \frac{1}{k}\right)^1 \cdot OPT$. By taking greedy strategy we have each step, we cover **at least** $\frac{1}{k}$ of uncovered elements.Thus,

$$
\begin{aligned}
C_{i+1} &\geq C_i + \frac{n - C_i}{k} \\
&= \left(1 - \frac{1}{k}\right) C_i + \frac{n}{k} \\
&\geq \left(1 - \frac{1}{k}\right) \left[1 - \left(1 - \frac{1}{k}\right)^i\right] \cdot OPT + \frac{n}{k} \\
&= \left[1 - \left(1 - \frac{1}{k}\right)^{i+1}\right] \cdot OPT + \frac{n}{k} - \frac{OPT}{k} \qquad (n \geq OPT) \\
&\geq \left[1 - \left(1 - \frac{1}{k}\right)^{i+1}\right] \cdot OPT
\end{aligned} \tag{6}
$$

Let $i + 1 = k$, we have:

$$
C_k \geq \left[1 - \left(1 - \frac{1}{k}\right)^k\right] \cdot OPT
$$

3 Problem 3

3.1 Q1

The expected size of random cut is given by:

$$
\begin{aligned}
\mathbb{E}\left[|E(S,T)|\right] &= \sum_{(u,v)\in E} \Pr\left[u \in S, v \in T\right] \\
&= \sum_{(u,v)\in E} \Pr\left[u \in S\right] \cdot \Pr\left[v \in T\right] \\
&= \sum_{(u,v)\in E} \frac{1}{2} \cdot \frac{1}{2} \\
&= \frac{1}{4} \cdot OPT
\end{aligned} \tag{7}
$$

According to what we have shown above, the approximation of random cut is $\frac{1}{4}$.

3.2 Q2

Similar to Q1:

$$
\begin{aligned}
\mathbb{E}\left[|E(S,T)|\right] &= \sum_{(u,v)\in E} \Pr\left[u \in S, v \in T\right] \\
&= \sum_{(u,v)\in E} \Pr\left[u \in S\right] \cdot \Pr\left[v \in T\right] \\
&= \sum_{(u,v)\in E} \left(\frac{1}{4} + \frac{x_u^*}{2}\right) \cdot \left(1 - \left(\frac{1}{4} + \frac{x_v^*}{2}\right)\right) \\
&= \sum_{(u,v)\in E} \left(\frac{1}{4} + \frac{x_u^*}{2}\right) \cdot \left(\frac{1}{4} + \frac{1 - x_v^*}{2}\right) \\
&\geq \sum_{(u,v)\in E} \left(\frac{1}{4} + \frac{y_{u,v}^*}{2}\right)^2 \\
&= \sum_{(u,v)\in E} \left(\frac{1}{4} - \frac{y_{u,v}^*}{2}\right)^2 + \frac{y_{u,v}^*}{2} \\
&\geq \sum_{(u,v)\in E} \frac{y_{u,v}^*}{2} \\
&= \frac{1}{2} \cdot OPT_{LP}
\end{aligned}
\tag{8}
$$

We have:

$$
\mathbb{E}\left[|E(S,T)|\right] \geq \frac{1}{2} \cdot OPT_{LP} \geq \frac{1}{2} OPT
$$

the approximation ratio is expected to be $\frac{1}{2}$.

4 Problem 4

4.1 Q1

The expected number of satisfied clauses is:

$$
\begin{aligned}
E\left[\#\text{ of satisfied clauses}\right] &= \sum_{j=1}^{m} \Pr\left[C_j \text{ is satisfied}\right] \\
&= \sum_{j=1}^{m} \left[1 - \prod_{i \in S_j^+} (1 - f(x_i^*)) \cdot \prod_{i \in S_j^-} f(x_i^*)\right] \\
&\geq \sum_{j=1}^{m} \left[1 - \prod_{i \in S_j^+} 4^{-x_i^*} \prod_{i \in S_j^-} 4^{x_i^*-1}\right] \\
&= \sum_{j=1}^{m} \left[1 - 4^{\sum_{i \in S_j^-}(x_i^*-1) - \sum_{i \in S_j^+} x_i^*}\right] \\
&\geq \sum_{j=1}^{m} \left[1 - 4^{-y_j^*}\right] \\
&\geq \sum_{j=1}^{m} \left(1 - \frac{1}{4}\right) y_j^* \qquad (convexity) \\
&\geq \frac{3}{4} \cdot OPT
\end{aligned}
\tag{9}
$$

4.2 Q2

4.3 Q3

It's not possible.

5 Problem 5

5.1 Q1

Let $x_j^i = 1$ if element j in subset S_i, otherwise $x_j^i = 0$, let $y_i = 1$ if $i \in C$, otherwise $y_i = 0$. The integer program can be modeled as below:

$$\begin{aligned} \text{minimize} \ & \sum_{i=1}^{m} w_i \cdot y_i \\ \text{subject to} \ & \sum_{i=1}^{m} x_j^i \geq 1 \\ & x_j^i \in \{0,1\}, \quad 1 \leq j \leq n, 1 \leq i \leq m \\ & y_i \in \{0,1\}, \quad 1 \leq i \leq m \end{aligned} \tag{10}$$

By performing LP relaxation, the integer program transforms to LP:

$$\begin{aligned} \text{minimize} \ & \sum_{i=1}^{m} w_i \cdot y_i \\ \text{subject to} \ & \sum_{i=1}^{m} x_j^i \geq 1 \\ & x_j^i \in [0,1], \quad 1 \leq j \leq n, 1 \leq i \leq m \\ & y_i \in [0,1], \quad 1 \leq i \leq m \end{aligned} \tag{11}$$

5.2 Q2