# الگوریتم های تقریبی(جلسه ی هفتم)

فرنام منصوری

۲۰۱۷ October

## ۱ اثبات مسئلهِ ی درخت اشتاینر از جلسه ی قبل

صورت مسئله این بود که یک راس r داشتیم و هر یال یک هزینه داشت میخواستیم با کم ترین هزینه تعدادی یال انتخاب کنیم که با آن ها از s به مجموعه ی دلخواه از که قبلا داده شده از راس ها بتوان رسید. در جلسه ی قبل LP و دوگان آن را برای مسئله نوشتیم. :صورت مسئله ی دوگان

$$max \sum_{s \in A} z_s$$
$$\forall e : \sum_{S : e \in \delta(S)} z_s \leq c_e$$
$$z_s \geq 0$$

و گفتیم که اول به ازای تمام مجموعه های تک عضوی مقدار $z_s$ را در هر لحظه به مقدار ثابتی اضافه میکنیم تا اینکه دو تا از مجموعه ها مجموع $z_s$ شان برابر با یک یال شد آنوقت آن یال را به مجموعه یال هایمان اضافه میکنیم بعد دو مجموعه ای که مانع ادامه ی کار شدن را با هم merge می کنیم

$$\sum_e x_e c_e = \sum_e \sum_{S : e \in \delta(S)} z_s x_e = \sum_{S \in A} z_s ( \sum_{e \in \delta(S)} x_e) \quad (۱)$$

حالا مقدار $\sum_{S \in A} z_s (\sum_{e \in \delta(S)} x_e)$ در نظر بگیرید فرض کنید مقادیر $z_s$ ها با سرعت $c$ بر ثانیه در حال افزایش است و سیگمای دوم هم در عبارت به معنی تعداد یال های متصل به یک راس است در نتیجه در اول این مقدار به اندازه ی مجموع درجات گراف ضرب در $c$ کم میشود و چون درخت است میشود $2(n1)c$ و هر لحظه که یک یال اضافه می شود و در واقع دو تا راس merge می شوند و به عبارت اگر k تا راس مانده باشد $2(k-1)c$ زیاد می شود در حالی که اگر k راس مانده باشد k تا $z_s$ در حال اضافه شدن اند پس به $2 \sum_s z_s$ هر ثانیه $2kc$ اضافه می شود پس هر لحظه $\sum_{S \in A} z_s (\sum_{e \in \delta(S)} x_e)$ بیشتر از $2 \sum_s z_s$ اضافه میشود و در اول هم همه ی $z_s$ ها صفر اند پس هر دو صفر اند پس در آخر $\sum_{S \in A} z_s (\sum_{e \in \delta(S)} x_e)$ تر است

$$\sum_e x_e c_e = \sum_e \sum_{S : e \in \delta(S)} z_s x_e = \sum_{S \in A} z_s ( \sum_{e \in \delta(S)} x_e) \leq 2 \sum_s z_s = 2z_D^* = 2z_{LP}^* \leq 2 * OPT \quad (۲)$$

## ۲ مسئله ی مکان یاب تجهیزات بدون ظرفیت

### ۱.۲ صورت مسئله

یکسری تجهیزات داریم و یکسری آدم، هزینه‌ی خرید تجهیزات $i$ ام $f_i$ است و هزینه ی استفاده از تجهیزات $i$ ام برای فرد $j$ ام $c_{ij}$ است. میخواهیم یک تعدادی از تجهیزات را بخریم طوری که مجموع هزینه تجهیزات و مجموع هزینه ی تجهیز کردن افراد از تجهیزاتی که هزینه اش مینیمم است کمینه شود. از طرفی می دانیم که نامساوی مثلثی برقرار است، یعنی در واقع برای تجهیزات $l, j$ و آدم های $i, k$، $c_{ij} \leq c_{il} + c_{kl} + c_{kj}$

## ۲.۲ راه حل

اول IP را مینویسیم:

$$min \sum_i f_i y_i + \sum c_{ij} x_{ij}$$

$$subject\ to \qquad \sum_{i \in F} x_{ij} = 1, \qquad \forall j \in D$$

$$x_{ij} \leq y_i, \qquad \forall i \in F, j \in D$$

$$x_{ij} \in \{0,1\} \qquad \forall i \in F, j \in D$$

$$y_i \in \{0,1\} \qquad \forall i \in F$$

حال تبدیل به LP میکنیم:

$$min \sum_i f_i y_i + \sum c_{ij} x_{ij}$$

$$subject\ to \qquad \sum_{i \in F} x_{ij} = 1, \qquad \forall j \in D$$

$$x_{ij} \leq y_i, \qquad \forall i \in F, j \in D$$

$$0 \leq x_{ij} \qquad \forall i \in F, j \in D$$

$$0 \leq y_i \qquad \forall i \in F$$

توجه کنید که اگر شرط کوچک تر از ۱ را نمینوشتیم در جواب مسئله تاثیر داشت، حال مسئله ی dual را مینویسنم:

$$max \sum_{j \in D} v_j$$

$$subject\ to \qquad \sum_{j \in D} w_{ij} = f_i, \qquad \forall i \in F$$

$$v_j - w_{ij} \leq c_{ij}, \qquad \forall i \in F, j \in D$$

$$w_{ij} \geq 0, \qquad \forall i \in F, j \in D$$

### ۱.۲.۲ مرحله ی یک

اول$(x^*, y^*)$ جواب $LP$ را حل میکنیم و سپس $(v^*, w^*)$ جواب $D$ را حل میکنیم

### ۲.۲.۲ مرحله ی دوم

اگر $x_{ij} > 0$باشد، $v_i - w_{ij} = c_{ij}$ در نتیجه $v_j > c_{ij}$ // پس به این نتیجه میرسیم که اگر $N_j$ را تعریف کنیم تمام $i$ هایی که $x_{ij} > 0$ و یک مجموعه مثل S از تجهیزات داشته باشیم که به ازای هر j $N(j)$ با S اشتراک داشته باشد، کافیست یکی از $x_{ij}$ های بزرگتر ازصفر در S را یک در نظر گیریم.

$$\sum c_{ij} x_{ij} > \sum_j v_j = z_D^* = OPT \tag{۳}$$

و فقط کافیست خوبیS بگیریم که جمع اول خوب شود

### ۳.۲.۲ مرحله ی سوم

فرض کنید که مجموعه ی تجهیزات را به چند قسمت تقسیم کردیم که هر قسمت $N(j)$ برای یک j باشد و یک سری تجهیزات هم درون هیچ یک از این مجموعه ها نباشد اسم این مجموعه ها را $F_1, F_2, , F_t$ می‌نامیم و کم هزینه ترین تجهیزات را در مجموعه ی i ام را $f_{i_k}$ می‌نامیم:

$$f_{i_k} = f_{i_k} \sum_{i \in F} x^*_{ij} = \sum_{i \in F_k} f_{i_k} x^*_{ij} \leq \sum_{i \in F_k} f_i x^*_{ij} \leq \sum_{i \in F_k} f_i y^*_i \quad (۴)$$

در نتیجه :

$$\sum_i f_{i_k} = \sum_{i \in F} f_i y^*_i \leq Z^*_{LP} = OPT \quad (۵)$$

### ۴.۲.۲ الگوریتم

۱. $(x^*, y^*)$ و $(v^*, w^*)$ را پیدا کن

۲. $C \to D$

۳. $while\, C \neq \oslash$

   (آ) $k \to k+1$

   (ب) $argmin_{j \in D^*} v^*_j = j_k$

   (ج) $C = C - j_k - N^2(j_k)$

۴. جواب برابر است با $O = i_k | i_k = mini \in N(j_k)$

### ۵.۲.۲ اثبات الگوریتم

و برای هر j ای که به صورت $j_k$ ها نیست اولین $F_k$ ای که در آن همسایه که در نظر بگیرید ، یک همسایه در $F_k$ مثل s داریم و:

$$c_{i_k j} \leq c_{sj} + csj_k + xi_k j_k \leq v^* j + 2v^* j_k \leq 3 * v^* j \quad (۶)$$

اگر هم j به صورت $j_k$ باشد هم بدیهتا $c_{ij_k} < v^*_{j_k}$

$$\sum_{i_k \in O, j \in D} c_{i_k j} = \sum_{i_k \in O, j_k} c_{i_k j_k} + \sum_{i_k \in O, j \neq j_k} c_{i_k j} \leq \sum_{j_k} v^*_{j_k} + \sum_{j \neq j_k} 3 * v^*_j = \sum_{j \in D} v^*_j = 3 * OPT \quad (۷)$$

از جمع عبارت ۵ که در مرحله ی ۳ به دست آوردیم و عبارت بالا به دست نتیجه می رسیم که

$$\sum y_i + \sum c_{ij} \leq OPT + 3 * OPT = 4 * OPT \quad (۸)$$

پس الگوریتم $4-$تقریب است