Санкт-Петербургский государственный университет

БАДАЖКОВА Ольга Александровна

Выпускная квалификационная работа

Пропорциональный механизм в задаче k-медианой кластеризации

Образовательная программа: бакалавриат «Математика»
Направление и код: 01.03.01 «Математика»
Шифр ОП: СВ.5000.2017

Научный руководитель:
доцент факультета математики
и компьютерных наук,
PhD university of Geneva
Калинин Никита Сергеевич

Рецензент: professor, Institute
for theoretical Computer Science,
School of information
Management and Engineering,
Shanghai University of Finance
and Economics, PhD, кандидат
физико-математических наук
Гравин Николай Вадимович

Санкт-Петербург
2021

Аннотация

В общем случае задача о размещении объектов (также известная как задача о нахождении k-центра) состоит в том, чтобы для n точек $\ell_1, \ell_2, \ldots, \ell_n$ в метрическом пространстве Ω найти k точек $f_1, f_2, \ldots, f_k$ таких, чтобы сумма

$$\sum_{i=1}^{n} \min_{j=1}^{k} d(\ell_i, f_j)$$

была минимальной.

В силу NP-сложности нахождения точного решения, интересны механизмы хорошо аппроксимирующие оптимум, такие как Пропорциональный механизм. В этой работе, используя метод k-медиан, мы докажем, что он аппроксимирует оптимум с константой $8 + 4\log k$.

Содержание

Введение

Задача размещения объектов в простейшем случае была поставлена ещё Пьером Ферма в первой половине XVII века. В формулировке Ферма она состаяла в том, чтобы для трёх данных точек A, B и C найти такую точку X, чтобы сумма расстояний $AX + BX + CX$ была минимальной. Изящный ответ на этот вопрос в 1645 году дал Эванджелиста Торричелли.

В более общем случае задача о размещении объектов (также известная как задача о нахождении k-центра) состоит в том, чтобы для n точек $\ell_1, \ell_2, \ldots, \ell_n$ — координат игроков в метрическом пространстве Ω найти k точек $f_1, f_2, \ldots, f_k$ — координат объектов таких, что сумма

$$\sum_{i=1}^{n} \min_{j=1}^{k} d(\ell_i, f_j)$$

была минимальной. Здесь через $d(\ell_i, f_j)$ обозначено расстояние между точками ℓ_i и f_j в Ω. Более подробная формулировка приведена в подразделе 1.1.

Как в 1981 году показали в работе [3] Фаулер, Патерсон и Танимото, в общем случае задача о размещении NP-трудна. Более того, как показано в статье [6] также NP-трудно получить алгоритм аппроксимирующий оптимум с константой 1,463 в предположении того, что $\mathrm{P} \neq \mathrm{NP}$.

Лучший на данный момент полиномиальный алгоритм, описанный Ли в 2011 году в статье [7] аппроксимирует оптимум с константой 1,488.

В этой работе Пропорциональный механизм работает естественным образом, принимая решения о строительстве объектов в k последовательных раундах. Каждый раз, мы выбираем следующее место в локации одного из игроков, давая каждому игроку шанс быть выбранным пропорционально расстоянию игрока от набора построенных на данный момент объектов.

Простота описания, полиномиальное время работы и возможность остановить работу алгоритма в любой момент делают этот механизм очень эффективным на практике.

Простые случаи ($k = 1$ и $k = 2$) были рассмотрены в работе [4]. В статьях [1] и [5] был рассмотрен общий случай работы пропорционального механизма и доказана оценка аппроксимации $4k$.

Также в работе [5] описан частный случай, для которого пропорциональный механизм не может аппроксимировать оптимум с константой лучшей, чем $O(\log k)$.

В этой работе мы докажем (см. теорему 3.1), что Пропорциональный механизм аппроксимирует оптимум с константой $O(\log k)$.

1 Постановка задачи

1.1 Задача о размещении объектов

Пусть (Ω, d) — метрическое пространство с метрикой d. В точках $\ell_1, \ell_2, \ldots, \ell_n$ этого пространства находятся n игроков. Обозначим через $\mathcal{L} = (\ell_1, \ell_2, \ldots, \ell_n)$ список координат этих игроков в Ω, а через $d(\ell_i, S)$ — расстояние от точки ℓ_i до множество S в Ω, иначе говоря минимум расстояний между ℓ_i и точками из S.

Игра «k-facility» заключается в том, что n игроков сообщают *планировщику* список $\mathcal{L}' = (\ell'_1, \ell'_2, \ldots, \ell'_n)$ своих координат в Ω (сообщённые координаты могут отличаться от реальных, но в этой работе мы будем считать, что игроки всегда сообщают своё настоящее положение, то есть $\mathcal{L} = \mathcal{L}'$). После этого, основываясь на полученной информации, планировщик устанавливает k *объектов* в точках множества $\mathcal{F}(\mathcal{L}') \subset \Omega$. Механизм выбора точек $\mathcal{F}$ может быть как детерминированным, так и недетерминированным. В последнем случае $\mathcal{F}(\mathcal{L}')$ — случайная величина, распределение которой также определяет механизм.

В детерминированном случае *цена* для игрока i равна $d(\ell_i, \mathcal{F}(\mathcal{L}'))$ — расстоянию от реального местоположения игрока до ближайшего установленного объекта. Обозначим её через

$$\mathrm{cost}_{\mathcal{F}}(i, \mathcal{L}') = d(\ell_i, \mathcal{F}(\mathcal{L}')) = \min_{f \in \mathcal{F}(\mathcal{L}')} d(\ell_i, f).$$

Аналогично, в случае с недетерминированным механизмом мы определим *цену* следующим образом:

$$\mathrm{cost}_{\mathcal{F}}(i, \mathcal{L}') = \mathbb{E}_{S \sim \mathcal{F}(\mathcal{L}')} \left[d(\ell_i, S) \right].$$

Задача о размещении объектов состоит в нахождении *оптимального* механизма $\mathcal{F}_0$, для которого

$$\sum_{i=1}^{n} \mathrm{cost}_{\mathcal{F}_0}(i, \mathcal{L}')$$

будет минимальной.

В общем случае задача NP-трудна (см. [3]). Поэтому интересны различные простые механизмы, *хорошо аппроксимирующие* (см. определение 1.1) минимум. Ясно также, что ложные сведения о своём местонахождении, которые могут сообщить игроки, стараясь минимизировать свою личную цену $\mathrm{cost}_{\mathcal{F}}(i, \mathcal{L}')$, весьма усложняют задачу. Встаёт вопрос о нахождении механизма не просто минимизирующего цену, но и *правдивого* (то есть для которого игрокам не выгодно сообщать ложное местоположение).

Обозначим теперь *общую цену* для всех игроков через

$$\mathrm{CS}_{\mathcal{F}}(\mathcal{L}) = \sum_{i=1}^{n} \mathrm{cost}_{\mathcal{F}}(i, \mathcal{L}).$$

Общую цену для оптимального механизма $\mathcal{F}_0$ обозначим через $\mathrm{opt}(\mathcal{L}) = \mathrm{CS}_{\mathcal{F}_0}(\mathcal{L})$.

1.2 Пропорциональный механизм

Определение 1.1. Будем говорить, что механизм $\mathcal{F}$ *аппроксимирует оптимум с константой* $\gamma = \gamma(k)$, если для любых $\ell_1, \ell_2, \ldots, \ell_n \in \Omega$ выполнено

$$\mathrm{CS}_{\mathcal{F}}(\ell_1, \ell_2, \ldots, \ell_n) \leqslant \gamma \cdot \mathrm{opt}(\ell_1, \ell_2, \ldots, \ell_n).$$

В этой работе мы будем рассматривать следующий простой механизм. Который обозначим через$\mathcal{F}_1$. Здесь и делее мы обозначаем через $[k] = \{1, 2, \ldots, k\}$ — множество первых k натуральных чисел.

Пропорциональный механизм
Data: $\ell_1, \ell_2, \ldots, \ell_n$
Result: установка *объектов*
равномерно случайно выбираем одну из точек $i \in [n]$;
$f_1 := \ell_i$, $S := \{i\}$, $N := [n] \setminus \{i\}$;
for $j = 2, \ldots, k$ **do**
  выбираем $i \in N$ случайно с вероятностью $\frac{d(l_i,S)}{\sum_{m=1}^{n} d(l_m,S)}$;
  $f_j := \ell_i$, $S := S \cup \{i\}$, $N := N \setminus \{i\}$;
end
устанавливаем *объекты* во всех точках $f_1, f_2, \ldots, f_k$;

Algorithm 1: Пропорциональный механизм

Практическое удобство этого механизма состоит, в частности, в том, что количество шагов (т.е. количество устанавливаемых объектов) не обязательно предопределять заранее. Работа механизма может быть остановлена на любом шаге, и порядок константы аппроксимации от этого не изменится (см. теорему 3.1). Это очень полезно, например, в случае построения в городе нескольких больниц, когда бюджет выделенный изначально на постройку может быть сокращён или увеличен.

Случаи $k = 1$ и $k = 2$ были рассмотрены в работе [4]. В ней же доказано, что при $k = 2$ подходит $\gamma = 4$. В общем случае в работе [1] было доказано, что Пропорциональный механизм аппроксимирует оптимум с константой $4k$. В этой работе мы усилим оценку до $O(\log k)$.

2 Предварительные сведения

Зафиксируем $\mathcal{L} = (\ell_1, \ell_2, \ldots, \ell_n)$ — список координат игроков в пространстве Ω. Пусть оптимальный механизм $\mathcal{F}_0$ устанавливает объекты в точках $f_1, f_2, \ldots, f_k$. Для каждого $i \in [k]$ рассмотрим C_i — множество игроков, для которых объект f_i — ближайший. Будем называть эти множества *кластерами.* Мы также будем употреблять термин «кластер», имея в виду не самих игроков, а их координаты.

Дальнейшие рассуждения мы будем проводить опираясь на разделение игроков на кластеры. Будем говорить, что кластер i *обслужен*, если в одной из точек C_i установлен объект; в противном случае будем говорить, что кластер *не обслужен.* Для всех $i \in [k]$ обозначим через $b_i = \sum_{j \in C_i} d(\ell_j, f_i)$ общую цену для игроков из кластера i в случае оптимальной установки объекта в кластере i.

Рассмотрим ситуацию в которой несколько объектов уже установлены. Для оценки математического ожидания общей цены для людей из одного кластера нам понадобится следующая лемма, приведённая в работе [5].

Лемма 2.1. *Пусть $\ell_1, \ell_2, \ldots, \ell_m$ — точки в Ω, а $d_1, d_2, \ldots, d_m$ — соответствующие расстояния от этих точек до ближайших к ним объектов. Выберем одну из точек ℓ_i случайно с вероятностью $\frac{d_i}{\sum_{j=1}^m d_j}$ и установим в неё ещё один объект. После этого для всех ℓ_i пересчитаем расстояние до ближайшего объекта. Обозначим через ξ случайную величину равную сумме расстояний до ближайших объектов, то есть $\xi = \sum_{j=1}^m \min\{d_j, d(\ell_j, \ell_i)\}$ с вероятностью $\frac{d_i}{\sum_{j=1}^m d_j}$.*

Тогда для произвольной точки $p \in \Omega$ выполнено

$$\mathbb{E}[\xi] \leqslant 4 \sum_{i=1}^m d(p, \ell_i).$$

Доказательство. Обозначим через $r_1, r_2, \ldots, r_m$ расстояния от p до точек $\ell_1, \ell_2, \ldots, \ell_m$ соответственно. И положим $d_{ij} = d(\ell_i, \ell_j)$. Тогда

$$\mathbb{E}[\xi] = \sum_{i=1}^m \frac{d_i}{\sum_{j=1}^m d_j} \sum_{j=1}^m \min\{d_j, d_{ij}\}.$$

Значит, нам нужно доказать, что

$$\sum_{i=1}^m \left[d_i \sum_{j=1}^m \min\{d_j, d_{ij}\} \right] \leqslant 4 \sum_{i=1}^m d_i \cdot \sum_{j=1}^m r_j. \tag{2.1}$$

Напишем следующую верхнюю оценку для левой части (2.1):

$$\sum_{i=1}^{m}\left[d_i\sum_{j=1}^{m}\min\{d_j,d_{ij}\}\right]\leqslant\sum_{i=1}^{m}\left[d_i\sum_{j=1}^{m}\min\{d_j,r_i+r_j\}\right]\leqslant$$
$$\leqslant\sum_{i=1}^{m}\left[d_i\sum_{j=1}^{m}r_j\right]+\sum_{i=1}^{m}\sum_{j=1}^{m}d_i\min\{d_j,r_i\}\leqslant$$
$$\leqslant\sum_{i=1}^{m}d_i\sum_{j=1}^{m}r_j+\sum_{i=1}^{m}\sum_{j=1}^{m}\left(\frac{1}{m}\sum_{v=1}^{m}(r_i+r_v+d_v)\right)\min\{d_j,r_i\}\leqslant$$
$$\leqslant\sum_{i=1}^{m}d_i\sum_{j=1}^{m}r_j+\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{v=1}^{m}((r_i+r_v)d_j+d_vr_i)\leqslant$$
$$\leqslant 4\sum_{i=1}^{m}d_i\sum_{j=1}^{m}r_j,$$

что завершает доказательство леммы. □

3 Основной результат

Теорема 3.1. *Пропорциональный механизм при $k\geqslant 3$ аппроксимирует оптимум с константой $\gamma(k)=24\log k$. Иначе говоря, для любого набора точек $\ell_1,\ell_2,\dots,\ell_n$ в Ω общая цена для для этого механизма*

$$\mathrm{CS}(\ell_1,\ell_2,\dots,\ell_n)\leqslant 24\log k\cdot\mathrm{opt}(\ell_1,\ell_2,\dots,\ell_n).$$

Замечание 3.2. *На самом деле теорема верна для константы аппроксимации*

$$8+4\log k.$$

Доказательство. Можно считать, что любой шаг Пропорционального механизма мы можем назвать нулевым. Введём теперь одну полезную функцию.

Определение 3.1. *Условной общей ценой* $\mathrm{CCS}(a_{i_1},\dots,a_{i_m};\xi_{i_{m+1}},\dots,\xi_{i_k};t;S)$ будем называть условное математическое ожидание общей цены на шаге t алгоритма 1, если на нулевом шаге объекты были установлены в точках из множества S, обслужены кластеры $i_{m+1},\dots,i_k$, общие цены для них равны соответственно $\xi_{i_{m+1}},\dots,\xi_{i_k}$, а общие цены для необслуженных кластеров $i_1,\dots,i_m$ равны соответственно $a_{i_1},\dots,a_{i_m}$.

В случае, когда не установлено ещё вообще ни одного объекта, будем полагать a_i равными количествам игроков в соответствующих кластерах C_i.

Удобство определения состоит, в частности, в том, что при вышеописанных условиях вероятность обслужить новый кластер i на первом шаге равна

$$\frac{a_i}{\sum_{j=1}^m a_{i_j}+\sum_{j=m+1}^k\xi_{i_j}},$$

а вероятность установить объект в один из уже обслуженных кластеров равна

$$\frac{\sum_{j=m+1}^k\xi_{i_j}}{\sum_{i=1}^m a_{i_j}+\sum_{j=m+1}^k\xi_{i_j}}.$$

Основная часть доказательства соотоит в оценке сверху условной общей цены на каждом шаге алгоритма. Не умаляя общности, будем считать $i_j = j$ для всех j от 1 до k.

Утверждение 3.3. *При всех возможных значениях аргументов условная общая цена*

$$\mathrm{CCS}(a_1, \ldots, a_m; \xi_{m+1}, \ldots, \xi_k; t; S) \leqslant \\ \leqslant (1 + H_t) \left(\sum_{j=m+1}^{k} \xi_j + 4 \sum_{j=1}^{m} b_j \right) + \frac{m-t}{m} \sum_{j=1}^{m} a_j, \quad (3.1)$$

где $H_t = \sum_{l=1}^{t} \frac{1}{l}$ — сумма первых t членов гармонического ряда, а b_i — общая цена для игроков кластера C_i в случае оптимальной установки объекта в этом кластере.

Замечание 3.4. *Оценка не зависит от множества точек S, в которых установлены объекты.*

Доказательство утверждения 3.3. Будем доказывать индукцией по t.
База. При $t = 0$ условная общая цена равна сумме общих цен по всем кластерам. Поэтому неравенство (3.1) верно:

$$\mathrm{CCS}(a_1, \ldots, a_m; \xi_{m+1}, \ldots, \xi_k; 0; S) = \sum_{i=1}^{m} a_i + \sum_{i=m+1}^{k} \xi_i \leqslant \\ \leqslant (1 + H_0) \left(\sum_{i=m+1}^{k} \xi_i + 4 \sum_{i=1}^{m} b_m \right) + \frac{m-0}{m} \sum_{i=1}^{m} a_i.$$

Переход. Рассмотрим варианты первого шага. Обозначим через ℓ точку, в которую алгоритм устанавливает объект на первом шаге, (случайную величину с распределением X). Через X_i обозначим условное распределение в случае, когда ℓ — координата игрока из кластера C_i, обслуженного на первом шаге, а через X_0 — условное рапределение в случае, когда ℓ — координата игрока из кластера, уже обслуженного ранее. Далее «обнулим счётчик алгоритма», то есть будем считать, что мы снова находимся на нулевом шаге.

Будет справедливо следующее соотношение:

$$\mathrm{CCS}(a_1, \ldots, a_m; \xi_{m+1}, \ldots, \xi_k; t; S) = \\ = \sum_{i=1}^{m} \frac{a_i}{\sum_{j=1}^{m} a_j + \sum_{j=m+1}^{k} \xi_j} \cdot \mathbb{E}_{\ell \sim X_i} \left[\mathrm{CCS}(\overline{a}^{\ell}_{-i}; \xi^{\ell}_{m+1}, \ldots, \xi^{\ell}_k, \xi_i; t-1; S \cup \{\ell\}) \right] + \\ + \frac{\sum_{j=m+1}^{k} \xi_j}{\sum_{i=1}^{m} a_j + \sum_{j=m+1}^{k} \xi_j} \cdot \mathbb{E}_{\ell \sim X_0} \left[\mathrm{CCS}(a^{\ell}_1, \ldots, a^{\ell}_m; \xi^{\ell}_{m+1}, \ldots, \xi^{\ell}_k; t-1; S \cup \{\ell\}) \right],$$

где $\overline{a}^{\ell}_{-i} = \left(a^{\ell}_1, \ldots, a^{\ell}_{i-1}, a^{\ell}_{i+1}, \ldots, a^{\ell}_m\right)$, a^{ℓ}_j — новая общая цена для необслуженного кластера C_j после установки ещё одного объекта в точку ℓ, а ξ^{ℓ}_j — для обслуженного. ξ^{ℓ}_j и a^{ℓ}_j — случайные величины, но все их возможные значения уж точно не могут превосходить ξ_j и a_j соответственно. Значит, справедливы оценки

$$\mathbb{E}_{\ell \sim X_i}[a^{\ell}_j] \leqslant a_j \quad \text{и} \quad \mathbb{E}_{\ell \sim X_0}[\xi^{\ell}_j] \leqslant \xi_j. \quad (3.2)$$

Пусть $A = \sum_{j=1}^{m} a_j$, $\Xi = \sum_{j=m+1}^{k} \xi_j$, а $B = \sum_{j=1}^{m} b_j$.

В силу предположения индукции будут справедливы оценки:

$$\mathrm{CCS}(\overline{a}_{-i}^{\ell}; \xi_{m+1}^{\ell}, \ldots, \xi_k^{\ell}, \xi_i; t-1; S \cup \{\ell\}) \leqslant \\ \leqslant (1 + H_{t-1}) \left(\sum_{j=m+1}^{k} \xi_j^{\ell} + \xi_i + 4 \left(\sum_{j=1}^{m} b_j - b_i \right) \right) + \frac{m-t}{m-1} \sum_{\substack{j=1 \\ j \neq i}}^{m} a_j^{\ell};$$

$$\mathrm{CCS}(a_1^{\ell}, \ldots, a_m^{\ell}; \xi_{m+1}^{\ell}, \ldots, \xi_k^{\ell}; t-1; S \cup \{\ell\}) \leqslant \\ \leqslant (1 + H_{t-1}) \left(\sum_{j=m+1}^{k} \xi_j^{\ell} + 4 \sum_{j=1}^{m} b_j \right) + \frac{m-t+1}{m} \sum_{j=1}^{m} a_j^{\ell};$$

В силу этих оценок, соотношений (3.2) и линейности математического ожидания, условную общую цену можно оценить сверху:

$$\mathrm{CCS}(a_1, \ldots, a_m; \xi_{m+1}, \ldots, \xi_k; t; S) \leqslant \\ \leqslant \sum_{i=1}^{m} \frac{a_i}{A + \Xi} \cdot \mathbb{E}_{\ell \sim X_i} \left[(1 + H_{t-1}) \left(\sum_{j=m+1}^{k} \xi_j^{\ell} + \xi_i + 4(B - b_i) \right) + \frac{m-t}{m-1} \sum_{\substack{j=1 \\ j \neq i}}^{m} a_j^{\ell} \right] + \\ + \frac{\Xi}{A + \Xi} \cdot \mathbb{E}_{\ell \sim X_0} \left[(1 + H_{t-1}) \left(\sum_{j=m+1}^{k} \xi_j^{\ell} + 4B \right) + \frac{m-t+1}{m} \sum_{j=1}^{m} a_j^{\ell} \right] \overset{(3.2)}{\leqslant} \\ \overset{(3.2)}{\leqslant} \sum_{i=1}^{m} \frac{a_i}{A + \Xi} \cdot \left[(1 + H_{t-1})(\Xi + \mathbb{E}_{\ell \sim X_i}[\xi_i] + 4(B - b_i)) + \frac{m-t}{m-1} \cdot (A - a_i) \right] + \\ + \frac{\Xi}{A + \Xi} \cdot \left[(1 + H_{t-1})(\Xi + 4B) + \frac{m-t+1}{m} \cdot A \right]. \quad (3.3)$$

Из леммы 2.1 для набора точек $\{\ell_j\}_{j \in C_i}$, расстояний от них до ближайших объектов и p, равного оптимальной координате объекта в кластере C_i, следует $\mathbb{E}_{\ell \sim X_i}[\xi_i] \leqslant 4b_i$. Значит, правую часть 3.3 можно оценить сверху:

$$\sum_{i=1}^{m} \frac{a_i}{A + \Xi} \cdot \left[(1 + H_{t-1})(\Xi + \mathbb{E}_{\ell \sim X_i}[\xi_i] + 4(B - b_i)) + \frac{m-t}{m-1} \cdot (A - a_i) \right] + \\ + \frac{\Xi}{A + \Xi} \cdot \left[(1 + H_{t-1})(\Xi + 4B) + \frac{m-t+1}{m} \cdot A \right] \leqslant \\ \leqslant (1 + H_{t-1})(\Xi + 4B) + \sum_{i=1}^{m} \frac{a_i}{A + \Xi} \cdot \left[\frac{m-t}{m-1} \cdot (A - a_i) \right] + \\ + \frac{\Xi}{A + \Xi} \cdot \frac{m-t+1}{m} \cdot A = \\ = (1 + H_{t-1})(\Xi + 4B) + \frac{m-t}{m-1} \cdot \frac{A^2 - \sum_{i=1}^{m} a_i^2}{A + \Xi} + \frac{m-t+1}{m} \cdot \frac{A\Xi}{A + \Xi}.$$

По неравенству между средним квадратичным и средним арифметическим $\sum_{i=1}^m a_i^2 \leqslant \leqslant \frac{1}{n}A^2$. Теперь оценку можно продолжить как

$$(1 + H_{t-1})(\Xi + 4B) + \frac{m-t}{m-1} \cdot \frac{A^2 - \sum_{i=1}^m a_i^2}{A + \Xi} + \frac{m-t+1}{m} \cdot \frac{A\Xi}{A+\Xi} \leqslant \leqslant (1 + H_{t-1})(\Xi + 4B) + \frac{m-t}{m} \cdot \frac{A^2}{A+\Xi} + \frac{m-t+1}{m} \cdot \frac{A\Xi}{A+\Xi} = = (1 + H_{t-1})(\Xi + 4B) + \frac{m-t}{m} \cdot \frac{A(A+\Xi)}{A+\Xi} + \frac{1}{m} \cdot \frac{A\Xi}{A+\Xi}.$$

В таком случае, нам достаточно доказать, что

$$(1 + H_{t-1})(\Xi + 4B) + \frac{m-t}{m} \cdot \frac{A(A+\Xi)}{A+\Xi} + \frac{1}{m} \cdot \frac{A\Xi}{A+\Xi} \leqslant \leqslant (1 + H_t)(\Xi + 4B) + \frac{m-t}{m} \cdot A.$$

Это равносильно неравенству

$$\frac{1}{m} \cdot A\Xi \leqslant \frac{1}{t}(\Xi + 4B)(A + \Xi). \tag{3.4}$$

Заметим, что в любой момент количество шагов, которые ещё не сделал алгоритм, не может быть больше количества необслуженных кластеров. Поэтому $t \leqslant m$. А значит неравенство (3.4) верно. □

Остаётся сказать, что по определению

$$\mathrm{CS}(\mathcal{L}) = \mathrm{CCS}(|C_1|, \dots, |C_k|;; k; \varnothing) \leqslant (1 + H_k) \cdot 4 \sum_{i=1}^k b_i + \frac{k-k}{k} \cdot \sum_{i=1}^k |C_i| \leqslant \leqslant (2 + \log k) \cdot 4 \,\mathrm{opt}(\mathcal{L}) \leqslant 24 \log k \cdot \mathrm{opt}(\mathcal{L})$$

при $k \geqslant 3$.

Что и завершает доказательство. □

4 Случай кластерных пространств с равными a_i

В этом разделе мы рассмотрим один частный случай основной теоремы 3.1. Он интересен тем, что даёт ещё один возможный способ доказывать оценку константы аппроксимации $O(\log k)$.

Лемма 4.1. *Пусть пространство Ω состоит только лишь из точек, в которых находятся игроки, и пусть для любых двух точек ℓ_i и ℓ_j из разных кластеров расстояние между ними $d(\ell_i, \ell_j) = 1$, а в каждом кластере C_i находится по N игроков на расстоянии $\varepsilon_i \leqslant 1$ друг от друга.*

Тогда Priportionality mechanism аппроксимирует оптимум с константой $\gamma(k) = = O(\log k)$.

Доказательство. Не трудно заметить, что это разбиение на кластеры действительно является оптимальной расстановкой объектов. Причём объект при оптимальном расположении может находиться в любой точке кластера.

Обозначим через $b_i = \varepsilon_i(N-1)$ общую цену для кластера C_i, если он обслужен. А через $a = N$ — общую цену для кластера C_i, если он не обслужен (в этом случае она равна для всех кластеров). Также пусть $B = \sum_{i=1}^{k} b_i$.

Не умаляя общности, пусть уже обслужены кластеры $1, 2, \ldots, t$. Тогда на математическое ожидание $\mathbb{E}[steps(1)]$ шагов алгоритма для обслуживания нового кластера можно написать следующее соотношение:

$$\mathbb{E}[steps(1)] = 1 \cdot \frac{(k-t)a}{(k-t)a + b_1 + b_2 + \ldots + b_t} + \\ + (1 + \mathbb{E}[steps(1)]) \cdot \left(1 - \frac{(k-t)a}{(k-t)a + b_1 + b_2 + \ldots + b_t}\right).$$

Значит,

$$\mathbb{E}[steps(1)] = \frac{(k-t)a + b_1 + b_2 + \ldots + b_t}{(k-t)a} \leqslant 1 + \frac{B}{(k-t)a}.$$

Заметим, что оценка не завит от того, какие именно кластеры обслужены (только от их количества). Теперь в силу линейности математического ожидания мы можем сказать что математическое ожидание количества шагов алгоритма для обслуживания t_0 кластеров не превосходит

$$t_0 + \sum_{i=1}^{t_0-1} \frac{B}{(k-i)a}.$$

Будем говорить, что алгоритм *совершает ошибку*, если он устанавливает новый объект в кластере с уже установленным объектом. Значит, математическое ожидание количества ошибок при обслуживании t_0 кластеров не превосходит $\sum_{i=1}^{t_0-1} \frac{B}{(k-i)a}$.

Предположим, что сделав k шагов, алгоритм продолжит свою работу, пока не обслужит все кластеры. Ясно, что в таком случае он совершит не меньше ошибок, чем за k шагов. Значит, математическое ожидание количества ошибок за k шагов $\mathbb{E}[steps(k)] \leqslant \sum_{i=1}^{k-1} \frac{B}{(k-i)a}$.

Теперь оценим $\mathrm{CS}(\mathcal{L})$ через количество ошибок. Пусть в конце были обслужены кластеры $i_1, i_2, \ldots, i_n$. Тогда общая цена будет равна

$$b_{i_1} + b_{i_2} + \ldots + b_{i_n} + (k-n)a \leqslant B + (k-n)a.$$

В таком случае алгоритм совершает как раз $k-n$ ошибок. Поскольку оценка выше не зависит от номеров обслуженных кластеров, а только от их количества, справедливо следующее:

$$\mathrm{CS}(\mathcal{L}) \leqslant B + a \cdot \mathbb{E}_m \leqslant B + a \cdot \sum_{i=1}^{k-1} \frac{B}{(k-i)a} = B(1 + H_{k-1}),$$

где H_{k-1} — сумма первого $k-1$ члена гармонического ряда.

Остаётся только заметить, что $1 + H_{k-1} = O(\log k)$. $\square$

Список литературы

[1] Dimitris Fotakis and Christos Tzamos, *Winner-Imposing Strategyproof Mechanisms for Multiple Facility Location Games*, Springer, 234–245, 2010.

[2] Соловьёв Ю. П. *Неравенства*, МЦНМО, 2005.

[3] Fowler R. J., Paterson M. S., Tanimoto S. L. *Optimal packing and covering in the plane are NP-complete*, Information Processing Letters, Vol. 12, 133 – 137, 1981.

[4] Lu, P., Sun, X., Wang, Y., Zhu, Z. A. *Asymptotically optimal strategy-proof mechanisms for two-facility games*, In ACM Conference on Electronic Commerce, 315 – 324, 2010.

[5] Nick Gravin, Dominik Scheder, *In Defense of Bureaucracy in the Metric Facility Location Problem*, pre-print https://arxiv.org/abs/1202.1231

[6] GUHA, S. AND KHULLER, S. 1999. Greedy strikes back: Improved facility location algorithms. J. Algorithms 31, 1, 228–248.

[7] LI, S. 2011. A 1.488 approximation algorithm for the uncapacitated facility location problem. In ICALP (2). 77–88.