Студент: Тимур Хабибуллин
Группа: 1
Дата: 5 мая 2022 г.

---

1. (a) Пусть мы нашли связный подграф G_1 минимального веса графа G, содержащий терминальные вершины T. Для того, чтобы это было дерево, надо показать, что G_1 не имеет циклов. Пусть это не так и в G_1 есть цикл. Выберем в этом цикле ребро с наибольшим весом и удалим. Тогда у нас ничего не испортится - граф останется связным, а кроме того, вес этого подграфа не увеличится, потому что веса ребер неотрицательные (мы могли удалить ребро с весом ноль). Тогда такую процедуру можно продолжить - удалять ребра с наибольшим весом в остальных циклах подграфа G_1, пока не получим дерево.

   (b) Рассмотрим случай $|T| = 3$. Если найти кротчайшие пути между какими-нибудь двумя вершинами, то мы получим дерево, однако оно не обязательно будет иметь наименьший вес. Поэтому будем рассуждать по-другому - пусть у нас есть дерево Штейнера, терминальные вершины x_1, x_2, x_3 должны быть листами этого дерева, иначе можно убрать ненужные ребра. Пусть y - какая-то промежуточная вершина, которая соединена со всеми терминальными вершинами (это может быть и одна из терминальных вершин). Тогда для того, чтобы у нас получился минимальный вес дерева, достаточно, чтобы велина

$$d(x_1, y) + d(x_2, y) + d(x_3, y) \to \min$$

   Тогда можно поступить так - запустим из каждой терминальной вершины алгоритм Дейкстры (еще будем для каждой вершины сохранять откуда мы пришли) и посчитаем минимальные расстояния для всех вершин исходного графа, на это мы потратим $O(E \log(V))$ времени. Затем для каждого $y \in V$ посчитаем

$$d(x_1, y) + d(x_2, y) + d(x_3, y)$$

   и найдем минимум за $O(V)$. Затем восстановим пути $x_i \to y$ и получим граф с минимальным весом.

   (c) В случае $|T| = 4$ будем искать вес вершины по более общей формуле (терминальные вершины в каком-то порядке):

$$d(x_1, y) + d(x_2, y) + d(y, z) + d(x_3, z) + d(x_4, z) \to \min$$

   Здесь мы рассматриваем уже 2 промежуточные точки, потому что если у нас 1 точка, то такой способ может быть не оптимальным, а если, например, три, то можно свести его с случаю, когда мы рассматриваем две промежуточные точки. Здесь неоптимально применять алгоритм Дейкстры, потому что придется каждый раз пересчитывать $d(y, z)$, поэтому воспользуемся алгоритмом Флойда-Уоршелла - посчитаем кратчайшие расстояния между всеми вершинами графа за $O(V^3)$ (сохраним для каждой вершины его родителя), для каждой пары $(y, z) \in V \times V$ посчитаем вес подграфа по формуле (надо еще поиграться с перестановкой y и z)

$$d(x_1, y) + d(x_2, y) + d(y, z) + d(x_3, z) + d(x_4, z),$$

   и найдем минимум за $O(V^2)$. Затем восстановим ответ. В результате алгоритм за $O(V^3)$.

2. Пусть нам дана матрица расстояний $d[v][u]$ между вершинами графа. Каждое расстояние соответсвует длине некоторого пути между вершинами. Пусть мы изменили ребро (x, y) в графе. Попробуем для каждой пары вершин выяснить, можно ли изменить путь так, чтобы он проходил через это ребро и будет ли это вообще выгодно (т.е. укоротится длина пути из u в v или нет). Для этого посчитаем величины

$$d'[v][u] = d[v][x] + \omega'_{x,y} + d[y][u].$$

Если $d'[v][u] = +\infty$, то добраться по такому пути до этих вершин через это ребро невозможно. Иначе возможно 2 ситуации - старый путь проходил через ребро (x, y) или нет. После изменения ребра, может оказаться так, что путь через это ребро стал короче, чем исходный путь. Тогда можно сравнить значения и обновить матрицу расстояний:

$$d[u][v] = \min(d[u][v], d'[u][v])$$

Проделываем такую процедуру для всех элементом матрицы и получаем алгоритм за $O(V^2)$.

3. (a) Пусть имеем какой-то путь $p(v, u) : v \to x_1 \to ... \to x_k \to u$, тогда его длина пути есть:

$$d(p) = \omega(x_1, v) + ... + \omega(u, x_k)$$

После изменения весов ребер с помощью потенциалов, длина этого пути станет

$$d'(p) = \omega(x_1, v) + \varphi(x_1) - \varphi(u) + ... + \omega(u, x_k) + \varphi(v) - \varphi(x_k) = d(p) + \varphi(v) - \varphi(u)$$

Получается, что длина изменяется за счет разности потенциалов начальной и конечной вершин графа. Тогда если путь $p(v, u)$ был кратчайшим между вершинами v и u, то если возьмем какой-то другой путь $\hat{p}(v, u) : v \to y_1 \to ... \to y_k \to u$, то

$$d(\hat{p}) > d(p)$$

$$d'(\hat{p}) = d(\hat{p}) + \varphi(v) - \varphi(u) > d'(p) = d(p) + \varphi(v) - \varphi(u)$$

Значит, при таком отображении длины весов кратчайшие пути отображаются в кратчайшие

(b) Пусть у нас есть ребро, вес которого после такого отображения стал отрицательным:

$$\omega'(v, u) < 0$$

Но тогда у нас нарушается правило треугольника для метрики:

$$\omega'(v, u) = \omega(v, u) + \rho(s, v) - \rho(s, u) < 0$$

$$\rho(s, v) + \omega(v, u) < \rho(s, u),$$

а должно быть наоборот. Значит, такого быть не может.

(c) Пусть есть граф с отрицательными ребрами. В предыдущих пунктах показали, что при отображении весов с помощью потенциала кратчайшие пути отображаются в кратчайшие, а так же что если в качестве потенциала выбрать расстояние до какой-то вершины (если расстояния конечные), то тогда веса становятся положительными. Применим эти для поиска матрицы расстояний - выберем какую-нибудь вершину s и сделаем такое отображение весов, отрицательные веса станут положительными. Для этого надо посчитать расстояние от s до каждой вершины. Это можно сделать с помощью алгоритма Форда-Беллмана за $O(VE)$ Тогда запустим алгоритм Дейкстры от каждой вершины и получим расстояния до остальных вершин $O(VE\log(V))$. Когда будем доставать вершини из кучи, будем сразу делать обратное отображение расстояния

$$d[v][u] = d'[v][u] - (\rho(s, v) - \rho(s, u))$$

и записывать в ответ.