Первый курс, осенний семестр 2020/21
Практика по алгоритмам #3

Потоки, потоки, потоки!

20 сентября

Собрано 20 сентября 2021 г. в 22:45

Содержание

Потоки, потоки, потоки!

Напомним, что такое *транспортная сеть*. У нас есть:

1. ориентированный граф $G(V, E)$;
2. исток $s \in V$ и сток $t \in V$ $(s \neq t)$;
3. пропускные способности $c\colon V^2 \to \mathbb{R}_+$;
4. потоки по рёбрам и обратным рёбрам $f\colon V^2 \to \mathbb{R}$ (могут быть отрицательны).

И должны быть выполнены следующие свойства:

1. $\forall v, u \in V \quad f(v,u) = -f(u,v)$;
2. $\forall v, u \in V \quad f(v,u) \leqslant c(v,u)$;
3. $\forall v \in V \setminus \{s,t\} \quad \sum_{u \in V} f(v,u) = 0$;

Величина потока $|f| = \sum_{u \in V} f(s,u) = \sum_{u \in V} f(u,t)$.

У нас есть алгоритм Форда-Фалкерсона, который dfs-ом ищет дополняющий путь из s в t и, пока находит, пускает по нему поток. Для $c(v,u) \in \mathbb{Z}$ работает за $\mathcal{O}(|f| \cdot E)$.

1. **Форд-Фалкерсон не такой уж и медленный**

   Оцените время работы алгоритма Форда-Фалкерсона на произвольном графе с *единичными* пропускными способностями.

2. **Вершинно-непересекающиеся пути**

   Научитесь за $\mathcal{O}(kE)$ находить k вершинно непересекающихся путей из s в t.
   (a) в орграфе, (b) в неорграфе.

3. **Восстановление матрицы**

   Даны суммы элементов матрицы в каждом столбце и каждой строке.
   Восстановить матрицу при условии, что она составлена из целых чисел от 0 до 100.

4. **Турнир**

   Дана матрица уже сыгранных матчей групового футбольного турнира «каждый с каждым».
   Ничьих не бывает. Победа даёт команде одно очко.
   Можно ли так доиграть турнир, чтобы $\forall i$: i-я команда имела в итоге ровно a_i побед?

5. **Крутая декомпозиция**

   Научитесь декомпозировать поток на пути за $\mathcal{O}(VE)$.

   Указание: с лекции умеете за $\mathcal{O}(E^2)$, вспомните как, оптимизируйте.

6. **Минимизация максимальной степени**

   Ориентировать данный неорграф так, чтобы max исходящая степень была минимальна.
   (a) $\mathcal{O}(E^2 \log V)$, (b) $\mathcal{O}(E^2)$.

7. **Взвешенные Cover, IS**

   Дан двудольный граф. У каждой вершины есть неотрицательный вес. Найти вершинное покрытие минимального веса, независимое множество максимального веса.

   Указание: вспомните с лекции, как потоками искать в двудольном графе matching, IS, VC.

8. **Замкнутое подмножество (лекции по матану)**

   Каждой вершине орграфа сопоставлено число (возможно, отрицательное) – её вес. Подмножество вершин A называется *замкнутым*, если из него не исходят рёбра в $V \setminus A$. Найдите замкнутое подмножество вершин максимального суммарного веса.

9. **(*) Декомпозиция на циклы**

   Научитесь декомпозировать циркуляцию на циклы.

10. **(*) Разрез минимального среднего веса**

    Найти разрез с min средним весом – отношением суммарного веса рёбер к их количеству.

11. **(*) Уголки**

    Какое max количество чёрных уголков можно разместить на шахматной доске $n \times m$ с дырками? Чёрный уголок – фигура из трёх клеток: центральная клетка **чёрного цвета** и две смежных с ней по стороне белых клетки. При этом фигура 1×3 – не уголок.

Разбор задач практики

1. **Форд-Фалкерсон не такой уж и медленный**

   $|f| \leqslant E \Rightarrow$ время работы $\mathcal{O}(E^2)$.
   Точнее $|f| \leqslant \min(\deg_s, \deg_t)$. Без кратных ребер $\mathcal{O}(VE)$.

2. **Вершинно не пересекающиеся пути**

   Орграф: раздваиваем вершину, посередине ребро пропускной способности 1, входящие рёбра в первую половину, исходящие из второй.

   Неорграф: приводим к орграфу, делаем то же самое. Заметим, что ребро (a, b) превратится в граф из 4 вершин, 4 рёбер: $a_1 \to a_2 \to b_1 \to b_2 \to a_1$.

3. **Восстановление матрицы**

   Коротко: двудольный граф из строк и столбцов.
   Решение. Соединяем с истоком n вершин-строк, на ребре пишем сумму в строке. Аналогично столбцы соединяем со стоком. Каждую строку соединяем с каждым столбцом, на ребре пишем 100. Числа на рёбрах – пропускные способности.
   Пускаем max поток. Если какое-то из рёбер, соединенных со стоком и истоком, не насыщено, то ответа нет. Иначе ответ — поток на ребрах, соединяющих строки и столбцы.

4. **Турнир**

   Коротко: двудольный граф из команд и несыгранных матчей.
   Решение. Заведем вершину для каждой команды, проведем в нее ребро из истока, пишем на нем число будущих побед этой команды. Заведем вершину для каждого еще не сыгранного матча, из нее ребро в сток, на ребре пишем 1. Проводим в каждый матч ребра из двух команд, которые в нем играют, на ребрах 1. Пускаем max поток. Если из команды в матч течет 1, то команда выиграла этот матч. Если не насытилось какое-то ребро из истока в команду или из матча в сток, то так доиграть невозможно.

5. **Крутая декомпозиция**

   Заставим dfs работать за $\mathcal{O}(V)$. Для этого

   a) Если dfs по ребру не нашёл путь, на обратном ходу рекурсии удалим ребро
   b) Если dfs зациклился, закончим dfs, отменим поток по циклу

   В сумме k dfs-ов отработают за $\mathcal{O}(E + kV) \Rightarrow$ декомпозиция за $\mathcal{O}(VE)$.

6. **Минимизация максимальной степени**

   Коротко: рассмотрим двудольный граф «рёбра $\to$ вершины».
   Каждое ребро в зависимости от ориентации увеличит на 1 степень одного из концов.

   Из истока s проводим ребро в каждую вершину исходного графа. Создаём для каждого ребра исходного графа по вершине, из них проводим рёбра пропускной способности 1 в сток. Если ребро $e = \{u, v\}$, то проводим рёбра $u \to e$ и $v \to e$ пропускной способности 1. Поток, пущенный по ребру $u \to e$ будет означать, что e ориентировано от u к v.
   Рёбер в новом графе $V + 3E = \mathcal{O}(E)$.

   a) $\mathcal{O}(E^2 \log V)$. Бинпоиск по ответу: проверяем, можно ли d. Пишем d на рёбрах из s в вершины. Пускаем max поток, проверяем $|f| = E$ исходного графа. Поток за $\mathcal{O}(E \cdot |f|) = \mathcal{O}(E^2)$.

b) $\mathcal{O}(E^2)$. Пишем сначала $d = 1$, затем $d = 2$, и так далее, пока не найдётся поток размера E. При увеличении d мы для старого потока dfs-ом ищем новые пути, пока ищется. *Время работы:* $\leqslant V$ не удачных запусков dfs-а и $\leqslant E$ удачных.

7. **Взвешенные Cover, IS**

Построим сеть для поиска паросочетаний: добавим слева исток и справа сток. Исходные рёбра сделаем веса $+\infty$ и направим слева направо. Теперь есть биекция между (S, T)-разрезами без бесконечных рёбер и вершинными покрытиями:
$VC =$ вершины левой доли из $S \cup$ вершины правой доли из T.

Отсюда решение: на ребре, соединяющем вершину v с истоком или стоком, пишем w_v, на рёбрах исходного графа пишем $+\infty$. Ищем min разрез. max IS – дополнение к min VC.

8. **Замкнутое подмножество (лекции по матану)**

Будем подгонять разрез в графе вида «исходные n вершин, исток, сток» под нашу задачу. Во-первых, каждое ребро исходного графа $i \to j$ добавим с пропускной способностью $+\infty \Rightarrow$ оно точно не войдёт в min разрез. Теперь подгоним пропускные способности из истока и в сток так, чтобы минимизация $C(S, T)$ максимизировала $W(S) = \sum\limits_{v \in S} w_v$.

Заметим, что S точно замкнуто благодаря бесконечным рёбрам.
Распишем $W(S) =$

$$\sum_{v \in S, w_v \geqslant 0} w_v - \sum_{v \in S, w_v < 0} |w_v| = \sum_{w_v \geqslant 0} w_v - \sum_{v \notin S, w_v \geqslant 0} w_v - \sum_{v \in S, w_v < 0} |w_v| = \texttt{const} - \Big(\sum_{v \notin S, w_v \geqslant 0} w_v + \sum_{v \in S, w_v < 0} |w_v|\Big)$$

Хотим минимизировать то, что вычитается, а это ровно $C(S, T)$ в графе, где из стока проведены ребра в положительные вершины, а из отрицательных в сток.

9. **(*) Декомпозиция на циклы**

Коротко: также, как декомпозиция на пути.

Из каждой вершины запускаем dfs, пока можем. dfs нашёл цикл – отменим по нему поток. За $\mathcal{O}(E^2)$, если совсем в лоб, за $\mathcal{O}(VE)$, если так же, как научились сегодня.

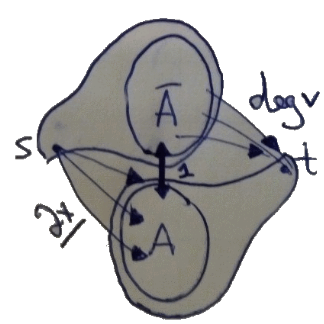
10. **(*) Разрез минимального среднего веса**

Бинпоиск по ответу. Внутри приём, как в задаче «K-best»: $\frac{E(A)}{|A|} \leqslant x \Leftrightarrow E(A) - x|A| \leqslant 0$. $B = V \setminus A$, мы хотим выбрать пропускные способности, чтобы A: $E(A) - x|A| = \min$, находилось как минимальный разрез: $S = A + \{s\}, T = B + \{t\}$.

$E(A) = \frac{1}{2}(\sum_{a \in A} deg[a] - E(A, B))$, $-x|A| = x|B| - x|V|$, $x|V|$ – константа $\Rightarrow$
Берём рёбра $s \to v$: x, $v \to u$: $\frac{1}{2} isedge(v, u)$, $v \to t$: $\frac{1}{2} deg[a]$, ищем min разрез.

Старое решение.
Бинпоиск по ответу. Хотим найти такое A, что $\frac{E(A,A)}{|A|} \geqslant x$.

$E(A, A) = E - E(A, \overline{A}) - E(\overline{A}, \overline{A})$. Хотим $E \geqslant E(A, \overline{A}) + E(\overline{A}, \overline{A}) + x|A|$.
$2E \geqslant 2E(A, \overline{A}) + 2E(\overline{A}, \overline{A}) + 2x|A|$. $2E \geqslant E(A, \overline{A}) + \sum\limits_{\overline{A}} deg_v + 2x|A|$.



Строим граф, где правая сторона неравенства будет разрезом. Добавим все рёбра исходного. Из истока рёбра пропускной способности $2x$ во все вершины. Из каждой вершины v в сток ребро пропускной способности deg_v. Разрезом будет $(\{s\} \cup \overline{A}, \{t\} \cup A)$.

11. **(*) Уголки**

Граф из четырех слоев: белые клетки на чётных строках, раздвоенные черные, белые клетки на нечётных строках. И, конечно, исток и сток. Уголок = белая из чётной строки + чёрная + белая из нечётной строки $\Rightarrow$ уголок = путь из истока в сток. Уголки не должны пересекаться $\Rightarrow$ всё рёбра пропускной способности 1 и найдём max поток.

Время: $\mathcal{O}(size^2)$, где $size$ – площадь поля.

Домашнее задание

3.1. Обязательная часть

1. **(2) Единственность минимального разреза**

   Дан граф и выделенные вершины s и t. За $\mathcal{O}(\texttt{Polynom}(V, E))$ проверить, правда ли, что существует **единственный** минимальный разрез?

2. **(3) Грузовики. Непересекающиеся во времени пути**

   Дан ориентированный граф с начальной и конечной вершинами. В начальной вершине есть K грузовиков. Грузовикам нужно попасть в конечную вершину. Время дискретно. За единицу времени каждый грузовик или стоит на месте, или перемещается в одну из соседних вершин. В любой вершине могут одновременно стоять несколько грузовиков. По любому из рёбер в каждый момент времени должен ехать не более чем один грузовик.
   Минимизируйте время, когда все грузовики окажутся в конечной вершине.

   a) **(2)** $\mathcal{O}(\texttt{Polynom}(V, E, K))$
   b) **(1)** $\mathcal{O}(K(V + K)E)$

3. **(3) Задача про трисочетание**

   Даны девочки, мальчики и собачки.
   Для каждой пары «мальчик, девочка» известно, хочет ли девочка дружить с мальчиком.
   Для каждой пары «собачка, девочка» известно, нравится ли собачка девочке.
   Нужно максимальному количеству девочек выделить по мальчику и собачке так, что:

   a) каждый мальчик не более чем с одной девочкой;
   b) каждая собачка не более чем у одной девочки;
   c) девочка хочет дружить с выбранным ей мальчиком, и собачка ей нравится.

4. **(2) Цикл через данные вершины в неорграфе**

   Дан неориентированный граф.
   Найти за $\mathcal{O}(E)$ вершинно простой цикл, проходящий через две данные вершины.

5. **(2) Цикл через данные вершины в орграфе**

   Докажите, что аналогичная задача в орграфе `NP`-трудна.
   Указание: необходимо свести к ней задачу про два пути.

   Задача про два пути: даны орграф и вершины A, B, C, D; нужно найти не пересекающиеся по рёбрам пути $A \to B, C \to D$. Задача `NP`-трудна.

6. **(3) Равномерное распределение**

   Есть n рабочих и m работ. И есть матрица умений: «какой рабочий какие работы умеет делать». Нужно максимально равномерно распределить работы между рабочими. То есть, каждой работе сопоставить рабочего, который умеет делать эту работу, а кроме того минимизировать $\max\limits_{i=1..n} k_i$, где k_i – количество работ, выданных i-му рабочему.

3.2. Дополнительная часть

1. **(2) Единственность разреза. $\mathcal{O}(E)$.**

   Пусть дан какой-то максимальный поток.

   a) За $\mathcal{O}(E)$ проверить единственность минимального разреза. С доказательством.
   b) За $\mathcal{O}(E)$ найти минимальный разрез $V = S \sqcup T \colon |S| = \max$.

2. **(3) Тест против Форда-Фалкерсона**

   Найдите ориентированный граф с целочисленными пропускными способностями, на которых детерминированный алгоритм Форд-Фалкерсона с фиксированным порядком перебора рёбер, пропускающий $\min_e(c_e - f_e)$ по пути, работает за экспоненту от V.

   (+2) балла за тест против ФФ, в котором перед каждым `dfs` делается `random_shuffle` рёбер.

3. **(4) Япония. Инструменты и заказы.**

   Ситуация из Японии. Есть заказы и инструменты. Для каждого заказа известен список инструментов, который нужен, чтобы его выполнить. Каждый инструмент сделан умелыми японскими рабочими, поэтому бесконечно прочный, его можно один раз купить и много раз использовать. У каждого инструмента есть цена p_i. У каждого заказа есть прибыль, которую можно получить, выполнив заказ. Вы – бедный китайский рабочий. У вас изначально нет инструментов, но зато вы можете под нулевой процент в банке взять сколь угодно большой кредит, чтобы купить инструментов.

   a) **(2)** Вопрос: какую максимальную прибыль вы можете получить?
   b) **(2)** А теперь тот же вопрос, но ещё есть разные скидочные предложения! Скидка позволяет два инструмента i, j купить по специальной цене d: $\max(p_i, p_j) < d < p_i + p_j$. Каждый инструмент присутствует не более чем в одном скидочном предложении.