

Задача 1.

Пусть точка $x \in P$, где $x_1, x_2, \dots, x_m$ суть вершины P . Тогда x является выпуклой комбинацией вершин P , значит $\exists \lambda_i \geq 0$ такие, что $\sum \lambda_i x_i = x$, а $\sum \lambda_i = 1$. Выберем из всех таких представлений такое, в котором количество ненулевых коэффициентов минимально. Покажем, что их не больше, чем $n + 1$: рассмотрим все вершины $y_1, y_2, \dots, y_l$ соответствующие ненулевым λ_i . Т.к. $l \geq n + 2$, то $\exists \lambda'_i$ такие, что $\sum \lambda'_i y_i = 0$ и $\sum \lambda'_i = 0$, действительно, можно переупорядочить набор так, что y_l принадлежит линейной оболочке $y_1, y_2, \dots, y_{l-1}$ (можно выбрать максимальное подмножество линейно независимых векторов вида $(y_1 - y_j)$ и выразить $(y_1 - y_l)$ через них) то есть $y_l = \sum_{j=1}^{l-1} \lambda'_j y_j$, где $\sum \lambda'_j = 1$. Приняв $\lambda'_l = -1$ получим нужный нам набор. Тогда для любого вектора $\lambda'' = \lambda + \lambda' \mu$ верно, что $\sum \lambda''_i y_i = x$ и $\sum \lambda''_i = 1$. Заметим, что можно подобрать μ чтобы одна из его компонент занулилась, что противоречит выбору λ . Значит, существует выпуклая комбинация некоторых $n + 1$ вершин политопа дает x .

Задача 4.

Будем считать, что данный граф $G = (V, E = A \sqcup B)$ связан, иначе, можно применить алгоритм для каждой компоненты связности.

Докажем сначала, то все циклы сбалансированы тогда и только тогда, когда $\exists f : V \rightarrow R$ такая, что $\forall v \in A, u \in B$ и $vu \in E f(u) - f(v) = c(vu)$

1) Пусть нашелся несбалансированный цикл $v_1, v_2, \dots, v_l$. Просуммировав равенства имеющиеся условия получим:

$$\sum (f(v_i) - f(v_{i+1})) = \pm \sum c(v_i v_{i+1}) (-1)^i$$

Правая часть, очевидно, равна нулю, значит выбранный цикл сбалансирован. Противоположное.

2) Пусть все циклы сбалансированы. Покажем, что существует функция удовлетворяющая указанным условиям: Построим f следующим образом: выберем произвольную вершину $v \in V$ и положим $f(v) = 0$. Запустим из вершины v поиск в глубину. При добавлении каждой новой вершины u будем устанавливать для нее $f(u)$ исходя из значений $f(w)$ и $c(uw)$, где w - предок u . Покажем, что для ребер не входящих в дерево T поиска в глубину условие на функцию также f выполняются: пусть $\exists a \in A, b \in B$ и $ab \in E f(a) - f(b) \neq c(ab)$ Рассмотрим пусть между a и b проходящий по ребрам T . Добавив к нему ребро ab , получим сбалансированный цикл в котором условия на f выполняются для всех ребер кроме ab . Рассмотрим

сумму равенств вида $(f(v_i) - f(v_{i+1})) = c(vu)(-1)^i$ получим противоречие как и в пункте 1.

Заметим, что изложенные рассуждения дают нам алгоритм работающий за $O(E)$. Для этого можно отдельно проверить выполняются ли равенства на f после окончания DFS, а можно делать это сразу (если на текущей итерации вершина имеет посещенного соседа, то смотрим верно ли соответствующие этой паре вершин равенство). Таким образом алгоритм или строит функцию f , что доказывает факт сбалансированности всех циклов, или находит препятствие - ребро принадлежащее несбалансированному циклу.

Задача 6. Сопоставим каждому ребру по неотрицательной переменной. Каждой вершине сопоставим следующее условие: сумма переменных соответствующих всем инцидентным ребрам не меньше единицы. Имеем $x \geq 0, b$ - вектор из единиц. Матрица A является матрицей инцидентности графа G и имеет вид: $|V|$ строк и $|E|$ столбцов. Строка соответствующая вершине $v \in V$ состоит из нулей и единиц, причем единицы стоят в столбцах соответствующим инцидентным ребрам. Функционал оптимизации: $c^T x \rightarrow \min$, где $c = b$ Заметим, что в общем случае не является totally unimodular. Покажем, что полиэдр P полученной линейной программы

$$\begin{cases} Ax \geq b \\ x \geq 0 \\ c^T x \rightarrow \min \end{cases}$$

является целым, когда G - двудольный граф. Для этого покажем, что если точка $p \in P$ не является целой, то она не является вершиной и $\exists u : p \pm u \in P$. Заметим, что если $p_i > 1$, то такая точка вершиной являться не может: все условия которые содержат p_i останутся верными при замене p_i на $p_i \pm \epsilon$. Достаточно рассматривать все точки для которых каждая из компонент вектора не превосходит одного. Разобьем все ребра графа на две группы: назовем ребро целым если соответствующая ему переменная равна 0 или 1, и дробным в ином случае. Удалим из графа все целые ребра. Заметим, что если из вершины выходит только одно дробное ребро, и неравенство связанное с этой вершиной выполнено, то оно строгое: действительно, значит, что из нее выходит хотя бы одно целое ребро переменная которого равна 1. Иначе, из вершины выходит

хотя бы два дробных ребра. Т.к. по предположению есть хотя бы одно нецелое ребро, то в полученном графе найдется или четный цикл, или простой путь, оба конца которого - листы (L). Остается заметить, что в качестве u можно выбрать вектор в котором все переменные соответствующие нечетным ребрам L равны ϵ , четным $-\epsilon$, а остальные компоненты равны 0. Очевидно, что $p \pm u \in P$: для любой вершины кроме, возможно, листов L сумма переменных для инцидентных ребер не изменилась. Для листов сумма была строго больше 1, поэтому можно подобрать такой ϵ , что условия останутся выполненными. Значит, дробная точка вершиной являться не может. В частности, максимум функционала оптимизации будет достигаться в точке каждая координата которой равна 0 или 1, что соответствует некоторому подмножеству ребер, которое является реберным покрытием. Заметим, что функционал оптимизации будет равен количеству выбранных ребер, поэтому приведенная линейная программа действительно решает задачу о минимальном реберном покрытии. Построим двойственную линейную программу: В прямой задаче все условия вида $\geq$, и являются "естественными" для задачи на минимизацию, следовательно, в двойственной программе все переменные неотрицательны. Направление оптимизации противоположно направлению оптимизации в прямой задаче. В прямой задаче все переменные неотрицательны, следовательно в обратной все условия будут "естественными" для задачи максимизации, а значит - вида $\leq$

$$\begin{cases} A^T y \leq c \\ y \geq 0 \\ b^T t \rightarrow \max \end{cases}$$

Вектора b и c - единичные. Матрица A^T имеет $|E|$ строк и $|V|$ столбцов, каждое условие соответствует ребру $ab \in E$ и имеет вид $y_a + y_b \leq 1$. Несложно видеть, что если рассматривать полученную программу как целочисленную, то она соответствует задаче о наибольшем независимом множестве вершин графа. Кроме того, очевидно, что любое независимое множество вершин V' является препятствием для задачи о минимальном реберном покрытии: действительно, чтобы покрыть ребрами все вершины из V' нужно хотя бы $|V'|$ ребер.