Курсовые по теории алгоритмов. Осенний семестр 2021 года

В качестве тем для курсовых работ предлагаются различные оптимизационные задачи. Целью работы является исследование данной задачи. Минимальные требования (на тройку) включают в себя следующие пункты.

1. Формулировка соответствующей задачи распознавания, доказательство ее NP-полноты.
2. Исследование частных случаев (какие являются полиномиальными, а какие остаются NP-полными). Как минимум, нужно указать хотя бы один полиномиальный частный случай.
3. Точный экспоненциальный алгоритм решения оптимизационной задачи с оценкой его сложности.
4. Полиномиальный приближенный алгоритм решения оптимизационной задачи (с оценкой времени работы и точности приближения).

Для тех, кто претендует на более высокую оценку, чем 3, предлагаются также следующие пункты. Не обязательно, чтобы они все были в работе (даже на пятерку), но для того, чтобы получить больше тройки хотя бы один из этих пунктов обязательно присутствовать.

- Доказательство того, что наличие полиномиального алгоритма для задачи распознавания влечет наличие такового и для оптимизационной задачи. (В том числе возможность построения оптимальной конфигурации, а не только нахождение оптимальной константы.)
- Нижние оценки погрешности для приближенных алгоритмов (при условии $P \neq NP$).
- Вероятностные алгоритмы.
- Исследование различных вариаций формулировки. (Здесь речь идет о различных изменениях формулировки, не являющихся частными случаями исходной задачи. Нужно исследовать то, как различные изменения в формулировке влияют на алгоритмическую сложность задачи, существование приближенных полиномиальных алгоритмов и прочее.)

В работе можно и нужно использовать доступную литературу, в том числе научные статьи. Для поиска статей на нужную тему рекомендуется использовать сервисы zbMATH (https://zbmath.org) и MathSciNet (https://mathscinet.ams.org/mathscinet/). С недавних пор, к zbMATH открыт свободный доступ. К MathSciNet свободного доступа нет, но скорее всего у библиотеки СПбПУ должна быть на него подписка: если это так, то на MathSciNet должно быть возможно зайти с компьютеров из читального зала. Скорее всего, должна быть также возможность настроить себе удаленный доступ из дома, это можно уточнить в библиотеке. Использованная литература должна быть перечислена в библиографии.

Важно, чтобы вы хорошо понимали написанный текст (как определения и формулировки, так и доказательства теорем) и могли отвечать на вопросы по нему. То есть прежде чем что-нибудь написать, настоятельно рекомендуется досконально во всем разобраться.

Работа должна быть аккуратно оформлена. Доказательства должны быть полными и подробными. Рекомендуемый объем ~ 15 страниц.

Работу рекомендуется выполнять в среде LaTeX. Впрочем, допускаются и другие системы оформления документов. Сдавать работу нужно будет в виде файла в формате PDF (пожалуйста, не нужно присылать мне файлы docx и т.п.)

Задачи

Minimum Multiway Cut. Даны конечный неориентированный граф $G = (V, E)$, весовая функция $w : E \to \mathbb{N}$ и подмножество $S = \{s_1, s_2, \dots, s_k\} \subset V$. Требуется найти подмножество $T \subset E$ наименьшего веса, чтобы в графе $G - T$ все вершины множества S лежали в разных компонентах связности.

Max Leaf Spanning Tree. Дан конечный неориентированный граф $G = (V, E)$. Требуется найти остовное дерево T графа G с наибольшим числом висячих вершин (листьев).

Min Degree Spanning Tree. Дан конечный неориентированный граф $G = (V, E)$. Требуется найти остовное дерево T графа G, для которого $\Delta(T)$ наименьшая (через $\Delta(T)$ обозначается максимальная степень вершины графа T).

Minimum Makespan Scheduling. Даны n задач, для каждой из которых известно время ее выполнения p_1, p_2, $\ldots$, p_n и m процессоров ($m, n, p_1, \ldots, p_n \in \mathbb{N}$. Требуется найти распределение задач по процессорам, при котором время работы окажется минимальным).

Minimum Makespan Scheduling on unrelated machines. Даны n задач и m процессоров, а также матрица (p_{ij}), где p_{ij} — время выполнения задачи номер i на процессоре номер j ($m, n, p_{ij} \in \mathbb{N}$). Требуется найти распределение задач по процессорам, при котором время работы окажется минимальным.

Maximum Edge Disjoint Paths. Дан конечный неориентированный граф $G = (V, E)$ и множество $T = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subset V^2$. Требуется найти наибольшее число путей $P_{i_1}, \ldots, P_{i_m}$, где путь P_{i_j} имеет начало в s_{i_j} и конец в t_{i_j}, все индексы $i_1, \ldots, i_m$ различны и пути $P_{i_1}, \ldots, P_{i_m}$ не имеют общих ребер.

Max-Cut. Дан конечный неориентированный граф $G = (V, E)$. Требуется найти такое подмножество $S \subset V$, чтобы количество ребер, соединяющих S и его дополнение $\overline{S}$ было максимальным.

k-Median. Даны полный граф $G = (V, E)$ с весовой функцией $d : V^2 \to \mathbb{N}$, удовлетворяющей неравенству треугольника, и натуральное число $k < |V|$. Требуется найти такое подмножество $S \subset V$, что $|S| = k$ и величина $cost(S) = \sum_{v \in V} d(v, S)$ (где $d(v, S) = \min_{u \in S} d(u, v)$) была наименьшей.

Set Cover. Дано множество U, $|U| = n$ и семейство его подмножеств $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$, объединение которых совпадает с U. Требуется выделить из $\mathcal{F}$ наименьшее подсемейство, дающее в объединении U.

Steiner Tree. Даны конечный неориентированный граф $G = (V, E)$, весовая функция $w : E \to \mathbb{N}$ и подмножество $T \subset V$. Требуется найти поддерево ST графа G наименьшего веса, содержащее все вершины множества T.

Group Steiner Tree. Даны конечный неориентированный граф $G = (V, E)$, весовая функция $w : E \to \mathbb{N}$ и семейство подмножеств $X_1, \ldots, X_k \subset V$. Требуется найти поддерево ST графа G наименьшего веса, содержащее хотя бы по одной вершине каждого из множеств $X_1, \ldots, X_k$.

Directed Steiner Tree. Даны конечный ориентированный граф $D = (V, A)$, весовая функция $w : A \to \mathbb{N}$, вершина $v \in V$ и подмножество $T \subset V$. Требуется найти корневое поддерево ST орграфа D наименьшего веса, с корнем r и содержащее все вершины множества T.

Euclidean TSP. Дано множество точек плоскости с целыми координатами. Расстояние между точками считается по формуле $d'(A, B) = \lceil |AB| \rceil$. Требуется найти замкнутый циклический маршрут наименьшей длины, проходящий через каждый город ровно по одному разу.

MAX TSP. Даны n городов $v_1, \ldots, v_n$ и попарные расстояния между ними $d_{ij} \in \mathbb{N}$. Требуется найти замкнутый циклический маршрут наибольшей длины, проходящий через каждый город ровно по одному разу.

k-means. Даны n точек с целыми координатами в евклидовом пространстве размерности d; нужно разделить их на k кластеров, так чтобы сумма квадратов расстояний от точек до центров кластеров была минимальной. Центр кластера — это среднее арифметическое входящих в него точек.

Maximum Acyclic Subgraph. Дан конечный ориентированный граф $D = (V, A)$. Требуется найти наибольшее подмножество $A' \subset A$, такое, что подграф $D' = (V, A')$ не содержит циклов.

Shortest Superstring. Дан конечный алфавит Σ и множество из n строк $S = \{s_1, \ldots, s_n\} \subset \Sigma^*$. Требуется найти кратчайшую строку $s \in \Sigma^*$, которая содержит каждую s_i в качестве подстроки.

MAX-SAT. Дана булева формула в КНФ с переменными $x_1, \ldots, x_n$. Требуется присвоить этим переменным такие значения, чтобы наибольшее возможное число клозов были выполнены.

Linear Equations Over $\mathbb{F}_2$. Дана система n линейных уравнений с m неизвестными с коэффициентами из поля $\mathbb{F}_2$. Требуется присвоить этим переменным значения из поля $\mathbb{F}_2$ так, чтобы они удовлетворяли как можно большему числу уравнений.