На правах рукописи

Моршинин Александр Владимирович

ПРИБЛИЖЕННОЕ И ТОЧНОЕ РЕШЕНИЕ РАЗЛИЧНЫХ ВАРИАНТОВ ЗАДАЧИ КЛАСТЕРИЗАЦИИ ВЕРШИН ГРАФА

Специальность 01.01.09 – Дискретная математика и математическая кибернетика

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Новосибирск – 2022

Работа выполнена в Омском филиале ФГБУН «Институт математики им. С.Л. Соболева СО РАН».

Научный руководитель:

Ильев Виктор Петрович, доктор физико-математических наук, ФГБОУ высшего образования «Омский государственный университет им Ф.М. Достоевского».

Официальные оппоненты:

Родионов Алексей Сергеевич, доктор технических наук, ФГБУН «Институт вычислительной математики и математической геофизики СО РАН».

Мокеев Дмитрий Борисович, кандидат физико-математических наук, ФГАОУ высшего образования «Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского».

Ведущая организация: ФГАОУ высшего образования «Уральский федеральный университет им. первого Президента России Б.Н. Ельцина».

Защита диссертации состоится 18 мая 2022 г. в 17 ч. 00 мин. на заседании диссертационного совета Д 003.015.01 при ФГБУН «Институт математики СО РАН» по адресу: 630090 г. Новосибирск, пр. Академика Коптюга, 4.

С диссертацией можно ознакомиться в библиотеке Института математики СО РАН и на сайте math.nsc.ru.

Автореферат разослан «____» _____ 2022 г.

Ученый секретарь
диссертационного совета
к.ф.-м.н. Ц. Ч.-Д. Батуева

Общая характеристика работы

Актуальность темы. В диссертационной работе исследуются различные варианты задач кластеризации вершин графа. В задачах кластеризации требуется разбить заданное множество объектов на несколько подмножеств (кластеров), основываясь только на сходстве объектов. В задаче кластеризации вершин графа отношение сходства объектов задается при помощи ребер неориентированного графа, вершины которого взаимно однозначно соответствуют объектам. В машинном обучении задачи кластеризации относят к разделу обучения без учителя. Также изучаются варианты задач и методы кластеризации с частичным обучением, в которых фиксированное подмножество объектов изначально распределено по кластерам. Задачи кластеризации вершин графа наряду с задачами о минимальном разрезе в графе являются наиболее адекватными математическими моделями задач кластеризации и классификации взаимосвязанных объектов. Однако, в отличие от задачи о минимальном разрезе в задачах кластеризации вершин графа минимизируется не только число «лишних» связей между классами, но и число «недостающих» связей внутри классов.

Актуальность темы диссертации обусловлена тем, что задачи кластеризации вершин графа являются математическими моделями множества практически важных задач социальной психологии [9], теории кодирования [13], вычислительной биологии [6, 12], кластеризации документов [4], кластеризации многомерных данных [10] и др.

Задачи кластеризации вершин графа относятся к классу NP-трудных, отыскание их точных решений представляет собой весьма сложную проблему. Поэтому возрастает актуальность построения эффективных приближенных алгоритмов и получения априорных гарантированных оценок точности этих алгоритмов, а также их экспериментального исследования.

Цель работы. Целью диссертации является исследование различных вариантов задач кластеризации вершин графа, разработка и анализ точных и приближенных алгоритмов решения этих задач.

Методы исследования. При выполнении работы использовались методы дискретной оптимизации, теории графов, а также методы экспериментального исследования алгоритмов с применением современной вычислительной техники.

Основные результаты работы.

1. Для задачи кластеризации вершин графа, в которой число кластеров не превосходит 3, предложены два полиномиальных приближенных алгоритма. Получены априорные гарантированные оценки точности этих алгоритмов.

2. Для задачи кластеризации вершин графа на два кластера разработаны

процедура локального поиска и два полиномиальных приближенных алгоритма с гарантированными оценками точности.

3. Предложены приближенные алгоритмы с частичным обучением для нового варианта задачи кластеризации вершин графа, в котором число кластеров равно 2. Получены априорные гарантированные оценки точности этих алгоритмов.

4. Предложены два точных метода решения различных вариантов задачи кластеризации вершин графа. Первый использует идею метода ветвей и границ, а второй опирается на известные и новые модели целочисленного линейного программирования рассматриваемых задач. На сериях случайных графов проведено сравнение среднего времени работы точных методов, а также экспериментальное исследование качества решений, найденных рассмотренными в работе приближенными алгоритмами.

Научная новизна. В диссертационной работе разработаны новые полиномиальные алгоритмы приближенного решения различных вариантов задачи кластеризации вершин графа, а также алгоритмы их точного решения. Рассмотрена новая постановка задачи и предложены новые алгоритмы кластеризации с частичным обучением. Получены новые априорные гарантированные оценки точности разработанных приближенных алгоритмов.

Практическая и теоретическая ценность. Полученные в диссертации теоретические результаты применимы в научных исследованиях, а также в учебном процессе. Предложенные алгоритмы могут быть использованы при решении задач достаточно большой размерности.

Апробации работы. Основные результаты диссертации докладывались на IV и V региональных конференциях магистрантов, аспирантов и молодых ученых «ФМХ ОмГУ» (Омск, 2016 и 2017); I и IV Всероссийских научно-практических конференциях «Омские научные чтения» (Омск, 2017 и 2020); VII Международной конференции «Проблемы оптимизации и их приложения (ОРТА 2018)» (Омск, 2018); XVIII Международной конференции «Mathematical Optimization Theory and Operations Research (MOTOR 2019)» (Екатеринбург, 2019); XIX Международной конференции «Mathematical Optimization Theory and Operations Research (MOTOR 2020)» (Новосибирск, 2020); XX Международной конференции «Mathematical Optimization Theory and Operations Research (MOTOR 2021)» (Иркутск, 2021), на объединенном семинаре «Моделирование систем информатики» ИВМиМГ СО РАН и кафедры вычислительных систем ММФ НГУ, а также на научных семинарах в Институте математики им. С.Л. Соболева СО РАН и его Омском филиале.

Публикации. По теме диссертации автором опубликовано 11 научных ра-

бот, из них 6 статей в рецензируемых научных журналах и изданиях, определенных ВАК. В совместных работах соискателю принадлежат доказательства результатов, включенных в диссертацию. Конфликта интересов с соавторами нет.

Структура и объем работы. Диссертация объемом 123 стр. состоит из введения, трех глав, заключения, списка литературы из 45 наименований и приложения.

Краткое содержание работы

Во введении обосновывается актуальность темы диссертации, приводятся постановки исследуемых задач, содержится обзор известных результатов, относящихся к рассматриваемым задачам.

Будем рассматривать только графы без петель и кратных ребер, т.е. *обыкновенные графы*. Обыкновенный граф называется *кластерным графом*, если каждая его компонента связности является полным графом [12]. Обозначим через $\mathcal{M}(V)$ множество всех кластерных графов на множестве вершин V, $\mathcal{M}_k(V)$ – множество всех кластерных графов на V, имеющих ровно k компонент связности, $\mathcal{M}_{\leqslant k}(V)$ – множество всех кластерных графов на V, имеющих не более k компонент связности, $2 \leqslant k \leqslant |V|$.

Если $G_1 = (V, E_1)$ и $G_2 = (V, E_2)$ – обыкновенные помеченные графы на одном и том же множестве вершин V, то расстояние $\rho(G_1, G_2)$ между ними определяется как $\rho(G_1, G_2) = |E_1 \Delta E_2| = |E_1 \setminus E_2| + |E_2 \setminus E_1|$.

Через $N_G(v)$ обозначим окрестность вершины v, т.е. множество вершин графа $G = (V, E)$, смежных с вершиной v.

Для множеств $V_1, \ldots, V_s \subseteq V$ таких, что $V_i \cap V_j = \emptyset$ для любых $i, j \in \{1, \ldots, s\}$ и $V_1 \cup \ldots \cup V_s = V$, обозначим через $M(V_1, \ldots, V_s)$ кластерный граф из множества $\mathcal{M}_{\leqslant s}(V)$ с компонентами связности, порожденными множествами $V_1, \ldots, V_s$. Сами множества $V_1, \ldots, V_s$ будем называть *кластерами*. Некоторые из V_i могут быть пустыми.

В литературе рассматривались три варианта задачи кластеризации вершин графа, ранее известной как задача аппроксимации графов.

Задача GC (GRAPH CLUSTERING). Для произвольного графа $G = (V, E)$ найти такой граф $M^* \in \mathcal{M}(V)$, что

$$\rho(G, M^*) = \min_{M \in \mathcal{M}(V)} \rho(G, M).$$

Задача $\mathbf{GC_k}$ (k-GRAPH CLUSTERING). Дан произвольный граф $G = (V, E)$ и целое число $k, 2 \leqslant k \leqslant |V|$. Найти такой граф $M^* \in \mathcal{M}_k(V)$, что

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_k(V)} \rho(G, M).$$

Задача $\mathbf{GC_{\leqslant k}}$ ($[k]$-GRAPH CLUSTERING) формулируется аналогично.

Все варианты задачи кластеризации вершин графов являются NP- трудными [1, 4, 8, 11, 12].

В 2004 г. Бансал, Блюм и Чаула [4] разработали 3-приближенный алгоритм решения задачи $\mathbf{GC_{\leqslant 2}}$. Агеевым, Ильевым, Кононовым и Телевниным [1] в 2006 г. было доказано, что для задачи $\mathbf{GC_{\leqslant 2}}$ существует полиномиальная приближенная схема, а Гиотис и Гурусвами [8] предложили полиномиальную приближенную схему для задачи $\mathbf{GC_{\leqslant k}}$ (при любом фиксированном $k \geqslant 2$). В 2008 г. Коулман, Саундерсон и Вирт [7] предложили 2-приближенный алгоритм решения задачи $\mathbf{GC_{\leqslant 2}}$, при этом они указали на сложность полиномиальной приближенной схемы из [8], что лишает ее перспективы практического применения. Для задачи $\mathbf{GC_2}$ Ильев, Ильева и Навроцкая [2] разработали $(3 - \frac{6}{|V|})$-приближенный алгоритм.

В главе 1 изучается вариант задачи кластеризации вершин графа с ограниченным числом кластеров.

§1.1 содержит обзор известных результатов для задачи $\mathbf{GC_{\leqslant 2}}$: 3- приближенный алгоритм **BBC** (Bansal-Blum-Chawla) [4], процедуру локального поиска $\mathbf{LS_{\leqslant 2}(M, X, Y)}$ (Local Search for $\mathbf{GC_{\leqslant 2}}$) [7] и 2-приближенный алгоритм **CSW** (Coleman-Saunderson-Wirth) [7].

В §1.2 предложен 6-приближенный алгоритм для задачи $\mathbf{GC_{\leqslant 3}}$, использующий идеи алгоритмов **BBC** и **CSW** .

Алгоритм $\mathbf{NLS_{\leqslant 3}}$ (Neighbourhood with Local Search for $\mathbf{GC_{\leqslant 3}}$).

Вход: граф $G = (V, E)$, $|V| = n$.
Выход: кластерный граф $M_{NLS} \in \mathcal{M}_{\leqslant 3}(V)$.

Шаг 1. Если $n \leqslant 2$, то $M_1 = G$, иначе переход на шаг 2.
Шаг 2. Для каждой вершины $v \in V$ выполнить:
Шаг 2.1. Положить $V_1 = \{v\} \cup N_G(v)$. Если $V_1 = V$, то M_v – полный граф K_n, иначе переход на шаг 2.2.
Шаг 2.2. Обозначить через G_1 подграф графа G, порожденный множеством $V \setminus V_1$. Приближенно решить задачу $\mathbf{GC_{\leqslant 2}}$ на графе G_1 алгоритмом **CSW**, полученный кластерный граф обозначить через $M = M(V_2, V_3)$ (возможно, $V_3 = \emptyset$). Положить $M_v = M(V_1, V_2, V_3)$.
Шаг 3. Среди графов M_v выбрать ближайший к G кластерный граф M_{NLS}:

$$\rho(G, M_{NLS}) = \min_{v \in V} \rho(G, M_v).$$

Теорема 1.3. *Для любого графа $G = (V, E)$ имеет место неравенство*

$$\rho(G, M_{NLS}) \leqslant 6\rho(G, M^*),$$

где $M^* \in \mathcal{M}_{\leqslant 3}(V)$ – оптимальное решение задачи $\mathbf{GC}_{\leqslant \mathbf{3}}$ на графе G, а M_{NLS} – кластерный граф, построенный алгоритмом $\mathbf{NLS}_{\leqslant \mathbf{3}}$.

В §1.3 представлен еще один полиномиальный алгоритм приближенного решения задачи $\mathbf{GC}_{\leqslant \mathbf{3}}$ с лучшей гарантированной оценкой точности, основанный на оригинальной идее и не использующий локальный поиск.

Алгоритм $\mathbf{DN}_{\leqslant \mathbf{3}}$ (Double Neighbourhood for $\mathbf{GC}_{\leqslant \mathbf{3}}$).

Вход: граф $G = (V, E)$, $n = |V|$, $n \geqslant 3$.
Выход: кластерный граф $M_{DN} \in \mathcal{M}_{\leqslant 3}(V)$.

Шаг 1. Для каждой упорядоченной пары вершин $(v, w) \in V \times V, v \neq w$, выполнить:
Шаг 1.1. Положить $V_1 = \{v\} \cup (N_G(v) \setminus \{w\})$. Переход на шаг 1.2.
Шаг 1.2. Обозначить через G_1 подграф графа G, порожденный множеством $V \setminus V_1$. Положить $V_2 = \{w\} \cup N_{G_1}(w)$, $V_3 = V \setminus (V_1 \cup V_2)$ (возможно, $V_3 = \emptyset$). Положить $M_{vw} = M(V_1, V_2, V_3)$.
Шаг 2. Среди построенных графов M_{vw} и графа K_n выбрать ближайший к G кластерный граф $M_{DN} \in \mathcal{M}_{\leqslant 3}(V)$:

$$\rho(G, M_{DN}) = \min_{\substack{(v,w) \in V \times V, \\ v \neq w}} \{\rho(G, M_{vw}), \rho(G, K_n)\}.$$

Теорема 1.4. *При $n \geqslant 3$ для любого n-вершинного графа $G = (V, E)$ имеет место неравенство*

$$\rho(G, M_{DN}) \leqslant (6 - \frac{12}{n})\rho(G, M^*),$$

где $M^* \in \mathcal{M}_{\leqslant 3}(V)$ – оптимальное решение задачи $\mathbf{GC}_{\leqslant \mathbf{3}}$ на графе G, а M_{DN} – кластерный граф, построенный алгоритмом $\mathbf{DN}_{\leqslant \mathbf{3}}$.

В главе 2 рассматриваются задачи кластеризации вершин графа с фиксированным числом кластеров.

В §2.1 исследуется задача $\mathbf{GC_2}$. Для этой задачи предложены два полиномиальных приближенный алгоритма. Первый алгоритм рассматривает лишь кластерные графы из множества $\mathcal{M}_2$ (т.е. лишь допустимые решения задачи $\mathbf{GC_2}$).

Алгоритм $\mathbf{N_2}$ (Neighbourhood for $\mathbf{GC_2}$).

Вход: граф $G = (V, E)$.
Выход: кластерный граф $M_N = M(X, Y) \in \mathcal{M}_2(V)$.

Шаг 1. Для каждой упорядоченной пары вершин $(v, w) \in V \times V, v \neq w$, построить кластерный граф $M_{vw} = M(X, Y) \in \mathcal{M}_2(V)$, где $X = \{v\} \cup (N_G(v) \setminus \{w\}), Y = V \setminus X$.
Шаг 2. Среди всех кластерных графов M_{vw} выбрать ближайший к G кластерный граф M_N:

$$\rho(G, M_N) = \min_{\substack{(v,w) \in V \times V, \\ v \neq w}} \rho(G, M_{vw}).$$

Теорема 2.1. *Для любого графа $G = (V, E)$ имеет место неравенство*

$$\rho(G, M_N) \leqslant 3\rho(G, M^*),$$

где $M^* \in \mathcal{M}_2(V)$ – оптимальное решение задачи $\mathbf{GC_2}$ на графе G, а M_N – кластерный граф, построенный алгоритмом $\mathbf{N_2}$.

Пусть $G = (V, E)$ – произвольный граф. Для вершины $v \in V$ и множества $A \subseteq V$ обозначим через A_v^+ количество таких вершин $u \in A$, что $vu \in E$. Через A_v^- обозначим число таких вершин $u \in A$, что $vu \notin E$.

Для того, чтобы сформулировать второй приближенный алгоритм, нам понадобится следующая процедура локального поиска.

Процедура $\mathbf{LS_2(M,X,Y,Z_1,Z_2)}$ (Local search for 2 components).

Вход: кластерный граф $M = M(X, Y) \in \mathcal{M}_2(V)$, $Z_1 \subseteq X, Z_2 \subseteq Y$.
Выход: кластерный граф $M' = M(X', Y') \in \mathcal{M}_2(V)$.

Итерация 0. Положить $X_0 = X, Y_0 = Y$.
Итерация $\boldsymbol{k}$($\boldsymbol{k \geqslant 1}$).
Шаг 1. Для каждой вершины $u \in V \setminus (Z_1 \cup Z_2)$ вычислить следующую величину $\delta_k(u)$ (изменение значения целевой функции при переносе вершины u в другой кластер):

$$\delta_k(u) = \begin{cases} (X_{k-1})_u^- - (X_{k-1})_u^+ + (Y_{k-1})_u^+ - (Y_{k-1})_u^- & \text{для } u \in X_{k-1} \setminus Z_1, \\ (Y_{k-1})_u^- - (Y_{k-1})_u^+ + (X_{k-1})_u^+ - (X_{k-1})_u^- & \text{для } u \in Y_{k-1} \setminus Z_2. \end{cases}$$

Шаг 2. Выбрать вершину $u_k \in V \setminus (Z_1 \cup Z_2)$ такую, что

$$\delta_k(u_k) = \max_{u \in V \setminus (Z_1 \cup Z_2)} \delta_k(u).$$

Шаг 3. Если $\delta_k(u_k) \leqslant 0$, то **СТОП.** Положить $X' = X_{k-1}$, $Y' = Y_{k-1}$, $M' = M(X', Y')$. **Конец.**

Шаг 4. Если $u_k \in X_{k-1}$, то положить $X_k = X_{k-1} \setminus \{u_k\}$, $Y_k = Y_{k-1} \cup \{u_k\}$. Если же $u_k \in Y_{k-1}$, то положить $X_k = X_{k-1} \cup \{u_k\}$, $Y_k = Y_{k-1} \setminus \{u_k\}$. **Перейти на итерацию $k+1$.**

Теперь можно описать еще один приближенный алгоритм.

Алгоритм $\mathbf{NLS_2}$ (Neighbourhood with Local Search for $\mathbf{GC_2}$).

Вход: граф $G = (V, E)$.
Выход: кластерный граф $M_{NLS} = M(X, Y) \in \mathcal{M}_2(V)$.

Шаг 1. Пусть F – множество всех допустимых решений, построенных алгоритмом $\mathbf{N_2}$. Применить процедуру $\mathbf{LS_2}$ к каждому кластерному графу из множества F.
Шаг 2. Среди всех локальных оптимумов, построенных на шаге 1, выбрать ближайший к G кластерный граф M_{NLS}.

Теорема 2.2. *Для любого графа $G = (V, E)$ верно следующее неравенство:*

$$\rho(G, M_{NLS}) \leqslant 2\rho(G, M^*),$$

где $M_{NLS} \in \mathcal{M}_2(V)$ – решение, построенное алгоритмом $\mathbf{NLS_2}$, а $M^* \in \mathcal{M}_2(V)$ – оптимальное решение задачи $\mathbf{GC_2}$ на графе G.

В отличие от доказательства гарантированной оценки точности алгоритма Коулмана, Саундерсона и Вирта [7], доказательство этой теоремы не использует технику переключений, неприменимую для задачи $\mathbf{GC_2}$.

В §2.2 исследуются новые задачи и методы кластеризации вершин графа с частичным обучением.

Задача $\mathbf{SSGC_k}$ (k-SET SEMI-SUPERVISED GRAPH CLUSTERING). Дан обыкновенный граф $G = (V, E)$ и целое число k, $2 \leqslant k \leqslant |V|$. Выделено семейство $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$ попарно непересекающихся непустых подмножеств множества V. Требуется найти такой граф $M^* \in \mathcal{M}_k(V)$, что

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_k(V)} \rho(G, M),$$

где минимум берется по кластерным графам $M = (V, E_M) \in \mathcal{M}_k(V)$, в которых все множества семейства $\mathcal{Z}$ являются подмножествами множеств вершин разных компонент связности (т.е. разных кластеров) графа M.

Задача $\mathbf{SGC_k}$ (k-SEMI-SUPERVISED GRAPH CLUSTERING) формулируется аналогично при $|Z_1| = \cdots = |Z_k| = 1$.

В случае $k = 2$ для задач $\mathbf{SGC_2}$ и $\mathbf{SSGC_2}$ предложены два полиномиальных приближенных алгоритма.

Алгоритм $\mathbf{NS_2}$ (Neighbourhood semi-supervised for $\mathbf{SGC_2}$ and $\mathbf{SSGC_2}$).

Вход: граф $G = (V, E), Z_1, Z_2$ – непустые непересекающиеся подмножества множества V.

Выход: кластерный граф $M_{NS} = M(X, Y) \in \mathcal{M}_2(V)$, Z_1, Z_2 – подмножества разных кластеров.

Шаг 1. Для каждой вершины $v \in V$ выполнить:

(а) Если $v \notin Z_1 \cup Z_2$, то построить два кластерных графа $\overline{M}_v = M(\overline{X}, \overline{Y})$ и $\overline{\overline{M}}_v = M(\overline{\overline{X}}, \overline{\overline{Y}})$, где

$$\overline{X} = \{v\} \cup ((N_G(v) \cup Z_1) \setminus Z_2), \overline{Y} = V \setminus \overline{X},$$
$$\overline{\overline{X}} = \{v\} \cup ((N_G(v) \cup Z_2) \setminus Z_1), \overline{\overline{Y}} = V \setminus \overline{\overline{X}}.$$

(б) Если $v \in Z_1 \cup Z_2$, то построить граф $M_v = M(X, Y)$, где

$$X = \{v\} \cup ((N_G(v) \cup Z) \setminus \overline{Z}), Y = V \setminus X.$$

Здесь $Z = Z_1, \overline{Z} = Z_2$ при $v \in Z_1$, или $Z = Z_2, \overline{Z} = Z_1$ при $v \in Z_2$.

Шаг 2. Среди всех кластерных графов, построенных на шаге 1, выбрать ближайший к G кластерный граф $M_{NS} = M(X, Y)$.

Теорема 2.3. *Для любого графа $G = (V, E)$ имеет место неравенство*

$$\rho(G, M_{NS}) \leqslant 3\rho(G, M^*),$$

где $M^* \in \mathcal{M}_2(V)$ – оптимальное решение задачи $\mathbf{SGC_2}$ или $\mathbf{SSGC_2}$ на графе G, а M_{NS} – кластерный граф, построенный алгоритмом $\mathbf{NS_2}$.

Алгоритм $\mathbf{NSLS_2}$ (Neighbourhood semi-supervised with Local Search for $\mathbf{SGC_2}$ and $\mathbf{SSGC_2}$).

Вход: граф $G = (V, E), Z_1, Z_2$ – непустые непересекающиеся подмножества множества V.

Выход: кластерный граф $M_{NSLS} = M(X, Y) \in \mathcal{M}_2(V)$, Z_1, Z_2 – подмножества разных кластеров.

Шаг 1. Пусть F – множество всех допустимых решений, построенных алгоритмом $\mathbf{NS_2}$. Применить процедуру $\mathbf{LS_2}$ к каждому кластерному графу из множества F.

Шаг 2. Среди всех локальных оптимумов, построенных на шаге 1, выбрать ближайший к G кластерный граф M_{NSLS}.

Теорема 2.4. *Для любого графа $G = (V, E)$ имеет место неравенство*

$$\rho(G, M_{NSLS}) \leqslant 2\rho(G, M^*),$$

где $M^* \in \mathcal{M}_2(V)$ – оптимальное решение задачи $\mathbf{SGC_2}$ или $\mathbf{SSGC_2}$ на графе G, а M_{NSLS} – кластерный граф, построенный алгоритмом $\mathbf{NSLS_2}$.

В §2.3 исследованы взаимосвязи между задачами $\mathbf{GC_2}$, $\mathbf{SGC_2}$ и $\mathbf{SSGC_2}$. Задача $\mathbf{SGC_k}$ является частным случаем задачи $\mathbf{SSGC_k}$ при $|Z_1| = \cdots = |Z_k| = 1$. Также, решая задачу $\mathbf{SGC_2}$ для каждой упорядоченной пары вершин $(u, v) \in V \times V$ некоторого графа $G = (V, E)$, мы тем самым можем найти решение для задачи $\mathbf{GC_2}$, поскольку хотя бы одна из пар вершин (u, v) будет содержать вершины, принадлежащие разным кластерам оптимального решения задачи $\mathbf{GC_2}$.

В главе 3 предложены и исследуются точные методы решения задач $\mathbf{GC}$, $\mathbf{GC_{\leqslant k}}$, $\mathbf{GC_k}$, $\mathbf{SGC_k}$ и $\mathbf{SSGC_k}$, а также описаны результаты экспериментального исследования.

В §3.1 предложены два подхода к нахождению точных решений для различных вариантов задачи кластеризации вершин графа. Первый подход основан на следующей универсальной схеме метода ветвей и границ, способной находить оптимальное решение для любой из рассматриваемых задач для произвольного графа $G = (V, E), |V| = n$. Через $(S_1, \ldots, S_k)$ обозначим разбиение подмножества множества вершин V графа G на k множеств, а через $(I_1, \ldots, I_k)$ – начальное разбиение.

Алгоритм $\mathbf{BBM(G, (I_1, \ldots, I_k))}$.

Шаг 1. Положить $S_1 = I_1, \ldots, S_k = I_k$; // $k = n$ для задачи $\mathbf{GC}$
Шаг 2. $record = \frac{n(n-1)}{2}$;
Шаг 3. $A = \{j : S_j \neq \emptyset\}$; // множество индексов непустых кластеров
Шаг 4. $B = \{1, \ldots, k\} \setminus A$; // множество индексов пустых кластеров
Шаг 5. $\mathbf{Branch((S_1, \ldots, S_k), record, A, B)}$.

Процедура $\mathbf{Branch((S_1, \ldots, S_k), record, A, B)}$.
Если $S_1 \cup \ldots \cup S_k \neq V$:
Шаг 1. Выбрать $v \notin V \setminus (S_1 \cup \ldots \cup S_k)$;
Шаг 2. Для каждого $i \in A$:
Шаг 2.1. $b = \mathbf{Bound}((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k})$;
Шаг 2.2. Если $(b < record)$
$\mathbf{Branch((S_1, \ldots, S_i \cup \{v\}, \ldots, S_k), record, A, B)}$;
Шаг 3. Если $B \neq \emptyset$: // выбрать любое пустое множество S_i
Шаг 3.1. Взять произвольный $i \in B$;
Шаг 3.2. $b = \mathbf{Bound}((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k})$;
Шаг 3.3. Если $(b < record)$
$\mathbf{Branch((S_1, \ldots, S_i \cup \{v\}, \ldots, S_k), record, A, B)}$;
иначе
Шаг 4. $b = \mathbf{Bound}((\mathbf{S_1}, \ldots, \mathbf{S_k})$;

Шаг 5. Если $(\mathbf{updateRecord}(\mathbf{record}, \mathbf{b}, (\mathbf{S_1}, \ldots, \mathbf{S_k})))$ $record = b$.

Второй подход к поиску оптимальных решений опирается на модели целочисленного линейного программирования. В 2005 г. Чарикар, Гурусвами и Вирт [5] предложили модель булева программирования для задачи **GC**, введя следующие бинарные переменные: x_{ij} соответствует каждой паре вершин i и j. Если вершины i и j находятся в одном кластере, то $x_{ij} = 0$, иначе $x_{ij} = 1$ при $i \neq j$ (по умолчанию $x_{ii} = 0$). Легко видеть, что если $x_{ij} = 0$ и $x_{jr} = 0$, то и $x_{ir} = 0$, а значит $x_{ir} \leqslant x_{ij} + x_{jr}$ выполняется для каждой упорядоченной тройки вершин i, j, r. Таким образом, мы можем построить модель булева программирования для задачи $\mathbf{GC_{\leqslant k}}$.

$$\sum_{ij \in E} x_{ij} + \sum_{ij \notin E} (1 - x_{ij}) \to \min$$

$$x_{ir} \leqslant x_{ij} + x_{jr}, \qquad i, j, r \in \{1, \ldots, n\}$$

$$x_{i_1 i_2} + \ldots x_{i_{k-1} i_k} \leqslant \frac{(k+2)(k-1)}{2}, \qquad i_1, \ldots, i_k \in \{1, \ldots, n\}$$

$$x_{ij} \in \{0, 1\}, \qquad i, j \in \{1, \ldots, n\}$$

Аналогичные модели построены и для других задач.

В §3.2 приводятся результаты вычислительного эксперимента о влиянии локального поиска на точность и время работы алгоритмов. Так, для задачи $\mathbf{GC_{\leqslant 2}}$ для сравнения с алгоритмами **BBC** (Bansal-Blum-Chawla) [4] и **CSW** (Coleman-Saunderson-Wirth) [7] был предложен эвристический алгоритм $\mathbf{N1LS_{\leqslant 2}}$ (Neighbourhood with one Local Search for $\mathbf{GC_{\leqslant 2}}$). Ключевое изменение этого алгоритма в сравнении с алгоритмом **CSW** – процедура локального поиска $\mathbf{LS_{\leqslant 2}}$ применяется лишь к лучшему допустимому решению, полученному алгоритмом **BBC**, что значительно сокращает время его работы. Экспериментальное исследование проводилось в два этапа: предварительный эксперимент на графах малой размерности и основной эксперимент на графах большей размерности.

Предварительный эксперимент проводился на графах размерности от 15 до 50 вершин, по 100 графов в серии. Для таких графов алгоритмом **BBM** и с помощью решателя Gurobi, использующего модель целочисленного линейного программирования, удалось найти точные решения. Стоит отметить, что при $n = 31$ время работы решателя Gurobi в среднем было равно 2000 сек., и для графов большей размерности он не использовался. Время работы алгоритма **BBM** при $n = 50$ в среднем было равно 1781 сек.

Точностью алгоритма будем называть отношение значения целевой функции на решении, полученном этим алгоритмом, к оптимальному значению.

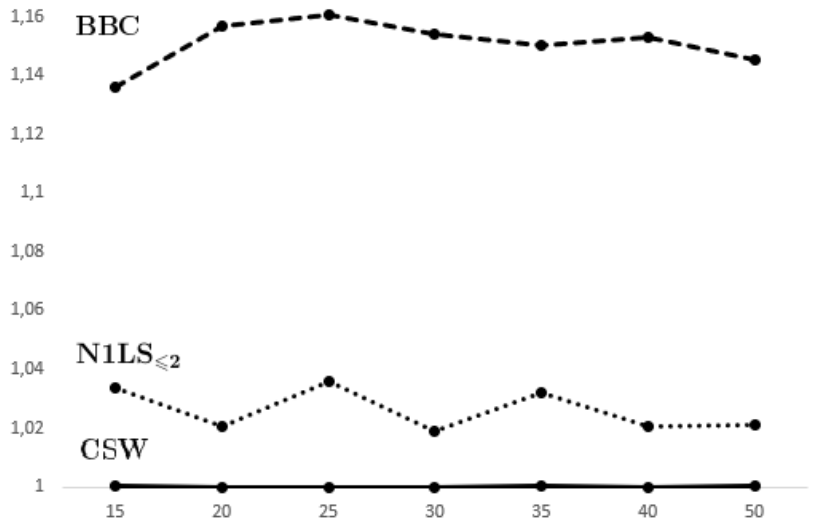

Рис. 1: Средняя точность алгоритмов **BBC**, **CSW** и $\mathbf{N1LS_{\leqslant 2}}$ на графах малой размерности.

Обозначим через $\delta_{\mathbf{BBC}}(n), \delta_{\mathbf{CSW}}(n)$ и $\delta_{\mathbf{N1LS_{\leqslant 2}}}(n)$ точности алгоритмов **BBC**, **CSW** и $\mathbf{N1LS_{\leqslant 2}}$ соответственно.

По итогам предварительного эксперимента удалось получить представления о характере изменения средней точности алгоритмов (рис 1.). Среднее отклонение от оптимума алгоритма **CSW** близко к нулю и достигает максимального значения 0.06% при $n = 15$. Среднее отклонение от оптимума алгоритма $\mathbf{N1LS_{\leqslant 2}}$ также достаточно близко к нулю и достигает максимума 3.6% при $n = 25$. При том же значении n достигает максимума среднее отклонение от оптимума алгоритма **BBC** и составляет 16%. Очевидно, что точность алгоритма **CSW** всегда не меньше, чем точность алгоритмов **BBC** и $\mathbf{N1LS_{\leqslant 2}}$, поэтому в качестве объекта дальнейшего исследования были выбраны две случайные величины:

$$d_{\mathbf{BBC}}(n) = \frac{\delta_{\mathbf{BBC}}(n)}{\delta_{\mathbf{CSW}}(n)}, \; d_{\mathbf{N1LS_{\leqslant 2}}}(n) = \frac{\delta_{\mathbf{N1LS_{\leqslant 2}}}(n)}{\delta_{\mathbf{CSW}}(n)}.$$

В эксперименте на графах большей размерности (при n от 100 до 6000, решалось по 100 задач в серии) исследовалось поведение случайных величин $d_{\mathbf{BBC}}(n)$ и $d_{\mathbf{N1LS_{\leqslant 2}}}(n)$. Тенденции изменения средних значений и границ доверительных интервалов при уровне значимости 0.05 представлены на рис. 2. Комментируя результаты, можно сказать следующее.

С ростом n обе исследуемые случайные величины уменьшаются, а ширина доверительного интервала сужается настолько, что интервалы фактически невозможно увидеть на рисунке. Величина $d_{\mathbf{N1LS_{\leqslant 2}}}(n)$ находится в окрестности единицы, что позволяет говорить о том, что точность алгоритма $\mathbf{N1LS_{\leqslant 2}}$ стремится к точности алгоритма **CSW** с ростом n. Учитывая, что

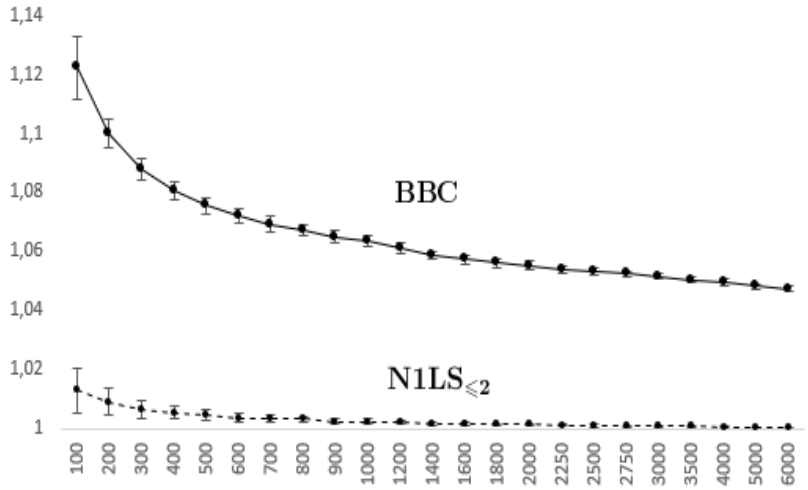

Рис. 2: Средние значения случайных величин $d_{\mathbf{BBC}}(n)$ и $d_{\mathbf{N1LS_{\leqslant 2}}}(n)$.

при $n = 6000$ среднее время работы алгоритма $\mathbf{N1LS_{\leqslant 2}}$ составило 149.21 сек, а среднее время работы алгоритма **CSW** составило 580.68 сек, использование алгоритма $\mathbf{N1LS_{\leqslant 2}}$ является наиболее предпочтительным.

Экспериментальное исследование показало, что решения, найденные алгоритмами **CSW** и $\mathbf{N1LS_{\leqslant 2}}$, как правило, очень близки к оптимальным.

Для задач $\mathbf{GC_{\leqslant 3}}$, $\mathbf{GC_2}$ и $\mathbf{SGC_2}$ были проведены аналогичные экспериментальные исследования.

В заключении приведены основные результаты диссертации.

Список литературы

[1] Агеев А.А., Ильев В.П., Кононов А.В., Талевнин А.С. Вычислительная сложность задачи аппроксимации графов // Дискретный анализ и исследование операций. Сер. 1. 2006. Т. 13. N 1. С. 3–15.

[2] Ильев В.П., Ильева С.Д., Навроцкая А.А. Приближенные алгоритмы для задач аппроксимации графов // Дискретный анализ и исследование операций. 2011. Т. 18. N 1. С. 41–60.

[3] Ailon N., Charikar M., Newman A. Aggregating inconsistent information: Ranking and clustering // Journal of ACM. 2008. V. 55. N 5. P. 1–27.

[4] Bansal N., Blum A., Chawla Sh. Correlation clustering // Machine Learning. 2004. V. 56. P. 89–113.

[5] Charikar M., Guruswami V., Wirth A. Clustering with qualitative information// Journal of Computer and System Sciences. 2005. V. 71. N 3. P. 360–383.

[6] Chen Z.-Z., Jiang T., Lin G. Computing philogenetic roots with bounded degrees and errors // SIAM Journal on Computing. 2003. V. 32. N 4. P. 864–879.

[7] Coleman T., Saunderson J., Wirth A. A local-search 2-approximation for 2-correlation-clustering // Algorithms – ESA 2008: Lecture Notes in Computer Science. 2008. V. 5193. P. 308–319.

[8] Giotis I., Guruswami V. Correlation clustering with a fixed number of clusters // Theory of Computing. 2006. V. 2. N 1. P. 249–266.

[9] Harary F. On the notion of balance of a signed graph // Michigan Mathematical Journal. 1953. N 2. P. 143–146.

[10] Kriegel H. P., Kroger P., Zimek, A. Clustering high-dimensional data // ACM Transactions on Knowledge Discovery from Data. 2009. N 3. P. 1–58.

[11] Křivánek M., Morávek J. NP-hard problems in hierarchical-tree clustering // Acta informatica. 1986. V. 23. P. 311-323.

[12] Shamir R., Sharan R., Tsur D. Cluster graph modification problems // Discrete Applied Mathematics. 2004. V. 144. N 1–2. P. 173–182.

[13] Sole P., Zaslavsky T. A coding approach to signed graphs // SIAM Journal on Descrete Mathematics. 1994. N 7. P. 544–553.

Публикации автора по теме диссертации

Статьи в изданиях, рекомендованных ВАК

1. Ильев В.П., Ильева С.Д., Моршинин А.В. 2-приближённые алгоритмы для двух задач кластеризации на графах // Дискретный анализ и исследование операций. 2020. Т. 27. N 3. С. 88–108.
2. Ильев В.П., Ильева С.Д., Моршинин А.В. Алгоритмы приближённого решения одной задачи кластеризации графа // Прикладная дискретная математика. 2019. N 45. С. 64–77.
3. Моршинин А.В. Об одной задаче кластеризации графа // Вестник Омского университета. 2018. Т. 23. N 1. С. 4–9.

4. Моршинин А.В. Точные алгоритмы для задач кластеризации вершин графа // Вестник Омского университета. 2021. Т. 26. N 2. С. 23–29.
5. Il'ev V., Il'eva S., Morshinin A. A 2-approximation algorithm for the graph 2-clustering problem // In: M. Khachay et al. (Eds.) MOTOR 2019. Lecture Notes in Computer Science. Springer. 2019. V. 11548. P. 295—308.
6. Il'ev V., Il'eva S., Morshinin A. An approximation algorithm for a semi-supervised graph clustering problem // In: Yu. Kochetov et.al. (Eds.) MOTOR 2020. Communications in Computer and Information Science. Springer. 2020. V. 1275. P. 23–29.

Тезисы и труды конференций

7. Моршинин А.В. Приближенное решение одной задачи кластеризации графа // Всероссийская научно-практическая конференция «Омские научные чтения». Материалы конференции. Омск 2017. С. 1041–1043.
8. Моршинин А.В. Метод ветвей и границ для задач кластеризации вершин графа // Четвертая всероссийская научно-практическая конференция «Омские научные чтения». Материалы конференции. Омск 2020. С. 2161–2165.
9. Моршинин А.В. Алгоритм приближенного решения одной задачи кластеризации графа // IV региональная конференция магистрантов, аспирантов и молодых ученых по физике, математике и химии «ФМХ ОмГУ 2016». Сборник статей конференции. Омск 2016. С. 15–18.
10. Моршинин А.В. Приближенное решение задачи кластеризации графа // V региональная конференция магистрантов, аспирантов и молодых ученых по физике, математике и химии «ФМХ ОмГУ 2017». Сборник статей конференции. Омск 2017. С. 15–18.
11. Ильев В.П., Ильева С.Д., Моршинин А.В. Одна задача кластеризации с частичным обучением // VII Международная конференция «Проблемы оптимизации и их приложения (ОРТА 2018)». Тезисы докладов конференции. Омск 2018. С. 85.

Моршинин Александр Владимирович

ПРИБЛИЖЕННОЕ И ТОЧНОЕ РЕШЕНИЕ РАЗЛИЧНЫХ ВАРИАНТОВ ЗАДАЧИ КЛАСТЕРИЗАЦИИ ВЕРШИН ГРАФА

Автореферат
диссертации на соискание ученой степени
кандидата физико-математических наук

Отпечатано с оригинал-макета,
предоставленного автором

Подписано в печать Формат 60×84 1/16.
Печ. л. 1,0. Уч.-изд. л. 1,0. Тираж 100 экз. Заказ № .

ООО «Издательский центр КАН»
644077, г. Омск, ул. Красный Путь 30
email: pc_kan@mail.ru