РОССИЙСКАЯ АКАДЕМИЯ НАУК
СИБИРСКОЕ ОТДЕЛЕНИЕ
ИНСТИТУТ МАТЕМАТИКИ им. С.Л. СОБОЛЕВА

---

На правах рукописи

**Моршинин Александр Владимирович**

# ТОЧНОЕ И ПРИБЛИЖЕННОЕ РЕШЕНИЕ РАЗЛИЧНЫХ ВАРИАНТОВ ЗАДАЧИ КЛАСТЕРИЗАЦИИ ВЕРШИН ГРАФА

01.01.09 – дискретная математика и математическая кибернетика

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Новосибирск – 2021

Работа выполнена в федеральном государственном бюджетном образовательном учреждении науки Институт математики им. С.Л. Соболева Сибирского отделения Российской академии наук.

Научный руководитель: доктор физико-математических наук
Ильев Виктор Петрович.

Официальные оппоненты: доктор физико-математических наук
_____,

кандидат физико-математических наук
_____.

Ведущая организация: _____

Защита диссертации состоится « » _____ 2021 г. в ____ часов на заседании диссертационного совета Д 003.015.01 при Федеральном государственном бюджетном учреждении науки Институт математики им. С.Л. Соболева Сибирского отделения Российской академии наук (630090, Новосибирск, пр. академика Коптюга, 4).

С диссертацией можно ознакомиться в библиотеке Института математики им. С.Л. Соболева СО РАН.

Автореферат разослан 2021 г.

Ученый секретарь
диссертационного совета

_____ _____

## Общая характеристика работы

**Актуальность темы.** В диссертационной работе исследуются различные варианты задачи кластеризации вершин графа. Эти задачи 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 задаче 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 сложных практических задач, описывающими процессы в реальных системах. Задачи кластеризации вершин графа относятся к классу $NP$-трудных, поэтому отыскание точного решения представляет собой весьма сложную проблему. В то же время возрастает актуальность построения эффективных приближенных алгоритмов и получения гарантированных оценок точности этих алгоритмов, а также их экспериментального исследования.

**Цель работы.** Целью диссертации является исследование различных вариантов задачи кластеризации вершин графа, а также 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.

**Научная новизна.** В диссертационной работе предложены новые полиномиальные алгоритмы приближенного решения различных вариантов задачи кластеризации вершин графа. Получены гарантированные оценки точности приближенных алгоритмов. Рассмотрена новая постановка задачи кластеризации вершин графа – задача кластеризации вершин графа с частичным обучением. Разработаны два метода нахождения точных решений этих задач.

**Основные результаты работы.**

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

2. Предложена универсальная процедура локального поиска, применимая для разных вариантов задачи кластеризации вершин графа, в которой число кластеров равно 2.

3. Для задачи кластеризации вершин графа, в которой число кластеров равно 2, и для задачи кластеризации вершин графа с частичным обучением с 2 компонентами предложено несколько приближенных алгоритмов с гарантированными априорными оценками точности. Исследована взаимосвязь между этими задачами.

4. Предложены два алгоритма нахождения оптимальных решений различных вариантов задачи кластеризации вершин графа. Первый алгоритм основан на методе ветвей и границ, второй опирается на модели целочисленного линейного программирования.

5. Предложены эвристические алгоритмы решения задач кластеризации вершин графа. Проведено экспериментальное исследование точных и 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 региональной конференции магистрантов, аспирантов и молодых ученых по физике, математике и химии «ФМХ ОмГУ 2016» (Омск 2016); V региональной конференции магистрантов, аспирантов и молодых ученых по физике, математике и химии «ФМХ ОмГУ 2017» (Омск 2017); Всероссийской научно-практической конференции «Омские научные чтения» (Омск 2017); VII Международной конференции «Проблемы оптимизации и их приложения (OPTA 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); IV Всероссийской научно-практической конференции «Омские научные чтения» (Омск 2020), а также на научных семинарах в Омском государственном университете им. Ф.М.Достоевского, в Институте математики им. С.Л. Соболева СО РАН и его Омском филиале.

**Публикации.** По теме диссертации автором опубликовано 10 научных работ, из них 5 в рецензируемых изданиях из списка ВАК. Конфликта интересов с соавторами нет, в совместных работах соискателю принадлежат идеи доказательств результатов, включенных в диссертацию.

**Структура и объем работы.** Диссертация состоит из введения,

трех глав, заключения, списка литературы и приложения.

## Краткое содержание работы

**Во введении** обосновывается актуальность темы диссертации, приводятся постановки исследуемых задач, содержится обзор известных результатов, посвященных 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 [9]. 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 $k$ непустых компонент связности, $\mathcal{M}_{\leqslant k}(V)$ – множество всех кластерных графов на $V$, имеющих не более $k$ компонент связности, $2 \leqslant k \leqslant |V|$.

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 $vu \notin E$.

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

$$\rho(G_1, G_2) = |E_1 \Delta E_2| = |E_1 \setminus E_2| + |E_2 \setminus E_1|,$$

т.е. $\rho(G_1, G_2)$ – это число несовпадающих ребер в графах $G_1$ и $G_2$.

В литературе рассматривались три варианта задачи кластеризации вершин графа.

**Задача GC.** 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 $\mathbf{GC_k}$.** 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 $\mathbf{GC_{\leqslant k}}$** формулируется аналогично.

Все варианты задачи кластеризации вершин графов являются $NP$-трудными [1, 4, 7, 8, 9].

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

**В главе 1** рассмотрен вариант задачи кластеризации вершин графа, в которой число кластеров не превосходит $k$.

**§1.1** содержит обзор известных результатов для задачи $\mathbf{GC}_{\leqslant \mathbf{2}}$: 3- приближенный алгоритм **BBC(Bansal-Blum-Chawla)** [4], процедуру локального поиска $\mathbf{LS}_{\leqslant \mathbf{2}}(\mathbf{M}, \mathbf{X}, \mathbf{Y})$ **(Local Search for no more than 2 components)** [6] и 2-приближенный алгоритм **Алгоритм CSW (Coleman-Saunderson-Wirth)** [6].

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

**Алгоритм $\mathbf{NLS}_{\leqslant \mathbf{3}}$ (Neighborhood with Local Search for $\mathbf{GC}_{\leqslant \mathbf{3}}$).**
**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 $w \in V$ выполнить:
**Шаг 2.1.** $V_1 = \{w\} \cup N_G(w)$. Если $V_1 = V$, то $M_w$ – 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 $\mathbf{GC}_{\leqslant \mathbf{2}}$ на графе $G_1$ алгоритмом **CSW**, полученный кластерный граф обозначить через $M = M(V_2, V_3)$ (возможно, $V_3 = \emptyset$). Положить $M_w = M(V_1, V_2, V_3)$.
**Шаг 3.** Среди всех графов $M_w$ выбрать ближайший к $G$ кластерный граф $M_{NLS}$:

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

**Теорема 1.1.** *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, основанной 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 $(u, v) \in V \times V$ такой, что $u \neq v$, выполнить:
**Шаг 1.1.** Положить $V_1 = \{u\} \cup (N_G(u) \setminus \{v\})$. Переход на шаг 1.2.
**Шаг 1.2.** Обозначить через $G_1$ подграф графа $G$, порожденный множеством $V \setminus V_1$. Положить $V_2 = \{v\} \cup N_{G_1}(v)$, $V_3 = V \setminus (V_1 \cup V_2)$ ($V_3$ может быть пустым). Положить $M_{uv} = M(V_1, V_2, V_3)$.
**Шаг 2.** Среди построенных графов $M_{uv}$ и графа $K_n$ выбрать ближайший к $G$ кластерный граф $M_{DN} \in \mathcal{M}_{\leqslant 3}(V)$:

$$\rho(G, M_{DN}) = \min_{(u,v) \in V \times V} \{\rho(G, M_{uv}), \rho(G, K_n)\},$$

где минимум берется по всем парам $(u, v) \in V \times V$ таким, что $u \neq v$.

**Теорема 1.2.** *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, в которых число кластеров равно $k$.

**В §2.1** исследуется задача $\mathbf{GC_2}$. Для этой задачи предложены два полиномиальных приближенный алгоритма. Первый алгоритм в целом похож на алгоритм **BBC**, однако имеет существенное отличие: этот 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 $M_{v,w} = M(X, Y) \in \mathcal{M}_2(V)$, где $X = \{v\} \cup (N_G(v) \setminus \{w\}), Y = V \setminus X$.

**Шаг 2.** Среди всех кластерных графов $M_{v,w}$ выбрать ближайший к $G$ кластерный граф $M_N$:

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

**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* $\mathbf{N_2}$.

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

**Процедура $\mathbf{LS_2}$(M,X,Y,$\mathbf{Z_1}$,$\mathbf{Z_2}$) (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.** Положим $X_0 = X, Y_0 = Y$.
**Итерация $k$.**
**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). При $\delta_k(u) > 0$ эту величину будем называть *локальным улучшением вершины $u$ на итерации $k$*:

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

**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 построить еще один приближенный алгоритм.

**Алгоритм $\mathbf{NLS_2}$ (Neighborhood 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.** Для каждой упорядоченной пары вершин $(v, w) \in V \times V, v \neq w$, выполнить:
**Шаг 1.1.** Построить кластерный граф $M_{v,w} = M(X, Y) \in \mathcal{M}_2(V)$, где $X = \{v\} \cup (N_G(v) \setminus \{w\}), Y = V \setminus X$.
**Шаг 1.2.** Запустить процедуру локального поиска
$\mathbf{LS}(M_{v,w}, X, Y, \{v\}, \{w\})$. Обозначить полученный граф через $M'_{v,w}$.
**Шаг 2.** Среди всех локальных оптимумов $M'_{v,w}$, построенных на шаге 1.2, выбрать ближайший к $G$ кластерный граф $M_{NLS}$:

$$\rho(G, M_{NLS}) = \min_{\substack{(v,w) \in V \times V, \\ v \neq w}} \rho(G, M'_{v,w}).$$

**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 **CSW**, доказательство этой теоремы не использует технику переключений.

**В §2.2** исследуются две новые задачи кластеризации вершин графа с частичным обучением.

**Задача $\mathbf{SGC_k}$.** Дан обыкновенный граф $G = (V, E)$ и целое число $k$, $2 \leqslant k \leqslant |V|$. Выделено множество попарно различных вершин $Z = \{z_1, \ldots, z_k\} \subset V$. 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 всем кластерным графам $M = (V, E_M) \in \mathcal{M}_k(V)$, в которых никакие две вершины множества $Z = \{z_1, \ldots, z_k\}$ не принадлежат одной и той же компоненте связности (т. е. одному кластеру) графа $M$.

**Задача $\mathbf{SSGC_k}$.** 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 всем кластерным графам $M = (V, E_M) \in \mathcal{M}_k(V)$, в которых все множества семейства $\mathcal{Z}$ являются подмножествами разных компонент связности (т.е. разных кластеров) графа $M$.

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

**Алгоритм $\mathbf{NS_2}$ (Neighborhood 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 $Z = Z_1, \overline{Z} = Z_2$ если $v \in Z_1$, или $Z = Z_2, \overline{Z} = Z_1$ если $v \in Z_2$.

**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 $\mathbf{NSLS_2}$ (Neighborhood 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.** Для каждой вершины $v \in V$ выполнить:

**Шаг 1.1.** (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 $Z = Z_1, \overline{Z} = Z_2$ если $v \in Z_1$, или $Z = Z_2, \overline{Z} = Z_1$ если $v \in Z_2$.
**Шаг 1.2.** (а) Если $v \notin Z_1 \cup Z_2$, то дважды применить процедуру локального поиска $\mathbf{LS_2}(\overline{\mathbf{M}}_\mathbf{v}, \overline{\mathbf{X}}, \overline{\mathbf{Y}}, \mathbf{Z_1}, \mathbf{Z_2})$ и $\mathbf{LS_2}(\overline{\overline{\mathbf{M}}}_\mathbf{v}, \overline{\overline{\mathbf{X}}}, \overline{\overline{\mathbf{Y}}}, \mathbf{Z_1}, \mathbf{Z_2})$. Полученные графы обозначить через $M'_v$ и $M''_v$.
(б) Если $v \in Z_1 \cup Z_2$, то применить процедуру локального поиска $\mathbf{LS_2}(\mathbf{M}_\mathbf{v}, \mathbf{X}, \mathbf{Y}, \mathbf{Z_1}, \mathbf{Z_2})$. Обозначить полученный граф через $M_v$.
**Шаг 2.** Среди всех локальных оптимумов, построенных на шаге 1.2, выбрать ближайший к $G$ кластерный граф $M_{NSLS} = M(X, Y)$.

**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** показана связь 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 неупорядоченной пары вершин $(v, w) \in V \times V$ некоторого графа $G = (V, E)$, мы тем самым находим решение для задачи $\mathbf{GC_2}$, поскольку хотя бы одна из пар вершин $(v, w)$ будет содержать вершины, принадлежащие разным кластерам оптимального решения задачи $\mathbf{GC_2}$.

**В главе 3** предложены и исследуются точные алгоритмы для 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 алгоритма нахождения точных решений для различных вариантов задачи кластеризации вершин графа. Первый алгоритм основан на следующем универсальном методе ветвей и границ, способном находить оптимальное решение для любой из рассматриваемых задач для произвольного графа $G = (V, E), |V| = n$. Через

$(S_1, \ldots, S_k)$ обозначим частичную кластеризацию вершин графа $G$ на $k$ кластеров, а через $(I_1, \ldots, I_k)$ – начальную кластеризацию.

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

Шаг 1. Положить $S_1 = I_1, \ldots, S_k = I_k$; // $k = n$ для задачи **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. **Branch**$((\mathbf{S_1}, \ldots, \mathbf{S_k}), \mathbf{record}, \mathbf{A}, \mathbf{B})$.

**Процедура Branch**$((\mathbf{S_1}, \ldots, \mathbf{S_k}), \mathbf{record}, \mathbf{A}, \mathbf{B})$.

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. $b =$ **Bound**$((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k})$;

Шаг 2.2. Если $(b < record)$ **Branch**$((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k}), \mathbf{record}, \mathbf{A}, \mathbf{B})$;

Шаг 3. Если $B \neq \emptyset$: // выбрать любое пустое множество $S_i$

Шаг 3.1. Взять произвольный $i \in B$;

Шаг 3.2. $b =$ **Bound**$((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k})$;

Шаг 3.3. Если $(b < record)$ **Branch**$((\mathbf{S_1}, \ldots, \mathbf{S_i} \cup \{\mathbf{v}\}, \ldots, \mathbf{S_k}), \mathbf{record}, \mathbf{A}, \mathbf{B})$; иначе

Шаг 4. $b =$ **Bound**$((\mathbf{S_1}, \ldots, \mathbf{S_k})$;

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

Второй алгоритм 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 множестве разбиения, 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, например, модель булева программирования для задачи $\mathbf{GC}_{\leqslant \mathbf{k}}$.

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

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

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

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

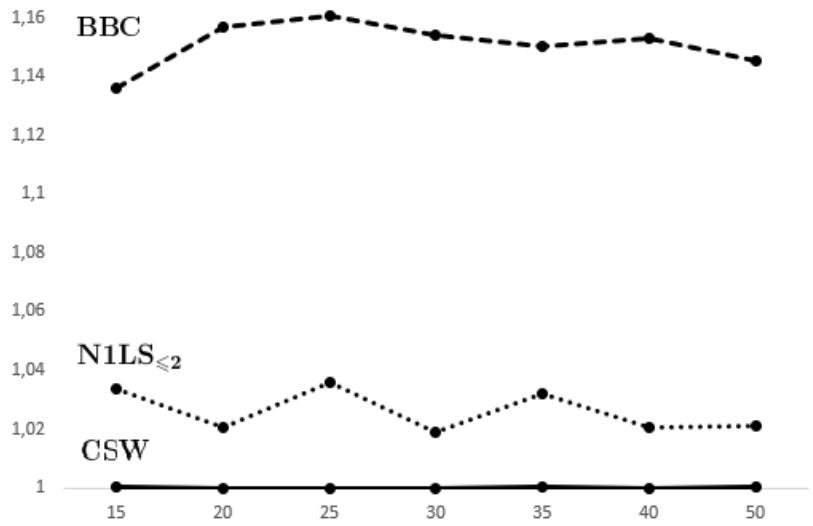


Рис. 1: Средняя точность алгоритмов **BBC**, **CSW** и $\mathbf{N1LS_{\leqslant 2}}$ на графах малой размерности.

Аналогичные модели построены и для других задач.

**В §3.2** приводятся результаты вычислительного эксперимента. Так, для задачи $\mathbf{GC_{\leqslant 2}}$ для сравнения с алгоритмами **BBC** и **CSW** был предложен эвристический алгоритм $\mathbf{N1LS_{\leqslant 2}}$ (Neighborhood with one Local Search for $\mathbf{GC_{\leqslant 2}}$). Ключевое изменение этого алгоритма в сравнении с алгоритмом **CSW** – процедура $\mathbf{LS_{\leqslant 2}}$ 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 большой 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 функции на решении, полученном этим алгоритмом, к оптимальному значению целевой функции. 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.).

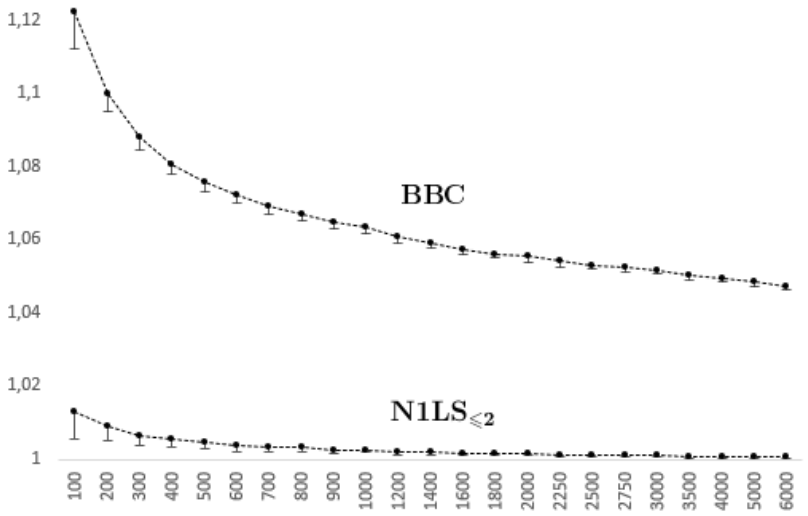


Рис. 2: Средние значения случайных величин $d_{\mathbf{BBC}}(n)$ и $d_{\mathbf{N1LS_{\leqslant 2}}}(n)$.

Средняя отклонение от оптимума алгоритма **CSW** близка к нулю и достигает максимального значения 0,06% при $n = 15$. Среднее отклонение от оптимума алгоритма $\mathbf{N1LS_{\leqslant 2}}$ также достаточна 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 большой 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 при $n = 6000$ среднее время работы алгоритма $\mathbf{N1LS_{\leqslant 2}}$ составило 116 сек., а среднее время работы алгоритма **CSW** составило 493 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 // Дискрет. анализ и исслед. 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 // Дискрет. анализ и исслед. операций. 2011. Т. 18, N 1. С. 41–60.

[3] Ailon N., Charikar M., Newman A. Aggregating inconsistent information: Ranking and clustering // J. ACM. 2008. V. 55, No. 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// J. Comput. Syst. Sci. 2005. V. 71, No. 3. P. 360-383.

[6] 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.

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

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

[9] Shamir R., Sharan R., Tsur D. Cluster graph modification problems // Discrete Appl. Math. 2004. V. 144, N 1-2. P. 173–182.


## Публикации автора по теме диссертации

1. 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.
2. 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.
3. Моршинин А.В. Приближенное решение одной задачи кластеризации графа // Всероссийская научно-практическая конференция «Омские научные чтения». Материалы конференции. Омск 2017. С. 1041-1043.
4. Моршинин А.В. Об одной задаче кластеризации графа // Вестник Омского университета. 2018. Т 23, №1. С. 4-9.
5. Ильев В.П., Ильева С.Д., Моршинин А.В. Одна задача кластеризации с частичным обучением // VII Международная конференция «Проблемы оптимизации и их приложения (OPTA 2018)». Тезисы докладов конференции. Омск 2018. С. 85.
6. 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. Vol. 11548. P. 295–308.
7. Ильев В.П., Ильева С.Д., Моршинин А.В. Алгоритмы приближённого решения одной задачи кластеризации графа // Прикладная дискретная математика. 2019. № 45. P. 64-77.
8. 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. Vol. 1275. P. 23-29.
9. Ильев В.П., Ильева С.Д., Моршинин А.В. 2-приближённые алгоритмы для двух задач кластеризации на графах // Дискретный анализ и исследование операций. 2020. V. 27. № 3. P. 88-108.
10. Моршинин А.В. Метод ветвей и границ для задач кластеризации вершин графа // Четвертая всероссийская научно-практическая конференция «Омские научные чтения». Материалы конференции. Омск 2020.

Моршинин Александр Владимирович

# ТОЧНОЕ И ПРИБЛИЖЕННОЕ 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 2013. Формат 60×84 1/16.
Печ. л. 1,0. Уч.-изд. л. 1,0. Тираж 100 экз. Заказ № .

---

Издательство ОмГУ
644077, г. Омск, пр. Мира, 55а