

УДК 519.8

**АЛГОРИТМЫ ПРИБЛИЖЁННОГО РЕШЕНИЯ  
ОДНОЙ ЗАДАЧИ КЛАСТЕРИЗАЦИИ ГРАФА**

В. П. Ильев\*, С. Д. Ильева\*, А. В. Моршинин\*\*

*\* Омский государственный университет им. Ф. М. Достоевского, г. Омск, Россия,**\*\* Институт математики им. С. Л. Соболева СО РАН, г. Омск, Россия*

Изучается задача кластеризации графа. Для варианта задачи, в котором число кластеров не превосходит 3, разработаны три приближённых алгоритма. Первый алгоритм использует в качестве процедуры известный алгоритм Коулмана — Саундерсона — Вирта, который приближённо решает аналогичную задачу с числом кластеров, не превосходящим 2, и многократно применяет локальный поиск. Второй алгоритм основан на оригинальной идее и вообще не использует локальный поиск. В третьем алгоритме процедура локального поиска применяется к допустимому решению, возвращенному вторым алгоритмом. Получены априорные гарантированные оценки точности всех трёх алгоритмов, лучшая из которых равна  $(6 - 12/n)$ , где  $n$  — число вершин данного графа. Проведено также экспериментальное сравнение предложенных алгоритмов.

**Ключевые слова:** *граф, кластеризация, приближённый алгоритм, гарантированная оценка точности.*

DOI 10.17223/20710410/45/7

**APPROXIMATE ALGORITHMS FOR GRAPH CLUSTERING PROBLEM**

V. P. Il'ev\*, S. D. Il'eva\*, A. V. Morshinin\*\*

*\* Dostoevsky Omsk State University, Omsk, Russia,**\*\* Sobolev Institute of Mathematics SB RAS, Omsk, Russia***E-mail:** iljev@mail.ru

In the paper, the graph clustering problem is studied. For the version of the problem when the number of clusters does not exceed 3, we develop three approximate algorithms. The first algorithm uses as a procedure the known Coleman — Saunderson — Wirth algorithm which approximately solves the similar problem when the number of clusters does not exceed 2, and repeatedly applies a local search. On the contrary, our second algorithm is based on an original idea and does not use a local search at all. The main difference between these algorithms is the following. The first algorithm looks over all vertices of a given graph, for each vertex forms the cluster involving this vertex and all its neighbors and on the rest of the vertices forms one or two clusters using the Coleman — Saunderson — Wirth algorithm. The second algorithm looks over all ordered pairs of vertices of a given graph and for every pair forms two clusters at once, each of which contains only one vertex of this pair with some of its neighbors, placing the rest of the vertices to the third cluster (the third cluster may be empty). Finally, the third algorithm applies the local search only once to the feasible solution returned by the second one. A priori performance guarantees of all approximate algorithms are obtained, the best is equal to  $(6 - 12/n)$  for the second and the third

algorithms, where  $n$  is the number of vertices of a given graph. Also, experimental comparing of the developed algorithms was carried out. Experimental testing show that running time of our second and third algorithms is essentially less than running time of the first algorithm. At the same time the third algorithm demonstrated the best results in sense of accuracy of the solutions. Thus, the third algorithm has the best characteristics both from point of view of theoretical analysis and experimental study.

**Keywords:** *graph, clustering, approximation algorithm, performance guarantee.*

### Введение

В задаче кластеризации требуется разбить заданное множество объектов на несколько подмножеств (кластеров) на основе сходства объектов друг с другом. Одной из наиболее наглядных формализаций этой задачи является *задача кластеризации графа* [1, 2]. В ней объектам взаимно однозначно сопоставлены вершины графа, а рёбра соединяют вершины, соответствующие похожим объектам.

Будем рассматривать только *обыкновенные графы*, т. е. графы без петель и кратных рёбер. Обыкновенный граф называется *кластерным графом* [3], если каждая его компонента связности является полным графом. Пусть  $\mathcal{M}(V)$  — множество всех кластерных графов на множестве вершин  $V$ ;  $\mathcal{M}_k(V)$  — множество всех кластерных графов на  $V$ , имеющих ровно  $k$  непустых компонент связности;  $\mathcal{M}_{\leq k}(V)$  — множество всех кластерных графов на  $V$ , имеющих не более  $k$  компонент связности,  $2 \leq k \leq |V|$ .

Если  $G_1 = (V, E_1)$  и  $G_2 = (V, E_2)$  — 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 рёбер в графах  $G_1$  и  $G_2$ .

В литературе в основном рассматривались следующие три варианта задачи кластеризации 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 GC<sub>k</sub>.** Дан произвольный граф  $G = (V, E)$  и целое число  $k$ ,  $2 \leq k \leq |V|$ . Найти такой граф  $M^* \in \mathcal{M}_k(V)$ , что

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

**Задача GC<sub>≤k</sub>.** Дан произвольный граф  $G = (V, E)$  и целое число  $k$ ,  $2 \leq k \leq |V|$ . Найти такой граф  $M^* \in \mathcal{M}_{\leq k}(V)$ , что

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

В последние годы задачи кластеризации графов неоднократно переоткрывались и независимо изучались под разными названиями (Correlation Clustering [4], Cluster Editing [3, 5] и др.). В этих и других работах рассматривались также более общие постановки задач.

Вычислительная сложность задач кластеризации графов долгое время оставалась неизвестной. В 1986 г. М. Крживанек и Дж. Моравек [6] доказали, что задача  $\mathbf{GC}$  является NP-трудной, однако их работа осталась незамеченной. В 2004 г. в [4] и независимо в [3] доказана NP-трудность задачи  $\mathbf{GC}$ . В [4] доказано также, что задача  $\mathbf{GC}_k$  является NP-трудной при любом фиксированном  $k \geq 2$ ; в 2006 г. в [7] опубликовано более простое доказательство этого результата. В том же году в [8] доказано, что задачи  $\mathbf{GC}_2$  и  $\mathbf{GC}_{\leq 2}$  NP-трудны уже на кубических графах, откуда сделан вывод, что все упомянутые ранее задачи кластеризации графа являются NP-трудными, включая и задачу  $\mathbf{GC}_{\leq k}$ .

В [4] приведён 3-приближённый алгоритм для задачи  $\mathbf{GC}_{\leq 2}$ , в [8] доказано существование рандомизированной полиномиальной приближённой схемы для задачи  $\mathbf{GC}_{\leq 2}$ , а в [7] предложена рандомизированная полиномиальная приближённая схема для задачи  $\mathbf{GC}_k$  (для любого фиксированного  $k \geq 2$ ), сложность которой лишает её перспективы практического применения. В [9] представлен 2-приближённый алгоритм решения задачи  $\mathbf{GC}_{\leq 2}$ , основанный на применении процедуры локального поиска к каждому допустимому решению, полученному с помощью 3-приближённого алгоритма из [4]. Для задачи  $\mathbf{GC}_2$  в [10] предложен  $(3 - 6/|V|)$ -приближённый алгоритм.

Что касается задачи  $\mathbf{GC}$ , то в [11] показано, что задача  $\mathbf{GC}$  является APX-трудной, и разработан 4-приближённый алгоритм её решения. В [12] предложен 2,5-приближённый алгоритм для задачи  $\mathbf{GC}$ , в [13] —  $(2,06 - \varepsilon)$ -приближённый алгоритм.

Более подробный обзор известных результатов по задачам кластеризации графов можно найти в [14].

В п. 1 настоящей работы приводится краткий обзор известных результатов для задачи кластеризации графа, в которой число кластеров не превосходит 2. В п. 2 рассматривается задача  $\mathbf{GC}_{\leq 3}$ , в которой число кластеров не превосходит 3. Для этой задачи предложены два полиномиальных приближённых алгоритма. Первый 6-приближённый алгоритм использует в качестве процедуры известный алгоритм Коулмана — Саундерсона — Вирта, который приближённо решает аналогичную задачу с числом кластеров, не превосходящим 2, и многократно применяет локальный поиск. Второй алгоритм основан на оригинальной идее и не использует локальный поиск. Доказаны априорные гарантированные оценки точности обоих алгоритмов, лучшая из которых, у второго алгоритма, равна  $6 - 12/n$ , где  $n$  — число вершин графа. В п. 3 предложен ещё один полиномиальный алгоритм приближённого решения задачи  $\mathbf{GC}_{\leq 3}$ , в котором процедура локального поиска применяется к допустимому решению, возвращённому вторым алгоритмом. Приведены также результаты экспериментального исследования всех трёх разработанных алгоритмов.

### 1. Приближённые алгоритмы для задачи $\mathbf{GC}_{\leq 2}$

Через  $N_G(v)$  обозначим окрестность вершины  $v$ , т. е. множество вершин графа  $G = (V, E)$ , смежных с вершиной  $v$ . Через  $\bar{N}_G(v)$  обозначим множество вершин графа  $G$ , не смежных с  $v$ :  $\bar{N}_G(v) = V \setminus (N_G(v) \cup \{v\})$ .

Пусть  $G_1 = (V, E_1)$  и  $G_2 = (V, E_2)$  — два графа на множестве вершин  $V$ ,  $n = |V|$ . Через  $D(G_1, G_2)$  обозначим граф на множестве вершин  $V$  с множеством рёбер  $E_1 \Delta E_2$ . Заметим, что  $\rho(G_1, G_2)$  — это количество рёбер в графе  $D(G_1, G_2)$ .

Следующее утверждение легко доказывается с помощью леммы о рукопожатиях.

**Лемма 1** [10]. Пусть  $d_{\min}$  — минимум степеней вершин в графе  $D(G_1, G_2)$ . Тогда

$$\rho(G_1, G_2) \geq nd_{\min}/2.$$

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

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  $(v, u) \in E$ ;  $A_v^-$  — число таких вершин  $u \in A$ , что  $(v, u) \notin E$ .

Будем говорить, что кластерный граф  $M(\bar{V}_1, \dots, \bar{V}_s)$  получен из графа  $M(V_1, \dots, V_s)$  путём переноса вершины  $v \in V_i$  из кластера  $V_i$  в кластер  $V_j$ , если  $\bar{V}_i = V_i \setminus \{v\}$ ,  $\bar{V}_j = V_j \cup \{v\}$  и  $\bar{V}_k = V_k$  для всех  $k \notin \{i, j\}$ .

В 2004 г. разработан 3-приближённый алгоритм 1 решения задачи  $\mathbf{GC}_{\leq 2}$  [4].

---

### Алгоритм 1. ВВС

---

**Вход:** граф  $G = (V, E)$ .

**Выход:** кластерный граф  $M \in \mathcal{M}_{\leq 2}(V)$ .

- 1: Для каждой вершины  $v \in V$  определить кластерный граф  $M_v = M(V_1, V_2)$  следующим образом:  $V_1 = \{v\} \cup N_G(v)$ ,  $V_2 = \bar{N}_G(v)$  (возможно  $V_2 = \emptyset$ ).
  - 2: Среди всех графов  $M_v$  выбрать такой граф  $M$ , что  $\rho(G, M) = \min_{v \in V} \rho(G, M_v)$ .
- 

Справедлива следующая гарантированная оценка точности алгоритма **ВВС**.

**Теорема 1** [4]. Для любого графа  $G = (V, E)$  имеет место неравенство

$$\rho(G, M) \leq 3\rho(G, M^*),$$

где  $M^* \in \mathcal{M}_{\leq 2}(V)$  — оптимальное решение задачи  $\mathbf{GC}_{\leq 2}$  на графе  $G$ ;  $M$  — кластерный граф, построенный алгоритмом **ВВС**.

В 2008 г. Т. Коулман, Дж. Саундерсон и А. Вирт [9] предложили 2-приближённый алгоритм для задачи  $\mathbf{GC}_{\leq 2}$ , применив процедуру локального поиска **LS** к каждому допустимому решению, полученному на шаге 1 алгоритма **ВВС** (алгоритм 2).

---

### Алгоритм 2. Процедура LS( $M, X, Y$ )

---

**Вход:** кластерный граф  $M = M(X, Y) \in \mathcal{M}_{\leq 2}(V)$ .

**Выход:** кластерный граф  $\bar{M} = M(\bar{X}, \bar{Y}) \in \mathcal{M}_{\leq 2}(V)$ .

- 1: Для каждой вершины  $v \in X$  вычислить  $\delta(v) = X_v^- - X_v^+ + Y_v^+ - Y_v^-$ . Выбрать такую вершину  $v_0 \in X$ , что  $\delta(v_0) = \max_{v \in X} \delta(v)$ .
  - 2: Для каждой вершины  $u \in Y$  вычислить  $\delta(u) = Y_u^- - Y_u^+ + X_u^+ - X_u^-$ . Выбрать такую вершину  $u_0 \in Y$ , что  $\delta(u_0) = \max_{u \in Y} \delta(u)$ .
  - 3: **Если**  $\delta(v_0) \leq 0$  и  $\delta(u_0) \leq 0$ , **то стоп**; положить  $\bar{X} := X$  и  $\bar{Y} := Y$ , **иначе** переход на шаг 4.
  - 4: **Если**  $\delta(v_0) \geq \delta(u_0)$ , **то** положить  $X := X \setminus \{v_0\}$ ,  $Y := Y \cup \{v_0\}$ .
  - 5: **Если**  $\delta(v_0) < \delta(u_0)$ , **то** положить  $X := X \cup \{u_0\}$ ,  $Y := Y \setminus \{u_0\}$ . Переход на шаг 1.
- 

С учётом **ВВС** и процедуры локального поиска **LS** 2-приближённый алгоритм Коулмана, Саундерсона и Вирта примет следующий вид (алгоритм 3).

**Алгоритм 3. CSW****Вход:** граф  $G = (V, E)$ .**Выход:** кластерный граф  $M \in \mathcal{M}_{\leq 2}(V)$ .

- 1: Для каждой вершины  $v \in V$ :
- 2: определить кластерный граф  $M_v = M(V_1, V_2)$  следующим образом:

$$V_1 = \{v\} \cup N_G(v), \quad V_2 = \overline{N}_G(v);$$

- 3: выполнить процедуру  $\mathbf{LS}(M_v, V_1, V_2)$ . Построенный граф обозначить  $\overline{M}_v$ .
- 4: Среди всех графов  $\overline{M}_v$  выбрать такой граф  $M$ , что  $\rho(G, M) = \min_{v \in V} \rho(G, \overline{M}_v)$ .

Справедлива следующая гарантированная оценка точности алгоритма **CSW**.

**Теорема 2** [9]. Для любого графа  $G = (V, E)$  имеет место неравенство

$$\rho(G, M) \leq 2\rho(G, M^*),$$

где  $M^* \in \mathcal{M}_{\leq 2}(V)$  — оптимальное решение задачи  $\mathbf{GC}_{\leq 2}$  на графе  $G$ ;  $M$  — кластерный граф, построенный алгоритмом **CSW**.

**2. Приближённое решение задачи  $\mathbf{GC}_{\leq 3}$** **2.1. 6-Приближённый алгоритм для задачи  $\mathbf{GC}_{\leq 3}$** 

Используя идеи [4, 9], можно построить полиномиальный алгоритм 4 приближённого решения задачи  $\mathbf{GC}_{\leq 3}$ .

**Алгоритм 4.  $\mathbf{A}_1$** **Вход:** граф  $G = (V, E)$ ,  $|V| = n$ .**Выход:** кластерный граф  $M_1 \in \mathcal{M}_{\leq 3}(V)$ .

- 1: Если  $n \leq 2$ , то  $M_1 := G$ , иначе переход на шаг 2.
- 2: Для каждой вершины  $w \in V$ :
- 3:  $V_1 := \{w\} \cup N_G(w)$ . Если  $V_1 = V$ , то  $M_w$  — полный граф  $K_n$ , иначе переход на шаг 4.
- 4: Обозначить  $G_1$  подграф графа  $G$ , порождённый множеством  $V \setminus V_1$ . Приближённо решить задачу  $\mathbf{GC}_{\leq 2}$  на графе  $G_1$  алгоритмом **CSW**, полученный кластерный граф обозначить  $M = M(V_2, V_3)$  (возможно,  $V_3 = \emptyset$ ). Положить  $M_w := M(V_1, V_2, V_3)$ .
- 5: Среди всех графов  $M_w$  выбрать ближайший к  $G$  кластерный граф  $M_1$ :

$$\rho(G, M_1) = \min_{w \in V} \rho(G, M_w).$$

**Замечание 1.** Трудоёмкость алгоритма  $\mathbf{A}_1$  равна  $O(n^6)$ .

Справедлива следующая гарантированная оценка точности алгоритма  $\mathbf{A}_1$ .

**Теорема 3.** Для любого графа  $G = (V, E)$  имеет место неравенство

$$\rho(G, M_1) \leq 6\rho(G, M^*),$$

где  $M^* \in \mathcal{M}_{\leq 3}(V)$  — оптимальное решение задачи  $\mathbf{GC}_{\leq 3}$  на графе  $G$ ;  $M_1$  — кластерный граф, построенный алгоритмом  $\mathbf{A}_1$ .

**Доказательство.** Пусть  $v$  — вершина минимальной степени  $d_{\min}$  в графе  $D = D(G, M^*)$ . Рассмотрим кластерный граф  $\overline{M} \in \mathcal{M}_{\leq 3}(V)$ , полученный из  $M^*$  путём переноса  $d_{\min}$  вершин в другие компоненты графа  $M^*$ : вершины множества  $N_G(v) \cap N_D(v)$  переместим в компоненту, содержащую вершину  $v$ , а вершины, принадлежащие множеству  $\overline{N}_G(v) \cap N_D(v)$ , перенесём в любую из компонент, не содержащих  $v$ . Тогда кластер  $V_1 = \{v\} \cup N_G(v)$  — один и тот же в  $\overline{M}$  и в  $M_v$ .

Если  $d_{\min} = 0$ , то кластерный граф  $\overline{M}$  совпадает с кластерным графом  $M^*$ , а значит,

$$\rho(G, \overline{M}) = \rho(G, M^*) \leq 3\rho(G, M^*).$$

Пусть  $d_{\min} > 0$ . Заметим, что при переносе  $d_{\min}$  вершин значение целевой функции не может увеличиться более чем на  $nd_{\min}$ , так как перенос одной вершины увеличивает значение целевой функции не более чем на  $n = |V|$ . Отсюда с учётом леммы 1 получаем

$$\rho(G, \overline{M}) \leq \rho(G, M^*) + nd_{\min} \leq \rho(G, M^*) + 2\rho(G, M^*) = 3\rho(G, M^*).$$

Итак,

$$\rho(G, \overline{M}) \leq 3\rho(G, M^*) \quad (1)$$

Докажем, что  $\rho(G, M_v) \leq 6\rho(G, M^*)$ . Рассмотрим кластер  $V_1 = \{v\} \cup N_G(v)$ . Возможны два случая:

а)  $V_1 = V$ . В соответствии с шагом 3 алгоритма **A<sub>1</sub>** полагаем  $M_v := K_n = \overline{M}$ . Следовательно,

$$\rho(G, M_v) = \rho(G, \overline{M}) \leq 3\rho(G, M^*) \leq 6\rho(G, M^*).$$

б)  $V_1 \neq V$ . Обозначим через  $M_1^*$  оптимальное решение задачи **GC<sub>≤2</sub>** на графе  $G_1$ , порождённом множеством  $V \setminus V_1$ . Рассмотрим кластерный граф  $\widetilde{M} = M(V_1) \cup M_1^*$ , где  $M(V_1)$  — полный граф на множестве  $V_1$ . Очевидно, что

$$\rho(G, \widetilde{M}) \leq \rho(G, \overline{M}).$$

Тогда с учётом (1)

$$\rho(G, \widetilde{M}) \leq 3\rho(G, M^*). \quad (2)$$

Пусть  $M = M(V_2, V_3)$  — допустимое решение задачи **GC<sub>≤2</sub>** на графе  $G_1$ , найденное алгоритмом **CSW** на шаге 4, если на шаге 3 выбрана вершина  $v$ . Тогда  $M_v = M(V_1, V_2, V_3)$  (множество  $V_3$  может быть пустым).

Обозначим через  $s$  сумму числа отсутствующих рёбер в подграфе графа  $G$ , порождённом множеством вершин  $V_1 = \{v\} \cup N_G(v)$ , и величины разреза  $(V_1, V \setminus V_1)$  в графе  $G$ . Очевидно, что

$$\rho(G, M_v) = s + \rho(G_1, M), \quad \rho(G, \widetilde{M}) = s + \rho(G_1, M_1^*). \quad (3)$$

Если  $|V \setminus V_1| \leq 2$ , то  $M = G_1$  и  $\rho(G_1, M) = \rho(G_1, M_1^*) = 0$ . Тогда с учётом (2) и (3)

$$\rho(G, M_v) = \rho(G, \widetilde{M}) \leq 3\rho(G, M^*) \leq 6\rho(G, M^*).$$

Пусть  $|V \setminus V_1| \geq 3$ . Тогда по теореме 2

$$\rho(G_1, M) \leq 2\rho(G_1, M_1^*).$$

В силу (2) и (3) получим

$$\begin{aligned}\rho(G, M_v) &= s + \rho(G_1, M) \leq s + 2\rho(G_1, M_1^*) \leq 2s + 2\rho(G_1, M_1^*) = \\ &= 2(s + \rho(G_1, M_1^*)) = 2\rho(G, \widetilde{M}) \leq 6\rho(G, M^*).\end{aligned}$$

На шаге 5 алгоритма  $\mathbf{A}_1$  среди всех графов будет рассмотрен граф  $M_v$ , где  $v$  — вершина минимальной степени в графе  $D = D(G, M^*)$ . Отсюда получается требуемая оценка точности алгоритма  $\mathbf{A}_1$ . ■

В п. 2.2 представлен ещё один полиномиальный алгоритм приближённого решения задачи  $\mathbf{GC}_{\leq 3}$  с лучшей гарантированной оценкой точности, основанный на другой идее и не использующий локальный поиск.

### 2.2. Приближённый алгоритм для задачи $\mathbf{GC}_{\leq 3}$ , не использующий локальный поиск

**Лемма 2.** Пусть  $G = (V, E)$  —  $n$ -вершинный граф,  $n \geq 3$ ,  $M = M(V_1, \dots, V_s)$ , вершина  $v \in V_i$  такова, что её окрестность в графе  $D = D(G, M)$  не пуста. Рассмотрим кластерный граф  $M' \in \mathcal{M}_{\leq s}(V)$ , полученный из графа  $M$  следующим образом: либо произвольная вершина  $u \in N_G(v) \cap N_D(v)$  перенесена в кластер  $V_i$ , либо произвольная вершина  $u \in \overline{N}_G(v) \cap N_D(v)$  перенесена в любой из кластеров  $V_j$ , не содержащий вершину  $v$ . Тогда

$$\rho(G, M') \leq \rho(G, M) + (n - 3).$$

*Доказательство.* Пусть  $u \in N_D(v)$ . Очевидно, что графы  $D$  и  $D' = D(G, M')$  отличаются только рёбрами вида  $(u, w)$ ,  $w \in V$ , остальные рёбра у них совпадают. Следовательно,

$$\rho(G, M') - \rho(G, M) = |N_{D'}(u)| - |N_D(u)|.$$

Так как  $v \in N_D(u)$ , а  $N_{D'}(u) \subseteq V \setminus \{u, v\}$ , то  $|N_D(u)| \geq 1$ ,  $|N_{D'}(u)| \leq n - 2$ , откуда

$$\rho(G, M') - \rho(G, M) \leq n - 2 - 1 = n - 3.$$

Лемма 2 доказана. ■

Пусть  $M^* = M(V_1^*, V_2^*, V_3^*) \in \mathcal{M}_{\leq 3}(V)$  — оптимальное решение задачи  $\mathbf{GC}_{\leq 3}$  на графе  $G = (V, E)$ , причём  $M^* \neq K_n$ , где  $n = |V|$ . Через  $d_D(v)$  обозначим степень вершины  $v$  в графе  $D = D(G, M^*)$ .

Положим  $d_1 = \min\{d_D(v) : v \in V\} = d_D(v_1)$ ; обозначим  $V_1^*$  тот кластер графа  $M^*$ , который содержит вершину  $v_1$ ,  $n_1 = |V_1^*|$ . Пусть  $d_2 = \min\{d_D(v) : v \in V \setminus V_1^*\} = d_D(v_2)$ ,  $V_2^*$  — тот кластер графа  $M^*$ , который содержит вершину  $v_2$ . Очевидно, что  $d_1 \leq d_2$ .

Так как  $\rho(G, M^*)$  равно числу рёбер в графе  $D(G, M^*)$ , из леммы о рукопожатиях и леммы 1 вытекает

**Лемма 3.** Для произвольного  $n$ -вершинного графа  $G = (V, E)$  имеет место неравенство

$$\rho(G, M^*) \geq \frac{d_1 n_1 + d_2 (n - n_1)}{2} \geq \frac{nd_1}{2},$$

где  $M^*$  — оптимальное решение задачи  $\mathbf{GC}_{\leq 3}$  на графе  $G$ .

Алгоритм 5 основан на идее, отличной от той, что использована в [4, 9] для приближённого решения задачи  $\mathbf{GC}_{\leq 2}$ .

**Замечание 2.** Трудоемкость алгоритма  $\mathbf{A}_2$  равна  $O(n^3)$ .

**Алгоритм 5. А<sub>2</sub>**

**Вход:** граф  $G = (V, E)$ ,  $n = |V|$ ,  $n \geq 3$ .

**Выход:** кластерный граф  $M_2 \in \mathcal{M}_{\leq 3}(V)$ .

- 1: Для каждой упорядоченной пары вершин  $(u, v) \in V \times V$ , такой, что  $u \neq v$ , выполнить:
- 2: положить  $V_1 = \{u\} \cup (N_G(u) \setminus \{v\})$ ;
- 3: обозначить через  $G_1$  подграф графа  $G$ , порождённый 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  $M_{uv} = M(V_1, V_2, V_3)$ .
- 4: Среди построенных графов  $M_{uv}$  и графа  $K_n$  выбрать ближайший к  $G$  кластерный граф  $M_2 \in \mathcal{M}_{\leq 3}(V)$ :

$$\rho(G, M_2) = \min_{(u,v) \in V \times V} \{\rho(G, M_{uv}), \rho(G, K_n)\},$$

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

Справедлива следующая гарантированная оценка точности алгоритма **А<sub>2</sub>**.

**Теорема 4.** При  $n \geq 3$  для любого  $n$ -вершинного графа  $G = (V, E)$  имеет место неравенство

$$\rho(G, M_2) \leq \left(6 - \frac{12}{n}\right) \rho(G, M^*),$$

где  $M^* \in \mathcal{M}_{\leq 3}(V)$  — оптимальное решение задачи **GC<sub>≤3</sub>** на графе  $G$ ;  $M_2$  — кластерный граф, построенный алгоритмом **А<sub>2</sub>**.

*Доказательство.* Возможны два случая.

**С л у ч а й 1:**  $M^* = K_n$ . Тогда на шаге 2 алгоритма **А<sub>2</sub>** в качестве  $M_2$  будет выбран граф  $K_n$ , следовательно,

$$\rho(G, M_2) = \rho(G, M^*) \leq \left(6 - \frac{12}{n}\right) \rho(G, M^*).$$

**С л у ч а й 2:**  $M^* = M(V_1^*, V_2^*, V_3^*) \neq K_n$ . Пусть, как и ранее,  $v_1, v_2 \in V$  — такие вершины, что  $d_D(v_1) = d_1$ ,  $d_D(v_2) = d_2$ , причём  $v_1 \in V_1^*$ ,  $v_2 \in V_2^*$ . Рассмотрим кластерный граф  $M \in \mathcal{M}_{\leq 3}(V)$ , полученный из графа  $M^*$  путём переноса не более чем  $d_1$  вершин в другие кластеры графа  $M^*$ : вершины множества  $(N_G(v_1) \setminus \{v_2\}) \cap N_D(v_1)$  перенесены в кластер, содержащий вершину  $v_1$  (т.е. в кластер  $V_1^*$ ), а вершины множества  $\overline{N}_G(v_1) \cap N_D(v_1)$  — в кластер, не содержащий ни  $v_1$ , ни  $v_2$  (т.е. в кластер  $V_3^*$ ). Пусть  $M = M(V_1, V_2, V_3)$ , где  $V_1 = \{v_1\} \cup (N_G(v_1) \setminus \{v_2\})$  и  $V_2$  — кластер, содержащий вершину  $v_2$ .

Если  $d_1 > 0$ , то, применив не более чем  $d_1$  раз лемму 2, получим

$$\rho(G, M) \leq \rho(G, M^*) + d_1(n - 3).$$

Если  $d_1 = 0$ , то граф  $M$  совпадает с графом  $M^*$ , а значит,

$$\rho(G, M) = \rho(G, M^*) = \rho(G, M^*) + d_1(n - 3).$$

Итак, показано, что в любом случае

$$\rho(G, M) \leq \rho(G, M^*) + d_1(n - 3). \quad (4)$$

Заметим, что  $V_1 \neq V$ , поскольку  $v_2 \notin V_1$ . Пусть  $G_1$  — подграф графа  $G$ , порождённый множеством вершин  $V \setminus V_1$ . Рассмотрим кластерный граф  $M_{v_2} \in \mathcal{M}_{\leq 2}(V \setminus V_1)$ , построенный следующим образом: вершина  $v_2$  и все смежные с ней вершины графа  $G_1$  принадлежат одному кластеру графа  $M_{v_2}$ , а все несмежные с  $v_2$  вершины — другому кластеру графа  $M_{v_2}$  (этот кластер может быть пустым).

Пусть  $M_1 \in \mathcal{M}_{\leq 2}(V \setminus V_1)$  — подграф графа  $M$ , порождённый множеством вершин  $V \setminus V_1$ , т. е.  $M_1 = M(V_2, V_3)$ . Положим  $D_1 = D(G_1, M_1)$ . Обозначим через  $\bar{d}_2$  степень вершины  $v_2$  в графе  $D_1$ . Очевидно, что кластерный граф  $M_{v_2}$  получен из графа  $M_1$  путём переноса  $\bar{d}_2$  вершин: вершины множества  $N_{G_1}(v_2) \cap N_{D_1}(v_2)$  перенесены в кластер, содержащий вершину  $v_2$  (т. е. в компоненту, порождённую множеством  $V_2$ ), а все вершины множества  $\bar{N}_{G_1}(v_2) \cap N_{D_1}(v_2)$  — в кластер, не содержащий  $v_2$  (т. е. в компоненту, порождённую множеством  $V_3$ ).

Если  $|V \setminus V_1| \leq 2$ , то, очевидно,  $M_{v_2} = G_1$  и  $\rho(G_1, M_{v_2}) = 0$ . Если при этом  $\bar{d}_2 = 0$  то  $M_{v_2} = M_1$ . Таким образом,

$$\rho(G_1, M_{v_2}) = \rho(G_1, M_1) = \rho(G_1, M_1) + \bar{d}_2(|V \setminus V_1| - 3).$$

Пусть теперь  $\bar{d}_2 > 0$ . Это возможно только при  $|V \setminus V_1| = 2$ . В этом случае  $\bar{d}_2 = 1$ . Это означает, что  $\rho(G_1, M_1) = 1$ . Тогда

$$\rho(G_1, M_{v_2}) = 0 = \rho(G_1, M_1) - 1 = \rho(G_1, M_1) + \bar{d}_2(|V \setminus V_1| - 3).$$

Теперь рассмотрим случай  $|V \setminus V_1| \geq 3$ . Если  $\bar{d}_2 > 0$  то, применив  $\bar{d}_2$  раз лемму 2, получим

$$\rho(G_1, M_{v_2}) \leq \rho(G_1, M_1) + \bar{d}_2(|V \setminus V_1| - 3).$$

Если  $\bar{d}_2 = 0$ , то графы  $M_{v_2}$  и  $M_1$  совпадают, следовательно,

$$\rho(G_1, M_{v_2}) = \rho(G_1, M_1) = \rho(G_1, M_1) + \bar{d}_2(|V \setminus V_1| - 3).$$

Итак, показано, что в любом случае

$$\rho(G_1, M_{v_2}) \leq \rho(G_1, M_1) + \bar{d}_2(|V \setminus V_1| - 3).$$

Так как граф  $M$  получен из графа  $M^*$  путём переноса не более чем  $d_1$  вершин в другие кластеры, то  $|V_1| \geq |V_1^*| - d_1 = n_1 - d_1$ , откуда

$$\rho(G_1, M_{v_2}) \leq \rho(G_1, M_1) + \bar{d}_2(n - n_1 + d_1 - 3). \quad (5)$$

Покажем, что  $\bar{d}_2 \leq d_2$ . В самом деле,  $d_2 = |N_D(v_2)|$ ,  $\bar{d}_2 = |N_{D_1}(v_2)|$ . Множество  $N_D(v_2)$  состоит из всех вершин  $u$ , таких, что либо  $u \in N_G(v_2) \cap (V_1^* \cup V_3^*)$ , либо  $u \in \bar{N}_G(v_2) \cap V_2^*$ , а множество  $N_{D_1}(v_2)$  — из всех вершин  $u$ , таких, что либо  $u \in N_{G_1}(v_2) \cap V_3$ , либо  $u \in \bar{N}_{G_1}(v_2) \cap V_2$ , т. е.

$$\begin{aligned} N_D(v_2) &= (N_G(v_2) \cap (V_1^* \cup V_3^*)) \cup (\bar{N}_G(v_2) \cap V_2^*), \\ N_{D_1}(v_2) &= (N_{G_1}(v_2) \cap V_3) \cup (\bar{N}_{G_1}(v_2) \cap V_2). \end{aligned}$$

Очевидно, что  $V_3 \subseteq V_1^* \cup V_3^*$  и  $V_2 \subseteq V_2^*$ , а так как  $G_1$  — подграф графа  $G$ , то  $N_{G_1}(v_2) \subseteq N_G(v_2)$  и  $\bar{N}_{G_1}(v_2) \subseteq \bar{N}_G(v_2)$ . Следовательно,

$$\begin{aligned} N_{G_1}(v_2) \cap V_3 &\subseteq N_G(v_2) \cap (V_1^* \cup V_3^*), \\ \bar{N}_{G_1}(v_2) \cap V_2 &\subseteq \bar{N}_G(v_2) \cap V_2^*. \end{aligned}$$

Значит,  $N_{\widetilde{D}_1}(v_2) \subseteq N_D(v_2)$ , откуда получаем  $\widetilde{d}_2 = |N_{\widetilde{D}_1}(v_2)| \leq |N_D(v_2)| = d_2$ .

Итак,  $\widetilde{d}_2 \leq d_2$ . Тогда из (5) следует, что

$$\rho(G_1, M_{v_2}) \leq \rho(G_1, M_1) + d_2(n - n_1 + d_1 - 3). \quad (6)$$

В соответствии с шагом 4 алгоритма **A**<sub>2</sub> положим  $M_{v_1v_2} = M(V_1) \cup M_{v_2}$ , где  $M(V_1)$  — полный граф на множестве  $V_1$ . Обозначим через  $s$  сумму числа отсутствующих рёбер в подграфе графа  $G$ , порождённом множеством вершин  $V_1$ , и величины разреза  $(V_1, V \setminus V_1)$  в графе  $G$ . Очевидно, что

$$\rho(G, M_{v_1v_2}) = s + \rho(G_1, M_{v_2}), \quad \rho(G, M) = s + \rho(G_1, M_1).$$

Отсюда с учётом (4) и (6) имеем

$$\begin{aligned} \rho(G, M_{v_1v_2}) &= s + \rho(G_1, M_{v_2}) \leq s + \rho(G_1, M_1) + d_2(n - n_1 + d_1 - 3) = \\ &= \rho(G, M) + d_2(n - n_1 + d_1 - 3) \leq \rho(G, M^*) + d_1(n - 3) + d_2(n - n_1 + d_1 - 3), \end{aligned}$$

т. е.

$$\rho(G, M_{v_1v_2}) \leq \rho(G, M^*) + d_1(n - 3) + d_2(n - n_1 + d_1 - 3). \quad (7)$$

Теперь покажем, что  $d_2 \leq n_1 + n/2$ . Действительно,

$$d_2 = |N_D(v_2)| = |N_D(v_2) \cap V_1^*| + |N_D(v_2) \cap (V_2^* \cup V_3^*)|.$$

Очевидно, что  $|N_D(v_2) \cap V_1^*| \leq |V_1^*| = n_1$ . Обозначим  $d' = |N_D(v_2) \cap (V_2^* \cup V_3^*)|$  и докажем, что  $d' \leq n/2$ . Предположим противное:  $d' > n/2$ . Рассмотрим кластерный граф  $\widetilde{M}$ , полученный из  $M^*$  путём переноса вершины  $v_2$  из кластера  $V_2^*$  в кластер  $V_3^*$ . Очевидно, что  $V_2 \cup V_3 = V_2^* \cup V_3^*$  и графы  $D$  и  $\widetilde{D} = D(G, \widetilde{M})$  отличаются только рёбрами вида  $(u, v_2)$ , где  $u \in V_2^* \cup V_3^*$ , остальные рёбра у них совпадают. Следовательно,

$$\rho(G, \widetilde{M}) - \rho(G, M^*) = |N_{\widetilde{D}}(v_2) \cap (V_2^* \cup V_3^*)| - |N_D(v_2) \cap (V_2^* \cup V_3^*)|.$$

Так как

$$N_{\widetilde{D}}(v_2) \cap (V_2^* \cup V_3^*) = (V_2^* \cup V_3^*) \setminus (\{v_2\} \cup N_D(v_2)) = (V_2^* \cup V_3^*) \setminus ((\{v_2\} \cup N_D(v_2)) \cap (V_2^* \cup V_3^*)),$$

получим

$$|N_{\widetilde{D}}(v_2) \cap (V_2^* \cup V_3^*)| - |N_D(v_2) \cap (V_2^* \cup V_3^*)| = (n - n_1 - d' - 1) - d' = n - n_1 - 2d' - 1.$$

По предположению,  $d' > n/2$ , поэтому  $\rho(G, \widetilde{M}) - \rho(G, M^*) < -n_1 - 1 < 0$ , что противоречит оптимальности кластерного графа  $M^*$ . Следовательно,  $d' \leq n/2$ , откуда

$$d_2 \leq n_1 + n/2. \quad (8)$$

Оценим слагаемые из правой части неравенства (7) с учётом (8), леммы 3 и неравенства  $d_1 \leq d_2$ .

а) Дадим оценку суммы  $\rho(G, M^*) + d_1(n - 3)$ :

$$\begin{aligned} \rho(G, M^*) + d_1(n - 3) &= \rho(G, M^*) + d_1n(1 - 3/n) \leq \\ &\leq \rho(G, M^*) + 2\rho(G, M^*)(1 - 3/n) = (3 - 6/n)\rho(G, M^*). \end{aligned}$$

б) Оценим слагаемое  $d_2(n - n_1 + d_1 - 3)$ :

$$\begin{aligned} d_2(n - n_1 + d_1 - 3) &= d_1n_1 - d_1n_1 + d_2(n - n_1) + d_1d_2 - 3d_2 \leq \\ &\leq 2\rho(G, M^*) - d_1n_1 + d_1d_2 - 3d_2 \leq 2\rho(G, M^*) - d_1n_1 + d_1(n_1 + n/2) - 3d_2 = \\ &= 2\rho(G, M^*) + d_1n/2 - 3d_2 \leq 2\rho(G, M^*) + d_1n/2 - 3d_1 = \\ &= 2\rho(G, M^*) + d_1n/2(1 - 6/n) \leq 2\rho(G, M^*) + \rho(G, M^*)(1 - 6/n) = (3 - 6/n)\rho(G, M^*). \end{aligned}$$

Итак,

$$\rho(G, M_{v_1v_2}) \leq (6 - 12/n)\rho(G, M^*).$$

На шаге 4 алгоритмом  $\mathbf{A}_2$  среди всех графов  $M_{uv}$  будет, в частности, рассмотрен и граф  $M_{v_1v_2}$ . Отсюда получается требуемая оценка точности алгоритма  $\mathbf{A}_2$ . ■

### 3. Вычислительный эксперимент

Для сравнения точности приближённых алгоритмов  $\mathbf{A}_1$  и  $\mathbf{A}_2$  проведён вычислительный эксперимент. Исследован также алгоритм  $\mathbf{A}_3$ , являющийся модификацией алгоритма  $\mathbf{A}_2$  (в нём к приближённому решению, полученному алгоритмом  $\mathbf{A}_2$ , применяется процедура локального поиска). Очевидно, что алгоритм  $\mathbf{A}_3$  полиномиален и его гарантированная оценка точности не хуже, чем гарантированная оценка точности алгоритма  $\mathbf{A}_2$ . Все алгоритмы реализованы на языке программирования C++ в среде JetBrains Clion 2017.3.3. Вычисления проводились на персональном компьютере модели RT I5N с процессором Intel Core i5-2400, тактовой частотой 3,10 ГГц и объёмом оперативной памяти 8 Гбайт.

Опишем класс графов, на которых производились вычисления. Зафиксируем параметр  $p \in (0, 1)$ , случайный  $n$ -вершинный граф  $G = (V, E)$  будем получать с использованием следующей процедуры. Для каждой пары вершин  $(u, v)$  проводится независимый случайный эксперимент, исходом которого является наличие ребра  $uv$  с вероятностью  $p$  и отсутствие ребра с вероятностью  $1 - p$ . В [15, 16] семейство  $n$ -вершинных графов с таким распределением обозначается  $G(n, p)$  и используется как при теоретическом изучении графов, так и в экспериментальных исследованиях (модель Эрдеша — Реньи).

Параметр  $p$  представляет собой математическое ожидание плотности случайного графа  $G = (V, E)$ , которая определяется как  $2|E|/(n(n - 1))$ . В проведённых экспериментах использовались значения  $p \in \{0,33, 0,5, 0,67\}$ .

На графах малой размерности (при  $n$  от 10 до 20 вершин) алгоритмом полного перебора были найдены точные решения. Это позволило получить предварительные сведения об изменении точности алгоритмов в зависимости от параметров  $n$  и  $p$ . Для каждой пары значений  $n$  и  $p$  было решено по 50 задач.

Далее *точностью (отклонением от оптимума)* приближённого алгоритма будем называть отношение значения целевой функции, полученного данным алгоритмом, к её оптимальному значению. Обозначим через  $\delta_1(n, p)$ ,  $\delta_2(n, p)$  и  $\delta_3(n, p)$  средние точности алгоритмов  $\mathbf{A}_1$ ,  $\mathbf{A}_2$  и  $\mathbf{A}_3$  для  $n$ -вершинных графов при фиксированном значении параметра  $p$ .

По итогам первого эксперимента наилучшие показатели принадлежат алгоритму  $\mathbf{A}_3$ : его среднее отклонение от оптимума достигает максимума при значениях параметров  $n = 18$ ,  $p = 0,33$  и составляет 3,6%, в то время как алгоритмы  $\mathbf{A}_1$  и  $\mathbf{A}_2$  при тех же параметрах отклоняются на 7,3 и 12,3% соответственно. При всех остальных значениях параметров  $n$  и  $p$  алгоритм  $\mathbf{A}_3$  также даёт решения с наименьшим средним отклонением от оптимума среди исследуемых алгоритмов (рис. 1).

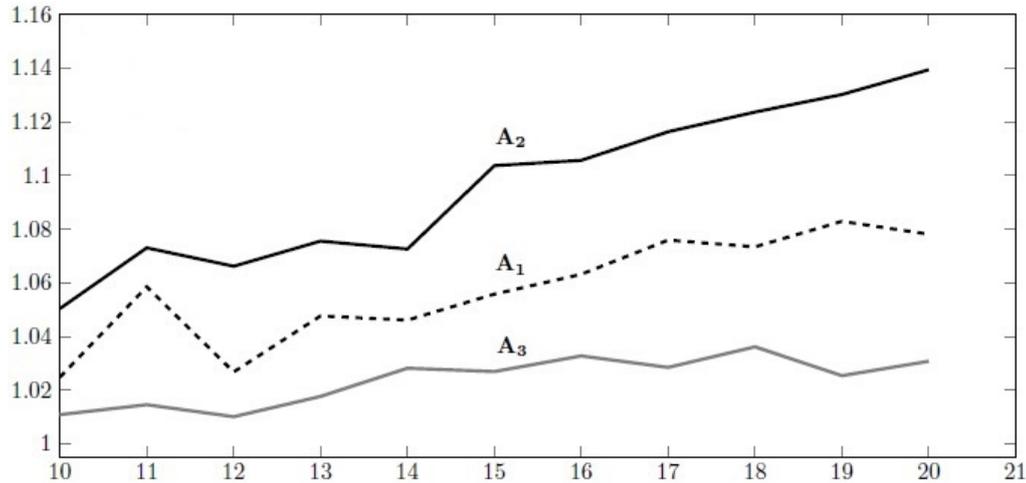


Рис. 1. Средняя точность приближённых алгоритмов при  $p = 0,33$

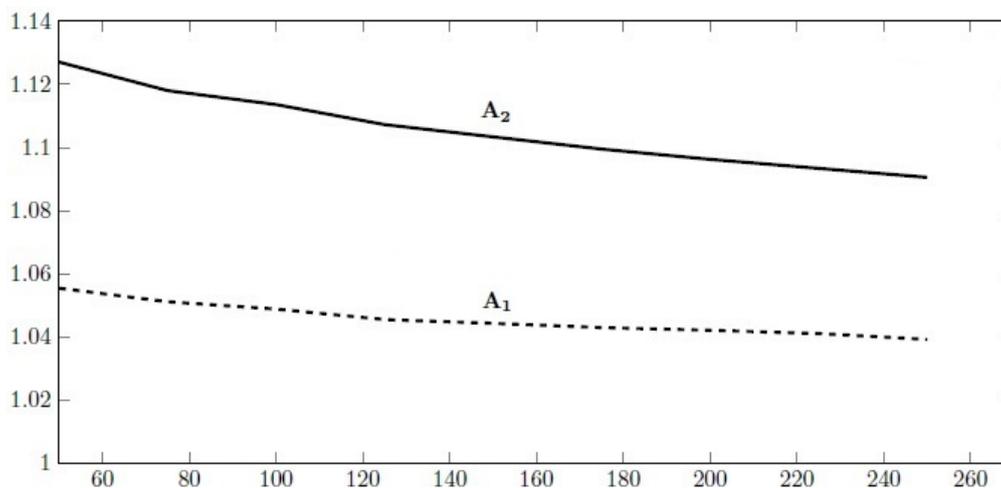
Таким образом, наиболее перспективным приближённым алгоритмом можно считать алгоритм **A<sub>3</sub>**. Для подтверждения того, что он продолжает доминировать и при других значениях  $n$ , был проведён эксперимент на графах большей размерности (при  $n$  от 50 до 250 вершин). Однако в этом случае нет возможности найти оптимальное решение за приемлемое время, поэтому в качестве исследуемых были выбраны следующие величины:

$$d_1(n, p) = \frac{\delta_1(n, p)}{\delta_3(n, p)}, \quad d_2(n, p) = \frac{\delta_2(n, p)}{\delta_3(n, p)}.$$

Выбор таких величин объясняется тем, что, во-первых, их вычисление не требует знания оптимального решения, а во-вторых, они позволяют ответить на вопрос, действительно ли алгоритм **A<sub>3</sub>** даёт в среднем лучшие решения, чем алгоритмы **A<sub>1</sub>** и **A<sub>2</sub>**, на графах большей размерности. Для каждой пары значений  $n$  и  $p$  было решено по 100 задач.

При всех значениях  $n$  и  $p$  величины  $d_1(n, p)$  и  $d_2(n, p)$  оказались больше единицы. При этом в каждой решённой задаче алгоритм **A<sub>3</sub>** даёт значение целевой функции не большее, чем значения целевых функций, полученных алгоритмами **A<sub>1</sub>** и **A<sub>2</sub>** (рис. 2). Это позволяет сделать вывод о том, что алгоритм **A<sub>3</sub>** продолжает доминировать над алгоритмами **A<sub>1</sub>** и **A<sub>2</sub>** независимо от значений параметров  $n$  и  $p$ . При этом время работы алгоритмов **A<sub>2</sub>** и **A<sub>3</sub>** с ростом  $n$  увеличивается достаточно медленно (меньше 1 с при  $n = 50$  и около 10 с при  $n = 250$ ). Для алгоритма **A<sub>1</sub>** рост количества вершин оказал существенное влияние на время работы (меньше 1 с при  $n = 50$  и около 800 с при  $n = 250$ ).

Из результатов вычислительного эксперимента можно сделать вывод, что наилучшим алгоритмом приближенного решения задачи **GC<sub>≤3</sub>** является **A<sub>3</sub>**: решения, полученные этим алгоритмом, наиболее близки к оптимальным, при этом время его работы остаётся сравнительно небольшим при росте количества вершин.

Рис. 2. Значения величин  $d_1(n, p)$  и  $d_2(n, p)$  при  $p = 0,33$ 

### Заключение

В работе рассматривается задача кластеризации графа, в которой число кластеров не превосходит трёх. Предложены три полиномиальных алгоритма приближённого решения этой задачи. Первый, 6-приближённый алгоритм  $\mathbf{A}_1$ , многократно использует локальный поиск и основан на идеях, применяемых для решения задачи, в которой число кластеров не превышает двух. Вторым алгоритм  $\mathbf{A}_2$  основан на оригинальной идее и не использует локальный поиск. Приближённый алгоритм  $\mathbf{A}_3$  является модификацией второго и отличается тем, что к приближённому решению, найденному алгоритмом  $\mathbf{A}_2$ , применяется процедура локального поиска. Доказано, что алгоритмы  $\mathbf{A}_2$  и  $\mathbf{A}_3$  имеют гарантированную оценку точности  $6 - 12/n$ , где  $n$  — число вершин кластеризуемого графа.

### ЛИТЕРАТУРА

1. *Kulis B., Basu S., Dhillon I., and Mooney R.* Semi-supervised graph clustering: a kernel approach // *Machine Learning*. 2009. V. 74. No. 1. P. 1–22.
2. *Schaeffer S. E.* Graph clustering // *Computer Science Review*. 2005. V. 1. No. 1. P. 27–64.
3. *Shamir R., Sharan R., and Tsur D.* Cluster graph modification problems // *Discr. Appl. Math.* 2004. V. 144. No. 1-2. P. 173–182.
4. *Bansal N., Blum A., and Chawla S.* Correlation clustering // *Machine Learning*. 2004. V. 56. P. 89–113.
5. *Ben-Dor A., Shamir R., and Yakhimi Z.* Clustering gene expression patterns // *J. Comput. Biol.* 1999. V. 6. No. 3-4. P. 281–297.
6. *Křivánek M. and Morávek J.* NP-hard problems in hierarchical-tree clustering // *Acta Informatica*. 1986. V. 23. P. 311–323.
7. *Giotis I. and Guruswami V.* Correlation clustering with a fixed number of clusters // *Theory of Computing*. 2006. V. 2. No. 1. P. 249–266.
8. *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. № 1. С. 3–11.
9. *Coleman T., Saunderson J., and Wirth A.* A local-search 2-approximation for 2-correlation-clustering // *LNCS*. 2008. V. 5193. P. 308–319.
10. *Ильев В. П., Ильева С. Д., Навроцкая А. А.* Приближенные алгоритмы для задач аппроксимации графов // *Дискрет. анализ и исслед. операций*. 2011. Т. 18. № 1. С. 41–60.

11. Charikar M., Guruswami V., and Wirth A. Clustering with qualitative information // J. Comput. Syst. Sci. 2005. V. 71. No. 3. P. 360–383.
12. Ailon N., Charikar M., and Newman A. Aggregating inconsistent information: Ranking and clustering // J. ACM. 2008. V. 55. No. 5. P. 1–27.
13. Chawla S., Makarychev K., Schramm T., and Yaroslavtsev G. Near optimal LP algorithm for correlation clustering on complete and complete  $k$ -partite graphs // STOC'15 Symp. on Theory of Computing. N.Y.: ACM, 2015. P. 219–228.
14. Il'ev V., Il'eva S., and Kononov A. Short survey on graph correlation clustering with minimization criteria // LNCS. 2016. V. 9869. P. 25–36.
15. Эрдеш П., Спенсер Дж. Вероятностные методы в комбинаторике. М.: Мир, 1976.
16. Alon N. and Spencer J. The Probabilistic Method. N.Y.: Wiley and Sons, 1992.

## REFERENCES

1. Kulis B., Basu S., Dhillon I., and Mooney R. Semi-supervised graph clustering: a kernel approach. Machine Learning, 2009, vol. 74, no. 1, pp. 1–22.
2. Schaeffer S. E. Graph clustering. Computer Science Review, 2005, vol. 1, no. 1, pp. 27–64.
3. Shamir R., Sharan R., and Tsur D. Cluster graph modification problems. Discr. Appl. Math., 2004, vol. 144, no. 1-2, pp. 173–182.
4. Bansal N., Blum A., and Chawla S. Correlation clustering. Machine Learning, 2004, vol. 56, pp. 89–113.
5. Ben-Dor A., Shamir R., and Yakhimi Z. Clustering gene expression patterns. J. Comput. Biol., 1999, vol. 6, no. 3-4, pp. 281–297.
6. Křivánek M. and Morávek J. NP-hard problems in hierarchical-tree clustering. Acta Informatica, 1986, vol. 23, pp. 311–323.
7. Giotis I. and Guruswami V. Correlation clustering with a fixed number of clusters. Theory of Computing, 2006, vol. 2, no. 1, pp. 249–266.
8. Ageev A. A., Il'ev V. P., Kononov A. V., and Talevnin A. S. Computational complexity of the graph approximation problem. J. Appl. Indust. Math., 2007, vol. 1, no. 1, pp. 1–8.
9. Coleman T., Saunderson J., and Wirth A. A local-search 2-approximation for 2-correlation-clustering. LNCS, 2008, vol. 5193, pp. 308–319.
10. Il'ev V. P., Il'eva S. D., and Navrotskaya A. A. Approximation algorithms for graph approximation problems. J. Appl. Indust. Math., 2011, vol. 5, no. 4, pp. 569–581.
11. Charikar M., Guruswami V., and Wirth A. Clustering with qualitative information. J. Comput. Syst. Sci., 2005, vol. 71, no. 3, pp. 360–383.
12. Ailon N., Charikar M., and Newman A. Aggregating inconsistent information: Ranking and clustering. J. ACM, 2008, vol. 55, no. 5, pp. 1–27.
13. Chawla S., Makarychev K., Schramm T., and Yaroslavtsev G. Near optimal LP algorithm for correlation clustering on complete and complete  $k$ -partite graphs. STOC'15 Symp. on Theory of Computing, New York, ACM, 2015, pp. 219–228.
14. Il'ev V., Il'eva S., and Kononov A. Short survey on graph correlation clustering with minimization criteria. LNCS, 2016, vol. 9869, pp. 25–36.
15. Erdős P. and Spencer J. Probabilistic Methods in Combinatorics. Budapest, Académiai Kiadó, 1974.
16. Alon N. and Spencer J. The Probabilistic Method. New York, Wiley and Sons, 1992.