УДК 004.421
DOI: 10.25559/SITITO.15.201901.45-51

Алгоритмы сравнительного исследования двух инвариантов графа

Б. Ф. Мельников[1*], Н. П. Чурикова[2]
[1] Российский государственный социальный университет, г. Москва, Россия
[*] bf-melnikov@yandex.ru
[2] Самарский национальный исследовательский университет имени академика С.П. Королева, г. Самара, Россия

Аннотация

Алгоритмы определения изоморфности графов имеют широкое применение в различных областях, таких как: задачи синтаксического и структурного распознавания образов, задачи математической химии, исследование социальных сетей и Интернета, а также соответствующих математических моделей. Кроме того, эти алгоритмы имеют большое значение в теории графов: например, они используются как вспомогательные при вычислении древесной ширины графов.

В статье рассмотрены основные существующие подходы к решению проблемы изоморфизма, как точные, так и эвристические. На настоящий момент еще не построен алгоритм, позволяющий решить эту задачу за полиномиальное время, но и доказательств невозможности построения такого алгоритма нет.

Работа посвящена исследованию простых связных графов и некоторых быстро вычисляемых инвариантов, а именно: индекса Рандича и вектора степеней второго порядка. Описывается алгоритм создания базы данных, формат данных, программная реализация вычислений и полученные результаты. Приведены результаты сравнения дифференцирующей способности этих инвариантов. Вычисления проводились для всех связных графов с количеством вершин 8, 9 и 10.

В качестве программных средств для реализации используются система управления базами данных MongoDB и язык программирования Python 2.7, для создания базы данных использовалась программа nauty, позволяющая получить набор, состоящий из всех связных неизоморфных графов с заданным количеством вершин.

Ключевые слова: изоморфизм графов, инвариант графа, эвристический алгоритм.

Для цитирования: *Мельников Б. Ф., Чурикова Н. П.* Алгоритмы сравнительного исследования двух инвариантов графа // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 1. С. 45-51. DOI: 10.25559/SITITO.15.201901.45-51

Algorithms of Comparative Analysis of Two Invariants of a Graph

B. F. Melnikov[1*], N. P. Churikova[2]
[1] Russian State Social University, Moscow, Russia
[*] bf-melnikov@yandex.ru
[2] Samara National Research University, Samara, Russia

Abstract

Algorithms of definition of isomorphism of graphs have wide application in various areas, such as: problems of syntactic and structural recognition of images, problems of mathematical chemistry, research of social networks and the Internet, and also corresponding mathematical models. In addition, these algorithms are of great importance in graph theory: for example, they are used as auxiliary ones in calculating the wood width of graphs.
In this paper, we consider the main existing approaches to solving the problem of isomorphism, both full and heuristic ones. At present, there is no algorithm that allows solving this problem in polynomial time, but there is no evidence of the impossibility of constructing such an algorithm.
The work is devoted to the study of simple connected graphs and some quickly calculated invariants, namely: the Randić Index and the second-order degree vector. The algorithm of database creation, data format, programmatic implementation of calculations and obtained results are described. The results of comparison of differentiation ability of these invariants are given. Calculations were performed for all connected graphs with the number of vertices 8, 9, and 10.
The MongoDB database management system and the Python 3.6 programming language are used as software tools for implementation, and a nauty program was used to create the database, which allows you to get a set consisting of all connected Non-isomorphic graphs with a specified number of vertices.
The obtained results allow us to speak about the possibility of their application in the tasks of equivalent conversion of non-deterministic finite machines, construction of effective algorithms for determining the wood width of the graph and some others.

Keywords: Graph Isomorphism, Graph Invariant, Heuristic Algorithm.

For citation: Melnikov B.F., Churikova N.P. Algorithms of Comparative Analysis of Two Invariants of a Graph. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(1):45-51. DOI: 10.25559/SITITO.15.201901.45-51

Введение и мотивация

Проблема изоморфизма графов, всё ещё нерешённая, остаётся основополагающей теоретической задачей с неопределённым статусом сложности в теории P/NP. В ней в качестве входа обычно рассматриваются два неориентированных графа, не имеющих весов для вершин и рёбер. Отношением изоморфизма между такими графами называется биекция между множествами вершин графов, сохраняющая смежность вершин. (При определении изоморфности для ориентированных или взвешенных графов накладываются специальные дополнительные ограничения.)

Распознавание изоморфности графов применяется в качестве вспомогательного алгоритма в самых разных областях: в задачах синтаксического и структурного распознавания образов [1], в задачах математической химии и хемоинформатики (т.е. при исследовании молекулярных структур химических соединений), в задачах, связанных с исследованием социальных сетей [2], в математических моделях самого Интернета [3 ,4].

Но, по-видимому, ещё более важную роль эти алгоритмы играют во «внутренних» задачах самой теории графов. Например, подобные алгоритмы очень важны как вспомогательные при вычислении значений древесной ширины графа [10]. Из последних публикаций, в которых упоминается изоморфизм графов, отметим такие разделы этой теории и соответствующие им статьи: фундаментальные проблемы теории графов [11, 12]; раскраска графов [13]; поиск специальных объектов (обычно — подграфов) в исследуемых графах[14, 15, 16].

Перейдём к описанию подходов к решению проблемы изоморфизма графов — как точных подходов, так и эвристических. Задача проверки изоморфизма графов принадлежит к классу задач, для которых пока неизвестно, являются ли они полиномиально разрешимыми или нет, т.е.:

- с одной стороны, пока не построен полиномиальный алгоритм проверки изоморфизма любых графов;
- но, с другой стороны, не получено и доказательство несуществования такого алгоритма.

Недавно, в 2015 году, Л. Бабай анонсировал алгоритм, позволяющий решить задачу изоморфизма за квазиполиномиальное время ($exp\left(loglog\left(n\right)^{O(1)}\right)$, где n — число вершин графа). В 2017 году этот алгоритм был уточнён [17]. Однако, корректность алгоритма до сих пор не доказана и вопрос о квазиполиномиальности задачи распознавания изоморфизма графов остается открытым. Непосредственная проверка на изоморфизм двух n-вершинных графов заключается в рассмотрении всех n! перестановок вершин и установлении совмещаемости всех рёбер графов хотя бы при одной перестановке. Понятно, что такой подход, даже при относительно небольшом количестве вершин, требует огромных вычислительных ресурсов.

Пример одного из возможных алгоритмов распознования изоморфности двух ориентированных графов описан ещё в 1975 г. — см. [18]; причём, по-видимому, имеются и более ранние публикации на эту тему. В этом алгоритме при помощи матрицы расстояний графа (в такой матрице каждый элемент является длиной кратчайшего пути между двумя вершинами) ограничивается дерево поиска возможных соответствий между вершинами при определении изоморфности двух ориентированных графов.

Изоморфизм можно рассматривать как перенумерацию вершин графа, поэтому любая количественная характеристика структуры графа остается неизменной. Такие характеристики называются инвариантами графа [19]. Среди множества инвариантов графа выделяются полные инварианты: инвариант называются полным, если равенство его значений для двух разных графов означает изоморфность данных графов. Известные на данный момент полные инварианты (например, макси-код) являются трудновычислимыми и не позволяют эффективно решать задачу определения изоморфизма графов.

На практике применяются более простые процедуры, от которых ожидается хорошая работа в большинстве случаев. Например, используются эвристики для доказательства, что два графа не изоморфны [5].

Существует подход к исследованию графов на изоморфизм, в основе которого лежит построение канонического кода графа, не зависящего от порядка нумерации вершин [20, 21]. На основе данного подхода в 1981 году Б. Маккей разработал программу nauty [20], а затем программу traces [21]. Два графа изоморфны тогда и только тогда, когда совпадают их канонические коды, которым соответствует каноническая нумерация вершин. Для решения этой задачи в пакете nauty используется представление множества всех вершин графа в виде разбиения — упорядоченного набора непересекающихся ячеек (подмножеств вершин). Перебору всех перестановок вершин соответствует перебор и уточнение разбиений, в результате которых появляется каноническая нумерация вершин. Уточнение разбиений сопоставляется обходу по дереву разбиений (дереву поиска). Перебор разбиений при нахождении канонической перестановки существенно сокращается при использовании группы автоморфизмов графа. Составной частью nauty является генерация автоморфизмов графа, в результате действия которых появляются орбиты вершин — подмножества вершин, в которые переходят заданные вершины. С помощью nauty можно найти каноническую нумерацию, построить автоморфизмы в неориентированных и ориентированных графах (а также в раскрашенных графах) и провести исследование графов на изоморфизм [22].

Алгоритм

В настоящей работе термин «граф» означает связный неориентированный граф без петель и кратных рёбер.

В общем случае задача проверки двух графов на изоморфизм имеет вид: даны графы G_1 и G_2 с множеством вершин $V(G_1)$, $V(G_2)$ и множеством ребер $E(G_1)$ и $E(G_2)$. $|V(G_1)|=|V(G_2)|=n$, $|E(G_1)|=|E(G_2)|$. Необходимо определить, существует ли биективное отображение $\varphi : V_A \to V_B$, такое что,

$$\left(i, j\right) \in E\left(G_1\right) \Leftrightarrow \left(\varphi\left(i\right), \varphi\left(j\right)\right) \in E\left(G_2\right).$$

Для доказательства того, что два графа не изоморфны, используются различные эвристики, например используют различные простые инварианты, — при обнаружении пары графов с различными значениями инварианта, делается заключение о неизоморфности этих графов. В [7] был предложен один из вариантов проверки двух графов на неизоморфность

— а именно, эвристический алгоритм определения конкретной последовательности проверки инвариантов. Мы рассматриваем в работе только следующие два инварианта графов:

1. Индекс Рандича

$$\sum_{(v_i,v_j)\in V} \frac{1}{\sqrt{d(v_i)d(v_j)}},$$

где v_i и v_j — вершины, образующие ребро, $d(v_k)$ — степень вершины v_k.

2. Вектор степеней второго порядка — вектор, каждый элемент, которого представляет собой список степеней вершин, смежных с данной [23].

В ходе исследования была создана база данных графов, содержащая все простые связные графы на 8, 9 и 10 вершин; количество таких графов — 11117, 261080, 11716571 соответственно, см. [24]. Для хранения графов используется формат graph6 — тестовый формат для хранения неориентированных графов. Формат graph6 (g6) поддерживается программой nauty [20]. Для генерации базы графов использовалась программа nauty, с её помощью были созданы файлы содержащие все простые связные графы на заданное количество вершин в формате g6.

Программа для вычислений реализована на языке программирования Python 3.6, для хранения полученных результатов используется система управления базами данных MongoDB. Данная СУБД использует для хранения данных формат BSON, являющийся бинарным аналогом JSON, что позволяет эффективно описывать сложные структуры данных, быстрее выполнять операции поиска и обработки.

Функция вычисления инвариантов для графа в формате g6 и записи в базу данных на языке Python выглядит следующим образом:

```
defcomputing_gr(graph):
    g = nx.parse_graph6(graph)

# количество ребер
edges = len(nx.edges(g))

    # значение инварианта индекс Рандича
    # число с плавающей точкой, с точностью до
6
randic = invariants_tools.get_randic_index(g)

# вектор степеней второго порядка
    # отсортированный список
sec_vec = invariants_tools.get_sec_vec(g)

    # записьвбазуданных
mongo_tools.insert_base_item(posts, graph,
str(edges), str(randic),  str(sec_vec))
```

Так как вычисления для такого количества графов весьма затратны по времени и вычислительным ресурсам, программа была написана с использованием парадигмы параллельного программирования. Вычисления инвариантов для графов производились одновременно в 4 потока, полностью используя вычислительные ресурсы машины. Для каждого графа из базы были вычислены 2 вышеописанных инварианта. Обработка данных в 4 потока:

```
pool = Pool(4)
pool.map(computing_gr, read_input(g6f))
pool.close()
pool.join()
```

В итоге для каждого графа в базе данных хранится запись следующего вида:

```
{ "_id" : ObjectId("5be6be0ebc54f4673d590
0c0"), "sec_vec" : "[[2, 1, 1, 1, 1], [5, 2],
[2, 1], [5], [5], [5], [5], [2]]", "graph6" :
"G??EDw", "randic" : "3.3121889292032174", "date"
: ISODate("2018-11-10T11:16:30.618Z"), "nodes" :
"7" }
```

После создания базы данных, проведена работа по поиску пар графов, дифференцируемых индексом Рандича или вектором степеней второго порядка. Функция поиска данных графов:

```
#на вход функция получает все возможные уни-
кальные варианты пар графов
deffind_base_result(pair):
        ...
if (sec_vec_0 == sec_vec_1):
ifmath.isclose(randic_0, randic_1, abs_
tol=0.000001) is False:
if g6_0 != g6_1:
res_srt = "A: {0} B: {1} \n".format(g6_0, g6_1)
fl_randic_base.write(res_srt)
else:
ifmath.isclose(randic_0, randic_1, abs_
tol=0.0001) is True:
if g6_0 != g6_1:
res_srt = "A: {0} B: {1} \n".format(g6_0, g6_1)
fl_sec_vec_base.write(res_srt)
```

Рассмотрим алгоритм работы программы на примере графов с 8 вершинами.

Алгоритм 1. Схема алгоритма восстановления случайного графа по заданному вектору степеней.

Вход: список всех графов с 8 вершинами.

1. По очереди выбираются графы с заданным количеством рёбер (для графов с 8 вершинами — от 7 до 28 рёбер).

2. Для набора с заданным количество рёбер строится набор уникальных пар графов.

3. Для каждой пары из набора вызывается функция поиска заданных графов, find_base_result, описанная выше.

Выход: все пары графов проверены.

(Конец описания алгоритма)

Полученные результаты

В результате работы программы получим набор файлов. Имя файла включает в себя информацию о количестве рёбер, количестве вершин и название инварианта. То есть файл с названием 8_result_nodes_sec_vec_14.txt содержит в себе такие пары графов, для которых вектор степеней второго порядка отличается, а индекс Рандича совпадает.

В результате вычислений для простых связных графов с 8, 9 и 10 вершинами были найдены примеры графов [25], для которых совпадает индекс Рандича, но не совпадает вектор степеней второго порядка. Обратных примеров, когда совпадает вектор степеней второго порядка, но не совпадает индекс Рандича, найдено не было.

В качестве одного из этих примеров рассмотрим инварианты для двух графов. В следующих двух таблицах 1 и 2 матрицы приведены смежности этих графов. Важно отметить, что этот пример не был приведен в предыдущих работах авторов, он публикуется впервые.

Таблица 1.
Table 1.

0	0	0	1	0	1	1	1	1
0	0	0	0	1	0	1	1	1
0	0	0	0	0	0	1	1	1
1	0	0	0	0	1	0	1	1
0	1	0	0	0	0	1	1	1
1	0	1	1	0	0	1	1	1
1	1	1	0	1	1	0	1	1
1	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	0

Таблица 2.
Table 2.

0	0	0	0	1	1	1	1	1
0	0	0	0	1	0	1	1	1
0	0	0	0	0	1	1	1	1
0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	1	1
1	0	1	1	0	0	1	1	1
1	1	1	1	0	1	0	1	1
1	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	0

Для этих графов значения большинства рассматриваемых нами инвариантов совпадают — как и для примера, рассмотренного ранее в работе [9]:

Таблица 3.
Table 3.

	Граф на таблице 1	Граф на таблице 2
Число вершин	9	9
Число ребер	25	25
Вектор степеней	[5, 4, 4, 4, 4, 6, 7, 8, 8]	[5, 4, 4, 4, 4, 6, 7, 8, 8]
Диаметр	2	2
Индекс Рандича	4.366087889367704	4.366087889367704

Однако при этом векторы степеней второго порядка различаются; для первого графа –

[[8, 4, 6, 7, 8], [8, 4, 7, 8], [8, 6, 7, 8], [5, 8, 6, 8], [8, 4, 7, 8],
[5, 4, 4, 7, 8, 8], [5, 4, 4, 4, 6, 8, 8], [5, 4, 4, 4, 4, 6, 7, 8],
[5, 4, 4, 4, 4, 6, 7, 8] ,

а для второго графа –

[[8, 4, 6, 7, 8], [8, 4, 7, 8], [8, 6, 7, 8], [8, 6, 7, 8], [5, 4, 8, 8],
[5, 4, 4, 7, 8, 8], [5, 4, 4, 4, 6, 8, 8], [5, 4, 4, 4, 4, 6, 7, 8],
[5, 4, 4, 4, 4, 6, 7, 8]].

(мы для наглядности тоже привели вектора и их элементы согласно применённому нами порядку вершин). Однако очевидно, что эти вектора при упорядочивании не совпадут — что проще всего показывается путём упорядочивания вектора первого графа: получается вектор [8, 8, 6, 5], не имеющий во втором графе инварианта.

Заключение

В качестве возможных направлений дальнейших работ отметим следующие.

- Строгое доказательство невозможности «превосходства» индекса Рандича над вектором степеней второго порядка — ни для какого графа.
- Описание и исследование математической модели, предназначенной для описания качества инвариантов и их последовательностей.
- Описание нескольких вариантов применения исследования изоморфизма графов в задачах эквивалентного преобразования недетерминированных конечных автоматов, см. [26,27] и др.
- Описание применения простых инвариантов графа в задаче построения эффективных алгоритмов определения древесной ширины графа.
- Обобщение рассмотренных здесь инвариантов на орграфы, прежде всего — на турниры.

Список использованных источников

[1] *Poddar A., Sahidullah M., Saha G.* Speaker verification with short utterances: a review of challenges, trends and opportunities // IET Biometrics. 2018. Vol. 7, Issue 2. Pp. 91-101. DOI: 10.1049/iet-bmt.2017.0065

[2] *Zhang X., Moore C., Newman M. E. J.* Random graph models for dynamic networks // The European Physical Journal B. 2017. Vol. 90, Issue 10. Article: 200. DOI: 10.1140/epjb/e2017-80122-8

[3] *Raigorodskii A.M.* Small subgraphs in preferential attachment networks // Optimization Letters. 2017. Vol. 11, Issue 2. Pp. 249-257. DOI: 10.1007/s11590-015-0945-9

[4] Algorithms and Models for the Web Graph. Proceedings of the 15th International Workshop (WAW 2018), Moscow, Russia, May 17-18, 2018 / A. Bonato, P. Pralat, A.M. Raigorodskii (eds.). Vol. 10836. Springer International Publishing, 2018. 185 p. DOI: 10.1007/978-3-319-92871-5

[5] *Мельников Б.Ф., Сайфуллина Е.Ф.* Применение мультиэвристического подхода для случайной генерации графов с заданным вектором степеней // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. 2013. № 3(27). С. 70-83. URL: https://izvuz_fmn.pnzgu.ru/sf6313 (дата обращения: 18.12.2018).

[6] *Babai L., Erdos P., Selkow S.M.* Random Graph Isomorphism // SIAM Journal on Computing. 1980. Vol. 9, Issue 3. Pp. 628-635. DOI: 10.1137/0209047

[7] *Мельникова Е.А., Сайфуллина Е.Ф.* Подход к проверке изоморфизма графов с помощью построения инвариантов // Вектор науки Тольяттинского государственного университета. 2013. № 1(23). С. 113-120. URL: https://elibrary.ru/item.asp?id=20394979 (дата обращения: 18.12.2018).

[8] *Melnikov B.F., Melnikova E.A., Pivneva S.V., Dudnikov V.A., Davydova E.V.* Geometric and game approaches for some discrete optimization problems // CEUR Workshop Proceedings. 2018. Vol. 2212. Pp. 312-321. URL: http://ceur-ws.org/Vol-2212/paper42.pdf (дата обращения: 18.12.2018).

[9] *Мельников Б.Ф., Сайфуллина Е.Ф., Терентьева Ю.Ю., Чурикова Н.П.* Применение алгоритмов генерации случайных графов для исследования надёжности сетей связи // Информатизация и связь. 2018. № 1. С. 71-80. URL: https://elibrary.ru/item.asp?id=32651307 (дата обращения: 18.12.2018).

[10] *Chalermsook P., Das S., Even G., Laekhanukit B., Vaz D.* Survivable Network Design for Group Connectivity in Low-Treewidth Graphs // arXiv:1802.10403 [cs.DS]. URL: https://arxiv.org/abs/1802.10403 (дата обращения: 18.12.2018).

[11] *Chellali M., Haynes T.W., Hedetniemi S.T., Lewis T.M.* On *ve*-degrees and *ev*-degrees in graphs // Discrete Mathematics. 2017. Vol. 340, Issue 2. Pp. 31-38. DOI: 10.1016/j.disc.2016.07.008

[12] Graph Theory — Favorite Conjectures and Open Problems — 1 / R. Gera, S. Hedetniemi, C. Larson (eds.) // Problem Books in Mathematics. Springer, Cham, 2016. 291 p. DOI: 10.1007/978-3-319-31940-7

[13] *Chartrand G., English S., Zhang P.* Kaleidoscopic colorings of graphs // Discussiones Mathematicae Graph Theory. 2017. Vol. 37, Issue 3. Pp. 711-727. DOI: doi.org/10.7151/dmgt.1950

[14] *Desormeaux W.J., Haynes T.W., Hedetniemi S.T., Moore C.* Distribution centers in graphs // Discrete Applied Mathematics. 2018. Vol. 243. Pp. 186-193. DOI: 10.1016/j.dam.2018.02.009

[15] *Andrews E., Chartrand G., Lumduanhom C., Zhang P.* Stars and Their k-Ramsey Numbers // Graphs and Combinatorics. 2017. Vol. 33, Issue 2. Pp. 257-274. DOI: 10.1007/s00373-017-1756-9

[16] *Ahangar H.A., Fujie-Okamoto F., Samodivkin V.* On the forcing connected geodetic number and the connected geodetic number of a graph // Ars Combinatoria. 2016. Vol. 126. Pp. 323-335.

[17] *Babai L.* Graph Isomorphism in Quasipolynomial Time // arXiv:1512.03547 [cs.DS]. URL: https://arxiv.org/abs/1512.03547 (дата обращения: 18.12.2018).

[18] *Schmidt D.C., Druffel L.E.* A Fast Backtracking Algorithm to Test Directed Graphs for Isomorphism Using Distance Matrices // Journal of the ACM. 1976. Vol. 23, Issue 3. Pp. 433-445. DOI: 10.1145/321958.321963

[19] *Харари Ф.* Теория графов. М.: Ленанд, 2015. 304 с.

[20] *McKay B.D.* Practical graph isomorphism // Congressus Numerantium. 1981. Vol. 30. Pp. 45-87. URL: http://users.cecs.anu.edu.au/~bdm/nauty/pgi.pdf (дата обращения: 18.12.2018).

[21] *McKay B.D., Piperno A.* Practical graph isomorphism, II // Journal of Symbolic Computation. 2014. Vol. 60. Pp. 94-112. DOI: 10.1016/j.jsc.2013.09.003

[22] *Володичева М.И., Леора С.Н.* Исследование изоморфизма графов с помощью жордановых форм матриц смежности // Прикладная дискретная математика. 2018. № 40. С. 87-99. DOI: 10.17223/20710410/40/7

[23] *Мельников Б.Ф., Сайфуллина Е.Ф.* Генерация графов с заданным вектором степеней второго порядка и задача проверки изоморфизма // Стохастическая оптимизация в информатике. 2014. Т. 10, № 2. С. 24-36. URL: https://elibrary.ru/item.asp?id=22764354 (дата обращения: 18.12.2018).

[24] *Harary F., Palmer E.M.* Graphical Enumeration. New York; London; Oxford; Bosto; San Diego: Academic Press, 2014. 286 p.

[25] *Чурикова Н.П.* О дифференциации графов на основе быстро вычисляемых инвариантов // XII Белорусская математическая конференция: материалы Междунар. науч. конф. Минск, 5-10 сентября 2016 г. / Под ред. С.Г. Красовского. Часть 4. Мн.: Институт математики НАН Беларуси, 2016. С. 67-69.

[26] *Мельников Б.Ф., Сайфуллина М.Р.* О некоторых алгоритмах эквивалентного преобразования недетерминированных конечных автоматов // Известия высших учебных заведений. Математика. 2009. № 4. С. 67-72. URL: https://kpfu.ru/portal/docs/F1684373104/10_04ref.pdf (дата обращения: 18.12.2018).

[27] *Melnikov B.F., Melnikova A.A.* Edge-minimization of non-deterministic finite automata // Korean Journal of Computational and Applied Mathematics. 2001. Vol. 8, Issue 3. Pp. 469-479. DOI: 10.1007/BF02941980

Поступила 18.12.2018; принята к публикации 20.02.2019; опубликована онлайн 19.04.2019.

Об авторах:

Мельников Борис Феликсович, профессор, кафедра информационных систем, сетей и безопасности, факультет информационных технологий, Российский государственный социальный университет (129226, Россия, г. Москва, ул. Вильгельма Пика, д. 4, стр. 1), доктор физико-математических наук, ORCID: http://orcid.org/0000-0002-6765-6800, bf-melnikov@yandex.ru

Чурикова Надежда Петровна, аспирант, кафедра информатики и вычислительной математики, Самарский национальный исследовательский университет имени академика С.П. Королева (Самарский университет) (443086, Россия, г. Самара, ул. Московское шоссе, д. 34), ORCID: http://orcid.org/0000-0002-4082-5838, claorisel@gmail.com

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

[1] Poddar A., Sahidullah M., Saha G. Speaker verification with short utterances: a review of challenges, trends and opportunities. *IET Biometrics*. 2018; 7(2):91-101. (In Eng.) DOI: 10.1049/iet-bmt.2017.0065

[2] Zhang X., Moore C., Newman M. E. J. Random graph models for dynamic networks. *The European Physical Journal B.* 2017; 90(10):200. (In Eng.) DOI: 10.1140/epjb/e2017-80122-8

[3] Raigorodskii A.M. Small subgraphs in preferential attachment networks. *Optimization Letters*. 2017; 11(2):249-257.

(In Eng.) DOI: 10.1007/s11590-015-0945-9

[4] Bonato A., Pralat P., Raigorodskii A.M. (eds.) Algorithms and Models for the Web Graph. *Proceedings of the 15th International Workshop (WAW 2018)*, Moscow, Russia, May 17-18, 2018. Vol. 10836. Springer International Publishing, 2018. (In Eng.) DOI: 10.1007/978-3-319-92871-5

[5] Melnikov B.F., Sayfullina E.F. Applying multiheuristic approach o randomly generating graphs with a given degree sequence. *University proceedings. Volga region. Physical and mathematical sciences. Mathematics*. 2013; 3(27):70-83. Available at: https://izvuz_fmn.pnzgu.ru/sf6313 (accessed 18.12.2018). (In Russ.)

[6] Babai L., Erdos P., Selkow S.M. Random Graph Isomorphism. *SIAM Journal on Computing*. 1980; 9(3):628-635. (In Eng.) DOI: 10.1137/0209047

[7] Melnikova E.A., Sayfullina E.F. Approach to the verification of graph isomorphism by constructing invariants. *Vector science Togliatti State University*. 2013; 1(23):113-120. Available at: https://elibrary.ru/item.asp?id=20394979 (accessed 18.12.2018). (In Russ.)

[8] Melnikov B.F., Melnikova E.A., Pivneva S.V., Dudnikov V.A., Davydova E.V. Geometric and game approaches for some discrete optimization problems. *CEUR Workshop Proceedings*. 2018; 2212:312-321. Available at: http://ceur-ws.org/Vol-2212/paper42.pdf (accessed 18.12.2018). (In Eng.)

[9] Melnikov B.F., Sayfullina E.F., Terentyeva Y.Y., Churikova N.P. Application of algorithms for generating random graphs for investigating the reliability of communication networks. *Informatization and communication*. 2018; 1:71-80. Available at: https://elibrary.ru/item.asp?id=32651307 (accessed 18.12.2018). (In Russ.)

[10] Chalermsook P., Das S., Even G., Laekhanukit B., Vaz D. Survivable Network Design for Group Connectivity in Low-Treewidth Graphs. *arXiv*:1802.10403 [cs.DS]. Available at: https://arxiv.org/abs/1802.10403 (accessed 18.12.2018). (In Eng.)

[11] Chellali M., Haynes T.W., Hedetniemi S.T., Lewis T.M. On ve-degrees and ev-degrees in graphs. *Discrete Mathematics*. 2017; 340(2):31-38. (In Eng.) DOI: 10.1016/j.disc.2016.07.008

[12] Gera R., Hedetniemi S., Larson C. (eds.) Graph Theory — Favorite Conjectures and Open Problems — 1. *Problem Books in Mathematics*. Springer, Cham, 2016. (In Eng.) DOI: 10.1007/978-3-319-31940-7

[13] Chartrand G., English S., Zhang P. Kaleidoscopic colorings of graphs. *Discussiones Mathematicae Graph Theory*. 2017; 37(3):711-727. (In Eng.) DOI: doi.org/10.7151/dmgt.1950

[14] Desormeaux W.J., Haynes T.W., Hedetniemi S.T., Moore C. Distribution centers in graphs. *Discrete Applied Mathematics*. 2018; 243:186-193. (In Eng.) DOI: 10.1016/j.dam.2018.02.009

[15] Andrews E., Chartrand G., Lumduanhom C., Zhang P. Stars and Their k-Ramsey Numbers. *Graphs and Combinatorics*. 2017; 33(2):257-274. (In Eng.) DOI: 10.1007/s00373-017-1756-9

[16] Ahangar H.A., Fujie-Okamoto F., Samodivkin V. On the forcing connected geodetic number and the connected geodetic number of a graph. *Ars Combinatoria*. 2016; 126:323-335. (In Eng.)

[17] Babai L. Graph Isomorphism in Quasipolynomial Time. *arXiv*:1512.03547 [cs.DS]. Available at: https://arxiv.org/abs/1512.03547 (accessed 18.12.2018). (In Eng.)

[18] Schmidt D.C., Druffel L.E. A Fast Backtracking Algorithm to Test Directed Graphs for Isomorphism Using Distance Matrices. *Journal of the ACM*. 1976; 23(3):433-445. (In Eng.) DOI: 10.1145/321958.321963

[19] Harary F. Graph Theory. Reading, MA: Addison-Wesley, 1969. (In Eng.)

[20] McKay B.D. Practical graph isomorphism. *Congressus Numerantium*. 1981; 30:45-87. Available at: http://users.cecs.anu.edu.au/~bdm/nauty/pgi.pdf (accessed 18.12.2018). (In Eng.)

[21] McKay B.D., Piperno A. Practical graph isomorphism, II. *Journal of Symbolic Computation*. 2014; 60:94-112. (In Eng.) DOI: 10.1016/j.jsc.2013.09.003

[22] Volodicheva M.I., Leora S.N. Study of graph isomorphism using Jordan forms of adjacency matrices. *Prikladnaya Diskretnaya Matematika*. 2018; 40:87-90. (In Russ.) DOI: 10.17223/20710410/40/7

[23] Melnikov B.F., Saifullina E.F. Generation of graphs with prespecified sequences of degrees of order two and the isomorphism detection problem. *Stohastičeskaâ optimizaciâ v informatike*. 2014; 10(2):24-36. Available at: https://elibrary.ru/item.asp?id=22764354 (accessed 18.12.2018). (In Russ.)

[24] Harary F., Palmer E.M. Graphical Enumeration. New York; London; Oxford; Bosto; San Diego: Academic Press, 2014. (In Eng.)

[25] Churikova N.P. On the differentiation of graphs on the basis of quickly calculated invariants. *Proceedings of the "XII Belorussian Mathematical Conference"*. Vol. 4. Minsk, 2016, pp. 67-69.

[26] Melnikov B.F., Saifullina M.R. Some algorithms for equivalent transformation of nondeterministic finite automata. *Russian Mathematics*. 2009; 53(4):54-57. (In Eng.) DOI: 10.3103/S1066369X09040100

[27] Melnikov B.F., Melnikova A.A. Edge-minimization of non-deterministic finite automata. *Korean Journal of Computational and Applied Mathematics*. 2001; 8(3):469-479. (In Eng.) DOI: 10.1007/BF02941980

Submitted 18.12.2018; revised 20.02.2019;
published online 19.04.2019.

About the authors:

Boris F. Melnikov, Professor, Department of Information Systems, Networks and Safety, Faculty of Information Technology, Russian State Social University (4 Wilhelm Pieck St., build. 1, Moscow 129226, Russia), Dr. Sci. (Phys.-Math.), ORCID: http://orcid.org/0000-0002-6765-6800, bf-melnikov@yandex.ru

Nadezhda P. Churikova, post-graduate student, Department of Computer Science and Computational Mathematics, Samara National Research University (34 Moskovskoye shosse, Samara 443086, Russia), ORCID: http://orcid.org/0000-0002-4082-5838, claorisel@gmail.com

All authors have read and approved the final manuscript.