Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра теоретических основ компьютерной безопасности и криптографии

Генерация неизоморфных k-раскрасок для заданного графа без проверки на изоморфизм

АВТОРЕФЕРАТ

дипломной работы

студента 6 курса 631 группы
специальности 10.05.01 Компьютерная безопасность
факультета компьютерных наук и информационных технологий
Разумовского Петра Владимировича

Научный руководитель
профессор, д.ф.-м.н. М.Б. Абросимов

Заведующий кафедрой
профессор, к.ф.-м.н. В.Н. Салий

Саратов 2017

ВВЕДЕНИЕ

Одной из задач, решаемых в теории графов, является задача генерации цветных графов, или, для краткости, раскрасок графа. Очевидно, что при генерации изоморфные друг другу графы не представляют интереса, поэтому задача состоит не просто в генерации раскрасок, но в генерации неизоморфных раскрасок. Простейшим способом отсечения изоморфных графов является проверка каждого графа на изоморфизм с другими, ранее полученными графами. Однако, так как на данный момент не существует полиномиальных алгоритмов проверки изоморфности двух графов, а также количество графов может расти экспоненциально, такой метод является крайне неэффективным. Поэтому существуют методы, позволяющие производить генерацию раскрасок без дополнительной проверки на изоморфизм.

В данной работе исследуется генерация всех неизоморфных k-раскрасок для заданного графа. Рассматриваются как вершинные, так и реберные раскраски. Задачу о нахождении неизоморфных реберную k-раскрасок удалось переложить на генерирование вершинных k-раскрасок графа.

Будут рассматриваться раскраски неориентированных графов, однако полученные алгоритмы естественным образом можно перенести и на случай ориентированных графов.

В первом разделе приводятся основные определения, связанные с раскрасками графов: дается определения раскраски, k-раскрашиваемости графа и связанные с ним понятия. Рассматривается приведение задачи о генерации реберных раскрасок к генерации вершинных раскрасок.

Описывается разработанный Брenданом МакКеем метод построения группы автоморфизмов, используемый для генерации неизоморфных раскрасок.

Рассматривается построение канонической нумерации для цветного графа и использование данного построения для ускорения генерации неизоморфных раскрасок.

Поскольку выясняется, что в алгоритме МакКея построения группы автоморфизмов не учитывается цикличность, вводится дополнительная проверка раскраски, не встречалась ли она ранее.

Далее задается алгоритм генерации неизоморфных k-раскрасок для заданного графа, представляется его вычислительная сложность.

Для сравнения описывается алгоритм из [5] и приводится его вычислительная сложность.

В практической части описывается интерфейс программы, реализующий алгоритм генерации, приводятся результаты работы программы на различных входных графах: веретене Мозера, цепях и циклах до 11 вершин, графе Петерсена.

КРАТКОЕ СОДЕРЖАНИЕ

Дипломная работа состоит из определений, введения, 8 разделов, заключения, списка использованных источников и одного приложения. Общий объем работы – 55 страниц, из них 43 страниц – основное содержание, включая 17 рисунков и 8 таблиц, список использованных источников из 10 наименований.

Первый раздел является вводным и задает теоретическую базу дипломной работы: в нем даются определения вершинной и реберной раскраски, а также связанные с ними понятия. В последнем пункте данного раздела приводятся сферы, в которых задача раскраски находит свое применение.

Задача о генерации реберной раскраски для графа сводится к задаче о генерации вершинной раскраски его реберного графа, а значит необходимо описать, что такое реберный граф, и задать метод построения реберного графа из исходного. Поэтому данный раздел 2 содержит определение реберного графа, частные случаи, в которых построение реберного графа не представляется возможным. Далее в данном разделе описывается метод построения реберного графа на примере заданного графа.

В разделе 3 описываются автоморфизмы графа (и связанные с ними определения) и алгоритм построения группы автоморфизмов графа, разработанный Бренданом МакКеем [2]. Данный алгоритм позволяет избавиться от проверки очередной раскраски на изоморфность другим, полученным ранее, раскраскам. Раскрывается, что алгоритм в ходе работы умеет строить каноническую нумерацию, что обеспечивает минимальность и ускорение процесса нахождения новых неизоморфных раскрасок. Алгоритм построения группы автоморфизмов обладает полезными возможностями, применяемыми в ходе генерации раскрасок: вычисление орбит для заданного разбиения графа и построение канонической нумерации. Данный алгоритм реализован в программе *nauty* [8].

В следующем разделе приводится простейший алгоритм генерации неизоморфных раскрасок, рассматриваемый в [5]. Такой алгоритм позволяет генерировать раскраски, выбирая по одному представителю из очередного множества орбит, полученного разбиением, и проверять каждую новую раскраску, существовала ли она ранее, путем хранения множества всех раскрасок и сравнения, есть ли новая раскраска в данном множестве. Нетрудно заметить, что такой подход является крайне неэффективным: хранить множество раскраски и сравнивать каждую новую с текущим множеством – довольно затратно как по времени, так и по памяти. Кроме того, такой подход нельзя назвать генерация без проверки на изоморфизм. Поэтому далее приводится метод, который позволит составить более быстрый алгоритм, который не использует проверку на изоморфизм.

В разделе 5 описывается алгоритм отсечения изоморфизмов – он является абстрактным и подходит для любых методов генерации неизоморфных графов. На базе него в данном разделе строится метод генерации неизоморфных вершинных раскрасок – он позволяет отсекать раскраски, которые потенциально изоморфны ранее полученным раскраскам, не достраивая раскраски до конца, то есть для всех вершин. Итак, метод генерации неизоморфных раскрасок получен, и теперь по нему можно построить пошаговый алгоритм. Однако, есть еще один момент, который следует отметить.

Данный момент состоит в особенности программы *nauty* и описан в разделе 6. Дело в том, что программа не учитывает наличие циклов в графе, поэтому для нее две одинаковые раскраски на цикле могут считаться неизоморфными. Вследствие данного факта, приходится вводить дополнительную проверку раскраски, заключающуюся в переборе возможных пар цветов, перекраске раскраски и проверки: во-первых, меньше ли данная раскраска, чем исходная; во-вторых, удовлетворяет ли перекрашенная раскраска условиям метода генерации неизоморфных раскрасок. Если оба условия выполняется, то считается, что данная раскраска уже существует, и она отсекается. Эта проверка несколько ухудшает скорость генерации, однако позволяет генерировать абсолютно неизоморфные раскраски для заданного графа.

Теперь можно перейти к описанию пошагового алгоритма генерации. Данный алгоритм описывается в седьмом разделе. Основным алгоритмом является алгоритм генерации неизоморфных k-раскрасок для заданного графа без проверки на изоморфизм. После того, как этот алгоритм задается, можно рассмотреть его модификации. Алгоритм модифицируется для следующих задач: генерация неизоморфных реберных раскрасок, поиск правильных вершинных и реберных раскрасок, поиск хроматического числа и хроматического индекса. Для генерации реберных раскрасок также описывается пошаговый алгоритм построения реберного графа. В конце раздела приводится вычислительная сложность построения реберного графа, основного алгоритма и всех его модификаций.

Раздел 8 содержит практическую часть. В первом пункте раздела приводится руководство к использованию программы, реализующий алгоритм генерации, задается формат входных данных и режимы работы программы, способ ее запуска. Далее, во втором пункте, представлены результаты запуска программы на различных входных графах – цепях и циклах до 11 вершин, графе Петерсена и веретене Мозера.

Результаты показали существенное улучшение скорости по сравнению с реализованным алгоритмом из [5]. Так, например, для веретена Мозера время, затраченное на вычисление реберных 6-раскрасок, составило 6.991 секунд, тогда как для предыдущего алгоритма – 29.863 секунд.

Для цепей и циклов размера n были посчитаны вершинные и реберные раскраски с количеством цветов от 2 до $n-1$. Пример количества вершинных раскрасок для циклов представлен в таблице 1.

Таблица 1.

Количество вершин	Количество цветов	Количество раскрасок	Количество правильных раскрасок
3	2	1	0
4	2	3	1
	3	3	1

5	2	1	0
	3	1	1
	4	6	3
6	2	9	1
	3	35	2
	4	32	10
	5	9	6
7	2	16	0
	3	107	6
	4	152	25
	5	72	28
	6	12	9
8	2	35	1
	3	324	11
	4	704	78
	5	502	127
	6	140	65
	7	16	12
9	2	70	0
	3	1083	31
	4	3316	318
	5	3297	662
	6	1319	457
	7	241	128
	8	20	16
10	2	131	1
	3	3202	66
	4	14282	1035
	5	19822	3059
	6	11194	2981
	7	2955	1214
	8	389	230
	9	25	21
11	2	262	0
	3	8947	130
	4	57864	3498
	5	111363	15436
	6	86259	21108
	7	31650	11829
	8	5960	3099
	9	594	402
	10	30	26

ЗАКЛЮЧЕНИЕ

В данной работе было произведено исследование генерации неизоморфных цветных графов с цветными вершинами и цветными ребрами, то есть генерация вершинных и реберных k-раскрасок соответственно. Была использована техника генерации раскрасок без проверки на изоморфизм – при помощи построения канонической нумерации графа и группы его автоморфизмов.

Описываются два алгоритма генерации неизоморфных раскрасок. Приведены доводы, почему первый алгоритм является неэффективным по сравнению с новым алгоритмом; к тому же он не может называться алгоритмом без проверки на изоморфизмы.

Так как алгоритм построения группы автоморфизмов не учитывает циклы в графе, была введена дополнительная проверка раскраски, не встречалась ли она ранее.

Был выведен алгоритм генерации неизоморфных вершинных раскрасок без проверки на изоморфизм, использующий программу *nauty* для вычисления орбит и канонической нумерации. Были описаны его модификации для реберных раскрасок, поиска правильных раскрасок, хроматического числа и хроматического индекса.

Далее данный алгоритм был программно реализован и протестирован на различных типах графов: веретене Мозера, цепях и циклах до 11 вершин, графе Петерсена.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Абросимов, М.Б. Практические задания по графам: учебное пособие / М.Б. Абросимов, А.А. Долгов. Сар.: Научная книга, 2009. 76 с.

2. McKay, B.D. Practical Graph Isomorphism II [Электронный ресурс] / B.D. McKay, A. Piperno. J. Symbolic Computation, 2013. #60, P.94-112. URL: http://cs.anu.edu.au/~bdm/nauty/pgi.pdf (дата обращения: 10.11.2016) Загл. с экрана. Яз. англ.

3. Харари, Ф. Теория графов / Ф. Харари. М.: Мир, 1973. 296 с.

4. Павлов, Д.А. Каноническая нумерация графов и библиотека nauty [Электронный ресурс] / Д.А. Павлов // Компьютерные инструменты в образовании [Электронный ресурс]. Санкт-Петербург: Санкт-Петербургский государственный электротехнический университет "ЛЭТИ" им. В.И. Ульянова (Ленина), 2009. URL: http://ipo.spb.ru/journal/content/1113/Каноническая%20нумерация%20графов%20и%20библиотека%20nauty.pdf (дата обращения: 09.12.2016). Загл. с экрана. Яз. рус.

5. Абросимов, М.Б. Генерация неизоморфных вершинных k-раскрасок графа / Абросимов М.Б., Разумовский П.В. // Компьютерные науки и информационные технологии: материалы Междунар. науч. конф. Саратов: издат. центр «Наука» , 2016. С 13-15.

6. Brinkman, G. Isomorphism Rejection in Structure Generation Programs / G. Brinkman // DIMACS series in discrete mathematics and theoretical computer science: discrete mathematical chemistry. Providence, RI: 2000. P. 25-38.

7. Богомолов, А.М. Алгебраические основы теории дискретных систем // А.М. Богомолов, В.Н. Салий. М.: Наука. Физматлит, 1997. 368 с.

8. McKay, B.D. nauty User's Guide (version 2.5) [Электронный ресурс] / B. D. McKay. Tech. Rpt., Dept. Computer Science [Электронный ресурс] // Austral. Nat. Univ., 1990. 94 P. URL: http://cs.anu.edu.au/~bdm/nauty/nug26.pdf (дата обращения: 11.12.2016) Яз. англ.

9. Веретено Мозера [Электронный ресурс] / Википедия [Электронный ресурс]: свободная энциклопедия / текст доступен по лицензии Creative

Commons Attribution-ShareAlike; Wikimedia Foundation, Inc, некоммерческой организации. Электрон. дан. (712413 статей, 2479181 страниц, 117104 загруженных файлов). Wikipedia®, 2001- . URL: http://ru.wikipedia.org/wiki/Веретено_Мозера (дата обращения: 21.12.2016). Загл. с экрана. Последнее изменение страницы: 21:26, 25 августа 2016. Яз. рус.

10. Граф Петерсена [Электронный ресурс] / Википедия [Электронный ресурс]: свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike; Wikimedia Foundation, Inc, некоммерческой организации. Электрон. дан. (712413 статей, 2479181 страниц, 117104 загруженных файлов). Wikipedia®, 2001- . URL: http://ru.wikipedia.org/wiki/Граф_Петерсена (дата обращения: 21.12.2016). Загл. с экрана. Последнее изменение страницы: 21:22, 26 декабря 2014. Яз. рус.