МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра теоретических основ компьютерной безопасности и криптографии

Поиск зависимостей между инвариантами графов

АВТОРЕФЕРАТ

дипломной работы

студентки 6 курса 631 группы

специальности 10.05.01 Компьютерная безопасность

факультета компьютерных наук и информационных технологий

Килимнюк Анастасии Сергеевны

Научный руководитель

д. ф.-м. н., доцент ___________ М. Б. Абросимов

23.01.2020 г.

Заведующий кафедрой

д. ф.-м. н., доцент ___________ М. Б. Абросимов

23.01.2020 г.

Саратов 2020

ВВЕДЕНИЕ

Задача проверки изоморфизма графов состоит в нахождении биективного отображения множества вершин одного графа на множество вершин другого графа с сохранением отношения смежности. Характеристики графа, одинаковые для всех изоморфных ему графов, называются инвариантами. Таким образом получается, что, если значение хотя бы одного инварианта различается для двух графов, можно сделать вывод, что эти графы не изоморфны. Однако графы с одинаковым значением некоторого набора инвариантов не обязательно будут изоморфными. Полным инвариантом называется такой инвариант, который определяет граф однозначно с точностью до изоморфизма. На данный момент неизвестно эффективного алгоритма решения задачи проверки изоморфизма, соответственно неизвестно и эффективного вычислимых полных инвариантов.

Помимо очевидных инвариантов, таких как число рёбер или вершин, часто для решения задачи изоморфизма используют и другие инварианты графов. В данной работе будут описаны и вычислены такие из них как:

- количество компонентов связности,
- вершинная связность,
- рёберная связность,
- локальная связность,
- двудольность,
- обхват,
- чётный и нечётный обхват,
- окружность,
- количество треугольников,
- радиус,
- диаметр,
- хроматическое число,
- хроматический индекс,

- примитивность,
- экспонент,
- максимальная клика,
- число независимости,

а также рассмотрены наборы инвариантов и возможность их использования для предварительной оценки изоморфизма данных графов.

Другой целью работы является построение и проверка гипотез взаимосвязи значений для нахождения потенциальных зависимостей между различными инвариантами графа.

Дипломная работа состоит из введения, 4 разделов, заключения, списка использованных источников и 4 приложений. Общий объем работы – 91 страница, из них 40 страниц – основное содержание, включая 17 рисунков и 3 таблицы, список использованных источников из 26 наименований.

КРАТКОЕ СОДЕРЖАНИЕ

В разделе 1 приводятся основные необходимые определения теории графов.

Раздел 2 посвящен инвариантам графов и алгоритмам их вычислений. В пункте 2.1 рассматриваются связность и количество компонент связности. В подпункте 2.1.1 описана реализация алгоритма поиска в глубину, с помощью которого осуществляется подсчет компонент связности графа. Следующий пункт – 2.2 – описывает такие инварианты как вершинная и рёберная связности. Для нахождения этих инвариантов используются теоремы Менгера и алгоритм Форда-Фалкерсона нахождения максимального потока в графе, которые рассматриваются в 2.2.1.

Теорема 1 (Менгера для вершинной связности).

Граф k-вершинно связен тогда и только тогда, когда любая пара его вершин соединена по крайней мере k-вершинно непересекающимися цепями.

Теорема 2 (Менгера для рёберной связности).

Граф G k-рёберно связен тогда и только тогда, когда любая пара его вершин соединена по крайней мере k-рёберно непересекающимися цепями.

Таким образом для нахождения рёберной связности необходимо перебрать все пары вершин s и t, найти количество непересекающихся путей из s в t и выбрать из них минимум.

Пусть этот минимум равен l. Граф является l-связным, причем l - максимально. Значит, по определению, реберная связность равна l.

Для нахождения количества непересекающихся путей из s в t необходимо воспользоваться алгоритмом нахождения максимального потока. Сопоставим каждому ребру пропускную способность, равную 1 и найдем максимальный поток с помощью алгоритма Форда-Фалкерсона. Найденный максимальный поток и будет равен количеству путей.

Для нахождения вершинной связности используется тот же алгоритм, но с тем отличием, что происходит поиск вершинно непересекающихся путей. В этом случае каждой вершине сопоставляется пропускнаю способность, равная 1. Для этого каждую вершину v графа разбивается на две вершины v_1 и v_2. Все рёбра, которые входили в v будут входить в v_1. Все рёбра, которые выходили из v будут выходить из v_2. Так же добавляется ребро (v_1, v_2) с пропускной способностью 1.

После этого для нахождения количества вершинно непересекающихся путей в исходном графе происходит поиск количества рёберно непересекающихся путей в новом графе. Тем самым задача сводится к нахождению рёберной связности.

В 2.3 даны определения окружения вершины графа, локально связной вершины и локальной связности графа. Для вычисления локальной связности необходима проверка на связность окружения каждой из вершин графа.

В пункте 2.4 определены такие инварианты как обхват, чётный и нечётный обхваты, а также окружность графа. Алгоритм нахождения этих инвариантов, как и подсчет компонентов связности, использует поиск в глубину. Только в этом случае необходимо произвести поиск всех возможных циклов в графе, а после найти минимальный (обхват), минимальный чётный и нечётный (чётный и нечётный обхваты соответственно) и максимальный из них (окружность).

Пункт 2.5 – количество треугольников в графе – количество различных циклов длины 3. Для вычисления этого инварианта используются три цикла, с помощью которых проверяется наличие рёбер между каждой тройкой вершин.

Для нахождения радиуса и диаметра графа (2.6) необходимо вычислить все эксцентриситеты вершин. Для этого сначала находятся кратчайшие пути между всеми парами вершин с помощью алгоритма Флойда-Уоршелла и для каждой вершины вычисляется максимальное расстояние от неё до всех остальных вершин. После этого находится радиус и диаметр поиском минимального и максимального из эксцентриситетов графа. В подпункте 2.6.1 описан алгоритм Флойда-Уоршелла.

Пункт 2.7 дает определение двудольного графа. Для проверки графа на двудольность используется поиск в ширину, описанный в подпункте 2.7.1.

Для хроматического числа и хроматического индекса в пункте 2.8 приводятся теоремы, устанавливающие верхние и нижние границы значений. Для вычисления хроматического числа необходимо предпринять попытку раскраски вершин графа в определенное количество цветов в некотором диапазоне. Нижней границей для поиска установим кликовое число, а верхней – $1 + \Delta$. Для вычисления хроматического индекса граф в первую очередь нужно проверить на двудольность, и в случае положительного исхода значение индекса устанавливается как Δ. Поиск хроматического индекса не двудольных графов происходит с помощью полного перебора вариантов раскраски рёбер графа в Δ цветов. В случае неудачи хроматическим индексом графа будет значение $\Delta + 1$.

Пункт 2.9 дает определения примитивного графа и экспонента графа. В подпункте 2.9.1 описан алгоритм проверки графа на примитивность и нахождение его экспонента.

Пункты 2.10 и 2.11 описывают такие инварианты как максимальная клика и число независимости соответственно. Для вычисления этих инвариантов используется алгоритм Брона-Кербоша (2.10.1). Задачи нахождения числа независимости и максимальной клики по сути эквивалентны. Каждая из них получается из другой, путем построения дополнения графа. Поэтому для нахождения числа независимости можно использовать алгоритм Брона-Кербоша, запустив его для дополнения графа.

Раздел 3 содержит информацию о практической части дипломной работы. В пункте 3.1 описывается способ генерации графов в формате *graph6* с использованием генератора *geng* из пакета *nauty*.

Информация о реализованных в ходе работы программах содержится в пункте 3.2. Первая программа – для вычисления инвариантов – для большего быстродействия была написана на языке программирования C++ в среде JetBrains CLion 2018.3. Вторая программа представляет собой набор команд для анализа результатов, полученных в первой программе. Она написана на языке Python 3 в среде JetBrains PyCharm Community Edition 2018.1. В пункте

3.2.1 приводится список вычисленных инвариантов и их символьные обозначения в контексте данной работы. Также там описывается принцип работы первой программы и показан пример данных, полученных в результате её работы. Листинг данной программы можно найти в приложении А.

В 3.2.2 приводится анализ результата разбиения значений инвариантов на классы. Для этого используются такие инварианты как: количество компонент связности, двудольность, диаметр, радиус, локальная связность, вершинная и рёберная связности, количество рёбер, максимальная степень вершин графа и значение экспонента (Таблица 1).

Таблица 1 – Разбиение графов на классы

n	**Количество графов**	**Количество классов**
4	11	11
5	34	33
6	156	116
7	1044	335
8	12346	1058
9	274668	3320
10	12005168	11253

Программа поиска зависимостей между инвариантами имеет консольный интерфейс для удобства взаимодействия. В 3.2.3 подробно описываются все пять режимов её работы:

1 – вычисление зависимостей между инвариантами

2 – поиск гипотез по инварианту

3 – поиск гипотез, корректных только для определенного класса графов

4 – подсчет количества инвариантов

5 – разбиение значений на классы

Первые три режима основные – они работают с шаблонами гипотез и уже полученными гипотезами. В программе выполняется проверка гипотез по следующим шаблонам:

- $\text{инвариант1} \leq \text{инвариант2} + \text{инвариант3}$
- $\text{инвариант1} \leq \text{инвариант2} * \text{инвариант3}$
- $\text{инвариант1} \leq \text{инвариант2}^{\text{инвариант3}}$
- $\text{инвариант1} \leq \text{инвариант2} + 1$
- $\text{инвариант1} \leq \text{инвариант2} - 1$
- $\text{инвариант1} \leq \text{инвариант2} * 2$
- $\text{инвариант1} \leq \text{инвариант2}/2$

Полный список гипотез, корректных только для графов определенного типа, можно найти в приложении Г.

Четвертый и пятый режимы работы программы необходимы для получения других результатов по графам. Результаты работы четвертого режима представлены в таблицах в приложении В. Для анализа, проведенного в 3.2.2, использовался пятый режим работы. Листинг второй программы представлен в приложении Б.

Раздел 4 включает в себя обзор некоторых известных ресурсов с графами. В 4.1 рассматривается онлайн-база данных «Мир Графов», в 4.2 – веб портал The House of Graphs, а в 4.3 – описана библиотека графов ГрафоMann.

ЗАКЛЮЧЕНИЕ

В данной работе были описаны и реализованы алгоритмы нахождения таких инвариантов графа как: количество компонент связности, вершинная и рёберная связности, локальная связность, двудольность, обхват, чётный и нечётный обхват, окружность, количество треугольников, радиус, диаметр, хроматическое число и хроматический индекс, примитивность и экспонент, максимальная клика и число независимости. Был проведен анализ результатов разбиения графов на классы по значениям самых эффективно вычислимых инвариантов из вышеназванного списка.

Была разработана программа для генерации гипотез по определённым шаблонам для заданных инвариантов в классе определённых графов. Отдельной главой были рассмотрены несколько ресурсов, связанных с графами. Все они содержат базу данных графов и некоторые их характеристики. Преимуществом программ, написанных в рамках дипломной работы, может считаться расчёт инвариантов для всех неизоморфных графов до десяти вершин включительно, а также получение базы гипотез. Вычисленные зависимости могут быть использованы для дальнейших исследований связей выбранных инвариантов графов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Богомолов, А.М. Алгебраические основы теории дискретных систем / А. М. Богомолов, В.Н. Салий. – М. : «Физико-математическая литература» РАН, 1997.

2 Абросимов, М. Б. Практические задания по графам: учеб. пособие / М. Б. Абросимов, А. А. Долгов. – 2-е изд. – Саратов : Изд-во «Научная книга», 2009.

3 Поиск в глубину [Электронный ресурс] // Википедия [Электронный ресурс]: свободная энциклопедия. URL: https://ru.wikipedia.org/wiki/Поиск_в_глубину (дата обращения: 05.09.2019). – Загл. с экрана. – Яз. рус.

4 Алгоритм Флойда-Уоршела нахождения кратчайших путей между всеми парами вершин // Иванов Максим [Электронный ресурс] // MAXimal :: algo [Электронный ресурс]. URL: http://e-maxx.ru/algo/floyd_warshall_algorithm (дата обращения: 07.10.2019). – Загл. с экрана. – Яз. рус.

5 Поиск в ширину // Иванов Максим [Электронный ресурс] // MAXimal :: algo [Электронный ресурс]. URL: http://e-maxx.ru/algo/bfs (дата обращения: 06.09.2019). – Загл. с экрана. – Яз. рус.

6 Зыков, А.А. Основы теории графов / А.А. Зыков. – М. : Вузовская книга, 2004.

7 Салий, В. Н. Минимальные примитивные расширения ориентированных графов / В. Н. Салий // Прикладная дискретная математика. 2008. №1(1).

8 Когос, К.Г. Положительные свойства неотрицательных матриц / К.Г. Когос, В.М. Фомичев // Прикладная дискретная математика. 2012. №4(18).

9 Клика (теория графов) [Электронный ресурс] // Википедия [Электронный ресурс]: свободная энциклопедия. URL: https://ru.wikipedia.org/wiki/Клика_(теория_графов) (дата обращения: 05.09.2019). – Загл. с экрана. – Яз. рус.

10 Алгоритм Брона-Кербоша [Электронный ресурс] // Википедия [Электронный ресурс]: свободная энциклопедия. URL: https://ru.wikipedia.org/wiki/Алгоритм_Брона_—_Кербоша (дата обращения: 09.09.2019). – Загл. с экрана. – Яз. рус.

11 McKay, B.D. Practical Graph Isomorphism / B.D. McKay, A. Piperno // Journal of Symbolic Computation, 60 (2014).

12 Кормен, Т. Алгоритмы. Построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн. – 3-е изд.: Пер. с англ. – М. : ООО «И. Д. Вильямс», 2013.

13 Харари, Ф. Теория графов / Ф. Харари. – пер. с англ. – М. : Издательство «Мир», 1973.

14 Алгоритм Форда-Фалкерсона [Электронный ресурс] // Википедия [Электронный ресурс]: свободная энциклопедия. URL: https://ru.wikipedia.org/wiki/Алгоритм_Форда_—_Фалкерсона (дата обращения: 15.10.2019). – Загл. с экрана. – Яз. рус.

15 Вершинная, рёберная связность, связь между ними и минимальной степенью вершины [Электронный ресурс] // Университет ИТМО [Электронный ресурс]. URL: http://neerc.ifmo.ru/wiki/index.php?title= Вершинная,_рёберная_связность,_связь_между_ними_и_минимальной _степенью_вершины (дата обращения: 12.10.2019). – Загл. с экрана. – Яз. рус.

16 Chartrand, G. Locally connected graphs / G. Chartrand, R. E. Pippert // Časopis pro pěstování matematiky, Vol. 99 (1974), Praha. No. 2, 158—163.

17 Емеличев, В. А. Лекции по теории графов / В. А. Емеличев, О. И. Мельников, В. И. Сарванов, Р. И. Тышкевич – М. : «Наука» Физматлит, 1990.

18 Свами, М. Графы, сети и алгоритмы / М. Свами, К. Тхуласираман – М. : Издательство «Мир», 1984.

19 Оре, О. Теория графов / О. Оре – М. : «Наука» Физматлит, 1980.

20 Bondy, J.A. Graph Theory / J. A. Bondy, U. S. R. Murty – 1st edition: Springer Publishing Company, Incorporated, 2008.

21 Мир графов [Электронный ресурс]. URL: http://graphworld.ru/ (дата обращения 23.11.2019). – Загл. с экрана. – Яз. рус.

22 Лутц, М. Изучаем Python / М. Лутц. – 4-е изд.: Пер. с англ. – СПб. : Символ-Плюс, 2011.

23 Страуструп, Б. Язык программирования C++ / Б. Страуструп. – Пер. с англ. – М. : Бином, 2018.

24 McKay, B. Description of graph6, sparse6 and digraph6 encodings [Электронный ресурс]. URL: http://users.cecs.anu.edu.au/~bdm/data/formats.txt (дата обращения: 04.10.2019). – Загл. с экрана. – Яз. англ.

25 The House of Graphs [Электронный ресурс]. URL: https://hog.grinvin.org/ (дата обращения: 12.01.2020). – Загл. с экрана. – Яз. англ.

26 ГрафоMann. Библиотека графов [Электронный ресурс]. URL: http://www.apmath.spbu.ru/grafomann/graphlib/ (дата обращения: 11.01.2020). – Загл. с экрана. – Яз. рус.