МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

РЕАЛИЗАЦИЯ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ РЕШЕНИЯ ЗАДАЧ О ПОИСКЕ НАИБОЛЬШЕГО ПАРОСОЧЕТАНИЯ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 441 группы
направления 02.03.03 Математическое обеспечение и администрирование информационных систем
факультета компьютерных наук и информационных технологий
Ионовой Дарьи Арсеньевны

Научный руководитель
зав. кафедрой ИиП,
к. ф.-м. н., доцент _______________ М.В. Огнева

зав. кафедрой ИиП,
к.ф.-м.н., доцент ______________ М.В. Огнева
Саратов 2021

Саратов 2021

ВВЕДЕНИЕ

Задача о поиске наибольшего паросочетания является одной из фундаментальных задач комбинаторной оптимизации. Она связана с такими прикладными задачами, в которых необходимо назначать исполнителей (людей, механизмы и т.п.) для выполнения однотипных работ.

Это достаточно актуально для работодателей в различных отраслях, таких, например, как менеджмент, производство и т.д. В этом случае предполагается выполнение одним «работником» одной «работы», и требуется получить максимальную эффективность на выходе. Или, например распределять работников по кабинетам в здании офиса, а также распределять сотрудников, например, по различным офисам компании.

Целью бакалаврской работы является изучение методов решения задачи о поиске наибольшего паросочетания, их реализация и сравнительный анализ.

Поставленная цель определила **следующие задачи**:

1. Изучение литературы, разбор основных понятий предметной области.
2. Изучение основных методов решения задачи о поиске наибольшего паросочетания.
3. Выявление достоинств и недостатков данных методов.
4. Реализация и тестирование различных методов.
5. Сравнительный анализ полученных результатов.

Методологические основы реализация алгоритмов решения задач о поиске наибольшего паросочетания представлены в работах Корбута, Финкельштейна.

Теоретическая и/или практическая значимость бакалаврской работы. Выявление достоинств и недостатков данных методов, реализация и тестирование различных их, а также сравнительный анализ полученных результатов.

Структура и объём работы. Бакалаврская работа состоит из введения, 6 разделов, заключения, списка использованных источников и 9 приложений. Общий объем работы – 112 страниц, из них 61 страниц – основное содержание, включая 2 рисунка и 7 таблиц, цифровой носитель в качестве приложения, список использованных источников информации – 20 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Основные определения» посвящен основным определениям, которые используются в выпускной квалификационной работе.

Второй раздел «Методы решения задачи поиска наибольшего паросочетания» посвящен алгоритмам Куна и Хопкрофта-Карпа. Прежде чем дать определение алгоритму Куна, сформулируем теорему Бержа.

Теорема Бержа

Формулировка: Паросочетание является максимальным тогда и только тогда, когда не существует увеличивающих относительно него цепей.

Алгоритм Куна является непосредственным применением теоремы Бержа и работает следующим образом.

1)Берется пустое паросочетание.

2)Пока в графе удаётся найти увеличивающую цепь, — выполняется чередование паросочетания вдоль этой цепи.

3)Повторять процесс поиска увеличивающей цепи.

4)Как только такую цепь найти не удалось — процесс останавливается, — текущее паросочетание и есть максимальное.

Увеличивающую цепь в п.3 можно искать разными способами. Алгоритм Куна использует обход в глубину в ширину, просматривая все вершины графа по очереди.

Алгоритм Куна исполняется за время $O(nm)$, что в худшем случае есть $O(n^3)$. Где n – число вершин в первой доле.

Алгоритм Хопкрофта-Карпа – алгоритм, принимающий на вход двудольный граф, на выходе которого получается паросочетание максимальной мощности (наибольший набор ребер, таких, что никакие две вершины не могут иметь общую точку).

Третий раздел «Поиск максимального потока» посвящен решению задачи о поиске наибольшего паросочетания через поиск максимального потока 2 способами: метод Форда-Фалкерсона и метод «поднять в начало».

Поиск максимального потока с помощью метода Форда-Фалкерсона

Дан неориентированный граф G (V, E) и требуется найти максимальное паросочетание в нём. Доли исходного графа обозначаются как L и R. Строится граф G′ (V′, E′) следующим образом:

V′=V∪{s,t} (т.е. добавляется новый исток s и сток t);

Изначально текущее паросочетание пусто. На каждом шаге алгоритма в текущее найденное паросочетание входят те и только те ребра, которые направлены из R в L.

Порядок выполнения алгоритма

В графе G′ находится путь из s в t поиском в глубину.

Если путь найден, перезаписывается текущее паросочетание. Далее инвертируются все рёбра на пути (ребро (u,v)становится ребром (v,u)) и удаляются (s,L) и (R,t) ребра, покрывающие вершины, принадлежащие текущему паросочетанию.

Если путь не был найден, значит текущее паросочетание является максимальным, и алгоритм завершает работу. Иначе переходим к пункту 1.

Поиск в глубину работает за O(E), запускаюсь из вершины s не более чем L раз и при каждом запуске 1 из этих ребер инвертируется, каждая инвертация и перезапись паросочетания так же занимает O(E)времени. Тогда все время алгоритма ограничено O(VE).

Метод «поднять в начало»

Алгоритм «поднять-в-начало» основан на методе проталкивания предпотока, однако из-за определенного порядка выполнения операций проталкивания и подъема, время выполнения данного алгоритма составляет

$O(V^3)$, что асимптотически не хуже, чем $O(V^2E)$[10].

Метод проталкивания предпотока — обобщенный алгоритм нахождения максимального потока в графе.

Четвертый раздел «Методы решения задачи о назначениях» посвящен методам решения задачи о назначениях, а именно венгерскому алгоритму, алгоритму Мака, метод «ветвей и границ» и сведение задачи о назначениях к задаче о потоке минимальной стоимости.

Общий метод венгерского алгоритма: Алгоритм, решающий задачу, работает с графом, как с матрицей весов.

1) Вычитается из каждой строки значение ее минимального элемента. Теперь в каждой строке есть хотя бы один нулевой элемент.

2) Вычитается из каждого столбца значение его минимального элемента. Теперь в каждом столбце есть хотя бы один нулевой элемент.

3) Ищется в текущем графе полное паросочетание из ребер нулевого веса.

4) Если оно найдено, то желаемый результат достигнут, алгоритм закончен.

5) В противном случае, покрываются нули матрицы весов минимальным количеством строк и столбцов (это нахождение минимального вершинного покрытия в двудольном графе). Пусть Xc и Yc — множества вершин минимального вершинного покрытия из левых и правых долей (то есть, строк и столбцов) соответственно, тогда применяется преобразование $Xc \uparrow\downarrow (Y \setminus Yc)$. Для этого преобразования d будет минимумом по всем ребрам

между $X \setminus Xc$ и $Y \setminus Yc$, то есть, ребер нулевого веса здесь нет, поэтому, после его выполнения в матрице весов появится новый нуль. После этого переход к шагу 1.

Метод Мака

В основе методики решения данной задачи методом Мака лежит принцип, заключающийся в том, что положения оптимального выбора не меняются, если к каждому элементу какой-либо строки или столбца добавить одно и то же число или вычесть его.

Метод «ветвей и границ»

В основе данного алгоритма лежит идея последовательного разбиения множества допустимых решений. На каждом шаге метода элементы разбиения (подмножества) подвергаются анализу – содержит ли данное подмножество оптимальное решение или нет.

Сведение задачи о назначениях к задаче о потоке минимальной стоимости

Задача о назначениях сводится к задаче нахождения потока минимальной стоимости.

Пятый раздел «Реализация» посвящен реализации консольного приложения для работы с методами решения задачи о поиске наибольшего паросочетания, а также для решения задачи о назначениях. Представлены сущности: KuhnClass, HopcroftKarpClass, FordFulkersonClass, RelabelToFrontClass, HungarianClass, MackClass, BranchesClass и MaxFlowClass. Для каждой сущности были созданы DAL и BLL слои, а также интерфейсы для создания слабой связанности в приложении, построенным на трехслойной архитектуре, где DAL – слой доступа к данным, а BLL – слой бизнес-логики. С помощью класса HostBuilder были созданы зависимости между слоями в приложении Program. Также был создан слой PL в виде консольного приложения. В каждом классе полями являются целочисленная и вещественная матрицы, задающиеся случайным образом с помощью функций GetRandomMas, GetRandomMasAdj и GetRandomMasDouble соответственно, находящимися в классе Worker.

Шестой раздел «Сравнительный анализ» посвящен сравнительному анализу алгоритмов.

Исходя из результатов сравнительного анализа, можно сделать вывод, что среди алгоритмов о поиске наибольшего паросочетания Кун работает лучше, чем Хопкрофт-Карп на всех размерностях. А вот среди алгоритмов поиска максимального потока на малых размерностях быстрее всех справился Форд-Фалкерсон на всех размерностях.

Среди же методов решения задачи о назначениях можно заметить, что Венгерский алгоритм оказался очень эффективным на малых значениях, однако уже на средних и больших значениях значительно медленнее поиска максимального потока. На вещественных матрицах, казалось бы, результаты похожи с целочисленными, однако можно заметить, что алгоритм Мака работает значительно лучше на вещественных значениях и даже является самым быстрым по времени выполнения на вещественных матрицах.

ЗАКЛЮЧЕНИЕ

В ходе работы были изучены и реализованы основные методы решения задачи о поиске наибольшего паросочетания, выявлены недостатки и достоинства данных методов, а именно алгоритмов Хопкрофта-Карпа и Куна. В том числе были изучены и реализованы методы для сведения задачи к поиску максимального потока (Форд-Фалкерсон, «Поднять-в-начало»).

Аналогично были исследованы и реализованы алгоритмы для решения задач о назначениях, а именно Венгерский, метод Мака, метод Ветвей и границ и поиск максимального потока.

Исходя из результатов сравнительного анализа, можно сделать вывод, что среди алгоритмов о поиске наибольшего паросочетания Кун работает лучше, чем Хопкрофт-Карп на всех размерностях. А вот среди алгоритмов поиска максимального потока на малых размерностях быстрее всех справился Форд-Фалкерсон на всех размерностях.

Среди же методов решения задачи о назначениях можно заметить, что Венгерский алгоритм оказался очень эффективным на малых значениях, однако уже на средних и больших значениях значительно медленнее поиска максимального потока. На вещественных матрицах, казалось бы, результаты похожи с целочисленными, однако можно заметить, что алгоритм Мака работает значительно лучше на вещественных значениях и даже является самым быстрым по времени выполнения на вещественных матрицах. Но также стоит заметить, что метод ветвей и границ работает хуже на вещественных, чем на целочисленных. Относительно двудольных целочисленных графов: алгоритм Мака стал работать лучше на двудольных графах и даже оказался самым быстрым на малых размерностях. Однако, например Венгерский алгоритм стал работать хуже на всех матрицах. Метод ветвей и границ, как и метод Мака работает намного быстрее на двудольных. Поиск потока оказался самым быстрым на средних и больших размерах матрицы. А вот метод поднять-в-начало стал работать хуже на двудольных графах. Тем не менее на вещественных графах поиск потока работает быстрее всех на матрицах малых и средних размерностях, однако на больших размерностях метод Мака оказался самым быстрым из представленных.

В завершении, можно сделать вывод, что выбор алгоритма зависит от многих различных факторов и, в общем и целом, не существует идеального алгоритма, поэтому стоит выбирать алгоритм в зависимости от задачи.

Основные источники информации:

1. Основные определения теории графов [Электронный ресурс]. URL:https://neerc.ifmo.ru/wiki/index.php?title=Основные_определения_теории_графов (Дата обращения 05.04.2021). Загл. с экр. Яз. Рус.

2. Глоссарий теории графов. [Электронный ресурс]. URL:https://ru.wikipedia.org/wiki/Глоссарий_теории_графов (Дата обращения 05.04.2021). Загл. с экр. Яз. Рус.

3.Паросочетания: основные определения, теорема о максимальном паросочетании и дополняющих цепях. [Электронный ресурс]. URL:https://neerc.ifmo.ru/wiki/index.php?title=Паросочетания:_основные_определения,_теорема_о_максимальном_паросочетании_и_дополняющих_цепях (Дата обращения 05.04.2021). Загл. с экр. Яз. Рус.

4. ПРИМЕНЕНИЕ МЕТОДА МАКА В РЕШЕНИИ ЗАДАЧ О НАЗНАЧЕНИЯХ Россолов С. Ю. [Электронный ресурс]. URL: https://publikacija.ru/images/PDF/2017/13/primenenie-metoda-maka.pdf (Дата обращения 04.04.2020). Загл. с экр. Яз. Рус.

5. А. А. Корбут., Ю. Ю. Финкельштейн. - Дискретное программирование - М. Издательство «Наука» ,1970 — 366 с.

6. Бирюков А.С., Ищук М.А., Огнева М.В. Использование алгоритмов теории графов в задачах по программированию повышенной сложности//Информационные технологии в образовании. 2016. с. 25-30.

7. Банди Б. Основы линейного программирования /Пер. с англ. - М.: Радио и связь, 1989. - 176с.

8. Ху Т. Ч., Шинг М. Т. Комбинаторные алгоритмы / Пер. с англ. Нижний Новгород: Изд-во Нижегородского госуниверситета им. Н. И. Лобачевского, 2004. — 330 с.

9. Кирсанов М. Н. Графы в Maple. Задачи, алгоритмы, программы. — М.: Издательства ФИЗМАТЛИТ, 2007. — 168 с