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

МОДЕЛИ И АЛГОРИТМЫ ПОСТРОЕНИЯ ОПТИМАЛЬНЫХ РЁБЕРНЫХ ОТКАЗОУСТОЙЧИВЫХ РЕАЛИЗАЦИЙ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

АВТОРЕФЕРАТ
НАУЧНО-КВАЛИФИКАЦИОННОЙ РАБОТЫ

аспиранта 4 курса
направления 09.06.01 – Информатика и вычислительная техника направленности
«Математическое моделирование, численные методы и комплексы программ»

Судани Хайдера Хуссейна Карима

Научный руководитель
д.ф.-м.н., доцент Абросимов Михаил Борисович

Зав. кафедрой
д.ф.-м.н., доцент Абросимов Михаил Борисович

Саратов 2020

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

В 1993 году F. Harary и J. Hayes предложили основанную на графах модель для исследования отказов связей в технической системе. Эта модель была дальнейшим развитием модели, которую предложил J. Hayes в 1976 году для исследования отказов элементов технической системы. Системе Σ сопоставляется помеченный граф $G(\Sigma)$, вершины которого соответствуют элементам системы Σ, рёбра – связям между элементами. Если связи являются симметричными, то граф будет неориентированным графом. Если все элементы системы и связи являются однотипными, то граф будет без меток. Модель оказалась удобной для разработки отказоустойчивых вычислительных систем, где элементами являются вычислительные узлы, а связями между элементами – информационные каналы. Вычислительными узлами могут быть ядра, процессоры или компьютеры. В такой системе элементы и связи, как правило, являются однотипными. Под *отказом* элемента системы Σ понимается удаление соответствующей ему вершины из графа системы $G(\Sigma)$ и всех связанных с ней рёбер. Под отказом связи в системе Σ понимается удаление соответствующего ребра (или дуги для ориентированного графа) из графа системы $G(\Sigma)$. Много работ посвящено исследованию отказов элементов и несколько меньше работ, в которых исследуются отказы связей. В данной работе мы будет рассматривать только отказы связей.

Система Σ^* называется *рёберной k-отказоустойчивой реализацией* (edge fault tolerance) системы Σ, если отказ любых k связей системы Σ^* приводит к графу, в который можно вложить граф системы Σ. Построение рёберной k-отказоустойчивой реализации системы Σ можно представить себе как введение в неё определенного числа избыточных элементов и связей. При этом предполагается, что в нормальном режиме работы избыточные элементы и связи *маскируются*, а в случае отказа происходит *реконфигурация* системы до исходной структуры. На языке теории графов рёберную k-отказоустойчивую реализацию системы Σ мы будем называть рёберным k-расширением (Р-kР) её графа $G(\Sigma)$. Автор работы является сотрудником Министерства науки и технологий Ирака, где занимается вопросами, связанными с облачными вычислениями и технологиями связи. Тема отказоустойчивости в этих областях является чрезвычайно важной. Исторически вопросы отказоустойчивости вышли на передний план с появлением сложных вычислительных устройств, и первые идеи были реализованы уже в первом поколении ЭВМ. Например, в ЭВМ EDVAC (1949 год) было реализовано два арифметико-логических устройства, результаты которых сравнивались. По мере усложнения вычислительной техники появлялись более сложные методы.

В 1956 году Джон фон Нейман рассматривал, как построить надёжную систему из ненадёжных элементов. Прошло больше 60 лет, и за это время надёжность элементов и связей между ними очень выросла. Например, среднее время наработки на отказ современных процессоров составляет более 100 тысяч часов, обычного жёсткого диска *Seagate Barracuda* приближается к миллиону часов, а для серии *Constellation* составляет 1.4 миллиона часов. Первое вычислительное устройство ENIAC использовало около 18000 вакуумных ламп, и это было самое сложное устройство своего времени, которое было чрезвычайно ненадёжным. Сейчас, по-видимому, самыми сложными вычислительными устройствами являются суперкомпьютеры. С 1993 года начал работать проект TOP500, который составляет рейтинг 500 самых мощных общественно известных вычислительных систем мира. Суперкомпьютер №1 в самом первом рейтинге от июня 1993 года – CM-5/1024

– имел 1024 ядра. В июне 2020 года самым мощным стал японский суперкомпьютер Фугаку с числом ядер 7 299 072. Такое усложнение систем требует при разработке учитывать отказоустойчивость.

L. Bautista-Gomez с соавторами проанализировали отчёты о работе ряда суперкомпьютеров. Анализировались ошибки всех компонент системы: оборудования, программного обеспечения, сети и среды. Интересно, что более половины ошибок приходилось на аппаратное обеспечение, причём среднее время наработки на отказ составляло от 10 до 23 часов. В работе авторы указывают, что, если взять систему из 100 000 современных процессоров со средним временем наработки на отказ несколько лет, то для такой системы среднее время наработки на отказ с учётом только процессоров составит 17.3 дня. Очевидно, что нужно предусмотреть возможность для системы продолжать работу в присутствии отказа элемента или связи между элементами.

В 1971 году A. Avižienis предложил два подхода для повышения надежности вычислительных систем:

1) предотвращение ошибок;
2) отказоустойчивость.

Первое направление связано с уменьшением вероятности возникновения ошибки, то есть в повышении надёжности компонентов системы. Во втором направлении используется введение в систему избыточных структур для придания ей свойств отказоустойчивости. Позднее отказоустойчивость стала рассматриваться как одна из составляющих более общего свойства – гарантоспособности системы.

A. Avižienis определяет *отказоустойчивость* как обеспечение системы способностью противостоять ошибке и возможностью продолжать работу в присутствии этой ошибки. Выделяют два уровня отказоустойчивости:

1) *полная отказоустойчивость* – система продолжает работать в присутствии ошибок без существенной потери функциональных свойств;

2) *амортизация отказов* – система продолжает работать в присутствии ошибок с частичной деградацией функциональных возможностей.

В данной работе будет рассматриваться только полная отказоустойчивость. Мы будем рассматривать только отказы связей и изучать рёберные расширения графов. Модель, в которой исследуются отказы элементов, занимается вершинными расширениями графов. Построить рёберную k-отказоустойчивую реализацию означает добавить к исходной системе определённое количество элементов и связей. Стоимость элементов существенно выше связей, поэтому накладывается ограничение, что дополнительные элементы вводиться не могут, добавляются только дополнительные связи. На языке теории графов это означает, что при построении рёберных k-расширений новые вершины к графу не добавляются, добавляются только рёбра. Другое ограничение, которое можно рассмотреть, – отсутствие кратных рёбер, то есть будут рассматриваться только простые неориентированные графы. В этом случае не для каждой системы может быть построено рёберное k-расширение. Например, для полносвязной системы, в которой все элементы попарно соединены, моделируемой полным графом, нет возможности добавить новые рёбра, поэтому у такой системы не будет рёберного k-расширения ни при каком натуральном k. Задача определения, какие графы имеют рёберное k-расширение при заданном k, представляет отдельный интерес. Если же рёберное k-расширение существует, то представляет интерес найти расширение с минимальным числом дополнительных связей.

Рёберная k-отказоустойчивая реализация Σ^* системы Σ называется *оптимальной*, если система Σ^* содержит столько же элементов, сколько и система Σ, а среди всех рёберных k-отказоустойчивых реализаций с тем же числом элементов система Σ^* имеет наименьшее число связей. На языке теории графов рёберную оптимальную k-отказоустойчивую реализацию системы Σ мы будем называть минимальным рёберным k-расширением (МР-kР) её графа $G(\Sigma)$. Вместо минимальности числа связей могут рассматриваться и другие ограничения, которые по той или иной причине предпочтительны, например, наличие определённой группы автоморфизмов.

В своей первой работе J. Hayes предложил схемы построения минимальных вершинных 1-расширений для цепей и циклов, а также для частного случая помеченного дерева. В следующей работе были получены аналогичные результаты для минимальных рёберных 1-расширений. Большой обзор аналитических результатов можно найти в монографии М.Б. Абросимова «Графовые модели отказоустойчивости». В частности, можно отметить результаты по аналитическому построению вершинных и рёберных k-расширений для различных классов графов, полученные М.Ф. Караваем, А.В. Киреевой, М.А. Кабановым, А.А. Долговым, О.В. Моденовой, М.Б. Абросимовым, F. Harary, J. Hayes, L.-H. Hsu и в ряде других работ.

М.Б. Абросимов доказал, что задача проверки того, что граф является рёберным k-расширением заданного графа, является NP-полной. Это означает, что эффективный поиск минимальных рёберных k-расширений для произвольных графов реализовать, видимо, невозможно. В работе предлагается алгоритм построения минимальных рёберных k-расширений графов, основанный на переборе всех подходящих кандидатов. С помощью программы, реализованной на его основе, были получены минимальные рёберные 1-расширения всех неориентированных графов с числом вершин до 7. Описанный алгоритм сложен для распараллеливания, так как во время поиска необходимо поддерживать список всех найденных рёберных расширений. Алгоритм позволяет строить рёберные расширения для небольших графов. Программная реализация на основе этого алгоритма была использована для построения всех минимальных рёберных 1-расширений графов с числом вершин до 7, а также для некоторых отдельных графов с большим числом вершин. Существуют методы построения различных графов без непосредственной проверки на изоморфизм, и в этом случае можно не хранить все найденные результаты. В основе таких методов лежит идея оставлять очередного кандидата только в том случае, если его форма является канонической. Существуют различные техники сокращения полного перебора кандидатов, которые в иностранной литературе называют методами Рида-Фараджева и МакКея. Хороший обзор современного состояния по этому вопросу можно найти в работе G. Brinkmann, который также является автором нескольких известных генераторов, в которых реализованы методы построения графов без проверки на изоморфизм. В англоязычной литературе эта техника называется *isomorphism rejection*. Алгоритмы генерации без проверки на изоморфизм хорошо подходят для распараллеливания.

Всё вышесказанное определяет актуальность данной работы, а также её цели и задачи.

Цели и задачи. Основная цель данной работы состоит в разработке методов математического моделирования и алгоритмов для решения оптимизационной задачи определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами, а также построения всех неизоморфных минимальных рёберных k-расширений заданного графа без проверки на изоморфизм и с возможностью распараллеливания. Основные задачи работы:

1. Разработать методы математического моделирования для решения задачи определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех минимальных рёберных k-расширений для соответствующего графа системы.
2. Разработать алгоритмы определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех минимальных рёберных k-расширений для соответствующего графа системы.
3. Провести исследование разработанных алгоритмов, оценить возможность их распараллеливания и эффективность. Проверить корректность работы на известных данных.
4. Разработать программный комплекс, реализующий алгоритмы определения минимального числа связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех её минимальных рёберных k-расширений.
5. Провести вычислительные эксперименты по построению минимальных рёберных k-расширений всех неориентированных графов с числом вершин до 10 для нескольких значений k и проанализировать полученные результаты.
6. Провести вычислительные эксперименты по построению минимальных рёберных 1-расширений для отдельных наиболее интересных с практической точки зрения графов: решёток, торов.

Объектом исследования является граф, представляющий модель вычислительной системы.

Предметом исследования являются математические модели и алгоритмы, позволяющие строить для заданного графа все неизоморфные минимальные рёберные k-расширения.

Научная новизна работы заключается в следующем:

1. Предложены методы математического моделирования с использованием техники исключения изоморфных копий для задачи определения минимального количества дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения минимальных рёберных k-расширений для соответствующего графа системы.
2. Разработаны новые численные методы решения задачи определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы.
3. Разработаны алгоритмы построения всех неизоморфных минимальных рёберных k-расширений графа заданной системы с использованием техники исключения изоморфных копий.
4. Реализованы и исследованы параллельные версии разработанных алгоритмов для построения минимальных рёберных k-расширений заданного графа.
5. Создан программный комплекс для построения всех неизоморфных минимальных рёберных k-расширений для заданного графа или списка графов на основе алгоритмов

данной работы и вычисления минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы.

6. Построены минимальные рёберные 1-расширения всех неориентированных графов с числом вершин до 10, а также минимальные рёберные 2-, 3- и 4-расширения для всех неориентированных графов с числом вершин до 9.
7. Построены минимальные рёберные 1-расширения для некоторых интересных решёток, торов и гиперкубов с числом вершин до 20.

Методология и методы исследования. В работе используются методы теории графов, теории алгоритмов, комбинаторной оптимизации, специальные методы разработки алгоритмов генерации структур без проверки на изоморфизм, вычислительных экспериментов и статистического анализа.

Положения и результаты, выносимые на защиту.

1. Предложены методы математического моделирования для решения задачи определения минимального количества дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и нахождения всех неизоморфных минимальных рёберных k-расширений графа соответствующей системы без проверки на изоморфизм.
2. Разработаны алгоритмы определения минимального количества дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и нахождения всех неизоморфных минимальных рёберных k-расширений графа соответствующей системы без проверки на изоморфизм.
3. Реализованы параллельные алгоритмы определения минимального количества дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и нахождения всех неизоморфных минимальных рёберных k-расширений графа соответствующей системы без проверки на изоморфизм.
4. Разработан программный комплекс на основе предложенных методов, позволяющий для поданных на вход графов определять минимальное количество дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и нахождения всех неизоморфных минимальных рёберных k-расширений графа соответствующей системы без проверки на изоморфизм.

Теоретическая и практическая значимость работы. Теоретическая значимость работы обусловлена разработкой алгоритмов для построения всех неизоморфных минимальных рёберных k-расширений графа. В основе разработанных алгоритмов лежат техники исключения изоморфных копий и метод канонических представителей. Практическая значимость работы состоит в разработанном программном комплексе для построения минимальных рёберных k-расширений заданного графа на основе алгоритмов, предложенных в работе. В результате проведённых экспериментов удалось построить минимальные рёберные расширения для некоторых решёток и торов с числом вершин до 20. Для 16-вершинного гиперкуба удалось установить, что известное ранее минимальное рёберное 1-расширение является единственным с точностью до изоморфизма. С помощью разработанного программного комплекса были построены все минимальные рёберные 1-расширения для графов с числом вершин до 9 и выложены в открытый доступ в онлайн-энциклопедии «Мир графов». Реализованный программный комплекс может быть использован на этапе анализа и разработки вычислительных архитектур, для которых необходимо предусмотреть отказоустойчивость связей между элементами.

Личный вклад автора. Все основные результаты, выносимые на защиту, получены автором лично. Автор участвовал в определении цели работы и постановке задач. Автор самостоятельно разрабатывал все программные части совместного программного комплекса, которые относятся к задаче построения рёберных расширений. Все основные алгоритмы из работы были реализованы как отдельное приложение, а также как часть программного комплекса FTConstructor, который позволяет для заданного графа строить как рёберные, так и вершинные расширения. Вклад автора в FTConstructor – все программные модули для построения рёберных расширений. В остальных совместных работах автору также принадлежит ведущая роль в вопросах, связанных с рёберными расширениями графов. Результаты данной работы докладывались автором на различных конференциях и семинарах.

Степень достоверности и апробация результатов исследования. Полученные результаты согласуются с известными экспериментальными и аналитическими результатами. Основные результаты работы докладывались и обсуждались на следующих семинарах и конференциях:

- научный семинар кафедры теоретических основ компьютерной безопасности и криптографии Саратовского национального исследовательского государственного университета имени Н.Г. Чернышевского (Саратов, 2017, 2018, 2019, 2020);
- Международная научная конференция «International Conference on Computing, Communication, Control and Automation» (ICCUBEA-2017) (Пуна, Республика Индия, 2017);
- VIII Международная научная конференция «Компьютерные науки и информационные технологии» памяти А.М. Богомолова (Саратов, 2018);
- Всероссийская конференция «XVIII Сибирская научная школа-семинар с международным участием «Компьютерная безопасность и криптография» – SIBECRYPT'19» (Томск, 2019);
- XV, XVI, XVII, XVIII Белорусско-российская научно-техническая конференция «Технические средства защиты информации» (Минск, Республика Беларусь, 2017, 2018, 2019, 2020);
- XXVII Международная конференция студентов, аспирантов и молодых учёных «Ломоносов» (Москва, 2020).

Соответствие темы диссертации требованиям паспорта специальностей научных работников.

Результаты работы соответствуют пунктам 1, 3, 4 и 5 паспорта специальности 05.13.18 «Математическое моделирование, численные методы и комплексы программ».

Публикации. Результаты работы опубликованы в 15 изданиях, 4 из которых являются изданиями, рекомендованными ВАК Минобрнауки РФ, 3 индексируются в базе WoS и SCOPUS, 1 – свидетельство о регистрации программы для ЭВМ.

Структура работы. Диссертационная работа состоит из введения, 4 глав, заключения, списка сокращений и условных обозначений, списка использованной литературы и приложений. Работа содержит 150 страниц, 29 рисунков, 20 таблиц, библиографический список из 109 наименований и 4 приложения.

Содержание работы

Во введении обосновывается актуальность диссертационной работы, аргументируется научная новизна исследований, определяются цели и задачи, показывается практическая значимость полученных результатов, представляются выносимые на защиту научные положения.

В первой главе приводятся основные понятия теории графов. Рассматривается модель отказоустойчивости связей в вычислительной системе, которая была предложена в 1993 году F. Harary и J. Hayes. Приводится базовый алгоритм построения минимальных рёберных k-расширений.

Граф H называется *рёберным k-расширением* (Р-kР) графа G, если G вкладывается в каждый граф, полученный из H удалением произвольных k рёбер. Р-kР H графа G называется *минимальным* (МР-kР), если выполняются следующие условия:

1. $n(H) = n(G)$.
2. Среди всех МР-kР графа G с тем же числом вершин H имеет минимальное число рёбер.

У графа может быть как несколько расширений, так и может не быть ни одного. Например, полный граф K_n не имеет МР-kР ни при каких натуральных значениях n и k. Расширения графов являются терминологическими эквивалентами для введённых ранее понятий, связанных с отказоустойчивыми реализациями: минимальное рёберное k-расширение (МР-kР) – оптимальная рёберная k-отказоустойчивая реализация (optimal k-edge fault-tolerant);

Известно, что задача проверки того, что граф H является рёберным k-расширением графа G, относится к классу NP-полных. Это означает, что эффективного алгоритма для поиска всех МР-kР, видимо, быть не может. Основная задача данной диссертации состоит в разработке алгоритмов построения для заданного графа G всех его неизоморфных минимальных рёберных k-расширений. За основу был взят базовый алгоритм построения МР-kР, описанный в монографии М.Б. Абросимова.

Алгоритм 1 – Построение всех МР-kР для заданного графа G

1. $m := -1$.
2. $m := m + 1$.
3. Строим все графы, получающиеся из графа G добавлением m дополнительных рёбер.
4. Выбираем среди построенных на шаге 3 графов рёберные k-расширения графа G.
5. Если на шаге 4 были найдены графы, то переходим на шаг 8.
6. Если на шаге 4 был построен полный граф, то граф G не имеет минимальных рёберных k-расширений, и алгоритм завершает работу.
7. Переход на шаг 2.
8. Среди графов, выбранных на шаге 4, оставляем по одному представителю от изоморфных графов. Полученные графы будут являться минимальными рёберными k-расширениями графа G.

Во второй главе рассматриваются все вычислительно сложные подзадачи базового алгоритма и определяются пути его оптимизации. Предлагаются алгоритмы для всех основных функций базового алгоритма, которые затем будут улучшаться. Рассматривается метод канонических представителей и анализируется возможность его использования для построения минимальных рёберных *k*-расширений.

Функция **FIRST_SUPERGRAPH**(***G*, &*H*, *m***) должна возвращать значение TRUE при существовании как минимум одного суперграфа H графа G, в котором на m рёбер больше, чем в G.

Функция **NEXT_SUPERGRAPH**(***G*, &*H*, *m***) должна возвращать значение TRUE при существовании следующего за *H* графа. Если все графы перебраны, то результатом должен быть FALSE.

Описанные функции устанавливают линейный порядок на множестве суперграфов с m дополнительными рёбрами и перебирают графы в соответствии с этим порядком.

Функция **EDGE_EXTENSION** – это проверка на расширение по определению. Для проверки на расширение по определению необходимо уметь проверять вложение графов. Для того, чтобы проверить граф на вложение, необходимо найти отображение, которое является вложением. Это можно сделать путём перебора отображений и проверки, является ли отображение вложением. Отображения в данном случае можно отождествить с перестановками на множестве вершин графа. Количество отображений достаточно велико и растёт экспоненциально с ростом количества вершин в графе, поэтому необходимо использовать способы оптимизации перебора. В данной работе будет использован метод поиска с возвратом (бэктрекинг, backtracking). Заключается данный метод в том, что построение искомого объекта происходит постепенно. На каждом шаге нужно проверять построенную часть, может ли из данной части быть получен хоть один необходимый объект.

Проверка на расширение является одним из самых частых действий. Проверка по определению является вычислительно сложной, так как состоит из ряда проверок вложений – вычислительно сложной задачи. Уменьшить количество проверок на расширение можно за счёт использования леммы, доказанной М.Б. Абросимовым:

Лемма 1. Если минимальная степень вершины графа *G* есть $d > 0$, то его минимальное рёберное *k*-расширение не содержит вершин степени ниже $d + k$.

Данная лемма позволяет сократить количество проверок на расширение. Особенно хорошо она работает на регулярных графах, а также на графах, в которых большое количество вершин минимальной ненулевой степени.

Функция **CONTAINS_ISOMORPH_COPY** отвечает за проверку наличия в списке изоморфных графов. Вычислительная сложность задачи определения изоморфизма двух графов остаётся одной из нерешённых современных проблем: является ли она NP-полной или нет, до сих пор неизвестно.

Функция может быть реализована несколькими способами.

1. Прямая проверка на изоморфизм.
2. Сравнение полных инвариантов.
3. Метод канонических представителей.

В работе рассматриваются все эти способы и обосновывается преимущество метода канонических представителей.

Матричным кодом графа называется последовательность, выписанная из элементов матрицы смежности графа в определённом порядке. Есть несколько вариантов построения такого кода. В работе рассматриваются три кода. *Максимальным матричным кодом* (ММК) называется наибольший матричный код среди всех кодов изоморфных графов.

Максимальный матричный код можно улучшить с точки зрения скорости его построения для графов, которые не являются однородными: если задать определённый порядок степеней вершин (например, по возрастанию), то уменьшится количество изоморфных графов, которые нужно перебирать. Такой код будем называть степенным максимальным матричным кодом – СММК. Третий вид кода, который подробно рассматривается в работе, называется *верхним кодом*: необходимо среди всех изоморфных графов найти максимальную пару (w, c), где w – это последовательность чисел, каждое из которых равно количеству элементов ниже главной диагонали в данной строке матрицы, а c – это код из элементов ниже главной диагонали матрицы смежности графа, выписанный по строкам. Был проведён вычислительный эксперимент с целью сравнения скорости построения кода. В среднем быстрее всего строится степенной максимальный матричный код.

Метод канонических представителей является техникой исключения изоморфизма (isomorphism rejection), не основанной на непосредственной проверке изоморфизма. Один из основоположников этого метода – И.А. Фараджев. Данный метод хорошо описан в книге «Алгоритмические исследования в комбинаторике» под его редакцией, а современное состояние – в статье G. Brinkmann «Isomorphism rejection in structure generation programs».

Метод состоит в том, что в каждом классе изоморфных графов необходимо выбрать одного представителя, который называется *каноническим*. Необходимо иметь способ проверки каноничности графа. Также для того, чтобы метод работал на множестве графов, необходимо, чтобы для каждого класса изоморфизма, представители которого находятся в множестве, канонический представитель данного класса также находился в этом множестве.

В качестве канонического представителя можно выбрать граф с максимальным матричным кодом. В работе показывается, что обычные матричные коды не подходят для нашей задачи. Поэтому для использования метода канонических представителей необходим иной полный инвариант.

В третьей главе рассматривается маршрутный код (или М-код), на основе которого можно реализовать метод канонических представителей. Предлагаются 7 различных алгоритмов для решения задачи построения всех неизоморфных минимальных рёберных k-расширений. Рассматривается зависимость скорости выполнения от представления графа, способы распараллеливания, их положительные и отрицательные стороны.

Для улучшений алгоритма необходимо знать, на какие подзадачи затрачивается больше времени в алгоритме построения МР-kР. Для этого был проведён ряд вычислительных экспериментов. Оказалось, что существенным моментом является форма исходного графа. Наиболее распространённым способом получения графов определенного вида является использование генераторов. Например, чтобы получить все графы с заданным числом вершин, можно воспользоваться генератором *geng* из пакета *nauty and Traces*. Было рассмотрено влияние различных форм графа на скорость выполнения алгоритмов, а также использование леммы 1.

Использованные формы графов:

1. GENG – графы, полученные командой «geng».
2. GENG-L – графы, полученные командой «geng -l».
3. СММК1 – степенной максимальный матричный код с упорядочиванием степеней по возрастанию.
4. СММК2 – степенной максимальный матричный код с упорядочиванием степеней по убыванию.

5. ММК – максимальный матричный код.
6. ВК – верхний код.

Были построены МР-1Р и МР-2Р всех 7-вершинных графов. По результатам получилось, что среднее время, затраченное на один граф, оказалось наименьшим при использовании СММК2, ММК и ВК. Лемма значительно сократила количество проверок на расширение – приблизительно на четверть.

В алгоритме каждый из построенных графов проверяется на расширение, поэтому количество проверок на расширение будет больше, чем количество проверок списка на наличие изоморфных копий. По полученным статистическим данным и для МР-1Р, и для МР-2Р их количество соотносится примерно 19 к 1, если применять лемму.

Среднее время решения задачи исключения изоморфных копий меньше, чем среднее время проверки на расширение. Отсюда следует, что две данные проверки можно поменять местами. Это приведёт к увеличению производительности, однако при построении графов будет расходоваться память на хранение максимальных матричных кодов всех суперграфов, что для больших графов невозможно.

К тому же, расширения при поиске встречается намного реже, чем неизоморфные суперграфы. Это связано с тем, что нахождение первого расширения означает, что поиск завершится на текущем количестве дополнительных рёбер. Из этого следует, что проверок на существование изоморфных копий при первичной проверке на расширение будет меньше, чем проверок на расширение при первичной проверке на существование неизоморфной копии. Поэтому описанные изменения алгоритма необходимо проверять на практике.

Рассмотрим новый вид кода, который будем называть *маршрутным* или *М-кодом*. М-код представляет собой специальный вид матричного кода: для n-вершинного графа задаётся порядок выписывания элементов его матрицы смежности. Максимальный М-код среди всех изоморфных графов будем называть *каноническим маршрутным кодом* (*КМ-кодом*), а граф, соответствующий данному коду, – *каноническим представителем* (*КП*) или просто *каноническим*.

Зададим способ построения М-кода $C_G(H)$ n-вершинного графа H, который основан на матрице смежности n-вершинного графа G:

1. Выписывание элементов происходит в соответствии с порядком обхода элементов: по строкам сверху вниз, обход строки происходит в порядке слева направо, в обходе участвуют только элементы, находящиеся ниже главной диагонали.
2. Выписывание элементов происходит в два этапа.
3. На первом этапе выписываются те элементы матрицы смежности графа H, на соответствующих позициях которым в матрице смежности графа G стоят единицы.
4. На втором этапе выписываются те элементы матрицы смежности графа H, на соответствующих позициях которым в матрице смежности графа G стоят нули.

В зависимости от того, в каком порядке выполняются действия, было реализовано 7 алгоритмов. Названия алгоритмов являются аббревиатурой от порядка основных действий в алгоритме:

- ЛРМ – реализация алгоритма, при которой суперграфы строятся с помощью перебора сочетаний, следом выполняется проверка леммы, после проверка графа на k-расширение, далее происходит исключение изоморфных копий построением и сравнением СММК;

- ЛРК – реализация алгоритма, при которой суперграфы строятся с помощью перебора сочетаний, следом проверка леммы, после проверка графа на k-расширение, далее происходит проверка каноничности М-кода;
- ЛКР – реализация алгоритма, при которой суперграфы строятся с помощью перебора сочетаний, следом выполняется проверка леммы, после проверяется каноничность М-кода, а далее происходит проверка графа на k-расширение;
- ПРМ – реализация алгоритма, при которой суперграфы строятся с помощью оптимизированного перебора сочетаний 2, после проверка графа на k-расширение, далее происходит исключение изоморфных копий построением и сравнением СММК;
- ПРК – реализация алгоритма, при которой суперграфы строятся с помощью оптимизированного перебора сочетаний, после проверка графа на k-расширение, далее проверка каноничности М-кода;
- ПКР – реализация алгоритма, при которой суперграфы строятся с помощью оптимизированного перебора сочетаний, следом проверка каноничности М-кода, а далее проверка графа на k-расширение;
- ДЛР – реализация алгоритма, при которой происходит построение графов с помощью дерева канонических представителей, далее проверяется лемма, следом происходит проверка на k-расширение.

Для исследования затрат времени на проверку каноничности графа и влияния замены построения полного инварианта на каноничность были проведены вычислительные эксперименты. В верхнем коде были посчитаны МР-1Р и МР-2Р всех 7- и 8-вершинных графов с использованием канонического кода. Собранная статистика показывает, что целесообразно использовать подход с проверкой каноничности перед проверкой на расширение. Также было обнаружено, что для МР-1Р всех 7-вершинных графов лемме 1 не удовлетворяет около 24% графов, участвующих в переборе. Для МР-2Р 7-вершинных графов – 44%. Для 8-вершинных графов доля отрицательного результата проверки леммы изменилась до 22% и 45% соответственно.

Лемма в большой степени уменьшает количество вычислительно сложных проверок у тех графов, которые не имеют изолированных вершин, и достаточно много вершин в которых имеют минимальную степень. Особенно эффективно эта лемма работает на регулярных графах.

Рассмотрим построение дерева канонических представителей. Пусть дан граф G, для которого необходимо построить рёберное расширение. Строится дерево $T(G)$ по следующим правилам:

1. Узлами дерева являются М-коды всех суперграфов графа G.
2. Корнем дерева является М-код $C_G(G) = 1\ldots10\ldots0$.
3. Для отличного от корня узла $C_G(H) = 1\ldots1b_1\ldots b_t\underline{\mathbf{1}}0\ldots0$ узлом-предком является М-код $C_G(Q) = 1\ldots1b_1\ldots b_t\underline{\mathbf{0}}0\ldots0$.

Если граф с кодом $C_G(H)$ $1\ldots1b_1\ldots b_t\underline{\mathbf{1}}0\ldots0$ каноничен, то его узел-предок $C_G(Q) = 1\ldots1b_1\ldots b_t\underline{\mathbf{0}}0\ldots0$ также каноничен. Это означает, что канонические коды образуют поддерево $T_C(G)$ дерева $T(G)$.

Оптимизированный перебор позволяет более эффективно строить графы, удовлетворяющие лемме 1, и позволяет уменьшить количество получаемых неканонических графов. Для перебора необходимо составить наборы рёбер, из которых нужно выбрать ненулевое количество элементов. При переборе нужно сначала выбирать элементы из этих групп, а после дополнить их до необходимого количества. Если последовательность нужной длины построена, но для некоторых наборов ещё не набрано нужное количество элементов, то такая последовательность не принимается. В работе рассматриваются оптимизации для наборов двух типов – с учётом на леммы 1 и принципа дерева канонических суперграфов.

В четвертой главе рассматриваются параллельные реализации описанных алгоритмов, проводятся вычислительные эксперименты по построению МР-*k*Р для разных графов.

Для реализации параллельных вычислений была выбрана технология MPI. Вычисления проводились на кластере Поволжского Регионального Центра Новых Информационных Технологий Саратовского государственного университета имени Н.Г. Чернышевского с использованием 40 ядер.

Для проверки работы параллельных алгоритмов программы для k, равного 1, 2, 3 и 4, были посчитаны МР-*k*Р для 8- и 9-вершинных графов.

При построении МР-*k*Р с ростом k возрастает сложность проверки на k-расширение. Доля времени, затрачиваемая на проверку на расширение, увеличивается с ростом k вне зависимости от алгоритма. Также возрастает влияние перебора и проверки леммы. С ростом k время проверки каноничности падает. Это связано с тем, что меняется набор графов, которые проверяются на каноничность. Количество проверок при обходе дерева канонических представителей уменьшается в разы по сравнению с обычным и усовершенствованным перебором, но количество успешных проверок становится больше. Проверка каноничности происходит до проверки леммы, что увеличивает количество проверяемых графов, в том числе канонических.

Были проведены исследования МР-1Р циклов, так как эта тема представляет большой интерес. F. Harary и J. Hayes предложили схему для построения МР-1Р циклов, известны и другие схемы. Проводились вычислительные эксперименты по построению всех МР-1Р для циклов. Лучший на момент написания результат приведён в работе М.Б. Абросимова и С.А. Сухова. Алгоритмами ДЛР, ПРМ и ПРК были также построены МР-1Р циклов. Лучшие результаты показал алгоритм ПКР, с помощью которого удалось найти все МР-1Р для циклов с числом вершин до 22.

Проведённые вычислительные эксперименты показали эффективность алгоритмов в среднем, однако с ростом количества вершин в графе время на его обработку увеличивается. Была предпринята попытка найти ответ на следующий вопрос: можно ли выбрать какой-то один алгоритм, который бы в среднем вёл себя лучше остальных? Были отдельно обработаны графы разных видов: связные и несвязные, двудольные, регулярные и деревья.

Исходя из экспериментальных данных, несвязные графы обрабатываются быстрее алгоритмом ДЛР, а регулярные – ПКР. Для построения расширений деревьев при малом значении k лучше использовать ДЛР, а с ростом k – переходить на алгоритм ПКР. Алгоритмы ЛРК и ЛРМ можно использовать только для построения МР-1Р, так как при $k > 1$ сложность проверки на k-расширение достаточно сильно растёт.

В работе М.Б. Абросимова были получены статистические данные по МР-1Р 4-, 5-, 6 и 7-вершинных графов. В нашей работе были получены данные по 8-, 9- и 10-вершинным графам. Для примера в таблице 1 приведём данные по количеству дополнительных рёбер $ec(G)$.

Таблица 1 – Количество дополнительных рёбер в MP-1P 10-вершинных графов

$ec(G)$	MP-1P	$ec(G)$	MP-1P
0	1	11	3256
1	63	12	1182
2	799	13	516
3	8630	14	234
4	75526	15	111
5	554875	16	50
6	2995525	17	23
7	7060790	18	9
8	1216416	19	5
9	74068	20	2
10	13085	21	1
		Нет	1

Были построены MP-1P некоторых графов с большим количеством вершин. Это торы, решётки и 16-вершинный гиперкуб. Напомним, что n-мерной *решёткой* называется граф, являющийся декартовым произведением n цепей $Pm_1 \times \ldots \times Pm_n$, а n-мерным *тором* называется граф, являющийся декартовым произведением n циклов $Cm_1 \times \ldots \times Cm_n$. Решётки и торы представляют большой интерес с точки зрения архитектуры компьютера. Разработчики и инженеры при обозначении таких архитектур часто полагают, что цикл C_2 изоморфен цепи P_2. Так узел современного суперкомпьютера Фугаку, который состоит из 12 ядер, разработчики характеризуют как трёхмерный тор 2×2×3.

Заметим, что 16-вершинный гиперкуб можно рассматривать как тор 4×4 или решётку 2×2×2×2. В работе F. Harary и J. Hayes приводится MP-1P 16-вершинного гиперкуба. В результате нашего вычислительного эксперимента это расширение было найдено (см. рисунок 1), причём оно является единственным с точностью до изоморфизма. Аналитического доказательства этого утверждения пока неизвестно. Данные по графам, времени построения и количеству MP-1P представлены в таблице 2. Интересными являются случаи решёток 3×4 и 2×2×3. F. Hararay и J. Hayes высказали гипотезу, что минимальным рёберным 1-расширением n-мерной решетки $Pm_1 \times \ldots Pm_n$ является n-мерный тор $Cm_1 \times \ldots Cm_n$. R.-S. Chou и L.-H. Hsu эту гипотезу опровергли и предложили схему построения рёберного 1-расширения для решёток, однако доказательства минимальности для общего случая этой схемы предоставлено не было. Полученные нами экспериментальные результаты согласуются с результатами R.-S. Chou и L.-H. Hsu. Так для решёток 2×5, 4×4 и 2×2×4 MP-1P единственно с точностью до изоморфизма и соответствует схеме построения из их работы. Однако решётки 3×4 и 2×2×3 имеют еще неизоморфные MP-1P. Все MP-1P для решётки 3×4 представлены на рисунке 2. У тора 2×2×3, или что то же самое 3×4, 21 MP-1P с 12 дополнительными рёбрами. Одно из MP-1P изображено на рисунке 3. Этот тор является моделью схемы соединения *Tofu* вычислительных ядер в суперкомпьютерах *Фугаку* и *K computer*.

Таблица 2 – Время обработки некоторых графов

Граф	Кол-во вершин	Время	Кол-во дополнительных рёбер	Кол-во МР-1Р
Решётка 2×5	10	0.1	3	1
Решётка 3×4	12	0.7	5	2
Решётка 4×4	16	3:40.0	6	1
Решётка 2×2×3	12	0.4	6	3
Решётка 2×2×4	16	0.1	4	1
Тор 2×7	14	0.1	7	1
Тор 2×8	16	1.5	8	7
Тор 2×9	18	44.3	9	1
Тор 2×10	20	33:45.2	10	9
Тор 3×4	12	2:18:25.0	12	21
Гиперкуб	16	0.4	8	1

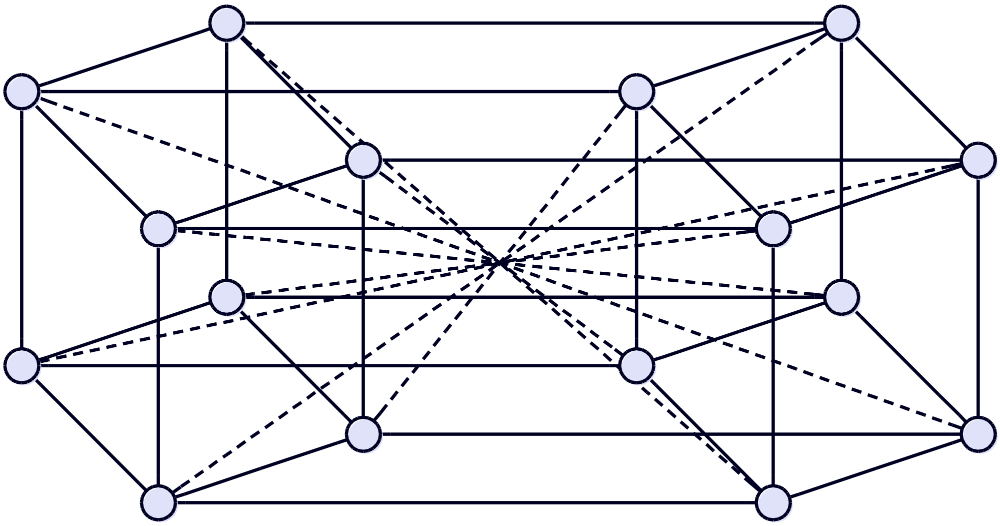

Рисунок 1 – Единственное МР-1Р 16-вершинного гиперкуба

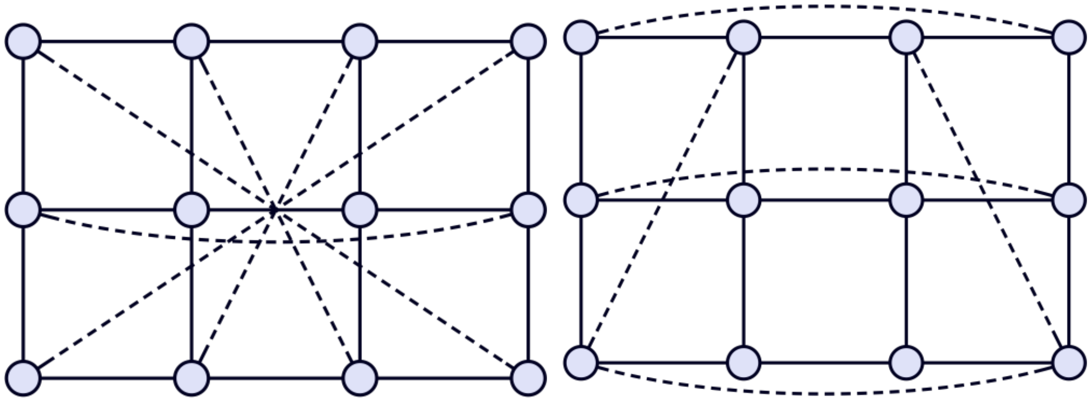

Рисунок 2 – Два МР-1Р решётки 3×4

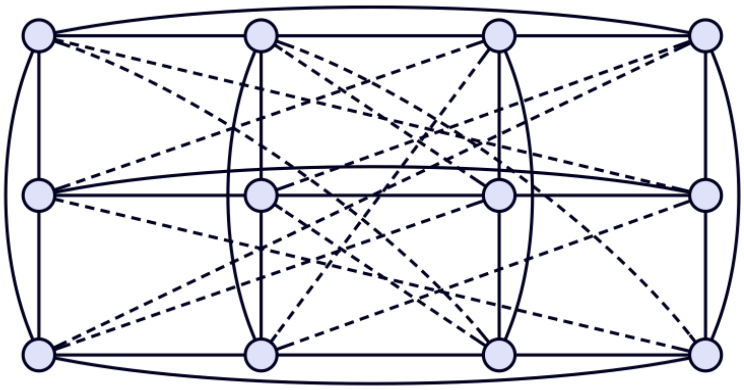

Рисунок 3 – Одно из 27 МР-1Р тора 2×2×3

В заключении приведены основные результаты работы и сделаны выводы.

Основные результаты работы:

1. Предложены методы математического моделирования для решения задачи определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех минимальных рёберных *k*-расширений для соответствующего графа системы.
2. Разработаны алгоритмы определения минимального числа дополнительных связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех минимальных рёберных *k*-расширений для соответствующего графа системы.
3. Исследованы разработанные алгоритмы, оценена возможность их распараллеливания и эффективность. Проверена корректность работы на известных данных. Полученные результаты согласуются с известными данными. Разработанные алгоритмы достаточно существенно зависят от вида графа, который подаётся на вход. Проведены эксперименты, на основе которых можно определить алгоритм, более подходящий в каждом конкретном случае.
4. Разработан программный комплекс, реализующий алгоритмы определения минимального числа связей системы, устойчивой к заданному числу отказов связей между элементами системы, и построения всех её минимальных рёберных *k*-расширений. В этот комплекс были включены 4 лучших из предложенных в работе алгоритмов.

5. Построены минимальные рёберные 1-расширения всех неориентированных графов с числом вершин до 10, а также минимальные рёберные 2-, 3- и 4-расширения для графов с числом вершин до 9.
6. Построены минимальные рёберные 1-расширения для решёток и торов с числом вершин до 20. Отдельно отметим: экспериментальным путём было установлено, что 16-вершинный гиперкуб имеет единственное с точностью до изоморфизма минимальное рёберное 1-расширение.

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ РАБОТЫ

Публикации в изданиях, рекомендованных ВАК Минобрнауки РФ

1. Судани Х.Х. Обеспечение предельной отказоустойчивости параллельных вычислений с интерфейсом передачи сообщений MPI //Современная наука: Серия: Естественные и технические науки. – 2018. – №4 – С. 84–86.
2. Судани Х.Х. Повышение доступности системы безопасности при использовании техники отказоустойчивости // Перспективы науки. – 2018. – № 10(109). – С.125–128.
3. Судани Х.Х.К., Абросимов М.Б. Отказоустойчивость и безопасность доступа в облачных вычислениях // Южно-Сибирский научный вестник. – 2019. – № 2(26). – С. 55–59.
4. Судани Х.Х.К. Рёберно-отказоустойчивые расширения 8-, 9- и 10-вершинных графов // International Journal of Open Information Technologies. – 2020. – Vol. 8, № 8. – P. 46–50.

Публикации в изданиях, индексируемых SCOPUS и Web of Science

5. Абросимов М.Б., Судани Х.Х.К., Лобов А.А. Построение минимальных рёберных расширений графа без проверки на изоморфизм // Изв. Сарат. ун-та. Нов. сер. Сер. Математика. Механика. Информатика. – 2020. – Т. 20, вып. 1. – С. 105–115.
6. Камил И.А.К., Судани Х.Х.К., Лобов А.А., Абросимов М.Б. Построение всех неизоморфных суперграфов без проверки на изоморфизм // Прикладная дискретная математика. – 2020. – № 48. – С.82–92.
7. Abrosimov M., Sudani K., Mahajan H. Fault Tolerance to Balance for Messaging Layers in Communication Society // 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), PUNE, India. – 2017. – P. 423-428.

Объекты интеллектуальной собственности

8. Абросимов М.Б., Камил И.А.К., Судани Х.Х.К., Лобов А.А. Построение оптимальных отказоустойчивых реализаций графов FTConstructor // Свид. о гос. регистрации программы для ЭВМ № 2020614773. Зарегистрирована в Реестре программ для ЭВМ 24.04.2020.

Публикации в других изданиях

9. Камил И.А., Судани Х.Х, Абросимов М.Б. К вопросу о параллельных алгоритмах построения минимальных вершинных и рёберных 1-расширений графов // Компьютерные науки и информационные технологии : Материалы Междунар. науч. конф. – Саратов : Издат. центр «Наука», 2018. – С. 173–176.

10. Камил И.А.К., Судани Х.Х.К., Лобов А.А., Абросимов М.Б. Построение минимальных расширений графа методом канонических представителей // Прикладная дискретная математика. Приложение. – 2019. – № 12. – С. 179–182.
11. Судани Х.Х.К., Абросимов М.Б. Киберугрозы и отказоустойчивость // Технические средства защиты информации: Тезисы докладов XVI Белорусско-российской научно-технической конференции, 5 июня 2018 г., Минск. Минск: БГУИР. – 2018. – С. 90.
12. Судани Х.Х.К., Абросимов М.Б. Безопасность и надежная среда в системах обеспечения отказоустойчивости // Технические средства защиты информации: Тезисы докладов XVII Белорусско-российской научно-технической конференции, 11 июня 2019 г., Минск. Минск: БГУИР. – 2019. – С. 68.
13. Судани Х.Х.К., Абросимов М.Б. Безопасность и техники отказоустойчивости // Технические средства защиты информации: Тезисы докладов XVIII Белорусско-российской научно-технической конференции, 9 июня 2020 г., Минск. Минск: БГУИР. – 2020. – С. 74.
14. Судани Х.Х.К. К вопросу об использовании техники исключения изоморфизма для построения оптимальных рёберно-отказоустойчивых расширений графов // Материалы Международного молодежного научного форума «ЛОМОНОСОВ-2020» [Электронный ресурс] / Отв.ред. И.А. Алешковский, А.В. Андриянов, Е.А. Антипов. – Электрон. текстовые дан. (1500 Мб.) – М.: МАКС Пресс, 2020. – Режим доступа: https://lomonosov-msu.ru/archive/Lomonosov_2020/index.htm, свободный – Материалы Международного молодежного научного форума «ЛОМОНОСОВ-2020».
15. Sudani H.H., Abrosimov M.B. Balance and security for messaging layers // Технические средства защиты информации: Тезисы докладов XV Белорусско-российской научно-технической конференции, 6 июня 2017 г. Минск. Минск: БГУИР. – 2017. – С. 77.