

Федеральное государственное бюджетное образовательное учреждение высшего образования «Омский государственный технический университет»

На правах рукописи



ЮДИНА Мария Николаевна

**МЕТОДЫ И АЛГОРИТМЫ УСКОРЕННОГО РАСЧЕТА ЧАСТОТ
ВСТРЕЧАЕМОСТИ СЕТЕВЫХ МОТИВОВ В БОЛЬШИХ
СЛУЧАЙНЫХ ГРАФАХ**

Специальность: 05.13.18 – Математическое моделирование, численные методы и комплексы программ

ДИССЕРТАЦИЯ

на соискание ученой степени кандидата технических наук

Научный руководитель:
доктор технических наук, доцент
ЗАДОРОВНИЙ Владимир Николаевич

Омск 2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
ГЛАВА 1 ИССЛЕДОВАНИЕ БОЛЬШИХ СЕТЕЙ	10
1.1 Основные обозначения и соглашения	11
1.2 Структурные характеристики графов	15
1.3 Графы сетей интернета	23
1.4 Графы молекулярных сетей клетки	33
1.5 Случайные графы	40
1.6 Выводы по первой главе	43
ГЛАВА 2 МЕТОД СЛУЧАЙНОГО ВЫБОРА ОСТОВНЫХ ДЕРЕВЬЕВ	45
2.1 Статистические методы расчета	45
2.2 Метод случайного выбора остовных деревьев	63
2.3 Статистические эксперименты по двум различным остовным деревьям	64
2.4 Статистические эксперименты по g остовным деревьям	71
2.5 Сравнение статистических методов по показателям качества	73
2.6 Выводы по второй главе	74
ГЛАВА 3 РАЗРАБОТКА АЛГОРИТМОВ И ИХ АНАЛИЗ	76
3.1 Структура хранения графа	76
3.2 Случайный выбор остовных деревьев типа «4-ветвление»	78
3.3 Случайный выбор остовных деревьев типа «4-путь»	84
3.4 Определение класса изоморфизма графа	90
3.5 Алгоритмы программ	91
3.6 Скорость и точность расчетов	97
3.7 Рандомизация графа и расчет статистической значимости подграфов	101
3.8 Выводы по третьей главе	107
ГЛАВА 4 КОМПЛЕКС ПРОГРАММ	109
4.1 Библиотека классов SocAndBioNetworksAnalysis	109
4.2 Библиотека классов MotifsLib и программа MFSView	111
4.3 Система агентного моделирования Simbigraph 2	114
4.4 Выводы по четвертой главе	116
ЗАКЛЮЧЕНИЕ	117
СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ	120
СПИСОК ЛИТЕРАТУРЫ	121
Приложение А Регистрация программ и программных библиотек	135
Приложение Б Акты внедрения	141

ВВЕДЕНИЕ

Актуальность темы исследования. Исследованию больших сетей в последние два десятилетия уделяется пристальное внимание. Одной из таких больших сетей является «сеть сетей» интернет. Интернет можно представить в виде сети автономных систем, содержащей всего несколько десятков тысяч узлов. Также интернет можно представить как сеть, связывающую сотни миллионов маршрутизаторов по всему миру. К сетям интернета относят сеть ссылок веб-страниц, социальные сети и т.д. Анализ сетей интернета является важным для задач обеспечения устойчивости и безопасности функционирования интернета на уровне государства. Анализ сетей интернета важен и на уровне отдельных организаций для задач сетевого маркетинга, внедрения новых концепций и технологий, таких как «Индустрия 4.0», «интернет вещей». Другими примерами больших сетей являются молекулярные сети клетки, сети химических взаимодействий, сети передачи электроэнергии, транспортные сети и т.д. Изучением общих свойств больших сетей, которые, как правило, представляются в виде графов, занимается новая отрасль науки – «Наука о сетях» (Network Science). В рамках «Науки о сетях» возникли новые подходы к изучению структурных свойств сетей. Один из таких подходов заключается в выявлении так называемых сетевых мотивов. Концепция сетевых мотивов позволяет получить статистическую картину взаимодействий между ближайшим окружением узлов сети. Анализ больших сетей на основе выявления сетевых мотивов привел к открытию новых белковых взаимодействий, новых способов диагностирования рака, новых подходов к оценке надежности сетей, выявлению «скрытых особенностей» этих сетей.

Степень разработанности темы исследования.

Сетевые мотивы на k -вершинах (k -мотивы) выявляются при сравнении частот встречаемости связных подграфов, относящихся к различным классам изоморфизма, в исследуемом графе сети и в его рандомизированных версиях. Те подграфы, которые встречаются в исследуемом графе чаще, называются сетевыми мотивами. Сначала для графа геномной регуляторной сети бактерии кишечной палочки, а затем и для графов других сетей, Р. Майло с соавторами выявили сетевые мотивы, связав их с природой

исследуемых сетей. В последние два десятилетия разработано большое количество методов, алгоритмов и программ для выявления сетевых мотивов. Наиболее известными учеными, предложившими новые методы, алгоритмы и программы для решения этой задачи являются Ф. Шрайбер (Mavisto, 2005), С. Верник (Fanmod, 2006), Ж. Чен (NeMoFinder, 2006), З. Р. Кашани (Kavosh, 2009), Л.А. Мейра (AccMotif, 2013), И. Ю. Смолий (веб-сервис MotifNet, 2017), А. Пинар (ESCAPE, 2017) и другие.

Для больших графов выявление уже 4-мотивов занимает огромное количество времени. Поэтому стали разрабатываться методы, позволяющие получать статистические оценки для частот встречаемости подграфов. Среди исследователей, внесших большой вклад в разработку методов получения статистических оценок для частот встречаемости подграфов и выявления сетевых мотивов, следует выделить С. Верника (Fanmod, 2006), М. Бруена (GUISE, 2012), М. Рахмана (Graft, 2014), П. Вонга (MOSS-5, 2018).

В нашей стране исследованием структурных свойств больших сетей и их моделей занимаются Райгородский А. М., Мигов Д. А. [16], Прохоренкова Л. А. [21], Жуковский М. Е., Родионов И. В., Губанов Д. А., Чхартишвили А. Г., Дробышевский М. Д. [5], Буркин А. В. [2], Самосват Е. А. [22].

Объектом исследования являются большие сети, в том числе сети интернета и молекулярные сети клетки.

Предметом исследования являются методы вычисления статистических оценок для частот встречаемости связных подграфов и методы выявления сетевых мотивов.

Целью исследования является разработка методов, алгоритмов и программ для выявления сетевых мотивов в графах больших сетей.

Задачи исследования:

1) Анализ состоятельности, эффективности и несмещенности получаемых статистических оценок для частот встречаемости подграфов в известных статистических методах расчета.

2) Разработка метода получения статистических оценок для частот встречаемости подграфов.

3) Разработка алгоритмов и программ для выявления 3- и 4-мотивов.

Положения и результаты, выносимые на защиту

1. Метод случайного выбора остовных деревьев, позволяющий получать состоятельные, несмещенные и эффективные статистические оценки для частот встречаемости связных подграфов на заданном числе вершин.
2. Ускоренные алгоритмы случайного выбора остовных деревьев типа «4-ветвление» и типа «4-путь».
3. Показатель R' статистической значимости подграфов для выявления сетевых мотивов по статистическим оценкам частот встречаемости подграфов.
4. Алгоритм выявления сетевых мотивов на четырех вершинах.

Научная новизна работы

1. Метод случайного выбора остовных деревьев является развитием других известных методов «случайного выбора пути» (П. Рибейро с соавторами, 2019). Методы «случайного выбора пути» позволяют получать статистические оценки для частот встречаемости подграфов на трех (С. Сишадри с соавторами, 2013), четырех (М. Джа с соавторами, 2015) и пяти вершинах (П. Вонг с соавторами, 2018) в неориентированных графах. Разработан метод получения статистических оценок для частот встречаемости подграфов на k вершинах не только в неориентированных графах, но и в ориентированных, и в смешанных графах. При этом итоговые оценки частот встречаемости подграфов определяются как линейные комбинации соответствующих частных оценок. Коэффициенты линейных комбинаций рассчитываются по полученным в диссертации формулам, обеспечивающим получение эффективной итоговой оценки.
2. Ускоренные алгоритмы случайного выбора остовных деревьев отличаются от алгоритма случайного выбора «3-пути» (С. Сишадри с соавторами, 2013) и «4-пути» (М. Джа с соавторами, 2015) использованием метода стратификации (расслоения) при проведении статистических испытаний. Это позволяет ускорить расчет и уменьшить объем дополнительной памяти при проведении расчетов.
3. Введенный в диссертации показатель значимости R' является обобщением известного показателя R , используемого при точных расчетах частот встречаемости.

4. Предложенный алгоритм выявления сетевых мотивов реализует метод случайного выбора остовных деревьев. При получении статистических оценок для частот встречаемости подграфов на четырех вершинах также оценивается погрешность расчетов. При этом возможно использование двух режимов работы:

- 1) минимизация абсолютной погрешности при получении статистических оценок для частот встречаемости подграфов;
- 2) минимизация относительной погрешности при получении статистических оценок для частот встречаемости подграфов.

Теоретическая и практическая значимость работы

1. Разработан метод случайного выбора остовных деревьев, позволяющий получать в отличие от других методов «случайного выбора пути» состоятельные, несмещенные и эффективные статистические оценки для частот встречаемости подграфов в неориентированных, ориентированных и смешанных графах.

2. Реализация разработанных алгоритмов при расчетах статистических оценок для частот встречаемости связных подграфов на четырех вершинах позволила ускорить расчет по сравнению с другими известными статистическими методами расчета. Использование параллельных вычислений позволило получить большее ускорение.

3. Предложенный показатель R' статистической значимости подграфов позволяет статистически достоверно выявлять сетевые мотивы даже по двум статистическим расчетам частот встречаемости подграфов: одного – в исследуемом графе сети, другого – в его рандомизированной версии.

4. Алгоритм выявления сетевых мотивов на четырех вершинах позволяет выявлять сетевые мотивы в разы быстрее, чем при использовании популярного алгоритма, реализованного в таких программах как FANMOD, MotifNet, igraph.

Результаты диссертационной работы используются:

– в учебном процессе на факультете информационных технологий и компьютерных систем Омского государственного технического университета (ОмГТУ) при проведении практических и лабораторных занятий, выполнении курсовых и выпускных квалификационных работ;

- для разработки эффективных стратегий информационной деятельности ОмГТУ в социальной сети «ВКонтакте»;
- для выработки эффективной структуры ссылок веб-страниц информационного портала АН ПОО «МАНО»;
- для выполнения фундаментальных научных исследований ИППУ СО РАН с использованием визуализации сетей химических реакций в виде графа, с целью дальнейшего анализа его структуры.

Методами исследования поставленных в диссертации задач являются методы «Науки о сетях», методы теории вероятностей и метод Монте-Карло.

Степень достоверности полученных результатов. Достоверность полученных результатов обоснована корректным применением метода Монте-Карло и подтверждается анализом точности выполненных расчетов.

Апробация работы. Основные результаты диссертационной работы доложены лично автором диссертации на следующих научных мероприятиях.

– Всероссийские конференции по имитационному моделированию и его применению в науке и промышленности ИММОД, темы докладов: «Модуль анализа частот встречаемости типовых подграфов в системе агентного моделирования Simbigraph» (ИММОД-2017, г. Санкт-Петербург) и «Агентное моделирование и случайные графы» (ИММОД-2019, г. Екатеринбург).

– Сибирский симпозиум «Наука о данных и инженерия данных» (г. Новосибирск, Академгородок, апрель 2017), тема доклада: «Calculation of number of network motifs using random sampling of frames method».

– Международная конференция «Прикладная математика и фундаментальная информатика» (г. Омск, 2017), тема доклада: «Использование сложных приращений в графах предпочтительного связывания».

– Международные научные конференции «Математическое и компьютерное моделирование» (Омск, 2017, 2018), темы докладов: 1) «Подсчет частот встречаемости типовых подграфов в направленных сетях», 2) «Анализ микровзаимодействий в сложных сетях на основе статистического подхода».

– Международные IEEE-конференции «Динамика систем, механизмов и машин» (Омск, 2014, 2016, 2017, 2018, 2019), темы докладов: 1) «Четыре задачи ускорения вычислений за счет разбиения на подмножества элементов»; 2) «Программа для сбора информации о взаимосвязях пользователей социальной сети ВКонтакте»; 3) «Graphs with complex stochastic increments»; 4) «Комбинированный метод расчета частот встречаемости сетевых мотивов»; 5) Preferential attachment graphs as agent interaction structure.

– Международная конференция «Проблемы машиноведения» (г. Омск, 2019), темы докладов: «Assessment of accuracy in calculations of network motif concentration by Rand-ESU algorithm».

– Международная конференция «Математика в современном мире» (г. Новосибирск, Академгородок, 2017), тема доклада: «Расчет частот встречаемости сетевых мотивов методом случайной выборки каркасов».

– Международная IEEE-конференции SIBCON (Казахстан, г. Астана, 2017), тема доклада: «Калибровка графов предпочтительного связывания».

– Международная конференция по графам, сетям и их приложениям (г. Москва, 2019), тема доклада: «Mixed random sampling of frames method for counting number of motifs».

– Расширенное заседание лаборатории «Комбинаторных и вычислительных методов алгебры и логики» (заведующий лабораторией В. Н. Ремесленников) в ФГБУН Институт математики им. С. Л. Соболева СО РАН (г. Омск, 2019) с представлением результатов диссертационного исследования.

– Онлайн-семинар «Теория управления организационными системами» лаборатории № 57 ИПУ СО РАН (г. Москва, 2020) с представлением результатов диссертационного исследования.

Также результаты докладывались на семинарах научно-исследовательской лаборатории ОмГТУ «Имитационное моделирование, системный анализ и информационные технологии», ежегодных (2014–2018 гг.) Всероссийских конференциях «Информационные технологии и автоматизация управления», проводимых кафедрой АСОИУ ОмГТУ и ежегодной Всероссийской с

международным участием научно-практической конференции «Системы управления, информационные технологии и математическое моделирование».

Публикации. По теме диссертации опубликовано 20 статей, в том числе 5 статей в журналах, включенных в Перечень ВАК рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук, и 5 статей, проиндексированных в Scopus.

Личное участие автора. Представленные в диссертации и выносимые на защиту метод случайного выбора остовных деревьев, ускоренные алгоритмы случайного выбора остовных деревьев типа «4-ветвление» и типа «4-путь», показатель R' статистической значимости подграфов для выявления сетевых мотивов по их статистическим оценкам и ускоренный алгоритм выявления сетевых мотивов на четырех вершинах принадлежат лично автору диссертации.

Структура и объем работы. Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы из 136 наименований и двух приложений. Общий объем работы составляет 145 страниц.

ГЛАВА 1 ИССЛЕДОВАНИЕ БОЛЬШИХ СЕТЕЙ

Развитие информационных технологий, с одной стороны, привело к возникновению разнообразных больших сетей, включая всемирную сеть веб-страниц, социальные сети (например, ВКонтакте, Twitter, Facebook), сети передачи данных и другие сети, которые сейчас объединяются под общим названием – информационно-коммуникационная сеть интернета. С другой стороны, развитие информационных технологий позволило аккумулировать данные о структуре взаимодействий не только в интернете, но и в геномных сетях (проект «Геном человека», проект «BioGrid» и др.), в сетях белковых взаимодействий (проект «Протеом человека») и т.д. Таким образом, появилась потребность формулировать и решать актуальные для современного сетевого общества задачи на языке теории графов (математическом представлении сетей). От решения этих задач зависит скорость научно-технического и социального прогресса, включающего разработку новых технологий в медицине (составление карт взаимодействия белков и генов пациентов для назначения лечения), в промышленности («Индустрия 4.0» и «Интернет вещей»), в сфере обслуживания («Электронное правительство») и т.д. На уровне безопасности отдельных стран для сетей интернета является актуальной задача определения устойчивости сетей к случайным отказам и хакерским атакам («Суверенный интернет»). Актуальной является и задача контроля «сетевого контента» для предотвращения нежелательных социальных явлений, например, пропаганды идей экстремизма и терроризма и т.д. Эти задачи сейчас, как правило, решаются методами «Науки о сетях» (Network Science).

В первой главе рассматриваются структурные характеристики исследуемых в работе графов сетей интернета (десять различных графов). Более подробно описан граф подсети социальной сети ВКонтакте, узлами которой являются аккаунты, владельцы которых указали местом учебы ОмГТУ (этот граф сети использовался для разработки эффективных стратегий информационного освещения деятельности ОмГТУ в социальной сети ВКонтакте, Приложение Б).

Также подробно описан граф сети ссылок веб-страниц (этот граф использовался для оптимизации ссылок на сторонние ресурсы при разработке сайта АН ППО «МАНО», Приложение Б).

Кроме исследований сетей интернета в главе приводится анализ некоторых молекулярных сетей клетки. Молекулярные сети клетки и сети химических взаимодействий (акт внедрения в ИППУ СО РАН, представленный в Приложении Б), наряду с сетями интернета, часто становятся объектом исследования в «Науке о сетях». Анализ молекулярных сетей клетки позволяет расширить представление о сфере применения исследуемых в диссертации методов, алгоритмов и программ.

В результате выполненного в главе аналитического обзора ставится основная задача диссертации – задача расчета частот встречаемости подграфов, относящихся к различным классам изоморфизма.

1.1 Основные обозначения и соглашения

Объектом исследования являются большие сети, состоящие из узлов и связей между ними. Математически сеть представляет собой граф G , который определяется через множество вершин V и множество ребер E между этими вершинами: $G = (V, E)$. При этом, как правило, устанавливается взаимно-однозначное соответствие узлов сети и вершин графа, связей сети и ребер графа.

Если связь между узлами симметричная, то ребро, связывающее соответствующие вершины, будем называть ненаправленным. Граф, содержащий только ненаправленные ребра, будем называть неориентированным графом. Если связь несимметричная, то ребро будет направленным. Граф, в котором некоторые ребра могут быть направленными, а некоторые ненаправленными будем называть смешанным графом.

Ребра, имеющие общую вершину, называются смежными. Каждое ребро инцидентно паре вершин – концевым вершинам этого ребра. Направленное ребро (u, v) инцидентно вершинам u и v , при этом говорят, что первая указанная

вершина u – это начальная вершина направленного ребра, а вторая указанная вершина v – конечная.

Если ребру инцидентна единственная вершина, то такое ребро называют петлей. Два ребра, инцидентные одной и той же паре вершин называются кратными. Граф, в котором нет петель и кратных ребер, называют простым графом. В диссертации рассматриваются только простые графы. Если граф исследуемой сети содержит петли и кратные ребра, то все петли удаляются, а из кратных ребер всегда оставляется одно, остальные удаляются.

Для вершины $v \in V$ обозначение $Nei(v)$ определяет окрестность вершины v : $Nei(v) = \{u \in V \mid (v, u) \in E\}$, т.е. множество вершин, связанных с вершиной v ребром. Для вершин ориентированных и смешанных графов существуют также понятия полуокрестность исхода и полуокрестность захода. В полуокрестность исхода вершины v входят все конечные вершины направленных ребер, начальной вершиной которых является вершина v , а также все вершины, связанные ненаправленным ребром с v . В полуокрестность захода вершины v входят все начальные вершины направленных ребер, конечной вершиной которых является вершина v , а также все вершины, связанные с v ненаправленным ребром.

Для множества вершин $v \in V'$ окрестность $Nei(V')$ определяется по формуле $Nei(V') = \bigcup_{v \in V'} Nei(v) \setminus V'$.

Степень связности $\deg(v)$ вершины v определяется количеством инцидентных вершине v ребер. Полустепень исхода вершины определяется количеством вершин, входящих в полуокрестность исхода вершины v , а полустепень захода вершины v определяется количеством вершин, входящих в полуокрестность захода вершины v .

Последовательность ребер графа называется путем длины n , если любые два рядом стоящие в этой последовательности ребра – смежные. Неориентированный граф называется связным, если между любыми его вершинами существует путь. Связность смешанного графа определяется также как и для неориентированного,

только без учета направления ребер (это соответствует понятию «слабой связности» вершин в теории графов).

Подграф $G' = (V', E')$ графа $G = (V, E)$ имеет в качестве вершин подмножество $V' \subseteq V$, а в качестве ребер – подмножество ребер $E' \subseteq E$, которые инцидентны только вершинам из V' . В работе будут рассматриваться только связные подграфы. Остовное дерево – связный подграф, в который входят все его k вершин, а также $k - 1$ ребро.

Графы $G = (V, E)$ и $G' = (V', E')$ называются изоморфными, если существует взаимно однозначное отображение $\Phi: V \rightarrow V'$, которое определяет взаимно однозначное отображение ребер $E \rightarrow E'$ друг на друга, то есть $(u, v) \in E \Leftrightarrow \{\Phi(u), \Phi(v)\} \in E'$. Множество графов, изоморфных друг другу, называется классом изоморфизма графов. На Рисунке 1.1 представлено множество подграфов на четырех вершинах в неориентированном графе, относящихся к разным классам изоморфизма. В диссертации для краткости изложения под «расчетом частот встречаемости подграфов» подразумевается «расчет частот встречаемости подграфов, относящихся к разным классам изоморфизма». При этом рассчитывается либо абсолютная частота встречаемости подграфа, которая определяет, сколько раз подграф заданного класса изоморфизма встречается в графе, либо относительная частота встречаемости. Относительная частота встречаемости подграфа рассчитывается как отношение абсолютной частоты встречаемости подграфа к общему числу подграфов на заданном числе вершин.

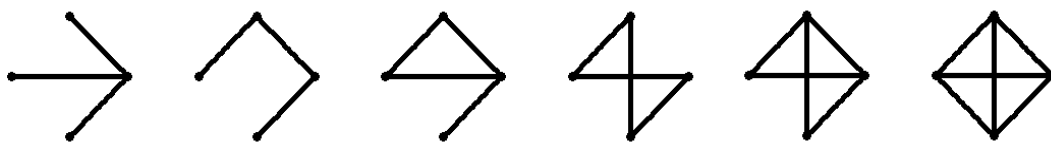


Рисунок 1.1 – Подграфы, относящиеся к разным классам изоморфизма, на четырех связных вершинах в неориентированном графе

Сравнение программ (сторонних и разработанных в рамках диссертации) по времени выполнения выполняется экспериментально и теоретически.

Все замеры времени выполняются на одном и том же оборудовании – моноблоке HP Z1 Workstation с четырехядерным процессором Intel Xeon E3-1245, тактовой частотой 3,3 ГГц и объемом ОЗУ 8 ГБ.

Время выполнения программ оценивается на основе подхода, используемого рядом исследователей в области информатики и программирования, в том числе Д. Кнута [14]. В соответствии с этим подходом время выполнения программы определяется стоимостью и частотой выполнения операций. При этом стоимость выполнения зависит от компьютера и программного обеспечения, а частота зависит от используемого алгоритма и входных данных. Анализ алгоритмов, таким образом, связан с подсчетом частот выполнения операций [14, 23], при этом рассчитываются частоты выполнения лишь базовых (как правило, самых трудоемких) операций при заданных входных данных.

Для оценивания верхней границы частот выполнения операций (времени выполнения) разработанных алгоритмов используется символ большого O . Так, для двух функций $f(N)$ и $g(N)$, говорят, что $f(N)$ имеет порядок $O(g(N))$, если существуют такие значения N и c , что для всех $N > N_0$ выполняется $|f(N)| = c \cdot g(N)$. По возможности, в диссертационной работе частоты выполнения базовых операций (например, количество обращений к основной структуре данных) рассчитываются точно (а не используется верхняя оценка). При этом чтобы игнорировать относительно мелкие слагаемые, в выражениях для упрощения расчета используется знак « $\sim$ ». Запись $g(N) \sim f(N)$ для функций $g(N)$ и $f(N)$ означает, что $g(N)/f(N) \rightarrow 1$, при $N \rightarrow \infty$.

Например, для расчета в неориентированном графе, содержащем N вершин, числа всех «треугольников» (подграфов на трех вершинах, в которых все пары вершин связаны ребрами) необходимо проверить $C_N^3 = N(N-1)(N-2)/6$ троек вершин. Соответственно, при хранении графа в матрице смежности M число обращений к ней будет равно $N(N-1)(N-2)/2$, т.к. для каждой тройки вершин следует еще проверить существование ребер между вершинами. А при аппроксимации старшим членом число обращений к матрице смежности M будет оценено как $\sim N^3/2$. В то же время верхняя граница числа обращений к матрице

смежности имеет порядок $O(N^3)$. Для рассмотренной задачи подсчета всех «треугольников» в неориентированном графе описание алгоритма приведено на Рисунке 1.2а), описание схемы программы на Рисунке 1.2б). Для описания алгоритмов исследуемых методов в диссертации используется псевдокод. Для описания алгоритмов программ в диссертации используются схемы алгоритмов.

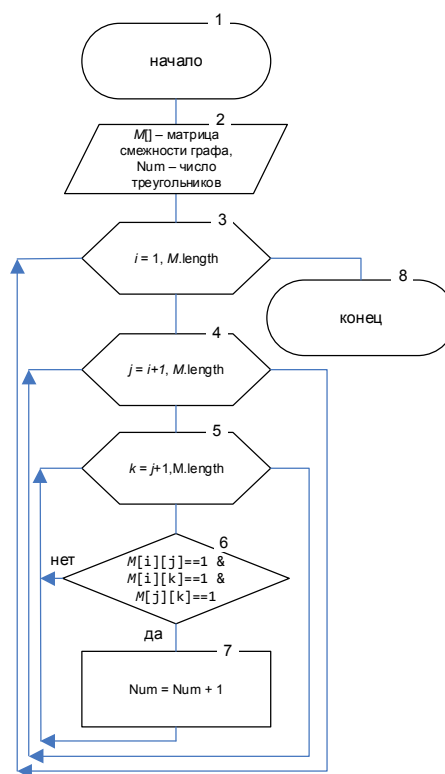


Рисунок 1.2 – Схема алгоритма программы подсчета «треугольников» методом полного перебора

1.2 Структурные характеристики графов

В данном разделе рассмотрены структурные характеристики графов исследуемых сетей. Эти характеристики разделены на три множества – глобальные характеристики, локальные характеристики и сетевые мотивы.

1.2.1 Глобальные характеристики

К глобальным характеристикам графов исследуемых сетей отнесем следующие интегральные показатели, характеризующие граф в целом.

а) Средняя степень связности $\langle k \rangle$ вершин – характеристика, определяющая среднее арифметическое степеней связности $\deg(v_i)$ вершин v_i графа $G(V, E)$:

$$\langle k \rangle = \frac{1}{|V|} \sum_{i=1}^{|V|} \deg(v_i). \quad (1.1)$$

б) Среднее расстояние $\langle l \rangle$ между вершинами графа – среднее арифметическое длин кратчайших путей между парами вершин. Характеристика измеряется числом ребер между вершинами:

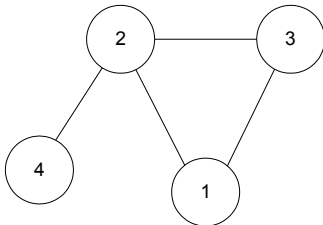
$$\langle l \rangle = \frac{1}{\sum_{i=1}^{|V|} \sum_{j=1}^{|V|} [d(i, j) < \infty]} \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} d(i, j) [d(i, j) < \infty], \quad (1.2)$$

где $d(i, j)$ – длина кратчайшего пути от вершины i до вершины j .

Для компактной записи формулы (1.2) использована идея Айверсона, применяемая Кнудом в его работе [15]. Идея Айверсона состоит в том, чтобы заключать истинное или ложное утверждение в квадратные скобки и считать при этом, что значение полученного выражения равно единице, если утверждение в квадратных скобках истинно, и равно нулю, если утверждение ложно. Запись $d(i, j) < \infty$ означает, что существует путь из вершины i в вершину j .

в) Диаметр графа – наибольшая из длин кратчайших путей. Для телекоммуникационных сетей от этого показателя зависит максимальное время отклика в сети.

г) Коэффициент кластеризации C' рассчитывается как отношение утроенного числа n_{Δ} «треугольников» в графе к числу n_V «вилкок», т.е. числу путей длиной два ребра:



$$C' = \frac{3 \cdot n_{\Delta}}{n_V}. \quad (1.3)$$

Рисунок 1.3 – Простой неориентированный граф

Проиллюстрируем эти формулы на небольшом примере. На Рисунке 1.3 изображен граф, для которого число треугольников $n_{\Delta} = 1$, а число путей длиной два ребра $n_V = 5$, глобальный коэффициент кластеризации по формуле (1.3): $C' = 3 \cdot 1 / 5 = 0,6$.

1.2.2 Локальные характеристики

К локальным характеристикам графа отнесем следующие меры важности [31] вершин графа.

а) Степень близости к другим вершинам (англ. closeness centrality) $C_C(v)$ – величина, обратная сумме кратчайших путей $d(v, w_i)$ от вершины v до других вершин w_i графа $G(V, E)$:

$$C_C(v) = \left(\sum_{i=1}^{|V|} [d(v, w_i) < \infty] \right) / \sum_{i=1}^{|V|} d(v, w_i) [d(v, w_i) < \infty]. \quad (1.4)$$

Таким образом, чем меньше сумма кратчайших путей от нее до других вершин, тем больше значение C_C вершины.

б) Степень посредничества (англ. betweenness centrality) $C_B(v)$ – характеристика вершины, показывающая насколько часто данная вершина лежит на кратчайших путях между другими вершинами в графе:

$$C_B(v) = \sum_{s \neq v, t \neq v, t \neq s} \frac{\sigma_{st}(v)}{\sigma_{st}}, \quad (1.5)$$

где σ_{st} – количество кратчайших путей из вершины $s \in V$ в вершину $t \in V$;

$\sigma_{st}(v)$ – количество этих путей, проходящих через вершину $v \in V$.

Узел сети, соответствующий вершине с высокой степенью посредничества $C_B(v)$, будет хорошо информирован при передаче информации, если передача будет проходить по кратчайшим путям.

в) Степень влиятельности (англ. eigenvector centrality) $C_E(v)$ – рекурсивно рассчитываемая характеристика вершины, основанная на значении этой характеристики у соседних вершин графа. Чем «влиятельнее» соседние вершины v , тем «влиятельнее» она сама:

$$C_E(v) = \frac{1}{\lambda} \sum_{t \in Nei(v)} C_E(t) = \frac{1}{\lambda} \sum_{t \in G} A_{v,t} \cdot C_E(t), \quad (1.6)$$

где $Nei(v)$ – окрестность вершины v для неориентированного графа или полуокрестность исхода для смешанного графа;

λ – числовая константа;

$A_{v,t}$ – элемент матрицы смежности.

Значения $C_E(v)$ являются элементами вектора $\mathbf{x}$, которые определяются решением уравнения $\mathbf{Ax}=\lambda\mathbf{x}$, где $\mathbf{A}$ – матрица смежности, λ и $\mathbf{x}$ – соответственно, собственное значение и собственный вектор матрицы $\mathbf{A}$.

Также к локальным характеристикам следует отнести степень связности вершины $\deg(v)$. Если говорить о моделируемой графом сети, то степень связности вершины характеризует меру «активности» соответствующего узла сети.

1.2.3 Сетевые мотивы

В «Науке о сетях» большое распространение получил подход, заключающийся в выявлении так называемых сетевых мотивов [93, 96, 118]. Целью анализа является поиск таких классов изоморфизма, подграфы которых встречаются в графе G чаще, чем в рандомизированных версиях графа G .

Задачу нахождения сетевых мотивов на k вершинах (k -мотивов) в графе G можно разбить на следующие три подзадачи.

1. Получение множества G_R рандомизированных версий графа G .
2. Подсчет в графе G и в каждом графе множества G_R частот встречаемости подграфов, относящихся к различным классам изоморфизма. При этом решается задача определения классов изоморфизма для найденных подграфов на k вершинах.
3. Анализ сетевых мотивов на основе их встречаемости в графе G и в графах множества G_R .

Далее рассмотрим подробнее каждую подзадачу отдельно.

1.2.3.1 Рандомизация графа

Для получения множества рандомизированных графов, как правило, используются специальные процедуры рандомизации. Так, в программах FANMOD [126] и MotifNet [118] рандомизация графа начинается с создания графа-копии исследуемого графа G , затем выполняется перенаправление ребер этого графа-копии. Для каждого ребра (v,u) делаются попытки перенаправления конца ребра, инцидентного вершине u , на другую вершину. Для этого

случайным образом выбирается другое ребро графа (w,z) , в результате чего формируются ребра (v,z) и (w,u) , а ребра (v,u) и (w,z) удаляются из графа. По некоторым причинам перенаправление ребер может не выполняться (например, если оба ребра начинаются с одной и той же вершины). Тогда случайный выбор ребра (w,u) повторяется до тех пор, пока обмен не осуществится или количество попыток обмена не превысит заданное пользователем значение.

1.2.3.2 Нахождение изоморфных графов

Кроме самой задачи определения изоморфизма графов в процессе выявления сетевых мотивов необходимо определение класса изоморфизма. С этой целью в задачах выявления сетевых мотивов используются алгоритмы, основанные на применении канонической нумерации графов [73]. Каноническая нумерация графов позволяет пронумеровать все возможные графы так, что одно и то же целое число будет соответствовать графам тогда и только тогда, когда они изоморфны. Таким образом, каждый из подграфов на заданном числе вершин будет соответствовать номеру одного из сетевых мотивов на том же числе вершин. При этом существуют различные системы нумерации и алгоритмы получения номеров. В частности, в программе FANMOD [126] используется алгоритм Nauty [95] и одноименная нумерация. В пакете igraph используется каноническая нумерация BLISS. В WolframMathematica используется нумерация Пиперно [106]. Ряд специализированных программ использует нецифровую систему нумерации для решения задачи распознавания химических соединений и обеспечения их ускоренного поиска в специальных базах данных.

Однако даже самый быстрый алгоритм получения номеров графов, которым, по мнению специалистов в данной области [73, 126], является алгоритм Nauty, требует в общем случае экспоненциального времени выполнения (по числу вершин) для получения канонического номера [97].

1.2.3.3 Статистическая значимость подграфов

Сетевые мотивы исследователи больших сетей определяют исходя из значений различных показателей статистической значимости подграфов. Самым популярным таким показателем является так называемая стандартизованная оценка.

Стандартизированная оценка (*Z-score*) – это мера относительного разброса наблюдаемого значения, которая показывает, сколько стандартных отклонений составляет его разброс относительного среднего значения. Это безразмерный статистический показатель. Стандартизированная оценка при выявлении сетевых мотивов определяется по следующей формуле:

$$Z\text{-score} = \frac{n_G - M[n_R]}{\sigma[n_R]}, \quad (1.7)$$

где $M[n_R]$ и $\sigma[n_R]$ – математическое ожидание и стандартное отклонение абсолютной частоты встречаемости подграфа в сгенерированном множестве рандомизированных графов G_R ;

n_G – число подграфов того же класса изоморфизма в исходном графе G .

Иногда рассматриваются не абсолютные частоты встречаемости подграфов, а относительные, тогда вычисляется показатель $Z\text{-score}_C$:

$$Z\text{-score}_C = \frac{C_G - M[C_R]}{\sigma[C_R]},$$

где $M[C_R]$ и $\sigma[C_R]$ – математическое ожидание и стандартное отклонение относительной частоты встречаемости подграфа в сгенерированном множестве рандомизированных графов G_R ; C_G – число подграфов того же класса изоморфизма в исходном графе G .

Сетевыми мотивами считаются такие подграфы, которые относятся к классам изоморфизма с показателем $Z\text{-score}$, превышающем некоторое пороговое значение (сетевыми мотивами, как правило считаются подграфы с $Z\text{-score} > 2$ или $Z\text{-score}_C > 2$).

Другим популярным [128] статистическим показателем является показатель $R = n_G / M[n_R]$. Подграфы с показателем $R > 1,1$ считаются сетевыми мотивами.

Для оценки статистической значимости подграфов известны и другие статистические показатели, например $P\text{-value}$ и показатель Пикарда [105].

1.2.3.4 Алгоритмы выявления сетевых мотивов

Для выявления сетевых мотивов выделяют два семейства алгоритмов: сетевые и подграфо-центричные алгоритмы.

Сете-центричные алгоритмы реализуют расчет частот встречаемости всех возможных подграфов, содержащих заданное число вершин. Между собой сете-центричные алгоритмы различаются по эффективности расчета частот встречаемости подграфов. Так, в программе MFinder для расчета частот встречаемости используется алгоритм «исчерпывающей рекурсии» (exhaustive recursion algorithm) [96]. Другой популярный сете-центричный алгоритм, который в диссертации будет называться алгоритм Верника–Раше [126], основан на последовательном перечислении всех связанных подграфов путем построения т.н. дерева Верника–Раше. Дерево Верника–Раше является корневым, на k -м ярусе дерева перечислены все реализации подграфов на k вершинах в графе. Алгоритм Верника–Раше имеет хорошую скорость работы и может применяться для больших графов.

В сете-центричном алгоритме Kavosh [81] также выполняется перечисление подграфов, содержащих k вершин, путем направленного перебора вершин. Алгоритм выбирает вершину i и строит «неявное» дерево на k вершинах. Все подграфы относительно этой вершины извлекаются на основе алгоритма «вращающейся двери» [87], который рассматривается авторами как самый быстрый алгоритм генерации сочетаний вершин.

К сете-центричным алгоритмам следует отнести также алгоритм G-Tries [110], алгоритм AccMotif [92] и другие алгоритмы, как правило, интенсивно использующие память компьютера за счет чего выигрывают в скорости расчета.

Подграфо-центричные алгоритмы сначала формируют запросы на подграфы некоторых классов изоморфизма, а затем выполняют расчет с использованием сформированных запросов. Изначально термин «подграфо-центричный» ввели Грохоу и Келлис для описания своего алгоритма [75]. Название подграфо-центричный означает, что частота встречаемости подграфа, описываемого в виде запроса, исчерпывающе определяется путем поиска всех возможных отображений этого подграфа на все вершины графа. Преимущество алгоритма Грохоу–Келлиса заключается в эффективном процессе поиска изоморфных подграфов. Этот поиск позволяет избежать избыточного изоморфного тестирования, поэтому размер

подграфов, которые этот алгоритм находит, достигает 15-ти вершин. Тем не менее, предложенный алгоритм имеет существенный недостаток: для расчета всех подграфов всех возможных классов изоморфизма необходимо сформировать огромное число запросов. К подграфо-центричным алгоритмам относят алгоритм MODA, предложенный Омеди с соавторами [100] для подсчета частот встречаемости подграфов в неориентированных графах, а также алгоритм «гибкого поиска шаблонов» Шрейбера и Швобермайера [114], реализованный в программе Mavisto. В этом алгоритме «гибкого поиска шаблонов» также строится дерево подграфов, но при этом результаты расчета встречаемости одних подграфов используются для решения вопроса о необходимости расчета других.

1.2.3.5 Программы для анализа сетевых мотивов

Существует большое разнообразие программ, позволяющих рассчитывать частоты встречаемости подграфов, некоторые из этих программ позволяют оценивать статистическую значимость подграфов, и могут быть использованы для выявления сетевых мотивов. Наиболее известными являются следующие программы: MFinder [96] (2003), Mavisto [114] (2005), Fanmod [126] (2006), NeMoFinder [58] (2006) и более поздняя реализации того же функционала в программе LaMoFinder, Kavosh [81] (2009), библиотека igraph для системы R (2013 год), AccMotif [92] (2013). Однако, все перечисленные программы не позволяют за приемлемое время рассчитывать частоты встречаемости сетевых мотивов больших графов. Так, в работах [58, 81, 92, 96, 114], в которых предлагаются новые алгоритмы точного расчета встречаемости 4-мотивов, исследуемым графом с наибольшим числом вершин является граф Foldoc. При этом граф Foldoc содержит всего 12905 вершин и 109092 направленных ребер и описывает сеть терминов онлайн-библиотеки <http://www.foldoc.org/>. Узлы сети представляют собой термины онлайн-библиотеки, связь (X, Y) от термина X к Y существует, если термин Y используется для описания термина X . При использовании конфигурации оборудования, описанной в разделе 1.1 данной диссертации, наиболее быстрым из рассмотренных программ потребовалось следующее время для расчета: Kavosh – 559 секунд, Fanmod – 580 секунд, AccMotif – 6 секунд. Из этих результатов следует, что самой быстрой является программа

AccMotif. Но более тщательное экспериментальное исследование показало, что программа AccMotif требует значительно большего объема оперативной памяти, чем аналоги. Так, программа AccMotif не позволяет эффективно работать с графами с числом вершин больше двадцати тысяч из-за недостатка оперативной памяти, таким образом, AccMotif не может работать с большими графами. Этот факт, к сожалению, не был отмечен авторами AccMotif в их статье [92].

1.3 Графы сетей интернета

В данном разделе представлен анализ графов сетей интернета. Исследуемые графы некоторых сетей интернета доступны из следующих открытых источников:

- база графов Ю. Лесковеца из Стенфордского университета, США (URL: <http://snap.stanford.edu/data/>);
- база графов М. Ньюмана из Мичиганского университета, США (URL: <http://www-personal.umich.edu/~mejn/netdata/>);
- база графов А. Арены из университета Ровира и Вирхилий, Испания (URL: <http://deim.urv.cat/~alexandre.arenas/data/welcome.htm>).

Данные о графах подсетей социальной сети «ВКонтакте» [17, 18, 32] и подсети ссылок веб-страниц [20] получены студентами Ниткиным Д.А., Бондаренко А.С. и Пахомовым А.А. и описаны в совместных с автором диссертации публикациях.

В диссертации исследуются следующие десять графов сетей интернета.

1) Граф AS – граф интернета на уровне автономных систем (сеть реконструирована по таблицам BGP, опубликованным в рамках проекта Route Views). Узлы сети представляют собой автономные системы интернета, связи – наличие физического соединения между узлами. Граф был создан М. Ньюманом в 2006 году.

2) Граф Routers – граф сети маршрутизаторов интернета. Узлы сети представляют собой маршрутизаторы, а связи – физические соединения между ними. Граф сети получен В. Батагели в 2006 году.

3) Граф OmskWeb – граф подсети ссылок веб-страниц. Граф сети ссылок веб-страниц получен путем просмотра всех HTML-ссылок на сторонние веб-страницы, затем всех HTML-ссылок со сторонних веб-страниц и т.д. Поиск

выполнялся с главных веб-страниц трех образовательных учреждений города Омска – ОмГТУ, ОмГУ и ОмГАУ, начиная с каждой из этих веб-страниц, было получено по 100 тысяч ссылок. Узлы сети – полные доменные имена веб-ресурсов, на которых расположены просмотренные веб-страницы, связи означают наличие хотя бы одной веб-ссылки между веб-страницами, адреса которых относятся к разным доменам [20].

4) Граф VKOmgту – граф подсети пользователей социальной сети «ВКонтакте», указавших местом обучения ОмГТУ. Для анализа этой сети [32] было разработано специальное приложение, запрашивающее данные через API (интерфейс прикладного программирования). Узлами исследуемой подсети «ВКонтакте» являются пользователи, а связи выражают отношение «дружбы» между ними. Использование API существенно упрощает построение графа сети по сравнению с возможностями разбора веб-страниц [20, 51], который использовался при построении графа OmskWeb.

5) Граф Gnutella – граф одноранговой (децентрализованной) сети Gnutella с общим доступом к файлам. При построении графа сети использовались мгновенные снимки сети Gnutella, созданные в 2002-м году. Узлами сети являются клиентские приложения пользователей сети Gnutella, а связи означают факт передачи пользовательских данных между клиентскими приложениями [89].

6) Граф PGP – граф сети пользователей программы PGP для зашифрованного обмена данными [49]. Узлы – это клиентские приложения программы PGP, связь означает включение пользователем одного клиентского приложения в свою сеть доверия запросов от другого клиентского приложения.

7) Граф Google plus – граф подсети социальной сети Google plus, узлы – аккаунты пользователей социальной сети. Связь между узлами i и j означает, что пользователь, владеющий аккаунтом i , добавил в свой «круг общения» аккаунт j [94].

8) Граф Twitter – граф подсети социальной сети Twitter. Узел подсети – это аккаунт пользователя социальной сети Twitter. Связь – отношение «один аккаунт подписан на другой аккаунт» (связи несимметричны) [94].

9) Граф E-mail EU-All – граф сети пользователей электронной почты в европейском исследовательском институте, узлы сети – аккаунты электронной

почты сотрудников института, связи свидетельствуют о наличии электронной пересылки от одного аккаунта к другому (связи несимметричны) [89].

10) Граф E-mail Enron – граф сети пользователей электронной почты. Узлы сети – аккаунты электронной почты, связи свидетельствуют о наличии какой-либо пересылки электронного письма между аккаунтами. Использовались данные от 7 мая 2015 года [90].

1.3.1 Глобальные характеристики

В Таблице 1.1 и Таблице 1.2. представлены глобальные характеристики исследуемых в диссертации графов сетей интернета.

Таблица 1.1 – Глобальные характеристики смешанных графов сетей

Название характеристики	E-mail EU-All	Google plus	Gnutella	OmskWeb	Twitter
Число вершин	265214	107614	62586	106756	81306
Число направленных ребер	310006	10803117	147892	222600	916471
Число ненаправленных ребер	54475	1435168	0	36714	425839
Число компонентов связности	15836	1	12	1	1
Средняя степень вершин	3,159	254,1	4,726041	5,54588	43,49
Коэффициент кластеризации	0,00411	0,1536	0,003872019	0,107775	0,1706
Среднее расстояние между вершинами	4,205846	3,32337	9,200153	9,687077	4,909305

Таблица 1.2 – Глобальные характеристики неориентированных графов сетей

Название характеристики	AS	E-mail Enron	PGP	Routers	VK OmgTu
Число вершин	22963	36692	10680	124651	9927
Число ребер	48436	183831	24316	193620	35596
Число компонент связности	1	1065	1	1	1
Средняя степень вершин	4,218613	20,04	4,55356	3,106594	7,171
Коэффициент кластеризации	0,01114638	0,08531	0,37803	0,03862725	0,09461
Среднее расстояние между вершинами	3,84	4,025143	7,48554	11,27776	4,934083

Как можно видеть, представленные графы сетей содержат сотни тысяч вершин и ребер. Это достаточно большие графы. Однако следует признать, что некоторые исследуемые сети являются лишь фрагментами огромных сетей. Так, число

пользователей социальной сети Twitter составляет 319 миллионов (активных аккаунтов), а число пользователей отечественной социальной сети «ВКонтакте» – 97 миллионов (активных аккаунтов).

1.3.2 Локальные характеристики и сетевые мотивы

Приведем результаты расчета локальных характеристик двух исследуемых в диссертации графов: граф VKOmgту и граф OmskWeb.

1.3.2.1 Сеть VKOmgту

На Рисунке 1.4 представлено изображение графа VKOmgту. Несмотря на визуально сложную структуру графа, этот граф можно исследовать. В частности, можно найти наиболее важные вершины. Так, на Рисунке 1.4 выделена вершина, соответствующая узлу сети, который назовем пользователь «X»¹. Эта вершина, с учетом таких мер важности вершин как степень связности, степень близости к другим вершинам (формула 1.4), степень посредничества (формула 1.5), степень влияния (формула 1.6), является самой важной вершиной графа.

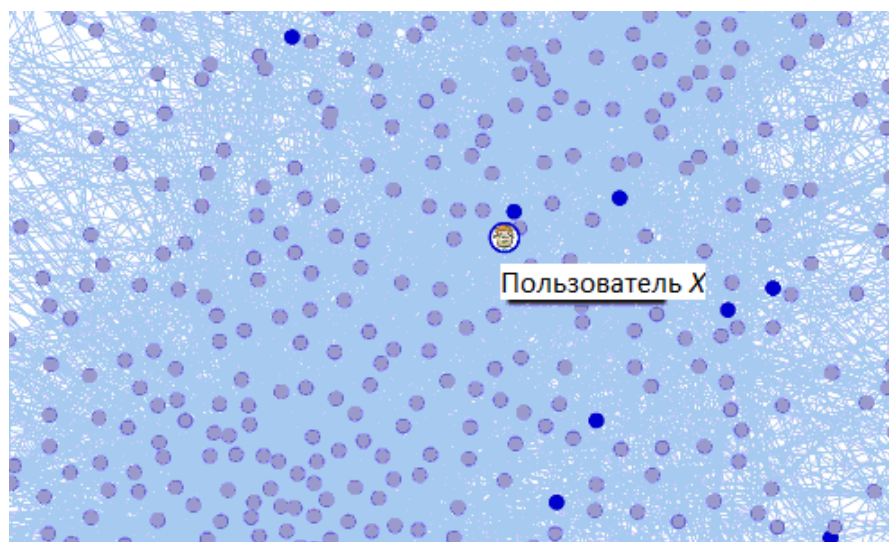


Рисунок 1.4 – Изображение графа VKOmgту

Результаты анализа локальных характеристик графа VKOmgту представлены в Таблице 1.3. Пользователи, соответствующие исследуемым вершинам графа, в целях сохранения их анонимности обозначены большими латинскими буквами: X – пользователь, дублирующий новости университета; Y , W – пользователи,

¹ Имена пользователей социальной сети «ВКонтакте» нельзя разглашать в соответствии с разъяснениями Роскомнадзора [13], однако в работе [31], написанной до разъяснений, приведены названия аккаунтов рассмотренных пользователей.

оказывающие платные услуги студентам; Z – популярный фотограф.

Таблица 1.3 – Важность отдельных вершин в графе VKOmgту

Мера важности вершин	Идентификаторы пользователей, соответствующих наиболее важным вершинам		
	1 место	2 место	3 место
По степени связности	X	Y	V
По степени влияния	X	Z	W
По степени близости к другим вершинам	X	W	Z
По степени посредничества	X	Y	W

В Таблице 1.4 представлены результаты выявления 3-мотивов и 4-мотивов в графе VKOmgту, при этом генерировались 1000 рандомизированных графов в каждом случае. В программе FANMOD время выявления 3-мотивов достигло 846 секунд, из них на выполнение процедур рандомизации графа ушло 158 секунд, а на расчет сетевых мотивов – 688 секунд. Как можно видеть, по показателю $Z\text{-score}_C$ самым значимым подграфом на трех вершинах оказался полный подграф, что свидетельствует о высокой степени кластеризации графа.

Таблица 1.4 – Результаты анализа 3-мотивов и 4-мотивов в графе VKOmgту

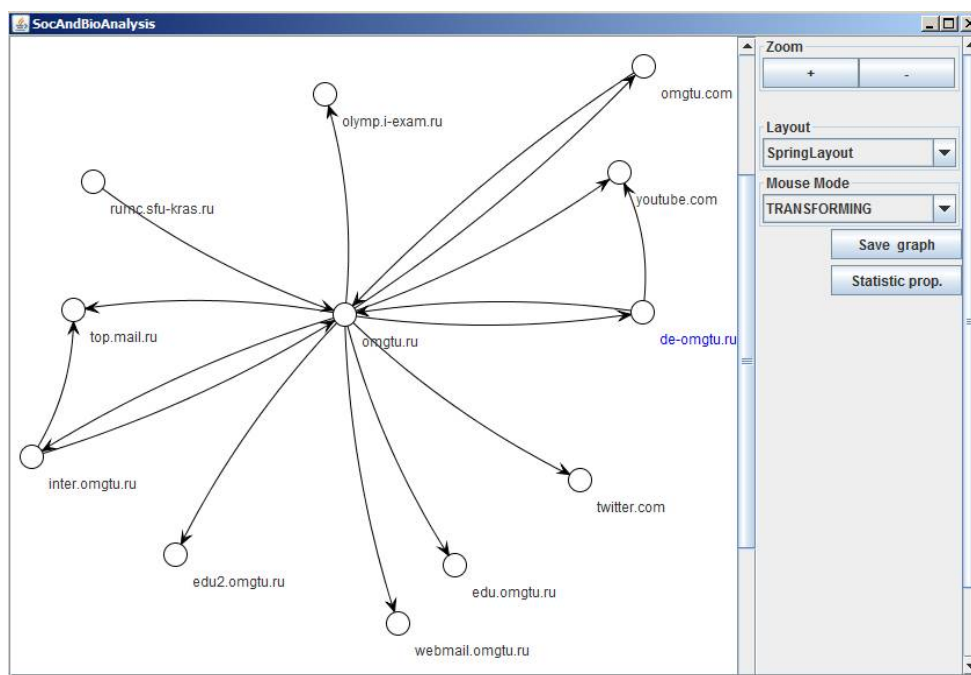
Подграф	Частота встречаемости C_G , %	Средняя частота встречаемости $M[C_R]$, %	Среднее квадратичное отклонение, $\sigma[C_R]$	$Z\text{-score}_C$
	96,634	99,996	9,4404e-005	-356,17
	3,3659	0,0035454	9,4404e-005	356,17
	0,05444	9,0192e-008	9,0095e-009	60424
	0,35621	0,00019242	4,7408e-006	750,97
	5,4441	0,016398	0,00042318	128,26
	61,443	65,747	0,0021401	-20,112
	0,16438	0,12105	4,8228e-005	8,9843
	32,538	34,115	0,0020738	-7,6054

Время расчета при выявлении 4-мотивов при этом достигло 59987 секунд, из них на выполнение процедур рандомизации графа ушло 157 секунд, а на подсчет сетевых мотивов 59830 секунд. Наиболее значимым подграфом на четырех вершинах

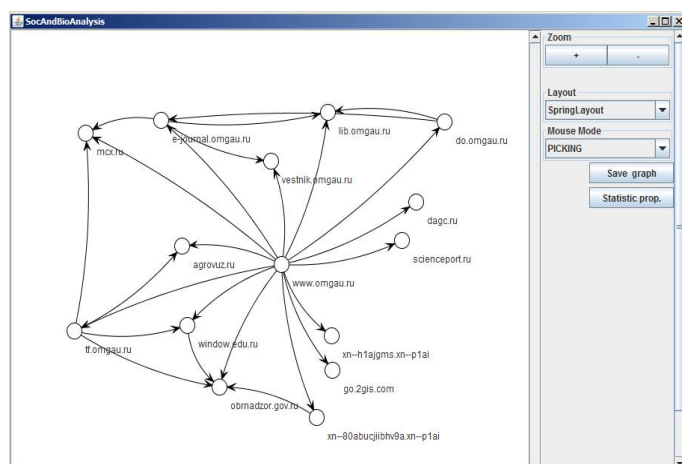
является полный подграф (показатель $Z\text{-score}_C = 60424$). Вторым по значимости – подграф, который содержит четыре вершины и пять ребер ($Z\text{-score}_C = 750,97$).

1.3.2.2 Сеть OmskWeb

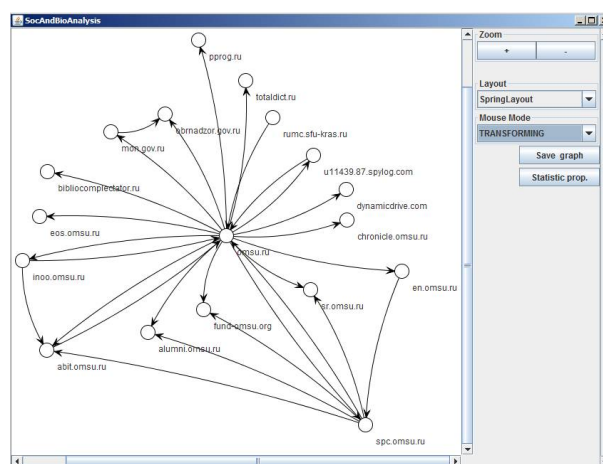
Рассмотрим локальные характеристики отдельных вершин графа OmskWeb (Рисунок 1.5) [20], который был получен путем обхода ссылок веб-страниц, начиная от трех веб-страниц образовательных учреждений города Омска <http://omgtu.ru> (ОмГТУ), <http://www.omgau.ru> (ОмГАУ) и <http://omsu.ru> (ОмГУ).



а)



б)



в)

Рисунок 1.5 – Визуализация ближайшего окружения трех вершин, соответствующих узлам (веб-страницам), с которых был осуществлен обход ссылок веб-страниц для построения графа OmskWeb: а) omgtu.ru, б) www.omgau.ru, в) omsu.ru

В графе OmskWeb при ранжировании вершин по убыванию мощности множества элементов, входящих в полуокрестность исхода, впереди оказались вершины, соответствующие веб-страницам с обзором технологий и политической ситуации (<http://cursor.org>), а также сайтов-блогов (например, <http://sideshow.me.uk>).

При ранжировании по убыванию мощности множества элементов, входящих в полуокрестность захода, впереди оказались вершины, соответствующие веб-страницам <http://facebook.com> (2880 ссылок), <http://twitter.com> (2556 ссылок), <http://vk.com> (1739 ссылок).

В Таблице 1.5 и в Таблице 1.6 приведены результаты выявления 3-мотивов и 4-мотивов, соответственно (использовалась программа FANMOD). В таблицах представлены по три подграфа, имеющих наибольшие значения $Z\text{-score}_C$. Для выполнения расчетов были сгенерированы по 10 рандомизированных графов. При выявлении 3-мотивов на рандомизацию потрачено 36 секунд, на расчет частот встречаемости подграфов – 446 секунд.

Таблица 1.5 – Результаты анализа 3-мотивов (граф OmskWeb)

Мотив	Частота встречаемости C_G , %	Средняя частота встречаемости $M[C_R]$, %	Среднее квадратичное отклонение, $\sigma[C_R]$	$Z\text{-score}_C$
	0,16826	0,00062626	8,9557e-007	1871,8
	1,4917	0,042889	1,476e-005	981,62
	0,11741	0,0050034	1,499e-006	749,84

При выявлении 4-мотивов на генерацию рандомизированных графов потрачено 34 секунды, на расчет сетевых мотивов – 298853 секунды.

Таблица 1.6 – Результаты анализа 4-мотивов (граф OmskWeb)

Мотив	Частота встречаемости C_G , %	Средняя частота встречаемости $M[C_R]$, %	Среднее квадратичное отклонение, $\sigma[C_R]$	$Z\text{-score}_C$
	0,00055221	6,4865e-009	7,4505e-011	74116
	0,00033574	4,0535e-009	5,7337e-011	58555
	0,000359	6,486e-009	7,4499e-011	48187

Отметим, что основной вычислительной проблемой при анализе сетевых мотивов является сложность расчета частот встречаемости подграфов, относящихся к различным классам изоморфизма. Даже при использовании наиболее быстрых программ – FANMOD и igraph, для расчета частот встречаемости подграфов требуется неприемлемо большое время (Таблица 1.7).

Таблица 1.7 – Время расчета (в секундах) частот встречаемости подграфов в графах сетей интернета

	Библиотека igraph для среды R		FANMOD	
	Подграфы на трех вершинах	Подграфы на четырех вершинах	Подграфы на трех вершинах	Подграфы на четырех вершинах
Google Plus	39442	–	–	–
Twitter	76	35217	1076	–
Email Enron	23	10882,83	27	11646
Email_AU-All	40	38508	136	272158
VKOmgtn	0,22	19	1	93

Отсутствие данных в соответствующих ячейках Таблицы 1.7 означает, что при использовании программы получить результат не удалось за пять дней работы.

1.3.3 Исследование сетевых процессов

Модели сетевых процессов широко освещены в работах [3, 78]. Можно выделить следующие семейства моделей.

– Модели анализа устойчивых взаимодействий между узлами сети. В этих моделях определяется функция, характеризующая взаимодействие между узлами, находится устойчивое состояние сети, при этом используются понятия равновесий Парето и Нэша, парная устойчивость взаимодействий узлов [78].

– Модели распространения идей, технологий, вирусов, в том числе модели диффузии [1, 42] и эпидемиологические модели [103].

– Модели оказания влияния на принятия решений. Например, модель де Грута [61], с помощью которой можно обосновать формулу (1.6). Помимо Марковских моделей, таких как модель де Грута, в которой принятие решений узлами на текущем шаге зависит только от принятых решений узлов ближайшего окружения на предыдущем шаге, большое распространение получили байесовские модели [35], в которых учитывается история принятий решений узлами.

– Игровые модели, в которых для каждого узла задается функция выгоды, зависящая от его состояния и состояния его соседей [41, 74].

Рассмотрим подробнее такие игровые модели как «Координационная игра» (Coordination game [53]) и «Простой комплемент» (Simple Complement [78]).

В модели Простой комплемент каждый узел играет одну из возможных стратегий «1» или «0». Если узел выбрал стратегию «1», то платеж узла рассчитывается по формуле: $u_i(1, m_i) = -k + m_i$, где m_i – число узлов ближайшего окружения узла i выбравших стратегию «1», k – число узлов в ближайшем окружении узла i (параметр модели). Если узел выбрал стратегию «0», то платеж, который он получает, равен нулю вне зависимости от выбора стратегий его соседей: $u_i(0, m_i) = 0$. Необходимо найти минимальное множество узлов S_1 , которым нужно выбрать стратегию «1», чтобы была получена максимальная суммарная выгода. В работе [31] описан алгоритм для получения множества S_1 . Модель Простой комплемент используется для оценки степени распространения допинга среди спортсменов, курения среди подростков, а также новых технологий, которые вытеснят старые.

Модель Координационная игра описывает противостояние двух технологий, партий, мнений. В этой игре каждый i -й узел также выбирает одну из возможных стратегий – либо играть «0», либо «1». В зависимости от выбранной стратегии, узлы получают некоторый платеж u_i . При этом платеж u_i , полученный узлом i , равен числу m_i соседних узлов, выбравших ту же стратегию, что и узел i , если таких вершин больше чем узлов, выбравших другую стратегию, иначе полученный узлом i платеж u_i равен нулю. Таким образом, узлу предпочтительнее выбрать играть «1», если большинство «соседей» выбирает ту же стратегию, и

наоборот. Модель Координационная игра называется каскадной, поскольку, изменение выбора стратегии одного узла может повлиять на выбор, который делают все соседние узлы, а те в свою очередь могут повлиять на своих соседей и т.д. В работе [31] описан «жадный» алгоритм, позволяющий найти минимальное множество узлов S_2 , таких что, если узлы этого множества выберут играть некоторую одинаковую стратегию, то это вызовет лавинообразное принятие всеми другими узлами сети выбор играть ту же стратегию.

Реализация алгоритмов для получения множеств S_1 и S_2 представлена в библиотеке классов SocAndBioNetworksAnalysis.

Также в библиотеку SocAndBioNetworksAnalysis, разработанную в рамках диссертации, включены: эпидемиологическая модель «восприимчивый – инфицированный – восприимчивый», модели случайных потерь связей в сети и атак на «концентраторы» [11, 135]. Разработанная система агентного моделирования Simbigraph 2 включает множество реализованных агентных моделей сетевых процессов: модель сегрегации «сахарный мир», ряд эпидемиологических моделей, модель «хищник-жертва» и другие.

Моделирование сетевых процессов позволяет получить важную информацию о влиянии структуры сети на развитие этих процессов. В работе [31] установлено, что при использовании в качестве структуры взаимодействий графа VKOmgту при увеличении значений мер важности увеличивается вероятность того, что вершина войдет в искомые множества S_1 и S_2 . Например, при параметре $k = 4$ в модели Простой комплемент вероятность того, что случайно выбранная вершина графа VKOmgту войдет в S_1 , равна 0,1790 (в S_1 вошло 1777 вершин из 9927 вершин). Но чем больше значение исследуемой меры важности вершины, тем выше у вершины вероятность войти во множество S_1 . Так, на Рисунке 1.6 представлена гистограмма попадания вершин во множество S_1 . На гистограмме вершины упорядочены по убыванию значений соответствующей меры важности и разбиты на подмножества по 500 вершин так, что минимальное значение важности вершины из подмножества, расположенного левее по оси абсцисс, не меньше максимального значения важности вершины из подмножества, расположенного правее. По оси ординат задается вероятность того, что вершины из соответствующего подмножества войдут в оптимальную стратегию. В

соответствии с Рисунком 1.6 можно видеть, что чем больше значение мер важности вершин, тем больше вероятность войти во множество S_1 .

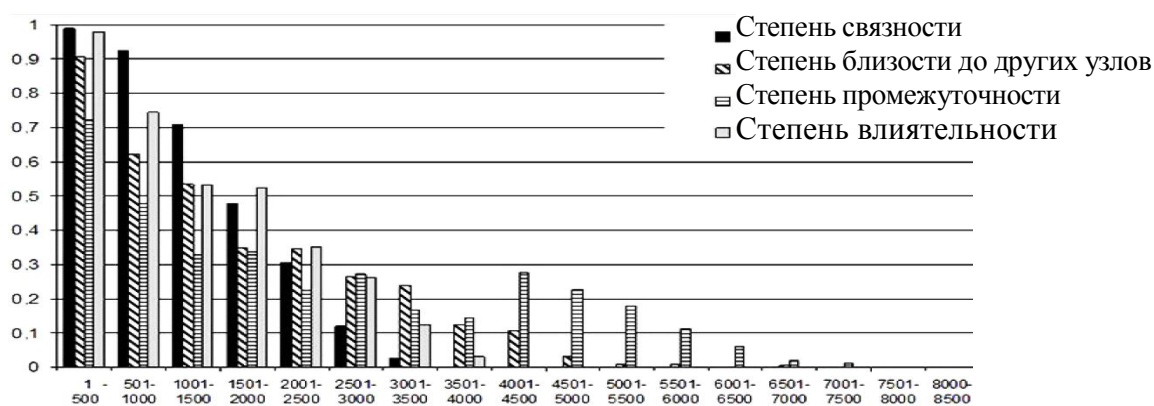


Рисунок 1.6 – Гистограмма попадания вершин во множество S_1

В модели Координационная игра при использовании в качестве структуры взаимодействия графа VKOmgту вероятность того, что случайно выбранная вершина войдет во множество S_2 , равна 0,1558 (во множество S_2 вошло 1547 вершин из 9927 [31]). При этом, из 30 вершин с наибольшей связностью в S_2 вошли все.

В работе [12] при исследовании схем надежности, которые авторы представляют в виде простого неориентированного графа, установлено, что учет мер важности вершин может использоваться для ускорения расчета надежности высоконадежных структурно сложных систем [12, 24]. В свою очередь в работе [65] показано, что сетевые мотивы (в частности 3- и 4-мотивы) могут быть использованы для комплексной оценки устойчивости сетей электростанций.

1.4 Графы молекулярных сетей клетки

Анализ молекулярных сетей клетки является драйвером развития биоинформатики, он помогает лучше понять структуру взаимодействия генов и белков, метаболитов и других химико-биологических составляющих клетки.

Среди молекулярных сетей клетки выделим сети белковых взаимодействий и генные сети. В сетях белковых взаимодействий узлы представляют собой белки, а связи определяют взаимодействия одного или нескольких белков, обеспечивающих выполнение их биологических функций, либо физико-химические взаимодействия с образованием белковых комплексов [68]. В генных сетях узлы – это гены, а связи обозначают влияние одного гена на другой (обычно посредством синтезируемых белков, кодируемых данными генами).

Следует заметить, что при исследовании сетей белковых взаимодействий помимо неточности методов обнаружения взаимодействий (например, один из самых популярных методов – двугибридный анализ – зависит от экспериментальной базы, а также имеет большое количество ложноположительных и ложноотрицательных результатов [72]) существует проблема размерности. Так, в отличие от решенной задачи анализа всех (порядка 20 тыс.) генов человека (международный проект «Геном человека», завершившийся в 2001 г.), массив белков человека (проект «Протеом человека») включает порядка миллиарда протеоформ белков [83].

Сети белковых взаимодействий

Белки выполняют различные функции в организме: белки участвуют в биохимических реакциях, белками являются антитела в иммунной системе, белки транспортируют другие молекулы, необходимые организму для поддержания жизни. Структурно белки состоят из одной или нескольких полипептидных цепей, эти цепи складываются в специфическую трехмерную структуру, называемую третичной структурой белка. На Рисунке 1.7 изображена третичная структура белка p53. Данные о структуре получены из базы <https://www.ncbi.nlm.nih.gov/Structure/pdb/2OCJ> и визуализированы в программе RasMol (<http://www.rasmol.org/>). При визуализации использована шаростержневая модель структуры и цветовая схема, в которой атомы окрашены согласно принятым цветовым обозначениям для отображения химических элементов.

Сеть белковых взаимодействий представляется в виде неориентированного графа. С использованием графа сети белковых взаимодействий некоторые клеточные явления намного легче понять. Например, в работе [85] демонстрируется, что известный белок p53 – супрессор опухоли – важен именно «из-за его расположения в сети транскрипционных факторов» [85].

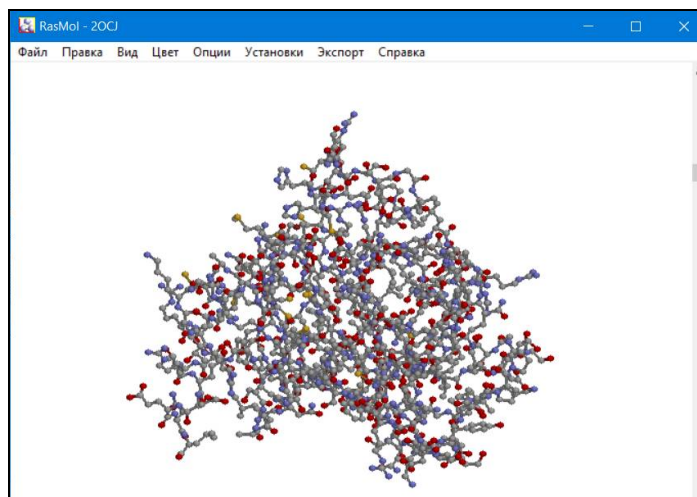


Рисунок 1.7 – Структура белка p53

Графовое представление взаимодействий белков оказывается полезно при разработке лекарственных препаратов. Когда патоген (вирус или бактерия) попадает в организм, он взаимодействует с различными белками. Понимание механизмов белковых взаимодействий клетки обеспечивает разработку новых лекарственных препаратов. Кроме того, реакции организма против инфекций, такие как воспалительные, могут быть объяснены путем анализа графа сети белковых взаимодействий [50]. На Рисунке 1.8 представлен граф белковых взаимодействий почкующихся дрожжей (*Saccharomyces cerevisiae*), визуализация выполнена с использованием разработанной автором диссертации библиотеки классов SocAndBioNetworksAnalysis, структура графа доступна по следующей ссылке: <http://vlado.fmf.uni-lj.si/pub/networks/data/bio/yeast/yeast.htm>.

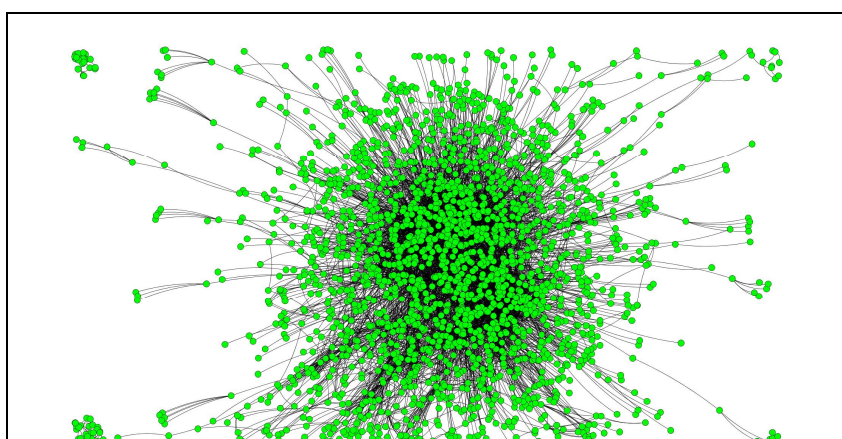


Рисунок 1.8 – Граф сети белковых взаимодействий

Генные сети

В генах информация о строении белков кодируется в виде дезоксирибонуклеиновой кислоты (ДНК). Структурно ДНК состоит из двух линейных нитей, которые намотаны друг на друга в форме двойной спирали. Каждая нить является длинной цепью, которая построена из четырех различных нуклеотидов в качестве элементов: аденин, гуанин, цитозин, тимин. Для сокращения, нуклеотиды в цепочке ДНК обозначаются первой буквой: аденин обозначается буквой «А», гуанин – «Г», цитозин – «С» и тимин – «Т». Две одиночные нити в ДНК удерживаются вместе водородными связями.

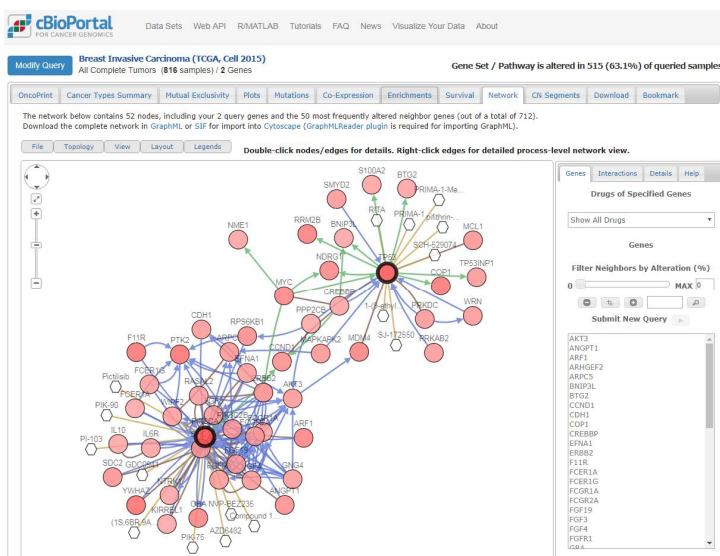


Рисунок 1.9 – Граф сети генетических мутаций, изображение получено онлайн на веб-ресурсе <http://www.cbioportal.org/>

граф реконструирован в результате исследования [54]. В частности, выделены ген DIC3CA, а также ген TP53, кодирующий белок p53. В свою очередь, ген DIC3CA кодирует белок фосфоинозитид-3-киназу, этот белок является ключевым элементом сигнального пути большинства клеток человека, он ответственен, в том числе, за уход от апоптоза (процесса программируемой клеточной гибели). Целью изучения генных сетей является исследование процессов в клетке, идентификация «мишеней» лекарств и выявление неизвестных функций генов. Как можно видеть, даже при анализе влияния этих двух генов задействуется 712 других (на Рисунке 1.9 представлены только 50 из них). Такое большое число задействованных генов – ситуация довольно типичная для генных сетей.

В диссертации рассмотрены пять графов молекулярных сетей клетки.

1) PPI_HS – неориентированный граф сети взаимодействий белков Homo Sapiens. Исходные данные о сети аккумулируются в базе данных BIOGRID [57].

2) YeastS – граф сети белковых взаимодействий в клетке *Saccharomyces cerevisiae* (пекарские дрожжи) [115].

3) GenReg – граф регуляторной сети генов, задействованных в появлении рака коры надпочечников у человека. Сеть регистрирует влияние одних генов на экспрессию других, граф построен в рамках проекта TCGA [119].

4) Regulon – граф генной сети, в которой гены связаны с возникновением рака желудка. Граф построен в лаборатории системной биологии университета Колумбия в США (<http://califano.c2b2.columbia.edu/>).

5) Pathway Commons – граф генной сети, построенной по данным о 21 известной подсети взаимодействий генов в геноме человека, описывающих 446 различных болезней. Сеть построена на основе литературного скрининга, профилирования экспрессии генов и полногеномного поиска ассоциаций [76].

Одним из способов доступа к исследуемым сетям является использование программы CyNDEx 2, которая встроена в систему Cytoscape и позволяет импортировать данные с онлайн-платформы NDEx (Network Data Exchange [64]).

1.4.1 Глобальные характеристики

В Таблице 1.8 представлены глобальные структурные характеристики графов и дата размещения графов на платформе NDEx. Как можно видеть, большинство графов получены исследователями в последние годы, графы сетей содержат порядка двадцати тысяч вершин и сотни тысяч ребер (дуг).

На Рисунке 1.10 изображены эмпирические распределения степени связности вершин (PCC) исследуемых графов. На оси абсцисс – степени связности вершин (либо полустепень захода и полустепень исхода для смешанных графов). По оси ординат – доля вершин с соответствующей степенью связности.

В работах [91, 112] рассматривается для графов молекулярных сетей клетки использовать аппроксимацию PCC в виде степенного закона $Q_k \sim c \cdot k^{-\alpha}$. Степенной закон PCC отображается как прямая линия при использовании логарифмических шкал при построении PCC на графике. Однако для рассмотренных графов PCC на логарифмически-логарифмической шкале сильно отличаются от прямых линий.

Таблица 1.8 – Характеристики графов молекулярных сетей клетки

Характеристики	Графы				
	PPI_HS	YeastS	GenReg	Regulon	Pathway Commons
Дата публикации	03.10. 2018	—	07.12.17	16.06.2018	22.08. 2017
Число вершин	22036	2361	16859	21097	19987
Число ребер	309245	6646	713649	521756	824675
Число компонент связности	27	101	1	1	1
Средняя степень вершин	28,06725	5,629818	84,66089	51,89439	82,52114
Коэффициент кластеризации	0,05779974	0,1023149	0,05910832	0,01580226	0,1242762
Средняя длина кратчайших путей	3,229954	4,376182	3,179365	2,966	2,965538

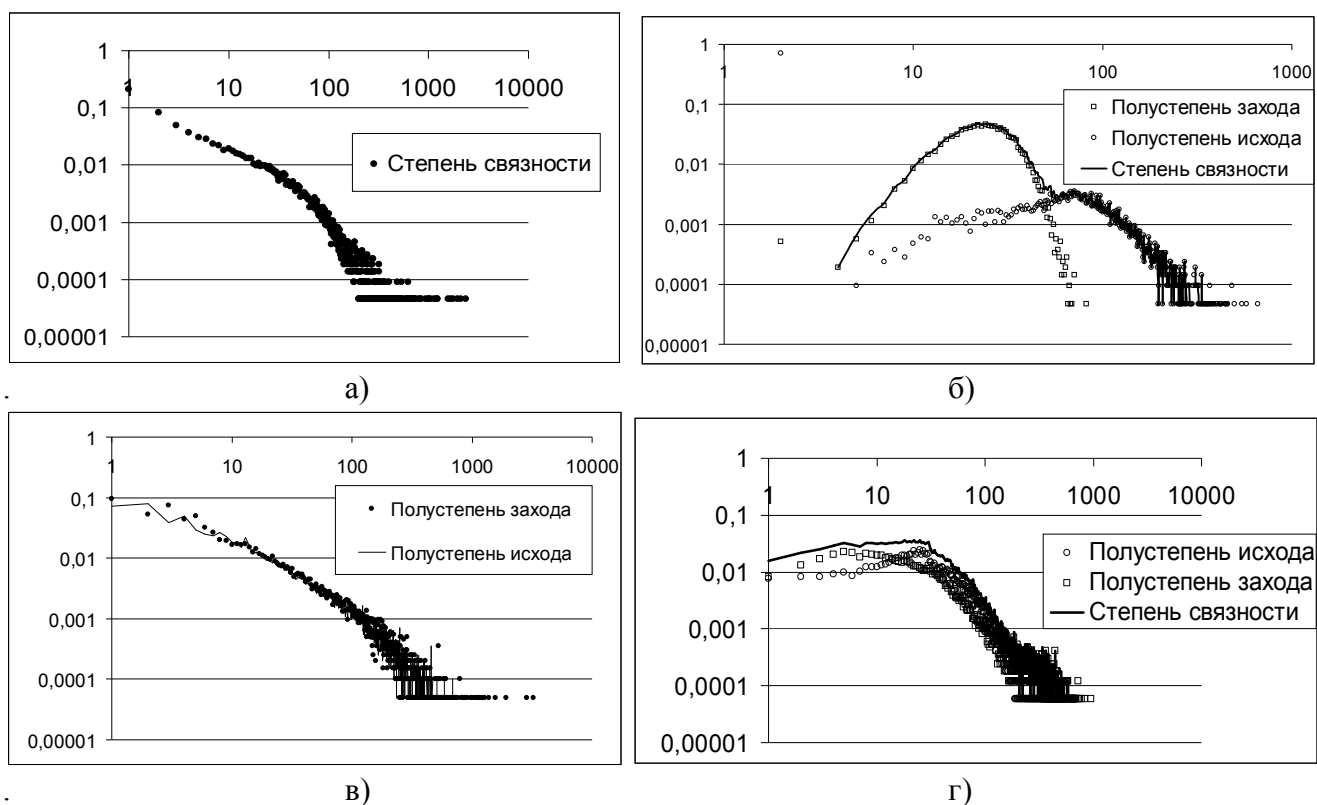


Рисунок 1.10 – РСС для графов: а) PPI_HS, б) Regulon, в) Pathway Commons, г) GenReg; шкалы представлены в логарифмическом масштабе

1.4.2 Локальные характеристики

Узлы, соответствующие наиболее важным вершинам исследуемых графов молекулярных сетей клетки, приведены в Таблице 1.9.

Таблица 1.9 – Наиболее важные узлы молекулярных сетей клетки

Название графа	Используемые метрики важности вершин в графах сетей		
	по влиятельности	по степени связности	по промежуточности
PPI_HS	Ntrk1	Trim25	Trim25
YeastS	Ymr049c	Ypr110c	Ynl189w
GenReg	NPAS2	RNF14	MSF
Regulon	EGLN2	FHL1	ELL2
Pathway Commons	SP1	SP1	MYC

Описание перечисленных в Таблице 1.9 генов и белков, а также их функции можно найти по резюме генов на следующих сайтах: <https://www.genecards.org> и <https://www.uniprot.org>, <https://www.ncbi.nlm.nih.gov/>.

1.4.3 Сетевые мотивы

Понятие сетевых мотивов впервые было использовано именно для анализа молекулярных сетей клетки [96]. В работе [120] А. Веспигнани объясняет наличие сетевых мотивов тем, что «эволюция сохраняет модули, которые выполняют конкретную ... функцию»¹. Для объяснения сетевых мотивов необходимы знания предметной области. Так, в сети белковых взаимодействий часто встречаются рецепторы (R), взаимодействующие с субстратом (A), такие как G-белковые рецепторы, и рецепторы, взаимодействующие с G-белком, в белковых сетях также часто встречаются каскады с участием киназ (K), фермента фосфатазы (F) и различных факторов транскрипции (TF). На Рисунке 1.11 представлены перечисленные белковые взаимодействия, все они приводят к образованию так называемого BI-FAN-мотива.

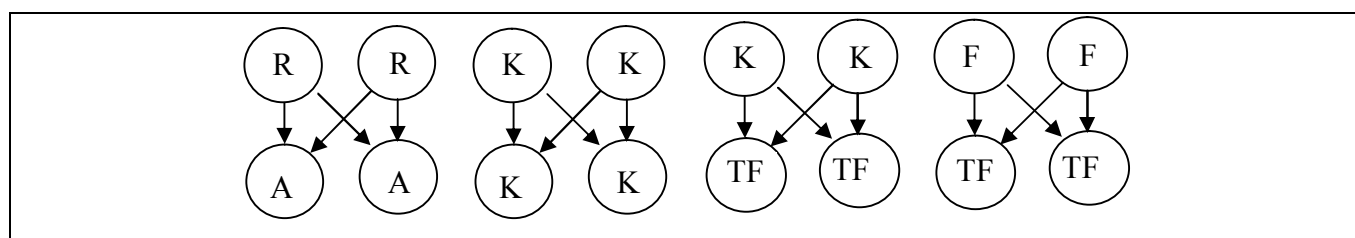


Рисунок 1.11 – BI-FAN-мотив и его реализации в белковых сетях

Сетевой мотив, получивший название FEED-FORWARD LOOP (Рисунок 1.12), часто встречается в генных сетях [93].

¹ «... evolution preserves modules that define specific ... functions»

Этот сетевой мотив определяет взаимодействие, при котором целевой ген В регулируется генами А и Б, при этом ген Б также регулируется геном А. Наличие сетевого FEED-FORWARD LOOP-мотива объясняется особенностями использования факторов транскрипции. Процесс синтеза белка исследуемого гена зависит от присутствия белков (факторов транскрипции), синтезируемых на основе других генов. Обычно необходимо присутствие целого ряда различных факторов транскрипции.

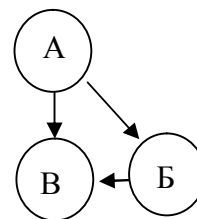


Рисунок 1.12 – Сетевой мотив FEED-FORWARD LOOP

Время, требуемое для анализа сетевых мотивов на трех и четырех вершинах в графах молекулярных сетей клетки, огромно. Так, для исследуемых графов только расчет частот встречаемости подграфов требует десятков тысяч секунд (Таблица 1.10).

Таблица 1.10 – Время, требуемое для расчета частот встречаемости подграфов в среде R (библиотека igraph) и в программе FANMOD

Граф	Время расчета подграфов, в секундах			
	igraph для среды R		Fanmod	
	на 3 вершинах	на 4 вершинах	на 3 вершинах	на 4 вершинах
PPI_HS	35	18824	381,026	182161
YeastS	0,01	0,35	0,052	1,62
GenReg	44	15163	513,986	188820
Regulon	13	2367	92,173	13361,8
Pathway Commons	300	388056	323,9	—

1.5 Случайные графы

С конца XX века в связи с появлением новых задач анализа и моделирования больших сетей было предложено множество моделей сетей на основе случайных графов. Случайные графы позволяют воспроизводить важные структурные характеристики сетей: «правило шести рукопожатий» («Тесный мир»), степенное распределение степени связности вершин («Безмасштабные сети»), наличие сообществ (высокий коэффициент кластеризации) и т.д.

1.5.1 Графы предпочтительного связывания

Широкую популярность получили растущие случайные графы, которые позволяют также прогнозировать рост сети на основе заданной модели роста. Самым известным классом растущих графов являются графы предпочтительного связывания, в том числе граф с линейным правилом предпочтительного связывания, предложенный американскими учеными А. Барабаши и Р. Альберт (граф БА). Две из десяти работ (это статьи [38, 43]) в области физики, имеющие больше всего ссылок по данным ISI, посвящены описанию и анализу графа БА. На статью [43], в которой был предложен граф БА, в Scopus содержится более 20 тысяч ссылок, что свидетельствует о важности графа БА для исследования природы больших сетей. Использование графа БА позволило объяснить устойчивость сетей интернета, сетей белковых взаимодействий и ряда других «безмасштабных» сетей к случайным отказам элементов сети – узлов и связей [44], восприимчивость к вирусам [44] и чувствительность к «атакам» на так называемые хабы – узлы с большим числом связей [37]. Поэтому граф БА стал основной моделью для исследования таких сетей интернета, включая социальные сети и сеть ссылок веб-страниц, молекулярные сети клетки и сети химических взаимодействий [38, 40, 43, 44, 45]. Появились также графы, которые являются более общими моделями по отношению к графу БА. Такие как случайные графы Купера-Фрези [52], Дороговцева-Мендеса-Самухина [66], Эргуна-Роджерса [71], Дериха-Мортерса [62], Пеннока [104], Джексона-Роджерса [79] и др. С помощью этих модификаций графа БА исследователи пытаются воспроизвести некоторые специфические особенности растущих сетей, такие как РСС вершин, коэффициент кластеризации, динамические характеристики сети. В данной работе эти особенности моделей больших сетей не имеют принципиального значения. Поэтому в качестве основной модели выбрана наиболее известная модель сетей – граф БА. Именно она будет использоваться при аналитическом исследовании разрабатываемых алгоритмов.

1.5.2 Граф Барабаши-Альберт

Выращивание графа БА выполняется итерационно, начиная с некоторого графа-затравки. Пусть этот граф-затравка представляет собой полный граф, содержащий m вершин. На каждом шаге выращивания графа добавляется новая вершина, инцидентная m новым ребрам (рассмотрим для простоты неориентированный граф БА). Оставшимися свободными концы новых ребер соединяются с вершинами графа БА случайно так, что вероятность выбрать вершину i зависит от ее степени связности k_i :

$$p_i = k_i / \sum_{j=1}^{|V|} k_j \quad (1.8)$$

Чем больше степень связности вершины, тем больше вероятность, что эта вершина будет выбрана для присоединения, так реализуется правило «богатый становится богаче». Если вместо ребер с новой вершиной добавляются исходящие дуги, такой граф будем называть ориентированным графом БА.

1.5.3 Вероятностные характеристики графа БА

Граф БА хорошо исследован. В частности, известно [43], что для графа БА распределение степени связности $Q_k \sim 2mk^{-3}$. Подобные степенные распределения наблюдаются у многих больших сетей (Рисунок 1.13).

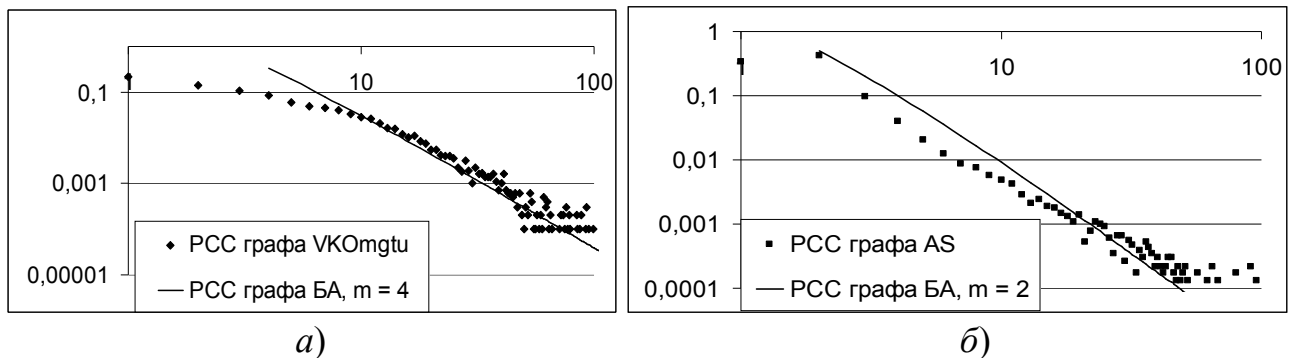


Рисунок 1.13 – PCC для графов AS (справа), VKOmgTu (слева), а также графов БА с соответствующими параметрами генерации m

В частности, в диссертационной работе будет использовано известное соотношение для среднего значения максимальной степени связности $K_{\max}$ в зависимости от числа вершин N в графе. Так, известно, что $K_{\max} \sim N^{1/2}$. В частности, при $m = 2$ справедлива следующая формула [10]:

$$K_{\max} = 3,05 \cdot N^{1/2}. \quad (1.9)$$

Граф БА в данной работе выбран как базовая модель в силу своей популярности и простоты. Однако следует заметить, что использование линейной функции предпочтения в формуле (1.8) существенно сужает сферу применения графа БА. Например, линейная функция предпочтения не позволяет моделировать известный психофизиологический закон Вебера-Фехнера, в соответствии с которым интенсивность ощущения прямо пропорциональна логарифму интенсивности раздражителя, а не линейна ему. Эта задача успешно решается при использовании графов с нелинейным правилом предпочтительного связывания [8].

За последние годы решен ряд актуальных задач в области моделирования сетей с использованием графов предпочтительного связывания:

- задача калибровки графов не только по РСС, но и по двумерному распределению степеней связности концевых вершин случайно выбранного ребра [9];
- задача калибровки графов по РСС и коэффициенту кластеризации [28, 134];
- задача [9, 133] моделирования сети путем «смешивания» случайных графов предпочтительного связывания, когда классические способы подбора параметров генерации [8] не приводят к решению;
- задача анализа динамики степени связности отдельно выбранных вершин в графах предпочтительного связывания [1];
- задача описания графов предпочтительного связывания в условиях случайных удалений вершин и ребер в процессе генерации графа [6-11, 135].

1.6 Выводы по первой главе

1. Разработка методов, алгоритмов и программ для анализа графов сетей интернета имеет важное общественное и научное значение. Она необходима для создания эффективных технологий передачи данных, выработки стратегий борьбы с компьютерными вирусами, деструктивным информационным влиянием и т. д. Одним из методов анализа графов больших сетей является выявление так называемых сетевых мотивов. Сетевые мотивы – это такие связные подграфы, которые значительно чаще встречаются в графе исследуемой сети, чем в его

рандомизированных версиях. Исследования больших сетей на основе выявления сетевых мотивов привели к открытию новых белковых взаимодействий [36, 116], новых способов диагностирования рака [59], новых подходов к оценке надежности технических и социальных сетей, выявлению «скрытых особенностей» этих сетей [136]. Преобладание тех или иных сетевых мотивов является особенностью всякой большой сети, отличающей ее от других сетей, в первую очередь, от сетей другой природы. Так, выявление FULLY CONNECTED TRIAD-мотива является признаком графов информационных сетей интернета [77], для графов молекулярных сетей клетки характерны BI-FAN- и FEED-FORWARD LOOP-мотивы, в графах трофических сетей обнаруживаются THREE-CHAIN- и BI-PARALLEL-мотивы и т.д.

2. Самой трудоемкой подзадачей при выявлении сетевых мотивов является подзадача расчета частот встречаемости подграфов, относящихся к различным классам изоморфизма. На решение этой подзадачи часто требуется неприемлемо большое время, даже если граф содержит всего несколько десятков тысяч вершин. При этом современные сети интернета содержат миллионы узлов и связей, а значит, использование только параллельных или распределенных вычислений не позволит обеспечить удовлетворительное решение данной подзадачи. Поэтому в последние годы развиваются статистические методы расчета частот встречаемости подграфов.

3. В последние два десятилетия предложено множество моделей сетей на основе случайных графов. Использование случайных графов позволяет исследовать наиболее общие закономерности больших сетей. Так, их использование позволило выработать и обосновать методы управления сетевыми процессами [46], выявить и объяснить важные свойства сетей [98]: устойчивость сетей интернета к случайным отказам элементов, нулевой порог чувствительности к распространению вирусов.

Среди графовых моделей больших сетей широкую популярность получили графы предпочтительного связывания, которые помимо воспроизводства заданных структурных характеристик, позволяют прогнозировать рост сети на основе использования модели роста. Самым известным графом предпочтительного связывания является граф Барабаши-Альберт. Именно этот граф будет использоваться в диссертации при аналитическом исследовании разрабатываемых алгоритмов.

ГЛАВА 2 МЕТОД СЛУЧАЙНОГО ВЫБОРА ОСТОВНЫХ ДЕРЕВЬЕВ

Статистические методы позволяют получать статистические оценки для относительных или абсолютных частот встречаемости подграфов, относящихся к различным классам изоморфизма. Выделяют четыре класса статистических методов: 1) методы на основе случайного блуждания, 2) методы перечисления подграфов со случайным отсечением выборки, 3) методы обобщений графов и 4) методы случайного выбора пути. Методы, относящиеся к первым трем классам, в общем случае, не позволяют получать несмещенные оценки частот встречаемости подграфов. Методы, относящиеся к четвертому классу, не позволяют исследовать ориентированные и смешанные графы. Поэтому в диссертации предлагается метод случайного выбора остовных деревьев, который расширяет методы случайного выбора пути и позволяет исследовать и неориентированные, и ориентированные, и смешанные графы. Причем предлагаются два режима работы, обеспечивающие достижение 1) минимальной абсолютной погрешности расчетов, 2) минимальной относительной погрешности расчетов.

2.1 Статистические методы расчета

Реализацией подграфа i -го класса изоморфизма в графе назовем подграф, относящийся к i -му классу изоморфизма.

Пусть $S(G)$ – множество всех реализаций различных подграфов, содержащих заданное число k вершин графа G . Множество $S(G)$ можно разбить на подмножества $S_i(G)$, в каждом из которых содержатся реализации подграфов i -го класса изоморфизма, $i \in X$, где X – множество классов изоморфизма. Таким образом, относительная частота встречаемости подграфов i -го класса изоморфизма $C_i(G)$ определяется следующим образом:

$$C_i(G) = \frac{|S_i|}{|S(G)|} = |S_i| \cdot \left(\sum_{j \in X} |S_j| \right)^{-1}.$$

Для множества S_i реализаций i -го класса изоморфизма, случайным образом взятых в G по некоторому алгоритму A , отображение $\hat{C}_i^A : (S, G \rightarrow [0,1])$ называется статистической оценкой относительной частоты $C_i(G)$ встречаемости подграфов, относящихся к i -му классу изоморфизма, и полученной при использовании алгоритма A . Отображение $\hat{n}_i^A : (S, G \rightarrow \{0, 1, \dots\})$ называется статистической оценкой абсолютной частоты $|S_i(G)|$ встречаемости подграфов, относящихся к i -му классу изоморфизма в графе G .

Будем говорить, что полученная при использовании алгоритма A статистическая оценка $\hat{C}_i^A$ или $\hat{n}_i^A$ несмещенная, если математическое ожидание $M(\hat{C}_i^A) = C_i(G)$ или $M(\hat{n}_i^A) = |S_i|$ соответственно.

Будем говорить, что статистическая оценка является состоятельной при использовании A , если при достаточно больших значениях объема выборки оценка $\hat{C}_i^A$ или $\hat{n}_i^A$ сходится к истинному значению $C_i(G)$ и $|S_i|$ соответственно.

Будем говорить, что статистическая оценка $\hat{C}_i^A$ или $\hat{n}_i^A$ является эффективной, если она имеет наименьшую дисперсию среди всех несмещенных оценок, вычисленных по выборкам одного и того же объема.

2.1.1 Классификация статистических методов расчета

Существующие статистические методы расчета можно разделить на четыре класса методов по выбранной стратегии расчета (Таблица 2.1).

- 1) методы на основе случайного блуждания (СБ);
- 2) методы перечисления подграфов со случайным отсечением выборки (СОВ);
- 3) методы обобщений графов (ОГ);
- 4) методы случайного выбора пути (СВП).

Следует заметить, что большинство известных методов не позволяют проводить расчет для смешанных графов. Некоторые методы не позволяют получить оценки абсолютных частот встречаемости подграфов, а только оценки их относительных частот встречаемости.

Таблица 2.1 – Методы статистического расчета частот встречаемости подграфов

Название алгоритма	Ссылка на файл для запуска	Год создания	Ограничение на число k вершин	Расчет $\hat{C}_i^A$ или $\hat{n}_i^A$	Возможность расчета в смешанных графах	Класс методов
ESA [82]	[39]	2004	Нет	$\hat{C}_i^A$	Да	СБ
RAND-ESU [126]	[127]	2005	Нет	$\hat{n}_i^A$	Да	СОВ
TNP [107]	–	2006	5	$\hat{C}_i^A$	Нет	ОГ
GUISE [47]	[48]	2012	5	$\hat{C}_i^A$	Нет	СБ
RAND-SCMD [121]	–	2012	Нет	$\hat{n}_i^A$	Да	ОГ
Wedge Sampling [113]	[86]	2013	3	$\hat{n}_i^A$	Нет	СВП
GRAFT [108]	[109]	2014	5	$\hat{C}_i^A$	Нет	ОГ
PSRW & MSS [122]	–	2014	Нет	$\hat{C}_i^A$	Нет	СБ
MHRW [88]	–	2015	Нет	$\hat{C}_i^A$	Нет	СБ
RAND-FaSE [101]	[102]	2015	Нет	$\hat{n}_i^A$	Да	СОВ
Path Sampling [80]	–	2015	4	$\hat{n}_i^A$	Нет	СВП
k-profile sparsifier [69]	[67]	2016	4	$\hat{C}_i^A$	Нет	ОГ
MOSS-5 [123]	[124]	2018	5	$\hat{n}_i^A$	Нет	СВП
SSRW [129]	–	2018	7	$\hat{C}_i^A$	Нет	СБ

2.1.2 Методы случайного блуждания

Первым известным статистическим методом расчета является метод случайного выбора ребра (СВР), предложенный в работе [82], и используемый для получения статистических оценок для относительных частот $\hat{C}_i^{CBP}$ встречаемости подграфов.

Метод СВР основан на реализации случайного процесса, который начинается с равновероятного выбора ребра. Далее из ближайшего окружения вершин, инцидентных выбранному ребру, выбирается третья вершина. Потом из ближайшего окружения этих трех вершин – четвертая и т.д. до получения реализации подграфа желаемого размера k . Алгоритм, реализующий метод СВР, представлен на Рисунке 2.1.

НАЧАЛО

Ввод: граф $G = (V, E)$, N^+ – число статистических испытаний для расчета относительных частот встречаемости подграфов на k вершинах

Вывод: счетчики $C[i]$, содержащие значения относительной частоты встречаемости подграфа $i \in X$, где X – множество классов изоморфизма

1. Для всех $i \in X$ делай: $C[i] \leftarrow 0$
2. $E_s \leftarrow \emptyset$, $V_s \leftarrow \emptyset$
3. Для всех $j = 1, N^+$ делай
 - 3.1. $(u, v) \leftarrow$ получить случайное ребро из E
 - 3.2. $V_s \leftarrow \{u, v\}$, $E_s \leftarrow \{(u, v)\}$
 - 3.3. Пока $|V_s| \neq k$ делай
 - 3.3.1. $L \leftarrow$ множество ребер, инцидентных вершинам из V_s
 - 3.3.2. $L \leftarrow L \setminus E_s$
 - 3.3.3. ЕСЛИ $(L = \emptyset)$ $j = j - 1$, продолжи цикл по j
 - 3.3.4. $(w, y) \leftarrow$ получить случайное ребро из L
 - 3.3.5. $V_s = V_s \cup \{w, y\}$
 - 3.3.6. $E_s = E_s \cup \{(w, y)\}$
 - 3.4. $Num = GetISO[G[V_s]]$ // получить класс изоморфизма $Num \in X$
 - 3.5. $C[Num] = C[Num] + 1$
4. Для всех $i \in X$ делай $C[i] = C[i]/N^+$

КОНЕЦ

Рисунок 2.1 – Алгоритм, реализующий метод СВР

Заметим, что при использовании метода СВР одни подграфы систематически находятся чаще других [125]. Рассмотрим это более подробно на примере расчета относительных частот встречаемости «треугольников» в графе, изображенном на Рисунке 2.2.

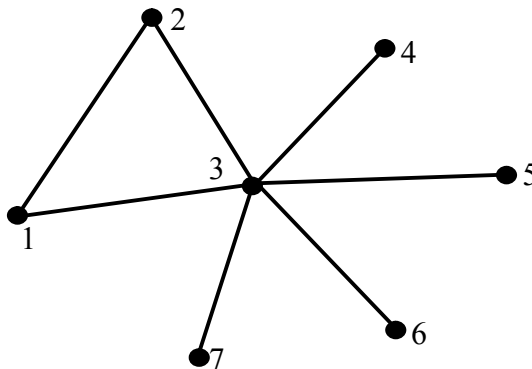


Рисунок 2.2 – Изображение графа, граф содержит 7 вершин и 7 ребер

Общее число подграфов на трех вершинах в графе на Рисунке 2.2 равно¹ 15, причем, только один из этих подграфов является «треугольником». Очевидно, что истинное значение относительной частоты встречаемости «треугольника»

будет $C_{\Delta}^{CBP} = \frac{1}{15}$. Если же посчитать математическое ожидание относительной

частоты встречаемости «треугольника», то получим $M[\hat{C}_{\Delta}^{CBP}] = \frac{1}{7} \cdot 1 + \frac{2}{7} \cdot \frac{2}{6} = \frac{5}{21}$.

Действительно, если в результате выполнения шага 3.1 алгоритма, реализующего метод СВР (Рисунок 2.1), будет выбрано ребро (1,2), что произойдет с вероятностью 1/7, то на следующих шагах алгоритма с вероятностью равной единице будет найден «треугольник». Если в результате выполнения шага 3.1 будет выбрано ребро (1,3) или (2,3), что может произойти с вероятностью 2/7, то для получения «треугольника» необходимо, чтобы на следующих шагах алгоритма было выбрано ребро (1,2) или (2,3) в первом случае и (1,2) или (1,3) во втором. Это возможно с вероятностью 2/6.

Поскольку $C_{\Delta}^{CBP} \neq M[\hat{C}_{\Delta}^{CBP}]$, то оценка частот встречаемости подграфов методом СВР является смещенной.

Другие методы случайного блуждания [47, 88, 129] основываются на применении метода Монте-Карло с использованием цепей Маркова. Особенностью этих методов является получение лишь оценок относительных частот встречаемости подграфов. При этом рассматриваются только неориентированные подграфы.

2.1.3 Методы обобщений графов

Идея методов обобщений графов заключается в точном расчете частот встречаемости подграфов в небольшой части графа или в редуцированном графе, полученном из исходного графа. На основе найденных значений оцениваются

¹ Подсчитать число реализаций подграфов на трех вершинах можно, предварительно подсчитав сумму всех путей длины два ребра (вилки), после чего вычесть утроенное число треугольников (одна реализация треугольника включает в себя три таких пути). У каждого пути два ребра существует центральная вершина – это вершина, инцидентная обоим ребрам пути. Подсчитать количество путей длиной два ребра, для которых вершина v является центральной, можно как $C_2^{\deg(v)}$. Таким образом, число всех реализаций подграфов равно $1+1+6(6-1)/2-2 = 15$.

частоты встречаемости подграфов в оригинальном графе. Алгоритмы из этого класса между собой отличаются по методам аппроксимации графа и по методам аппроксимации полученных статистических оценок.

Один из первых методов обобщений графов предложила Н. Пржил с соавторами [139] – метод обработки выбранных вершин (Targeted Node Processing – TNP). Этот метод ориентирован на исследование сетей белковых взаимодействий. Он основан на предположении, что из-за единообразия сетей белковых взаимодействий можно использовать расчет частот встречаемости подграфов, выполненный в небольшой части графа.

В методе RAND-SCMD предлагается использовать расчет частот встречаемости подграфов для редуцированных графов [185]. При этом нет никакой гарантии, что подграфы «сжимаются» равномерно, что приводит к получению погрешностей расчета.

Метод GRAFT [142] рассчитывает частоты встречаемости подграфов в неориентированных графах размером до 5 вершин. Метод основывается на случайном выборе ребра, после чего определяются все индуцированные этим ребром подграфы. Ребра выбираются без возвращения (поэтому при переборе всех ребер получается точный результат). Подобный алгоритм с удалением ребер с заданной вероятностью предложен также Эленбергом с соавторами [45, 46], отличием их метода от GRAFT является то, что они позволяют оценить частоту встречаемости всех подграфов на трех и четырех вершинах, а не только связных.

Все методы обобщений графов дают смещенные статистические оценки для частот встречаемости подграфов и разработаны для исследования сетей специфической структуры.

2.1.4 Методы перечисления подграфов со случайным отсечением выборки

На данный момент все методы перечисления подграфов со случайным отсечением выборки основаны на переборе всех подграфов и на построении корневого дерева, на каждом k -м ярусе которого могут быть найдены все связные подграфы, содержащие k вершин. В методах с отсечением выборки каждому k -му ярусу дерева ставится в соответствие некоторая вероятность p_k сохранения ветвей.

Вероятность того, что любой из искомых подграфов размера k будет сохранен, равна $\prod_{k=1}^d p_k$. Впервые алгоритм предложен в рамках диссертации С. Верника и получил название RandESU [193]. Дальнейшие модификации алгоритма Верника, предложенные Паредесом и Реберо [131], отличаются эффективной структурой хранения подграфов для ускорения процесса определения изоморфизма.

2.1.4.1 Метод Верника-Раше

Для перечисления всех подграфов на k вершинах в графе $G = (V, E)$ в работе [126] предложен метод, который в рамках диссертации будем называть методом Верника-Раше. Также в работе [126] авторы предложили статистическую версию метода, которую будем называть статистическим методом Верника-Раше. Чтобы лучше описать эти методы, необходимо использовать ряд соглашений. Положим, что все вершины графа G помечены целыми числами. Чтобы обозначить, что метка вершины u больше, чем метка вершины v , напомним « $u > v$ ». Для вершины v и множества $V' \subset V$ окрестность исключения N_{ext} вершины v относительно V' определяется как $N_{ext}(v, V') = Nei(\{v\}) \setminus Nei(V')$. Иначе говоря, окрестность исключения – это множество тех вершин, которые входят в окрестность вершины v , но не являются смежными какой-либо вершине из множества V' .

Рассмотрим алгоритм метода Верника-Раше на Рисунке 2.3. Работа алгоритма начинается с перебора всех вершин i графа G , $i = 1, |V|$; эти вершины добавляются в соответствующие множества V_{sub} . После выполнения шага 3 алгоритма в каждом таком множестве V_{sub} содержится одна вершина. В дальнейшем во множествах V_{sub} будут содержаться две, три и т.д. вершины, входящие в соответствующий подграф определенного класса изоморфизма. Для добавления новых вершин во множества V_{sub} используется вспомогательное множество V_{ext} , в которое на первом шаге добавляются все вершины из ближайшего окружения вершины i – $Nei(v_i)$, причем номер добавленной в V_{ext}^i вершины должен быть больше, чем i . Далее работа алгоритма выполняется

рекурсивно. На каждом шаге рекурсивного вызова функции (функция *ExtendSubgraphs* на Рисунке 2.3) происходит расширение множества V_{sub} . Расширение множества V_{sub} выполняется до тех пор, пока число элементов в нем не достигнет значения k (т.е. будет найдена очередная реализация подграфа на k вершинах), или пока множество V_{ext} не станет пустым. Получаемые множества V_{sub} можно получить как результат построения корневого дерева. Структура дерева Верника-Раше изображена на Рисунке 2.4. На первом ярусе дерева Верника-Раше во множествах V_{sub}^i задаются все реализации связных подграфов на одной вершине, на втором – на двух вершинах V_{sub}^{ij} (т.е. все ребра $\{i, j\}$), на третьем – на трех вершинах V_{sub}^{ijk} и т.д.

Начало алгоритма метода Верника-Раше

Ввод: граф $G(V, E)$, целое $1 \leq k \leq |V|$

Вывод: все подграфы в графе G , которые содержат k вершин

1. Для всех $v \in V$ делай
2. $V_{ext} \leftarrow \{u \in Nei(v) : u > v\}$
3. $V_{sub} \leftarrow \{v\}$
4. Выполнить процедуру *ExtendSubgraphs*(V_{sub}, V_{ext}, v)

Конец алгоритма метода Верника-Раше

Начало процедуры *ExtendSubgraphs*(V_{sub}, V_{ext}, v)

E1. Если $|V_{sub}| = k$ то верни $G[V_{sub}]$ и **Конец процедуры *ExtendSubgraphs***

E2. Пока $V_{ext} \neq \emptyset$ делай

E3. Удалить случайно выбранную вершину w из V_{ext}

E4. $V'_{ext} \leftarrow V_{ext} \cup \{u \in N_{ext}(w, V_{sub}) : u > v\}$

E5. Выполнить процедуру *ExtendSubgraphs*($V_{sub} \cup \{w\}, V'_{ext}, v$)

Конец процедуры *ExtendSubgraphs*

Рисунок 2.3 – Алгоритм, реализующий метод Верника-Раше

В процедуре *ExtendSubgraphs* (V_{sub}, V_{ext}, v) множество V_{sub} расширяется путем добавления одной вершины из множества V_{ext} – вершины w . После этого множество V_{ext} изменяется следующим образом: во-первых, из него удаляется добавленная во множество V_{sub} вершина w , во-вторых, в него добавляются вершины, метки которых больше метки вершины v , и которые находятся в окрестности исключения w относительно множества V_{sub} .

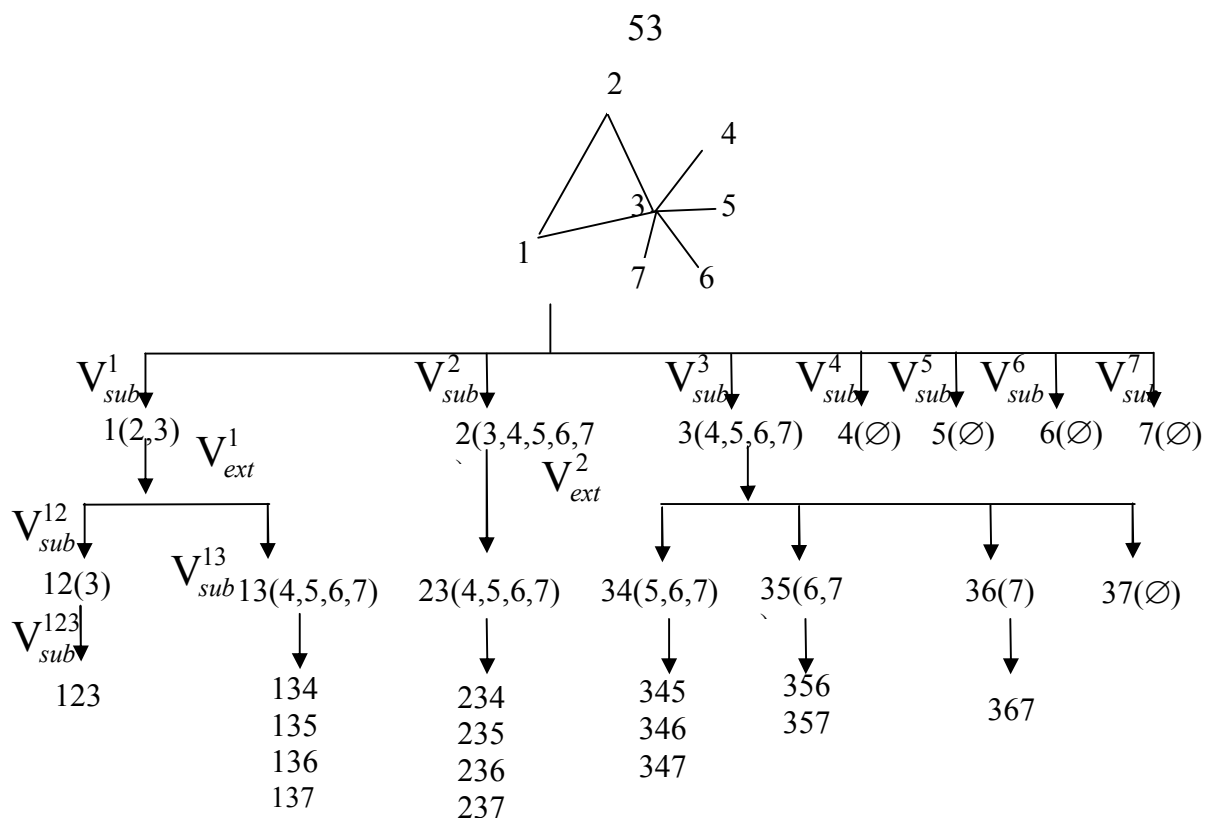


Рисунок 2.4 – Дерево Верника-Раше

Предложенный метод перебирает все подграфы заданного размера, не допуская перечисления одного и того же подграфа больше одного раза.

2.1.4.2 Статистический метод Верника-Раше

Если в алгоритме метода Верника-Раше заменить «Выполнить процедуру» в строках 4 и E5 на «Выполнить процедуру с вероятностью p_d », где $d = 1$ на шаге 4 и $d = |V_{sub}| + 1$ на шаге E5, то мы получим алгоритм статистического метода Верника-Раше [126].

Рассмотрим четыре сценария отсечения ветвей дерева Верника-Раше при оценке частот встречаемости подграфов на четырех вершинах:

- сценарий (0,3) – вероятность $p_d = 0,3$, $d = 1, 2, 3, 4$;
- сценарий (0,5) – вероятность $p_d = 0,5$, $d = 1, 2, 3, 4$;
- сценарий (0,7) – вероятность $p_d = 0,7$, $d = 1, 2, 3, 4$;
- сценарий (*) – вероятность $p_d = 1$, $d = 1, 2, 3$, $p_4 = 100\,000/|S|$, где $|S|$ – общее число подграфов на четырех вершинах в графе (должно быть известно).

Первые три сценария соответствуют ситуациям, когда на каждом ярусе дерева Верника-Раше просматривается в среднем 30%, 50% и 70% ветвей дерева. Сценарий (*) означает то, что из всех подграфов на четырех вершинах будет

выбрано в среднем 100 000 подграфов равновероятно на 4-м ярусе дерева Верника-Раше. Возможность в сценарии (*) установить вероятность отсечения ветвей на последнем ярусе дерева Верника-Раше возникает, так как известно общее число подграфов $|S|$ в графе (в общем случае оно неизвестно и контролировать точность расчетов статистических оценок для частот встречаемости подграфов в методе Верника-Раше нельзя). В Таблице 2.2 представлены результаты расчета статистических оценок для частот встречаемости в ориентированном графе, в Таблице 2.3 – в неориентированном графе.

Таблица 2.2 – Результаты расчета статистических оценок для частот встречаемости подграфов на четырех вершинах в смешанном графе Pathway Commons

КИ	Точный расчет	(0,7)	(0,5)	(0,3)	(*)	$\Delta(0,7)$	$\Delta(0,5)$	$\Delta(0,3)$	$\Delta(*)$
3	0,089005	0,001593	0,001787	0,002435	0,088091	0,086497	0,086303	0,085655	0,000914
7	0,117124	0,052438	0,046053	0,049795	0,115531	0,063093	0,069478	0,065736	0,001593
12	0,045387	0,002717	0,002401	0,002998	0,046197	0,043479	0,043796	0,043199	0,00081
13	0,086977	0,038461	0,036137	0,040013	0,088512	0,050051	0,052374	0,048499	0,001535
14	0,010193	0,000554	0,000552	0,00079	0,010345	0,00979	0,009792	0,009554	0,000152
17	0,007554	0,000886	0,000784	0,000949	0,007775	0,006889	0,00699	0,006825	0,00022
19	0,000879	0,000199	0,000184	0,000231	0,000901	0,000702	0,000717	0,00067	2,15E-05
20	0,00055	4,69E-05	4,49E-05	6,65E-05	0,000651	0,000604	0,000606	0,000585	0,000102
24	0,112958	0,069641	0,065138	0,070619	0,114054	0,044413	0,048916	0,043434	0,001096
29	0,016448	0,00386	0,003308	0,00326	0,017017	0,013157	0,013709	0,013757	0,000569
35	0,003987	0,00018	0,000157	0,000197	0,004059	0,003878	0,003902	0,003861	7,17E-05
41	0,060026	0,029645	0,027249	0,027604	0,060024	0,030379	0,032775	0,032421	1,93E-06
42	0,004097	0,001098	0,000946	0,000972	0,00407	0,002972	0,003124	0,003097	2,74E-05
47	0,007754	0,002428	0,002181	0,002345	0,008249	0,005822	0,006069	0,005904	0,000495
52	0,000819	0,000208	0,000197	0,000243	0,000802	0,000595	0,000606	0,000559	1,72E-05
58	0,00062	4,74E-05	4,52E-05	6,97E-05	0,000642	0,000595	0,000597	0,000572	2,24E-05
66	0,000729	3,37E-05	2,88E-05	3,62E-05	0,000872	0,000838	0,000843	0,000836	0,000142
67	0,00059	3,81E-05	3,13E-05	4,13E-05	0,000723	0,000685	0,000691	0,000681	0,000133
76	0,411869	0,77877	0,796057	0,778623	0,409504	0,369266	0,386553	0,369119	0,002365
77	0,018766	0,01637	0,01598	0,017809	0,01857	0,0022	0,00259	0,000761	0,000196
78	0,000779	8,77E-05	8,41E-05	0,00012	0,000712	0,000624	0,000628	0,000592	6,74E-05
80	0,000809	0,000282	0,000265	0,000307	0,000753	0,000471	0,000488	0,000446	5,64E-05
82	0,000929	0,000315	0,000294	0,000345	0,000843	0,000527	0,000549	0,000498	8,67E-05
83	0,001149	0,000102	9,6E-05	0,00013	0,001106	0,001004	0,00101	0,000976	4,35E-05
Сумма модулей отклонений от точных значений ->						0,738533	0,773106	0,738239	0,01074

В Таблице 2.2 и в Таблице 2.3 приведены классы изоморфизма (КИ) подграфов, относительные частоты встречаемости подграфов при точном расчете. Кроме того, приведены значения модулей отклонений от точных значений по каждому классу изоморфизма при использовании различных сценариев $\Delta(0,7)$, $\Delta(0,5)$, $\Delta(0,3)$ и $\Delta(*)$. Также по всем подграфам рассчитана сумма модулей отклонений полученных

статистических оценок от точных значений частот встречаемости. Расчеты выполнены с использованием библиотеки *igraph*.

Таблица 2.3 – Результаты расчета статистических оценок для частот встречаемости связанных подграфов на четырех вершинах в неориентированном графе PPI_HS

КИ	Точный расчет	(0,7)	(0,5)	(0,3)	(*)	$\Delta(0,7)$	$\Delta(0,5)$	$\Delta(0,3)$	$\Delta(*)$
2	0,674873	0,781392	0,784852	0,78437	0,673386815	0,106519	0,109979	0,109496306	0,001487
4	0,279971	0,191768	0,18864	0,188332	0,281598485	0,088203	0,091331	0,091639009	0,001628
6	0,040031	0,025261	0,024909	0,025534	0,040131546	0,01477	0,015122	0,014496761	0,000101
7	0,002713	0,000846	0,000846	0,000954	0,002700683	0,001867	0,001866	0,001758693	1,18E-05
8	0,002164	0,000695	0,000713	0,000765	0,001963227	0,001469	0,001452	0,001399102	0,000201
9	0,000248	3,81E-05	3,94E-05	4,51E-05	0,000219244	0,00021	0,000208	0,000202741	2,86E-05
Сумма модулей отклонений от точных значений ->						0,213038	0,219958	0,218992611	0,003457

Анализируя Таблицу 2.2 и Таблицу 2.3 можно увидеть, что увеличение числа найденных подграфов в сценариях (0,3), (0,5) и (0,7) не ведет к значимому увеличению точности расчетов. В Таблице 2.4 представлены данные по другим графам сетей, которые подтверждают данную гипотезу. Единственным корректным способом увеличивать точность расчетов является использование сценария (*), что фактически требует построения всего дерева Верника-Раше, кроме последнего яруса.

Таблица 2.4 – Сумма модулей отклонений статистических оценок для частот встречаемости подграфов от точных значений частот встречаемости. Время t выполнения расчетов приводится в секундах

Граф	$\Delta(0,7) / t$	$\Delta(0,5) / t$	$\Delta(0,3) / t$	$\Delta(*) / t$
Regulon	0,148679663 /151	0,147961094 /36,6	0,148062613 /5,3	0,008874712 /121
GenReg	0,329372548 /913	0,353253 /226	0,643309633 /32	0,352853366 /677
Pathway Commons	0,738532524 /72838	0,77310553 /19970	0,738238616 /2749	0,010739933 /3337
PPI_HS	0,213037662 /2557	0,219957614 /658	0,218992611 /94,7	0,003456787 /338

В Таблице 2.5 представлены значения частот встречаемости подграфов в каждом из сценариев.

Таблица 2.5 – Доля рассмотренных подграфов на четырех вершинах от общего числа подграфов на четырех вершинах в графах для различных сценариев работы статистического метода Верника-Раше

Граф	(0,7)	(0,5)	(0,3)	(*)
Regulon	0,056902888	0,001772095	0,01303791	1,53853E-05
GenReg	0,051457186	0,011377555	0,00157634	3,07911E-06
Pathway Commons	0,101014285	0,024725437	0,003259303	4,40305E-07
PPI_HS	0,122114761	0,028992322	0,003825863	5,74503E-06

Представленные результаты можно объяснить тем, что полученная статистическая оценка для частот встречаемости является смещенной¹. Проиллюстрируем это на примере. Как можно видеть в графе на Рисунке 2.4 имеются 10 различных подграфов на трех вершинах, из них 4 – «треугольник». Истинное значение частоты встречаемости «треугольника» $C_{\Delta} = 4/10$. При этом, если вероятность выполнения шага 4 в алгоритме на Рисунке 2.4 равна p , то математическое ожидание частоты встречаемости «треугольника» по формуле полной вероятности будет

$$M[C_{\Delta}^{BP}] = \sum_{i=1}^{N_C} P_i \cdot C_i, \quad (2.1)$$

где N_C – количество различных значений относительных частот встречаемости «треугольника», которые могут быть получены в результате проведения статистического эксперимента;

P_i – вероятность получения относительной частоты встречаемости «треугольника», равной C_i .

При подсчете суммы в формуле (2.1) можно рассматривать только ненулевые множители C_i . Для графа на Рисунке 2.5 таких значений всего четыре. Во-первых, относительная частота встречаемости «треугольника» будет ненулевой, если в результате выполнения шага 4 алгоритма на Рисунке 2.3 останется только вершина «1», вероятность этого события равна $p(1-p)^2$, а значения относительной частоты встречаемости «треугольника» при этом будет равно 4/6. Во-вторых, относительная частота встречаемости «треугольника» будет ненулевой, если в

¹ Этот результат опровергает теорему [125] о несмещенности статистических оценок для частот встречаемости подграфов при использовании статистического метода Верника-Раше

результате выполнения шага 4 останутся вершины «1» и «2», вероятность этого события равна $p^2(1-p)$, а значения относительной частоты встречаемости «треугольника» при этом равно 4/9. В-третьих, относительная частота встречаемости «треугольника» будет ненулевой, если в результате выполнения шага 4 останутся вершины «1» и «3», вероятность этого события также равна $p^2(1-p)$, а значение относительной частоты встречаемости «треугольника» при этом будет равно 4/7. Наконец, относительная частота встречаемости «треугольника» будет ненулевой, если в результате выполнения шага 4 останутся вершины «1», «2» и «3», вероятность этого события равна p^3 , а значение относительной частоты встречаемости «треугольника» при этом будет равно 4/10.

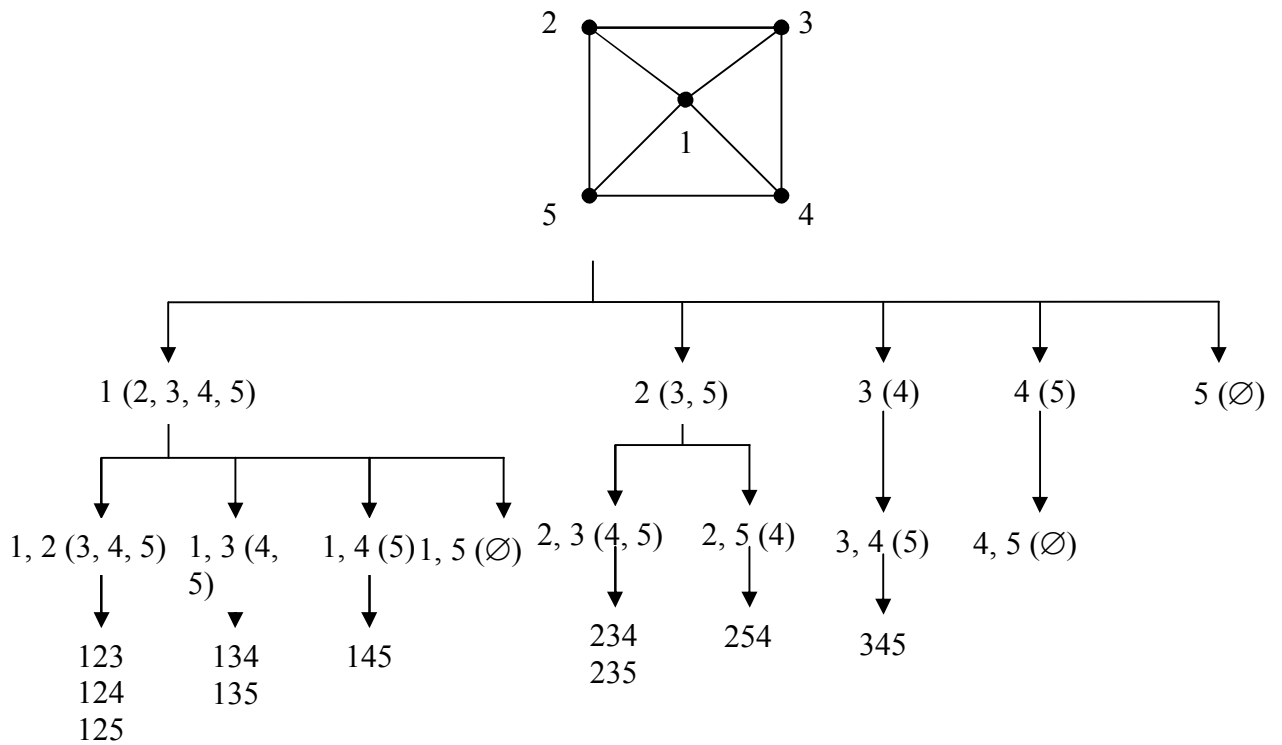


Рисунок 2.5 – Дерево Верника-Раше

Соответственно, математическое ожидание частоты встречаемости «треугольника» C_{Δ}^{BP} будет

$$M[C_{\Delta}^{BP}] = \sum_{i=1}^{N_C} p_i \cdot C_i = \frac{4}{6} p(1-p)^2 + \left(\frac{4}{9} + \frac{4}{7}\right) p^2(1-p) + \frac{4}{10} p^3.$$

Из графика, изображенного на Рисунке 2.6 можно видеть, что при различных значениях вероятности p частота встречаемости «треугольника»

оказывается заниженной, достигая истинного значения $C_\Delta = 4/10$ при вероятности выполнения шага 4 алгоритма на Рисунке 2.3 равной единице, $p_1 = 1$.

Таким образом, оценка частот встречаемости подграфов, получаемая статистическим методом Верника-Раше, в общем случае, является смещенной.

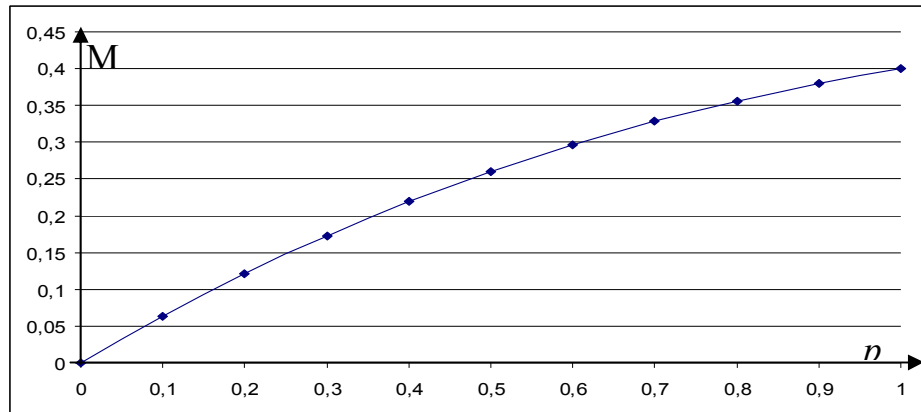


Рисунок 2.6 – Зависимость математического ожидания M частоты встречаемости «треугольника» от вероятности p выполнения шага 4 алгоритма на Рисунке 2.4

Представленное в работе [125] доказательство теоремы о несмещенности оценки в методе Верника-Раше в общем случае является ошибочным. Это доказательство основывается на том, что вероятность «сохранения» любого подграфа на k вершинах, в результате отсечений выборки, выполненной в соответствии с методом Верника-Раше, одинакова для всех подграфов на заданном числе вершин. Что действительно так, поскольку эта вероятность для любого подграфа зависит от одинаковых значений вероятностей отсечения выборки на заданном ярусе дерева Верника-Раше. Однако ошибочно полагать, что оценку относительной частоты встречаемости подграфа корректно рассчитывать как отношение числа найденных подграфов i -го класса изоморфизма к общему числу найденных подграфов на заданном числе вершин. Так, найденное число реализаций «треугольника» и найденное общее число подграфов на трех вершинах в статистическом эксперименте метода Верника-Раше – это зависимые случайные величины. При отсечении ветвей дерева Верника-Раше подграфы разного класса изоморфизма, которые будут найдены после окончания алгоритма, отсекаются из выборки в разной пропорции, зависящей от конкретной структуры графа. Единственная корректная реализация статистического метода Верника-Раше, при

которой отсутствует смещение оценок (поскольку все реализации подграфов различных классов изоморфизма, полученные в результате выполнения метода Верника-Раше, независимы), это реализация, когда отсекаются ветви дерева Верника-Раше только на последнем ярусе.

Выделим следующие особенности статистического метода Верника-Раше.

1. Оценки частот встречаемости подграфов, получаемые методом Верника-Раше, в общем случае, являются смещенными.
2. Статистический метод Верника-Раше можно корректно использовать только при отсеке нии подграфов на последнем ярусе дерева Верника-Раше.
3. В статистическом методе Верника-Раше отсутствует возможность контролировать точность расчета, размером выборки найденных подграфов можно управлять только косвенно.

Однако из всех существующих методов расчета статистических оценок для частот встречаемости подграфов метод Верника-Раше (Rand-ESU) позволяет, при корректном его использовании, рассчитывать частоты встречаемости подграфов в смешанных графах. Поэтому будем использовать этот метод для сравнения с ним метода, разработанного в рамках диссертационной работы.

2.1.5 Методы случайного выбора пути

Этот класс методов основан на методе Монте-Карло, и используется для получения статистических оценок относительных и абсолютных частот встречаемости подграфов на трех, четырех и пяти вершинах в неориентированных графах. Всего существует 21 класс изоморфизма для неориентированных подграфов на пяти вершинах, на четырех вершинах – 6, на трех – 2 (Рисунок 2.7).

В пионерской работе [113] по разработке методов данного класса предлагается использовать случайный равновероятный выбор пути из двух ребер и трех вершин («3-путь») и оценивать долю «треугольников» в графе. Зная общее число всех «3-путей» возможно оценить и общее число каждого из двух неориентированных подграфов на трех вершинах. В работах [80] и [130, 25] предлагается использовать случайный выбор пути на трех ребрах («4-путь») с последующим нахождением доли подграфов пяти из шести возможных классов изоморфизма. Число подграфов,

относящихся к шестому классу изоморфизма, вычисляется численно на основе данных о частотах встречаемости других подграфов и общего числа «4-путей».

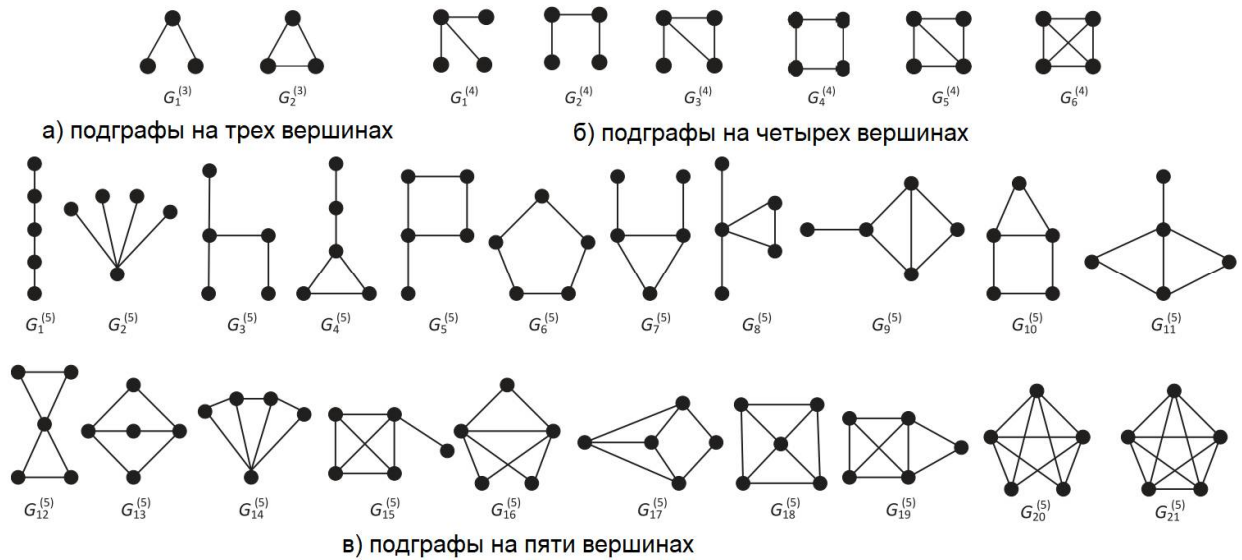


Рисунок 2.7 – Классы изоморфизма связных неориентированных подграфов на трех, четырех и пяти вершинах [123]

Между множеством $Z(G)$ всех реализаций «3-пути» и множеством X классов изоморфизма имеется сюръективное отображение (Рисунок 2.8). Причем каждому элементу множества S реализаций подграфа заданного класса изоморфизма всегда соответствует определенное количество *koef* реализаций «3-путей» из Z . Так, «треугольнику» всегда соответствуют три реализации «3-пути» (Рисунок 2.8).

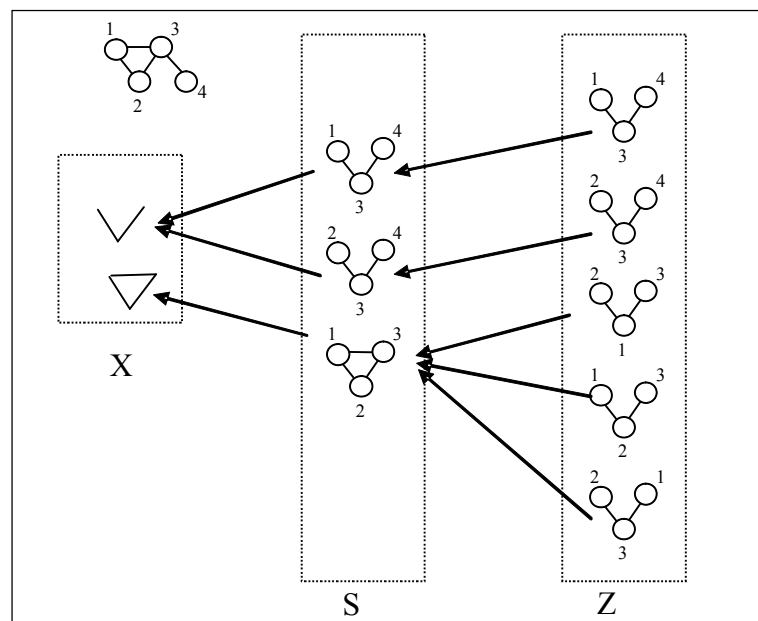


Рисунок 2.8 – Сюръекция $Z(G) \rightarrow S(G) \rightarrow X$, где Z – множество реализаций «3-путей», $S(G)$ – множество реализаций подграфов, X – классы изоморфизма

Аналогично при использовании «4-путей» каждому подграфу класса изоморфизма $G_i^{(4)}$ соответствует некоторое число вхождений подграфов класса изоморфизма $G_j^{(4)}$. Можно показать истинность следующего линейного соотношения:

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 2 & 4 \\ 0 & 1 & 2 & 4 & 6 & 12 \\ 0 & 0 & 1 & 0 & 4 & 12 \\ 0 & 0 & 0 & 1 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 & 6 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} n_1^{(4)} \\ n_2^{(4)} \\ n_3^{(4)} \\ n_4^{(4)} \\ n_5^{(4)} \\ n_6^{(4)} \end{pmatrix} = \begin{pmatrix} z_1^{(4)} \\ z_2^{(4)} \\ z_3^{(4)} \\ z_4^{(4)} \\ z_5^{(4)} \\ z_6^{(4)} \end{pmatrix}, \quad (2.2)$$

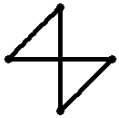
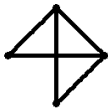
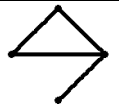
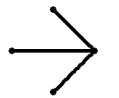
где a_{ij} – элемент матрицы, который задает число вхождений подграфа класса $G_i^{(4)}$ в подграф класса $G_j^{(4)}$, так, значение $a_{2,4} = 4$ означает, что в подграф «цикл на четырех вершинах» $G_4^{(4)}$ подграф «4-путь» $G_2^{(4)}$ входит четыре раза;

$n_i^{(4)}$ – искомое число подграфов класса изоморфизма $G_i^{(4)}$;

$z_i^{(4)}$ – число реализаций подграфа $G_i^{(4)}$ в графе.

В соответствии с формулой (2.2) в методе получения оценок для частот встречаемости подграфов необходимо использовать соответствующие понижающие коэффициенты (Таблица 2.6).

Таблица 2.6 – Поправочные коэффициенты при расчете частот встречаемости подграфов на основе метода случайного выбора «4-пути»

Подграф	Поправочный коэффициент	Подграф	Поправочный коэффициент
	1/4		1/6
	1/2		1/12
	–		1

Алгоритм расчета статистических оценок для абсолютных частот встречаемости подграфов методом случайного выбора пути представлен на Рисунке 2.9.

НАЧАЛО

Ввод: граф $G(V, E)$, массив поправочных коэффициентов $coef[i]$, $i \in X$, где X – множество классов изоморфизма

Вывод: статистические оценки частот встречаемости подграфов $n[i]$, $i \in X$

1. Для всех $i \in X$ делай $n[i] \leftarrow 0$; $C[i] \leftarrow 0$;
2. Рассчитать общее число путей N
3. Для $j = 1, \dots, N^+$ выполнить
 - 3.1. Выбрать равновероятно путь C_j , содержащий множество вершин V'
 - 3.2. $Num \leftarrow GetISO[G[V']]$ // получить класс изоморфизма $Num \in X$
 - 3.3. $C[Num] = C[Num] + 1$;
4. Для всех $i \in X$ делай $n[i] = N C[i] / N^+ / coef[i]$.

КОНЕЦ

Рисунок 2.9 – Алгоритм расчета статистических оценок для абсолютных частот встречаемости подграфов

В алгоритме на Рисунке 2.9 на шаге 3.1 равновероятно выбираются реализации случайного пути, на шаге 3.2 определяется класс изоморфизма найденного подграфа и на шаге 3.3 наращивается счетчик найденных реализаций подграфов. После выполнения статистического эксперимента (шаг 3 алгоритма на Рисунке 2.9) на шаге 4 с учетом поправочных коэффициентов рассчитываются оценки абсолютных частот встречаемости подграфов. Таким образом могут быть получены оценки частоты встречаемости для всех классов изоморфизма. При этом для класса $G_1^{(4)}$ выполнять эксперимент не обязательно: для него оценку частоты встречаемости можно рассчитать в соответствии с соотношением (2.2) по формуле [80]:

$$\hat{n}_1^{(4)} = \sum_{v \in V'} C_3^{k_v} - \hat{n}_3^{(4)} - 2\hat{n}_5^{(4)} - 4\hat{n}_6^{(4)},$$

где $\hat{n}_i^{(4)}$ – статистические оценки частот встречаемости подграфов, относящихся к соответствующим классам изоморфизма $G_i^{(4)}$,

$$\sum_{v \in V'} C_3^{\deg(v)} - \text{число всех «4-ветвлений» (остовных деревьев изоморфных } G_1^{(4)}).$$

Особенностью расчетов, представленных в работе [123], для получения статистических оценок для абсолютных частот встречаемости подграфов на пяти

вершинах, является использование двух различных статистических экспериментов: случайного выбора пути из четырех ребер («4-путь») и случайного выбора остовного дерева, относящегося к классу изоморфизма $G_3^{(5)}$.

Статистическую оценку для абсолютной частоты встречаемости подграфов, относящихся к классу изоморфизма $G_2^{(5)}$, в соответствии с [123] можно получить по формуле:

$$\hat{n}_2^{(5)} = \sum_{v \in V} C_4^{k_v} - \hat{n}_8^{(5)} - \hat{n}_{11}^{(5)} - \hat{n}_{12}^{(5)} - \hat{n}_{14}^{(5)} - \hat{n}_{15}^{(5)} - 2\hat{n}_{16}^{(5)} - \hat{n}_{18}^{(5)} - 2\hat{n}_{19}^{(5)} - 3\hat{n}_{20}^{(5)} - 5\hat{n}_{21}^{(5)},$$

где $\hat{n}_i^{(5)}$ – статистические оценки частот встречаемости подграфов, относящихся к соответствующим классам изоморфизма $G_i^{(5)}$,

$$\sum_{v \in V} C_4^{\deg(v)} - \text{число всех «5-ветвлений» (подграфов изоморфных } G_2^{(5)}).$$

2.2 Метод случайного выбора остовных деревьев

Предложим метод случайного выбора остовных деревьев, обобщающий методы случайного выбора пути и позволяющий рассчитывать частоты встречаемости подграфов не только в неориентированных, но и в смешанных графах.

Пусть выборочное пространство определяется как множество реализаций одного из возможных остовных деревьев на заданном числе вершин и наделенных равными вероятностями выбора. Направленность ребер не учитывается. В графе на трех вершинах существует только один класс изоморфизма остовных деревьев (Рисунок 2.10), на четырех вершинах – 2 класса, на пяти – 3, на шести – 6.

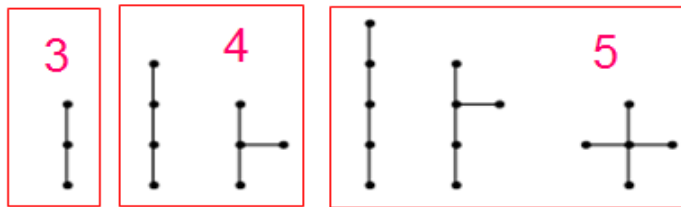


Рисунок 2.10 – Остовные деревья на трех, четырех и пяти вершинах

В методе случайного выбора остовных деревьев сначала выполняется случайный выбор реализации остовного дерева заданного класса изоморфизма. Эта

реализация остовного дерева определяет множество k вершин. Затем определяется подграф соответствующий этому множеству в графе (Рисунок 2.11). При определении

класса изоморфизма подграфа учитывается направление ребер.

Поскольку общее число N реализаций остовного дерева в графе при использовании метода случайного выбора остовных деревьев рассчитывается точно, несмещенная оценка $\hat{n}_i$ реализаций подграфа в графе определяется по формуле

$$\hat{n}_i = \hat{q}_i N / \text{coef}_i,$$

где $\hat{q}_i$ – относительная частота встречаемости подграфа i -го класса изоморфизма при случайном равновероятном выборе заданного остовного дерева, coef_i – число вхождений реализаций заданного остовного дерева в i -й подграф.

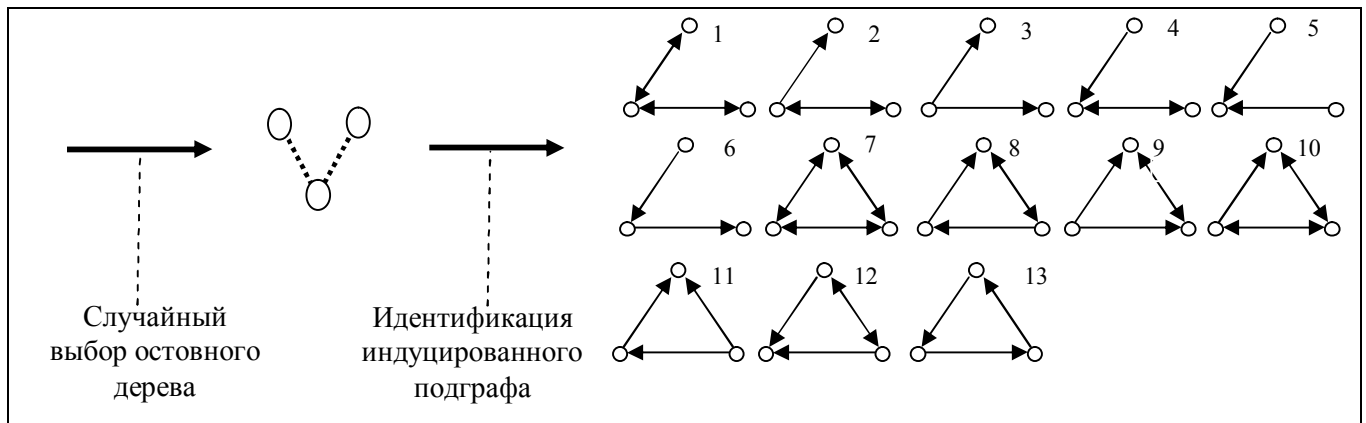


Рисунок 2.11 – Иллюстрация работы метода случайного выбора остовных деревьев для расчета частот встречаемости подграфов на трех вершинах в смешанном графе

Заметим, что при использовании остовных деревьев на четырех вершинах могут быть получены одна или две различные статистические оценки для частот встречаемости подграфов (по числу используемых остовных деревьев и, соответственно, числу статистических экспериментов). При использовании остовных деревьев на пяти вершинах могут быть получены одна, две или три различные оценки частот встречаемости подграфов.

2.3 Статистические эксперименты по двум различным остовным деревьям

На Рисунке 2.12 изображена диаграмма Венна, отражающая охват экспериментами при расчете частот встречаемости подграфов на четырех вершинах в смешанных графах. Всего существует 198 различных классов изоморфизма на четырех связанных вершинах.

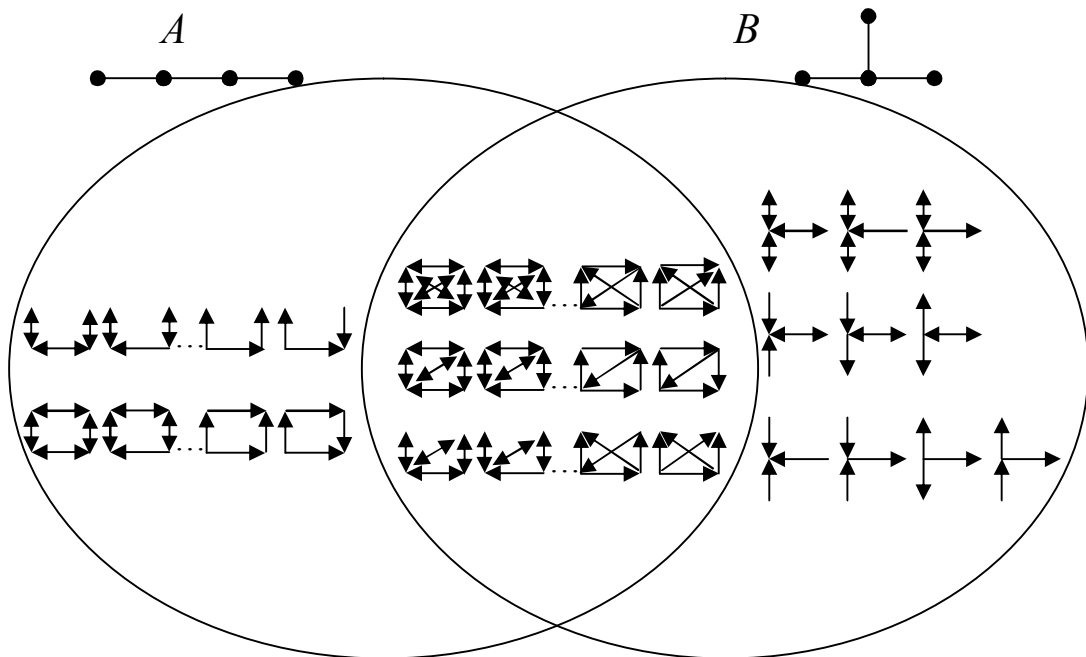


Рисунок 2.12 – Диаграмма Венна охвата остовными деревьями подграфов, относящихся к различным классам изоморфизма, A – множество подграфов, которые находятся на остовном дереве «4-путь», B – множество подграфов, которые находятся на остовном дереве «4-ветвление»

При расчете встречаемости подграфов на пяти вершинах с использованием трех различных остовных деревьев «5-путь», «5-ветвление» и «Т-дерево» для ряда подграфов также реализуются две оценки для частот встречаемости подграфов (Рисунок 2.13).

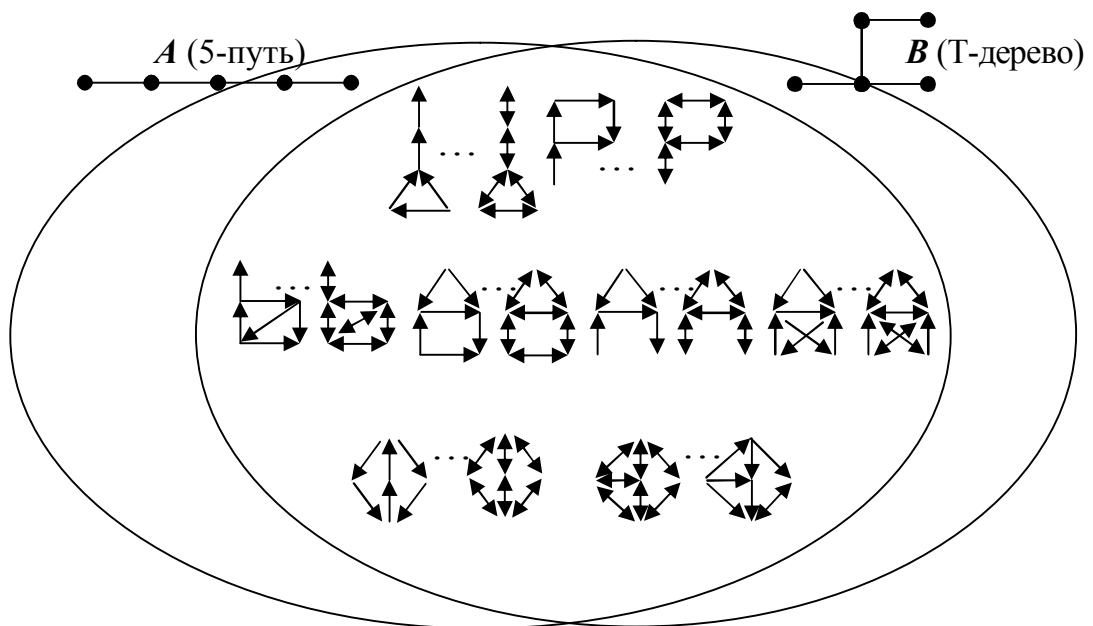


Рисунок 2.13 – Подграфы на пяти вершинах, для которых могут быть получены две различные статистические оценки

Рассмотрим случай получения двух различных статистических оценок подробнее. Предположим, что в результате N_A^+ случайных выборов реализаций остовных деревьев класса изоморфизма A (Рисунок 2.12 и Рисунок 2.13) подграф обнаружен C_A раз. В результате N_B^+ случайных выборов реализаций остовных деревьев класса изоморфизма B , этот же подграф будет обнаружен C_B раз. Относительные частоты встречаемости подграфа в проведенных экспериментах будут $\hat{q}_A = \frac{C_A}{N_A^+}$ и $\hat{q}_B = \frac{C_B}{N_B^+}$ соответственно.

Оценку абсолютной частоты встречаемости подграфа можно получить по результатам эксперимента как $\hat{n}^{[1]} = \hat{q}_A N_A / \text{coef}_A = \frac{C_A}{N_A^+} N_A / \text{coef}_A$, где N_A – точное число всех реализаций остовного дерева, относящихся к классу изоморфизма A . Аналогично при случайном выборе реализаций остовного дерева B получим: $\hat{n}^{[2]} = \hat{q}_B N_B / \text{coef}_B = \frac{C_B}{N_B^+} N_B / \text{coef}_B$, где N_B – точное число реализаций остовного дерева B в графе. Как можно видеть, оценки $\hat{n}^{[1]}$ и $\hat{n}^{[2]}$ также являются случайными величинами. Значения coef_A и coef_B – это соответствующие поправочные коэффициенты для искомого подграфа при использовании соответствующего остовного дерева.

Итоговую оценку абсолютной частоты встречаемости подграфа представим в виде линейной комбинации оценок, полученных при использовании различных остовных деревьев:

$$\hat{n} = \hat{n}^{[1]} + \lambda(\hat{n}^{[2]} - \hat{n}^{[1]}), \text{ где } 0 < \lambda < 1. \quad (2.3)$$

Далее рассмотрим такой выбор значения параметра λ , который минимизирует относительную или абсолютную погрешность полученной оценки.

Заметим, что случайные величины $\hat{n}^{[1]}$ и $\hat{n}^{[2]}$ при большом числе испытаний имеют нормальное распределение, что вытекает из центральной предельной теоремы, поскольку эти оценки получены при использовании схемы Бернулли, причем выполняется большое число независимых равновероятных выборов

подграфов. Величина $\hat{n}$ тоже характеризуется нормальным законом распределения, поскольку строится как линейная комбинация двух случайных величин с нормальным законом распределения. Для нормального закона распределения известно, что вероятность того, что значение случайной величины отклонится от ее математического ожидания не больше, чем на три среднеквадратических отклонения, не превышает 0,028, т.е. пренебрежимо мала. Поэтому в качестве абсолютной погрешности для случайной величины $\hat{n}$ будем рассматривать величину 3σ . В качестве аналога относительной погрешности для случайной величины $\hat{n}$ будем рассматривать ее коэффициент вариации $v[\hat{n}] = \sigma[\hat{n}]/M[\hat{n}]$, а в качестве самой относительной погрешности – величину $3v$.

2.3.1 Минимизация абсолютной погрешности

Для минимизации абсолютной погрешности расчета статистической оценки для частот встречаемости параметр λ в выражении (2.3) определим из условия минимизации дисперсии оценки для частоты встречаемости подграфа $D[\hat{n}(\lambda)]$.

Случайная величина $\hat{n}$ определяется через линейную комбинацию независимых случайных величин в выражении (2.3), ее математическое ожидание $M[\hat{n}]$ при любом $0 < \lambda < 1$ будет

$$\begin{aligned} M[\hat{n}] &= M[\hat{n}^{[1]} + \lambda(\hat{n}^{[2]} - \hat{n}^{[1]})] = M[(1-\lambda)\hat{n}^{[1]} + \lambda\hat{n}^{[2]}] = (1-\lambda)M[\hat{n}^{[1]}] + \lambda M[\hat{n}^{[2]}] = \\ &= (1-\lambda)M_A + \lambda M_B, \end{aligned}$$

а дисперсия

$$D[\hat{n}(\lambda)] = D[(1-\lambda)\hat{n}^{[1]}] + D[\lambda\hat{n}^{[2]}] = (1-\lambda)D[\hat{n}^{[1]}] + \lambda D[\hat{n}^{[2]}] = (1-\lambda)^2 D_A + \lambda^2 D_B. \quad (2.4)$$

Оптимальное значение λ_D определим из выражения (2.4), дифференцируя его по λ , и приравнивая полученное выражение к нулю:

$$((1-\lambda_D)^2 D_A + \lambda_D^2 D_B)' = -2D_A + 2\lambda_D D_A + 2\lambda_D D_B = 0.$$

Откуда выразим искомое λ_D :

$$\lambda_D = \frac{D_A}{D_B + D_A}. \quad (2.5)$$

Найденное значение λ_D действительно определяет минимум функции $D[\hat{n}(\lambda)]$, поскольку вторая производная $D[\hat{n}(\lambda)]'' = 2D_A + 2D_B$ положительна, в том числе в точке λ_D . Случай, когда дисперсии D_A и D_B одновременно равны нулю следует рассмотреть отдельно. Эта ситуация возможна только тогда, когда все связные подграфы в исследуемом графе являются реализацией искомого подграфа и находятся одновременно как при использовании остова дерева A , так и при использовании остова дерева B , в этом случае $M[\hat{n}] = M[\hat{n}^{[1]}] = M[\hat{n}^{[2]}]$.

Выведем расчетные формулы для дисперсий абсолютных частот встречаемости подграфа при использовании остова дерева A и B – D_A и D_B соответственно. Заметим, что индикаторные величины C_A и C_B определяют вероятность встречаемости исследуемого подграфа в экспериментах с использованием соответствующих остовных деревьев: $P_A \approx \hat{q}_A = \frac{C_A}{N_A^+}$ и

$P_B \approx \hat{q}_B = \frac{C_B}{N_B^+}$. Отсюда можно получить выражение для оценки дисперсии D_A :

$$\begin{aligned} D_A = D[\hat{n}] &\cong \hat{D}[\hat{q}_A N_A / \text{coef}_A] = \frac{N_A^2}{\text{coef}_A^2} D\left[\frac{C_A}{N_A^+}\right] \approx \frac{N_A^2}{\text{coef}_A^2 (N_A^+)^2} \sum_{i=1}^{N_A^+} P_A (1 - P_A) = \\ &= \frac{N_A^2}{\text{coef}_A^2 (N_A^+)^2} N_A^+ \frac{C_A}{N_A^+} \left(1 - \frac{C_A}{N_A^+}\right) = \frac{N_A^2}{\text{coef}_A^2 (N_A^+)^2} C_A \left(1 - \frac{C_A}{N_A^+}\right), \end{aligned}$$

аналогично вычисляется $D_B = \frac{N_B^2}{\text{coef}_B^2 (N_B^+)^2} C_B \left(1 - \frac{C_B}{N_B^+}\right)$.

Таким образом, в известных в результате статистического эксперимента обозначениях выражение (2.5) для λ_D имеет следующий вид:

$$\lambda_D = \frac{\frac{N_A^2}{\text{coef}_A^2 (N_A^+)^2} C_A \left(1 - \frac{C_A}{N_A^+}\right)}{\frac{N_B^2}{\text{coef}_B^2 (N_B^+)^2} C_B \left(1 - \frac{C_B}{N_B^+}\right) + \frac{N_A^2}{\text{coef}_A^2 (N_A^+)^2} C_A \left(1 - \frac{C_A}{N_A^+}\right)}.$$

2.3.2 Минимизация относительной погрешности

Для минимизации относительной погрешности расчета статистической оценки частоты встречаемости подграфа определим параметр λ , при котором будет достигаться минимальное значение коэффициента вариации этой оценки:

$$v^2 = \frac{D[\hat{n}(\lambda)]}{M^2[\hat{n}(\lambda)]} \rightarrow \min, \quad 0 < \lambda < 1. \quad (2.6)$$

Найдем оптимальное значение $\lambda_{к.в.}$ из условия (2.6), дифференцируя его по λ , и приравняв полученное выражение к нулю. Найдем производную $(v^2[\lambda])'$:

$$\begin{aligned} (v^2)' &= \left(\frac{D[\hat{n}(\lambda)]}{M^2[\hat{n}(\lambda)]} \right)' = \left[\frac{D[(1-\lambda)\hat{n}^{[1]}] + D[\lambda\hat{n}^{[2]}]}{M^2[(1-\lambda)\hat{n}^{[1]} + \lambda\hat{n}^{[2]}]} \right]' = \left[\frac{(1-\lambda)^2 D_A + \lambda^2 D_B}{((1-\lambda)M_A + \lambda M_B)^2} \right]' = \\ &= \frac{2_B \lambda - 2D_A(1-\lambda)}{(M_A(1-\lambda) + M_B \lambda)^2} - \frac{2(M_B - M_A)D_A(1-\lambda)^2 + D_B \lambda^2}{(M_A(1-\lambda) + M_B \lambda)^3}. \end{aligned}$$

Упростим последнее выражение и приравняем к нулю для получения оптимального значения $\lambda_{к.в.}$:

$$\begin{aligned} &\frac{(2D_B \lambda_{к.в.} + 2D_A(1-\lambda_{к.в.}))(M_A(1-\lambda_{к.в.}) + M_B \lambda_{к.в.})}{(M_A(1-\lambda_{к.в.}) + M_B \lambda_{к.в.})^3} - \\ &\quad - \frac{2(M_B - M_A)(D_A(1-\lambda_{к.в.})^2 + D_B \lambda_{к.в.}^2)}{(M_A(1-\lambda_{к.в.}) + M_B \lambda_{к.в.})^3} = 0. \end{aligned} \quad (2.7)$$

Разделим равенство (2.7) на общий ненулевой множитель $2/(M_A(1-\lambda_{к.в.}) + M_B \lambda_{к.в.})^3$ и получим следующее уравнение

$$\begin{aligned} &M_A D_B \lambda_{к.в.}(1-\lambda_{к.в.}) + M_B D_B \lambda_{к.в.}^2 - M_A D_A(1-\lambda_{к.в.})^2 - M_B D_A \lambda_{к.в.}(1-\lambda_{к.в.}) - \\ &\quad - M_B D_A(1-\lambda_{к.в.})^2 - M_B D_B \lambda_{к.в.}^2 + M_A D_A(1-\lambda_{к.в.})^2 + M_A D_B \lambda_{к.в.}^2 = 0, \end{aligned}$$

а после упрощения

$$M_A D_B \lambda_{к.в.} - M_B D_A + M_B D_A \lambda_{к.в.} = 0.$$

Откуда выразим искомое $\lambda_{к.в.}$:

$$\lambda_{к.в.} = \frac{M_B D_A}{M_A D_B + M_B D_A}. \quad (2.8)$$

В известных в результате статистического эксперимента обозначениях

$$\lambda_{\kappa.в.} = \frac{\frac{C_B}{coef_B N_B^+} N_B \frac{N_A^2}{coef_A^2 (N_A^+)^2} C_A \left(1 - \frac{C_A}{N_A^+}\right)}{\frac{C_A}{coef_A N_A^+} N_A \frac{N_B^2}{coef_B^2 (N_B^+)^2} C_B \left(1 - \frac{C_B}{N_B^+}\right) + \frac{C_B}{coef_B N_B^+} N_B \frac{N_A^2}{coef_A^2 (N_A^+)^2} C_A \left(1 - \frac{C_A}{N_A^+}\right)}.$$

Докажем, что $\lambda_{\kappa.в.}$ соответствует минимуму функции, определяющей коэффициент вариации. Для этого получим вторую производную функции $v^2(\lambda)$ в точке $\lambda_{\kappa.в.}$ и покажем, что она положительна.

$$\begin{aligned} (v^2)'' &= \left(\frac{2D_B\lambda - 2D_A(1-\lambda)}{(M_A(1-\lambda) + M_B\lambda)^2} - \frac{2(M_B - M_A)(D_A(1-\lambda)^2 + D_B\lambda^2)}{(M_A(1-\lambda) + M_B\lambda)^3} \right)' = \\ &= \frac{2D_A + 2D_B}{(M_A(1-\lambda) + M_B\lambda)^2} + \frac{-4(D_B\lambda - D_A(1-\lambda))(M_B - M_A)}{(M_A(1-\lambda) + M_B\lambda)^3} - \\ &- 2(M_B - M_A) \left[\frac{-2D_A(1-\lambda) + 2D_B\lambda}{(M_A(1-\lambda) + M_B\lambda)^3} + \frac{-3(M_B - M_A)(D_A(1-\lambda)^2 + D_B\lambda^2)}{(M_A(1-\lambda) + M_B\lambda)^4} \right]. \end{aligned}$$

Приведя полученное выражение к общему знаменателю, найдем

$$\begin{aligned} (v^2)'' &= \frac{(2D_A + 2D_B)(M_A(1-\lambda) + M_B\lambda)^2}{(M_A(1-\lambda) + M_B\lambda)^4} - \\ &- \frac{4(D_B\lambda - D_A(1-\lambda))(M_B - M_A)(M_A(1-\lambda) + M_B\lambda)}{(M_A(1-\lambda) + M_B\lambda)^4} - \\ &- 2(M_B - M_A) \left[\frac{-2D_A(1-\lambda) + 2D_B\lambda}{(M_A(1-\lambda) + M_B\lambda)^4} \right] + \\ &+ 2(M_B - M_A) \left[\frac{3(M_B - M_A)(D_A(1-\lambda)^2 + D_B\lambda^2)}{(M_A(1-\lambda) + M_B\lambda)^4} \right] = \\ &= \frac{2D_A M_B [M_B(3 - 2\lambda) - 2 \cdot M_A(1 - \lambda)] + 2D_B M_A (M_A + 2M_A\lambda - 2M_B\lambda)}{(M_A(1-\lambda) + M_B\lambda)^4}. \end{aligned}$$

Заметим, что знаменатель полученного выражения для $(v^2)''$ больше нуля ($M_A > 0$ и $M_B > 0$ по условию). А значит необходимо доказать, что в точке $\lambda_{\kappa.в.}$ положительным является также числитель, т.е.

$$2D_A M_B (M_B (3 - 2\lambda_{\kappa.в.}) - 2 \cdot M_A (1 - \lambda_{\kappa.в.})) + 2D_B M_A (M_A + 2M_A \lambda_{\kappa.в.} - 2M_B \lambda_{\kappa.в.}) > 0.$$

Подставим в это неравенство значение $\lambda_{\kappa.в.}$, полученное из выражения (2.8):

$$2D_A M_B (M_B (3 - 2 \frac{M_B D_A}{M_A D_B + M_B D_A}) - 2 \cdot M_A (1 - \frac{M_B D_A}{M_A D_B + M_B D_A})) + \\ + 2D_B M_A (M_A + 2M_A \frac{M_B D_A}{M_A D_B + M_B D_A} - 2M_B \frac{M_B D_A}{M_A D_B + M_B D_A}) > 0,$$

после упрощения имеем истинное неравенство $2(D_A M_B^2 + D_B M_A^2) > 0$.

Действительно, если $0 < \lambda < 1$, то значения M_A и M_B положительные. Случай, когда дисперсии D_A и D_B одновременно равны нулю, как и при минимизации абсолютной погрешности расчетов статистической оценки для абсолютной частоты встречаемости подграфа, рассматривается отдельно.

2.4 Статистические эксперименты по r остовным деревьям

Для расчетов методом случайного выбора остовных деревьев частот встречаемости подграфов на пяти вершинах могут быть получены три различные статистические оценки (Рисунок 2.14), на шести вершинах – шесть, на семи – одиннадцать и т.д.

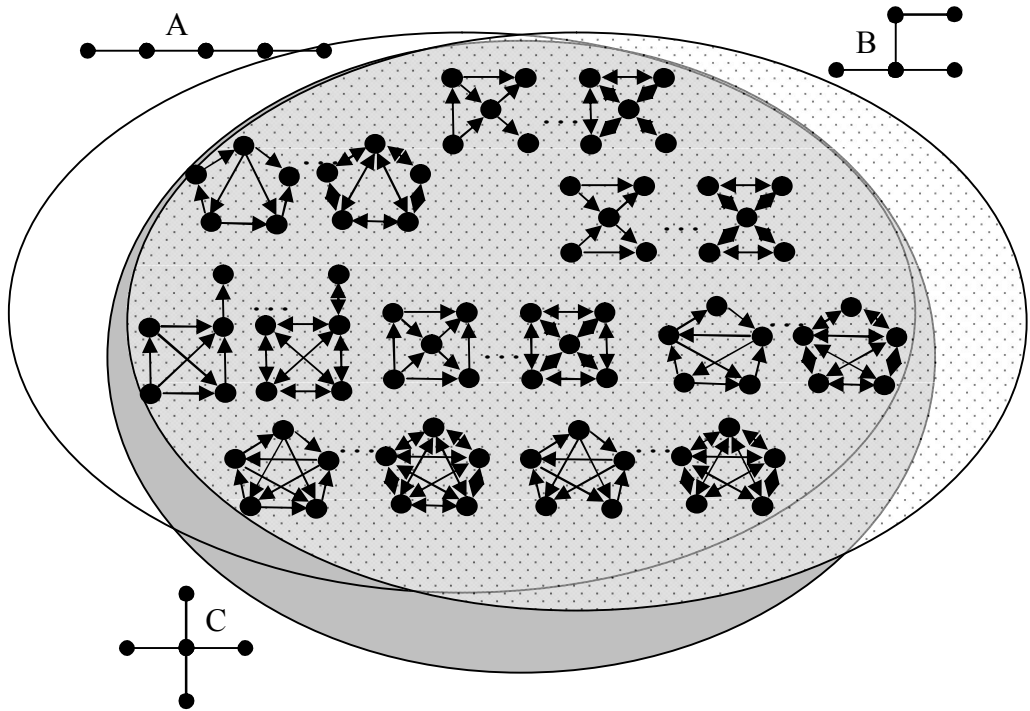


Рисунок 2.14 – Ориентированные подграфы на пяти вершинах, для которых могут быть получены три различные статистические оценки

Рассмотрим общий случай, когда в результате статистических экспериментов по всем возможным классам изоморфизма остовных деревьев были получены статистические оценки для частоты встречаемости подграфа при использовании r различных остовных деревьев.

Итоговую статистическую оценку абсолютных частот встречаемости в этом случае представим в виде линейной комбинации полученных оценок в результате статистических экспериментов:

$$\hat{n} = \lambda_1 \hat{n}^{[1]} + \lambda_2 \hat{n}^{[2]} + \dots + \lambda_r \hat{n}^{[r]}, \text{ где } \lambda_1 + \lambda_2 + \dots + \lambda_r = 1 \quad (2.9)$$

Далее найдем такие параметры $\lambda_1, \lambda_2, \dots, \lambda_r$, которые минимизируют дисперсию $D(\hat{n})$ итоговой оценки $\hat{n}$, т.е. обеспечивают получение эффективной оценки.

Заметим, что математическое ожидание $M[\hat{n}]$ будет

$$\begin{aligned} M[\hat{n}] &= M[\lambda_1 \hat{n}^{[1]} + \lambda_2 \hat{n}^{[2]} + \dots + \lambda_r \hat{n}^{[r]}] = \lambda_1 M[\hat{n}^{[1]}] + \lambda_2 M[\hat{n}^{[2]}] + \dots + \lambda_r M[\hat{n}^{[r]}] = \\ &= \lambda_1 M_1 + \lambda_2 M_2 + \dots + \lambda_r M_r. \end{aligned}$$

Найдем $D[\hat{n}] = D[\lambda_1 \hat{n}^{[1]} + \lambda_2 \hat{n}^{[2]} + \dots + \lambda_r \hat{n}^{[r]}] = \lambda_1^2 D_1 + \lambda_2^2 D_2 + \dots + \lambda_r^2 D_r \rightarrow \min$ при условии $\lambda_1 + \lambda_2 + \dots + \lambda_r = 1$ методом множителей Лагранжа.

Составим функцию Лагранжа

$$L = \lambda_1^2 D_1 + \lambda_2^2 D_2 + \dots + \lambda_r^2 D_r + \lambda(\lambda_1 + \lambda_2 + \dots + \lambda_r - 1). \quad (2.10)$$

Приравнивая частные производные функции Лагранжа (2.10) к нулю и присоединяя к полученным равенствам уравнение связи: $\lambda_1 + \lambda_2 + \dots + \lambda_r = 1$, получим следующую систему уравнений:

$$\begin{cases} L_1 = 2D_1 \cdot \lambda_1 + \lambda = 0, \\ L_2 = 2D_2 \cdot \lambda_2 + \lambda = 0, \\ \dots \\ L_r = 2D_r \cdot \lambda_r + \lambda = 0, \\ \lambda_1 + \lambda_2 + \dots + \lambda_r = 1. \end{cases}$$

Откуда получим:

$$\lambda_1 = \frac{D_1^{-1}}{D_1^{-1} + \dots + D_r^{-1}}, \dots, \lambda_r = \frac{D_r^{-1}}{D_r^{-1} + \dots + D_r^{-1}}. \quad (2.11)$$

Полученные значения $\lambda_1, \lambda_2, \dots, \lambda_r$ соответствуют минимуму $D[\hat{n}]$, поскольку дифференциал второго порядка функции L везде больше нуля, за исключением точек, когда $d\lambda_1, d\lambda_2, \dots, d\lambda_r$ одновременно равны нулю:

$$\begin{aligned} d^2 L &= L''_{\lambda_1 \lambda_1} (d\lambda_1)^2 + L''_{\lambda_2 \lambda_2} (d\lambda_2)^2 + \dots + L''_{\lambda_r \lambda_r} (d\lambda_r)^2 + \\ &\quad + 2L''_{\lambda_1 \lambda_2} d\lambda_1 d\lambda_2 + 2L''_{\lambda_1 \lambda_3} d\lambda_1 d\lambda_3 + \dots + 2L''_{\lambda_{r-1} \lambda_r} d\lambda_{r-1} d\lambda_r = \\ &= 2D_1 (d\lambda_1)^2 + 2D_2 (d\lambda_2)^2 + \dots + 2D_r (d\lambda_r)^2 > 0. \end{aligned}$$

2.5 Сравнение статистических методов по показателям качества

При анализе статистических методов расчета статистических оценок для частот встречаемости подграфов выявлено, что метод Верника-Раше в общем случае не позволяет получить несмещенные оценки частот встречаемости подграфов. В рамках диссертационной работы выявлен единственный режим работы статистического метода Верника-Раше, позволяющий получить несмещенную оценку относительных частот встречаемости подграфов, причем для его реализации необходимо построение всех ярусов дерева Верника-Раше, за исключением последнего его яруса, который достраивается частично. Назовем этот режим работы методом Верника-Раше (*).

Метод случайного выбора остовных деревьев дает несмещенные оценки абсолютных и относительных частот встречаемости подграфов при использовании остовных деревьев различных классов изоморфизма, подробное доказательство можно найти в работе [123]. Итоговая оценка представляется как линейная комбинация полученных оценок и также является несмещенной. При выборе соответствующих связующих коэффициентов по формулам (2.11) получаемые статистические оценки являются еще и эффективными.

Относительные частоты встречаемости подграфов в методе случайного выбора остовных деревьев также являются несмещенными, поскольку частоты

встречаемости подграфов, а также их сумма, определяются вследствие многократных испытаний Бернулли.

В Таблице 2.7 представлены основные характеристики статистических методов для расчета частот встречаемости подграфов.

Как можно видеть из Таблицы 2.7, предложенный метод случайного выбора остовных деревьев позволяет в отличие от других методов, рассмотренных в диссертации для анализа смешанных графов получать состоятельные, несмещенные оценки частот встречаемости подграфов.

Таблица 2.7 – Характеристики статистических методов для расчета статистических оценок для частот встречаемости подграфов

Название метода	Оценка относительных частот встречаемости подграфов		Оценка абсолютных частот встречаемости подграфов	
	Состоятельность	Несмещенность	Состоятельность	Несмещенность
Метод СВР	нет	нет	–	–
Статистический метод Верника-Раше	–	нет	–	–
Статистический метод Верника-Раше (*)	–	да	–	–
Метод случайного выбора остовных деревьев	да	да	да	да

2.6 Выводы по второй главе

1. Во второй главе рассмотрены статистические методы расчета частот (относительных и абсолютных) встречаемости подграфов. В частности исследуются состоятельность, несмещенность и эффективность получаемых этими методами статистических оценок. Особое внимание уделяется методам, позволяющим рассчитывать частоты встречаемости подграфов в смешанных графах. Среди таких методов наиболее популярным является статистический метод Верника-Раше. Выявляется единственный режим работы статистического метода Верника-Раше, при котором будут получены несмещенные статистические оценки для частот встречаемости подграфов.

2. Предложен метод случайного выбора остовных деревьев. Этот метод является развитием методов «случайного выбора пути» [128], разработанных для получения статистических оценок для частот встречаемости связных подграфов,

относящихся к различным классам изоморфизма, на трех [113], четырех [80] и пяти [123] вершинах в неориентированных графах. Метод случайного выбора остовных деревьев может использоваться при анализе и неориентированных, и ориентированных, и смешанных графов.

Основа предложенного метода заключается в равновероятном выборе реализаций остовного дерева на заданном числе вершин. Зная относительную частоту встречаемости подграфов при равновероятном выборе остовного дерева и общее число реализаций остовного дерева можно получить статистическую оценку для частот встречаемости подграфов.

Если на заданном числе вершин существуют остовные деревья, относящиеся к различным классам изоморфизма, то проводятся статистические испытания по каждому из них. При этом могут быть получены несколько статистических оценок для частот встречаемости некоторых подграфов. В этом случае итоговую оценку частоты встречаемости подграфа предлагается рассматривать как линейную комбинацию полученных оценок. При этом используемые линейные коэффициенты выбираются из условия минимизации дисперсии или минимизации коэффициента вариации получаемой итоговой оценки, тем самым достигается минимум абсолютной или относительной погрешности расчета, соответственно.

ГЛАВА 3 РАЗРАБОТКА АЛГОРИТМОВ И ИХ АНАЛИЗ

В главе предлагается ряд алгоритмов, необходимых для ускоренного выявления сетевых мотивов на четырех вершинах в больших сетях. В частности, алгоритм рандомизации графа, алгоритм метода случайного выбора остовных деревьев. Также предлагаются алгоритмы ускоренного выбора остовных деревьев типа «4-ветвление» и «4-путь», необходимые для получения статистических оценок для частот встречаемости подграфов. При анализе предложенных алгоритмов выполняется расчет требуемой дополнительной оперативной памяти, оценивается скорость работы алгоритмов. В качестве базовой структуры данных для хранения графов используется структура хранения графа, соответствующая классу `SparseGraph`, взятому из используемой в процессе разработки библиотеки JUNG: <http://jung.sourceforge.net/doc/api/edu/uci/ics/jung/graph/SparseGraph.html>.

3.1 Структура хранения графа

Смешанный граф удобно описывать тройкой $G = (V, U, A)$, где V – это множество вершин, U – множество ненаправленных ребер, A – множество направленных ребер. Множество всех ребер $E = U \cup A$. В качестве структуры хранения графов будем использовать (Рисунок 3.1) отображение *adjList* вершин $v_i \in V$ на тройку списков вершин. Эти три списка отражают связь вершины v_i с другими вершинами посредством:

- исходящих из вершины v_i направленных ребер (*outL*),
- входящих в вершину v_i направленных ребер (*inL*),
- инцидентных вершине v_i ненаправленных ребер (*udL*).

Помимо списка смежных вершин для обеспечения эффективной работы с объектами, представляющими собой ребра $e_j \in E$, используются два отображения. Одно отображение *dirEdges* (Рисунок 3.1) используется для определения соответствия направленного ребра и пары вершин, первая из которой является начальной, а вторая – конечной вершиной направленного ребра. Другое отображение

undirEdges (Рисунок 3.1) используется для получения соответствия ненаправленного ребра и инцидентных ему вершин. Для хранения *AdjList* требуется $O(|V|+2|U|+2|A|)$ элементов, а для хранения дополнительных отображений *dirEdges* и *undirEdges* еще $O(|E|)$ элементов. При этом все основные операции, необходимые для реализации метода случайного выбора остовных деревьев, имеют сложность $O(1)$. К основным операциям следует отнести:

- добавление вершины и добавление ребра;
- получение списка вершин, смежных заданной вершине;
- получение инцидентной ребру пары вершин;
- получение списка вершин;
- получение списка ребер.

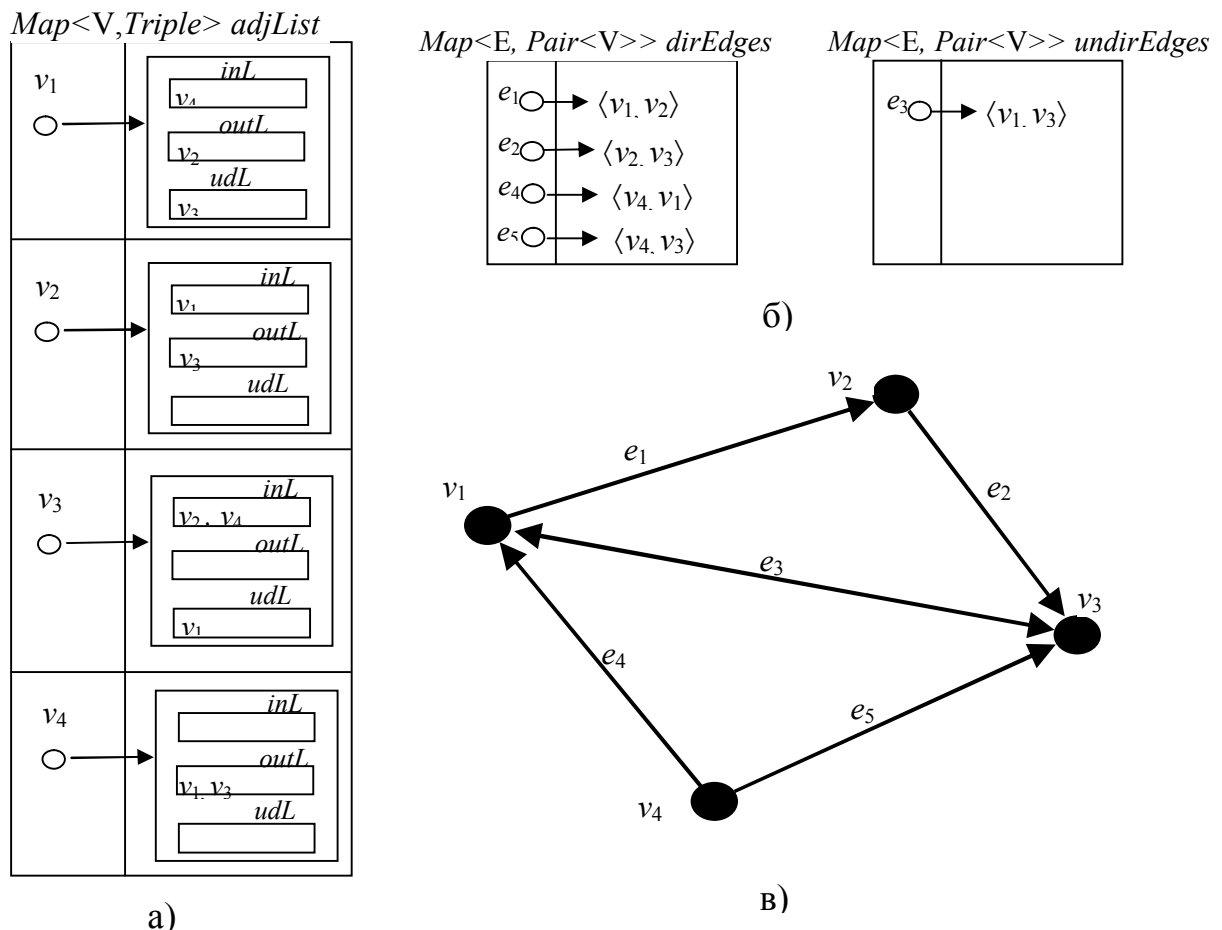


Рисунок 3.1 – Структура хранения графа в используемых алгоритмах: а) список смежных вершин *adjList*, б) отображения для сопоставления направленного (*dirEdges*) или (*undirEdges*) ненаправленного ребра на соответствующую пару вершин, в) иллюстрация смешанного графа

3.2 Случайный выбор остовных деревьев типа «4-ветвление»

Рассмотрим ускоренный алгоритм случайного выбора остовного дерева типа « k -ветвление» (остовное дерево на k вершинах, одна из которых – центральная, – имеет степень связности $k-1$, а остальные – степень связности один). В качестве базового возьмем алгоритм, предложенный в работе [113] для равновероятного выбора «3-ветвления». Этот алгоритм состоит из следующих двух шагов.

1) Выполняется случайный выбор вершины i с вероятностью p_i , пропорциональной числу реализаций в графе «3-ветвлений», центром которых является вершина i : $p_i = C_2^{k_i} N_V^{-1}$, где k_i – степень связности вершины i , N_V – число «3-ветвлений» в графе.

2) Далее равновероятно выбирается конкретная реализация «3-ветвления». Для этого выбираются два ребра смежных вершине i . Вероятность выбора «3-ветвления» на втором шаге составляет $p' = (C_2^{k_i})^{-1}$.

Вероятность выбора этого (и любого другого) «3-ветвления» за два таких шага составляет $p_i p' = 1 / N_V$. Таким образом, «3-ветвление» выбирается равновероятно.

Заметим, что аналогичные рассуждения можно применить при выборе «4-ветвлений» и «5-ветвлений», только биномиальный коэффициент $C_2^{k_i}$ заменяется биномиальным коэффициентом $C_3^{k_i}$ и $C_4^{k_i}$ соответственно. В методах расчета статистических оценок для частот встречаемости в неориентированных графах на четырех и пяти вершинах [113, 80] выбор «4-ветвлений» и «5-ветвлений» не реализовывался, поскольку в этом не было необходимости. В то же время в смешанных графах для расчета частот встречаемости подграфов на четырех и пяти вершинах проведение статистических экспериментов путем выбора остовных деревьев типа «4-ветвление» и «5-ветвление» необходимо.

3.2.1 Базовый алгоритм

Базовым алгоритмом случайного выбора «ветвления» назовем алгоритм, предложенный в работе [113]. Алгоритм представлен на Рисунке 3.2. При выборе случайной вершины пропорционально числу «ветвлений», центром которой она

является, используется классическая схема генерации случайной дискретной величины (шаг 4.1.3). Особенностью алгоритма является то, что вероятности p_v выбора вершины v рассчитываются предварительно и для хранения вероятностей требуется дополнительно $O(|V|)$ оперативной памяти.

НАЧАЛО

Ввод: граф $G(V,E)$, массив поправочных коэффициентов $coef[i]$, $i \in X$, где X – множество классов изоморфизма

Вывод: оценки абсолютных частот встречаемости подграфов $n[i]$, $i \in X$

1. Для всех $i \in X$ делай $n[i] \leftarrow 0$; $C[i] \leftarrow 0$; $N_V \leftarrow 0$

2. Рассчитать количество N_V реализаций « k -ветвления» в графе G :

Для $j = 1, \dots, |V|$ выполнить

$$N_V = N_V + C_k^{\deg(v_j)}$$

3. Рассчитать вероятности $p[j]$ для всех вершин $v_j \in V$

Для $j = 1, \dots, |V|$ выполнить

$$p_j \leftarrow C_k^{\deg(v_j)} N_V^{-1}$$

4. Для $j = 1, \dots, N^+$ выполнить

4.1. Выбрать множество вершин V' , определяющих случайное « k -ветвление»

4.1.1. $r \leftarrow \text{random}()$ // разыграть равномерную случайную величину $[0,1)$

4.1.2. $sum \leftarrow 0$ // переменная для расчета кумулятивной вероятности

4.1.3. Выбрать вершину v_s пропорционально числу «ветвлений»

Для $t = 1, \dots, |V|$ выполнить

$$sum = sum + p_t$$

Если $(sum \geq r)$

$$v_s \leftarrow v_t;$$

Выход из цикла по t

Конец Если

Конец цикла по t

4.1.4. Выбрать случайно $k-1$ вершин, смежных v_s : $\{v_{s1}, \dots, v_{sk-1}\}$

4.1.5. $V' = \{v_{s1}, \dots, v_{sk-1}\} \cup v_s$

4.2. $Num = \text{GetISO}[G[V']]$ // получить класс изоморфизма $Num \in X$

4.3. $C[Num] = C[Num] + 1$

5. Для всех $i \in X$ делай $n[i] \leftarrow N_V C[i] / N^+ / coef[i]$

КОНЕЦ

Рисунок 3.2 – Алгоритм равновероятного выбора « k -ветвлений»

Хранение вероятностей p_v , $v \in V$ позволяет существенно ускорить расчет по сравнению с подходом, когда вероятность выбора центральной вершины «ветвления» рассчитывалась бы непосредственно на шаге 4.1.3.

На Рисунке 3.3 представлено время (в секундах) расчета статистических оценок для частот встречаемости подграфов с использованием дополнительной памяти для хранения вероятностей p_v , и без ее использования. В качестве модели сети использован случайный граф Барабаши-Альберт (граф БА).

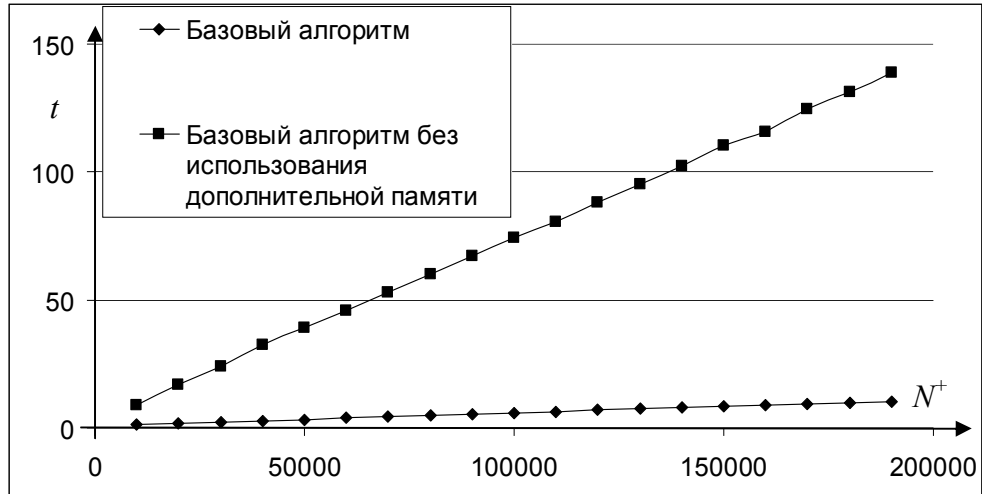


Рисунок 3.3 – Время t расчета статистических оценок для частот встречаемости подграфов путем случайного выбора «4-ветвлений» базовым алгоритмом. В качестве модели сети используется ориентированный граф БА, содержащий 500 000 вершин и 1 миллион направленных ребер

В репозитории <https://github.com/MNYudina/MFS> код проведенных экспериментов размещен в пакете `io.github.mnyudina.motifs.algorithms.branchspeedup`.

3.2.2 Алгоритм на основе метода расслоения

Предложим ускоренный метод случайного выбора « k -ветвлений» путем стратификации (расслоения) вершин по степени связности. Поскольку все вершины с одинаковой степенью связности имеют одинаковую вероятность выбора в качестве центральной вершины «ветвления», то можно предложить случайный выбор « k -ветвления» в три шага.

1) На первом шаге случайно выбирается множество вершин с одинаковой степенью связности. Вероятность P_t выбора множества вершин A_t , имеющих степень связности t , пропорциональна числу « k -ветвлений», серединами которых являются вершины из множества A_t : $P_t = |A_t| p_t = C_k^t N_V^{-1}$.

2) На втором шаге случайно, равновероятно выбирается вершина v из множества A_t .

3) На третьем шаге из множества вершин, смежных вершине v , равновероятно выбирается $(k - 1)$ вершина. Таким образом, находятся k вершин и $(k - 1)$ ребро, в совокупности представляющие собой реализацию « k -ветвления».

Использование расслоения вершин по степени связности позволяет ускорить случайный выбор «ветвлений», в частности, «4-ветвлений». На Рисунке 3.4 представлено время работы алгоритма для расчета статистических оценок для частот встречаемости подграфов в графе БА. Исследуется время работы алгоритма на основе метода расслоения и базовый алгоритм с использованием дополнительной памяти.

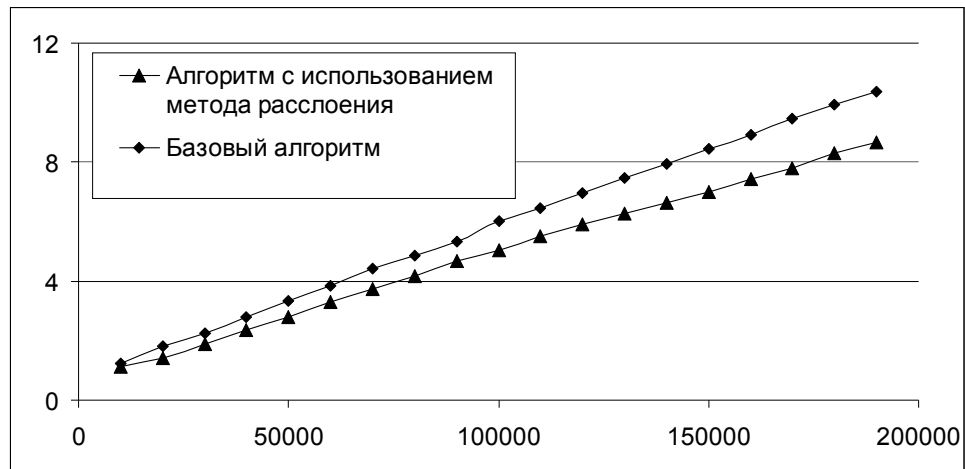


Рисунок 3.4 – Зависимость времени (в секундах) расчета частот встречаемости подграфов путем случайного выбора «4-ветвлений» от размера выборки; в качестве модели сети используется ориентированный граф БА, содержащий 500 000 вершин и 1 миллион направленных ребер

Также ускорение достигается при проведении расчета для исследуемых в диссертации графов сетей (Таблица 3.1).

Таблица 3.1 – Оценка времени расчета статистических оценок для частот встречаемости подграфов с использованием случайного выбора 100 000 «4-ветвлений»

Название графа	Время, в секундах		
	Базовый алгоритм	Базовый алгоритм без использования дополнительной памяти	Алгоритм на основе метода расслоения
Pathway Commons	17,176	4283,369	10,809
GenReg	9,439	2512,682	2,339
Google Plus	129,800	52449,466	58,004
Twitter	36,610	5281,677	6,140
PPI_HS	14,796	1522,176	4,205
AS	4,628	17,066	4,249
VKOngtu	1,981	118,108	0,796

Таким образом, в реализации метода случайного выбора остовных деревьев типа «ветвление», в силу быстродействия предпочтительнее оказывается алгоритм с использованием метода расслоения.

3.2.3 Дополнительные расходы оперативной памяти

Рассмотрим дополнительные расходы памяти при расчете частот встречаемости с использованием «4-ветвлений» алгоритмом на основе метода расслоения. Будем считать, что используется 32-разрядная виртуальная машина JAVA. Тогда при расчете встречаемости подграфов с использованием «4-ветвлений» требуется дополнительно память для хранения отображения m_v в виде хэш-таблицы. В качестве ключей хэш-таблицы используются степени связности k_v вершин, а в качестве значений – пары $pair_v$. Первым элементом каждой пары является список вершин $list_v$ со степенью связности k_v , а вторым элементом – вероятность выбора вершины из этого списка.

Пустое отображение m_v занимает память 136 байт. Каждый элемент отображения m_v представляет собой запись `java.util.HashMap.Entry`, содержащую ссылки на ключ и значение. Один элемент `Entry` занимает 24 байта. Все элементы m_v будут занимать $24 \cdot M_v$ байт, где M_v – количество элементов в m_v .

Для хранения одного ключа отображения k_v типа `Integer` необходимо 16 байт. Соответственно, для хранения всех ключей потребуется $16 \cdot M_v$ байт.

Для хранения одного значения отображения m_v – пары `Pair`, – требуется память 16 байт: для ссылки на список `verts` требуется 4 байта, для ссылки на вероятность `prob` еще 4 байта, на заголовок объекта 8 байт.

Для хранения списка `verts` (экземпляра класса `LinkedList`) в Java требуется 32 байта, а на хранение всех списков $32 \cdot M_v$ байт. Причем во всех списках содержатся ссылки на вершины графа, для хранения этих ссылок потребуется $4 \cdot |V|$ байт, где $|V|$ – число вершин графа.

Для хранения вероятностей p_v в значениях отображения потребуется $16 \cdot M_v$ байт (по 16 байт на одно значение типа `Double`).

Итого, дополнительно потребуется:

$$\begin{aligned}
 Memory_{Layers} &= 136 + 24 \cdot M_v + 16 \cdot M_v + 16 \cdot M_v + 32 \cdot M_v + 4 \cdot |V| + 16 \cdot M_v = \\
 &= 136 + 104 \cdot M_v + 4 \cdot |V| \text{ байт.}
 \end{aligned}
 \tag{3.1}$$

При использовании базового алгоритма потребуется хранить отображение вершин на их вероятности. На каждый элемент отображения Entry приходится 24 байта. При этом дополнительно необходимо хранить вероятность выбора вершины в типе Double (32 байта) и ссылку на вершину (4 байта):

$$Memory_{Standard} = 136 + (24 + 4 + 32) \cdot |V| = 136 + 60 \cdot |V| \text{ байт.} \tag{3.2}$$

В Таблице 3.2 представлены оценки необходимой дополнительной памяти при использовании предложенного алгоритма на основе метода расслоения и при использовании базового алгоритма.

Таблица 3.2 – Расчет дополнительной памяти (в байтах) при использовании алгоритма на основе метода расслоения и базового алгоритма

Название графа	$ V $	M_v	Алгоритм на основе метода расслоения	Базовый алгоритм
Pathway Commons	19 987	747	157 772	1 199 356
GenReg	16859	625	132 572	1 011 676
Google Plus	107 614	3705	815 912	6 456 976
Twitter	81 306	753	403 672	4 878 496
PPI HS	22 036	396	129 464	1 322 296
AS	22 963	161	108 732	1 377 916
VKOMgtu	9 927	103	50 556	595 756

Как можно видеть из Таблицы 3.2 для расчета самого большого графа исследуемых сетей методом расслоения требуется дополнительно 0,82 МБ, а базовым алгоритмом порядка 6,5 МБ.

Таким образом, алгоритм с использованием метода расслоения значительно менее требователен к использованию дополнительной памяти, чем базовый алгоритм. Для графа БА с параметром $m = 2$ число элементов (слоев) M_v в среднем можно вычислить по формуле (1.9). Соответственно коэффициент уменьшения объема используемой памяти с использованием формулы (3.1) и формулы (3.2) равен:

$$k = \frac{136 + 60N}{136 + 104M_v + 4N} = \frac{136 + 60N}{136 + 104\sqrt{3,05N} + 4N} = \frac{136 + 60N}{136 + 181,63\sqrt{N} + 4N}. \quad (3.3)$$

На Рисунке 3.5 представлена зависимость коэффициента уменьшения оперативной памяти, если используется алгоритм на основе метода расслоения, а не базовый алгоритм; использовалась формула (3.3).

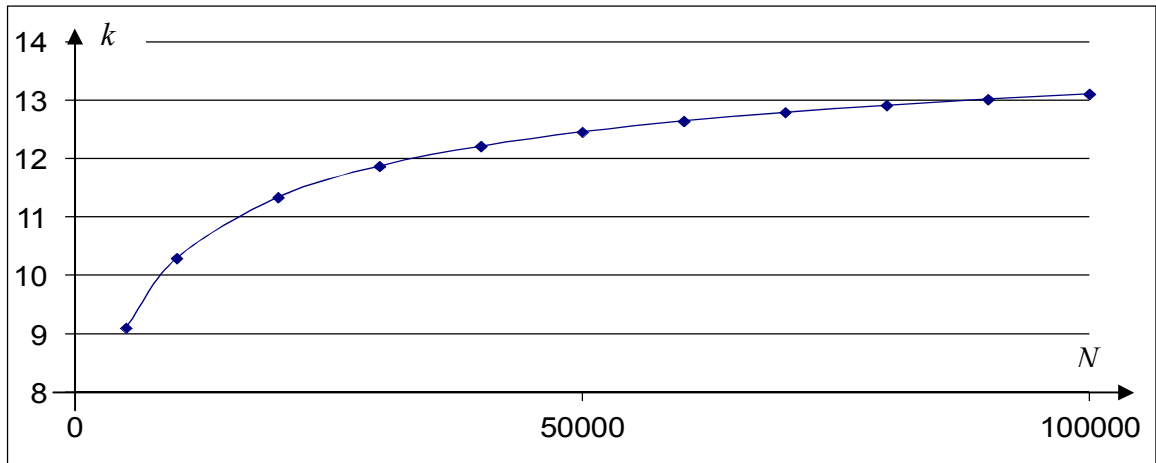


Рисунок 3.5 – Зависимость коэффициента уменьшения оперативной памяти при использовании алгоритма на основе метода расслоения

3.3 Случайный выбор остовных деревьев типа «4-путь»

Рассмотрим метод случайного выбора «4-пути» (остовного дерева на четырех вершинах, две вершины из которых имеют степень связности два и две – один).

3.3.1. Базовый алгоритм

Расчет статистических оценок для частот встречаемости подграфов путем проведения статистического эксперимента на основе случайного выбора остовного дерева типа «4-путь» может быть реализован с помощью алгоритма, представленного на Рисунке 3.6. В алгоритме сначала рассчитываются вероятности, что ребро является «центральным» в остовном дереве типа «4-путь» (шаги 1, 2, 3 на Рисунке 3.6). «Центральным» ребром остовного дерева «4-путь» будем называть такое ребро, инцидентные вершины которого имеют степень связности два. Затем проводится статистический эксперимент, в котором с соответствующей вероятностью выбирается ребро e_s и случайно два других смежных ребра так, чтобы ребра формировали остовное дерево типа «4-путь», центральным из которых является ребро e_s .

НАЧАЛО

Ввод: граф $G(V,E)$, массив поправочных коэффициентов $coef[i]$, $i \in X$, где X – множество классов изоморфизма, которые находятся с использованием «4-путей»

Вывод: статистические абсолютных оценки частот встречаемости подграфов $n[i]$, $i \in X$

1. Для всех $i \in X$ делай $n[i] \leftarrow 0$; $C[i] \leftarrow 0$; $N_{II} \leftarrow 0$;
2. Рассчитать общее число «3-путей» N_{II} графа $G(V,E)$:
 - 2.1. Для $j = 1, \dots, |E|$ выполнить
 - 2.2. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_j вершины
 - 2.3. $n_{\Delta} = |Nei(v_1) \cap Nei(v_2)|$
 - 2.4. $N_{II} \leftarrow (deg(v_1) - 1) \cdot (deg(v_2) - 1) - n_{\Delta}$
3. Рассчитать вероятности $p[j]$ для всех вершин $e_j \in G(V,E)$
 - 3.1. Для $j = 1, \dots, |E|$ выполнить
 - 3.1.2. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_j вершины
 - 3.1.3. $n_{\Delta} = |Nei(v_1) \cap Nei(v_2)|$
 - 3.1.4. $n_j \leftarrow (deg(v_1) - 1) \cdot (deg(v_2) - 1) - n_{\Delta}$
 - 3.1.5. $p_j = n_j N_{II}^{-1}$
4. Для $j = 1, \dots, N^+$ выполнить
 - 4.1. $r \leftarrow random()$; // разыграть равномерную случайную величину $[0,1)$
 - 4.2. $sum \leftarrow 0$; // переменная для расчета кумулятивной вероятности
 - 4.3. Выбрать ребро e_s пропорционально числу «3-путей»

Для $t = 1, \dots, |E|$ выполнить

$sum = sum + p_t$

Если $(sum \geq r)$

$e_s = e_t$; Выход из цикла по t

Конец цикла по t
 - 4.4. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_t вершины
 - 4.5. Пока $(v_3 \neq v_4)$ Делай

$v_3 \leftarrow$ случайно из $Nei(v_1)$

$v_4 \leftarrow$ случайно из $Nei(v_2)$

Конец цикла Пока
 - 4.6. $Num = GetISO[G[\{v_1, v_2, v_3, v_4\}]]$ // получить класс изоморфизма $Num \in X$
 - 4.7. $C[Num] = C[Num] + 1$;
5. Для всех $i \in X$ делай $n[i] \leftarrow N_{II} C[i] / N^+ / coef[i]$.

КОНЕЦ

Рисунок 3.6 – Базовый алгоритм случайного выбора остовных деревьев типа «4-путь»

3.3.2 Ускоренный алгоритм

Заметим, что расчет базовым алгоритмом требует расчета всех «треугольников» в графе (шаг 2.3 и шаг 3.1.3 на Рисунке 3.6), причем в сумме это делается шесть раз (поскольку один «треугольник» содержит три ребра). Эту ресурсоемкую операцию можно избежать с использованием ускоренного алгоритма (Рисунок 3.7).

Ускоренный алгоритм предложен в работе [80] для проведения расчета в неориентированных графах.

НАЧАЛО

Ввод: граф $G(V, E)$, массив поправочных коэффициентов $coef[i]$, $i \in X$, где X – множество классов изоморфизма, которые находятся с использованием «4-путей»

Вывод: статистические абсолютных оценки частот встречаемости подграфов $n[i]$, $i \in X$

1. Для всех $i \in X$ делай $n[i] \leftarrow 0$; $C[i] \leftarrow 0$; $N_{II} \leftarrow 0$; $success \leftarrow 0$

2. Рассчитать общее число «3-путей» N_{II} графа $G(V, E)$:

2.1. Для $j = 1, \dots, |E|$ выполнить

2.2. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_j вершины

2.3. $N_{II} \leftarrow (deg(v_1) - 1) \cdot (deg(v_2) - 1)$

3. Рассчитать вероятности $p[j]$ для всех ребер $e_j \in G(V, E)$

3.1. Для $j = 1, \dots, |E|$ выполнить

3.1.2. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_j вершины

3.1.3. $n_j \leftarrow (deg(v_1) - 1) \cdot (deg(v_2) - 1)$

3.1.4. $p_j = n_j N_{II}^{-1}$

4. Статистический расчет

4.1. Пока $success < N^+$ выполнить

4.1.1. $r \leftarrow \text{random}()$ // равномерно $[0, 1)$

4.1.2. $sum \leftarrow 0$; $e_s \leftarrow \text{null}$, $success = success + 1$,

4.1.3. Для $t = 1, \dots, |E|$ выполнить

$sum = sum + p_t$

Если $(sum \geq r)$

$e_s = e_t$; Выход из цикла по t

Конец Если

Конец цикла по t

4.1.4. $\{v_1, v_2\} \leftarrow$ получить инцидентные e_t вершины

4.1.5. $v_3 \leftarrow$ случайно из $Nei(v_1)$

4.1.6. $v_4 \leftarrow$ случайно из $Nei(v_2)$

4.1.7. Если $(v_3 \neq v_4)$ Делай

$Num = \text{GetISO}[G[\{v_1, v_2, v_3, v_4\}]]$

$C[Num] = C[Num] + 1$;

4.1.8 Иначе Если $(v_3 = v_4)$ $N^+ = N^+ + 1$

5. Для всех $i \in X$ делай $n[i] \leftarrow N_{II} \cdot C[i] / N^+ / coef[i]$

КОНЕЦ

Рисунок 3.7 – Ускоренный алгоритм выбора остовных деревьев типа «4-путь»

Основное отличие ускоренного алгоритма от базового состоит в том, что при выборе остовного дерева типа «4-путь» на шаге 4 алгоритма на Рисунке 3.7 случайное ребро (v_i, v_j) выбирается пропорционально $(deg(v_i) - 1)(deg(v_j) - 1)$.

Таким образом, вероятность выбора ребра (v_i, v_j) в ускоренном методе пропорциональна числу путей длины три ребра, в середине которых находится само ребро (v_i, v_j) , при этом не учитывается, выбран замкнутый путь (цикл) или остовное дерево. В таком алгоритме возможны «холостые ходы», поскольку могут быть выбраны как остовные деревья типа «4-путь», так и «треугольники». «Холостой ход» возникает, когда выбранный путь длиной три ребра является «треугольником». В этом случае «обнаруживается» ноль подграфов на четырех вершинах, поэтому счетчики подграфов не изменяются, но счетчик числа опытов наращивается на единицу.

В ускоренном алгоритме в отличие от базового подсчет числа треугольников не выполняется. Кроме того, изменяется шаг 4 так, чтобы размер выборки остовных деревьев обоих алгоритмов был одинаков. При этом реализуется небольшой выигрыш в скорости расчета (Рисунок 3.8).

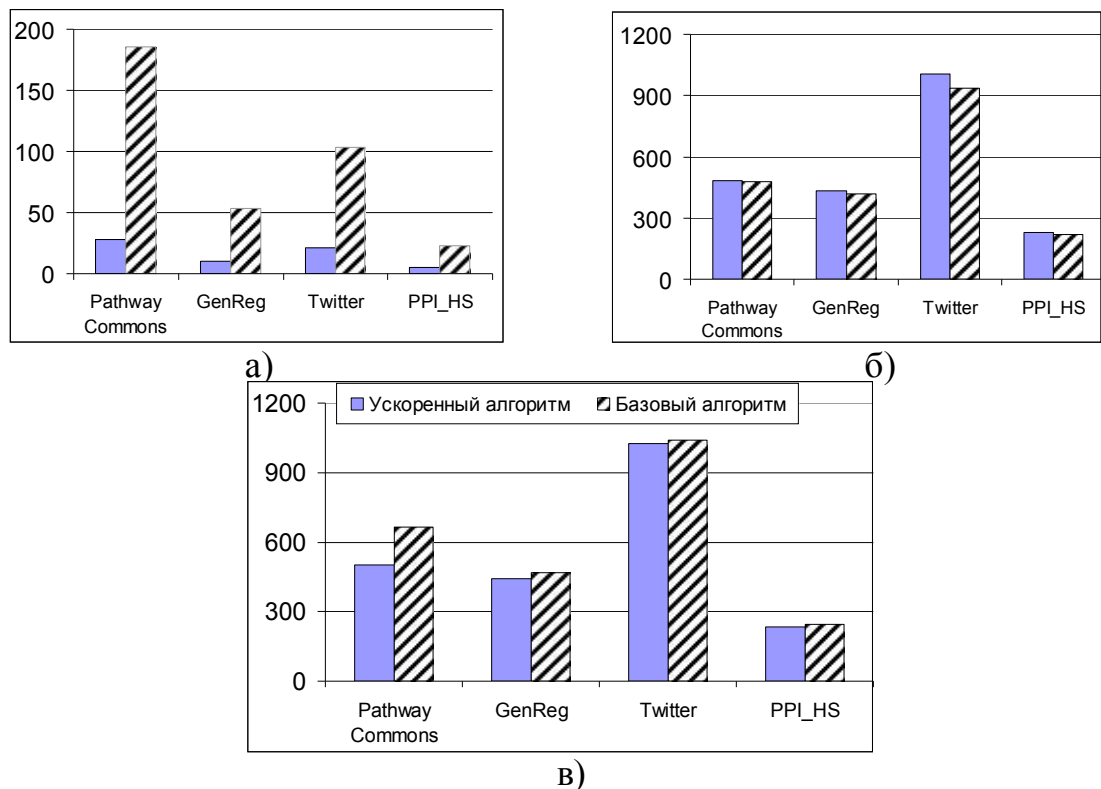


Рисунок 3.8 – Время расчета с использованием базового и ускоренного алгоритмов: а) предварительный расчет вероятностей (соответствует шагам 1-3 алгоритмов на Рисунке 3.6 и Рисунке 3.7); б) результаты статистического расчета; в) итоговое время расчета. Выборка выполнена по 100 000 случайным остовным деревьям типа «4-путь»

В репозитории <https://github.com/MNYudina/MFS> код проведенных экспериментов размещен в пакете `io.github.mnyudina.motifs.algorithms.pathspeedup`.

3.3.3 Использование метода расслоения

По аналогии с методом расслоения графа по степеням вершин можно построить алгоритм с использованием расслоения по ребрам, объединяя ребра (v_i, v_j) с одинаковым значением произведения $(\deg(v_i) - 1)(\deg(v_j) - 1)$. Тогда случайное остовное дерево типа «4-путь» равновероятно выбирается за три шага.

1) На первом шаге случайно выбирается слой B_t , содержащий ребра (v_i, v_j) с одинаковым значением $(\deg(v_i) - 1)(\deg(v_j) - 1)$.

2) На втором шаге случайно, равновероятно с вероятностью $1/(|B_t| - 1)$, из B_t выбирается ребро (v_k, v_r) .

3) На третьем шаге из множества вершин, смежных вершине v_k , а также смежных вершине v_r , равновероятно выбираются смежные вершины и тем самым случайный путь длины три ребра. Этот путь может быть «треугольником» и тогда это «холостой ход», иначе это остовное дерево типа «4-путь», на котором в этом случае фиксируется обнаружение некоторого подграфа на четырех вершинах.

Как и в случае использования метода расслоения при случайном выборе «4-ветвления» в случае использования метода расслоения по ребрам удастся получить выигрыш в производительности (Таблица 3.3)

Таблица 3.3 – Время расчета статистических оценок для частот встречаемости подграфов по 100 000 равновероятных выборов «4-путей»

Название графа	Время расчета, в секундах	
	Алгоритм Джа с соавторами	Алгоритм на основе метода расслоения
Pathway Commons	528,361	104,530
GenReg	482,391	58,370
Twitter	1028,138	80,676
PPI_HS	252,952	25,123
AS	10,719	5,631
VKOmgtu	14,627	1,857

3.3.4 Дополнительные расходы оперативной памяти

Использование при случайном выборе «4-путей» метода расслоения требует дополнительно память для хранения отображения m_e , в качестве ключей которого используются целые числа удвоенной точности k_e , а в качестве значений – пары. Первым элементом каждой пары является множество ребер $edges$. Вторым

элементом пары является вероятность $prob$ выбора ребра из соответствующего множества $edges$ в качестве центрального ребра «4-пути». Вычисляя количество памяти, которое необходимо для хранения отображения m_e , получим:

$$\begin{aligned} Memory_{Layer} &= 136 + 24 \cdot M_e + 16 \cdot M_e + 16 \cdot M_e + 32 \cdot M_e + 4 \cdot |E| + 16 \cdot M_e = \\ &= 136 + 104 \cdot M_e + 4 \cdot |E| \text{ байт,} \end{aligned} \quad (3.4)$$

где M_e – количество элементов, содержащихся в отображении m_e .

В тоже время при использовании базового метода потребуется также хранить отображение ребер на их вероятности. На каждый элемент отображения Entry приходится 24 байта. При этом дополнительно необходимо хранить вероятность выбора ребра в типе Double (32 байта) и ссылку на ребро (4 байта).

$$Memory_{Standard} = 136 + (24 + 4 + 32) \cdot |E| = 136 + 60 \cdot |E| \text{ байт.} \quad (3.5)$$

В Таблице 3.4 представлены оценки необходимой дополнительной памяти, полученные с использованием формулы (3.4) и формулы (3.5).

Таблица 3.4 – Требуемая дополнительная память при использовании метода случайного остоного дерева типа «4-путь»

Название графа	$ E $	M_e	Алгоритм на основе метода расслоения	Алгоритм Джа с соавторами
Pathway Commons	824 675	110 450	14 785 636	49 480 636
GenReg	713 649	53 579	8 426 948	42 819 076
Google Plus	13 673 453	2 653 104	330 616 764	820 407 316
Twitter	1 768 149	984 414	109 451 788	106 089 076
PPI_HS	309 245	32 826	4 651 020	18 554 836
AS	48 436	4 257	636 608	2 906 296
VKOMgtu	35 596	2 261	377 664	2 135 896

Таким образом, для расчета самого большого из исследуемых графов требуется дополнительно 331 МБ при использовании метода расслоения и 820 МБ при использовании алгоритма Джа с соавторами.

Таким образом, с учетом увеличения скорости и уменьшения требуемой оперативной памяти, как в случае разыгрывания «4-ветвлений», так и в случае разыгрывания «4-путей», предпочтение будем отдавать алгоритмам с использованием метода расслоения.

3.4 Определение класса изоморфизма графа

При расчете статистических оценок для частот встречаемости подграфов на четырех вершинах необходимо использовать отображение всех графов на соответствующие классы изоморфизма. В этом случае графы необходимо идентифицировать. Предложим кодировать графы битовой комбинацией, каждый бит которой означает наличие или отсутствие ребра. При этом неориентированные ребра будем представлять двумя противоположно направленными ребрами (Рисунок 3.9а). Тогда для описания графа на четырех вершинах необходимо 12 бит. Каждое направление между вершинами идентифицируется степенями двойки (Рисунок 3.9б). Каждому возможному графу, таким образом, можно сопоставить некоторое число Num в зависимости от наличия или отсутствия соответствующих ребер в графе. На Рисунке 3.9 в) и г) проиллюстрирован пример получения этого уникального числа посредством операции «побитовое «или»».

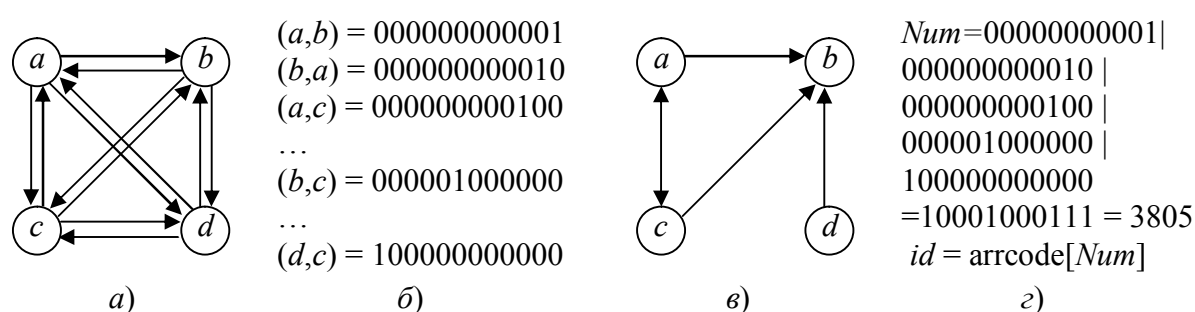


Рисунок 3.9 – а) возможные направленные ребра, б) двоичное представление ребер, в) пример исследуемого смешанного графа, г) получение индекса id графа в массиве идентификаторов классов изоморфизма $arrcode$

Размеры массива $arrcode$, используемого для идентификации подграфов, содержащих три, четыре и пять вершин приведены в Таблице 3.5.

Таблица 3.5 – Размеры массива $arrcode$, предназначенного для хранения идентификаторов классов изоморфизма

Число вершин	Размер массива $arrcode$
3	$2^6 = 64$
4	$2^{12} = 4096$
5	$2^{20} = 1\,048\,576$

Заметим, что при работе с подграфами на трех и четырех вершинах использование предпросчитанных идентификаторов классов изоморфизма представляется возможным. При работе с графами на пяти вершинах такой подход потребует чрезмерного использования дополнительной памяти для хранения массива `arcode`. Действительно, в этом случае необходимо дополнительно использовать гигабайты (Таблица 3.5) оперативной памяти, что, очевидно, нецелесообразно.

3.5 Алгоритмы программ

На Рисунке 3.10 представлена схема алгоритма программы, реализующей метод случайного выбора остовных деревьев для получения статистических оценок абсолютных частот встречаемости подграфов на четырех вершинах. Сначала рассчитываются частоты встречаемости подграфов при использовании случайного выбора «4-ветвления» и случайного выбора «4-пути» (блок 3 и блок 4).

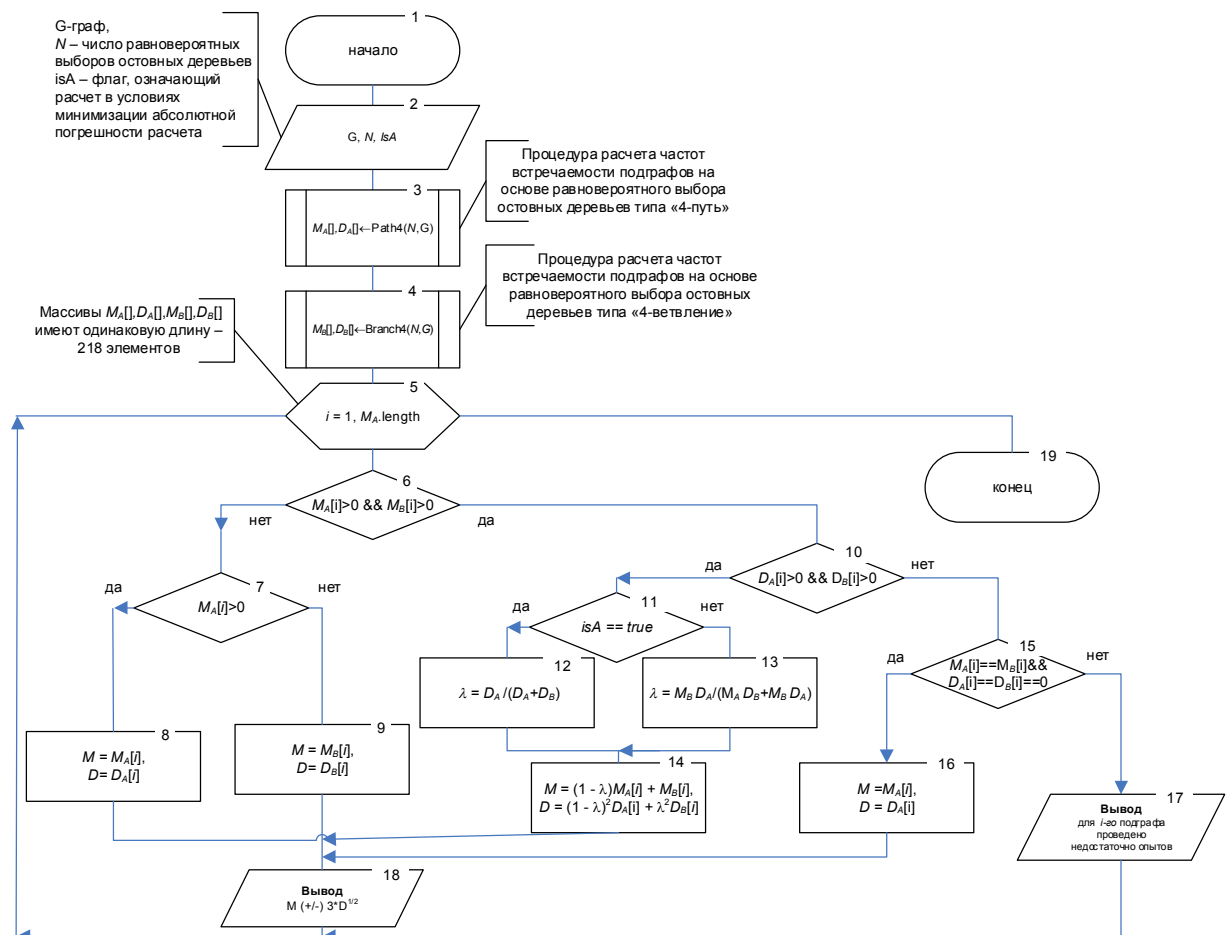


Рисунок 3.10 – Схема программы *MFS* получения статистических оценок для частот встречаемости подграфов

Расчет итоговых статистических оценок и доверительных интервалов, характеризующих погрешность вычислений, выполняется в блоках 5-17 схемы алгоритма на Рисунке 3.10. При получении итоговой оценки для каждого подграфа на четырех вершинах учитываются оценки частот встречаемости подграфов при использовании различных остовных деревьев. Если обе полученные оценки ненулевые, то итоговая оценка частоты встречаемости подграфов рассматривается как линейная комбинация двух оценок (блоки 10-14 на Рисунке 3.10). В случае минимизации абсолютной погрешности используется формула (2.5), в случае минимизации относительной погрешности – формула (2.8). Если только одна из двух статистических оценок ненулевая, то такая оценка принимается в качестве итоговой оценки. Если обе оценки нулевые – это свидетельствует о том, что данный подграф не встречается в графе. Блоки 15–17 выполняются тогда, когда в графе либо содержится только один из всех возможных подграфов, либо было проведено недостаточное количество статистических испытаний и результаты противоречивы.

Процедуры расчетов статистических оценок для частот встречаемости подграфов на основе равновероятного выбора «4-ветвлений» (алгоритм Branch4) и «4-пути» (алгоритм Path4) подробно описываются ниже.

3.5.1 Случайный выбор «4-ветвлений»

В алгоритме Branch4, реализующем случайный выбор «4-ветвлений» (Рисунок 3.11), рассчитываются статистические оценки для частот встречаемости подграфов на четырех вершинах. В качестве входных данных в алгоритме используется граф $G(V,E)$ и число N опытов (случайных выборов остовного дерева типа «4-ветвление» в статистическом эксперименте). Счетчики встречаемости подграфов представлены массивом $C[]$. Также используется заранее сформированный массив коэффициентов $coef$ (i -й элемент массива определяет, сколько реализаций остовного дерева типа «4-ветвление» можно найти в i -м подграфе).

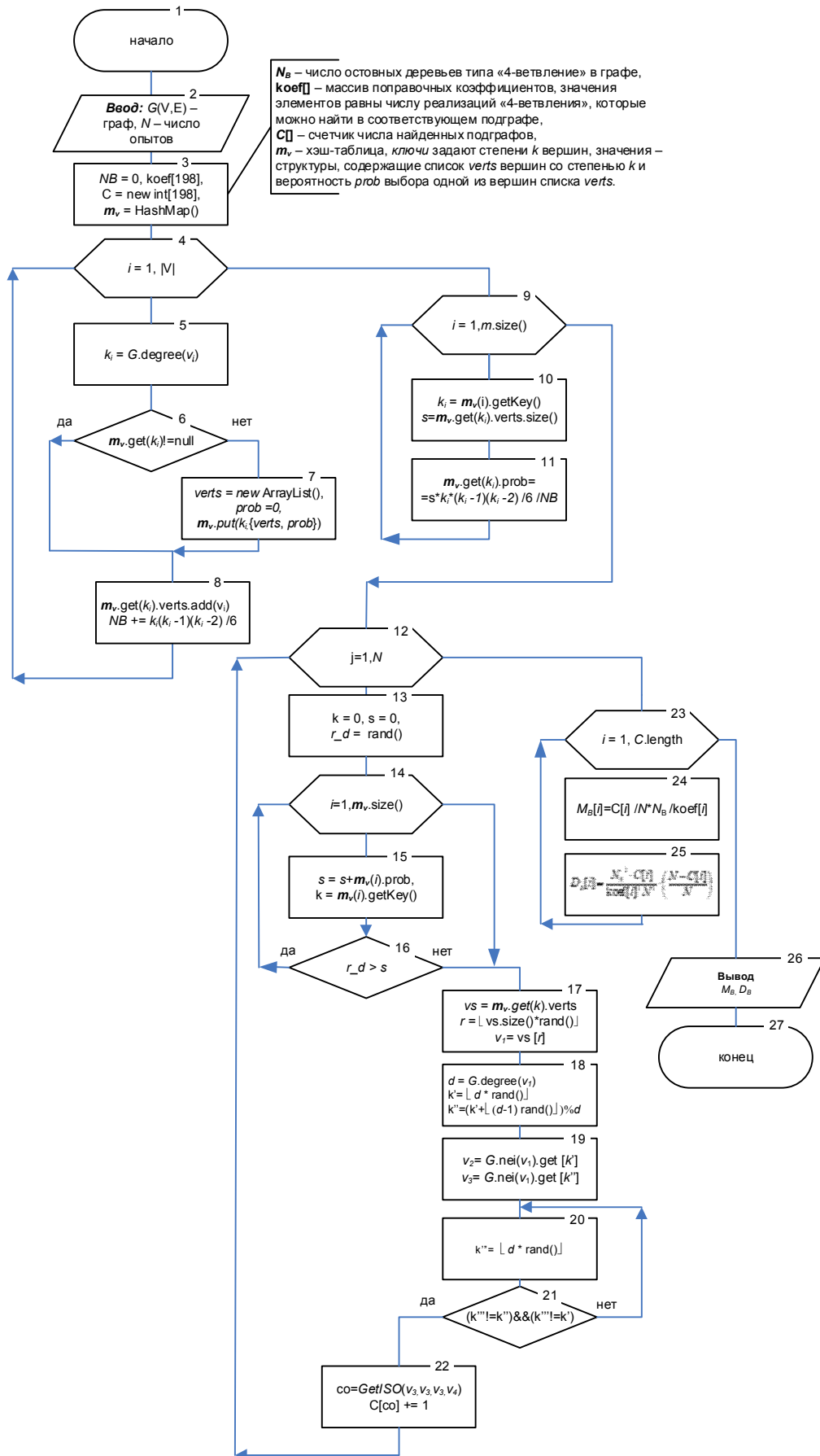


Рисунок 3.11 – Алгоритм процедуры Branch4 расчета статистических оценок для частот встречаемости подграфов методом случайного выбора «4-ветвлений»

В алгоритме используется вспомогательная структура данных для ускорения расчета – отображение (хэш-таблица) m_v , ключи отображения m задают степени связности вершин k , а значения – это пары $\langle \text{verts}, \text{prob} \rangle_k$. Первым элементом каждой пары является список verts вершин со степенью связности, соответствующей ключу отображения. Вторым элементом пары – действительное значение prob , которое иницируется значением, равным вероятности выбора в качестве центра «ветвления» любой из вершин, содержащейся во множестве verts .

В блоках 4-11 (Рисунок 3.11) происходит инициализация отображения m_v , в блоках 4-8 инициализируются списки вершин verts и устанавливаются соответствующие ключи отображения m_v , а в блоках 9-11 вычисляется вероятность prob .

В блоках 12-22 (Рисунок 3.11) выполняется статистический эксперимент. В соответствии с вероятностями prob выбирается список вершин verts , блоки 13-16. Далее из выбранного списка verts равновероятно выбирается вершина v_1 . Эта вершина v_1 выбирается с вероятностью, пропорциональной числу реализаций «4-ветвлений», центральной вершиной которых является вершина v_1 .

В блоках 18-20 (Рисунок 3.11) случайно выбираются три другие вершины v_2, v_3, v_4 из окрестности вершины v_1 , входящие в выбранную реализацию «4-ветвления», центральной вершиной которой является вершина v_1 .

В блоке 22 на Рисунке 3.11 путем вызова процедуры *GetISO* определяется класс изоморфизма найденного подграфа и увеличивается соответствующий счетчик найденных подграфов в массиве C .

После проведения статистического эксперимента (блоки 23-25 схемы алгоритма на Рисунке 3.11) выполняется расчет статистических оценок для частот встречаемости подграфов и оценивается погрешность расчетов.

Алгоритм определения класса изоморфизма *GetISO* представлен на Рисунке 3.12. Входными данными являются массив вершин подграфа, а также исследуемый граф (для проверки наличия ребер между переданными вершинами). Для получения класса изоморфизма используются предопределенные массивы *idx* и *arrcode*.

Размер массива *idx* при расчете частот встречаемости подграфов на четырех вершинах равен учетверенному значению числа вершин, т.е. шестнадцати. В массиве *idx* ребра рассматриваются как направленные, эти ребра кодируются степенью двойки. Массив *idx* можно разбить на соответствующие квартеты, задающие коды ребер, заходящих в вершину, номер которой соответствует порядковому номеру квартета, и исходящих из вершины, номер которой совпадает с порядковым номером внутри квартета. Так, первый квартет содержит коды ребер, связывающих первую вершину с вершиной с соответствующим порядковым номером внутри квартета. В алгоритме на Рисунке 3.12 перебираются (блоки 5-11) возможные ребра и путем применения побитовой операции «или» (блок 8) формируется значение переменной *code*, однозначно определяющее наличие соответствующих ребер в подграфе.

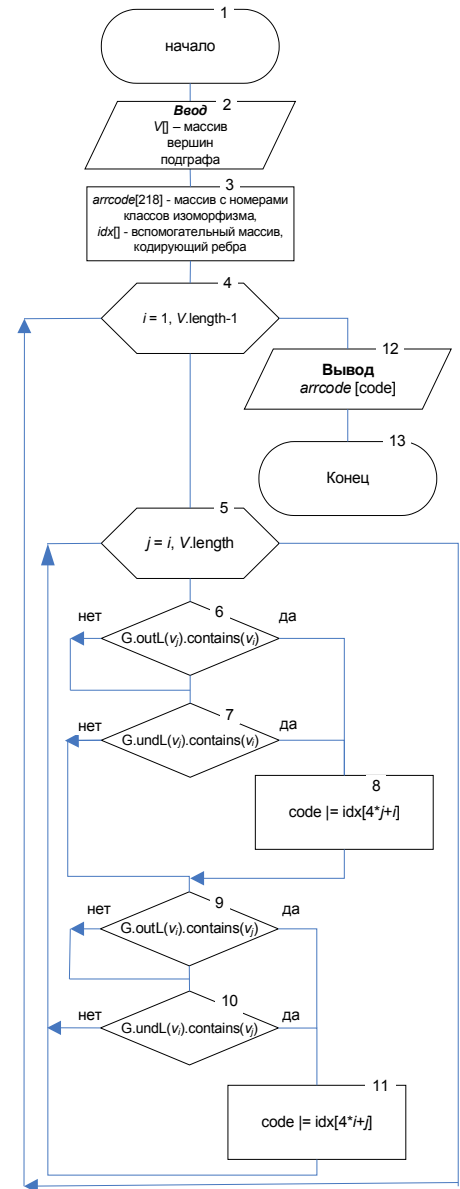


Рисунок 3.12 – Алгоритм процедуры *GetISO* для получения класса изоморфизма

3.5.2 Случайный выбор «4-путей»

В алгоритме Path4 выполняется случайный выбор «4-путей» и расчет статистических оценок для частот встречаемости подграфов на четырех вершинах (Рисунок 3.13).

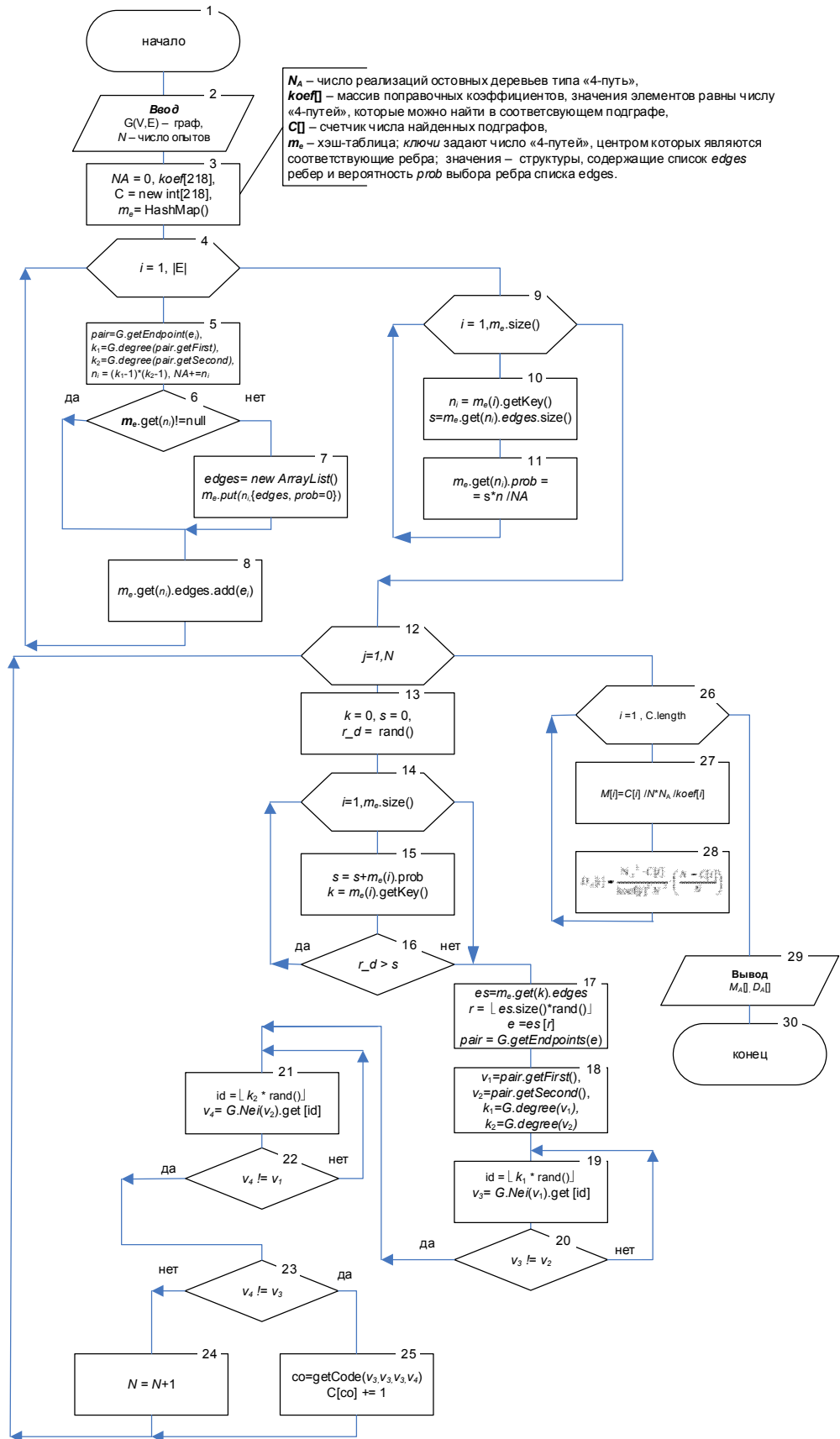


Рисунок 3.13– Алгоритм процедуры Path4 расчета частот встречаемости подграфов методом случайного выбора остовных деревьев типа «4-путь»

Входными данными является граф $G(V, E)$ и число опытов N в статистическом эксперименте. Счетчики встречаемости подграфов представлены массивом $C[]$, используется заранее сформированный массив коэффициентов $coef[]$, каждый i -й элемент которого определяет, сколько «4-путей» можно найти в i -м подграфе.

В алгоритме на Рисунке 3.13 используется вспомогательная хэш-таблица – отображение m_e . Ключи отображения идентифицируют ребра с одинаковой вероятностью выбора, значениями являются пары $\langle edges, prob \rangle$. Первый элемент пары – это список $edges$ ребер. Второй элемент пары – это действительное значение $prob$, определяющее вероятность выбора в статистическом эксперименте ребра графа, который входит в соответствующий список $edges$. В блоках 4-11 схемы алгоритма происходит инициализация отображения m_e , в блоках 4-8 инициализируются списки ребер $edges$ и устанавливаются соответствующие ключи отображения, в блоках 9-11 рассчитываются вероятности $prob$. В блоках 12-25 выполняется статистический эксперимент, в котором в соответствии с вероятностями $prob$ отображения m_e выбирается соответствующее множество $edges$ (блоки 13-16). Далее из этого множества равновероятно выбирается ребро, инцидентные вершины которого v_1 и v_2 являются двумя из четырех вершин «4-пути». Другие две вершины разыгрываются в блоках 19 и 21. В блоке 24 путем вызова процедуры *GetISO* определяется класс изоморфизма найденного подграфа и увеличивается его счетчик в массиве $C[]$. После проведения статистического эксперимента в блоках 26-28 (Рисунок 3.13) выполняется расчет математических ожиданий и дисперсий статистических оценок для частот встречаемости подграфов.

3.6 Скорость и точность расчетов

Рассмотрим точность и скорость расчета итоговых статистических оценок для частот встречаемости подграфов. При этом используются статистические эксперименты по двум различным остовным деревьям: «4-ветвление» и «4-путь».

3.6.1 Точность расчетов

Точность расчетов можно оценить путем расчета погрешности, а именно рассчитывая доверительные интервалы $\pm 3\sigma$ (для абсолютной погрешности) и $\pm 3v$

(для относительной погрешности). За пределы доверительного интервала получаемые оценки частот встречаемости подграфов не выходят за 99,7% случаев.

В Таблице 3.6 представлены результаты расчета в неориентированном графе. Уже при 100 000 опытов относительная погрешность расчета не превышает 10%.

Таблица 3.6 – Результаты расчета статистических оценок для частот встречаемости подграфов в неориентированном графе PPI_HS

Точные значения	Результаты расчета методом случайного выбора остовных деревьев, выполнено N случайных выборок по «4- ветвлениям» и N выборок по «4-путям»							
	Оценки для частот встречаемости подграфов				Погрешность расчета, $\pm 3\sigma$			
	$N = 1000$	$N=10000$	$N=100000$	$N = 1$ млн.	$N = 1000$	$N=10000$	$N=100000$	$N = 1$ млн.
1,18E+10	1,19E+10	1,18E+10	1,18E+10	1,18E+10	0,097538	0,030967	0,0098034	0,003098
4,89E+09	4,95E+09	4,95E+09	4,88E+09	4,89E+09	0,110808	0,03504	0,0111633	0,0035266
6,99E+08	6,77E+08	6,78E+08	7,03E+08	6,97E+08	0,188237	0,059467	0,0184688	0,0058633
47377822	40558738	47487523	48197301	47495972	0,612372	0,178965	0,0561747	0,0178928
37806037	30575565	36499582	38095242	38139207	0,53033	0,153493	0,0475109	0,0150142
4329477	5255175	3678623	4361796	4368005	0,904534	0,341881	0,0992843	0,0313711

В Таблице 2.11 представлены результаты анализа ориентированного графа GenReg. Граф GenReg содержит всего 24 различных подграфа.

Таблица 3.7 – Результаты расчета статистических оценок для частот встречаемости подграфов в ориентированном графе GenReg

Точные значения	Результаты расчета методом случайного выбора остовных деревьев, выполнено N случайных выборок по «4-ветвлениям» и N выборок по «4-путям»							
	Оценки для частот встречаемости подграфов				Погрешность расчета, $\pm 3\sigma$			
	$N = 1000$	$N=10000$	$N = 100000$	$N = 1$ млн.	$N = 1000$	$N = 10000$	$N=100000$	$N=1$ млн.
9,39E+08	1,16E+09	8,85E+08	9,55E+08	9,45E+08	0,345791	0,126184	0,038341	0,01219
2,39E+09	2,56E+09	2,31E+09	2,40E+09	2,40E+09	0,221505	0,074416	0,023008	0,007287
2,49E+09	2,38E+09	2,53E+09	2,49E+09	2,49E+09	0,260621	0,079591	0,025404	0,008026
6,89E+09	7,08E+09	7,00E+09	6,91E+09	6,90E+09	0,129854	0,04141	0,013223	0,004186
2,06E+08	1,58E+08	2,12E+08	1,99E+08	2,06E+08	0,595744	0,162252	0,052862	0,016442
2,1E+08	1,70E+08	1,81E+08	2,10E+08	2,10E+08	0,572998	0,175684	0,051432	0,016268
44827678	5,59E+07	4,42E+07	4,59E+07	4,44E+07	0,899545	0,320232	0,099326	0,031969
9791942	1,68E+07	1,03E+07	1,01E+07	1,00E+07	1,131541	0,456944	0,146213	0,046343
4,48E+09	4,53E+09	4,49E+09	4,44E+09	4,47E+09	0,154424	0,049192	0,015676	0,004931
1,27E+09	1,02E+09	1,23E+09	1,29E+09	1,28E+09	0,413521	0,118324	0,0365	0,011585
1,36E+08	1,45E+08	1,24E+08	1,42E+08	1,36E+08	0,621597	0,212666	0,06278	0,020312
5,23E+09	5,10E+09	5,27E+09	5,21E+09	5,22E+09	0,16388	0,050717	0,016163	0,005105
1,36E+08	1,20E+08	1,35E+08	1,38E+08	1,35E+08	0,684741	0,2034	0,063618	0,020337
3,73E+08	3,65E+08	3,81E+08	3,67E+08	3,73E+08	0,38791	0,120001	0,0387	0,012135
33252040	4,07E+07	3,15E+07	3,36E+07	3,22E+07	1,056409	0,379817	0,116212	0,037599
10597087	1,44E+07	1,01E+07	1,14E+07	1,05E+07	1,222087	0,462334	0,137756	0,045256
30568288	3,56E+07	2,59E+07	3,10E+07	3,03E+07	1,129918	0,419011	0,121095	0,038744
13799935	2,16E+07	1,35E+07	1,45E+07	1,39E+07	0,997049	0,400304	0,121845	0,03942
6,83E+09	6,57E+09	6,92E+09	6,85E+09	6,83E+09	0,116919	0,03537	0,011291	0,003577
6,47E+08	6,76E+08	6,59E+08	6,42E+08	6,49E+08	0,281043	0,090111	0,028893	0,009088
11858950	1,44E+07	1,08E+07	1,30E+07	1,21E+07	1,222087	0,446638	0,128645	0,042283
16439425	7211600	1,83E+07	1,54E+07	1,66E+07	1,730607	0,343367	0,118357	0,036024
23132607	2,64E+07	2,31E+07	2,31E+07	2,34E+07	0,902022	0,305378	0,096589	0,030323
5959638	1,32E+07	5408432	6466532	5881323,3	0,901525	0,446698	0,129137	0,042826

На Рисунке 3.14 показана зависимость величины относительной погрешности расчетов при получении статистических оценок для частот встречаемости подграфов от числа опытов в неориентированном графе PPI_HS.

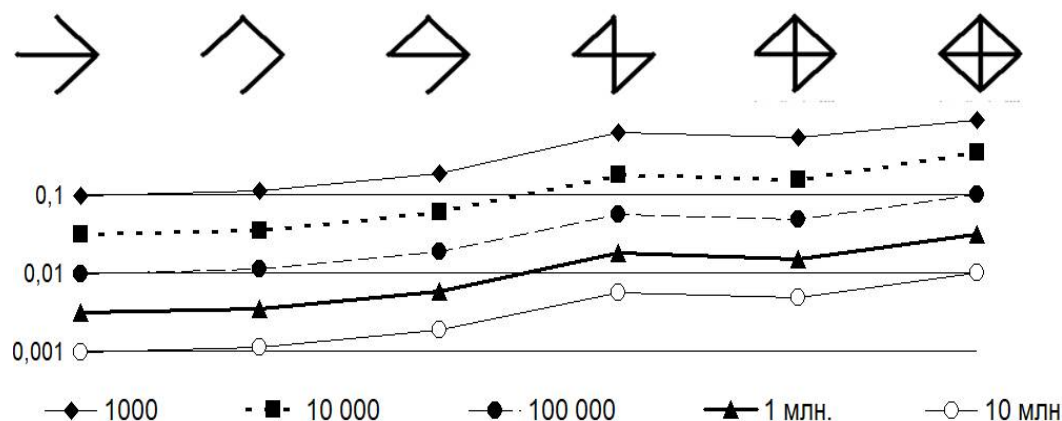


Рисунок 3.14 – Значения относительных погрешностей расчета статистических оценок для частот встречаемости подграфов на четырех вершинах

На Рисунке 3.15 показана зависимость погрешности расчета статистических оценок для частот встречаемости подграфов от числа опытов для графа GenReg.

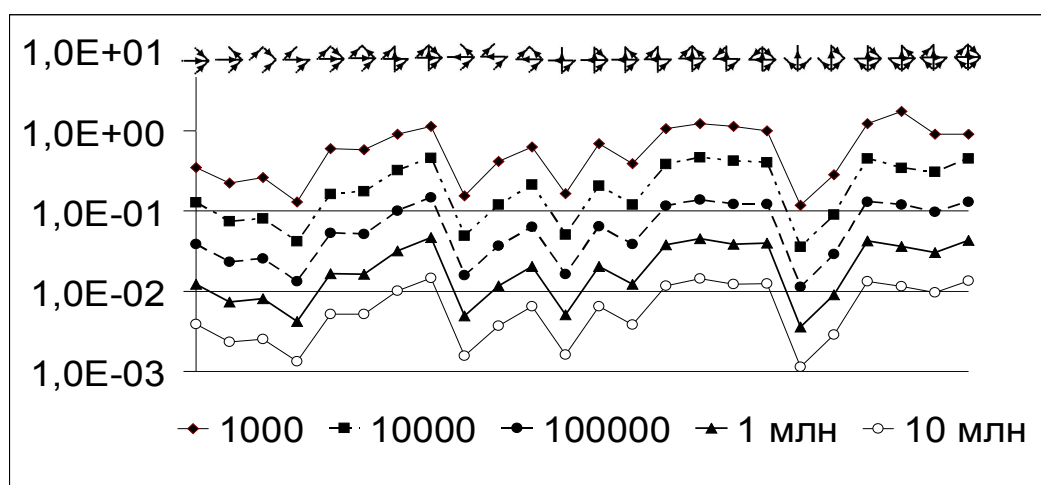


Рисунок 3.15– Значения относительных погрешностей расчета статистических оценок для частот встречаемости подграфов на четырех вершинах

С увеличением числа опытов N на порядок погрешность расчета для каждого подграфа уменьшается в $\sqrt{10} \approx 3$ раза. Такая же зависимость справедлива при планировании статистического эксперимента для получения статистической оценки общего числа подграфов, которое считается как сумма статистических

оценок для частот встречаемости подграфов различных классов изоморфизма:

$$\hat{N} = \sum_{i \in X} \hat{n}_i, \text{ при этом СКО вычисляется по формуле } \sigma(\hat{N}) = \sqrt{\sum_{i \in X} \sigma^2(\hat{n}_i)}.$$

Так, на Рисунке 3.16 представлены расчетные значения $\sigma(\hat{N})$ и прогнозные, получаемые с использованием расчетных при меньшем на порядок числе опытов.

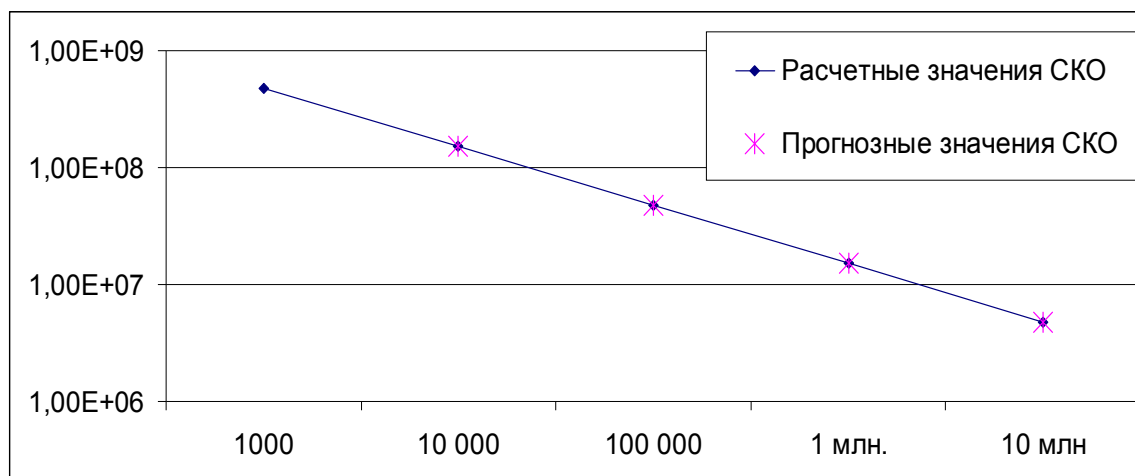


Рисунок 3.16 – Значения среднеекватратичного отклонения для расчетных оценок общего числа подграфов на четырех вершинах и прогнозных значений

3.6.2 Скорость расчетов

Рассмотрим время проведения расчетов при получении статистических оценок для частот встречаемости подграфов методом Верника-Раше (*) и методом случайного выбора остовных деревьев. В Таблице 3.8 приводится время, требуемое на расчет методом Верника-Раше (*), когда рассматривалось в среднем 100 000 подграфов. Также приводится время расчета методом случайного выбора остовных деревьев, когда проводились статистические эксперименты по 100 000 реализаций остовных деревьев типа «4-ветвление» и 100 000 реализаций «4-пути». Таким образом, при расчете методом случайного выбора остовных деревьев число просмотренных реализаций подграфов в два раза превышает это число при использовании метода Верника-Раше (*).

Как можно видеть из Таблицы 3.8, в рассмотренном случае метод случайного выбора остовных деревьев позволяет ускорить расчет встречаемости подграфов по сравнению с методом Верника-Раше(*).

Таблица 3.8 – Время расчета статистических оценок для частот встречаемости подграфов методом случайного выбора остовных деревьев и методом Верника-Раше (*), реализованного в программах igraph и Fanmod

Граф	Время расчета, в секундах		
	igraph	Fanmod	Метод случайного выбора остовных деревьев
PPI_HS	338	451	31
Pathway Common	3337	315	167
GenReg	677	756	73
Twitter	1354	1255	173
Google Plus	719333	386828	10684

Использование параллельных вычислений JAVA, введенных стандартом JSR 335: Streams, позволяет еще больше ускорить расчет. В Таблице 3.9 оценивается ускорение, полученное путем распараллеливания.

Таблица 3.9 – Время (в секундах) расчета методом случайного выбора остовных деревьев с использованием распараллеливания

Граф	Время расчета без распараллеливания	Время расчета с распараллеливанием	Коэффициент ускорения
PPI_HS	31	8	3,875
Pathway Commons	167	46	3,630435
GenReg	73	13	5,615385
Twitter	173	46	3,76087
Google Plus	3793	1834	2,068157

При использовании JSR 335: Streams распараллеливалось выполнение блоков 12-22 схемы алгоритма на Рисунке 3.11 и блоков 12-24 схемы алгоритма на Рисунке 3.13.

Также возможна реализация распараллеливания работы блоков 4-8, 9-11, 23-25 схемы алгоритма на Рисунке 3.11 и блоков 4-11, 9-11, 25-27 схемы алгоритма на Рисунке 3.13.

3.7 Рандомизация графа и расчет статистической значимости подграфов

Расчет частот встречаемости является самой трудоемкой подзадачей в задаче выявления сетевых мотивов. Рассмотрим другие подзадачи – рандомизацию и выявление статистической значимости подграфов.

3.7.1 Алгоритм рандомизации графов

Рассмотрим алгоритм рандомизации графа, используемый в программах FANMOD [127] и MOTIFNET [118], являющийся одним из базовых при выявлении сетевых мотивов [70]. В данном алгоритме рандомизированный граф G_R формируется путем перенаправления ребер графа-копии графа G исследуемой сети. При этом структура графа изменяется, а степени связности (в том числе полустепень исхода и полустепень захода) отдельных вершин нет. Алгоритм рандомизации представлен на Рисунке 3.17.

Ввод: граф $G = (V, U, D)$;

per_edge – число выполнений перенаправления каждого ребра;

attempts – максимальное число попыток перенаправления ребра на одном шаге

Вывод: $G_R = (V_R, U_R, D_R)$

НАЧАЛО

1. $G_R(V_R, U_R, D_R) \leftarrow G(V, U, D)$

2. Для всех $i = 1, per_edge$ делай

 Для всех $j = 1, |U_R|$ делай

 Для всех $k = 1, attempts$ делай

$e_j = \{a, b\}$

$e_r = \{c, d\} \leftarrow$ получить случайное ребро из U_R

 ЕСЛИ $((\{a, d\} \notin U_R) \wedge ((\{c, b\} \notin U_R) \wedge ((\{a, d\} \notin D_R) \wedge ((\{c, b\} \notin D_R))$ делай

$U_R \leftarrow U_R \setminus \{e_j = \{a, b\}, e_r = \{c, d\}\}$

$U_R \leftarrow U_R \cup \{e_j = \{a, d\}, e_r = \{c, b\}\}$

 Выход из цикла по k

 Конец цикла по k

 Конец цикла по j

Конец цикла по i

3. Для всех $i = 1, per_edge$ делай

 Для всех $j = 1, |D_R|$ делай

 Для всех $k = 1, attempts$ делай

$e_j = \{a, b\}$

$e_r = \{c, d\} \leftarrow$ получить случайное ребро из D_R

 ЕСЛИ $((\{a, d\} \notin U_R) \wedge ((\{c, b\} \notin U_R) \wedge ((\{a, d\} \notin D_R) \wedge ((\{c, b\} \notin D_R))$ делай

$D_R \leftarrow D_R \setminus \{e_j = \{a, b\}, e_r = \{c, d\}\}$

$D_R \leftarrow D_R \cup \{e_j = \{a, d\}, e_r = \{c, b\}\}$

 Выход из цикла по k

 Конец цикла по k

 Конец цикла по j

Конец цикла по i

КОНЕЦ

Рисунок 3.17 – Алгоритм *Randomize* для рандомизации графа

В представленном алгоритме сначала выполняется рандомизация направленных ребер (шаг 2). Так, два направленных ребра $e_1 = \{a, b\}$ и $e_2 = \{c, d\}$ графа выбираются случайно. Сначала ребра $e_1 = \{a, b\}$ и $e_2 = \{c, d\}$ удаляются из графа, а потом формируются новые ребра с теми же именами e_1 и e_2 . Это происходит следующим образом. Вершина a связывается с вершиной d (формируя e_1), а затем вершина d связывается с вершиной c (формируя e_2). Перед осуществлением такого перенаправления необходимо убедиться, что между вершинами a и d , а также между вершинами c и b , до перенаправления ребер не существовало. Аналогично, на следующем шаге происходит рандомизация ненаправленных ребер (шаг 3).

Для каждого направленного или ненаправленного ребра процедура перенаправления выполняется несколько раз (*per_edge*), причем каждый раз выполняется несколько попыток перенаправления *attempts*, поскольку при случайном выборе ребер существует вероятность, что перенаправление между выбранными ребрами будет невозможно в силу наложенных ограничений.

На Рисунке 3.18 представлена иллюстрация успешного перенаправления направленных ребер e_j и e_R .

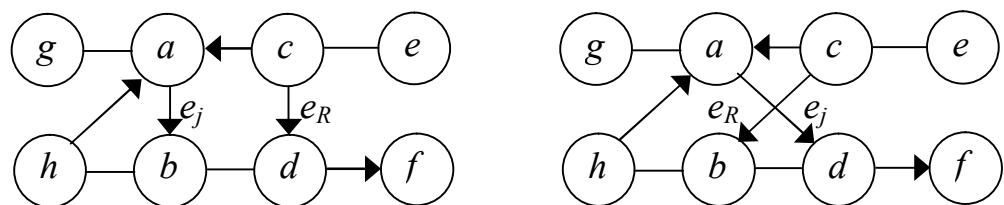


Рисунок 3.18 – Иллюстрация процесса перенаправления направленных ребер

3.7.2 Алгоритм вычисления статистической значимости подграфов

Использование для выявления сетевых мотивов показателя *Z-score* по формуле (1.7) требует значительного временного ресурса. В этом случае требуется провести расчет встречаемости подграфов не только для графа исследуемой сети, но и генерацию различных рандомизированных графов и расчет частот встречаемости подграфов для них. Только потом рассчитываются математическое ожидание частот встречаемости и стандартное отклонение по проведенным статистическим экспериментам. В Таблице 3.10 приводится среднее время, необходимое для проведения расчета статистических оценок для частот встречаемости подграфов на

четырёх вершинах с использованием разработанных алгоритмов. При этом в используемом методе случайного выбора остовных деревьев по обоим типам остовного дерева проводится по 100 000 случайных выборов реализаций.

Таблица 3.10 – Среднее время выявления сетевых мотивов

Граф	Время расчета (в секундах)	
	Процедура рандомизации	Расчет частот встречаемости подграфов
PPI_HS	6,5	21,3
Pathway Commons	82,6	116,1
GenReg	32,3	78,6
Twitter	80,637	111,8

Тем не менее, несмотря на полученное ускорение для больших графов требуется использовать другой показатель статистической значимости подграфов, позволяющий по одному расчету для рандомизированного графа и для исследуемого графа сети выявлять сетевые мотивы. Таким показателем, в частности, является показатель R [128], вычисляемый по формуле $R = n_G / M[n_R]$, где n_G – число подграфов заданного класса изоморфизма в графе, $M[n_R]$ – математическое ожидание частоты встречаемости подграфа в рандомизированных версиях графа. Как правило [128], сетевым мотивом называют подграф при $R > 1.1$.

Однако в случае использования метода случайного выбора остовных деревьев могут быть получены не точные значения n_G и n_R , а лишь их точечные и интервальные оценки (Рисунок 3.19).

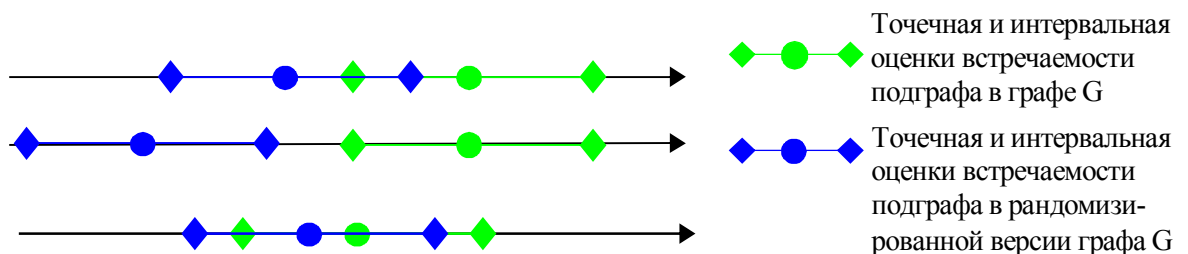


Рисунок 3.19 – Возможные точечные и интервальные оценки встречаемости подграфа в исследуемом графе и в его рандомизированной версии

Введем показатель значимости R' подграфа, вычисляемый по формуле:

$$R' = (\hat{n}_G - 3\sigma_G) / M[\hat{n}_R + 3\sigma_R],$$

где $\hat{n}_G$, $\hat{n}_R$ – оценки частот встречаемости подграфа, полученные при расчете в графе и его рандомизированной версии соответственно; σ_G , σ_R – соответствующие среднеквадратичные отклонения.

На Рисунке 3.20 представлен алгоритм, предназначенный для выявления сетевых мотивов на четырех вершинах с использованием трех различных показателей статистической значимости подграфов.

Ввод: граф G , N – параметр алгоритма *MFS* (для вычисления подграфов методом случайного выбора остовных деревьев), n – количество рандомизированных графов, *per_edge*, *attempts* – параметры алгоритма *Randomize* (для рандомизации графа G), *isA* – флаг проведения расчетов для абсолютных частот встречаемости подграфов

Вывод: $N_G[218]$ – статистические оценки абсолютных частот встречаемости подграфов в исследуемом графе, а также значения погрешности расчетов $3\sigma_G$ [218]; $N_{GR}[n][218]$ – статистические оценки абсолютных частот встречаемости подграфов в n рандомизированных версиях графа G , $3\sigma_{GR}[n][218]$ – соответствующие значения СКО, $R[218]$, $R'[218]$ $Z\text{-score}[218]$ – показатели статистической значимости R , R' и $Z\text{-score}$ соответствующих подграфов.

НАЧАЛО

1. Расчет частот встречаемости подграфов в исследуемом графе G

$$N_G[*], 3\sigma_G[*] \leftarrow MFS(G, N, isA)$$

2. Для $i=1, n$ Делай

2.1. $G' \leftarrow Randomize(G, per_edge, attempts)$

2.2. $N_{GR}[i][*], 3\sigma_{GR}[i][*] \leftarrow MFS(G', N, isA)$

2.3. Для $j = 1, 218$ Делай

$$R' = R' + (N_G[j] - 3\sigma_G[j]) / (N_{GR}[j] + 3\sigma_{GR}[j]) / n$$

$$R = R + N_G[j] / (N_{GR}[j] / n)$$

3. Расчет $Z\text{-Score}$

3.1. Для всех $i = 1, 218$ делай

3.1.1. $MX \leftarrow 0, \sigma_X \leftarrow 0,$

3.1.2. Для всех $j = 1, n$ делай

$$MX \leftarrow MX + N_{GR}[i][j] / n,$$

$$\sigma_X \leftarrow \sigma_X + (N_{GR}[i][j])^2 / n$$

3.1.3. $\sigma_X \leftarrow (\sigma_X - (MX)^2) \cdot n / (n - 1)$

3.1.3. $Z\text{-score}[i] \leftarrow (N_G[i] - MX) / \sigma_X$

КОНЕЦ

Рисунок 3.20 – Алгоритм выявления сетевых мотивов на четырех вершинах

В Таблице 3.11 приведены результаты анализа сетевых мотивов в неориентированном графе – графе PPI_HS, расчет выполнен менее чем за 30 секунд. По каждому типу остовного дерева выбрано по 100 000 случайных реализаций.

В Таблице 3.12 приведены результаты исследования сетевых мотивов в ориентированном графе GenReg, расчет R' выполнен менее чем за 2 минуты. По каждому типу остовного дерева выбрано по 100 000 случайных реализаций.

Таблица 3.11 – Результаты исследования сетевых мотивов в графе PPI_HS

id	$\hat{n}_G$	$3\sigma[\hat{n}_G]$	$M[\hat{n}_R]$	$M[3\sigma[\hat{n}_R]]$	R	R'
94	1,18E+10	9743510,795	1,19E+10	8932801,359	0,9886	0,9839
125	4,88E+09	9571997,962	5,67E+09	9254669,541	0,8608	0,8515
142	7,06E+08	5318599,276	6,31E+08	5166034,595	1,1185	1,0670
203	4,66E+07	1767784,486	3,19E+07	1509825,958	1,4606	1,1333
205	3,86E+07	1299471,459	1,89E+07	929136,3664	2,0395	1,5982
217	4277393,39	433502,0787	470413,533	146642,355	9,0928	3,2701

Таблица 3.12 – Результаты исследования сетевых мотивов в графе GenReg

id	$\hat{n}_G$	$3\sigma[\hat{n}_G]$	$M[\hat{n}_R]$	$M[3\sigma[\hat{n}_R]]$	R
3	9,1750E+08	1,0711E+09	0,8566	0,7944	-0,2125
7	2,4150E+09	2,5841E+09	0,9346	0,8934	0,1104
12	2,4994E+09	2,8199E+09	0,8864	0,8426	-0,0296
13	6,9143E+09	8,8490E+09	0,7814	0,7619	-0,4433
14	2,0649E+08	9,1315E+07	2,2613	1,9320	4,9079
17	2,1207E+08	8,8373E+07	2,3997	2,0491	5,5049
19	4,1816E+07	3,9063E+07	1,0705	0,6887	0,6199
20	8,9347E+06	1,2634E+06	7,0722	2,6975	36,8311
24	4,4803E+09	4,7995E+09	0,9335	0,9055	0,0953
29	1,2684E+09	2,2750E+09	0,5575	0,5221	-1,2077
35	1,4025E+08	5,1712E+07	2,7122	2,2189	6,1270
41	5,2174E+09	6,9174E+09	0,7542	0,7315	-0,5620
42	1,3556E+08	7,0856E+07	1,9132	1,5896	3,5735
47	3,7075E+08	1,7072E+08	2,1717	1,9336	4,6475
52	3,4448E+07	3,8433E+07	0,8963	0,5603	-0,3249
58	1,0032E+07	1,4154E+06	7,0874	2,8405	26,2518
66	3,1451E+07	1,8959E+07	1,6589	0,9470	1,9731
67	1,3481E+07	2,0614E+06	6,5397	2,9901	43,4739
76	6,8123E+09	7,2798E+09	0,9358	0,9154	0,0830
77	6,5371E+08	2,4802E+08	2,6357	2,4031	5,7933
78	1,2740E+07	1,1991E+06	10,6245	4,2876	38,4280
80	1,5866E+07	2,1128E+06	7,5095	3,5385	32,9139
82	2,2976E+07	2,5780E+06	8,9125	4,5779	44,1505
83	6,2056E+06	5,4260E+04	114,3679	14,5793	31,4693

В Таблице 3.12 приведены значения показателей значимости подграфов, выполненные по 10 различным реализациям рандомизированных графов с помощью разработанной программы MFSView. При этом на процедуру рандомизации в программе MFSView затрачено 320 секунд, а на расчет частот встречаемости подграфов 860 секунд. Для сравнения при использовании FANMOD при тех же параметрах рандомизации и меньшей точности расчетов требуется порядка 16 000 секунд.

3.8 Выводы по третьей главе

1. В главе предложен алгоритм выявления сетевых мотивов на четырех вершинах, использующий в качестве составных частей следующие алгоритмы.

а) Алгоритм рандомизации графа, эффективно использующий выбранную структуру хранения графа.

б) Алгоритм расчета статистических оценок для частот встречаемости подграфов на четырех вершинах с использованием метода случайного выбора остовных деревьев. Алгоритм позволяет использовать один из двух режимов работы:

- режим, гарантирующий минимизацию абсолютной погрешности при получении статистических оценок для частот встречаемости подграфов;
- режим, гарантирующий минимизацию относительной погрешности при получении статистических оценок для частот встречаемости подграфов.

Алгоритм выявления сетевых мотивов реализует расчет как известных показателей значимости сетевых мотивов (*Z-Score* и *R*), так и предложенный в рамках диссертационной работы показатель значимости подграфов R' .

Показатель статистической значимости подграфов R' позволяет статистически достоверно выявлять сетевые мотивы даже по двум статистическим расчетам частот встречаемости подграфов: один – в исследуемом графе сети, другой – в его рандомизированной версии.

2. В алгоритме расчета статистических оценок для частот встречаемости подграфов на четырех вершинах требуется проведение статистических

экспериментов с использованием остовных деревьев типа «4-ветвление» и «4-путь». Для их ускоренного выполнения предложены следующие алгоритмы.

а) Алгоритм ускоренного равновероятного выбора « k -ветвлений», основанный на методе стратификации (расслоения) вершин по их степени связности. При этом увеличивается скорость расчета и уменьшаются требования к объему дополнительной оперативной памяти, если сравнивать с использованием другого ускоренного алгоритма, предложенного в работе [113]. Приведены математические расчеты и результаты экспериментов с использованием графа БА и графов исследуемых в диссертации сетей. Анализ результатов расчетов подтверждает эффективность предложенного алгоритма случайного выбора « k -ветвления».

б) Алгоритм равновероятного выбора остовных деревьев типа «4-путь», позволяющий ускорить процедуру случайного выбора «4-путей» по сравнению с другим алгоритмом, предложенным в работе [80]. Предложенный алгоритм, как и в методе случайного выбора остовных деревьев типа «4-ветвление», основан на методе расслоения, только расслоение выполняется по ребрам, а не по вершинам. Приведены математические расчеты и результаты экспериментов, подтверждающие эффективность предложенного ускоренного алгоритма выбора «4-путей».

Предложенная реализация метода случайного выбора остовных деревьев позволила увеличить скорость расчета статистических оценок для частот встречаемости подграфов по сравнению с известным статистическим методом Верника-Раше, реализованного в программе FANMOD и в библиотеке *igraph* для среды R при большей точности расчета.

Использование параллельных вычислений JAVA, введенных стандартом JSR 335: Streams, позволило еще больше ускорить расчет статистических оценок для частот встречаемости подграфов.

ГЛАВА 4 КОМПЛЕКС ПРОГРАММ

В четвертой главе описаны разработанные в рамках диссертационной работы библиотеки классов и программы. Разработанный комплекс программ использует следующие свободно распространяемые сторонние библиотеки с открытым программным кодом.

1) Библиотека классов JUNG (<http://jung.sourceforge.net/>), предназначенная для работы с графами.

2) Библиотека классов Common Collections, поддерживающая дополнительные контейнеры для хранения объектов (<https://commons.apache.org/proper/commons-collections/>).

3) Библиотека классов Colt (<http://dsd.lbl.gov/~hoschek/colt/>), содержащая классы для вычислений с использованием матричной алгебры, классы, реализующие генераторы случайных чисел, средства сбора и анализа статистики.

В созданном комплексе программ реализованы предложенные в диссертационной работе методы и алгоритмы [26-34].

Разработанные библиотеки классов SocAndBioNetworksAnalysis, MotifsLib и YMN_GraphGenerators зарегистрированы как электронные ресурсы в ОФЭРНИО. На программу для выявления сетевых мотивов MFSView и систему агентного моделирования Simbigraph 2 получены свидетельства о государственной регистрации программ для ЭВМ. Все разработанные библиотеки и программы размещены в репозиториях GITHUB и доступны по ссылке: <https://github.com/MNYudina>.

В главе также описаны полученные при внедрении результатов диссертационной работы качественные и количественные преимущества разработанных алгоритмов и программ.

4.1 Библиотека классов SocAndBioNetworksAnalysis

Библиотека классов SocAndBioNetworksAnalysis содержит классы на языке JAVA для анализа больших сетей, в том числе сетей интернета и молекулярных сетей клетки.

С разработанной библиотекой предоставляются графы: 1) E-mail EU-All, 2) Twitter, 3) E-mail Enron, 4) VKOmgту, 5) PPI_HS, 6) YeastS, 7) GenReg, 8) Regulon.

Библиотека классов SocAndBioNetworksAnalysis обеспечивает следующую функциональность.

1. Сохранение графа в формате списка ребер.
2. Загрузка графа в формате списка ребер.
3. Загрузка графа в формате Pajek.
4. Определение максимальной степени связности вершин графа.
5. Расчет и вывод PCC графа в файл.
6. Визуализация графа (Рисунок 4.1).
7. Расчет оптимальной стратегии в сетевой игровой модели «Координационная игра».

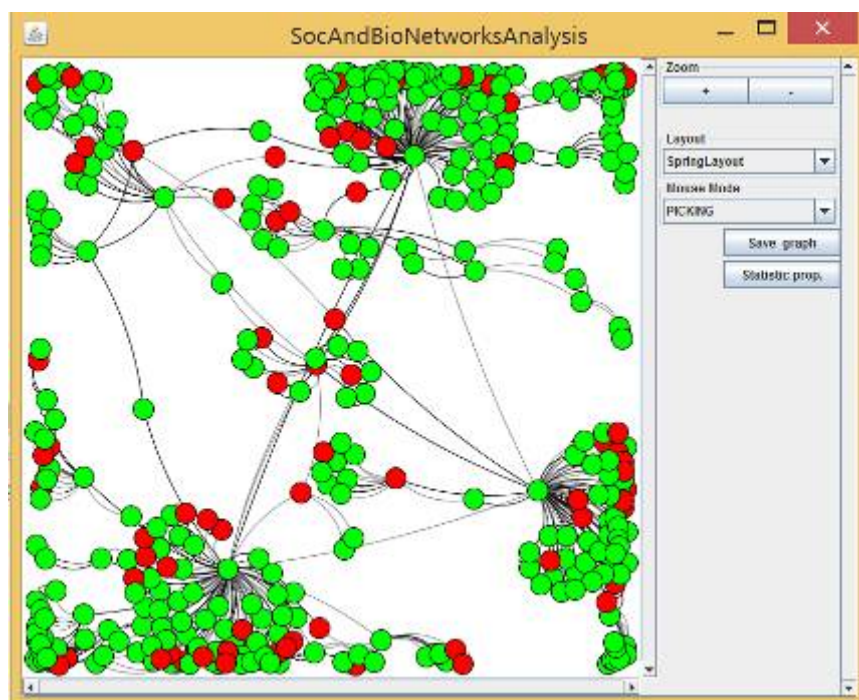


Рисунок 4.1 – Визуализация графов средствами SocAndBioNetworksAnalysis

Для работы с классами, реализованными в библиотеке, необходимо скачать исходный код по адресу: github.com/MNYudina/SocAndBioNetworksAnalysis.

4.2 Библиотека классов MotifsLib и программа MFSView

Библиотека классов MotifsLib является библиотекой на языке JAVA, которая содержит классы, обеспечивающие реализацию метода случайного выбора остовных деревьев при расчете статистических оценок для частот встречаемости связных подграфов на трех и четырех вершинах.

При использовании MotifsLib предусмотрены следующие три режима вывода результатов расчета.

1) Стандартный вывод (Рисунок 4.2).

2) Вывод результатов и погрешности расчетов в абсолютных значениях (Рисунок 4.3).

3) Вывод результатов и погрешности расчетов в относительных значениях (Рисунок 4.4).

```
Vertices:9927
Edges:35597
Arcs:0
-----
id      freq. motifs
-----
94:      3.642784867803E7
125:     1.9285963541106336E7
142:     3221141.0072616986
203:     104021.96168243399
205:     210254.4060116388
217:     33320.12344445607

Время расчета: 2.209 сек.
Выполнено: 100000 опытов по каждому типу остовных деревьев
```

Рисунок 4.2 – Расчет частот встречаемости подграфов в MotifsLib (стандартный вывод)

```
Vertices:9927
Edges:35597
Arcs:0
id      freq. motifs      (+/-)      3*sigma
-----
94:      3.6453181817820005E7      (+/-)      111039.14242670304
125:     1.9335679996649817E7      (+/-)      121772.59962938941
142:     3200854.1586351185      (+/-)      65250.334697464124
203:     98135.13685900313      (+/-)      15642.018525837382
205:     213823.30891686783      (+/-)      17725.63813121772
217:     32799.565499108634      (+/-)      6962.066416130458

Время расчета: 2.328 сек.
Выполнено: 100000 опытов по каждому типу остовных деревьев
```

Рисунок 4.3 – Расчет частот встречаемости подграфов в MotifsLib (вывод частот встречаемости и абсолютной погрешности расчетов)

```

Vertices:9927
Edges:35597
Arcs:0
id      freq. motifs      (+/-)      3*variation factor
94:      3.6407743011530004E7  (+/-)      0.3066346465115785%
125:      1.9372598426369727E7  (+/-)      0.6277615196478378%
142:      3226892.545412456      (+/-)      2.0294886454560586%
203:      95614.99409841295      (+/-)      16.147157837930287%
205:      207754.1939741662      (+/-)      8.410221038715408%
217:      32056.253131758138      (+/-)      21.46994529650755%

Время расчета: 2.247 сек.
Выполнено: 100000 опытов по каждому типу остовных деревьев

```

Рисунок 4.4 – Расчет статистических оценок для частот встречаемости подграфов в MotifsLib (вывод частот встречаемости и относительной погрешности расчета)

Библиотека MotifsLib представляет собой библиотеку с открытым кодом, которая размещается по адресу: github.com/MNYudina/MotifsLib.

Разработанная на основе библиотек MotifsLib и YMN_GraphGenerators программа MFSView имеет графический интерфейс по аналогии с программой FANMOD и размещается по адресу: github.com/MNYudina/MFSView.

Описанные выше программы были использованы для экспресс-анализа в рамках разработки эффективных стратегий информационной деятельности ОмГТУ в социальной сети «ВКонтакте». В Таблице 4.1 представлены результаты расчетов статистических оценок для частот встречаемости подграфов. Оценки получены программой MFSView, в том числе приведены показатели значимости R и R' . Также приведены результаты вычислений $Z\text{-score}$, выполненные в программе FANMOD, Таблица 1.4. Как можно видеть из Таблицы 4.1, при использовании «стандартных значений» $R > 1,1$, $R' > 1$ и $Z\text{-score} > 2$ выявляются те же сетевые мотивы, что и при точном расчете программой FANMOD, это подграфы с идентификаторами 142, 203, 205, 217. Показатели статистической значимости подграфов, соответствующие сетевым мотивам, выделены серым цветом. Использование программы MFSView при выявлении сетевых мотивов в графе VKOmgту позволило сократить время расчетов с 16 часов при работе с программой FANMOD до пары секунд.

Таблица 4.1 – Результаты выявления сетевых мотивов в графе VKOmgTu

id	$\hat{n}_G$	$3\sigma[\hat{n}_G]$	$M[\hat{n}_R]$	$M[3\sigma[\hat{n}_R]]$	R	R'	$Z\text{-Score}$
94	3,6431E+07	3,7112E+04	3,9414E+07	1,7724E+04	0,9243	0,9202	-20,112
125	1,9361E+07	4,0556E+04	2,0889E+07	1,8763E+04	0,9268	0,9185	-7,6054
142	3,2168E+06	2,1799E+04	7,5442E+05	1,0295E+04	4,2640	4,0130	128,26
203	9,2010E+04	5,0490E+03	3,5203E+04	2,8163E+03	2,6137	1,7608	8,9843
205	2,1147E+05	5,8759E+03	8,7283E+03	1,1230E+03	24,2282	16,0236	750,97
217	3,1995E+04	2,2920E+03	5,6415E+01	1,1283E+02	567,1389	63,6079	60424

Программа MFSView использована в комплексе с библиотекой SocAndBioNetworksAnalysis для анализа сети ссылок веб-страниц [20] – граф OmskWeb.

В Таблице 4.2. приведены результаты вычислений, выполненные в программе MFSView, при этом для расчетов генерировалось 100 рандомизированных графов. При расчете статистических оценок для частот встречаемости подграфов использовался метод случайного выбора остовных деревьев по 100 000 реализациям на каждый тип остовного дерева. При этом выводились статистические оценки для относительных частот встречаемости подграфов. В Таблице 4.2 приведены оценки относительных частот встречаемости подграфов $\hat{C}_G$ в графе OmskWeb, а также средние значения оценок частот встречаемости $M[\hat{C}_R]$ по рандомизированным версиям графа OmskWeb. Для сравнения приведены также точные значения частот встречаемости подграфов C_G . В Таблице 4.2 также представлены полученные значения показателей значимости подграфов: R_C и $Z\text{-score}_C$. Серым цветом выделены значения показателей значимости подграфов, которые соответствуют сетевым мотивам, в соответствии с принятыми значениями этих показателей: $R_C > 1.1$ и $Z\text{-score} > 2$. При этом на процедуру рандомизации в программе MFSView потрачено 247 секунд, а на расчет подграфов 597 секунд. При использовании FANMOD в режиме статистического расчета при меньшей точности вычислений и одинаковых параметрах рандомизации графов требуется 6120 секунд. Из них на процедуру рандомизации – 559 секунд, а на расчет частот встречаемости подграфов – 5561 секунда. Следует заметить, что программа FANMOD в этом режиме пригодна только для получения относительных частот встречаемости подграфов. Причем для

получения равновероятно 100 000 подграфов на четырех вершинах вероятность выбора на последнем ярусе Верника-Раше в FANMOD должна быть равна $9,0694E-6$ из условия $100\,000/N$, где N – общее число всех подграфов, и оно неизвестно без точного расчета. А пересчет всех подграфов на четырех вершинах в графе OmskWeb требует в программе FANMOD более 20 000 секунд.

Таблица 4.2 – Результаты выявления сетевых мотивов в графе OmskWeb

id	$\hat{C}_G$	C_G	$M[\hat{C}_R]$	R_C	$Z_C\text{-Score}$
3	0,832695	0,832695	0,749467	1,113092	95,87332
12	0,006275	0,006275	0,006662	0,948189	-1,82213
13	0,038021	0,038021	0,10729	0,354369	-143,444
14	0,003155	0,003155	0,00295	1,055269	1,513847
19	0,000635	0,000635	0,000385	1,640135	15,82531
20	0,000121	0,000121	1,57E-05	8,373142	15,27941
21	7,62E-05	0,000076	5,06E-08	1502,17	179,5532
41	0,003817	0,003817	0,004862	0,783465	-9,49725
43	0,000459	0,000459	0,001727	0,271139	-18,5372
45	0,003287	0,003287	0,01555	0,207467	-56,7462
47	0,000319	0,000319	8,97E-05	3,702296	27,66375
56	0,00772	0,007720	0,001647	4,639518	74,76263
76	0,077792	0,077792	0,070044	1,088131	8,314407
77	0,00065	0,000650	0,00037	1,815755	14,54241
82	0,000167	0,000167	2,02E-06	85,71977	217,5231
89	3,18E-05	0,000032	1,37E-09	23479,9	2359,718
92	0,005696	0,005696	0,006595	0,837725	-4,44628
94	0,003155	0,003155	0,00469	0,649645	-7,14562
96	0,002014	0,002014	0,001756	1,180346	5,154595
100	0,000328	0,000328	0,006253	0,054429	-60,563
103	0,000859	0,000859	0,000105	8,14376	80,58834
139	0,001041	0,001041	9,24E-05	11,18652	93,5445
176	0,000373	0,000373	5,47E-07	691,7545	514,9179
184	4,07E-05	0,000041	1,66E-08	2504,383	924,4296
213	0,000571	0,000571	2,21E-07	2588,163	1159,657
217	0,000567	0,000567	1,67E-07	3320,463	1789,312

4.3 Система агентного моделирования Simbigraph 2

В рамках диссертационной работы разработан также модуль для системы имитационного агентного моделирования Simbigraph 2. Использование системы Simbigraph 2 ориентировано на моделирование больших сетей: сетей интернета,

сетей белковых взаимодействий, генных сетей. Такие сети состоят из сотен тысяч узлов и связей между ними, что требует достаточно высокого уровня аппаратного обеспечения компьютера пользователя. Для обеспечения комфортной работы с графами сетей, содержащими большое число вершин и ребер, увеличиваются требования к тактовой частоте процессора и объему оперативной памяти.

Программа Simbigraph 2 (Рисунок 4.5) ориентирована на моделирование сетей и сетевых процессов различной природы в больших сетевых структурах методами агентного моделирования. Новая версия Simbigraph 2 включает в себя функционал системы агентного моделирования Simbigraph с переработанным интерфейсом и рядом новых модулей. В число новых модулей вошел модуль, осуществляющий расчет метрических характеристик сетей (диаметр графа, радиус, нахождение периферийных вершин и центра графа), модуль расчета статистических оценок для частот встречаемости подграфов на трех и четырех вершинах. В новой системе активно используются возможности параллельных вычислений Java: JSR 335: Streams.

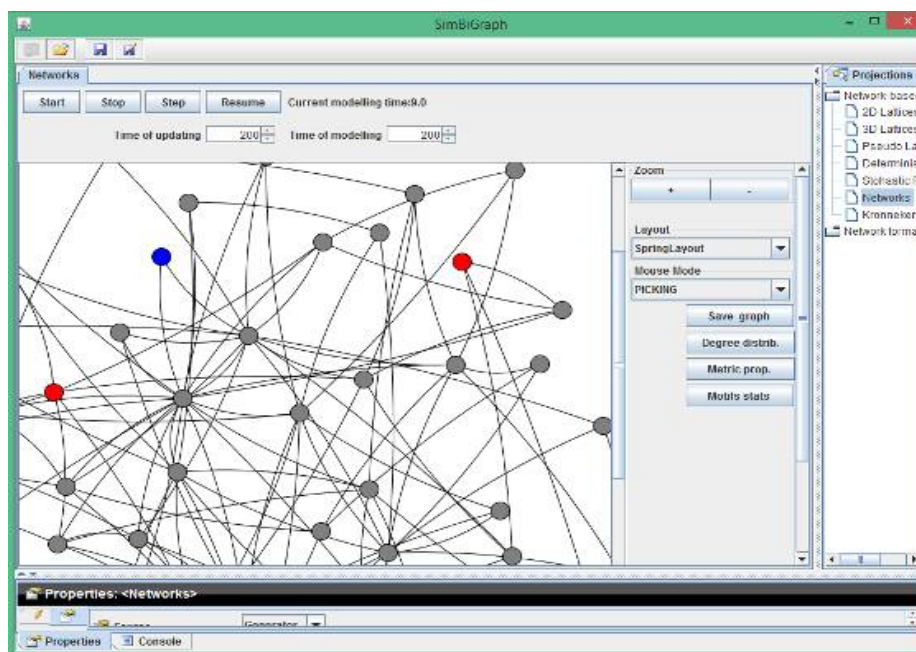


Рисунок 4.5 – Интерфейс программы Simbigraph 2

Творческим вкладом автора диссертации при создании Simbigraph 2 является разработка модуля, предназначенного для статистического расчета частот встречаемости подграфов на трех и четырех вершинах.

4.4 Выводы по четвертой главе

1. Разработанная библиотека классов SocAndBioNetworksAnalysis предназначена для анализа и визуализации графов сетей. Она успешно использовалась для анализа подсети социальной сети «ВКонтакте», узлами которой являются аккаунты, владельцы которых указали местом учебы ОмГТУ (акт внедрения в отделе корпоративных средств массовой информации ОмГТУ, Приложение Б) и для исследования сети ссылок веб-страниц (акт внедрения в АН ППО «МАНО», Приложение Б). Функционал библиотеки позволяет визуализировать граф, рассчитывать структурные вероятностные характеристики графов больших сетей: распределение степени связности вершин, коэффициент кластеризации и т.д. Библиотека может использоваться для анализа сетевых процессов информационного противодействия, выявлять узлы и множества узлов, играющих ключевую роль в информационном взаимодействии.

2. Разработанная библиотека классов MotifsLib реализует расчет статистических оценок для частот встречаемости подграфов методом случайного выбора остовных деревьев. Также библиотека MotifsLib содержит дополнительные классы для эффективной реализации процедуры рандомизации графов и выявления сетевых мотивов.

3. Разработанная программа для выявления сетевых мотивов MFSView отличается развитым графическим интерфейсом, позволяет настраивать параметры расчета и форму представления результатов. При использовании программы удастся существенно уменьшить время выявления сетевых мотивов. Программа, ее код и документация размещены по адресу <https://github.com/MNYudina/MFSView>.

4. Разработанный модуль для выявления сетевых мотивов внедрен в систему агентного моделирования Simbigraph 2, предназначенную для исследования больших сетей и сетевых процессов.

Видеоматериалы, иллюстрирующие работу программ, доступны по ссылкам:

<https://www.youtube.com/watch?v=B2ugWLpLABE&t=210s>,

<https://www.youtube.com/watch?v=Toq0YuPHLko>.

ЗАКЛЮЧЕНИЕ

В работе рассмотрена задача, поставленная в рамках новой отрасли науки, известной как «Наука о сетях» (Network Science). Объектом исследования «Науки о сетях» являются большие сети различной природы: сети интернета, молекулярные сети клетки, сети электропередач и т.д.

Одним из подходов к исследованию больших сетей, выработанных в «Науке о сетях», является выявление в графах сетей «сетевых мотивов». Сетевые мотивы – это связные подграфы на заданном числе вершин, которые встречаются в графе сети чаще, чем в рандомизированных версиях этого графа («чаще, чем случайно»). Набор сетевых мотивов вместе с частотами их встречаемости в графе сети является уникальной особенностью всякой большой сети, отличающей ее от других сетей, в первую очередь от сетей другой природы. Анализ сетевых мотивов позволил сделать ряд открытий в области сетей белковых и генных взаимодействий и все чаще применяется при анализе сетей интернета.

Одной из наиболее трудоемких задач при выявлении сетевых мотивов является задача расчета частот встречаемости подграфов, относящихся к заданным классам изоморфизма. Высокой трудоемкостью характеризуется и задача сравнения частот встречаемости подграфов в исходном графе с частотами их встречаемости в рандомизированных версиях графа. Для решения этих двух задач в ходе диссертационных исследований получены следующие результаты.

1. Метод расчета статистических оценок для частот встречаемости подграфов – метод случайного выбора остовных деревьев. Этот метод является развитием методов случайного выбора пути, разрабатываемых в последние годы научными коллективами из разных стран, в том числе США, России и Китая. Но в отличие от методов случайного выбора пути метод случайного выбора остовных деревьев позволяет работать не только с неориентированными, но и с ориентированными, и со смешанными графами.

2. Получаемые методом случайного выбора остовных деревьев статистические оценки для частот встречаемости подграфов по равновероятным выборкам

подграфов из нескольких разнородных генеральных совокупностей. Разработанные оценки являются состоятельными, несмещенными и эффективными.

3. Ускоренные алгоритмы случайного выбора остовных деревьев типа «4-ветвление» и «4-путь». Эти алгоритмы необходимы для эффективной реализации метода случайного выбора остовных деревьев для получения статистических оценок для частот встречаемости подграфов на четырех вершинах. Предложенные алгоритмы используют идею стратификации (расслоения) вершин или ребер при проведении статистических экспериментов. В диссертации приводятся аналитический и экспериментальный анализ алгоритмов, показывающий, что предложенные алгоритмы, при их сравнении с другими известными ускоренными алгоритмами, позволили не только ускорить расчет, но и сократить объемы требуемой дополнительной оперативной памяти.

4. Программы расчета статистических оценок для частот встречаемости подграфов на четырех вершинах, которые при одинаковой точности расчетов оказались значительно быстрее программ, реализующих ускоренный метод Верника-Раше в библиотеке *igraph*, и программы FANMOD (наиболее популярных программ, используемых для этих целей). Несмотря на то, что программа FANMOD и библиотека *igraph* реализованы на языке программирования C++, а разработанные в рамках диссертации программы – на более медленном языке программирования Java. Еще более значительное ускорение удалось получить путем использования параллельных вычислений.

5. Комплекс проблемно-ориентированных программ, включающий программы для загрузки графов из популярных форматов их представления, программы для экспресс-анализа вероятностных характеристик графов, программы для моделирования сетевых процессов. Разработанные программы применялись для исследования ряда больших сетей. В частности, для анализа подсети социальной сети «ВКонтакте» и сети ссылок веб-страниц. При этом, как показано в четвертой главе, разработанные программы позволили существенно ускорить анализ сетей.

6. Показатель R' статистической значимости подграфов, обобщающий известный показатель R , используемый при точном расчете частот встречаемости подграфов за счет их полного перебора. Показатель R' основан на асимптотически нормальном распределении оценок частот, получаемых в методе случайного выбора остовных деревьев, что позволяет эффективно использовать доверительные интервалы, в частности, «правило трех сигм», для выявления сетевых мотивов.

Перспективным представляется развитие предложенного метода случайного выбора остовных деревьев для расчета частот встречаемости подграфов на пяти и более вершинах. Другой перспективной возможностью развития диссертационного исследования является реализация комплекса параллельных и распределенных алгоритмов и программ, реализующих разработанные методы.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

граф БА	– граф Барабаши-Альберт
ДНК	– дезоксирибонуклеиновая кислота
МБ	– мегабайт
РСС	– распределение степени связности вершин графа
к.в.	– коэффициент вариации
СКО	– среднеквадратичное отклонение
метод СВР	– метод случайного выбора ребра
методы ОГ	– статистические методы расчета частот встречаемости подграфов на основе обобщений графов
методы СБ	– статистические методы расчета частот встречаемости подграфов на основе случайного блуждания
методы СВП	– статистические методы расчета частот встречаемости подграфов на основе случайного выбора пути
методы СОВ	– статистические методы расчета частот встречаемости подграфов путем их перечисления со случайным отсечением выборки
ОЗУ	– оперативное запоминающее устройство
ЭВМ	– электронно-вычислительная машина
API	– интерфейс прикладного программирования
BGP	– протокол граничного шлюза
GITHUB	– веб-сервис для хранения программных проектов.
GPL	– универсальная общественная лицензия
HTML	– язык гипертекстовой разметки
IEEE	– институт инженеров электротехники и электроники
ISI	– институт научной информации
JDK	– комплект разработчика приложений на языке Java
JRE	– среда выполнения для Java
JSR	–запрос на спецификацию Java
TCGA	– the cancer genome atlas (атлас ракового генома)

СПИСОК ЛИТЕРАТУРЫ

1. Бадрызлов, В. А. Оценка эффективности распространения информации в социальных сетях с использованием имитационного моделирования / В. А. Бадрызлов, В. В. Сидельцев // Креативная экономика. – 2018. – Т. 12, № 9. – С. 1359–1372.
2. Буркин, А.В. Задачи о распределении подграфов в случайных графах: специальность 01.01.05 «Теория вероятностей и математическая статистика» : автореферат диссертации на соискание ученой степени кандидата физико-математических наук / Буркин Антон Валерьевич ; Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Московский физико-технический институт (государственный университет)». – Москва, 2019. – 16 с.: ил. – Библиогр.: с. 16. – Место защиты: Московский физико-технический институт (национальный исследовательский университет).
3. Губанов, Д. А. Социальные сети: модели информационного влияния, управления и противоборства / Д. А. Губанов, Д. А. Новиков, А. Г. Чхартишвили. – Изд. 3-е, перераб. и доп. – Москва : Изд-во МЦМНО, 2018. – 223 с. – ISBN 978-5-4439-1302-5.
4. Дёмина, А. Р. Расчет 4-мотивов неполной сети с использованием параллельных вычислений / А. Р. Дёмина, М. Н. Юдина // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 26–28 апр. 2016 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск : Изд-во ОмГТУ, 2016. – С. 79–83.
5. Дробышевский, М. Д. Методы и программные средства моделирования и генерации сложных сетей с сохранением графовых свойств : специальность 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей» : автореферат диссертации на соискание ученой степени кандидата физико-математических наук / Дробышевский Михаил Дмитриевич ; Федеральное государственное бюджетное учреждение науки Институт системного программирования им. В. П. Иванникова РАН. – Москва, 2019. – 20 с.: ил. – Библиогр.: с. 19. – Место защиты: Ин-т системного программирования им. В. П. Иванникова РАН.

6. Задорожный, В. Н. Растущие сети с потерями узлов / В. Н. Задорожный // Омский научный вестник. Сер. Приборы, машины и технологии. – 2017. – № 1 (151). – С. 108–113.
7. Задорожный, В. Н. Растущие сети: динамика распределения степеней связности смежных узлов / В. Н. Задорожный // Омский научный вестник. Сер. Приборы, машины и технологии. – 2016. – № 2 (146). – С. 81–86.
8. Задорожный, В. Н. Случайные графы с нелинейным правилом предпочтительного связывания / В. Н. Задорожный // Проблемы управления. – 2010. – № 6. – С. 2–11.
9. Задорожный, В. Н. Калибровка случайных графов предпочтительного связывания по распределениям степеней вершин и ребер / В. Н. Задорожный, Е. Б. Юдин // Омский научный вестник. Сер. Приборы, машины и технологии. – 2017. – № 1 (151). – С. 114–118.
10. Задорожный, В. Н. Структурные свойства безмасштабного графа Барабаши–Альберт / В. Н. Задорожный, Е. Б. Юдин // Автоматика и телемеханика. – 2012. – № 4. – С. 131–150.
11. Задорожный, В. Н. Распределения степеней в растущих графах теряющих дуги / В. Н. Задорожный, Е. Б. Юдин, М. Н. Юдина // Омский научный вестник. Сер. Приборы, машины и технологии. – 2017. – № 5 (155). – С. 159–163.
12. Задорожный, В. Н. Методы ускоренного расчета и параметрической оптимизации высоконадежных структурно сложных систем / В. Н. Задорожный, М. Н. Юдина // Омский научный вестник. Сер. Приборы, машины и технологии. – 2016. – № 4 (148). – С. 132–138.
13. Зыков, В. Роскомнадзор запретил сбор данных пользователей «ВКонтакте» / В. Зыков // Известия. – 2017. – 31 июля. – URL: <https://iz.ru/625543/vladimir-zykov/roskomnadzor-zapretil-sobirat-dannye-polzovatelei-vkontakte> (дата обращения: 10.09.2019).
14. Кнут, Д. Э. Искусство программирования. В 4 т. Т. 2. Основные алгоритмы / Д. Э. Кнут. – Москва : Вильямс, 2019. – 720 с.

15. Кнут, Д. Конкретная математика. Основание информатики / Д. Кнут, Р. Грэхем, О. Паташник. – Москва: Мир, 2009. – 703 с.
16. Мигов, Д. А. Расчет вероятности связности случайного графа с применением сечений: специальность 05.13.18 «Математическое моделирование, численные методы и комплексы программ»: автореферат диссертации на соискание ученой степени кандидата физико-математических наук / Мигов Денис Александрович ; Институт вычислительной математики и математической геофизики СО РАН. – Новосибирск, 2008. – 15 с.: ил. – Библиогр.: с. 13–15. – Место защиты: Ин-т вычислительной математики и математической геофизики СО РАН.
17. Ниткин, Д. А. Программа для сбора информации о взаимосвязях пользователей социальной сети «ВКонтакте» / Д. А. Ниткин, М. Н. Юдина // Динамика систем, механизмов и машин. – 2016. – № 3. – С. 32–35.
18. Ниткин, Д. А. Анализ выборки пользователей из социальной сети «ВКонтакте» / Д. А. Ниткин, М. Н. Юдина // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 26–28 апр. 2016 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск : Изд-во ОмГТУ, 2016. – С. 112–116.
19. Павлов, Д. А. Каноническая нумерация графов и библиотека nauty / Д. А. Павлов // Компьютерные инструменты в образовании. – 2009. – № 5. – С. 27–34.
20. Пахомов, И. А. Сбор и анализ информации о структуре сети ссылок веб-страниц / И. А. Пахомов, М. Н. Юдина // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 18–19 мая 2017 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск : Изд-во ОмГТУ, 2017. – С. 117–122.
21. Прохоренкова, Л. А. Свойства случайных веб-графов, основанных на предпочтительном присоединении: специальность 01.01.05 «Теория вероятностей и математическая статистика» : автореферат диссертации на соискание ученой степени кандидата физико-математических наук / Прохоренкова Людмила Александровна ; Московский государственный университет им. М. В. Ломоносова.

– Москва, 2015. – 20 с.: ил. – Библиогр.: с. 19. – Московский государственный университет им. М. В. Ломоносова.

22. Самосват, Е. А. Моделирование интернет а с помощью случайных графов: специальность 05.13.18 «Математическое моделирование, численные методы и комплексы программ» : автореферат диссертации на соискание ученой степени кандидата физико-математических наук / Самосват Егор Александрович ; Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Московский физико-технический институт (государственный университет)». – Москва, 2014. – 29 с.: ил. – Библиогр.: с. 28. – Место защиты: Федеральное государственное бюджетное учреждение науки «Вычислительный центр имени А.А. Дородницына Российской академии наук».

23. Седжвик, Р. Фундаментальные алгоритмы на C++. Алгоритмы на графах / Р. Седжвик. – Санкт-Петербург : ООО «ДиаСофтЮП», 2002. – 496 с.

24. Юдин, Е. Б. Расчет надежности с использованием свойств монотонности функций и эквивалентных преобразований графа надежности / Е. Б. Юдин, В. Н. Задорожный, М. Н. Юдина // Омский научный вестник. Сер. Приборы, машины и технологии. – 2014. – № 3 (133). – С. 201–204.

25. Юдин, Е. Б. Расчет числа сетевых мотивов методом случайной выборки каркасов / Юдин Е. Б., В. Н. Задорожный // Омский научный вестник. Сер. Приборы, машины, технологии. – 2015. – № 2 (140). – С. 208–211.

26. Юдин, Е. Б. Модуль анализа частот встречаемости типовых подграфов в системе агентного моделирования Simbigraph / Е. Б. Юдин, М. Н. Юдина // Имитационное моделирование. Теория и практика (ИММОД-2017) : тр. Восьмой Всерос. науч.-практ. конф., 18–20 окт. 2017 г. – Санкт-Петербург : НП «НОИМ», 2017. – С. 304–309.

27. Юдина, М. Н. Подсчет частот встречаемости типовых подграфов в направленных сетях / М. Н. Юдина // Математическое и компьютерное моделирование : сб. материалов V Междунар. науч. конф. – Омск : Изд-во ОмГУ, 2017. – С. 135–138.

28. Юдина, М. Н. Использование сложных приращений в графах предпочтительного связывания / М. Н. Юдина // Информационный бюллетень Омского научно-образовательного центра ОмГТУ и ИМ СО РАН в области математики и информатики. – 2017. – Т. 1, № 1. – С. 130–132.

29. Юдина, М. Н. Используемые алгоритмы для подсчета частот встречаемости типовых подграфов в системе моделирования simbigraph 2 / М. Н. Юдина // Информационные технологии и автоматизация управления : материалы IX Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 17–19 мая 2018 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск: Изд-во ОмГТУ, 2018. – С. 190–196.

30. Юдина, М. Н. Модели игр на сетях: «Простые компоненты», «Лучший выбор», «Координационная игра» / М. Н. Юдина // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 26–28 апр. 2016 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск: Изд-во ОмГТУ, 2016. – С. 168–172.

31. Юдина, М. Н. Узлы в социальных сетях: меры центральности и роль в сетевых процессах / М. Н. Юдина // Омский научный вестник. Сер. Приборы, машины и технологии. – 2016. – № 4 (148). – С. 161–165.

32. Юдина, М. Н. Исследование подсети обучавшихся (обучающихся) в ОмГТУ лиц в социальной сети «ВКонтакте» / М. Н. Юдина, А. С. Бондаренко // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 26–28 апр. 2016 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск : Изд-во ОмГТУ, 2016. – С. 173–178.

33. Юдина, М. Н. Пакеты анализа сетей на базе hadoop / М. Н. Юдина, А. Р. Дёмина // Информационные технологии и автоматизация управления : материалы VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности (Омск, 18–19 мая 2017 г.) / Минобрнауки России ; ОмГТУ, каф. АСОИУ. – Омск : Изд-во ОмГТУ, 2017. С. 144–148.

34. Юдина, М. Н. Четыре задачи ускорения вычислений за счет разбиения на подмножества элементов / М. Н. Юдина, В. Н. Задорожный // Динамика систем, механизмов и машин. – 2014. – № 3. – С. 250–253.
35. Bayesian learning in social networks / D. Acemoglu [et al.] // Review of Economic Studies. – 2011. – Vol. 78, no. 4. – P. 1201–1236.
36. Albert, I. Conserved network motifs allow protein–protein interaction prediction / I. Albert, R. Albert // Bioinformatics. – 2004. – Vol. 20, no. 18. – P. 3346–3352.
37. Albert, R. Error and attack tolerance of complex networks / R. Albert, H. Jeong, A. L. Barabasi // Nature. – 2000. – Vol. 406. – P. 378–482.
38. Albert, R. Statistical mechanics of complex networks / R. Albert, A. Barabasi // Reviews of Modern Physics. – 2002. – Vol. 74. – P. 42–97.
39. Alon, U. Network Motif Software / U. Alon. – URL: <https://www.weizmann.ac.il/mcb/UriAlon/download/network-motif-software> (дата обращения: 10.09.2019).
40. Classes of behavior of small-world networks / L. A. Amaral [et. al.] // Proceedings of the National Academy of Sciences of the United States of America. PNAS. – 2000. – Vol. 97. – P. 11149–11152.
41. Ballester, C. Who's who in networks: wanted the key player / C. Ballester, A. Calvo-Armengol, Y. Zenou // Econometrica. – 2006. – Vol. 74, no. 5. – P. 1403–1417.
42. The Diffusion of Microfinance / A. Banerjee [et. al.]. – DOI:10.1126/science.1236498.9:52 // Science. – 2013. – Vol. 6144, no. 341. – P. 1236498.
43. Barabasi, A. Emergence of scaling in random networks / A. Barabasi, R. Albert // Science. – 1999. – Vol. 286. – P. 509–512.
44. Barabasi, A. Scale-Free Networks / A. Barabasi, E. Bonabeau // Scientific American. – 2003. – Vol. 288. – P. 60–69.
45. Barabasi, A. Scale-Free Networks: A Decade and Beyond / A. Barabasi // Science. – 2009. – Vol. 325. – P. 412–413.
46. Barabasi, A. The network takeover / A. Barabasi // Nature Physics. – 2012. – Vol. 8. – P. 14–16.

47. Bhuiyan, M. A. Guise: Uniform sampling of graphlets for large graph analysis / M. A. Bhuiyan, M. Rahman, M. Al. Hasan // Proceedings of the 12th International Conference on Data Mining (ICDM). – Brussels, Belgium, 2012. – P. 91–100.
48. Bhuiyan, M. GUISE: Uniform Sampling of Graphlets for Large Graph Analysis / Mansurul Bhuiyan, Mahmuda Rahman, Mohammad Al Hasan. – URL: <https://github.com/DMGroup-IUPUI/GUISE-Source> (дата обращения: 10.09.2019).
49. Models of social networks based on social distance attachment / M. Boguna [et al.] // Physical Review E. – 2004. – Vol. 70. – P. 056122.
50. A network based analysis of systemic inflammation in humans / S. E. Calvano [et al.] // Nature. – 2005. – Vol. 7061, no. 437 – P. 1032–1037.
51. Castillo, C. Effective web crawling: Ph.D. Thesis in Computer Science / C. Castillo. – Chile : University of Chile, 2004. – 180 p.
52. Cooper, C. A general Model of Web graphs / C. Cooper, A. Frieze // Random Structures and Algorithms. – 2002. – Vol. 22 – P. 132–139.
53. Cooper, R. Coordination Games / R. Cooper. – Cambridge : Cambridge University Press, 1998. – 180 p.
54. Comprehensive Molecular Portraits of Invasive Lobular Breast Cancer / G. Ciriello [et al.] // Cell. – Vol. 163, no. 2. – P. 506–519.
55. Chakrabarti, D. Epidemic Thresholds in Real Networks / D. Chakrabarti // ACM Transactions on Information and System Security. – 2008. – Vol. 10. – P. 1–26.
56. Chang, F. The valume of Gaint component of random graph with given expected degrees / F. Chang, L. Lu // Discrete Mathematics. – 2006. – Vol. 20. – P. 395–411.
57. The BioGRID interaction database: 2017 update / A. Chatr-Aryamontri [et al.] // Nucleic Acids Research. – 2017. – Vol. 45. – P. 369–379.
58. NeMoFinder: dissecting genome-wide protein-protein interactions with meso-scale network motifs / J. Chen [et al.] // Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. – Philadelphia, USA, 2006. – P. 106–115.

59. Identification of breast cancer patients based on human signaling network motifs / L. Chen [et al.] // Scientific Reports. – 2004. – Vol. 3368, no. 3. – P. 1–7.
60. Breakdown of the Internet under intentional attack / R. Cohen [et. al.] // Physical Review Letters. – 2001. – Vol. 86. – P. 3682–3685.
61. DeGroot, M. H. Reaching a Consensus / M. H. DeGroot // Journal of the American Statistical Association. – 1974. – Vol. 69. – P. 118–121.
62. Dereich, S. Random networks with sublinear preferential attachment: the giant component / S. Dereich, P. Morters // The Annals of Probability. – 2013. – Vol. 41. – P. 329–384.
63. Dezsó, Z. Halting viruses in scale-free / Z. Dezsó, A. Barabási // Physical Review. E. – 2002. – Vol. 65. – P. 055103.
64. NDEX, the Network Data Exchange / P. Dexter [et al.] // Cell systems. – 2015. – Vol. 4, no. 1. – P. 302–305.
65. Dey, A. K. What network motifs tell us about resilience and reliability of complex networks / Asim K. Dey, Yulia R. Gel, Vincent Poor // Proceedings of the National Academy of Science of the United States of America. PNAS. – 2019. – Vol. 116, no. 39. – P. 19368–19373.
66. Dorogovtsev, S. N. Structure of Growing Networks: Exact Solution of the Barabási-Albert's Model / S. N. Dorogovtsev, J. F. F. Mendes, A. N. Samukhin // Physical Review Letters. – 2000. – Vol. 85. – P. 4633–4636.
67. Elenberg, E. R. GraphLab PowerGraph implementation of 4-profile counting / E. R. Elenberg. – URL: <https://github.com/eelenberg/4-profiles> (дата обращения: 10.09.2019).
68. Protein function in the post-genomic era / D. Eisenberg [et al.] // Nature. – 2000. – Vol. 405. – P. 823.
69. Distributed estimation of graph 4-profiles / Ethan R. Elenberg, Karthikeyan Shanmugam, Michael Borokhovich, Alexandros G. Dimakis // Proceedings of the 25th International Conference on World WideWeb. – Montréal, Canada, 2016. – P. 483–493.
70. Erciyes, K. Complex networks: an algorithmic perspective / K. Erciyes. – Izmir, Turkey: CRC Press, 2014. – 360 p.

71. Ergun, G. Growing Random Networks with Fitness / G. Ergun, G. J. Rodgers // *Physica. A: Statistical Mechanics and its Application*. – 2002. – Vol. 303. – P. 261–272.
72. Fields, S. Interactive learning: Lessons from two hybrids over two decades / S. Fields // *Proteomics* 9. – 2009. – Vol. 23. – P. 5209–5222.
73. Foggia, P. A performance comparison of five algorithms for graph isomorphism / P. Foggia, C. Sansone, M. Vento // *Proceedings of the 3rd IAPR TC-15 Workshop on Graph-based Representations in Pattern Recognition*. – Italy, Ischia, 2001. – P. 188–199.
74. Network games / A. Galeotti [et al.] // *Review of Economic Studies*. – 2010. – Vol. 77. – P. 218–244.
75. Grochow, J. A. Network Motif Discovery Using Sub-graph Enumeration and Symmetry-Breaking / J. A. Grochow, M. Kellis // *Proceedings of the 11th annual international conference on Research in computational molecular biology*. – Oakland, USA, 2007. – P. 92–106.
76. Systematic Evaluation of Molecular Networks for Discovery of Disease Genes / J. K. Huang [et al.] // *Cell Systems*. – 2018. – Vol. 6, no. 4. – P. 484–495.
77. Coarse-graining and self-dissimilarity of complex networks / S. Itzkovitz [et al.] // *Physical Review. E*. – 2005. – Vol. 71. – P. 016127.
78. Jackson, M. O. *Social and Economic Networks* / M. O. Jackson. – Princeton: Institute Princeton University Press, 2009. – 520 p.
79. Jackson, M. O. Meeting Strangers and Friends of Friends: How Random are Socially Generated Networks / M. O. Jackson, B. W. Rogers // *American Economic Review*. – 2007. – Vol. 97, no. 3. – P. 890–915.
80. Jha, M. Path sampling: A fast and provable method for estimating 4-vertex subgraph counts / Madhav Jha, C. Seshadhri, Ali Pinar // *Proceedings of the 24th International Conference on World WideWeb*. – Florence Italy, 2015. – P. 495–505.
81. Kavosh: a new algorithm for finding network motifs / Z. R. Kashani [et al.] // *BMC Bioinformatics*. – 2009. – Vol. 10. – P. 318.

82. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs / N. Kashtan [et al.] // *Bioinformatics*. – 2004. – Vol. 20, no. 11. – P. 1746–1758.
83. Kelleher, N. L. A cell-based approach to the human proteome project / N. L. Kellehe // *Journal of The American Society for Mass Spectrometry*. – 2012. – Vol. 23, no. 10. – P.1617–1651.
84. Kesten, H. What is percolation? / H. Kesten // *Notices of the AMS*. – 2006. – Vol. 53, no. 5 – P. 572–573.
85. Kitano, H. Computational systems biology/ H. Kitano // *Nature*. – 2002. – Vol. 6912, no. 420. – P. 206–210.
86. Kolda, T. Triadic Measures on Graphs: The Power of Wedge Sampling / Tamara Kolda, Ali Pinar, C. Seshadhri. – URL: <http://www.sandia.gov/~tgkolda/feastpack/> (дата обращения: 10.09.2019).
87. Kreher, D. L. Combinatorial Algorithms: Generation, Enumeration, and Search / D. L. Kreher, D. R. Stinson. – London : CRC Press, 1998. – 344 p.
88. Saha, T. K. Finding Network Motifs Using MCMC Sampling // Tanay K. Saha, Mohammad Al Hasan // *Proceedings of the CompleNet*. – 2015. – Vol. 4. – P. 13–24.
89. Leskovec, J. Graph Evolution: Densification and Shrinking Diameters / J. Leskovec, J. Kleinberg, C. Faloutsos // *ACM Transactions on Knowledge Discovery from Data*. – 2007. – Vol. 1, no. 1. – P. 1.
90. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters / J. Leskovec [et al.] // *Internet Mathematics*. – 2009. – Vol. 6, no. 1. – P. 29–123.
91. Lima-Mendez, G. The powerful law of the power law and other myths in network biology / G. Lima-Mendez, J. van Helden // *Molecular BioSystems*. – 2009. – Vol. 5, no. 12. – P. 1482–1493.
92. Meira, L. A. A. Acc-Motif: Accelerated Network Motif Detection / L. A. A. Meira [et al.] // *IEEE/ACM Transactions on computational biology and bioinformatics*. – 2014. – Vol. 11, no. 5. – P. 853–862.

93. Formation of regulatory patterns during signal propagation in a Mammalian cellular network/ A. Ma'ayan [et al.] // Science. – 2005. – Vol. 310. – P. 1078–1161.
94. McAuley, J. J. Learning to Discover Social Circles in Ego Networks / J. J. McAuley, J. Leskovec // Proceedings of the Neural Information Processing Systems Conference. – Lake Tahoe, USA, 2012. – P. 1–9.
95. McKay, B. D. Practical graph isomorphism / B. D. McKay // Congressus Numerantium. – 1981. – Vol. 30. – P. 45–87.
96. Network motifs: Simple building blocks of complex networks / R. Milo [et al.] // Science. – 2002. – Vol. 594, no. 298. – P. 824–827.
97. Miyazaki, T. The complexity of McKay's canonical labeling algorithm, in Groups and Computation, II / T. Miyazaki // DIMACS: Series in Discrete Mathematics and Theoretical Computer Science. – 1995. – Vol. 28 – P. 239–256.
98. Newman, M. E. J. The Structure and Function of Complex Networks / M. E. J. Newman // SIAM Review. – 2003. – Vol. 45, no. 2, – P. 167–256.
99. Newman, M. Networks: An Introduction / M. E. J. Newman. – Oxford : University Press, 2010. – 772 p.
100. Omid, S. MODA: an efficient algorithm for network motif discovery in biological networks / S. Omid, F. Schreiber, A. Masoudi-Nejad // Genes & Genetic Systems. – 2009. – Vol. 84, no. 5. – P. 385–395.
101. Paredes, P. Rand-FaSE: fast approximate subgraph census / Pedro Paredes, Pedro Ribeiro // Social Network Analysis and Mining. – 2015. – Vol. 5, no. 1. – P. 17.
102. Paredes, P. FaSE – Fast Subgraph Enumeration // Pedro Paredes, Pedro Ribeiro. – URL: <https://github.com/ComplexNetworks-DCC-FCUP/fase> (дата обращения: 10.09.2019).
103. Pastor-Satorras, R. Epidemic Spreading in Scale-Free Networks / R. Pastor-Satorras, A. Vespignani // Physical Review Letters. – 2001. – Vol. 86. – P. 3200–3203.
104. Winners don't take all: Characterizing the competition for links on the web / D. M. Pennock [et al.] // Proceedings of the National Academy of Science of the United States of America. PNAS. – 2002. – Vol. 99, no. 8. – P. 5207–5211.

105. Assessing the Exceptionality of Network Motifs / F. Picard [et al.] // Journal of Computational Biology. – 2005. – Vol. 15, no. 1. – P. 1–20.
106. Piperno, A. Search Space Contraction in Canonical Labeling of Graphs / A. Piperno. – 2011. – URL: <http://arxiv.org/abs/0804.4881> (дата обращения: 10.09.2019).
107. Pržulj, N. Efficient estimation of graphlet frequency distributions in protein–protein interaction networks / N. Pržulj, D. G. Corneil, I. Jurisica // Bioinformatics. – 2006. – Vol. 22, no. 8. – P. 974–980.
108. Rahman, M. Graft: An efficient graphlet counting method for large graph analysis / Mahmudur Rahman, Mansurul Alam Bhuiyan, Mohammad Al Hasan // IEEE Transactions on Knowledge and Data Engineering. – 2014. – Vol. 26, no. 10. – P. 2466–2478.
109. Rahman, M. GRAFT: an approximate graphlet counting algorithm for large graph analysis / Mahmudur Rahman, Mansurul Bhuiyan, Mahmuda Rahman. – URL: <https://github.com/DMGroup-IUPUI/GRAFT-Source> (дата обращения: 10.09.2019).
110. Ribeiro, P. G-Tries: an efficient data structure for discovering network motifs / P. Ribeiro, F. Silva // ACM 25th Symposium on Applied Computing – Bioinformatics Track. – Sierre, Switzerland, 2010. – P. 1559–1566.
111. A Survey on Subgraph Counting: Concepts, Algorithms and Applications to Network Motifs and Graphlets / P. Ribeiro, P. Paredes, M. E. P. Silva [et al.]. – URL : <https://arxiv.org/pdf/1910.13011.pdf> (дата обращения: 22.06.2020).
112. A proteome-scale map of the human interactome network / T. Rolland [et al.] // Cell. – 2014. – Vol. 159, no. 5. – P. 1212–1226.
113. Seshadhri, C. Triadic measures on graphs: The power of wedge sampling / Comandur Seshadhri, Ali Pinar, Tamara G. Kolda // Proceedings of the SIAM International Conference on Data Mining. – SIAM, 2013. – P. 10–18.
114. Schreiber, F. Frequency concepts and pattern detection for the analysis of motifs in networks / F. Schreiber, H. Schwobbermeyer // Transactions on Computational Systems Biology III. – 2005. – Vol. 3737. – P. 89–104.
115. Topological structure analysis of the protein-protein interaction network in budding yeast / S. Sun [et al.] // Nucleic Acids Research. – 2003. – Vol. 31, no. 9. – P. 2443–2450.

116. Network motifs in the transcriptional regulation network of *Escherichia coli* / S. S. Shen-Orr [et al.] // *Nature Genetics*. – 2002. – Vol. 31. – P. 64–68.
117. Simon, H. A. On a class of skew distribution functions / H. A. Simon // *Biometrika*. – 1955. – Vol. 42. – P. 425–440.
118. MotifNet: A web-server for network motif analysis / I. Y. Smoly [et al.] // *Bioinformatics*. – 2017. – Vol. 33, no. 12. – P. 1907–1909.
119. Le, S. A high-performance pipeline for genome-wide network reconstruction from gene expression data / S. Le, A. Riva, D. Tran // *Proceedings of NetBio SIG, ISMB*. – Orlando, FL, USA, 2016. – URL: <https://doi.org/10.7490/fl000research.1115059.1> (дата обращения: 10.09.2019).
120. Vespignani, A. Evolution thinks modular / A. Vespignani // *Nature Genetics*. – 2003. – Vol. 35, no. 2. – P. 118–119.
121. Symmetry compression method for discovering network motifs / Jianxin Wang, Yuannan Huang, Fang-Xiang Wu, Yi Pan // *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*. – 2012. – Vol. 9, no. 6. – P. 1776–1789.
122. Efficiently estimating motif statistics of large networks / Pinghui Wang, John Lui, Bruno Ribeiro [et al.] // *ACM Transactions on Knowledge Discovery from Data (TKDD)*. – 2014. – Vol. 9, no. 2 – P. 8.
123. MOSS-5: A fast method of approximating counts of 5-node graphlets in large graphs / Pinghui Wang, Junzhou Zhao, Xiangliang Zhang [et al.] // *IEEE Transactions on Knowledge and Data Engineering*. – 2018. – Vol. 30, no. 1. – P. 73–86.
124. Wang P. MOSS-5: Fast Method of Approximating Counts of 5-Node Graphlets in Large Graphs. – URL: <http://nskeylab.xjtu.edu.cn/dataset/phwang/code/mosscode.zip> (дата обращения: 10.09.2019).
125. Wernicke, S. A faster algorithm for detecting network motifs / S. Wernicke // *Lecture Notes in Bioinformatics*. – 2005. – Vol. 3692. – P. 165–177.
126. Wernicke, S. FANMOD: a tool for fast network motif detection / S. Wernicke, F. Rasche // *Bioinformatics*. – 2006. – Vol. 22, no. 9 – P. 1152–1153.
127. Wernicke, S. FANMOD: a tool for fast network motif detection / S. Wernicke. – URL: <http://theinfl.informatik.uni-jena.de/motifs/> (дата обращения: 10.09.2019).

128. A Survey of Measures for Network Motifs / Feng Xia, Haoran Wei, Shuo Yu [et al.] // IEEE Access. – 2019. – Vol. 7. – P. 106576–106587.
129. SSRW: A Scalable Algorithm for Estimating Graphlet Statistics Based on Random Walk / Chen Yang, Min Lyu, Yongkun Li [et al.] // Proceedings of the International Conference on Database Systems for Advanced Applications. – Springer, 2018. – P. 272–288.
130. Yudin, E. B. Statistical Approach to Calculation of Number of Network Motifs / E. B. Yudin, V. N. Zadorozhnyi // International Siberian conference on control and communications, SIBCON : proceedings. – Omsk, 2015. – P. 1–4.
131. Yudina, M. N. Assessment of accuracy in calculations of network motif concentration by Rand ESU algorithm / M. N. Yudina // Journal of Physics: Conference Series. – 2019. – Vol. 1260. – P. 022012.
132. Yudina, M. N. Mixed Random Sampling of Frames method for counting number of motifs / M. N. Yudina, V. N. Zadorozhnyi, E. B. Yudin // Journal of Physics: Conference Series. – 2019. – Vol. 1260. – P. 022013.
133. Zadorozhnyi, V. N. Analytical and numerical methods of calibration for preferential attachment random graphs / V. N. Zadorozhnyi, E. B. Yudin, M. N. Yudina // International Siberian Conference on Control and Communications. – Kazakhstan, Astana, 2017. – P. 1–6.
134. Zadorozhnyi, V. N. Graphs with complex stochastic increments / V. N. Zadorozhnyi, E. B. Yudin, M. N. Yudina // Dynamics of Systems, Mechanisms and Machines : proceedings. – Omsk, 2017. – P. 1–7.
135. Zadorozhnyi, V. N. Distributions of degrees in growing graphs with loss of arcs / V. N. Zadorozhnyi, E. B. Yudin, M. N. Yudina // Moscow Workshop on Electronic and Networking Technologies. – Moscow, 2018. – P. 1–6.
136. Zeng, T. Spatio-temporal motifs for optimized vehicle-to-vehicle (v2v) communications / T. Zeng, O. Semiari, W. Saad // International Conference on Computing, Networking and Communications (ICNC). – Hawaii USA, 2018. – P. 789–794.

ПРИЛОЖЕНИЕ А

РЕГИСТРАЦИЯ ПРОГРАММ И ПРОГРАММНЫХ БИБЛИОТЕК

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020615195

Программа MFSView для выявления сетевых мотивов

Правообладатель: *Федеральное государственное бюджетное образовательное учреждение высшего образования «Омский государственный технический университет» (RU)*

Автор: *Юдина Мария Николаевна (RU)*



Заявка № 2020614256

Дата поступления 15 мая 2020 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 18 мая 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018614078

Система агентного моделирования Simbigraph 2

Правообладатели: *Юдин Евгений Борисович (RU), Задорожный Владимир Николаевич (RU), Юдина Мария Николаевна (RU)*

Авторы: *Юдин Евгений Борисович (RU), Задорожный Владимир Николаевич (RU), Юдина Мария Николаевна (RU)*



Заявка № **2018611393**

Дата поступления **14 февраля 2018 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **28 марта 2018 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное научное учреждение
ИНСТИТУТ УПРАВЛЕНИЯ ОБРАЗОВАНИЕМ РОССИЙСКОЙ АКАДЕМИИ
ОБРАЗОВАНИЯ

ОБЪЕДИНЕННЫЙ ФОНД ЭЛЕКТРОННЫХ РЕСУРСОВ "НАУКА И ОБРАЗОВАНИЕ"
(основан в 1991 году)

**СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ
ЭЛЕКТРОННОГО РЕСУРСА**

№ 23487

Настоящее свидетельство выдано на электронный ресурс, отвечающий
требованиям новизны и приоритетности:

Библиотека классов MotifsLib

Дата регистрации: 19 февраля 2018 года

Автор: Юдина М.Н.

Организация-разработчик: ФГБОУ ВО Омский государственный
технический университет

Директор ФЦИЕР ИУО РАО,
доктор экономических наук С.С. Неустроев

Руководитель ОФОРНАО, заместитель
руководителя науки и техники России А.Н. Галкина

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное научное учреждение
ИНСТИТУТ УПРАВЛЕНИЯ ОБРАЗОВАНИЕМ РОССИЙСКОЙ АКАДЕМИИ
ОБРАЗОВАНИЯ
ОБЪЕДИНЕННЫЙ ФОНД ЭЛЕКТРОННЫХ РЕСУРСОВ "НАУКА И ОБРАЗОВАНИЕ"
(основан в 1991 году)

**СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ
ЭЛЕКТРОННОГО РЕСУРСА**

№ 23770

ИУО РАО
ОФЭРНиО

Настоящее свидетельство выдано на электронный ресурс, отвечающий
требованиям новизны и приоритетности:

Библиотека классов SocAndBioNetworksAnalysis

Дата регистрации: 15 сентября 2018 года
Автор: Юдина М.Н.
Организация-разработчик: ФГБОУ ВО «Омский государственный
технический университет»

Директор ФГБНУ ИУО РАО,
доктор экономических наук  С.С. Неустроев

Руководитель ОФЭРНиО, почетный
работник науки и техники России  А.И. Галкина



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное научное учреждение
ИНСТИТУТ УПРАВЛЕНИЯ ОБРАЗОВАНИЕМ РОССИЙСКОЙ АКАДЕМИИ
ОБРАЗОВАНИЯ
ОБЪЕДИНЕННЫЙ ФОНД ЭЛЕКТРОННЫХ РЕСУРСОВ "НАУКА И ОБРАЗОВАНИЕ"
(основан в 1991 году)

**СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ
ЭЛЕКТРОННОГО РЕСУРСА**

ИУО РАО
ОФЭР№0

№ 23769

Настоящее свидетельство выдано на электронный ресурс, отвечающий
требованиям новизны и приоритетности:

Библиотека классов YMN_GraphGenerators

Дата регистрации: 15 сентября 2018 года

Автор: Юдина М.Н.

Организация-разработчик: ФГБОУ ВО «Омский государственный
технический университет»



Директор ФГБНУ ИУО РАО,
доктор экономических наук  С.С. Неустроев

Руководитель ОФЭР№0, почетный
работник науки и техники России  А.Н. Галанова

ПРИЛОЖЕНИЕ Б

АКТЫ ВНЕДРЕНИЯ

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Омский государственный технический университет»

УТВЕРЖДАЮ

Проректор по научной работе
Омского государственного
технического университета

Б.Д. Женазов

«___» _____ 2019 г.

АКТ

о внедрении результатов диссертационной работы «Методы и алгоритмы ускоренного
 расчета частот встречаемости сетевых мотивов в больших случайных графах»
 Юдиной Марии Николаевны для разработки эффективных стратегий информационного
 освещения деятельности ОмГТУ в социальной сети «ВКонтакте»

Комиссия в составе

председателя комиссии

Соломина Вячеслава Юрьевича, начальника Отдела
корпоративных средств массовой информации ОмГТУ,

и членов комиссии

Фефелова Василия Федоровича, начальника НИЧ ОмГТУ,
 Никифоровой Ангелины Юрьевны, начальника отдела
 организации НИР студентов и молодых ученых ОмГТУ,

составили настоящий акт о следующем.

Результаты диссертационной работы М.Н. Юдиной приняты к использованию для
 разработки эффективных стратегий информационного освещения деятельности ОмГТУ в
 социальной сети «ВКонтакте». В частности, в научных публикациях

1) Юдина М.Н. Узлы в социальных сетях: меры центральности и роль в сетевых процессах // Омский
 научный вестник. – 2016. – №4 (148). С. 161-165

2) Юдина М.Н., Бондаренко А.С. Исследование подсети обучающихся (обучающихся) в ОмГТУ лиц в
 социальной сети "ВКонтакте"// Информационные технологии и автоматизация управления: сборник
 материалов VIII Всероссийской научно-практической конференции студентов, аспирантов, работников
 образования и промышленности. – 2016. – С. 173-178

Юдиной М.Н. проанализирована структура подсети «ВКонтакте», узлами которой являются
 владельцы аккаунтов, указавшие в качестве места учебы ОмГТУ. Для представленной подсети
 выявлены наиболее значимые узлы (по критериям активности, влияния, удаленности, особой
 роли в развитии сетевых процессов). Информация об этих узлах позволяет повысить
 эффективность обсуждения значимых вопросов и учета мнения целевых групп, вовлеченных в
 корпоративную сеть ОмГТУ.

Полученные Юдиной М.Н. результаты дополнительно подтверждены при выполнении
 – НИР № 17306В «Разработка методов, алгоритмов и программных средств для структурной идентификации
 больших сетей (приказ № 895/1 от 01.11.2017 г.) (исполнитель Юдина М.Н.)

– Проекта № АААА-А18-118091490011-1 по региональному конкурсу «Молодой ученый»; тема проекта
 «Расчет частот встречаемости типовых подграфов в больших сетях»; сроки выполнения 04.05.2018 по
 30.11.2018 (руководитель Юдина М.Н.).

Председатель комиссии

начальник Отдела корпоративных
 средств массовой информации

В. Ю. Соломин

Члены комиссии:

начальник НИЧ ОмГТУ

В. Ф. Фефелов

начальник отдела организации НИР
 студентов и молодых ученых ОмГТУ

А. Ю. Никифорова

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ
ИНСТИТУТ ПРОБЛЕМ ПЕРЕРАБОТКИ УГЛЕВОДОРОДОВ
СИБИРСКОГО ОТДЕЛЕНИЯ РОССИЙСКОЙ АКАДЕМИИ НАУК
(ИППУ СО РАН)**



Утверждаю
Директор Института,
доктор химических наук
Лавренов А.В.
_____ 201__

Акт

О внедрении результатов диссертационной работы Юдиной М. Н.
«Методы и алгоритмы ускоренного расчета частот встречаемости сетевых мотивов в
больших случайных графах»

Комиссия в составе: председатель комиссии с.н.с., к.х.н. Мироненко Р.М., члены
комиссии с.н.с., к.х.н. Бельская О.Б., н.с., к.х.н. Леонтьева Н.Н.,

рассмотрев представленные материалы:

- 1) диссертационную работу Юдиной М.Н.,
 - 2) научные работы Юдиной М.Н. (публикации в журналах перечня ВАК, свидетельства о регистрации программ),
 - 3) разработанный Юдиной М.Н. комплекс программ для исследования молекулярных сетей клетки и систему агентного моделирования Simbigraph 2, разработанную Юдиной М.Н. в соавторстве,
 - 4) результаты исследования генетических сетей, сетей белковых взаимодействий, в том числе сеть Pathway Commons, описывающую пути биохимических реакций (включая процессы транспорта и катализа, а также физические взаимодействия с участием ДНК, РНК, различных белковых комплексов)
- установила, что:
- 1) предложенные в диссертации методы и алгоритмы исследования больших сетей позволяют эффективно исследовать большие сети, в том числе визуализировать их в виде графов, исследовать локальные и глобальные структурные характеристики этих графов, а также эффективно рассчитывать частоты встречаемости сетевых мотивов,
 - 2) основные положения диссертационной работы доведены до практической реализации в виде комплекса программ, эти программы размещены в открытом доступе,
 - 3) результаты диссертационной работы целесообразно принять к использованию при выполнении ИППУ СО РАН фундаментальных научных исследований и в разработках новых технологий,
 - 4) разработанные Юдиной М.Н. библиотеки классов SocAndBioNetworksAnalysis и MotifsLib, предназначенные для исследования больших сетей, целесообразно использовать в ИППУ СО РАН для визуализации сетей химических реакций в виде графа, с целью дальнейшего анализа методами Теории сетей (Р.-М. Jacob and A. Lapkin Statistics of the network of organic chemistry // Reaction Chemistry & Engineering, 2018), программу Simbigraph 2 целесообразно использовать в ИППУ СО РАН для физико-химического моделирования нано-структур, а также 3D визуализации физико-химических моделей.

Председатель комиссии
Кандидат химических наук, с.н.с.

Мироненко Р.М.

Члены комиссии:
Кандидат химических наук, с.н.с.
Кандидат химических наук, н.с.

Бельская О.Б.
Леонтьева Н.Н.

Автономная некоммерческая профессиональная образовательная организация
«Многопрофильная Академия непрерывного образования» (АН ПОО «МАНО»)

УТВЕРЖДАЮ



Ректор АН ПОО «МАНО», профессор,
доктор педагогических наук,
Заслуженный учитель РФ

В.И. Гам

2019 г.

АКТ

о внедрении результатов диссертационной работы «Методы и алгоритмы ускоренного расчета частот встречаемости сетевых мотивов в больших случайных графах»

Комиссия в составе

председателя комиссии Гама Антона Владимировича, декана факультета профессиональной переподготовки и повышения квалификации

и членов комиссии Бугаева Анатолия Петровича, начальника отдела информационно-технического обеспечения, Кичук Евгении Андреевны, методиста факультета профессиональной переподготовки и повышения квалификации

составили настоящий акт о следующем.

Методы исследования социальных сетей, а также методы анализа сетей ссылок веб-страниц, предлагаемые в публикациях Юдиной М.Н.

1) Юдина М.Н. Узлы в социальных сетях: меры центральности и роль в сетевых процессах // Омский научный вестник. – 2016. – №4 (148). С. 161-165

2) Пахомов И. А., Юдина М. Н. Сбор и анализ информации о структуре сети ссылок веб-страниц // Информационные технологии и автоматизация управления: сб. матер. VIII Всерос. науч.-практ. конф. студентов, аспирантов, работников образования и промышленности. – 2017. – С. 117-122

рассмотрены на производственном совещании с участием ведущих специалистов АН ПОО «МАНО». Приняты к использованию реализующие эти методы комплексы программ: SocAndBioNetworkAnalysis (свидетельство о регистрации электронного ресурса № 01712, дата регистрации 15.09.2018), библиотека классов MotifsLib (свидетельство о регистрации электронного ресурса № 23487, дата регистрации 19.02.2018), библиотека классов YMN_GraphGenerators (свидетельство о регистрации электронного ресурса № 01713, дата регистрации 15.09.2018).

По результатам обсуждения принято решение о применении разработанных Юдиной М. Н. методов, алгоритмов и программ для повышения информативности официального сайта АН ПОО «МАНО» и достижения сайтом посещаемости на уровне ведущих информационных интернет-ресурсов региона в сфере образования.

Председатель комиссии

декан факультета профессиональной переподготовки и повышения квалификации

А.В. Гам

Члены комиссии:

начальник отдела информационно-технического обеспечения

А.П. Бугаев

методист факультета ШиПК

Е.А. Кичук

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Омский государственный технический университет»

УТВЕРЖДАЮ

Проректор по учебной работе
Омского государственного
технического университета, д.т.н.



О. В. Кропотин

2019 г.

АКТ

о внедрении результатов диссертационной работы «Методы и алгоритмы ускоренного
расчета частот встречаемости сетевых мотивов в больших случайных графах»
Юдиной Марии Николаевны в учебный процесс ОмГТУ

Комиссия в составе:

председатель комиссии – Макаров В.В., к.т.н., декан факультета информационных технологий и компьютерных систем;

члены комиссии – Бояркин Г.Н., д.э.н., зав. кафедрой «Математические методы и информационные технологии в экономике»;
– Стариков В.И., к.т.н., доцент кафедры «Математические методы и информационные технологии в экономике»

составили настоящий акт о следующем:

Результаты диссертационной работы М.Н. Юдиной используются при изучении дисциплины «Имитационное моделирование» студентами факультета информационных технологий и компьютерных систем направлений подготовки 09.03.03 «Прикладная информатика». А комплексы библиотек классов SocAndBioNetworkAnalysis, MotifsLib, YMN_GraphGenerators применяются при проведении лабораторных работ по указанной дисциплине для исследования больших сетей и сетевых процессов.

Председатель комиссии

к.т.н., декан факультета информационных технологий и компьютерных систем

В.В. Макаров

Члены комиссии:

д.э.н., зав. кафедрой «Математические методы и информационные технологии в экономике»

Г.Н. Бояркин

к.т.н., доцент кафедры «Математические методы и информационные технологии в экономике»

В.И. Стариков