

Федеральное государственное бюджетное образовательное учреждение
высшего образования «Казанский национальный исследовательский
технологический университет»

На правах рукописи



АНТОНОВА ПОЛИНА ВАЛЕРЬЕВНА

**Методы моделирования систем массового обслуживания
с ограниченным временем пребывания заявки в очереди и
временем обслуживания, распределенным по закону Эрланга**

1.2.2. Математическое моделирование,
численные методы и комплексы программ

ДИССЕРТАЦИЯ
на соискание учёной степени
кандидата технических наук

Научный руководитель:
доктор технических наук, доцент
Титовцев Антон Сергеевич

Казань – 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1 АНАЛИТИЧЕСКИЙ ОБЗОР И ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ..	15
1.1 Обзор исследований, посвященных СМО с эрланговским распределением времени обслуживания	15
1.2 Федеральная государственная информационная система территориального планирования как система массового обслуживания.....	23
1.3 Постановка задачи.....	26
2 ХАРАКТЕРИСТИКИ СМО $M/Er/1$ С ОГРАНИЧЕННЫМ ВРЕМЕНЕМ ПРЕБЫВАНИЯ ЗАЯВКИ В ОЧЕРЕДИ	31
2.1 Стационарные характеристики одноканальной СМО.....	31
2.2 Нестационарные характеристики одноканальной СМО.....	46
2.3 Имитационная модель СМО	58
2.4 Разработка метамоделей системы массового обслуживания с использованием ансамблевых методов машинного обучения	64
3 ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ СТАБИЛЬНЫХ РЕЖИМОВ РАБОТЫ СМО	83
4 АПРОБАЦИЯ РЕЗУЛЬТАТОВ В ВИДЕ ПРОГРАММНОГО КОМПЛЕКСА С ПРИМЕНЕНИЕМ ПАРАЛЛЕЛЬНОЙ БИБЛИОТЕКИ ПЛАТФОРМЫ .NET	99
4.1 Постановка задачи.....	99
4.2 Принципы разработки многоканальной СМО с ограниченной очередью с применением parallel framework	101
4.3 Применение конкурентных коллекций.....	105
4.4 Результаты тестирования	107
ЗАКЛЮЧЕНИЕ	110
СПИСОК ЛИТЕРАТУРЫ.....	112
ПРИЛОЖЕНИЯ	122

ВВЕДЕНИЕ

Актуальность темы исследования. Задачи исследования систем массового обслуживания (СМО) с ограничениями на управляющие параметры приобретают особую актуальность. Задачи такого рода играют важную роль в различных предметных областях, таких как теория телетрафика, логистика, теория коммуникаций и др. Согласно указу Президента Российской Федерации № 145 «О стратегии научно-технологического развития Российской Федерации», в соответствии с п. 21 а, е, и п. 22 стратегии, развитие современных интеллектуальных телекоммуникационных систем относится к одному из приоритетных направлений научно-технологического развития страны.

В условиях наступившей четвертой промышленной революции, предполагающей новый подход к производству, основанный на массовом внедрении информационных технологий в промышленность, масштабной автоматизации бизнес-процессов и распространении искусственного интеллекта, важнейшими становятся задачи, связанные с повышением показателей эффективности различного рода технологических процессов и технических систем. Исследования в направлениях теории массового обслуживания с ограничениями на управляющие параметры не только улучшают существующие технологии и процессы, но и открывают новые возможности для будущих исследований, способствуя развитию как научных, так и практических дисциплин.

В простейших случаях объекты, функционирующие по принципу СМО, описываются наиболее известными классическими моделями. Однако для более глубокого изучения различных систем требуются сложные модели, учитывающие их различные технические особенности. К таким моделям можно отнести СМО, которые имеют те или иные ограничения на параметры системы.

Среди задач с ограничениями на управляющие параметры особое внимание уделяется тем, в которых ограничения касаются времени пребывания заявки в системе. Иными словами, некоторые заявки в таких системах являются «нетерпеливыми». Для подобных систем массового обслуживания можно выделить два ос-

новых подхода к постановке задачи. В первом случае заявка может покинуть систему как из очереди, так и с обслуживания. Во втором случае «нетерпеливая» заявка покидает систему, находясь в очереди. Модели такого типа остаются наименее изученными среди всех типов систем массового обслуживания. Исследование этих моделей имеет большое значение при оптимизации процессов в подобных системах, что открывает новые возможности для улучшения их эффективности.

Одна из основных задач изучения подобных систем заключается в следующем. Для расчетов в простейших моделях используются суммы либо бесконечного, либо конечного числа слагаемых.

Для того, чтобы найти решения для таких задач целесообразно использование численных методов. В этом случае все численные показатели системы массового обслуживания определяются по отдельности путем суммирования нескольких первых членов соответствующего ряда, который может быть как конечный, так и бесконечный. Однако данный подход не позволяет получить замкнутое аналитическое решение. Несмотря на это подход с использованием приближенных численных методов позволяет оценить основные характеристики системы, что может быть достаточным для ряда прикладных задач. Детальное же изучение процессов в таких системах, как правило, остается затруднительным.

В данной работе впервые представлена одноканальная система массового обслуживания смешанного типа с временем обслуживания, представляющим собой сумму независимых экспоненциально распределенных случайных величин, с накопителем неограниченной емкости и решается задача комплексного исследования режимов функционирования СМО с эрланговски распределенным временем обслуживания, что подразумевает под собой разработку моделей, методов и комплекса программ, направленных на определение характеристик системы, обеспечивающих требуемое качество обслуживания. Подобного рода системы могут быть рассмотрены при разработке систем сопровождения, администрирования, и эксплуатации программно-технических средств и информационных ресурсов, а также для мониторинга и оптимизации производственных процессов, а

значит, задача изучения СМО $M/E_r/1$ с ограничениями на время пребывания требования в очереди является актуальной.

Степень разработанности темы.

Теория массового обслуживания берет свои истоки в исследованиях, связанных с телефонными системами. Пионерами в этой области были такие ученые, как У. Феллер, М.Д. Кендалл и др. Феллер изучал вопросы, касающиеся концепции очередей. Кендалл ввел понятие вложенных цепей Маркова и разработал широко используемую классификацию СМО. Кларк первым нашел решение для уравнений процесса гибели и размножения с постоянными коэффициентами для переходного состояния.

Дальнейшие исследования были посвящены развитию методов решения задач теории массового обслуживания. С. Карлин и Мак-Грегор предложили метод решения с использованием коэффициентов, зависящих от числа требований в очереди. Т.Л. Саати разработал решение для переходного состояния марковской модели многоканальной СМО с использованием преобразования Лапласа. Эти ранние работы заложили основы в дальнейших исследованиях теории массового обслуживания (ТМО), которая нашла применение в различных предметных областях.

Вопросами, возникающими при изучении СМО в настоящее время в нашей стране, занимается ряд ученых: Т.И. Алиев, Ю.В. Гайдамака, А.Н. Дудин, В.Н. Задорожный, А.А. Назаров, А.Н. Моисеев, С.П. Моисеева, С.В. Пауль, А.В. Печинкин, В.А. Романенко, В.В. Рыков, К.Е. Самуйлов, С.П. Сущенко, В.Н. Тарасов, О.М. Тихоненко, М.П. Фархадов, М.А. Шнепс-Шнеппе и др. В большинстве случаев эти проблемы имеют частный характер и ограничиваются анализом первых моментов различных характеристик систем массового обслуживания.

В современных условиях в теории массового обслуживания отсутствие замкнутого аналитического решения является проблемой, которая рассматривается как прикладная область исследований. В области таких видов моделирования как имитационное и аналитическое существует немалое количество публикаций. Зна-

чительный вклад в развитие теории внесли такие ученые как А.А. Боровков, Б.В. Гнеденко, Д. Кендалл, Д. Кокс, Д. Литтл, Ф. Поллячек, В. Смит, Л. Такач, А.Я. Хинчин. Однако на практике аналитические расчеты до недавнего времени проводились с использованием формул, которые были предложены А. К. Эрлангом в начале прошлого века.

Классический метод Эрланга применим к системам с простейшим входящим потоком и показательным распределенным временем обслуживания. Однако если первое из названных допущений на основании ряда теоретических соображений можно признать реалистическим, то второе в большинстве приложений диктуется лишь соображениями «считаемости». Расчёт систем с эрланговски распределенным временем обслуживания и простейшим входным потоком (согласно классификации Кендалла $M/E_r/1$) практически возможен только после «фазовой» аппроксимации распределения обслуживания с экспоненциально распределённой задержкой в каждой.

В системах массового обслуживания, где время обслуживания представляет собой сумму нескольких независимых случайных величин, распределенных по одному и тому же экспоненциальному закону, число заявок в системе в данный момент времени уже не будет представлять собой марковский процесс. Однако для таких систем существует возможность построить марковский процесс с непрерывным временем, применяя метод фиктивных фаз. Таким образом несмотря на то, что число заявок не является марковским процессом напрямую, можно использовать специальные методы для моделирования такой системы как марковского процесса с непрерывным временем.

Для эрланговского распределения времени обслуживания предполагается, что каждая заявка, прибывшая в систему, должна пройти несколько фаз обслуживания, длительность которых распределена по экспоненциальному закону. Аналогичный подход используется и в случае, когда времена обслуживания можно описать допускающими аналогичную вероятностную трактовку распределениями фазового типа.

Системы $M/E_r/1$ находят широкое применение в задачах организации трафика телекоммуникационных сетей и в задачах промышленной инженерии.

Используя матрично-геометрический подход, М. Ньютс исследовал системы массового обслуживания с эрланговским распределением времени обслуживания. Следует отметить, что этот метод является чрезвычайно трудным для практической реализации и зачастую страдает от вычислительной нестабильности из-за плохой обусловленности используемых матриц большой размерности.

Первая попытка применения нового метода изучения СМО с ограничениями на время нахождения заявки в очереди или в системе в целом в случае, когда все заявки можно назвать “нетерпеливыми” была предпринята в работах А.П. Кирпичникова. В основе данного подхода лежало применение функции Г. Миттаг-Леффлера 1-го порядка.

С точки зрения практического применения, более перспективным представляется расширенный вариант задачи, в котором каждая заявка ожидает не дольше случайного времени, распределенного по экспоненциальному закону. До сих пор не проводились исследования СМО смешанного типа с эрланговски распределенным временем обслуживания.

Объектом исследования являются технические системы с очередями, функционирующие в информационных системах, финансовых учреждениях, службах преодоления чрезвычайных ситуаций, системах мониторинга качества и пр. Проблемы, возникающие в данных областях, структурируются вокруг очередей, поэтому формальное описание современных систем логично формулировать в терминах теории массового обслуживания.

Предметом исследования являются вопросы изучения одноканальных СМО смешанного типа с временем обслуживания, распределенным по закону Эрланга.

Соответствие исследования паспорту научной специальности. Область диссертационного исследования соответствует следующим пунктам паспорта научной специальности 1.2.2. Математическое моделирование, численные методы и комплексы программ (технические науки): п.6. Разработка систем компьютер-

ного и имитационного моделирования, алгоритмов и методов имитационного моделирования на основе анализа математических моделей (технические науки); п.8. Комплексные исследования научных и технических проблем с применением современной технологии математического моделирования и вычислительного эксперимента; п.9. Постановка и проведение численных экспериментов, статистический анализ их результатов, в том числе с применением современных компьютерных технологий.

Цель и задачи исследования.

Целью исследования является разработка комплекса инструментов для анализа характеристик СМО с ограничениями на время пребывания заявки в очереди, где время обслуживания подчиняется закону Эрланга.

Задачи исследования:

1. Разработать математическую модель СМО смешанного типа с ограниченным временем ожидания заявки в очереди и временем обслуживания, распределенным по закону Эрланга, получить формулы для стационарных и нестационарных характеристик модели.

2. Разработать метод исследования СМО с эрланговски распределенным временем обслуживания, позволяющий найти требуемое количество фаз, обеспечивающее заданный уровень качества обслуживания.

3. Разработать имитационную модель СМО смешанного типа, а также метамодель, аппроксимирующую ключевые характеристики СМО на основе методов машинного обучения, для оценки показателей эффективности системы по заданным входным параметрам.

4. Разработать комплекс программных средств для решения задач по расчету ключевых характеристик СМО и оценки её эффективности.

5. Провести апробацию полученных результатов.

Научная новизна представленных результатов характеризуется развитием методов математического моделирования СМО, качественных и приближенных методов исследования СМО, методов и алгоритмов имитационного моделирования, и заключается в следующем:

1. *Разработана новая математическая модель СМО смешанного типа, отличающаяся от известных моделей сочетанием свойств СМО с ограниченным временем ожидания заявки в очереди и СМО с временем обслуживания, распределенным по закону Эрланга. Строгими аналитическими методами получены общие формулы для характеристик модели в стационарном и нестационарном режимах, позволяющие задавать временные рамки нахождения заявок в очереди и количество фаз обслуживания (п. 8 паспорта специальности 1.2.2).*

2. *Предложен новый метод исследования СМО с эрланговски распределенным временем обслуживания, отличающийся возможностью нахождения требуемого количества фаз, обеспечивающего заданный уровень качества обслуживания, которое может быть оценено с помощью коэффициента вариации различных характеристик СМО. Выявлено, что граничные значения числа фаз обслуживания логарифмически связаны с изменением приведенной интенсивности входного потока заявок (п. 8 паспорта специальности 1.2.2).*

3. *Разработана имитационная модель СМО смешанного типа с использованием инструментальной среды AnyLogic PLE, а также метамодель, отличающаяся возможностью аппроксимации ключевых характеристик СМО на основе методов машинного обучения. Метамодель позволяет оперативно оценивать показатели эффективности системы по заданным входным параметрам без проведения ресурсоёмких имитационных экспериментов, что значительно ускоряет процесс анализа (п. 6 паспорта специальности 1.2.2).*

4. *Разработан комплекс специализированных программных средств, отличающийся возможностью расчета ключевых характеристик СМО смешанного типа и оценки её эффективности (п. 9 паспорта специальности 1.2.2).*

Теоретическая значимость диссертационной работы состоит в следующем:

– предложена новая разновидность моделей СМО с эрланговски распределенным временем обслуживания и ограниченным ожиданием в очереди, которые могут быть применены в различных областях;

- решена задача формализации стационарных и нестационарных характеристик СМО $M/E_r/1$ смешанного типа;
- разработан новый метод исследования СМО, основанный на вычислении предельных значений количества фаз обслуживания, при которых система поддерживает стабильный режим работы;
- впервые к исследованию СМО применены ансамблевые методы машинного обучения.

Практическая значимость диссертационной работы заключается в том, что полученные результаты могут найти применение в различных областях, таких как транспортные и телекоммуникационные системы, производство, логистика, сфера обслуживания и других, работающих по принципу СМО смешанного типа, что окажет влияние на эффективность функционирования данных объектов.

Методология и методы исследования включают применение теории вероятностей, теории функций комплексного переменного и теории случайных процессов и цепей Маркова. Для решения трансцендентных уравнений применялись соответствующие численные методы. Дополнительно была построена имитационная модель СМО смешанного типа с эрланговским распределением времени обслуживания в среде моделирования AnyLogic PLE. Для разработки метамоделей СМО применены ансамблевые методы машинного обучения. Разработка и реализация программного комплекса осуществлялись с использованием Wolfram Mathematica.

Положения, выносимые на защиту:

1. Математическая модель СМО смешанного типа, сочетающая свойства СМО с ограниченным временем ожидания заявки в очереди и СМО с временем обслуживания, распределенным по закону Эрланга.

2. Новый метод исследования СМО с эрланговски распределенным временем обслуживания, полученные численные результаты: основной из них определяет логарифмический характер поведения граничных значений числа фаз обслуживания, соответствующих стабильному режиму по длине очереди с изменением приведенной интенсивности входного потока требований.

3. Имитационная модель СМО смешанного типа, а также метамодель, аппроксимирующая ключевые характеристики СМО на основе методов машинного обучения.

4. Комплекс специализированных программных средств для решения задач по расчету ключевых характеристик СМО и оценки её эффективности.

Степень достоверности результатов диссертации обеспечивается корректной постановкой задач, строгим выполнением математических выкладок и проверкой полученных решений на соответствие известным частным случаям, описанным другими авторами.

Апробация. Разработанные методы и алгоритмы внедрены в деятельность следующих организаций, что подтверждается соответствующими справками:

1. ООО «Систематика Консалтинг», г. Москва при разработке системы технического сопровождения, администрирования, сервисной поддержки и эксплуатации программно-технических средств и информационных ресурсов Федеральной государственной информационной системы территориального планирования (ФГИС ТП).

2. ООО «НПО ЛАБС», г. Санкт-Петербург при разработке программно-аппаратного решения для мониторинга и оптимизации производственных процессов.

Полученные результаты исследований активно используются в учебном процессе ФГБОУ ВО «КНИТУ» для обучения студентов по направлениям подготовки 01.03.02 «Прикладная математика и информатика» и 02.03.03 «Математическое обеспечение и администрирование информационных систем»

Результаты диссертации докладывались на международной научной конференции «FarEastCon-2020» (г. Владивосток, 2020), IV международной школе-конференции «DCNAIR-2020» (г. Иннополис, 2020), 34-й международной научной конференции «Математические методы в технике и технологиях ММТТ-34» (г. Санкт-Петербург, 2021), 37-й международной научной конференции «Математические методы в технике и технологиях ММТТ-37» (г. Казань, 2024), международной научной конференции «Перспективные фундаментальные исследования и

научные методы» (г. Санкт-Петербург, 2024), международной научной конференции «Перспективные исследования в технических и естественных науках» (г. Санкт-Петербург, 2024).

Личный вклад автора. Все представленные результаты получены автором лично. В зарегистрированном программном обеспечении автору принадлежит ведущая роль в разработке основных модулей для расчетов необходимых характеристик.

Публикации. По теме диссертации опубликовано самостоятельно и в соавторстве 10 печатных работ: 5 работ в рецензируемых научных изданиях, рекомендованных ВАК РФ; 1 – в изданиях, индексируемых в Scopus, 4 – в прочих изданиях. Зарегистрировано 2 электронных ресурса.

Структура и объём работы. Диссертация состоит из введения, четырех глав, заключения, списка литературы и приложения. Диссертация напечатана в 1,5 межстрочных интервала, полный объём составляет 126 страниц, включая 46 рисунков. Библиографический список включает 95 литературных источников.

Структура диссертации.

В первой главе приводится обзор и анализ литературных источников, результатов научных исследований, ранее выполненных по теме диссертации, дается характеристика предметной области и приводится постановка задачи.

Вторая глава содержит аналитические результаты, полученные при формализации основных стационарных и нестационарных характеристик СМО с ограниченным временем ожидания в очереди и эрланговским распределением времени обслуживания. Были получены аналитические выражения для основных числовых и вероятностных характеристик модели, а также значения производящей функции, которая позволяет вычислить основные нестационарные характеристики модели. С использованием преобразования Лапласа и последующего его обращения была получена производящая функция, что позволяет вычислить основные нестационарные характеристики модели. Эти результаты обеспечивают теоретическую основу для последующего построения метамоделей и проведения имита-

ционных исследований, направленных на повышение эффективности и адаптивности систем массового обслуживания в реальных условиях эксплуатации

В третьей главе представлен метод исследования систем массового обслуживания (СМО) смешанного типа с эрланговским распределением времени обслуживания. Метод основан на определении критических значений числа фаз обслуживания, при которых система поддерживает стабильный режим работы. Для исследования разработана программа, с помощью которой был проведен ряд вычислительных экспериментов. Результаты представлены в виде графиков для различных приведенных интенсивностей модели СМО. Была обнаружена логарифмическая зависимость количества фаз обслуживания от скорости поступления заявок.

В четвертой главе рассмотрены принципы разработки многоканальной системы массового обслуживания с ограниченной очередью в виде программного продукта, реализованного на платформе .NET с использованием языка программирования C#. Особое внимание уделено высокопроизводительной обработке данных и параллельным вычислениям. Предлагаются архитектурные решения для реализации многоканальной системы массового обслуживания, обсуждаются основные компоненты системы, включая очередь, каналы обработки и механизмы управления заявками. Интеграция Parallel Framework в процесс разработки многоканальной СМО с ограниченной очередью позволяет оптимизировать распределение ресурсов и снизить время ожидания в системе. Приводятся примеры кода, которые демонстрируют аспекты реализации многопоточности и использование конкурентных коллекций для эффективного управления заявками в очереди.

Рассматриваются возможности для дальнейшего развития разработки многоканальных систем массового обслуживания на платформе .NET, учитывая постоянно меняющиеся требования и технологические возможности.

Особое внимание уделяется возможности расширения результатов, полученных в четвертой главе, для использования не только для представленной модели $M/E_r/1$, но и для многоканальных случаев. Подходы, использованные при разработке программного продукта, являются универсальными и могут быть

адаптированы для различных типов СМО, что позволяет улучшить масштабируемость и гибкость системы.

В заключении приводятся основные результаты, полученные в данной работе, а также рекомендации и перспективы дальнейшей разработки темы.

В приложении приводятся акты о внедрении результатов диссертационной работы, а также свидетельство о регистрации электронного ресурса.

1 АНАЛИТИЧЕСКИЙ ОБЗОР И ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

1.1 Обзор исследований, посвященных СМО с эрланговским распределением времени обслуживания

Зарождение теории массового обслуживания (ТМО) было обусловлено задачами телефонии. Впоследствии ТМО получила широкое развитие, о чем свидетельствует множество работ [1-45], посвященных аналитическим и имитационным методам расчета СМО. Основной вклад в разработку и анализ классических моделей ТМО внесли А.А. Боровков, Б.В. Гнеденко, Д. Кендалл, Д. Кокс, Д. Литтл, Ф. Поллячек, В. Смит, Л. Такач, А.Я. Хинчин. Но на практике до недавнего времени аналитические расчеты осуществлялись по формулам, предложенным А. К. Эрлангом в начале прошлого века.

Феллером [6] исследовались задачи, которые были связаны с понятием очереди. В работах М. Д. Кендалла было введено понятие вложенных цепей Маркова [7] и классификация систем массового обслуживания [8], широко используемая в настоящее время. А. Кларк первым получил решение уравнений процесса гибели и размножения [9] с постоянными коэффициентами для переходного состояния. С. Карлин и Мак-Грегор предложили решение этой задачи с помощью коэффициентов, представляющих из себя функции числа требований в очереди [10]. Далее, Т. Л. Саати было получено решение для переходного состояния марковской модели многоканальной системы, использующее преобразование Лапласа [2].

Начало исследований комбинированных моделей СМО можно найти в работах Дж. Козна, посвященных телефонии [11-22]. Эту же проблему активно изучали Л. Костен и А. Элдин, П. Ле-Галль, М. А. Шнепс, Г. Л. Ионин, Ю. Н. Корнышев. В литературе такие системы получили название "системы смешанного типа", так как они сочетают в себе элементы ожидания и отказов.

Х. Такаджи в своих работах [23,24] исследует функции распределения времени ожидания и времени обслуживания заявки в моделях $M/M/m/K$ и $M/M/m/K/N$ с дисциплиной FIFO (первым пришел, первым обслужен). Он также

явно вычисляет моменты распределения времени задержки, которые отражают качество обслуживания. Также рассматривает смешанные системы с ожиданием и отказами, которые включают несколько серверов и конечный буфер.

Задачи теории массового обслуживания в настоящее время решаются в работах ряда ученых: Т.И. Алиева, В.Н. Задорожного, А.А. Назарова, А.В. Печинкина, В.В. Рыкова, К.Е. Самуйлова, С.П. Сущенко, В.Н. Тарасова, М.П. Фархадова, М.А. Шнепс-Шнеппе и др. [25-38].

Системы массового обслуживания, у которых время обслуживания представляет собой сумму из нескольких независимых случайных величин, распределенных по одному и тому же экспоненциальному закону, которые, согласно классификации Кендалла имеют вид $M/E_r/1$, рассматривались в работах [2,3,11,13,39].

В работе [40] рассмотрена система с пуассоновским входным потоком и с эрланговски распределенным временем обслуживания. Предполагается, что время обслуживания фиксировано. Для этой модели изучена проблема оптимального выбора времени обслуживания в задаче минимизации интенсивности потока потерь. Численные результаты показывают, что при задании оптимальных значений регулирующих параметров системы может быть достигнуто значительное снижение скорости потерь в сети, особенно при умеренной нагрузке на сеть, когда распределение времени обслуживания имеет небольшую дисперсию.

В работе [41] представлена производственная модель системы, работающая по принципу СМО $M/E_r/1$. В работе представлена полная характеристика параметров, необходимых для оптимальной работы системы. Кроме того, исследуется влияние изменчивости времени обслуживания заявок системой.

Для нахождения вероятностных характеристик установившегося режима системы $M/E_r/1$ в статье [42] представлены два новых метода. В качестве одного из подходов к нахождению вероятностных характеристик вводится новый рекуррентный метод, являющийся более точным и эффективным в вычислительном отношении, чем существующие. Кроме того, в работе представлена новая аналити-

ческая формула для нахождения стационарных характеристик системы, что дает новое представление о ее поведении.

В работе [43] проводится комплексное имитационное исследование для верификации результатов, полученных для байесовской оценки средней длины очереди. Также в этой статье была найдена байесовская оценка интенсивности входного потока в модели $M/E_r/1$ с использованием метода наименьших квадратов.

При эксплуатации различных СМО, часто встречаются случаи, когда используются модели без так называемых 0-х входов и 0-х ожиданий. В работах [2,44] дано следующее определение:

Под **нулевым входом** понимается случай, когда на момент прихода заявки в систему имеется свободное обслуживающее устройство. При этом обслуживание заявки происходит незамедлительно.

Нулевое ожидание - случай, когда на момент прихода заявки все имеющиеся в системе обслуживающие устройства заняты, однако других заявок в очереди нет. Следовательно, заявка, ожидающая обслуживания, не имеет перед собой других требований в очереди.

При расчете параметров очереди необходимо учитывать состояния системы, в которых отсутствуют как 0-е входы, так и 0-е ожидания. Реальной очередью называется очередь, рассчитанная без учета этих двух случаев. Важно отметить, что минимальное число заявок для реальной очереди всегда равно 1 [44].

Впервые проблема реальных очередей была поставлена Т. Саати в его классической монографии [2], в которой для системы $M/M/m$ была рассмотрена величина, представляющая собой, как указано в [2], среднее число требований, которые действительно какое-то определённое время пребывали в очереди в ожидании начала обслуживания. В работе [46] получены результаты для СМО $M/E_r/1$ для случая реальной очереди, которые описаны ниже.

Предположим, что в некоторую систему массового обслуживания в разное время поступают пять однородных заявок. В трех случаях эти заявки обслуживаются немедленно, так как обслуживающий прибор на момент их поступления в

СМО был свободен. В двух других случаях заявки ожидают начала обслуживания в течение некоторого времени.

В одном из этих двух случаев первая заявка обнаруживает, что обслуживающий прибор занят другой заявкой, поступившей в систему ранее, и ожидающей начала обслуживания. В другом случае первая заявка обнаруживает в очереди перед собой две такие же заявки, ожидающие начала обслуживания.

Таким образом

$$Q = \frac{1 + 2}{5} = \frac{3}{5},$$

Однако учитывая только те две заявки, которые действительно ожидали в очереди на обслуживание, средняя длина очереди будет следующей

$$Q_{real} = \frac{1 + 2}{2} = \frac{3}{2}.$$

В данном случае Q можно условно называть математической очередью, в отличие от реальной очереди Q_{real} . Как видим, средняя длина математической очереди связана со средней длиной реальной очереди следующим отношением

$$Q_{real} = \frac{Q}{\sum p_k}. \tag{1.1}$$

где в знаменателе формулы стоит сумма вероятностей состояний системы, для которых действительно имеет место очередь. Для указанного числового примера формула (1.1) обеспечивает верный результат.

Чтобы вычислить среднюю длину реальной очереди, необходимо в нормировочном условии проводить суммирование не по всем состояниям системы, а лишь начиная с $k = 2$. Применяя формулы

$$P_j(z) = \sum_{i=1}^{\infty} p_{ij} z^i$$

и

$$\begin{aligned} -(\lambda + \mu)P_j(z) + \lambda z P_j(z) + \mu P_{j+1}(z) &= 0, 0 < j < r - 1, \\ -\lambda(P_r(z) + p_0) - \mu P_r(z) + \lambda z(P_r(z) + p_0) + \frac{\mu}{z} P_1(z) &= 0, \end{aligned} \tag{1.2}$$

и выражая P_j через P_{j-1} и применяя описанное выше нормировочное условие $\sum_{i=2}^{\infty} p_i = \sum_{i=2}^{\infty} \sum_{j=1}^r p_{ij} = 1$ получаем

$$\begin{aligned} \lambda p_0 &= \mu p_{11} \\ \lambda &= \mu \sum_{i=3}^{\infty} p_{i1} \end{aligned} \quad (1.3)$$

Принимая во внимание тот факт, что

$$p_{ij} = \frac{P_j^{(i)}(0)}{i!}$$

из (1.1.2) получаем

$$\frac{\lambda}{\mu} = \sum_{i=3}^{\infty} \frac{P_1^{(i)}(0)}{i!}. \quad (1.4)$$

Следовательно,

$$\frac{\lambda}{\mu} = P_1(1) - P_1(0) - P'(0) - \frac{P_1^{(2)}(0)}{2} \quad (1.5)$$

С учетом формул (1.2) и (1.5) получаем

$$\frac{\lambda}{\mu} = \frac{\lambda}{\mu} * \frac{p_0}{1-\rho} - \frac{\lambda}{\mu} * p_0 - \frac{\lambda}{\mu} * \left(\left(\frac{\lambda}{\mu} + 1 \right)^r - 1 \right) * p_0 \quad (1.6)$$

Таким образом, разделив обе части (1.6) на $\frac{\lambda}{\mu}$ получаем

$$p_0 = \frac{1-\rho}{\rho - (1-\rho) \left(\left(\frac{\lambda}{\mu} + 1 \right)^r - 1 \right)}. \quad (1.7)$$

Теперь, зная p_0 , можно рекуррентно вычислить совокупность вероятностей p_{ij} , исходя из системы уравнений Колмогорова:

$$\begin{aligned} \frac{dp_0(t)}{dt} &= -\lambda p_0(t) + \mu p_{11}(t), \\ \frac{dp_{1r}(t)}{dt} &= \lambda p_0(t) - (\lambda + \mu) p_{1r}(t) + \mu p_{21}(t) \\ \frac{dp_{ir}(t)}{dt} &= -(\lambda + \mu) p_{ir}(t) + \lambda p_{i-1,r}(t) + \mu p_{i+1,1}(t), \quad i > 1 \\ \frac{dp_{ij}(t)}{dt} &= -(\lambda + \mu) p_{ij}(t) + \lambda p_{i-1,j}(t) + \mu p_{i,j+1}(t), \quad i > 1, 1 < j < r - 1, \\ \frac{dp_{1j}(t)}{dt} &= -(\lambda + \mu) p_{1j}(t) + \mu p_{1,j+1}(t), \quad 1 < j < r - 1. \end{aligned} \quad (1.8)$$

Однако нужно отметить, что в данном случае вычисления иногда будут приводить к большим погрешностям из-за того, что придется складывать числа с противоположными знаками. Исходя из этого для вычисления совокупности вероятностей p_{ij} необходимо воспользоваться производящей функцией, которая будет иметь вид

$$P(z) = p_0 + \sum_j^r p_{ij} = p_0 * (1 - z) / (1 - z \left(1 + \frac{\lambda}{\mu} (1 - z)\right)^r) \quad (1.9)$$

Тогда, среднее число заявок в системе в случае реальной очереди определяется формулой

$$N_{real} = P'(1) = \rho + \frac{\rho^2(1+r^{-1})}{2(1-\rho)(\rho-(1-\rho)((\lambda/\mu+1)^r-1))}. \quad (1.10)$$

Тогда формула для расчета средней длины реальной очереди имеет вид

$$Q_{real} = N - \rho = \frac{\rho^2(1+r^{-1})}{2(1-\rho)(\rho-(1-\rho)((\lambda/\mu+1)^r-1))}. \quad (1.11)$$

Формула, полученная с применением теоремы полной вероятности и ранее выведенных результатов для преобразования Лапласа-Стилтьеса стационарного распределения времени ожидания начала обслуживания в условиях реальной очереди, имеет вид:

$$\omega(s) = \int_0^\infty e^{-sx} dW(x) = p_0 + \sum_{i=1}^\infty \sum_{j=1}^\infty \left(\frac{\mu}{\mu+s}\right)^{(i-1)r+j} = p_0 + \left(\frac{\lambda((\mu+s)^r - \mu^r)}{\lambda\mu^r - \lambda(\mu+s)^r + s(\mu+s)^r}\right) \quad (1.12)$$

Стационарное распределение времени пребывания заявки в системе $V(x)$ имеет преобразование Лапласа-Стилтьеса

$$\varphi(s) = \int_0^\infty e^{-sx} dV(x) = \omega(s) \left(\frac{\mu}{\mu+s}\right)^r \quad (1.13)$$

Тогда, с учетом формул (1.12) и (1.13), стационарные средние времена ожидания обслуживания и пребывания в системе в целом для случая реальной очереди имеют вид

$$\omega_{real} = -\omega'(0) = \frac{\lambda r^2 \left(1 + \frac{1}{r}\right) (\rho - 1)}{2(\mu - \lambda r)^2 (\rho - (1 - \rho) ((\hat{\rho} + 1)^r - 1))} = \frac{Q_{real}}{\lambda} \quad (1.14)$$

$$v_{real} = -\varphi'(0) = \frac{N_{real}}{\lambda} \quad (1.15)$$

Таким образом, получены формулы Литтла, которые связывают средние времена ожидания обслуживания и пребывания в системе со значениями средней длины очереди и среднего числа заявок в системе в целом в случае рассмотрения реальной очереди.

Среди задач ТМО выделяется класс задач с ограничениями на среднее время пребывания поступившей в систему заявки. Такие ограничения могут накладываться как на среднее время пребывания заявки в очереди, в ожидании начала обслуживания, так и в системе массового обслуживания в целом, как в очереди, так и под обслуживанием. Говоря другими словами, в СМО данного типа часть заявок являются «нетерпеливыми» заявками, которые, подождав некоторое время, могут уйти либо из очереди, либо из системы, в том числе находясь на стадии обслуживания. Модели такого типа являются наименее изученным классом систем массового обслуживания среди всех типов СМО.

Для эффективного управления системами, работающими по принципу систем массового обслуживания с ограничениями на среднее время пребывания поступивших заявок, необходимо учитывать множество факторов, таких как скорость обработки заявок, вероятность отказа и другие. Кроме того, эти системы могут быть подвержены влиянию внешних факторов, таких как изменение интенсивности потока заявок. Таким образом, разработка эффективных методов для расчета характеристик СМО с ограниченным временем пребывания заявок является актуальной и перспективной задачей.

Основной задачей при изучении такого рода систем является следующая [44]: в процессе вычислений появляются суммы конечного/бесконечного числа слагаемых, которые не сводятся к суммам стандартных геометрических прогрессий. Для решения таких задач могут быть использованы приближенные методы решения. В данном случае все числовые характеристики могут быть вычислены отдельно с помощью суммирования нескольких первых слагаемых соответствующего ряда как для бесконечного, так и для конечного случая. Таким образом не существует возможности для получения замкнутого аналитического решения, од-

нако есть возможность получить оценку основных характеристик СМО. Несмотря на это для некоторых прикладных задач этого может быть вполне достаточно.

В теории массового обслуживания, как области прикладных исследований, отсутствует замкнутое комплексное аналитическое решение, которое с необходимой точностью рассчитывало бы все ключевые числовые характеристики системы и взаимосвязи между ними. Иными словами, на данный момент не существует единого математического метода, позволяющего с требуемой точностью описать поведение системы массового обслуживания. Данный пробел существенно ограничивает возможности практического применения и не позволяет должным образом оптимизировать работу систем. Важно отметить, что отсутствие замкнутого решения не означает, что теория массового обслуживания не имеет практической ценности. Существуют различные приближенные методы, которые позволяют получить достаточно точные оценки характеристик систем массового обслуживания. Тем не менее, разработка комплексного аналитического решения является важной задачей, которая позволит существенно расширить возможности применения теории массового обслуживания в различных областях.

В исследованиях, которые приведены в работе [44] впервые был использован подход к анализу СМО с ограничениями на среднее время пребывания заявки как в очереди, так и в системе в целом. Для суммирования ранее не поддающихся анализу рядов была предложена функция Г. Миттаг-Леффлера 1-го порядка. Это позволило математически формализовать задачу, и предоставляло возможность вычисления 1-х и 2-х моментов числовых характеристик СМО. Однако в данной работе решается задача с ограничением на среднее время лишь в случае, когда все поступающие в систему заявки можно считать "нетерпеливыми". В целом, данный подход заложил основы для изучения СМО с ограничениями на среднее время пребывания заявок, но его применение ограничено случаем "нетерпеливых" заявок. С точки зрения возможных приложений, более интересной представляется расширенная постановка задачи: каждая поступившая в систему заявка будет может ожидать начала обслуживания не более случайного времени, распределенно-

го по экспоненциальному закону, с временами обслуживания, распределенными по закону Эрланга ($M/E_r/1$).

Модели СМО делятся на два типа: стационарные и нестационарные. Для стационарных моделей разработана четкая и в значительной степени завершенная теория [47]. Нестационарные модели значительно сложнее для анализа. Системы массового обслуживания описываются с помощью линейных дифференциальных уравнений, которые решаются в случае постоянных коэффициентов. Если же предположить, что вероятности являются константами во времени, то система дифференциальных уравнений сводится к системе линейных алгебраических уравнений с явными решениями. Это и является стационарным случаем.

Для нестационарного же случая вероятности будут являться не постоянными величинами (как в стационарном случае), а функциями времени. В работе [45] для модели $M/E_r/1$ были получены нестационарные характеристики с помощью введения производящей функции и преобразования Лапласа.

1.2 Федеральная государственная информационная система территориального планирования как система массового обслуживания

Обеспечение бесперебойного, стабильного функционирования информационных систем является сложной задачей. Появляются новые виды электронных услуг, возрастают требования к их качеству. Возникает проблема оценки параметров, необходимых для стабильной работы системы.

Федеральная государственная информационная система территориального планирования (ФГИС ТП) представляет собой информационную систему, построенную на базе геоинформационных технологий, и предназначенную для сбора, хранения и предоставления доступа к информации, необходимой для обеспечения деятельности органов государственной власти Российской Федерации, органов государственной власти субъектов Российской Федерации и органов местного самоуправления в области территориального планирования.

Посредством информационной системы территориального планирования с использованием официального сайта в сети Интернет, определенного федеральным органом исполнительной власти, уполномоченным на осуществление контроля за соблюдением порядка ведения информационной системы территориального планирования, должен обеспечиваться доступ органов государственной власти, органов местного самоуправления, физических и юридических лиц к следующей необходимой для подготовки документов территориального планирования информации.

Основными задачами ФГИС ТП являются:

- 1) организация доступа органов государственной власти Российской Федерации, органов государственной власти субъектов Российской Федерации, органов местного самоуправления к сведениям, необходимым для осуществления ими полномочий в области территориального планирования;
- 2) обеспечение согласования в электронном виде проектов документов территориального планирования;
- 3) обеспечение публичности принятия органами государственной власти Российской Федерации, органами государственной власти субъектов Российской Федерации, органами местного самоуправления решений по развитию территории Российской Федерации;
- 4) информационная поддержка деятельности органов государственной власти Российской Федерации, органов государственной власти субъектов Российской Федерации, органов местного самоуправления, юридических и физических лиц в сфере территориального планирования.

Система обеспечивает возможность обращения за методической и технической поддержкой по телефону бесплатной горячей линии.

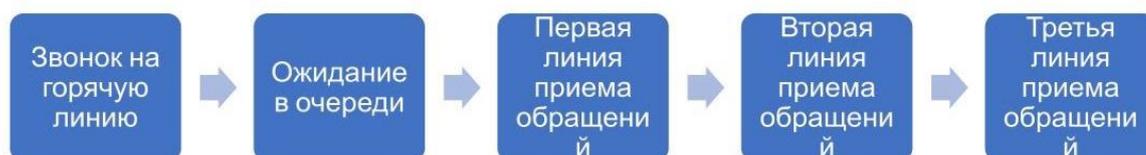


Рис. 1.2.1. Схема предоставления технической поддержки

В системе реализована многоуровневая поддержка пользователей.

Первый уровень первичную консультацию по общим вопросам. Сотрудники первого уровня поддержки предоставляют базовые знания и помогают пользователям сориентироваться в системе.

Второй уровень поддержки занимается более сложными вопросами, требующими углубленных знаний и навыков. Сюда относятся типовые задачи, возникающие при работе с системой.

Третий уровень поддержки – это команда экспертов, способных решать нестандартные проблемы, связанные с настройкой и обновлением программного обеспечения, а также с оценкой задач по реализации рекомендаций ФГИС ТП.

Основная функция первого уровня поддержки – оказать первую помощь пользователю. Сотрудники первого уровня поддержки консультируют по общим вопросам, предоставляют справочную информацию и помогают с базовыми настройками. Если же проблема выходит за рамки их компетенции, запрос перенаправляется на второй или третий уровень поддержки в зависимости от его сложности.

Таким образом, многоуровневая система поддержки обеспечивает комплексный подход к решению проблем пользователей, гарантируя им оперативную и квалифицированную помощь на всех этапах работы с системой.

Для получения информации о характеристиках системы технической поддержки могут использоваться различные методы и инструменты. В данной задаче имеет смысл применить аппарат математического моделирования системы технической поддержки для определения ее производительности и эффективности. Моделирование позволяет провести вычислительные эксперименты с различными параметрами системы, чтобы определить наилучшие варианты ее настройки и улучшения. Важно отметить, что система технической поддержки может иметь различные типы обслуживания и устройства для обработки заявок. Например, вместо одного обслуживающего устройства может быть несколько, что приведет к изменению типа системы массового обслуживания.

1.3 Постановка задачи

Рассмотренная выше система является системой, работающей по принципу систем массового обслуживания и, согласно классификации Кендалла, имеет вид $M/E_r/1$, так как подразумевает под собой многофазное обслуживание одним обслуживающим устройством (системой технической поддержки). В данной постановке существуют ограничения на среднее время пребывания поступившей в систему заявки. Каждая поступившая в систему заявка может покинуть очередь. Говоря другими словами, в СМО данного типа часть заявок являются «нетерпеливыми» заявками, которые, подождав некоторое время, уходят из очереди. Для обеспечения стабильной работы системы технической поддержки необходимо получать информацию об основных характеристиках системы.

Рассмотрим однолинейную систему $M/E_r/1$ с накопителем неограниченной емкости. В СМО поступает простейший поток заявок со скоростью λ , времена обслуживания заявок являются независимыми и распределены по закону Эрланга с параметрами r и μ .

Одним из методов работы с системами такого вида является построение марковского процесса, описывающий их функционирование, с использованием «фиктивных фаз». Данный метод подробно описан, в частности, в [2,3]. В основе данного метода лежит вероятностная интерпретация распределений типа РН. Суть метода заключается в следующем. Пусть $\nu(t)$ - число заявок в системе в момент времени t . В случае эрланговского времени обслуживания можно считать, что каждая заявка проходит r фаз обслуживания, каждая из которых имеет экспоненциальное распределение с параметром μ . Пусть $\xi(t)$ - число фаз, оставшихся для обслуживания заявки в момент времени t . Случайный процесс, характеризующий число заявок в СМО в момент времени t уже не будет для данной СМО марковским. Определим процесс $\{\eta(t), t \geq 0\}$ следующим образом: если в системе в момент времени t нет заявок, то $\eta(t) = \nu(t)$. В данном случае марковское свойство обеспечивается пуассоновским характером входного потока в виду того, что обслуживания в данном случае не происходит. В противном случае, если в системе

происходит обслуживание, то дополнительно указывается число оставшихся фаз в обслуживании, то есть $\eta(t) = (v(t), \xi(t))$. В силу экспоненциальности фаз обслуживания и пуассоновского характера входного потока заявок процесс $\eta(t)$ обладает марковским свойством.

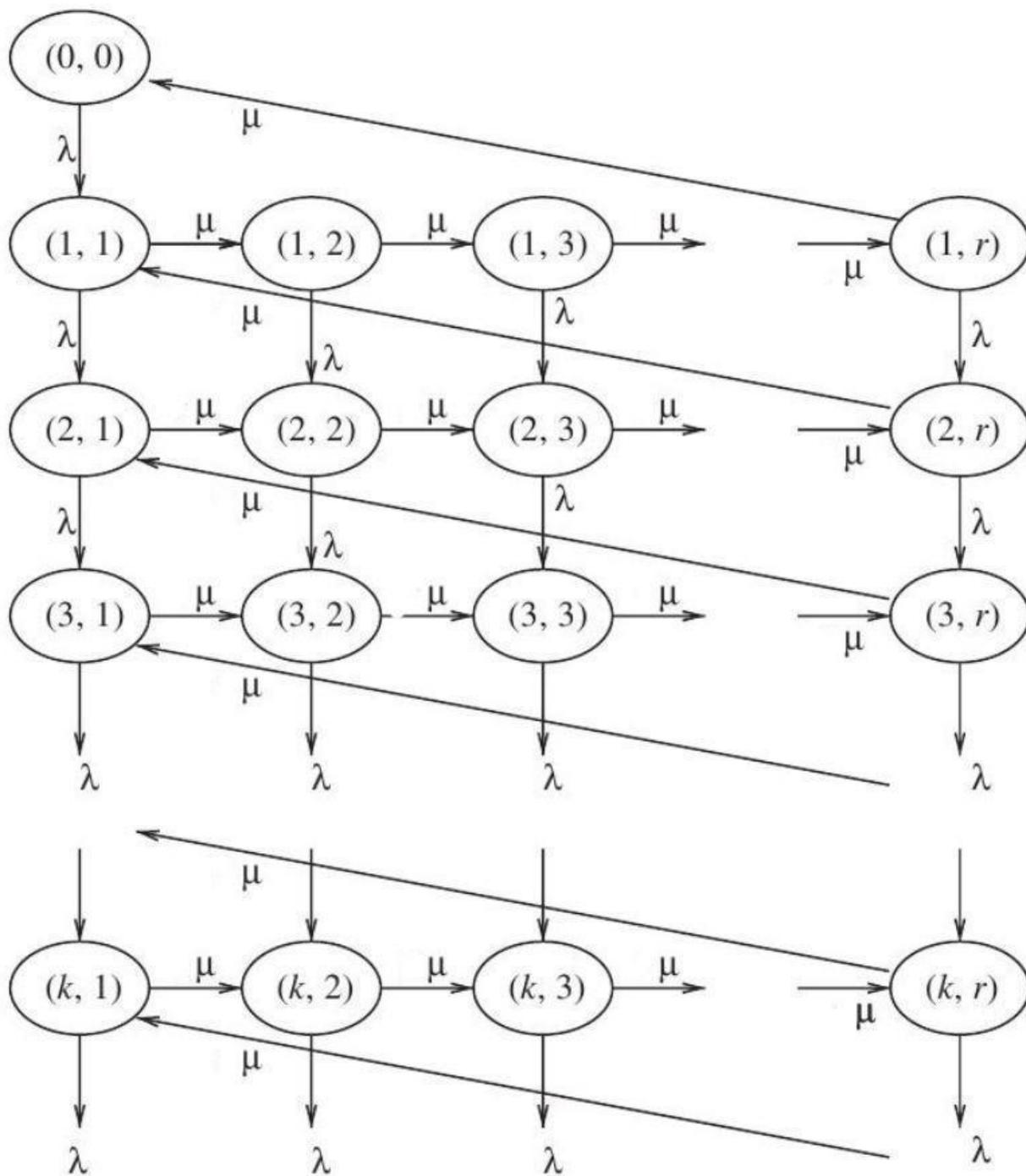


Рис. 1.3.1 Диаграмма состояний и переходов системы $M/E_r/1$

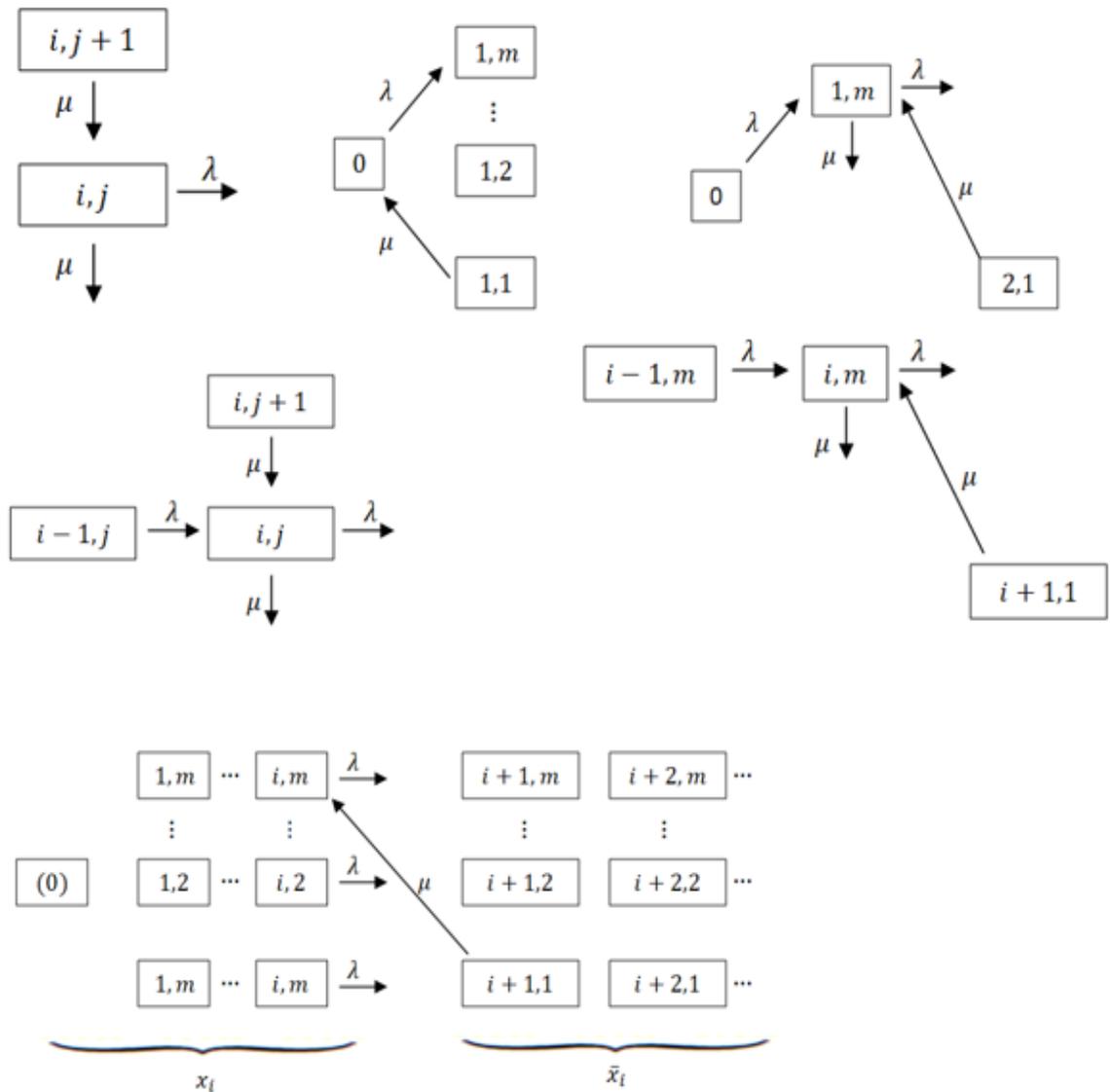


Рис. 1.3.2 Диаграмма переходов системы $M/E_r/1$

Для вывода уравнений Колмогорова необходимо определить вероятности перехода процесса за промежуток времени $(t, t + \Delta t)$ [13]:

- 1) Из (0) возможен переход только в $(1, r)$ за время Δ . Это означает, что в систему поступила первая заявка и началось обслуживание с первой фазы. Вероятность при этом будет равна $\lambda\Delta + o(\Delta)$.
- 2) Из (i, j) состояния, $i > 0, 1 < j < r + 1$, допустимы переходы в состояния $(i + 1, j)$ (заявка продолжает обслуживаться на текущей фазе, при этом еще одна заявка поступила в систему.) и $(i, j - 1)$ (закончилось об-

служивание j фазы), с вероятностями $\lambda\Delta + o(\Delta)$ и $\mu\Delta + o(\Delta)$ соответственно.

3) Из $(i, 1)$, $i > 1$ возможны переходы в $(i + 1, 1)$ и $(i - 1, r)$ с вероятностями $\lambda\Delta + o(\Delta)$ и $\mu\Delta + o(\Delta)$ соответственно.

4) Из $(1, 1)$ возможны переходы в $(2, 1)$ и в (0) с вероятностями $\lambda\Delta + o(\Delta)$ и $\mu\Delta + o(\Delta)$ соответственно.

Вероятности остальных переходов равны $o(\Delta)$.

В представленной на рис. 1.3.2 схеме можно увидеть, что некоторые состояния (кроме состояния 0) могут иметь **больше двух** возможных переходов. Это **отличает** данную схему от классической схемы гибели и размножения, где каждое состояние имеет лишь два перехода.

В рамках исследования системы смешанного типа $M/E_r/1$, для её детального анализа и моделирования целесообразно использовать систему дифференциальных уравнений Колмогорова. С учетом вышеизложенного и с помощью диаграммы состояний и переходов системы имеем:

$$\begin{aligned}
 \frac{dp_0(t)}{dt} &= -\lambda p_0(t) + \mu p_{11}(t), \\
 \frac{dp_{1r}(t)}{dt} &= \lambda p_0(t) - (\lambda + \mu)p_{1r}(t) + \mu p_{21}(t) \\
 \frac{dp_{ir}(t)}{dt} &= -(\lambda + \mu)p_{ir}(t) + \lambda p_{i-1,r}(t) + \mu p_{i+1,1}(t), i > 1 \\
 \frac{dp_{ij}(t)}{dt} &= -(\lambda + \mu)p_{ij}(t) + \lambda p_{i-1,j}(t) + \mu p_{i,j+1}(t), i > 1, 1 < j < r - 1, \\
 \frac{dp_{1j}(t)}{dt} &= -(\lambda + \mu)p_{1j}(t) + \mu p_{1,j+1}(t), 1 < j < r - 1.
 \end{aligned} \tag{1.16}$$

Полагая, что система перешла в установившийся режим, получаем систему:

$$\begin{aligned}
 0 &= -\lambda p_0 + \mu p_{11}, \\
 0 &= -(\lambda + \mu)p_{1j} + \mu p_{1,j+1}, \quad j = (1, r - 1), \\
 0 &= -(\lambda + \mu)p_{1r} + \lambda p_0 + \mu p_{21}, \\
 0 &= -(\lambda + \mu)p_{ij} + \lambda p_{i-1,j} + \mu p_{i,j+1}, \quad i \geq 2, \\
 0 &= -(\lambda + \mu)p_{ir} + \lambda p_{i-1,r} + \mu p_{i+1,1}, \quad i \geq 2.
 \end{aligned} \tag{1.17}$$

На основании уравнений 1.16 вводится производящая функция, определяется совокупность вероятностей p_{ij} , стационарные числовые и временные характеристики.

Для решения поставленной задачи рассмотрим расширенную постановку задачи, а именно систему $M/E_r/1$ с ограничениями на время пребывания заявки в очереди.

Для рассмотрения системы ФГИС ТП как системы массового обслуживания, а также для получения нестационарных характеристик предполагается описание процесса с помощью системы линейных дифференциальных уравнений. Для нестационарного случая, как было отмечено ранее, вероятности не являются постоянными величинами, а являются функциями времени.

Выводы

1. Недостаточное развитие теоретических основ и научно-обоснованных аналитических методов для расчета как стационарных, так и нестационарных характеристик систем смешанного типа препятствует дальнейшей модернизации аналитического аппарата для исследования аналогичных процессов.

2. Дальнейшее уточнение моделей может быть направлено на учет ограничений не только на среднее время пребывания в очереди, но и ограничений на время пребывания в СМО в целом.

3. Проблемой в теории массового обслуживания для систем смешанного типа является отсутствие аналитического решения как для стационарного, так и для нестационарного случаев, которое бы позволило рассчитать с достаточной точностью и связать между собой все основные характеристики системы.

2 ХАРАКТЕРИСТИКИ СМО $M/E_r/1$ С ОГРАНИЧЕННЫМ ВРЕМЕНЕМ ПРЕБЫВАНИЯ ЗАЯВКИ В ОЧЕРЕДИ

2.1 Стационарные характеристики одноканальной СМО

Рассмотрим одноканальную СМО с однородным бесконечным простейшим потоком заявок и очередью неограниченной длины. Пусть интенсивность потока заявок равна λ . Для времени обслуживания используется эрланговское распределение, где каждая заявка, поступающая в систему, проходит несколько фаз обслуживания. Длительности этих фаз распределены по экспоненциальному закону с параметром μ .

Первая постановка задачи:

В первой постановке задачи заявка может уходить только из очереди, то есть до того, как начнется её обслуживание. Если заявка не может быть обслужена сразу после поступления в очередь (например, из-за того, что обслуживающее устройство занято), она остаётся в очереди, пока не начнётся её обслуживание. Таким образом, заявка может покинуть систему только в том случае, если покидает очередь.

Вторая постановка задачи:

Во второй постановке задачи заявка может уйти как из очереди, так и с обслуживающего устройства в любой момент времени, включая любую из фаз обслуживания. То есть, даже если заявка уже начала обслуживаться, она может в любой момент времени прервать обслуживание и покинуть систему.

Особенности модели:

Поток заявок: Бесконечный простейший поток заявок с интенсивностью λ .

Очередь: Очередь имеет неограниченную длину, что означает, что число заявок в очереди может быть сколь угодно большим.

Время обслуживания: Время обслуживания каждой заявки состоит из нескольких фаз, длительности которых распределены по экспоненциальному закону с параметром μ .

Такие модели часто используются для анализа и оптимизации работы систем обслуживания в различных областях, таких как телекоммуникации, транспортные системы, интернет-сервисы и т.д. Использование эрланговского распределения для времени обслуживания позволяет более точно моделировать процессы обслуживания в реальных системах, где время обслуживания может варьироваться в зависимости от нагрузки и других факторов.

Представляет интерес первый вариант постановки задачи, в которой будем предполагать, что каждая поступившая заявка будет ожидать в очереди не более случайного времени, распределенного по экспоненциальному закону с параметром θ .

Как и ранее, пусть $\nu(t)$ - число заявок в системе в момент времени t . В случае эрланговского времени обслуживания будем считать, что каждая заявка проходит r фаз обслуживания, каждая из которых имеет экспоненциальное распределение с параметром μ , где $\xi(t)$ - число фаз, оставшихся для обслуживания заявки в момент времени t . Определим процесс $\{\eta(t), t \geq 0\}$ следующим образом: если в системе в момент времени t нет заявок, то $\eta(t) = \nu(t)$. В противном случае, если в системе происходит обслуживание, то дополнительно указывается число оставшихся фаз в обслуживании, то есть $\eta(t) = (\nu(t), \xi(t))$. Каждая прибывшая в систему заявка будет ожидать в очереди не более случайного времени, распределенного по экспоненциальному закону с параметром θ . Граф состояний системы изображен на рис.2.1.1.

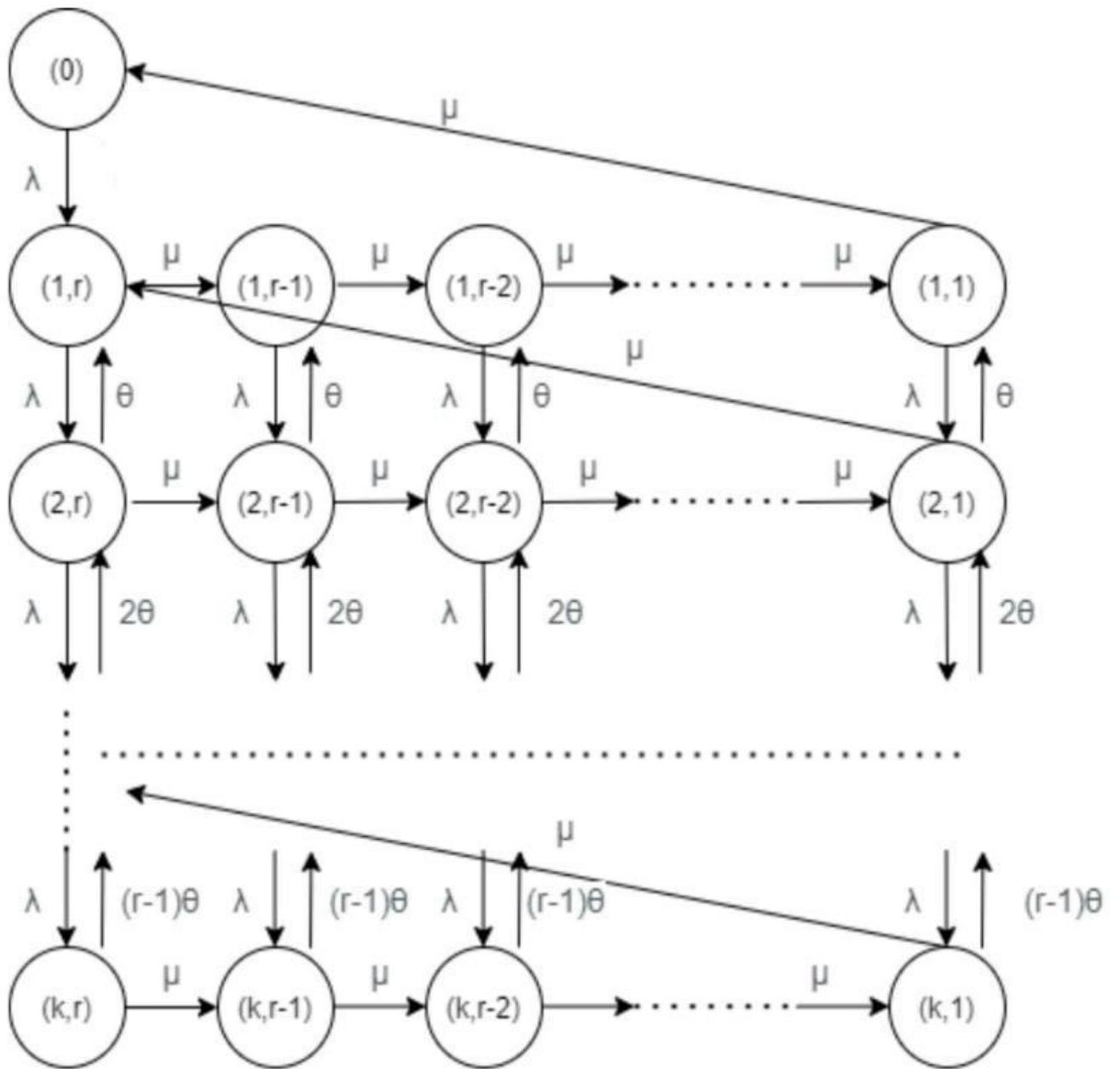


Рис. 2.1.1. Граф состояний и переходов СМО

Согласно графу, имеем:

$$\begin{aligned}\frac{dp_0(t)}{dt} &= -\lambda p_0(t) + \mu p_{11}(t), \\ \frac{dp_{1r}(t)}{dt} &= \lambda p_0(t) + \theta p_{2r}(t) - (\lambda + \mu)p_{1r}(t) + \mu p_{21}(t)\end{aligned}\quad (2.1)$$

$$\begin{aligned}\frac{dp_{ir}(t)}{dt} &= -(\lambda + \mu)p_{ir}(t) + \lambda p_{i-1,r}(t) + \mu p_{i+1,1}(t) + i\theta p_{i+1,r}(t) - \\ &- (i-1)\theta p_{ir}(t), i > 1\end{aligned}$$

$$\begin{aligned}\frac{dp_{ij}(t)}{dt} &= -(\lambda + \mu)p_{ij}(t) + \lambda p_{i-1,j}(t) + \mu p_{i,j+1}(t) + i\theta p_{i+1,j}(t) - \\ &- (i-1)\theta p_{ij}(t), i > 1, 1 < j < r-1,\end{aligned}$$

$$\frac{dp_{1j}(t)}{dt} = -(\lambda + \mu)p_{1j}(t) + \mu p_{1,j+1}(t) + \theta p_{2j}(t), 1 < j < r-1.$$

Предполагая, что система функционирует в установившемся режиме, стационарные вероятности удовлетворяют системе уравнений:

$$\begin{aligned}0 &= -\lambda p_0 + \mu p_{11}, \\ 0 &= \lambda p_0 + \theta p_{2r} - (\lambda + \mu)p_{1r} + \mu p_{21} \\ 0 &= -(\lambda + \mu)p_{ir} + \lambda p_{i-1,r} + \mu p_{i+1,1} + i\theta p_{i+1,r} - (i-1)\theta p_{ir}, i > 1 \\ 0 &= -(\lambda + \mu)p_{ij} + \lambda p_{i-1,j} + \mu p_{i,j+1} + i\theta p_{i+1,j} - (i-1)\theta p_{ij}, i > 1, \\ &1 < j < r-1, \\ 0 &= -(\lambda + \mu)p_{1j} + \mu p_{1,j+1} + \theta p_{2j}, 1 < j < r-1.\end{aligned}\quad (2.2)$$

Просуммировав уравнения (2.1) по i при фиксированном j , получим

$$\sum_i p_{i1} = \sum_i p_{i2} = \dots = \sum_i p_{ir} \quad (2.3)$$

Просуммировав уравнения (2.1) по j при фиксированном i , получим

$$\lambda \sum_j p_{ij} = \mu p_{i+1,1} + \sum_j i\theta p_{i+1,j}, i > 0.$$

Тогда, с учетом (2.2) и условия нормировки $p_0 + \sum_i \sum_j p_{ij} = 1$ получаем следующее равенство

$$\lambda p_0 + \sum_{i=1}^{\infty} \sum_j \lambda p_{ij} = \sum_{i=1}^{\infty} \sum_j i\theta p_{i+1,j} + \sum_{i=1}^{\infty} \mu p_{1,j+1}, \quad (2.4)$$

откуда

$$\lambda = \theta Q + \sum_{i=1}^{\infty} \mu p_{1,j+1}. \quad (2.5)$$

Получаем равенство

$$r\lambda = \theta rQ + (1 - p_0)\mu, \quad (2.6)$$

имеющее смысл: в стационарном режиме интенсивность входного потока, умноженная на r , равна сумме интенсивностей $(1 - p_0)\mu$ потоков обслуженных и θrQ «нетерпеливых» заявок.

Из (2.6) следует, что

$$p_0 = 1 - \frac{r(\lambda - \theta Q)}{\mu} \quad (2.7)$$

Для нахождения вероятностных и числовых характеристик системы снова воспользуемся производящей функцией $P_j(z) = \sum_{i=1}^{\infty} p_{ij}z^i$.

Умножая уравнения для p_{ij} из (1) на z^i и суммируя, получим

$$-(\lambda + \mu)P_j(z) + \lambda z P_j(z) + \mu P_{j+1}(z) + (1 - z) \sum_{i=2}^{\infty} (i - 1)z^{i-1} \theta p_{ij} = 0,$$

$$0 < j < r - 2,$$

$$-\lambda(P_r(z) + p_0) - \mu P_r(z) + \lambda z(P_r(z) + p_0) + \frac{\mu}{z} P_1(z) + (1 - z) \sum_{i=2}^{\infty} (i - 1)z^{i-1} \theta p_{ir} = 0.$$

Выразив $P_{j+1}(z)$ через $P_j(z)$ находим

$$P_j(z) = \left(1 + \frac{\lambda}{\mu}(1 - z)\right)^{j-1} P_1(z) - \frac{1-z}{\mu} \sum_{i=2}^{\infty} (i - 1)z^{i-1} \theta p_{ij}, \quad 1 < j < r,$$

$$P_r(z) = \frac{-p_0 \frac{\lambda}{\mu}(1-z) + \frac{P_1}{z} + \frac{(1-z) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\mu}}{1 + \frac{\lambda}{\mu}(1-z)}.$$

Для нахождения $P_1(z)$ воспользуемся формулами для $P_r(z)$ и $P_j(z)$.

$$P_1(z) = \frac{\frac{p_0 \lambda}{\mu} (1-z) - \frac{(1-z) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\mu \left(1 + \frac{\lambda}{\mu(1-z)}\right)}}{1 + \frac{\lambda}{\mu(1-z)} - \left(1 + \frac{\lambda}{\mu(1-z)}\right)^{-1+r} + \frac{1}{\left(1 + \frac{\lambda}{\mu(1-z)}\right)z}}$$

$$- \frac{(1-z) \left(-1 - \frac{\lambda}{\mu} + \left(1 + \frac{\lambda}{\mu(1-z)}\right)^r + \frac{\lambda}{\mu} z\right) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\mu \frac{\lambda}{\mu} (-1+z) \left(-1 - \frac{\lambda}{\mu} + \frac{\lambda}{\mu} z\right)} - \frac{1}{\left(1 + \frac{\lambda}{\mu(1-z)}\right)z} - \left(1 + \frac{\lambda}{\mu(1-z)}\right)^{-1+r}$$

После нахождения $P_1(z)$ подставив его в выражение для $P_j(z)$ получим общую формулу для вычисления производящей функции:

$$P(z) = p_0 + \sum_{j=1}^r P_j(z) =$$

$$p_0 + \sum_{j=1}^r \left(\frac{\left(1 + \frac{\lambda}{\mu(1-z)}\right)^{-1+j}}{-\left(1 + \frac{\lambda}{\mu(1-z)}\right)^{-1+r} + \frac{1}{\left(1 + \frac{\lambda}{\mu(1-z)}\right)z}} \right) *$$

$$\begin{aligned}
& * \left(\frac{\frac{p_0 \lambda}{\mu(1-z)}}{1 + \frac{\lambda}{\mu(1-z)}} - \frac{(1-z) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\mu \left(1 + \frac{\lambda}{\mu(1-z)}\right)} \right. \\
& \left. - \frac{(1-z) \left(-1 - \frac{\lambda}{\mu} + \left(1 + \frac{\lambda}{\mu(1-z)}\right)^r + \frac{\lambda}{\mu z}\right) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\frac{\mu \lambda}{\mu(-1+z) \left(-1 - \frac{\lambda}{\mu} + \frac{\lambda}{\mu z}\right)}} \right) \\
& - \left(\frac{\left((1-z) \left(-1 - \frac{\lambda}{\mu} + \left(1 + \frac{\lambda}{\mu(1-z)}\right)^j + \frac{\lambda}{\mu z}\right) \right)}{\mu \lambda} \right. \\
& \left. \frac{\mu \left(- \left(1 + \frac{\lambda}{\mu(1-z)}\right)^{-1+r} + \frac{1}{\left(1 + \frac{\lambda}{\mu(1-z)}\right) z} \right) (-1+z) \left(-1 - \frac{\lambda}{\mu} + \frac{\lambda}{\mu z}\right)}{\mu \lambda} \right) \\
& * \left(\sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij} \right) \\
& * \left(\frac{\frac{p_0 \lambda}{\mu(1-z)}}{1 + \frac{\lambda}{\mu(1-z)}} - \frac{(1-z) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\mu \left(1 + \frac{\lambda}{\mu(1-z)}\right)} \right. \\
& \left. - \frac{(1-z) \left(-1 - \frac{\lambda}{\mu} + \left(1 + \frac{\lambda}{\mu(1-z)}\right)^r + \frac{\lambda}{\mu z}\right) \sum_{i=2}^{\infty} (-1+i) \theta z^{-1+i} p_{ij}}{\frac{\mu \lambda}{\mu(-1+z) \left(-1 - \frac{\lambda}{\mu} + \frac{\lambda}{\mu z}\right)}} \right).
\end{aligned}$$

В задачах ТМО для нахождения производящей функции и характеристик, которые могут быть получены с помощью производящей функции, весьма полезной оказывается одна из тауберовых теорем, доказательство которых можно найти, например в [48-52], а именно

Теорема

Если ряд $\sum_{n=0}^{\infty} z^n a_n$, $a \geq 0$, сходится при $|z| < 1$ и $a_n = a < \infty$, то

$$(1 - z) \sum_{n=0}^{\infty} z^n a_n = a.$$

В данном случае $\sum_{i=2}^{\infty} (-1 + i)\theta z^{-1+i} p_{ij} = \sum_{i=0}^{\infty} i\theta z^i p_{i+1,j}$,

$$a_n = i\theta p_{i+1},$$

$$p_i = \sum_{j=1}^r p_{ij} = \frac{P^{(i)}(0)}{i!},$$

$$p_i \rightarrow 0 \text{ при } i \rightarrow \infty.$$

Таким образом, при $i \rightarrow \infty$ $\sum_{i=2}^{\infty} (-1 + i)\theta z^{-1+i} p_{ij} \rightarrow 0$.

При переходе к пределу при $i \rightarrow \infty$ множители вида $\sum_{i=2}^{\infty} (-1 + i)\theta z^{-1+i} p_{ij}$ обнуляются и выражение для $P(z)$ примет вид:

$$P(z) = p_0 + \sum_{j=1}^r P_j(z) = \tag{2.8}$$

$$p_0 + \sum_{j=1}^r \left(\frac{(1 + \lambda/\mu(1 - z))^{-1+j}}{-(1 + \lambda/\mu(1 - z))^{-1+r} + \frac{1}{(1 + \lambda/\mu(1 - z))z}} * \right. \\ \left. * \left(\frac{\frac{p_0 \lambda}{\mu(1 - z)}}{1 + \frac{\lambda}{\mu(1 - z)}} \right) \right).$$

Для нахождения характеристик модели необходимо найти производную производящей функции в точке 1.

$$N = P'(1) = \frac{\lambda(-2\mu + \lambda(-1+r))r(-\mu + r(\lambda + \theta))}{2\mu^3 - \lambda^2 r^2(1+r)\theta + 2\lambda\mu r^2(\lambda + \theta) - 2\mu^2 r(2\lambda + \theta)} \tag{2.9}$$

Для нахождения средней длины очереди сначала необходимо с помощью (2.6) и (2.9) получить значение для p_0 , а именно

$$1 - \frac{r\lambda - (1-p_0)\mu}{\theta r} = N - (1 - p_0).$$

С учетом этого равенства получим p_0 , то есть коэффициент простоя системы

$$p_0 = \frac{2(-1+\rho)^2(\mu-\lambda r-r\theta)}{2\mu-4\mu\rho+2\mu\rho^2-2r\theta+2r\rho\theta-\rho^2\theta-r\rho^2\theta} \quad (2.10)$$

Подставляя выражение для p_0 , получим следующую формулу для нахождения средней длины очереди

$$q = -1 + \frac{2(-1+\frac{\lambda r}{\theta})^2(\mu-\lambda r-r\theta)}{2\mu+6\lambda r^2+\frac{2\lambda^2\mu r^2}{\theta^2}-\frac{4\lambda\mu r}{\theta}+\frac{\lambda r^2}{\theta}-\frac{3\lambda^2 r^3}{\theta}-2r\theta} - \frac{\lambda(-2\mu+\lambda(-1+r))r(-1+\frac{\lambda r}{\theta})^2(\mu-\lambda r-r\theta)}{(\mu-\lambda r)^2(2\mu+6\lambda r^2+\frac{2\lambda^2\mu r^2}{\theta^2}-\frac{4\lambda\mu r}{\theta}+\frac{\lambda^2 r^2}{\theta}-\frac{3\lambda^2 r^3}{\theta}-2r\theta)} \quad (2.11)$$

Зная формулу для p_0 можно рекуррентно вычислить значения для p_{ij} исходя из системы уравнений (2.1). Однако, это может привести к большим погрешностям. Поэтому для вычисления стационарных вероятностей используется производящая функция:

$$p_i = \sum_{j=1}^r p_{ij} = \frac{P^{(i)}(0)}{i!}.$$

Одной из основных характеристик систем смешанного типа является P_S – стационарная вероятность того, что заявка будет обслужена. В систему за единицу времени поступает λ заявок. Так как прибор в среднем загружен долю времени $1 - p_0$, а при полной загрузке он обслуживает $\frac{\mu}{r}$ заявок, то фактически обслуживается $\frac{\mu}{r}(1 - p_0)$ заявок в единицу времени. Разделяя теперь среднее число обслуженных заявок на среднее число поступивших заявок, получаем формулу

$$P_S = \frac{1}{\rho}(1 - p_0) = \frac{1}{\rho} \left(1 - \frac{2(-1+\rho)^2(\mu-\lambda r-r\theta)}{2\mu-4\mu\rho+2\mu\rho^2-2r\theta+2r\rho\theta-\rho^2\theta-r\rho^2\theta} \right). \quad (2.12)$$

Для реализации данной формулы в **Wolfram Mathematica**, приведённой в уравнении (2.12), напишем код (рис. 2.1.2) для расчёта P_S :

```

(* определение параметров *)
rho = \[Lambda]/\[Mu]; (* приведенная интенсивность *)
\[Mu] = 1; (* средняя интенсивность обслуживания *)
r = 1; (* количество фаз обслуживания *)
\[Theta] = 0.5; (* время ожидания или ограничение по времени *)
\[Lambda] = 0.8; (* интенсивность входного потока *)

(* формула для PS (стационарная вероятность обслуживания) *)
PS = 1/rho * (1 - (2*(-1 + rho)^2*(\[Mu] - \[Lambda]*r - r*\[Theta])) /
    (2*\[Mu] - 4*\[Mu]*rho + 2*\[Mu]*rho^2 - 2*r*\[Theta] +
    2*rho*\[Theta] - rho^2*\[Theta] - r*rho^2*\[Theta]));

(* вывод результата *)
PSResult = N[PS]; (* численное значение PS *)

(* печать результата *)
Print["Стационарная вероятность обслуживания PS: ", PSResult];

```

Рис. 2.1.2 Код для расчёта P_S

Вторые моменты являются важными числовыми характеристиками различных типов систем массового обслуживания. Однако даже в современной научной литературе аналитические формулы для вычисления этих величин часто отсутствуют. В основном изучены моменты второго порядка для одноканальных моделей СМО различных типов.

Вторые моменты, такие как, например, дисперсия времени ожидания в очереди, важны для оценки производительности многоканальных СМО. Для простых моделей с бесконечной очередью существуют аналитические методы расчета этих моментов. Однако для более реалистичных моделей с ограниченной очередью получение точных значений этих моментов представляет собой сложную задачу, и в литературе представлены методы расчета лишь для некоторых числовых характеристик. Несмотря на обширные исследования различных прикладных аспектов ТМО, точные значения вторых моментов характеристик для многих СМО с ограниченной очередью остаются неизвестными.

Изучение этих характеристик позволяет делать ряд нетривиальных выводов о режимах функционирования подобных СМО. Это важно для оптимизации процессов обслуживания, повышения эффективности использования ресурсов и улучшения качества обслуживания клиентов [44, 53, 54].

Дисперсия числа заявок в очереди рассчитывается с использованием производящей функции, принимая во внимание тот факт, что $\sum_k kp_k = P'(1)$, а $\sum_k k^2 p_k = P'(1) + P''(1)$ [5].

$$\begin{aligned}\sigma_Q^2 &= \sum_k (k-1)^2 p_k - Q^2 = \sum_k k^2 p_k - 2 \sum_k kp_k + \sum_k p_k - Q^2 = \\ &= P'(1) + P''(1) - 2P'(1) + (1 - p_0) - Q^2 = \\ &= 1 - \frac{r^2 (\mu \hat{\rho} (-2 + (-1+r)\hat{\rho}) + \lambda (2+r(-2+\hat{\rho})\hat{\rho} + r^2 \hat{\rho}^2))^2}{(-2\mu(-1+\rho)^2 + r(2+r(-2+\hat{\rho})\hat{\rho} + \rho^2)\theta)^2} - \\ &\quad \frac{\lambda^2 r(1+r)(6\mu^2 - 4\lambda\mu(-1+r) + \lambda^2(-1+r)r)(-1+\rho)^2(-\mu+r(\lambda+\theta))}{3\mu(\mu-\lambda r)^3(2\mu(-1+\rho)^2 - r(2+r(-2+\hat{\rho})\hat{\rho} + \rho^2)\theta)} - \\ &\quad \frac{2\rho(-2+(-1+r)\hat{\rho})(-\mu+r(\lambda+\theta))}{-2\mu(-1+\rho)^2 + r(2+r(-2+\hat{\rho})\hat{\rho} + \rho^2)\theta} - \frac{2(-1+\rho)^2(-\mu+r(\lambda+\theta))}{-2\mu(-1+\rho)^2 + r(2+r(-2+\hat{\rho})\hat{\rho} + \rho^2)\theta},\end{aligned}$$

где $\hat{\rho} = \frac{\lambda}{\mu}$.

Дисперсия числа заявок в системе

$$\begin{aligned}\sigma_N^2 &= \sum_k k^2 p_k - N^2 = P'(1) + P''(1) - N^2 = \\ &= \frac{\rho^2 (-2 + (-1+r)\hat{\rho})^2 (\mu - \lambda r - r\theta)^2}{(2\mu - 4\mu\rho + 2\mu\rho^2 - 2r\theta + 2r\rho\theta - \rho^2\theta - r\rho^2\theta)^2} \\ &+ \frac{\lambda^2 r(1+r)(6\mu^2 - 4\lambda\mu(-1+r) + \lambda^2(-1+r)r)(-1+\rho)^2(\mu - \lambda r - r\theta)}{3\mu(\mu - \lambda r)^3(2\mu - 4\mu\rho + 2\mu\rho^2 - 2r\theta + 2r\rho\theta - \rho^2\theta - r\rho^2\theta)}\end{aligned}$$

В ТМО функция распределения времени ожидания начала обслуживания "терпеливой" заявки играет ключевую роль. Она позволяет вычислить средние времена пребывания заявки в СМО и очереди, что является важным показателем для оценки производительности системы.

Для определения временных характеристик системы, таких как среднее время пребывания заявки в СМО и среднее время ожидания в очереди, необходимо найти распределение времени ожидания для "терпеливой" заявки.

При этом, функция распределения времени ожидания для "терпеливой" заявки является критичной для оценки производительности системы массового обслуживания. Она позволяет оценить, насколько быстро заявки начинают обслуживаться после поступления в систему. Это важно для оптимизации процессов обслуживания, улучшения качества обслуживания клиентов и эффективного использования ресурсов.

Преобразование Лапласа-Стилтьеса функции распределения $W_i(x)$ имеет вид:

$$\omega_i(s) = \int_0^{\infty} e^{-sx} dW_i(x) = \frac{\frac{\mu}{r} + (i-1)\theta}{s + \frac{\mu}{r} + (i-1)\theta} * \frac{\frac{\mu}{r} + (i-2)\theta}{s + \frac{\mu}{r} + (i-2)\theta} * \dots * \frac{\frac{\mu}{r}}{s + \frac{\mu}{r}}.$$

Функция распределения общего времени ожидания

$$\omega(s) = \sum_i \omega_i(s) p_i. \quad (2.13)$$

Тогда стационарное время ожидания начала обслуживания «терпеливой» заявкой находится по формуле

$$\omega = -\omega'(0) = \frac{\lambda(-\lambda r + \mu)r(1+r)}{2\mu^3 - \lambda^2 r^2(1+r)\theta + 2\lambda\mu r^2(\lambda + t) - 2\mu^2 r(2\lambda + \theta)} = \frac{Q}{\lambda}$$

Определим стационарное распределение $V(x)$ времени пребывания заявки в СМО.

$$\varphi_i(s) = \int_0^{\infty} e^{-sx} dV_i(x) = \int_0^{\infty} e^{-sx} \theta e^{-\theta x} [1 - W_i(x)] dx + \int_0^{\infty} e^{-sx} \frac{\frac{\mu}{r}}{s + \frac{\mu}{r}} e^{-\theta x} dW_i(x).$$

$$v = -\varphi'(0) = \frac{(-2\mu + \lambda(-1+r))r(-\mu + r(\lambda + \theta))}{2\mu^3 - \lambda^2 r^2(1+r)\theta + 2\lambda\mu r^2(\lambda + \theta) - 2\mu^2 r(2\lambda + \theta)} = \frac{N}{\lambda}.$$

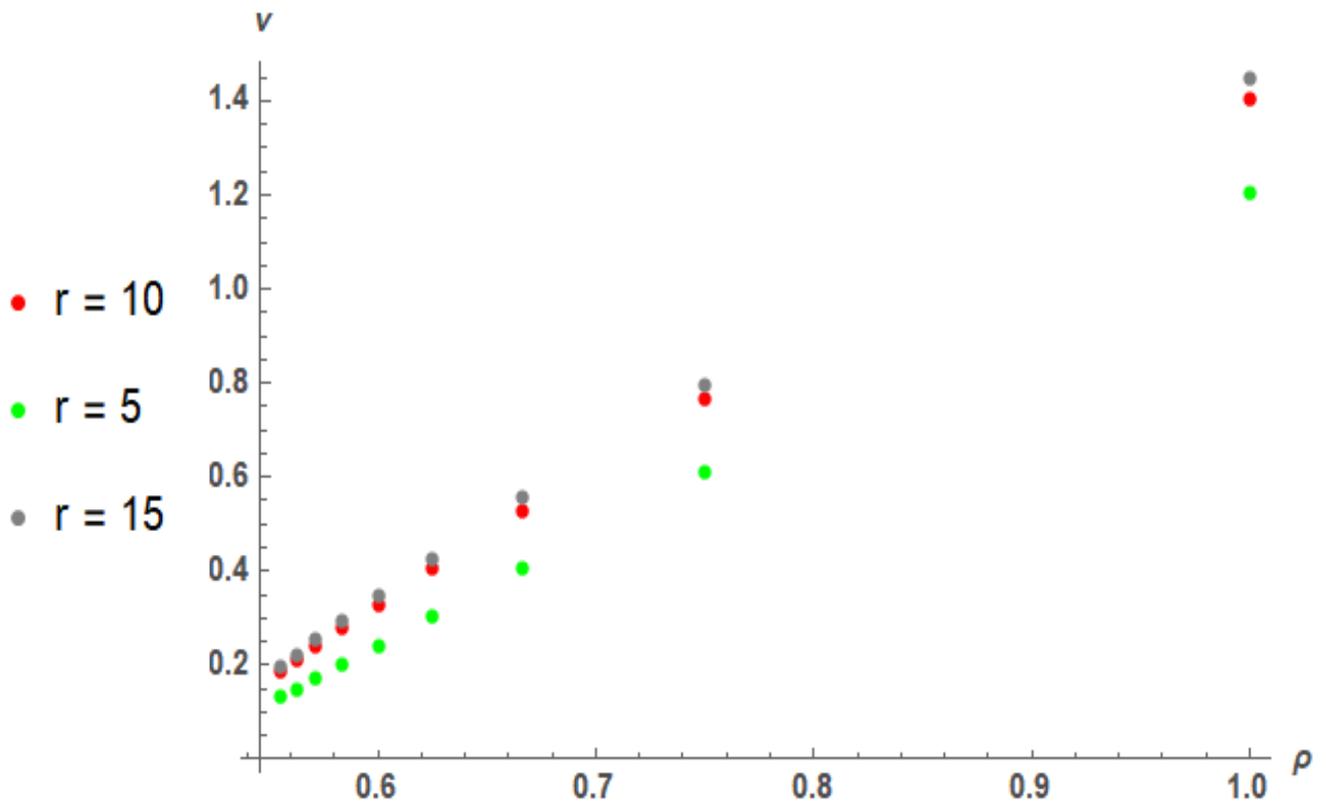


Рис.2.1.3 Зависимость стационарного среднего времени пребывания заявки в СМО от приведенной интенсивности

Среднее время пребывания заявки в системе увеличивается по мере роста приведенной интенсивности. Это связано с тем, что при увеличении интенсивности поступления заявок нагрузка на систему возрастает, и каждая заявка вынуждена дольше ждать своей очереди на обслуживание.

При малом числе фаз обслуживания среднее время пребывания заявки в системе минимально, когда нагрузка на систему низкая. Однако по мере увеличения нагрузки время пребывания растёт значительно быстрее. Это связано с тем, что система не успевает равномерно распределять поступающие заявки, и процесс становится менее эффективным.

При большом числе этапов время пребывания всегда оказывается больше по сравнению с меньшим числом фаз. Это связано с тем, что каждая заявка проходит большее количество этапов обслуживания, что увеличивает её общее время нахождения в системе. Однако рост времени происходит стабильно и без резких

скачков, даже при высокой нагрузке. При приближении системы к предельной нагрузке среднее время пребывания резко возрастает для всех случаев. Однако системы с малым числом фаз показывают меньшие задержки по сравнению с системами с большим числом фаз. Это происходит из-за дополнительного времени, необходимого на прохождение всех этапов обслуживания.

На рис. 2.1.4-2.1.5 представлен частичный пример кода для вычисления стационарных характеристик модели:

```

(* Стационарные характеристики *)

(* определение суммы a *)
a = Sum[t*(i - 1)*z^(i - 1)*p[i], {i, 2, Infinity}];

(* определение функции Pj[j] *)
Pj[j_] :=
  P1*(1 + rho*(1 - z))^(j - 1) - (1 - z)/m*a*
  Sum[(1 + rho*(1 - z))^k, {k, 0, j - 2}];

(* определение функции Pr *)
Pr = (-p*(rho*(1 - z)) + P1/z + (1 - z)/m*a)/(1 + rho*(1 - z));

(* Решение уравнения Pr == Pj[r] относительно P1 *)
P11 = Solve[Pr == Pj[r], P1];

(* определение новой функции P1new[z] *)
P1new[z_] := (
  (
    (p*rho*(1 - z))/(1 + rho*(1 - z)) -
    ((1 - z)*Sum[(i - 1)*t*z^(i - 1)*p[i], {i, 2, Infinity}])/
    (m*(1 + rho*(1 - z))) -
    ((1 - z)*(-1 - rho + (1 + rho*(1 - z))^r + rho*z)*
    Sum[(i - 1)*t*z^(i - 1)*p[i], {i, 2, Infinity}])/
    (m*rho*(-1 + z)*(-1 - rho + rho*z))
  )/
  (
    - (1 + rho*(1 - z))^(-1 + r) + 1/((1 + rho*(1 - z))*z)
  )
);

(* определение функции Pjnew[j] *)
Pjnew[j_] :=
  P1new[z]*(1 + rho*(1 - z))^(j - 1) -
  (1 - z)/m*(a*P1new[z])*
  Sum[(1 + rho*(1 - z))^k, {k, 0, j - 2}];

(* Вычисление суммы Pjnew от j=1 до r *)
SumPjnew = Sum[Pjnew[j], {j, 1, r}];

```

Рис. 2.1.4 Код для вычисления стационарных характеристик модели

```

(* Решение уравнения P == SumPjnew + p относительно P *)
solutionP = Solve[P == SumPjnew + p, P];

(* определение функции Pnew[z] *)
Pnew[z_] := p + Sum[
  (
    ((1 + rho*(1 - z))^(-1 + j))*
    (
      (p*rho*(1 - z))/(1 + rho*(1 - z)) -
      ((1 - z)*Sum[(i - 1)*t*z^(i - 1)*p[i], {i, 2, Infinity}])/
      (m*(1 + rho*(1 - z))) -
      ((1 - z)*(-1 - rho + (1 + rho*(1 - z))^r + rho*z)*
      Sum[(i - 1)*t*z^(i - 1)*p[i], {i, 2, Infinity}])/
      (m*rho*(-1 + z)*(-1 - rho + rho*z))
    )/
    (- (1 + rho*(1 - z))^(-1 + r) + 1/((1 + rho*(1 - z))*z)) -
    ((1 - z)*(-1 - rho + (1 + rho*(1 - z))^j + rho*z)*
    Sum[(i - 1)*t*z^(i - 1)*p[i], {i, 2, Infinity}])/
    (m*rho*(-1 + z)*(-1 - rho + rho*z))
  )/
  (m*rho*(-1 + z)*(-1 - rho + rho*z))
,
{j, 1, r}
];
PreN = D[Pnew[z], z]

```

Рис. 2.1.5 Код для вычисления стационарных характеристик модели

2.2 Нестационарные характеристики одноканальной СМО

В настоящее время достаточно хорошо изучены системы массового обслуживания с простейшим пуассоновским входным потоком заявок и экспоненциальным обслуживанием с постоянными параметрами в стационарном режиме. Такие модели часто используются для анализа и оптимизации производительности

систем, где заявки поступают случайным образом и обслуживаются с постоянной интенсивностью.

Однако, практический интерес представляют более сложные модели, которые являются комбинациями ранее известных моделей СМО. Это могут быть системы с различными типами входящих потоков, разными видами обслуживания или очередью. Такие модели позволяют более точно отражать реальные условия эксплуатации систем массового обслуживания, что особенно важно в прикладных областях, таких как телекоммуникации, транспортные системы и облачные сервисы.

При эксплуатации систем массового обслуживания актуальной является задача нахождения нестационарных характеристик. Основная сложность анализа СМО в нестационарном режиме заключается в необходимости решения больших или даже бесконечных систем дифференциальных уравнений. Это связано с тем, что параметры системы могут меняться во времени, что требует сложных математических моделей для описания их динамики и поведения.

Изучение нестационарных характеристик позволяет более глубоко понять и предсказать поведение систем массового обслуживания в реальных условиях эксплуатации. Это важно для оптимизации работы системы, улучшения её производительности и качества обслуживания.

Как было отмечено ранее, практический интерес имеют более сложные модели, являющиеся комбинациями ранее известных моделей СМО. При эксплуатации систем, работающих по принципу систем массового обслуживания, актуальной является задача нахождения нестационарных характеристик. Основная сложность анализа СМО в нестационарном режиме заключается в решении, как правило, большой или бесконечной системы дифференциальных уравнений [55].

$$\begin{aligned}\frac{dp_0(t)}{dt} &= -\lambda p_0(t) + \mu p_{11}(t), \\ \frac{dp_{1r}(t)}{dt} &= \lambda p_0(t) + \theta p_{2r}(t) - (\lambda + \mu)p_{1r}(t) + \mu p_{21}(t)\end{aligned}\quad (2.14)$$

$$\frac{dp_{ir}(t)}{dt} = -(\lambda + \mu)p_{ir}(t) + \lambda p_{i-1,r}(t) + \mu p_{i+1,1}(t) + i\theta p_{i+1,r}(t) - (i-1)\theta p_{ir}(t), i > 1$$

$$\frac{dp_{ij}(t)}{dt} = -(\lambda + \mu)p_{ij}(t) + \lambda p_{i-1,j}(t) + \mu p_{i,j+1}(t) + i\theta p_{i+1,j}(t) - (i-1)\theta p_{ij}(t), i > 1, 1 < j < r-1,$$

$$\frac{dp_{1j}(t)}{dt} = -(\lambda + \mu)p_{1j}(t) + \mu p_{1,j+1}(t) + \theta_{2j}, 1 < j < r-1.$$

Нестационарные характеристики системы находятся из системы (2.14) с помощью введения производящей функции $P_j(z, t) = \sum_{i=1}^{\infty} p_{ij}(t)z^i$.

Умножая уравнения для $p_{ij}(t)$ из (1) на z^i и суммируя, получим

$$\frac{dP_j(z, t)}{dt} = -(\lambda + \mu)P_j(z, t) + \lambda z P_j(z, t) + \mu P_{j+1}(z, t) + (1-z) \sum_{i=2}^{\infty} (i-1)z^{i-1} \theta p_{ij}(t), 0 < j < r-2,$$

$$\frac{dP_r(z, t)}{dt} + \frac{dp_0(t)}{dt} = -\lambda(P_r(z, t) + p_0(t)) - \mu P_r(z, t) + \lambda z(P_r(z, t) + p_0(t)) + \frac{\mu}{z} P_1(z, t) + (1-z) \sum_{i=2}^{\infty} (i-1)z^{i-1} \theta p_{ir}.$$

С использованием одной из тауберовых теорем получаем систему:

$$\frac{dP_j(z, t)}{dt} = -(\lambda + \mu)P_j(z, t) + \lambda z P_j(z, t) + \mu P_{j+1}(z, t), 0 < j < r-2,$$

$$\frac{dP_r(z, t)}{dt} + \frac{dp_0(t)}{dt} = -\lambda(P_r(z, t) + p_0(t)) - \mu P_r(z, t) + \lambda z(P_r(z, t) + p_0(t)) + \frac{\mu}{z} P_1(z, t) \quad (2.15)$$

Просуммировав уравнения (2.14) по i при фиксированном j , получим

$$\sum_i p_{i1}(t) = \sum_i p_{i2}(t) = \dots = \sum_i p_{ir}(t) \quad (2.16)$$

Просуммировав уравнения (2.14) по j при фиксированном i , получим

$$\sum_{i=1}^{\infty} \sum_{j=1}^r \frac{dp_{ij}}{dt} = \sum_{i=1}^{\infty} \mu p_{i+1,1}(t) + \sum_{i=1}^{\infty} \sum_{j=1}^r i\theta p_{i+1,j}(t) - \lambda \sum_{i=1}^{\infty} \sum_{j=1}^r p_{ij}(t)$$

Тогда, с учетом (2.16) и условия нормировки $p_0(t) + \sum_i \sum_j p_{ij}(t) = 1$ получаем следующее равенство

$$r\lambda = \theta r Q(t) + (1 - p_0(t))\mu, \quad (2.17)$$

Из (2.2.4) следует, что

$$p_0(t) = 1 - \frac{r(\lambda - \theta Q(t))}{\mu}. \quad (2.18)$$

Система дифференциальных уравнений (2.15) приводится к системе линейных алгебраических уравнений с помощью преобразования Лапласа:

$$\begin{aligned} \pi_j(z, s) &= \int_0^\infty e^{-st} P_j(z, t) dt, \\ \pi_0(s) &= \int_0^\infty e^{-st} p_0(t) dt : \\ s\pi_j(z, s) &= \lambda z \pi_j(z, s) + \mu \pi_{j+1}(z, s) - (\lambda + \mu) \pi_j(z, s) \\ s\pi_r(z, s) + s\pi_0(s) - 1 &= -\lambda(\pi_r(z, s) + \pi_0(s)) - \mu \pi_r(z, s) + \lambda z(\pi_r(z, s) + \pi_0(s)) \\ &+ \frac{\mu}{z} \pi_1(z, s) \end{aligned}$$

Тогда

$$\begin{aligned} \pi_j(z, s) &= \pi_1(z, s) * \left(\frac{s - \lambda z + \lambda + \mu}{\mu} \right)^{j-1}; \\ \pi_r(z, s) &= \pi_0(s) * \frac{\lambda * (z-1) - s}{s - \lambda * z + \lambda + \mu} + \left(\frac{1}{s + \lambda * (1-z) + \mu} \right) * \left(\frac{\mu}{z} * \pi_1(z, s) + 1 \right). \end{aligned} \quad (2.19)$$

Для нахождения $\pi_j(z, s)$ выразим $\pi_1(z, s)$ из второго уравнения (6):

$$\pi_1(z, s) = \frac{z - z(s + \lambda * (1 - z))\pi_0(s)}{\mu(-1 + z(\frac{s + \lambda * (1 - z) + \mu}{\mu})^r)}$$

Просуммировав уравнение для $\pi_j(z, s)$ по j , в силу аддитивности преобразования Лапласа, получим преобразование Лапласа $\pi(z, s)$ производящей функции $P(z, t) = \sum_j P_j(z, t)$:

$$\pi(z, s) = \frac{z - z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r - (-1 + z)(\lambda + s - \lambda z)\pi_0(s)}{(-s + \lambda(-1 + z))(-1 + z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r)}.$$

Представляя выражение для $\pi(z, s)$ в виде суммы дробей, найдем $P(z)$ для каждого из слагаемых в силу свойства линейности преобразования Лапласа

$$\pi(z, s) = \frac{1}{-\lambda - s + \lambda z} + \frac{-1 + z}{(-\lambda - s + \lambda z)(-1 + z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r)} + \frac{(-1 + z)\pi_0(s)}{-1 + z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r}. \quad (2.20)$$

Непосредственное применение формулы обращения преобразования Лапласа является затруднительным, поэтому для нахождения $P(z)$ применим теорему разложения [56].

Первые два слагаемых (2.20) могут быть обращены с использованием таблицы основных преобразований Лапласа и второй теоремы разложения. Для обращения третьего слагаемого воспользуемся теоремой Бореля формулой Дюамеля. Теорема Бореля и формулы Дюамеля дают дополнительные возможности нахождения оригинала по изображению [57].

Теорема Бореля

Изображение свёртки двух оригиналов равно произведению изображений свёртываемых оригиналов.

В развёрнутом виде формулы Дюамеля имеют вид (вывод и описание формул можно найти в литературе, например в [58-62])

$$sF(s)G(s) = f(0)g(t) + \int_0^t f'(\tau) \cdot g(t - \tau) d\tau = f(0)g(t) + \int_0^t f'(t - \tau) \cdot g(\tau) d\tau,$$

$$sF(s)G(s) = g(0)f(t) + \int_0^t f(\tau) \cdot g'(t - \tau) d\tau = g(0)f(t) + \int_0^t f(t - \tau) \cdot g'(\tau) d\tau.$$

Каждая из этих формул называется интегралом Дюамеля.

Пусть
$$\pi_1(z, s) = \frac{1}{-\lambda - s + \lambda z} + \frac{-1 + z}{(-\lambda - s + \lambda z)(-1 + z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r)}, \quad \pi_2(z, s) = \frac{(-1 + z)\pi_0(s)}{-1 + z(\frac{\lambda + \mu + s - \lambda z}{\mu})^r}.$$

Обращение преобразования Лапласа, полученное с помощью теоремы обращения [74] для функции $\pi_1(z, s)$ имеет вид

$$L^{-1}[\pi_1(z, s)] = -e^{\lambda t(-1+z)} + e^{\lambda t(-1+z)} - \frac{e^{t(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))(-1+z)}}{r(-1+(\frac{1}{z})^{\frac{1}{r}})z^2}.$$

Для нахождения $L^{-1}[\pi_2(z, s)]$ воспользуемся формулой Дюамеля, приведенной выше.

$$L^{-1}[\pi_2(z, s)] =$$

$$e^{(t-\tau)\left(\mu\left(-1+\left(\frac{1}{z}\right)^{\frac{1}{r}}\right)+\lambda(-1+z)\right)} \mu\left(-\mu\left(-1+\left(\frac{1}{z}\right)^{\frac{1}{r}}\right)-\lambda(-1+z)\right)(-1+z)p_0(\tau) - \int_0^t \frac{d\tau + \left(\frac{\mu(-1+z)}{r(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))z^2} + \frac{-1+z}{-1+z(\frac{\lambda+\mu-\lambda z}{\mu})r}\right)p_0(t)}{r(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))z^2}$$

Таким образом, выражение для $P(z, t)$ будет иметь вид

$$P(z, t) = L^{-1}[\pi_1(z, s)] + L^{-1}[\pi_2(z, s)] = -e^{\lambda t(-1+z)} + e^{\lambda t(-1+z)} -$$

$$\frac{e^{t(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))(-1+z)}}{r(-1+(\frac{1}{z})^{\frac{1}{r}})z^2} -$$

$$e^{(t-\tau)\left(\mu\left(-1+\left(\frac{1}{z}\right)^{\frac{1}{r}}\right)+\lambda(-1+z)\right)} \mu\left(-\mu\left(-1+\left(\frac{1}{z}\right)^{\frac{1}{r}}\right)-\lambda(-1+z)\right)(-1+z)p_0(\tau) \int_0^t \frac{d\tau + \left(\frac{\mu(-1+z)}{r(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))z^2} + \frac{-1+z}{-1+z(\frac{\lambda+\mu-\lambda z}{\mu})r}\right)p_0(t)}{r(\mu(-1+(\frac{1}{z})^{\frac{1}{r}})+\lambda(-1+z))z^2}$$

Чтобы однозначно определить $P(z, t)$ необходимо найти $Q(t)$ и, с использованием (2.18) выразить $p_0(t)$.

С помощью производящей функции найдем нестационарное среднее число заявок в системе N :

$$N(t) = P'(z, t) = \frac{\mu \int_0^t p_0(\tau) d\tau}{r} + \frac{-1+r+2\lambda r t-2\mu t}{2r} - \frac{(\lambda(-1+r)r+\mu(-1+3r))p_0(t)}{2r(-\mu+\lambda r)}.$$

Тогда

$$Q(t) = N(t) - (1 - p_0(t)) = -1 + \frac{\mu \int_0^t p_0(\tau) d\tau}{r} + \frac{-1+r+2\lambda r t - 2\mu t}{2r} + p_0(t) - \frac{(\lambda(-1+r)r + \mu(-1+3r))p_0(t)}{2r(-\mu + \lambda r)}.$$

С учетом (2.2.5) получаем $p_0(t)$:

$$p_0(t) = \frac{(\mu - \lambda r)(\theta - 2\mu + \theta r + 2\lambda r - 2\theta \lambda r t - 2\theta \mu \int_0^t p_0(\tau) d\tau + 2\theta \mu t)}{-\theta \mu - 2\mu^2 - \theta \lambda r + 5\theta \mu r + 2\lambda \mu r - \theta \lambda r^2}. \quad (2.21)$$

Наконец, получим выражение для $p_0(t)$ проинтегрировав (2.21) и решив дифференциальное уравнение относительно $p_0(t)$ с начальным условием $p_0(0) = 1$.

$$p_0'(t) = \frac{(\mu - \lambda r)(-2\theta \lambda r + 2\theta \mu - 2\theta \mu p_0(t))}{-\theta \mu - 2\mu^2 - \theta \lambda r + 5\theta \mu r + 2\lambda \mu r - \theta \lambda r^2},$$

$$p_0(t) = 1 + \frac{2\theta \mu (\mu - \lambda r) t}{(-1 + e^{\frac{2\theta \mu (\mu - \lambda r) t}{-\theta \mu - 2\mu^2 - \theta \lambda r + 5\theta \mu r + 2\lambda \mu r - \theta \lambda r^2}}) \lambda r}. \quad (2.22)$$

После однозначного определения $p_0(t)$ с помощью производящей функции выводятся остальные нестационарные характеристики модели.

Так, подставляя полученное выражение для $p_0(t)$ в (2.22) получим

$$Q(t) = \frac{p_0(t)\lambda}{\theta} + \frac{\mu - 2\mu r}{r(-\mu + \lambda r)},$$

$$N(t) = \frac{p_0(t)\lambda}{\theta} + \frac{p_0(t)\lambda^2 r^3 - p_0(t)\lambda \mu r^2 + \mu^2(-1 + 2r)}{\mu r(\mu - \lambda r)}.$$

Теорема

При $t \rightarrow \infty$ и выполнении условия $0 < \lambda < \frac{\mu}{r}$ и $0 < \theta < \frac{2\mu^2 - 2\lambda \mu r}{-\mu - \lambda r + 5\mu r - \lambda r^2}$ значение $p_0(t)$ сходится к стационарному значению p_0 для системы M/E_r/1 без ограничений на время пребывания заявки в очереди.

Доказательство:

Если в формуле (2.22) устремить значение t к бесконечности, то получим, что

$$1 + \frac{2\theta \mu (\mu - \lambda r) t}{(-1 + e^{\frac{2\theta \mu (\mu - \lambda r) t}{-\theta \mu - 2\mu^2 - \theta \lambda r + 5\theta \mu r + 2\lambda \mu r - \theta \lambda r^2}}) \lambda r} = 1 - \frac{\lambda r}{\mu} = p_{0, \text{стац}}$$

Данное значение, очевидно, будет справедливо только в случае, когда $e^{\frac{2\theta\mu(\mu-\lambda r)t}{2\mu(\mu-\lambda r)+\theta(\mu-5\mu r+\lambda r(1+r))}}$ будет стремиться к 0 при $t \rightarrow \infty$. Это в свою очередь возможно только при $\frac{2\theta\mu(\mu-\lambda r)t}{2\mu(\mu-\lambda r)+\theta(\mu-5\mu r+\lambda r(1+r))} < 0$. Неравенство $\frac{2\theta\mu(\mu-\lambda r)t}{2\mu(\mu-\lambda r)+\theta(\mu-5\mu r+\lambda r(1+r))} < 0$ выполняется в том случае, когда $0 < \lambda < \frac{\mu}{r}$ и $0 < \theta < \frac{2\mu^2-2\lambda\mu r}{-\mu-\lambda r+5\mu r-\lambda r^2}$. Теорема доказана.

На рис. 2.2.1-2.2.4 приведен частичный пример кода для расчета нестационарных характеристик согласно формулам, представленным в настоящей главе:

```

Pj[s_] = P1* ((s - l*z + l + m)/m)^(j - 1);
Pr = po[s]*(l*(z - 1) - s)/(s - l*z + l +
      m) + (1/(s + l*(1 - z) + m))*(m/z*P1 + 1);
Solve[Pr == P1*((s - l*z + l + m)/m)^(r - 1), P1 ];
P1 = (-1/(m + s + l (1 - z))) - ((-s + l (-1 + z)) po[s])/((
      l + m + s - l z))/
      m/((m + s + l (1 - z)) z) - ((l + m + s - l z)/m)^(-1 + r));
P[s_] = Sum[Pj[s], {j, 1, r}] + po[s]
A[s_] = (
      z - z ((l + m + s - l z)/m)^
      r)/((-1 - s + l z) (-1 + z ((l + m + s - l z)/m)^r));
Solve[-1 - s + l z == 0, s]
A1[s_] = -1/(-1 - s + l z);
A2[s_] = (-1 + z)/((-1 - s + l z) (-1 + z ((l + m + s - l z)/m)^r));
InverseLaplaceTransform[A1[s], s, t]
D[(-1 - s + l z) (-1 + z ((l + m + s - l z)/m)^r), s]
Solve[(-1 - s + l z) (-1 + z ((l + m + s - l z)/m)^r == 0, s];
A2Chis1 = -1 + z;
A2Chis2 = -1 + z;
A2Znam1 =
      1 - z ((l + m + (-1 + l z) - l z)/m)^r + (
      r z ((l + m + (-1 + l z) - l z)/m)^(-1 +
      r) (-1 - (-1 + l z) + l z))/m;
A2Znam2 =
      1 - z ((l + m + (-1 - m + m (1/z)^(1/r) + l z) - l z)/m)^r + (
      r z ((l + m + (-1 - m + m (1/z)^(1/r) + l z) - l z)/m)^(-1 +
      r) (-1 - (-1 - m + m (1/z)^(1/r) + l z) + l z))/m;
A2[z_, t_] =
      A2Chis1/A2Znam1*e^((-1 + l z)*t) +
      A2Chis2/A2Znam2*e^((-1 - m + m (1/z)^(1/r) + l z)*t)
A[z_, t_] = A2[z, t] + E^(-t (l - l z))
E^(-t (l - l z)) + (e^(t (-l + l z)) (-1 + z))/(1 - z) + (
      e^(t (-1 - m + m (1/z)^(1/r) + l z)) (-1 + z))/
      1 + (r (m - m (1/z)^(1/r)) z* z)/m - 1)

```

Рис. 2.2.1 Код для расчета нестационарных характеристик модели (начало)

```

Simplify[
  E^(-t (1 - 1 z)) + (e^(t (-1 + 1 z)) (-1 + z))/(1 - z) + (
    e^(t (-1 - m + m (1/z)^(1/r) + 1 z)) m (-1 + z))/(
    r (m - m (1/z)^(1/r)) z^2)]
B[s_] = (-(-1 + z) (1 + s - 1 z) po[
  s])/((-s + 1 (-1 + z)) (-1 + z ((1 + m + s - 1 z)/m)^r));
Solve[(-1 + z ((1 + m + s - 1 z)/m)^r) == 0, s]
Bnew[s_] = (-1 + z)/(s*(-1 + z ((1 + m + s - 1 z)/m)^r))
Chisl1 = -1 + z;
Chisl2 = -1 + z;
Solve[s (-1 + z ((1 + m + s - 1 z)/m)^r) == 0, s]
D[s (-1 + z ((1 + m + s - 1 z)/m)^r), s]
Znam1 = -1 + (r 0 z ((1 + m + 0 - 1 z)/m)^(-1 + r))/m +
  z ((1 + m + 0 - 1 z)/m)^r;
Znam2 = -1 + (
  r (-1 - m + m (1/z)^(1/r) + 1 z) z ((
    1 + m + (-1 - m + m (1/z)^(1/r) + 1 z) - 1 z)/m)^(-1 + r))/m +
  z ((1 + m + (-1 - m + m (1/z)^(1/r) + 1 z) - 1 z)/m)^r;
Bnew[z_, t_] =
  Chisl2/Znam2*e^((-1 - m + m (1/z)^(1/r) + 1 z)*t) +
  Chisl1/Znam1*e^(0*t)
B2[z_, t_] = (e^(
  t (m (-1 + (1/z)^(1/r)) + 1 (-1 + z))) (-1 +
  z))/(-1 + (r (m (-1 + (1/z)^(1/r)) + 1 (-1 + z)) z* z)/m +
  1) + (-1 + z)/(-1 + z ((1 + m - 1 z)/m)^r)
B2[z, 0]

```

Рис. 2.2.2 Код для расчета нестационарных характеристик модели (продолжение)

```

(* Воспользуемся формулой Дюамеля *)

(* Определение производной функции B2 по переменной x *)
ForInt = D[B2[z, t - x], x];

(* Вычисление Bb *)
Bb = B2[z, 0]*p0[t] - Integrate[p0[x]*ForInt, {x, 0, t}];

(* Определение функции Bfinal *)
Bfinal[z_, t_] := -Integrate[
  Exp[(t - x)*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))] *
  (m*(-1 + (1/z)^(1/r)) - l*(-1 + z)) *
  Log[E]*p0[x],
  {x, 0, t}
] + ((m*(-1 + z))/(r*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))*z^2) +
  (-1 + z)/(-1 + z*((1 + m - l*z)/m)^r))*p0[t];

(* Определение функции Pfinal *)
Pfinal[z_, t_] := Bfinal[z, t] + A[z, t];

(* Определение функций Q1, Q2, Q3, Q4 *)
Q1[z_] :=
  Exp[-t*(1 - l*z)] +
  (Exp[t*(-l + l*z)]*(-1 + z))/(1 - z) +
  (Exp[t*(-l - m + m*(1/z)^(1/r) + l*z)]*(-1 + z))/
  (1 + (r*(m - m*(1/z)^(1/r))*((1/z)^(1/r))^(1 - r)*z)/m -
  ((1/z)^(1/r))^r*z);

Q2[z_] :=
  -((m*(-m*(-1 + (1/z)^(1/r)) - l*(-1 + z))*(-1 + z))/(r*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))*z^2)

Q3[z_] := Integrate[
  Exp[(t - x)*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))] * p0[x],
  {x, 0, t}
];

Q4[z_] :=
  ((m*(-1 + z))/(r*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))*z^2) +
  (-1 + z)/(-1 + z*((1 + m - l*z)/m)^r))*p0[t];

```

Рис. 2.2.3 Код для расчета нестационарных характеристик модели (продолжение)

```

(* Вычисление пределов *)
limit1 = Limit[D[Exp[(t - x)*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))], z], z -> 1];
limit2 = Limit[Exp[(t - x)*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))] * p0[x], z -> 1];

(* Вычисление производных и интегралов при z -> 1 *)
a1 = Limit[D[Q1[z], z], z -> 1];
a2 = Limit[D[Q2[z], z], z -> 1];
a3 = Integrate[(1 - m/r)*(t - x)*p0[x], {x, 0, t}];
a4 = Integrate[p0[x], {x, 0, t}];
a5 = Limit[D[((m*(-1 + z))/(r*(m*(-1 + (1/z)^(1/r)) + l*(-1 + z))*z^2) +
(-1 + z)/(-1 + z*((1 + m - l*z)/m)^r)), z], z -> 1]*p0[t];

(* Вычисление Nn и Qq *)
Nn = a1 + a2*a4 + a3*Limit[Q2[z], z -> 1] + a5;
Qq = Nn - (1 - p0[t]);

(* Решение уравнения относительно p0[t] *)
solution = solve[Qq == (r*l - m*(1 - p0[t]))/(r*F), p0[t]];

(* Вычисление производной по времени *)
derivative = D[
((m - l*r)*(F - 2*m + F*r + 2*l*r - 2*F*l*r*t -
2*F*m*Integrate[p0[x], {x, 0, t}] + 2*F*m*t*Log[E]))/
(-F*m - 2*m^2 - F*l*r + 5*F*m*r + 2*l*m*r - F*l*r^2),
t
];

(* Решение дифференциального уравнения *)
solutionDSolve = DSolveValue[
{
p0'[t] == ((m - l*r)*(2*F*m - 2*F*l*r - 2*F*m*p0[t]))/
(-F*m - 2*m^2 - F*l*r + 5*F*m*r + 2*l*m*r - F*l*r^2),
p0[0] == 1
},
p0[t],
t
];

```

Рис. 2.2.4 Код для расчета нестационарных характеристик модели (конец)

2.3 Имитационная модель СМО

В данной работе в качестве инструмента имитационного моделирования применялась среда AnyLogic. Данный выбор обусловлен не только необходимостью проверки достоверности результатов, полученных на этапе аналитического моделирования, но и потребностью в адаптивном исследовательском инструменте, способном учитывать широкий спектр влияющих факторов.

Для достижения этой цели была разработана имитационная модель открытой одноканальной системы массового обслуживания (СМО) смешанного типа, в которой входной поток заявок подчиняется пуассоновскому распределению, а время обслуживания представлено эрланговским законом. Использование эрланговского распределения позволяет учесть многофазный характер обслуживания, а также неоднородность его параметров. В результате, такая модель может отражать более широкий спектр реальных ситуаций, возникающих в инженерных, транспортных, телекоммуникационных и других сложных системах.

Имитационное моделирование обладает рядом преимуществ по сравнению с аналитическим подходом. Во-первых, оно даёт возможность более детально учесть стохастический характер входного потока заявок и длительности обслуживания, позволяя проследить влияние случайных факторов на поведение системы. Во-вторых, имитационная модель обеспечивает гибкость при исследовании различных режимов работы: можно рассмотреть нестационарные условия, резко меняющиеся параметры, введение приоритетных обслуживаний или управляемого перенаправления заявок. Это особенно актуально при анализе сложных систем, для которых получение точного аналитического решения затруднено или невозможно. В-третьих, имитационный подход упрощает процесс валидации и верификации моделей, поскольку имитация позволяет непосредственно проследить процессы обслуживания, а также сравнивать результаты с натурными или лабораторными данными [64-67].

В контексте новых исследований имитационная модель, построенная в AnyLogic, может быть расширена и дополнена, позволяя исследовать новые ас-

пекты функционирования СМО. Например, можно рассмотреть различные механизмы адаптации числа фаз обслуживания в реальном времени, применение динамических политик перенаправления входящих заявок, а также оценить чувствительность системы к изменению параметров входного потока или интенсивности обслуживания при появлении периодических или сезонных флуктуаций нагрузки. Кроме того, возможно интегрировать имитационную модель с методами машинного обучения для предиктивного управления ресурсами и оптимизации сервисных параметров.

Таким образом, применение AnyLogic для имитационного моделирования открытых одноканальных СМО с эрланговским распределением времени обслуживания не только подтверждает и дополняет результаты, полученные аналитическим путём, но и открывает новые возможности для междисциплинарных исследований. Такой подход обеспечивает более глубокое понимание поведения системы в динамических условиях, расширяя пространство экспериментальных сценариев и закладывая основу для дальнейших исследований в области оптимизации и адаптивного управления системами массового обслуживания.

Пример кода для настройки входного потока и времени обслуживания в AnyLogic представлен на рис. 2.3.1.

Пример кода для AnyLogic, который создает имитационную модель открытой одноканальной системы массового обслуживания (СМО) с использованием распределения Эрланга для времени обслуживания приведен на рис. 2.3.2.

```
// настройка входного потока
Source source = new Source();
source.setArrivalProcess(new ProcessArrival(ExponentialDistribution(1.0)));
// настройка времени обслуживания
Service service = new Service();
service.setServiceTime(new ErlangDistribution(2, 3));
// добавление элементов в модель
add(source);
add(service);
```

Рис. 2.3.1. Код для настройки входного потока и времени обслуживания в AnyLogic

```

// Определяем параметры системы
double lambda = 2.0; // интенсивность входного потока
double mu = 1.0;     // интенсивность обслуживания
int numServers = 1; // количество серверов

// Создаем входной поток
Source source = new Source();
source.setArrivalRate(lambda); // Задаем интенсивность потока

// Создаем блок обслуживания
Service service = new Service();
service.setServiceTimeDistribution(new Erlang(2, mu)); // Эрланговское распределение

// Создаем сервер
Queue queue = new Queue();
queue.setCapacity(100); // Вместимость очереди
queue.setQueueType(QueueType.FCFS); // Первым пришел – первым обслужен

// Соединяем компоненты
source.addOutput(queue);
queue.addOutput(service);

// Создаем статистику для наблюдения
Statistics queueLengthStats = new Statistics(queue.getQueueLength());
Statistics waitingTimeStats = new Statistics(queue.getWaitingTime());

// Запускаем симуляцию
run();

```

Рис. 2.3.2 Код имитационной модели СМО с эрланговски распределенным временем обслуживания

Для моделирования задержки заявки в обслуживающем устройстве в AnyLogic можно использовать следующие элементы и настройки. На рис. 2.3.3 приведен пример кода, который создает систему с задержкой для заявки, используя блоки Source, Queue, и Service, а также добавляет мониторинг времени задержки:

```

// определяем параметры системы
double lambda = 2.0; // интенсивность входного потока
double mu = 1.0; // интенсивность обслуживания
int numServers = 1; // количество серверов

// Создаем входной поток
Source source = new Source();
source.setArrivalRate(lambda); // задаем интенсивность по-тока

// Создаем очередь для хранения заявок
Queue queue = new Queue();
queue.setCapacity(100); // вместимость очереди
queue.setQueueType(QueueType.FCFs); // первым пришел – первым обслужен

// Создаем блок обслуживания с задержкой
Service service = new Service();
service.setServiceTimeDistribution(new Erlang(2, mu)); // Эрланговское распределение
service.setNumberOfServers(numServers); // Задаем количество серверов

// добавляем мониторинг времени задержки
Statistics delayStats = new Statistics(queue.getAveragewaitingTime()); // Среднее время
Statistics totalDelayStats = new Statistics(queue.getTotalwaitingTime()); // общее время

// Соединяем компоненты
source.addOutput(queue);
queue.addOutput(service);

// Запускаем симуляцию
run();

```

Рис. 2.3.3 Код для создания системы с задержкой для заявки

Таким образом, использование AnyLogic для имитационного моделирования открытой одноканальной СМО с эрланговски распределенным временем обслуживания позволяет подтвердить или дополнить результаты, полученные аналитическим путем, и обеспечить более глубокое понимание работы системы в различных условиях эксплуатации.

Входной поток по умолчанию в Anylogic является простейшим. Время обслуживания распределено по закону Эрланга. Для каждого элемента требуется ввести значения параметров во вкладке «Свойства», в соответствии с условиями задачи. Структурная модель рассматриваемой системы представлена на рисунке 2.3.4. Основные параметры функционирования системы представлены на рисунках 2.3.5-2.3.7.

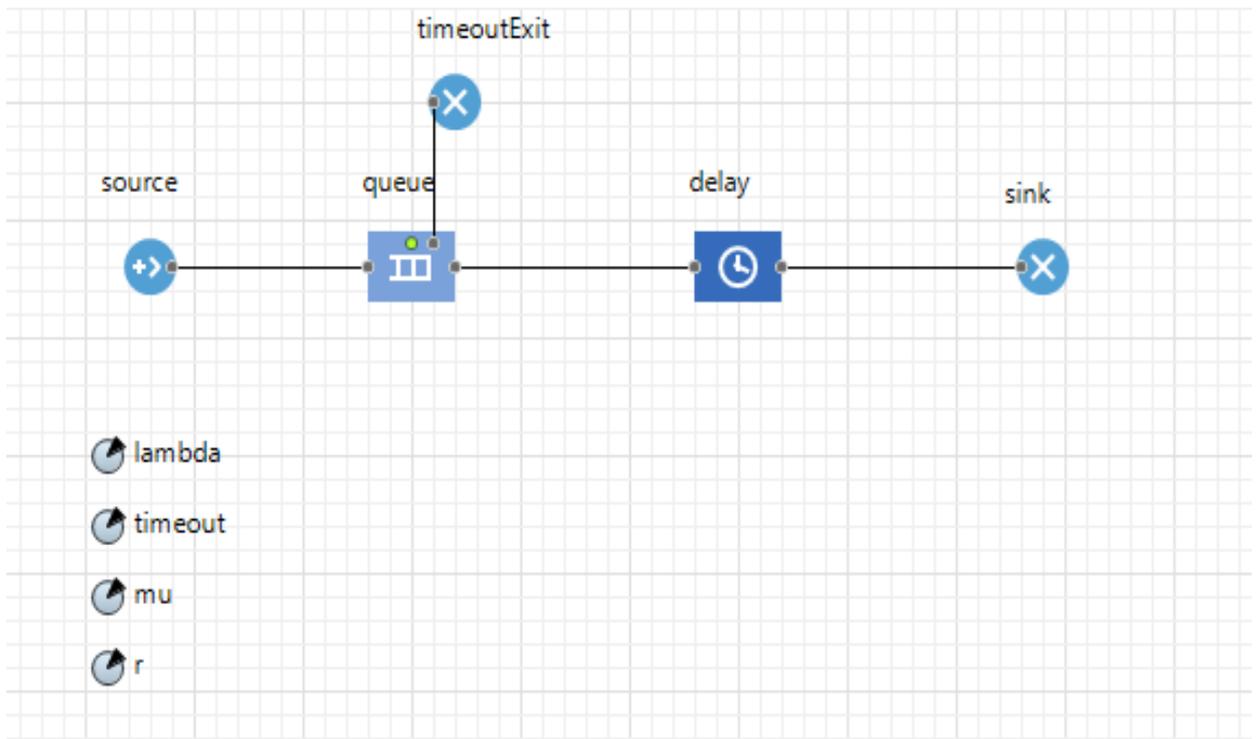


Рис.2.3.4. Структурная модель системы

source - Source

Имя: Отображать имя

Исключить

Прибывают согласно: ▾

Интенсивность прибытия: ▾

Считать параметры агентов из БД:

За 1 раз создается несколько агентов:

Ограниченное кол-во прибытий:

Местоположение прибытия: ▾

Рис.2.3.5 Задание входного потока

queue - Queue

Имя: Отображать имя

Исключить

Максимальная вместимость:

Место агентов:

▼ Специфические

Очередь:

Разрешить уход по таймауту:

Таймаут:

Разрешить вытеснение:

Вернуть агента в исходную точку:

Включить сбор статистики:

Рис.2.3.6 Определение «нетерпеливых» заявок

🕒 delay - Delay

Имя: Отображать имя

Исключить

Тип задержки: Определенное время
 До вызова функции stopDelay()

Время задержки:

Вместимость:

Максимальная вместимость:

Место агентов:

Рис.2.3.7 Задержка заявки в обслуживающем устройстве

Стоит отметить, что во всех моделях используются блоки для сбора статистики и блоки-файлы для сохранения статистики в файле.

2.4 Разработка метамоделей системы массового обслуживания с использованием ансамблевых методов машинного обучения

Одним из перспективных направлений применения машинного обучения к исследуемой модели является построение аппроксимирующей метамоделей. Данная мета модель позволит по входным параметрам системы (таким как интенсивность входного потока заявок, число фаз обслуживания, ограничения на время ожидания и другие характеристики) оперативно оценивать её производственные показатели (среднее время ожидания, длину очереди и так далее.).

Принципиальная идея заключается в том, что вместо непосредственного и многократного запуска вычислительно затратных имитационных экспериментов можно один раз сформировать выборку данных, отражающую поведение системы при различных конфигурациях параметров. Далее с помощью методов машинного обучения на этой обучающей выборке создаётся мета модель. Она служит своего рода «суррогатной» функцией: по заданным значениям входных параметров, вместо полного имитационного эксперимента, мета модель быстро выдаёт оценку ключевых характеристик системы с приемлемой точностью.

Данный подход существенно ускоряет процесс анализа, поскольку разовые затраты на формирование обучающей выборки и построение модели компенсируются множественной экономией времени при дальнейших расчётах. В результате появляется возможность оперативно менять параметры входного потока, число фаз обслуживания или интенсивность обслуживания и мгновенно получать оценки производительности системы, избегая проведения длительных имитационных экспериментов для каждого отдельного случая. Такая мета модель облегчает проведение параметрической оптимизации, повышает эффективность исследований и позволяет более гибко реагировать на изменяющиеся условия эксплуатации системы.

Аналитический подход

Рассмотрим модель открытой одноканальной системы массового обслуживания (СМО) с многофазовым обслуживанием и ограничениями по времени ожидания. Пусть входные параметры модели:

λ – интенсивность входного пуассоновского потока заявок;

r – число фаз обслуживания (целочисленный параметр);

θ – параметр, связанный с ограничением по времени ожидания

На выходе нас интересуют производственные характеристики системы, такие как:

Q – среднее число заявок в очереди;

W – среднее время ожидания;

Аналитическое решение для подобных СМО зачастую трудно получить в замкнутом виде. Вместо этого можно применить имитационное моделирование (например, с помощью AnyLogic или другой платформы). Предположим, что у нас есть процедура имитации, возвращающая оценки Q , W и других характеристик для заданных λ , r , θ .

Цель: построить метамодель \hat{f} , приближающую отображение:

$$(\lambda, r, \theta) \mapsto (Q, W)$$

Для построения метамодели:

1. **Определяется пространство параметров:**

$$\lambda \in [\lambda_{min}, \lambda_{max}], r \in \{r_{min}, \dots, r_{max}\}, \theta \in [\theta_{min}, \theta_{max}]$$

2. **Генерируется выборка точек:**

$$(\lambda_i, r_i, \theta_i), i = 1, 2, \dots, N$$

покрывающую данный диапазон параметров.

3. **Для каждой точки проводится имитационное моделирование с последующим получением оценки характеристик системы:**

$$(Q_i, W_i, \dots)$$

Это формирует обучающую выборку.

4. **Применяются методы машинного обучения** (например, случайный лес или градиентный бустинг) с целью обучения модели (ансамбля моделей):

$$\hat{f}: (\lambda, r, \theta) \rightarrow (\hat{Q}, \hat{W}, \dots)$$

После обучения метамоделю \hat{f} позволяет по заданным λ , r , θ мгновенно оценивать ключевые показатели без проведения имитационного моделирования. Таким образом, метамоделю аппроксимирует многомерное отображение, снижает вычислительные затраты и упрощает выполнение оптимизационных или сценарных анализов.

В данном примере мы рассмотрим процесс построения метамоделю для открытой одноканальной системы массового обслуживания (СМО) с многофазовым обслуживанием и ограничениями по времени ожидания. Метамоделю будет аппроксимировать зависимость выходных характеристик системы (Q и W) от входных параметров (λ , r , θ) с использованием ансамблевых методов машинного обучения, таких как **Градиентный бустинг** и **AdaBoost**.

В данном случае предполагается, что данные получены из имитационных экспериментов и сохранены в CSV-файле *smo_simulation_data.csv*

```
# Загрузка данных
data = pd.read_csv('smo_simulation_data.csv')

# Проверка наличия пропущенных значений
print(data.isnull().sum())

# Заполнение или удаление пропущенных значений, если необходимо
data = data.dropna()

# Разделение данных на признаки и целевые переменные
x = data[['lambda', 'r', 'theta']]
y_Q = data['Q']
y_W = data['W']

# Разделение на обучающую и тестовую выборки
x_train, x_test, yQ_train, yQ_test, yW_train, yW_test = train_test_split(
    x, y_Q, y_W, test_size=0.2, random_state=42
)

print(f"Размер обучающей выборки: {x_train.shape}")
print(f"Размер тестовой выборки: {x_test.shape}")
```

Рис. 2.4.1 Код первичной обработки данных

Ансамблевые методы машинного обучения комбинируют несколько базовых моделей для улучшения общей производительности и устойчивости к переобучению. В данном разделе рассматриваются три подхода: Градиентный бустинг, AdaBoost и Стэкинг-регрессор.

AdaBoost (Adaptive Boosting) — это ансамблевый метод, который комбинирует слабые модели (например, деревья решений) для создания сильной модели. В данном случае AdaBoost применяется для предсказания среднего времени ожидания (W).

StackingRegressor представляет собой ансамблевый метод, который объединяет несколько базовых моделей для создания более мощной финальной модели. В качестве финальной модели часто используется простая модель, такая как линейная регрессия, которая комбинирует предсказания базовых моделей.

Описание шагов:

1. **Определение базовых моделей:** для каждой целевой переменной (Q и W) определяются базовые модели, которые будут использоваться в стэкинге. В данном случае это градиентный бустинг и AdaBoost.

2. **Инициализация стэкинг-регрессора:** создаются экземпляры **StackingRegressor** для каждой целевой переменной. В качестве финальной модели используется линейная регрессия (**LinearRegression**). Параметр `passthrough=False` означает, что только предсказания базовых моделей используются для обучения финальной модели. Параметр `cv=5` указывает на использование 5-кратной перекрестной проверки при обучении.

3. **Обучение модели стэкинга:** модель стэкинга обучается на обучающей выборке для каждой целевой переменной.

4. **Предсказание и оценка:** выполняются предсказания на тестовой выборке и рассчитываются метрики **MSE** и **R²** для оценки качества модели.

Интерпретация результатов:

- **Стэкинг** может предложить улучшение точности по сравнению с отдельными базовыми моделями за счет комбинирования их сильных сторон.

- Анализ результатов оценки позволяет определить, превосходит ли ансамблевый подход отдельные методы в задаче предсказания.

После обучения и оценки метамоделей можно использовать ее для предсказания характеристик системы на новых наборах входных параметров без необходимости запуска имитационных экспериментов. Это существенно ускоряет процесс анализа и позволяет оперативно получать оценки производительности системы.

На рис. 2.4.2-2.4.5 представлен частичный код реализации метамоделей с использованием ансамблевых методов для предсказания Q и W . Код разделен на логические секции, соответствующие описанным выше этапам.

```
data = pd.read_csv('smo_simulation_data.csv')
data = data.dropna()

# Разделение данных на признаки и целевые переменные
X = data[['lambda', 'r', 'theta']]
y_Q = data['Q']
y_W = data['W']

# Разделение на обучающую и тестовую выборки
X_train, X_test, yQ_train, yQ_test, yW_train, yW_test = train_test_split(
    X, y_Q, y_W, test_size=0.2, random_state=42
)

# 3. Обучение моделей ансамблевых методов
# 3.1. Градиентный бустинг для предсказания Q
gb_Q = GradientBoostingRegressor(random_state=42)
gb_Q.fit(X_train, yQ_train)
yQ_pred = gb_Q.predict(X_test)
mse_Q = mean_squared_error(yQ_test, yQ_pred)
r2_Q = r2_score(yQ_test, yQ_pred)
print(f"Gradient Boosting - Q: MSE = {mse_Q:.4f}, R² = {r2_Q:.4f}")
```

Рис. 2.4.2 Код реализации метамоделей с использованием ансамблевых методов
(начало)

```

# AdaBoost для предсказания w
ada_w = AdaBoostRegressor(random_state=42)
ada_w.fit(X_train, yw_train)
yw_pred = ada_w.predict(X_test)
mse_w = mean_squared_error(yw_test, yw_pred)
r2_w = r2_score(yw_test, yw_pred)
print(f"AdaBoost - w: MSE = {mse_w:.4f}, R² = {r2_w:.4f}")

# Стэкинг (Stacking) для предсказания Q и W
estimators_Q = [
    ('gb', GradientBoostingRegressor(random_state=42)),
    ('ada', AdaBoostRegressor(random_state=42))
]

estimators_W = [
    ('gb', GradientBoostingRegressor(random_state=42)),
    ('ada', AdaBoostRegressor(random_state=42))
]

stack_regressor_Q = StackingRegressor(
    estimators=estimators_Q,
    final_estimator=LinearRegression(),
    passthrough=False,
    cv=5
)

stack_regressor_W = StackingRegressor(
    estimators=estimators_W,
    final_estimator=LinearRegression(),
    passthrough=False,
    cv=5
)

```

Рис. 2.4.3 Код реализации метамоделли с использованием ансамблевых методов
(продолжение)

```

stack_regressor_q.fit(X_train, yQ_train)
yQ_pred_stack = stack_regressor_q.predict(X_test)
mse_q_stack = mean_squared_error(yQ_test, yQ_pred_stack)
r2_q_stack = r2_score(yQ_test, yQ_pred_stack)
print(f"Stacking Regressor - Q: MSE = {mse_q_stack:.4f}, R² = {r2_q_stack:.4f}")

stack_regressor_w.fit(X_train, yW_train)
yW_pred_stack = stack_regressor_w.predict(X_test)
mse_w_stack = mean_squared_error(yW_test, yW_pred_stack)
r2_w_stack = r2_score(yW_test, yW_pred_stack)
print(f"Stacking Regressor - W: MSE = {mse_w_stack:.4f}, R² = {r2_w_stack:.4f}")

# Настройка гиперпараметров с использованием Grid Search

# Настройка Gradient Boosting для Q
param_grid_gb = {
    'n_estimators': [100, 200, 300],
    'learning_rate': [0.01, 0.1, 0.2],
    'max_depth': [3, 5, 7],
    'subsample': [0.8, 1.0]
}

grid_gb_q = GridSearchCV(
    GradientBoostingRegressor(random_state=42),
    param_grid_gb,
    cv=5,
    scoring='neg_mean_squared_error',
    n_jobs=-1
)

grid_gb_q.fit(X_train, yQ_train)

```

Рис. 2.4.4 Код реализации метамоделли с использованием ансамблевых методов
(продолжение)

```

# использование метамодел для предсказаний

# Пример новых параметров
new_parameters = pd.DataFrame({
    'lambda': [50, 75, 100],
    'r': [3, 5, 7],
    'theta': [0.1, 0.2, 0.3]
})

# Предсказание Q с помощью настроенного Gradient Boosting
Q_pred_new = grid_gb_Q.predict(new_parameters)
print(f"Предсказанные значения Q: {Q_pred_new}")

# Предсказание W с помощью настроенного AdaBoost
W_pred_new = grid_ada_W.predict(new_parameters)
print(f"Предсказанные значения W: {W_pred_new}")

# Предсказание Q и W с помощью модели стэкинга
Q_pred_stack_new = stack_regressor_Q.predict(new_parameters)
W_pred_stack_new = stack_regressor_W.predict(new_parameters)
print(f"Предсказанные значения Q (Stacking): {Q_pred_stack_new}")
print(f"Предсказанные значения W (Stacking): {W_pred_stack_new}")

# Заключение

# Вывод предсказанных значений
print("Предсказанные значения Q:", Q_pred_new)
print("Предсказанные значения W:", W_pred_new)
print("Предсказанные значения Q (Stacking):", Q_pred_stack_new)
print("Предсказанные значения W (Stacking):", W_pred_stack_new)

```

Рис. 2.4.5 Код реализации метамодел с использованием ансамблевых методов

(конец)

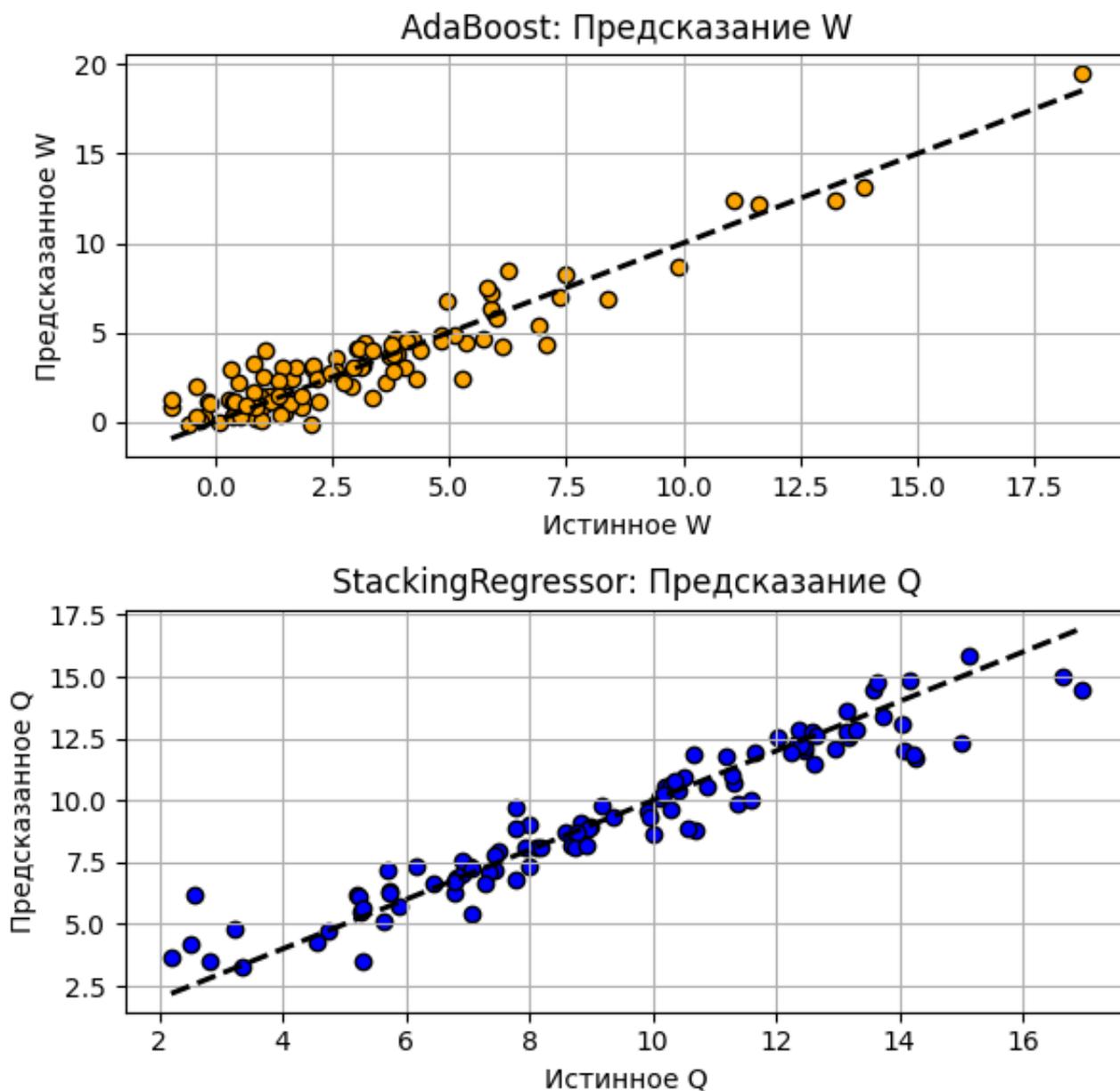


Рис. 2.4.6 Результаты предсказаний для характеристик системы массового обслуживания с использованием трех ансамблевых методов: AdaBoost, и StackingRegressor

Графики на рисунке 2.4.6 предсказаний показывают, что ансамблевые методы (AdaBoost, и StackingRegressor) продемонстрировали высокую точность при предсказании ключевых характеристик системы. Модели достаточно устойчивы, а ошибки распределены равномерно.

На графике (Рис. 2.4.7) представлена сходимость обучения двух ансамблевых методов: **AdaBoost** и **Gradient Boosting**, в зависимости от количества базовых моделей.

1. Кривая ошибок для AdaBoost:

- Показана оранжевой линией. Видно, что по мере увеличения количества базовых моделей (количество деревьев решений) ошибка на тестовой выборке постепенно уменьшается и стабилизируется.
- После достижения 100 базовых моделей ошибка практически не изменяется, что указывает на оптимальный баланс между сложностью модели и её точностью.

2. Кривая ошибок для Gradient Boosting:

- Показана зеленой линией. Ошибка также уменьшается с увеличением количества деревьев, но сходимость модели наблюдается немного быстрее, чем у AdaBoost.
- После 80 базовых моделей ошибка становится практически постоянной, что свидетельствует о достижении оптимального числа деревьев.

Вывод:

Графики сходимости подтверждают, что оба метода хорошо справляются с обучением. Gradient Boosting демонстрирует более быструю сходимость, тогда как AdaBoost требует большего количества базовых моделей для достижения стабильных результатов. Эти результаты могут быть использованы для выбора оптимальных гиперпараметров при обучении.

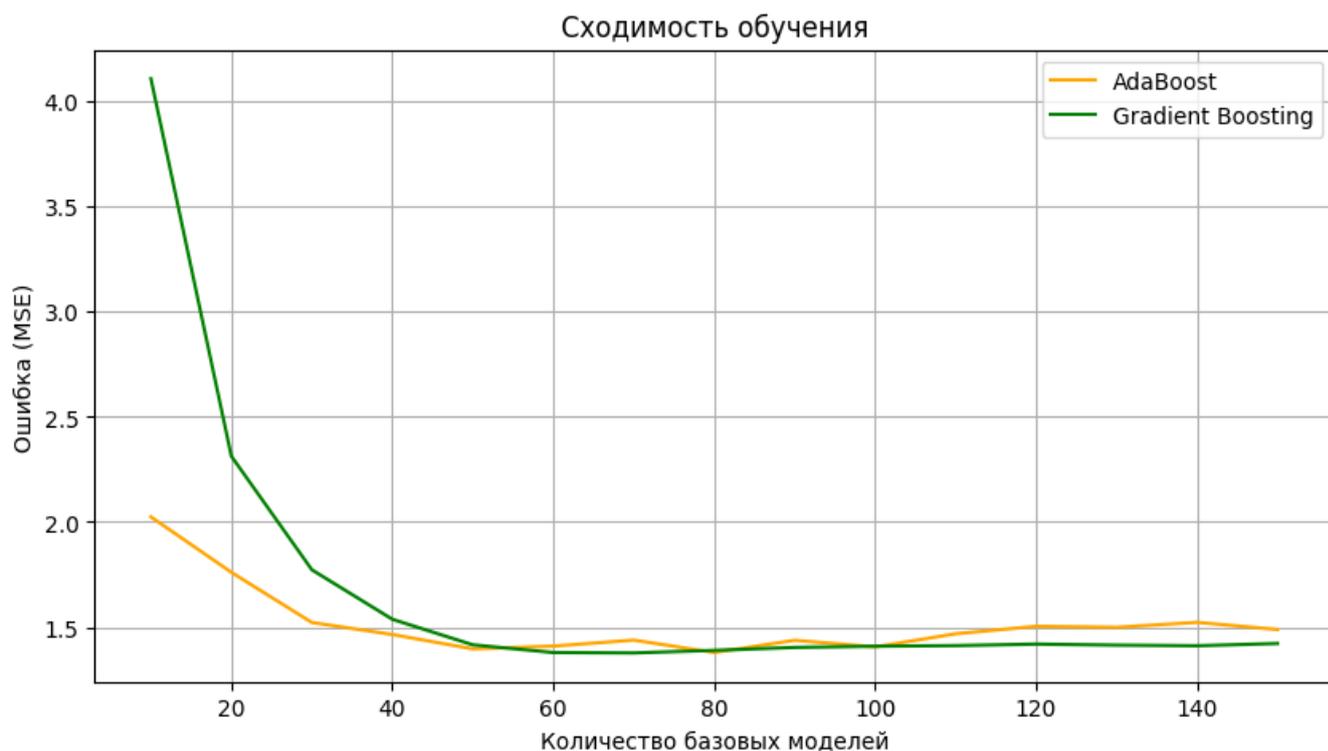


Рис. 2.4.7 Сходимость ансамблевых методов

В рамках исследования было принято решение увеличить количество данных в обучающей выборке до 5000 примеров. Данное изменение обосновано следующими причинами:

1. Повышение качества метамоделей:

Увеличение количества обучающих данных позволяет улучшить обобщающую способность моделей машинного обучения. Метамодель, обученная на большем объеме данных, становится более точной и менее подверженной переобучению.

2. Стабильность результатов:

Небольшие выборки могут приводить к нестабильным результатам из-за случайных выбросов и дисбаланса данных. Увеличение объема выборки позволяет более точно аппроксимировать зависимость между входными параметрами системы массового обслуживания (λ , r , θ) и выходными характеристиками (Q , W).

В рамках эксперимента были реализованы следующие этапы:

1. Для исследования использовался метод случайной генерации данных с заданными диапазонами параметров. Входные параметры включали интенсивность входного потока, число фаз обслуживания и ограничения по времени ожи-

дания. Значения интенсивности входного потока варьировались от минимального до максимального уровня, число фаз обслуживания принимало дискретные значения в определённом диапазоне, а параметр ограничения по времени ожидания был задан в виде непрерывного интервала. Для каждой комбинации входных параметров вычислялись целевые переменные, такие как среднее число заявок в очереди и среднее время ожидания. Генерация данных была проведена для пяти тысяч примеров, что позволило создать насыщенную и репрезентативную обучающую выборку для дальнейшего анализа.

2. На основе полученной выборки проводилось обучение ансамблевых моделей машинного обучения. В эксперименте использовались три метода: AdaBoost, Gradient Boosting и StackingRegressor. Для каждого метода осуществлялся подбор оптимальных гиперпараметров. В случае AdaBoost настраивались параметры, включая количество базовых моделей и глубину деревьев решений. Для Gradient Boosting дополнительно подбирались скорость обучения и количество деревьев. В StackingRegressor в качестве базовых моделей использовались предсказания AdaBoost и Gradient Boosting, а финальная модель, объединяющая результаты, представляла собой линейную регрессию.

3. Для оценки производительности обученных моделей применялись метрики качества на тестовой выборке. В качестве основных показателей использовались среднеквадратическая ошибка, которая характеризует среднее отклонение предсказанных значений от истинных, и коэффициент детерминации, позволяющий оценить долю объяснённой дисперсии. Для наглядного анализа качества предсказаний были построены несколько типов графиков. Графики предсказаний против истинных значений позволили визуально оценить точность моделей, а графики сходимости обучения продемонстрировали изменение ошибки в зависимости от количества базовых моделей в ансамбле. Дополнительно были построены гистограммы ошибок и диаграммы типа Boxplot для визуализации распределения ошибок и выявления потенциальных отклонений.

Таким образом, эксперимент охватывал этапы от генерации данных до комплексного анализа качества предсказаний, что позволило обоснованно оценить

эффективность ансамблевых методов машинного обучения для моделирования систем массового обслуживания.

Результаты эксперимента представлены на рисунках 2.4.8-2.4.10.

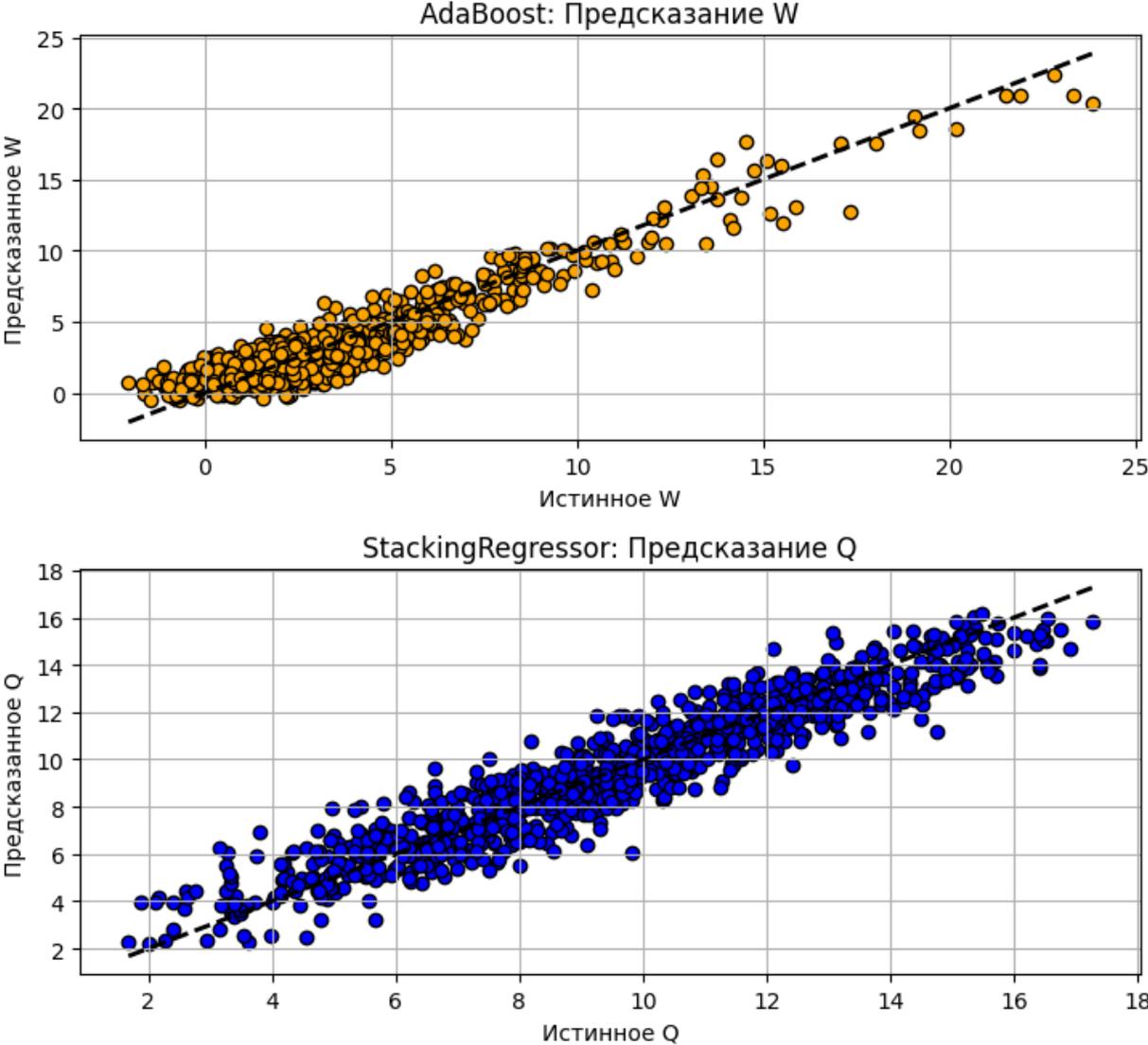


Рис. 2.4.8 Результаты предсказания моделей

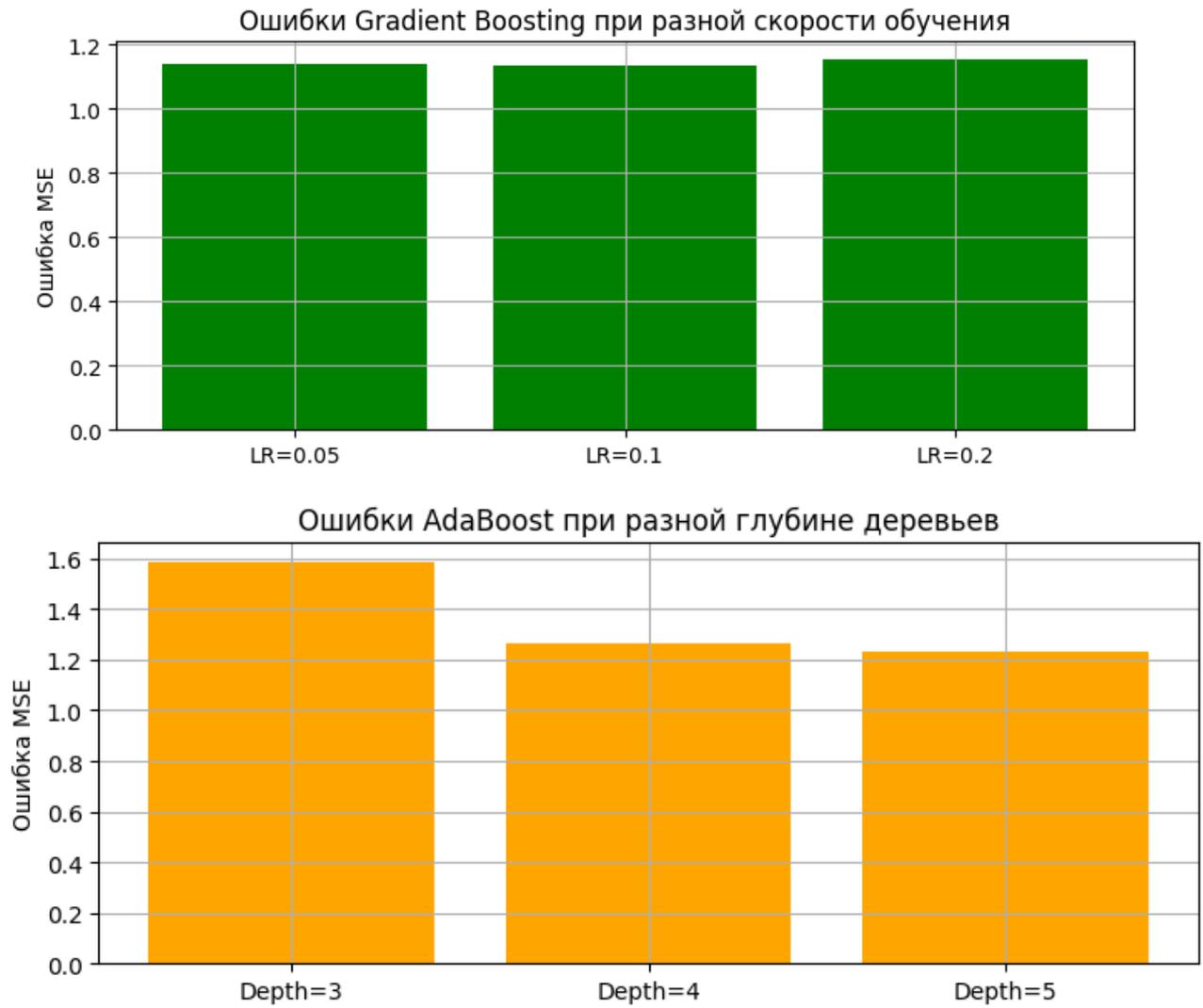


Рис. 2.4.9 Ошибки моделей при разных гиперпараметрах

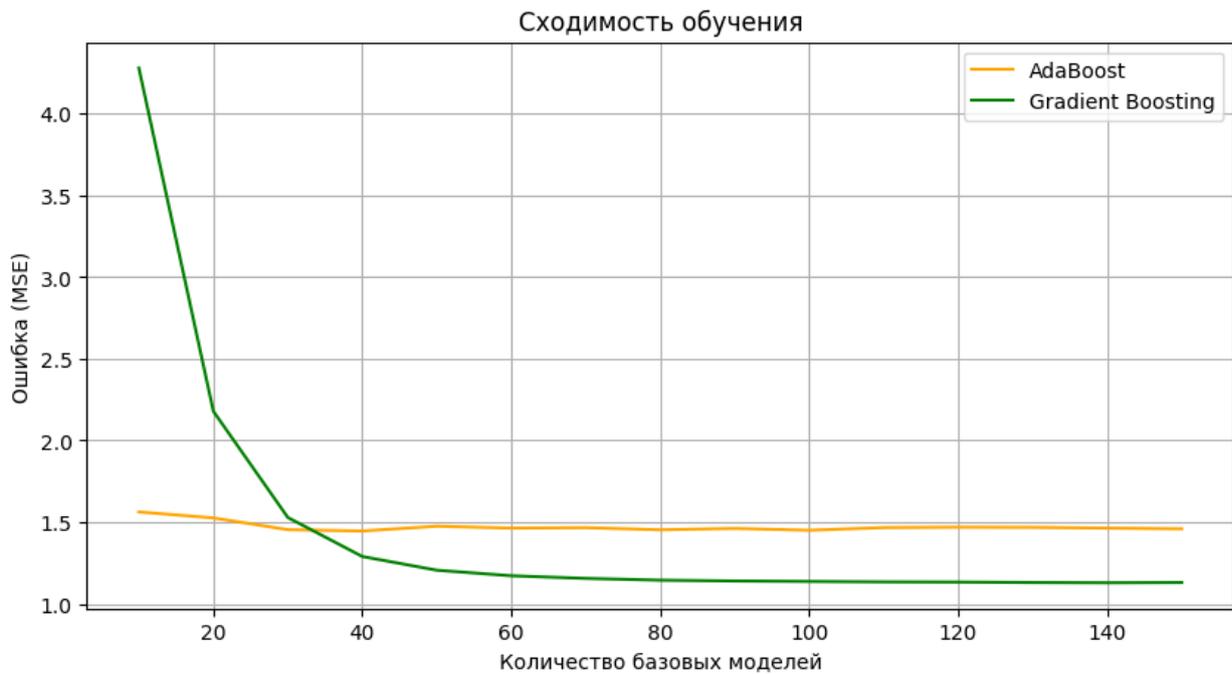


Рис. 2.4.10 Зависимость сходимости обучения от количества базовых моделей

На графике представлены кривые ошибок для моделей Gradient Boosting и AdaBoost. Для Gradient Boosting ошибка на первых этапах обучения достаточно высокая, однако по мере увеличения количества базовых моделей она быстро снижается. Начиная с 40-60 базовых моделей, ошибка стабилизируется и выходит на плато, достигая минимального значения. Такое поведение связано с тем, что Gradient Boosting на каждом шаге настраивается на минимизацию ошибки, последовательно корректируя предсказания. Это позволяет модели быстро обучаться и достигать хороших результатов даже при сравнительно небольшом числе деревьев решений.

В случае AdaBoost ошибка на начальном этапе уже сравнительно низкая и практически не изменяется при увеличении количества базовых моделей. Подобное поведение объясняется особенностями алгоритма: AdaBoost работает с слабыми моделями, такими как деревья решений небольшой глубины. При этом вклад каждого последующего дерева становится незначительным, если исходная ошибка уже мала. В результате AdaBoost демонстрирует меньшую подстройку

под данные по сравнению с Gradient Boosting и оказывается менее чувствительным к числу базовых моделей.

Причины наблюдаемого поведения заключаются в различиях между двумя методами. Gradient Boosting более агрессивно минимизирует ошибку на каждом этапе обучения, что позволяет деревьям решений глубже адаптироваться к данным. Это приводит к быстрой сходимости и стабилизации ошибки при увеличении числа моделей. В отличие от него, ансамбль AdaBoost оказывается менее чувствительным к числу базовых моделей, особенно если ошибка на начальных этапах уже невелика. Это объясняет горизонтальную форму кривой для AdaBoost на графике.

Для более реалистичного моделирования системы массового обслуживания рассмотрим характерные распределения для параметров входного потока и времени обслуживания. Вместо равномерного случайного распределения параметров, как это делалось ранее, генерация данных будет основываться на следующих классических распределениях:

1. Пуассоновское распределение для интенсивности входного потока: Интенсивность входного потока заявок описывается потоком Пуассона, который широко используется для моделирования случайного поступления заявок. Это позволит более точно отразить вероятностную природу появления заявок в системе за единицу времени.

2. Эрланговское распределение для времени обслуживания: Для моделирования времени обслуживания на каждой фазе используется эрланговское распределение. Оно представляет собой частный случай гамма-распределения и подходит для описания времени выполнения задач, разбитых на несколько последовательных этапов.

3. Пуассоновское распределение на каждой фазе: Интенсивность поступления заявок на каждой отдельной фазе обслуживания также будет подчиняться распределению Пуассона, что позволит учесть нелинейные изменения в процессе обработки заявок на разных фазах системы.

Результаты эксперимента приведены на рисунках 2.4.11, 2.4.12

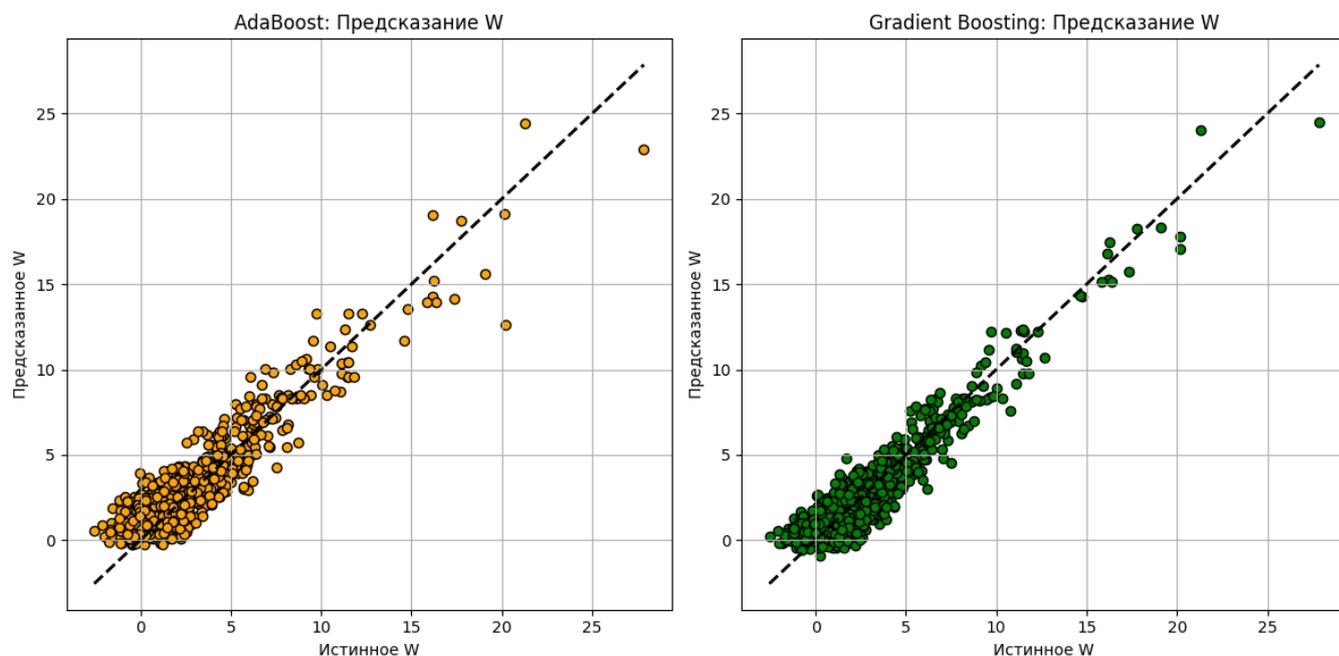


Рис. 2.4.11 Результаты предсказания моделей

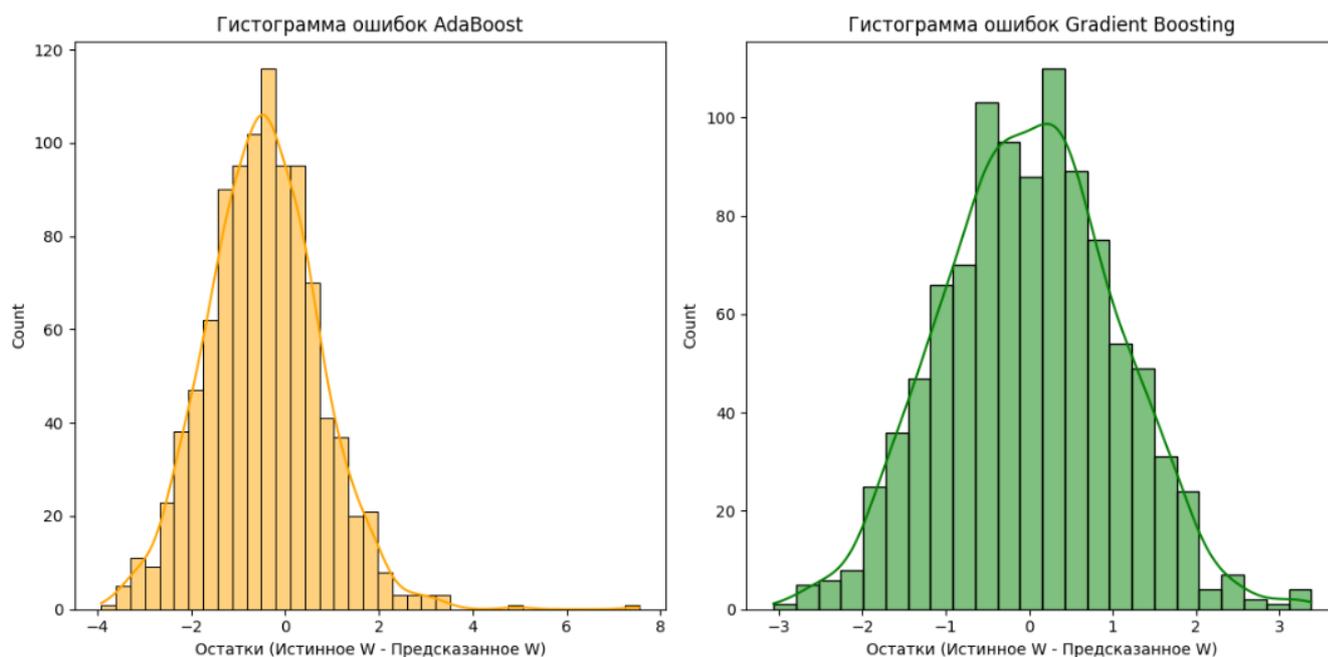


Рис. 2.4.12 Гистограммы ошибок моделей

Анализ графиков показал, что Gradient Boosting превосходит AdaBoost по точности предсказаний. Модель Gradient Boosting демонстрирует более плотное соответствие предсказанных значений с истинными, узкое распределение ошибок и меньшую абсолютную ошибку на тестовой выборке. AdaBoost также справляется с задачей предсказания, но его ошибки более вариативны, что отражается на

широкой гистограмме и большем размахе значений в Voxplot. Дополнительная визуализация распределений параметров входного потока позволяет подтвердить корректность генерации данных и приближение модели к реальным стохастическим процессам в системах массового обслуживания.

Таким образом, в ходе исследования была разработана и реализована методика генерации данных для систем массового обслуживания с учётом стохастической природы процессов. Генерация параметров системы производилась на основе классических вероятностных распределений, что позволило приблизить модель к реальным условиям эксплуатации. Интенсивность входного потока заявок моделировалась с использованием пуассоновского распределения, отражающего случайный характер поступления заявок во времени. Время обслуживания на этапах системы описывалось эрланговским распределением, что обеспечивало реалистичную вариативность длительности выполнения задач, разделённых на несколько последовательных фаз. Число фаз обслуживания принимало дискретные значения, что позволило учитывать различную сложность структуры обслуживания в системе. Использование вероятностных распределений для генерации входных данных обеспечило реалистичность и статистическую обоснованность моделируемых процессов. Сравнительный анализ ансамблевых методов показал, что Gradient Boosting является более эффективным инструментом для аппроксимации производственных характеристик систем массового обслуживания, характеризуясь высокой точностью и сходимостью. Результаты подтвердили целесообразность применения методов машинного обучения для построения метамоделей, способных оперативно предсказывать ключевые показатели системы на основе заданных параметров.

Выводы:

1. Анализ нестационарных условий: были рассмотрены не только стационарные, но и нестационарные условия работы системы массового обслуживания. Это позволяет более точно моделировать реальные условия эксплуатации, где параметры системы могут изменяться во времени.

2. Адекватность моделей: Аналитическое моделирование было использовано для получения аналитических выражений временных характеристик системы, учитывающих нестационарные условия. Имитационное моделирование в системе AnyLogic было разработано для проверки и валидации этих результатов. Оба метода показали хорошее соответствие между собой, что подтверждает адекватность моделей для оценки производительности и поведения системы в динамических условиях.

3. Практическое применение: Полученные результаты имеют практическую значимость для управления системами массового обслуживания в реальных условиях. Анализ нестационарных условий позволяет оптимизировать параметры системы, улучшать качество обслуживания клиентов и эффективность использования ресурсов. В результате проведенного исследования была предложена метамодель системы массового обслуживания с использованием ансамблевых методов машинного обучения. Эта метамодель может служить основой для дальнейших исследований и может быть адаптирована для моделирования различных сценариев работы системы, обеспечивая гибкость и точность в предсказании ключевых характеристик системы.

3 ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ СТАБИЛЬНЫХ РЕЖИМОВ РАБОТЫ СМО

В системах массового обслуживания (СМО) с пуассоновским потоком и экспоненциальным временем обслуживания всегда возникают очереди и задержки в обслуживании, что приводит к дестабилизации работы системы. В контексте многофазового обслуживания на практике возможность быстрого изменения параметров ограничена, в основном, числом фаз обслуживания.

Несмотря на обширное количество работ по теории массового обслуживания, опубликованных за последнее время [68-76], вопрос устойчивых режимов работы для сложных СМО, в том числе многофазных, остается недостаточно изученным. Анализ указанных характеристик позволяет получить ряд важных сведений о режимах функционирования систем данного типа, в особенности СМО с ограничениями времени ожидания. Исследование этих параметров дает возможность извлечь ряд нетривиальных заключений о режимах работы систем, в частности, СМО с ограничениями на время пребывания заявок в очереди.

Рассмотрим, как изменится поведение моментов длины очереди при изменении числа фаз обслуживания.

$$\begin{aligned} \sigma_Q^2 &= \sum_k (k-1)^2 p_k - Q^2 = \sum_k k^2 p_k - 2 \sum_k k p_k + \sum_k p_k - Q^2 = \\ &= P'(1) + P''(1) - 2P'(1) + (1 - p_0) - Q^2 = \\ &= 1 - \frac{r^2 (\mu \hat{\rho} (-2 + (-1 + r) \hat{\rho}) + \lambda (2 + r(-2 + \hat{\rho}) \hat{\rho} + r^2 \hat{\rho}^2))^2}{(-2\mu(-1 + \rho)^2 + r(2 + r(-2 + \hat{\rho}) \hat{\rho} + \rho^2) \theta)^2} \\ &\quad - \frac{\lambda^2 r (1 + r) (6\mu^2 - 4\lambda\mu(-1 + r) + \lambda^2(-1 + r)r)(-1 + \rho)^2 (-\mu + r(\lambda + \theta))}{3\mu(\mu - \lambda r)^3 (2\mu(-1 + \rho)^2 - r(2 + r(-2 + \hat{\rho}) \hat{\rho} + \rho^2) \theta)} \\ &\quad - \frac{2\rho(-2 + (-1 + r) \hat{\rho}) (-\mu + r(\lambda + \theta))}{-2\mu(-1 + \rho)^2 + r(2 + r(-2 + \hat{\rho}) \hat{\rho} + \rho^2) \theta} \\ &\quad - \frac{2(-1 + \rho)^2 (-\mu + r(\lambda + \theta))}{-2\mu(-1 + \rho)^2 + r(2 + r(-2 + \hat{\rho}) \hat{\rho} + \rho^2) \theta} \end{aligned}$$

$$\text{где } \hat{\rho} = \frac{\lambda}{\mu}.$$

Зависимость математического ожидания и дисперсии числа заявок, ожидающих обслуживания в очереди, от числа фаз обслуживания наглядно продемонстрирована на рис. 3.1.1.

Существует некоторое пороговое значение r , соответствующее точке пересечения кривых математического ожидания и дисперсии.

Эта точка делит ось абсцисс на две области:

- Область, где среднеквадратичное отклонение (СКО) длины очереди не превышает ее математическое ожидание. В этой области наблюдается стабильный и предсказуемый режим работы системы с точки зрения эксплуатации.
- Область, где разброс числа заявок в очереди превышает среднее значение. В этой области система функционирует в менее предсказуемом режиме.

Режим работы системы, при котором СКО длины очереди не выходит за пределы ее математического ожидания, является наиболее предпочтительным.

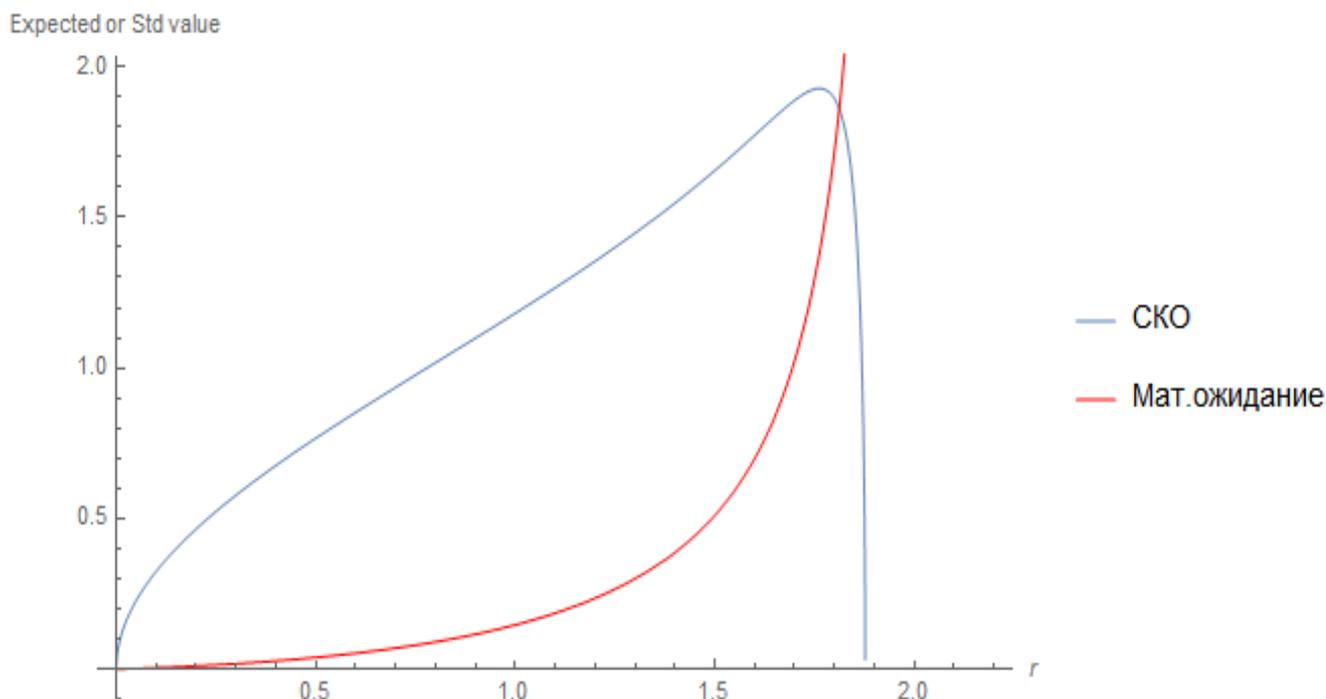


Рис. 3.1.1 – Зависимость математического ожидания и СКО числа заявок в очереди от числа фаз обслуживания

На основе математической модели, разработанной в главе 2, была создана специальная программа для вычисления граничного значения r . Программа ис-

пользует в качестве исходных данных известную интенсивность входного потока заявок и соответствующие ограничения. Для исследования зависимости граничного значения r от нагрузки на систему, создаваемой входным потоком, с помощью программы был проведен цикл вычислительных экспериментов.

Алгоритм, используемый для проведения экспериментов представлен на рисунке 3.1.2.

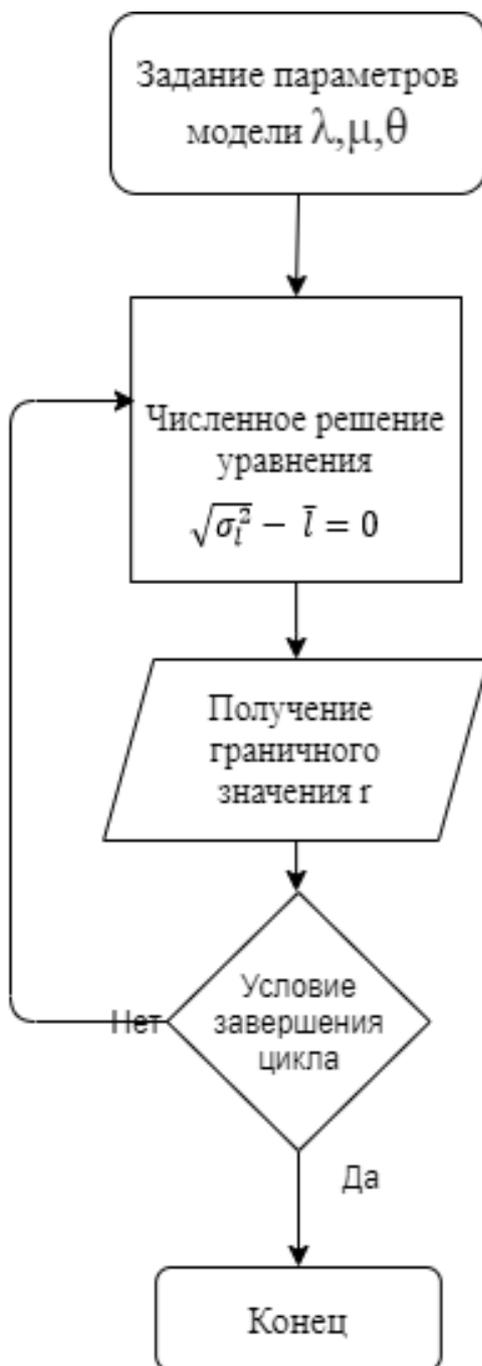


Рис. 3.1.2 – Блок-схема алгоритма расчета r граничного

Путем изменения интенсивности входного потока заявок были определены значения r граничного для различных моментов длины очереди.

Результаты вычислительных экспериментов для различных частных случаев рассматриваемой модели СМО будут представлены далее.

Описание алгоритма определения граничного значения r

1. Задание параметров модели:

На первом шаге задаются входные параметры системы массового обслуживания:

- λ – интенсивность поступления заявок;
- μ – интенсивность обслуживания;
- θ – параметр, связанный с ограничением времени ожидания заявки в очереди.

2. Численное решение уравнения:

После задания параметров необходимо численно решить уравнение: $\sqrt{(\sigma_Q^2)} - \bar{l} = 0$,

где \bar{l} – математическое ожидание длины очереди, а $\sqrt{(\sigma_Q^2)}$ – среднеквадратичное отклонение числа заявок в очереди.

Таким образом, ищется такое значение параметра r , при котором среднеквадратичное отклонение длины очереди равно её математическому ожиданию.

3. Определение граничного значения r :

В результате решения уравнения находится граничное значение r , при котором $\sqrt{(\sigma_Q^2)} = \bar{l}$. Значение r определяет границу между устойчивым (предсказуемым) режимом работы системы и режимом с более высоким разбросом длины очереди.

4. Проверка условия завершения:

После вычисления r проверяется условие завершения процесса вычислений. Если условие не выполнено, параметры или начальные приближения могут быть изменены, и шаг решения уравнения повторяется.

5. Завершение алгоритма:

При выполнении условия завершения алгоритм заканчивается. Полученное граничное значение r может использоваться для оценки и планирования режима работы системы.

Все расчеты для исследования проводились в системе компьютерной алгебры Wolfram Mathematica, широко используемой в научных, инженерных, математических и компьютерных областях.

С использованием алгоритма, блок-схема которого представлена на рисунке 3.1.2, необходимо найти формулу, отражающую зависимость граничного значения фаз обслуживания от коэффициента вариации длины очереди. Для определения формулы, отражающей зависимость граничного значения фаз обслуживания от коэффициента вариации длины очереди, необходимо использовать алгоритм, представленный на блок-схеме 3.1.2. В этом алгоритме задействована функция LinearModelFit из Wolfram Mathematica.

Эта функция возвращает объект, содержащий информацию о подгонке данных по точкам для модели линейной регрессии. Эта информация позволяет легко извлекать результаты моделирования и проводить диагностику. Таким образом, используя алгоритм 3.1.2 и функцию LinearModelFit, можно получить формулу, описывающую зависимость между граничным значением фаз обслуживания и коэффициентом вариации длины очереди. Набор данных для данной функции формируется исходя из определения стабильного режима работы системы: стабильным мы назовем режим работы системы, при котором СКО исследуемой величины не превосходит ее математического ожидания. Для формирования набора данных численно с помощью метода простых итераций решаем уравнение $\sigma_Q^2 = Q$ и находим значения r , удовлетворяющие данному равенству.

Использование функции LinearModelFit в среде Wolfram Mathematica не противоречит факту, что искомая зависимость может носить логарифмический характер. Данный метод регрессии предназначен для оценки параметров линейных моделей, однако при наличии априорных сведений о специфике зависимости

(например, о её логарифмической форме) исходные данные или аргументы могут быть преобразованы соответствующим образом.

Так, если предполагается, что граничное число фаз обслуживания связано с интенсивностью входящего потока посредством логарифмической зависимости, можно выполнить предварительное преобразование переменных. В частности, логарифмирование аргументов или результатов измерений позволяет «линеаризовать» исходную нелинейную связь. После такого преобразования процедура линейной регрессии применима к модифицированным данным и даёт возможность оценить параметры модели, которая в исходных единицах представляется логарифмической функцией.

Таким образом, применение LinearModelFit в контексте логарифмической зависимости является стандартной практикой: данные предварительно трансформируются так, чтобы модель стала линейной относительно новых переменных, и лишь затем к ней применяется линейная регрессия. Это обеспечивает корректное и удобное получение параметров модели, отражающей исходную логарифмическую зависимость.

Пример полученных результатов вычислительного эксперимента приведен на рис. 3.1.3.

Каждая точка, лежащая на кривой, соответствует равным значениям математического ожидания и СКО длины очереди при определенном значении приведенной интенсивности входного потока заявок. При этом коэффициент вариации длины очереди равен единице. Фактически это граница, выше которой СКО превышает математическое ожидание. Область под кривой соответствует стабильному режиму функционирования системы, при котором СКО длины очереди находится в пределах математического ожидания.

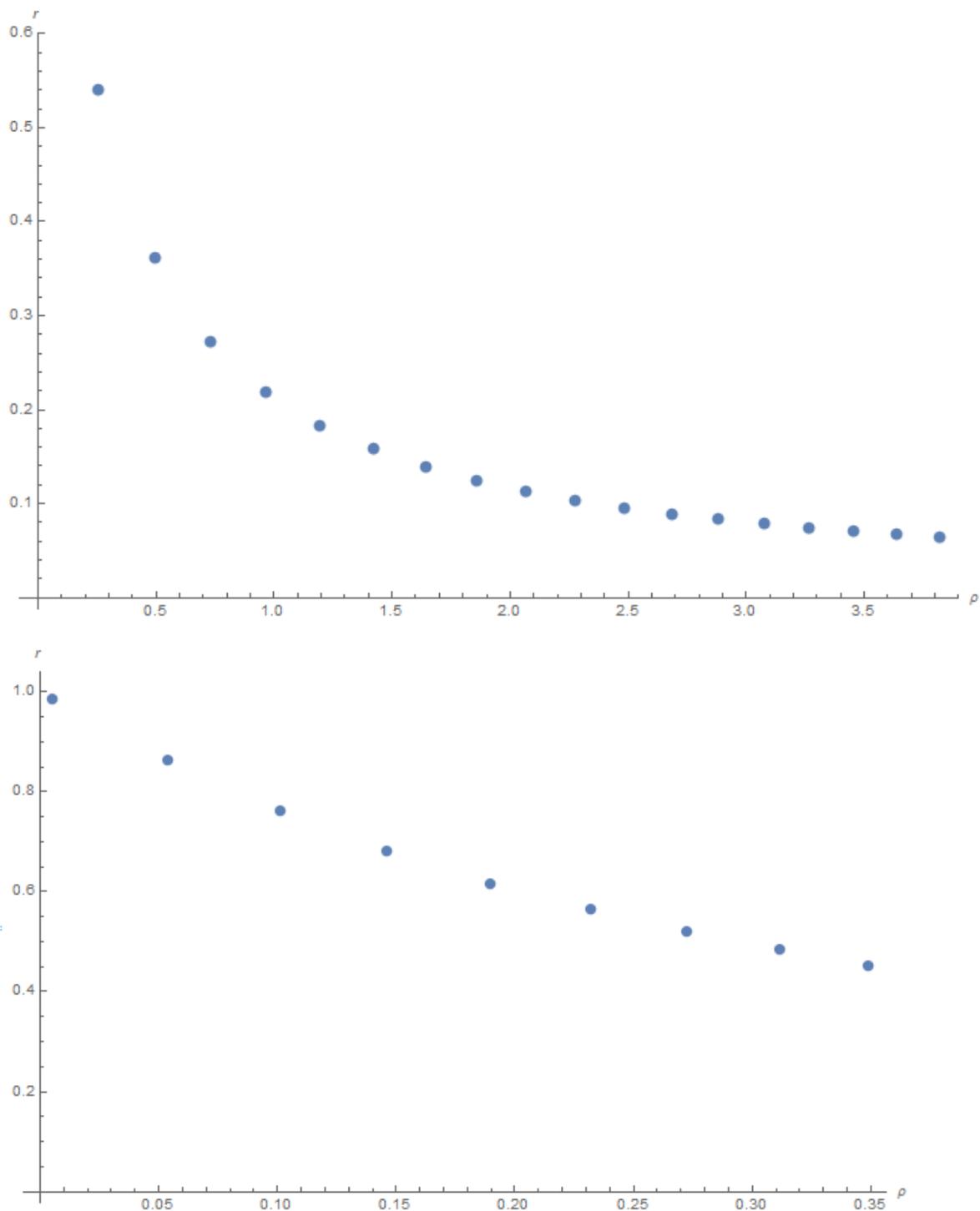
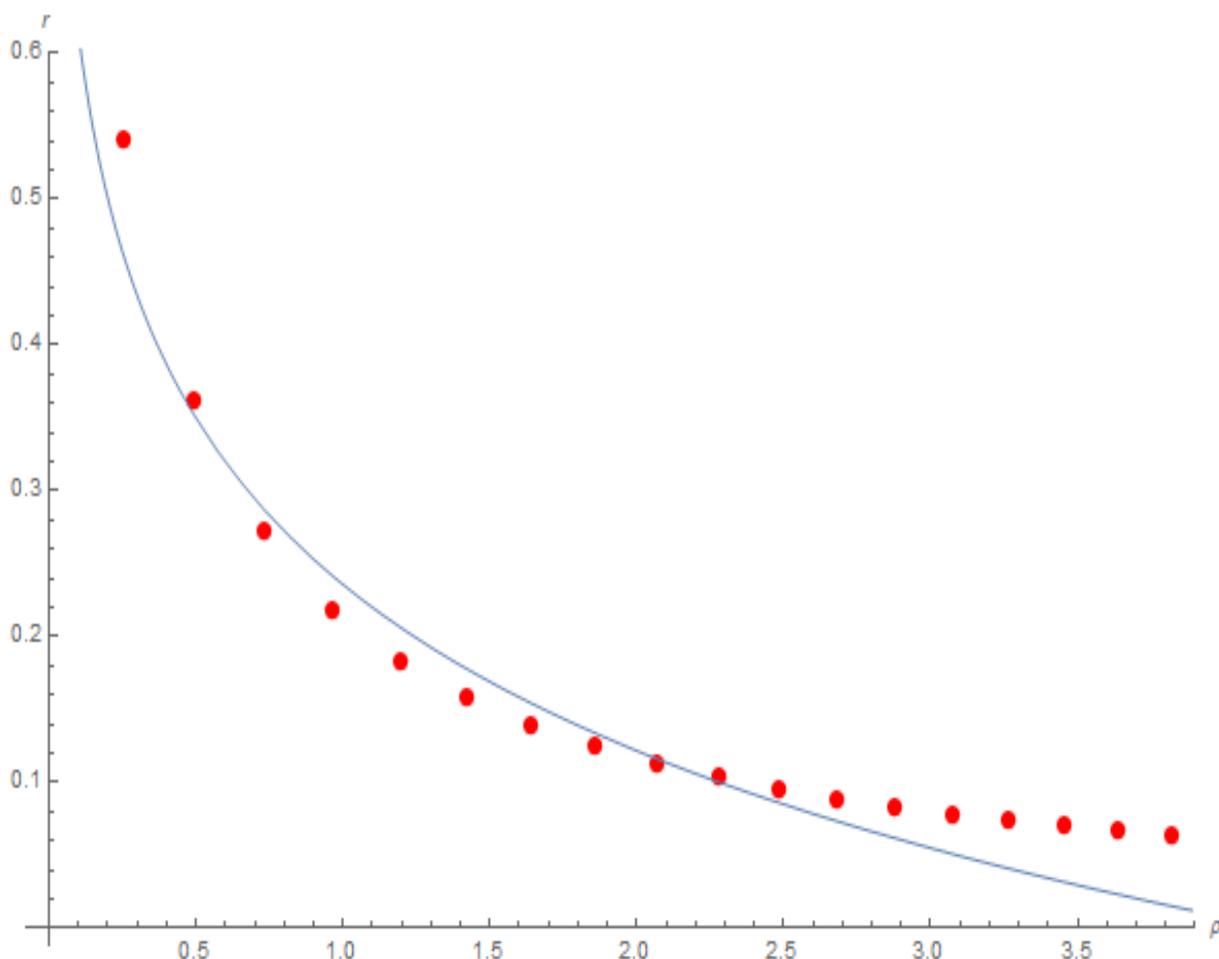


Рис.3.1.3 Зависимости граничного значения числа фаз от скорости поступления заявок

Для того чтобы определить функциональную зависимость с помощью вышеописанного метода LinearModelFit, отобразим данные и линию наилучшего соответствия при произвольных значениях приведенной интенсивности (рис.3.1.4).

В данном случае точками отображаются значения r граничного, сплошной голубой линией искомая аппроксимирующая функция:



3.1.4 Линия наилучшего соответствия модели

Далее с помощью функции *FittedModel* извлечем функциональную форму модели:

$$r(\rho) = a - b * \ln(\rho),$$

где a, b некоторые константы.

Таким образом, показана функциональная зависимость граничного числа фаз обслуживания, при которых обеспечивается стабильный режим работы системы от приведенных интенсивностей потока, которая носит логарифмический характер. Увеличение набора данных улучшает точность построенной аппроксимирующей функции, уменьшает невязку, однако существенно сказывается на времени расчета.

На рис. 3.1.5 представлен частичный пример кода для вычисления граничных значений числа фаз обслуживания:

```

L = Range[1, 1001, 50];
M = Range[100, 120, 1];
T = Range[100, 120, 1];

Length[L]
Length[M]
Length[T]
Rho = List[];
RGR = List[];

For[i = 1, i < Length[L], i++, rho = Part[L, i]/Part[M, i];
Q = (r*(Part[M, i]*rho*(-2 + (-1 + r)*rho) +
Part[L, i]*(2 + r*(-2 + rho)*rho + r^2*rho^2)))/(-2*
Part[M, i]*(-1 + r*rho)^2 +
r*(2 + r*(-2 + rho)*rho + r^2*rho^2)*Part[T, i]);
Nn = (r*
rho*(-2 + (-1 + r)*rho)*(Part[M, i] - Part[L, i]*r -
r*Part[T, i]))/(2*Part[M, i] - 4*Part[M, i]*r*rho +
2*Part[M, i]*r^2*rho^2 - 2*r*Part[T, i] + 2*r^2*rho*Part[T, i] -
r^2*rho^2*Part[T, i] - r^3*rho^2*Part[T, i]);
p0 = (2*(-1 + r*rho)^2*(Part[M, i] - Part[L, i]*r -
r*Part[T, i]))/(2*Part[M, i] - 4*Part[M, i]*r*rho +
2*Part[M, i]*r^2*rho^2 - 2*r*Part[T, i] + 2*r^2*rho*Part[T, i] -
r^2*rho^2*Part[T, i] - r^3*rho^2*Part[T, i]);
Disp = (Part[L, i]^2*p0*
r*(1 + r)*(6*Part[M, i]^2 - 4*Part[L, i] Part[M, i]*(-1 + r) +
Part[L, i]^2*(-1 + r)*r))/(6*
Part[M, i]*(Part[M, i] - Part[L, i]*r)^3 - 2*Nn + (1 - p0) -
Q^2;
Sko = Sqrt[Disp];
Funk[r_] = Sko - Q;

rgr = NSolve[Funk[r] == 0, r, PositiveReals, WorkingPrecision -> 5];
rgr1 = First[r /. rgr ];
rho1 = Part[L, i]/(Part[M, i] + Part[T, i]);
AppendTo[Rho, rho1];
AppendTo[RGR, rgr1];
]

Result = List[];
For[i = 1, i < Length[Rho], i++,
AppendTo[Result, {Part[Rho, i], Part[RGR, i]}]
]
ListPlot[Result, AxesLabel -> {\[Rho], r}]
Result

```

Рис. 3.1.5 Код для вычисления граничных значений числа фаз обслуживания

Рассмотрим альтернативный подход к определению зависимости граничного значения числа фаз обслуживания от коэффициента вариации длины очереди. Данный подход не противоречит ранее описанному, а лишь предлагает другой инструмент и программную среду для достижения схожих целей. В то время как ранее использовалась функция `LinearModelFit` в Wolfram Mathematica для определения линейной связи, в настоящем подходе предлагается:

1. Численно сформировать набор данных и соответствующие коэффициенты вариации длины очереди.
2. Применить методы регрессии в среде Python (например, с использованием библиотеки `scikit-learn`) для получения аппроксимации зависимости
3. При необходимости рассмотреть не только линейную, но и более гибкую модель (например, полиномиальную регрессию), которая, тем не менее, не противоречит исходной концепции.

На рис. 3.1.6-3.1.7 представлен пример кода, демонстрирующего вышеописанное решение:

```

import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from scipy.optimize import bisect

lambda_ = 0.8
mu = 1.0
theta = 0.5

def expected_queue_length(r, lambda_, mu, theta):
    if lambda_ < mu:
        return (lambda_ * r) / (mu - lambda_)
    else:
        return np.inf

def queue_variance(r, lambda_, mu, theta):
    eq = expected_queue_length(r, lambda_, mu, theta)
    varQ = 2 * eq
    return varQ

def f(r):
    eq = expected_queue_length(r, lambda_, mu, theta)
    varQ = queue_variance(r, lambda_, mu, theta)
    return varQ - eq

r_values = np.linspace(0.1, 10.0, 100)
r_solutions = []
cv_values = []

for guess in r_values:
    # проверяем знаки f в точках guess и guess+10
    if f(guess)*f(guess+10) < 0:
        try:
            r_root = bisect(f, guess, guess+10)
            eq = expected_queue_length(r_root, lambda_, mu, theta)
            varQ = queue_variance(r_root, lambda_, mu, theta)
            sigmaQ = np.sqrt(varQ)
            if np.isclose(varQ, eq, rtol=1e-3) and eq > 0:
                cv = sigmaQ / eq
                r_solutions.append(r_root)
                cv_values.append(cv)
        except:
            pass

```

Рис. 3.1.6 Код для вычисления зависимости числа фаз обслуживания от вариации длины очереди

```

if len(r_solutions) == 0:
    print("не удалось найти ни одной точки, удовлетворяющей уравнению на данном интервале.")
else:
    r_solutions = np.array(r_solutions).reshape(-1, 1)
    cv_values = np.array(cv_values).reshape(-1, 1)

    model = LinearRegression()
    model.fit(cv_values, r_solutions)

    a = model.coef_[0][0]
    b = model.intercept_[0]
    print(f"найденная линейная зависимость: r = {a:.4f} * cv(Q) + {b:.4f}")

    plt.figure(figsize=(10,6))
    plt.scatter(cv_values, r_solutions, color='blue')
    plt.plot(cv_values, model.predict(cv_values), color='red')
    plt.xlabel('cv(Q)')
    plt.ylabel('r')
    plt.grid(True)
    plt.show()

```

Рис. 3.1.7 Код для вычисления зависимости числа фаз обслуживания от вариации длины очереди (конец)

Если предполагается, что зависимость между граничным значением фаз обслуживания r и коэффициентом вариации длины очереди носит нелинейный характер, то вместо простой линейной регрессии можно использовать полиномиальные или другие нелинейные модели. На рис. 3.1.8-3.1.9 приведён пример кода на Python, который реализует полиномиальную регрессию с помощью класса `PolynomialFeatures` из библиотеки `scikit-learn`.

Для того чтобы численный алгоритм успешно находил решение, при котором среднеквадратичное отклонение длины очереди становится равным её математическому ожиданию, необходимо корректно выбрать параметры модели системы массового обслуживания. В упрощённом случае, рассмотренном ранее, применялся ограниченный набор зависимостей, в результате чего определить положительное значение параметра, удовлетворяющее искомому условию, было невозможно.

```

import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from scipy.optimize import bisect

lambda_ = 0.5
mu = 0.7
theta = 0.4
beta = 0.09

def expected_queue_length(r, lambda_, mu, theta):
    return (lambda_ * r) / (mu - lambda_) if lambda_ < mu else np.inf

def queue_variance(r, lambda_, mu, theta):
    eq = expected_queue_length(r, lambda_, mu, theta)
    varQ = eq*(1 + beta*(r - 1)**2)
    return varQ

def f(r):
    eq = expected_queue_length(r, lambda_, mu, theta)
    varQ = queue_variance(r, lambda_, mu, theta)
    return varQ - eq

r_values = np.linspace(0.1, 2.0, 100)
r_solutions = []
cv_values = []

for guess in r_values:
    if f(guess)*f(guess+0.1) < 0:
        try:
            r_root = bisect(f, guess, guess+0.1)
            eq = expected_queue_length(r_root, lambda_, mu, theta)
            varQ = queue_variance(r_root, lambda_, mu, theta)
            sigmaQ = np.sqrt(varQ)
            if np.isclose(varQ, eq, rtol=1e-3) and eq > 0:
                cv = sigmaQ / eq
                r_solutions.append(r_root)
                cv_values.append(cv)
        except:
            pass

```

Рис. 3.1.8 Код, который реализует полиномиальную регрессию

Чтобы добиться нетривиального решения, следует использовать более гибкий подход к заданию взаимосвязей между средней длиной очереди и её диспер-

сией. Вместо жёстко фиксированных зависимостей вводятся дополнительные параметры, позволяющие контролировать поведение исследуемых функций. Правильный выбор таких параметров гарантирует наличие точки пересечения кривых, отображающих среднее значение длины очереди и её среднеквадратичное отклонение. Это, в свою очередь, обеспечивает существование решения поставленной задачи при положительных значениях параметра.

На практике данный подход реализуется следующим образом. Сначала формируется новая, более гибкая функциональная зависимость, в которой варьируемые параметры позволяют сместить точку пересечения в нужную область значений. Затем, после проверки на численной модели, подтверждается, что искомый параметр действительно найден. Далее на основе полученных данных выполняется регрессионный анализ, позволяющий аппроксимировать найденную зависимость.

```
if len(r_solutions) == 0:
    print("Не удалось найти ни одной точки")
else:
    r_solutions = np.array(r_solutions).reshape(-1, 1)
    cv_values = np.array(cv_values).reshape(-1, 1)

    model = LinearRegression()
    model.fit(cv_values, r_solutions)

    a = model.coef_[0][0]
    b = model.intercept_[0]
    print("Найдено решение!")
    print(f"r = {a:.4f} * CV(Q) + {b:.4f}")

    plt.figure(figsize=(10,6))
    plt.scatter(cv_values, r_solutions, color='blue')
    plt.plot(cv_values, model.predict(cv_values), color='red')
    plt.xlabel('CV(Q)')
    plt.ylabel('r')
    plt.grid(True)
    plt.show()
```

Рис. 3.1.9 Код, который реализует полиномиальную регрессию (конец)

Таким образом, благодаря более гибкой постановке задачи и уточнённым зависимостям, обеспечивается возможность решения, при котором среднеквадратичное отклонение длины очереди оказывается равным её математическому ожиданию при заранее определённом положительном значении параметра.

Выводы:

В реальных условиях эксплуатации объектов и систем, работающих по принципу систем массового обслуживания, всегда важной является проблема очередей и задержек в обслуживании. Организация процесса эксплуатации таких систем с целью обеспечения стабильного режима работы является естественным желанием. Однако, параметром, который можно быстро изменить на практике, для случаев многофазового обслуживания, является только число фаз обслуживания.

Для более сложных моделей СМО, таких как многофазовые системы с ограничениями на длину очереди, понятие стабильных режимов работы пока не было достаточно изучено в литературе [45-75]. Исследование этих характеристик может привести к нетривиальным выводам о режимах функционирования таких СМО.

Для исследования поведения граничного числа фаз обслуживания была разработана специализированная программа, основанная на математической модели, представленной в предыдущих главах. Применяя численные методы и алгоритм, представленный на рисунке 3.1.2, удалось провести серию вычислительных экспериментов. В ходе этих экспериментов варьировались интенсивности входного потока заявок с целью определения условий, при которых система достигает режима, в котором среднеквадратичное отклонение длины очереди не превосходит её математического ожидания. Для упрощения модели и проверки различных сценариев были использованы искусственные зависимости, которые впоследствии модифицировались с учётом введения дополнительного параметра, гарантировавшего существование решения уравнения при положительных значениях параметра r .

Практическая реализация данного подхода была продемонстрирована на фрагментах программного кода (на языке Python), в котором применялись численные методы решения уравнений (например, метод бисекции), а также проводилась аппроксимация полученных результатов методами регрессии (линейной или полиномиальной). При этом увеличение размера выборки и детальность вычислительного эксперимента повышают точность аппроксимации данной зависимости, но возрастают и вычислительные затраты.

Таким образом, на основе сочетания аналитического и численного моделирования продемонстрировано, что регулирование числа фаз обслуживания оказывает существенное влияние на устойчивость работы системы массового обслуживания при различных уровнях входной нагрузки. Представленные вычислительные эксперименты, выполненные с помощью разработанного программного инструментария, подтверждают возможность настройки параметров системы для достижения устойчивого режима работы. Это, в свою очередь, создаёт предпосылки для дальнейших исследований в направлении более гибких и реалистичных моделей, а также для практического применения результатов при оптимизации параметров реальных СМО.

4 АПРОБАЦИЯ РЕЗУЛЬТАТОВ В ВИДЕ ПРОГРАММНОГО КОМПЛЕКСА С ПРИМЕНЕНИЕМ ПАРАЛЛЕЛЬНОЙ БИБЛИОТЕКИ ПЛАТФОРМЫ .NET

4.1 Постановка задачи

Теория массового обслуживания (ТМО) играет значимую роль в анализе и оптимизации процессов, связанных с обслуживанием потоков заявок, в различных областях, таких как телекоммуникации, производство, логистика, здравоохранение и финансы, как указано в работе [77]. Существует несколько способов, которыми ТМО применяется для этих целей.

Одним из ключевых направлений является определение ресурсов. ТМО позволяет точно определить количество необходимых ресурсов (например, операторов, рабочих мест или серверов) для обработки потока заявок при различной нагрузке. Это важно для планирования ресурсов и минимизации затрат.

Важным аспектом является снижение времени ожидания. В системах массового обслуживания одной из главных метрик является время ожидания. ТМО позволяет анализировать влияние различных факторов, таких как интенсивность потока заявок и количество каналов обслуживания, на время ожидания, и оптимизировать его.

Также ТМО используется для улучшения качества обслуживания. Качество обслуживания тесно связано с временем ожидания и доступностью ресурсов. ТМО позволяет проанализировать, как изменения в системе могут повлиять на качество обслуживания, и принять меры для его улучшения.

Оптимизация загрузки ресурсов является еще одним важным аспектом. ТМО используется для оптимизации загрузки ресурсов путем распределения заявок между различными каналами или обслуживающими устройствами, что способствует уменьшению простоев и повышению производительности.

Также ТМО используется для прогнозирования и планирования. С помощью моделей ТМО можно прогнозировать поведение системы в различных сце-

нарях и разрабатывать планы, учитывая возможные изменения в потоке заявок или доступности ресурсов.

Важным аспектом является снижение вероятности отказов в обслуживании. ТМО помогает анализировать и снижать вероятность отказов, что критически важно для систем с высокими требованиями к доступности и надежности.

Создание симуляционных моделей является еще одним аспектом использования ТМО. ТМО используется для создания симуляционных моделей реальных систем, что позволяет проводить эксперименты с моделями вместо их проведения на реальных системах, что может быть затратным или невозможным.

Знание принципов ТМО также помогает в проектировании и разработке архитектуры системы, обеспечивая их соответствие требованиям производительности и масштабируемости.

В данной главе рассматриваются принципы разработки многоканальной системы массового обслуживания (СМО) с ограниченной очередью, реализованной на платформе .NET с использованием языка программирования C#. Цель этой разработки – продемонстрировать универсальность и применимость инструментов .NET для различных типов СМО, включая многоканальные системы.

Для того чтобы исследовать универсальность и применимость методов, разработанных для одноканальной модели с эрланговски распределенным временем обслуживания заявок и ограниченной очередью, было решено использовать инструментарий .NET для более сложного случая – многоканальной СМО. Это позволило проверить эффективность и надежность подходов в условиях более сложной многоканальной системы.

Архитектурные решения для многоканальной системы базируются на общих принципах проектирования СМО, что включает использование высокопроизводительной обработки данных и параллельных вычислений. Эти подходы, хотя и не основаны напрямую на аналитических результатах, полученных для модели M/Er/1, позволяют применить аналогичные методы для решения задач многоканальных систем.

Разработка многоканальной системы массового обслуживания (СМО) с ограниченной очередью на платформе .NET является актуальной задачей, обусловленной несколькими ключевыми факторами. Высокая производительность многоканальных СМО позволяет обрабатывать несколько заявок одновременно, что увеличивает пропускную способность и уменьшает время ожидания для клиентов. Ограниченная очередь способствует оптимизации использования ресурсов и предотвращает их перегрузку. Гибкость и масштабируемость .NET делают его идеальным для разработки сложных систем, таких как многоканальные СМО.

Обработка больших данных является необходимым условием для современных систем обслуживания. .NET предлагает функции и инструменты для построения отказоустойчивых систем, обеспечивая безопасность данных и транзакций. Интеграция с другими системами упрощает работу с различными сервисами и повышает функциональные возможности системы. .NET также поддерживает кроссплатформенность, что позволяет разрабатывать приложения, работающие на различных операционных системах.

Таким образом, теория массового обслуживания является важным элементом в разработке и оптимизации систем обслуживания. Она предоставляет все необходимые инструменты для анализа характеристик системы, планирования ресурсов, улучшения качества услуг, минимизации затрат и рисков. Знание принципов ТМО помогает специалистам принимать обоснованные решения при проектировании и управлении системами обслуживания.

4.2 Принципы разработки многоканальной СМО с ограниченной очередью с применением parallel framework

Многоканальные системы массового обслуживания играют ключевую роль в различных сферах, включая телекоммуникации, банковское дело и здравоохранение. Эффективность таких систем часто определяется их способностью обрабатывать множество заявок параллельно. Рассмотрим принципы разработки многоканальной СМО с ограниченной очередью, используя Parallel Framework плат-

формы .NET [78]. Parallel Framework представляет собой часть платформы .NET, которая обеспечивает высокоуровневые инструменты и библиотеки для параллельного программирования. Он позволяет разработчикам легко внедрять многопоточность и параллелизм в своих приложениях, не работая с низкоуровневыми механизмами потоков.

На рисунке 4.2.1 изображена диаграмма деятельности [79] многоканальной СМО с ограниченной очередью, использующей Parallel Framework [80].

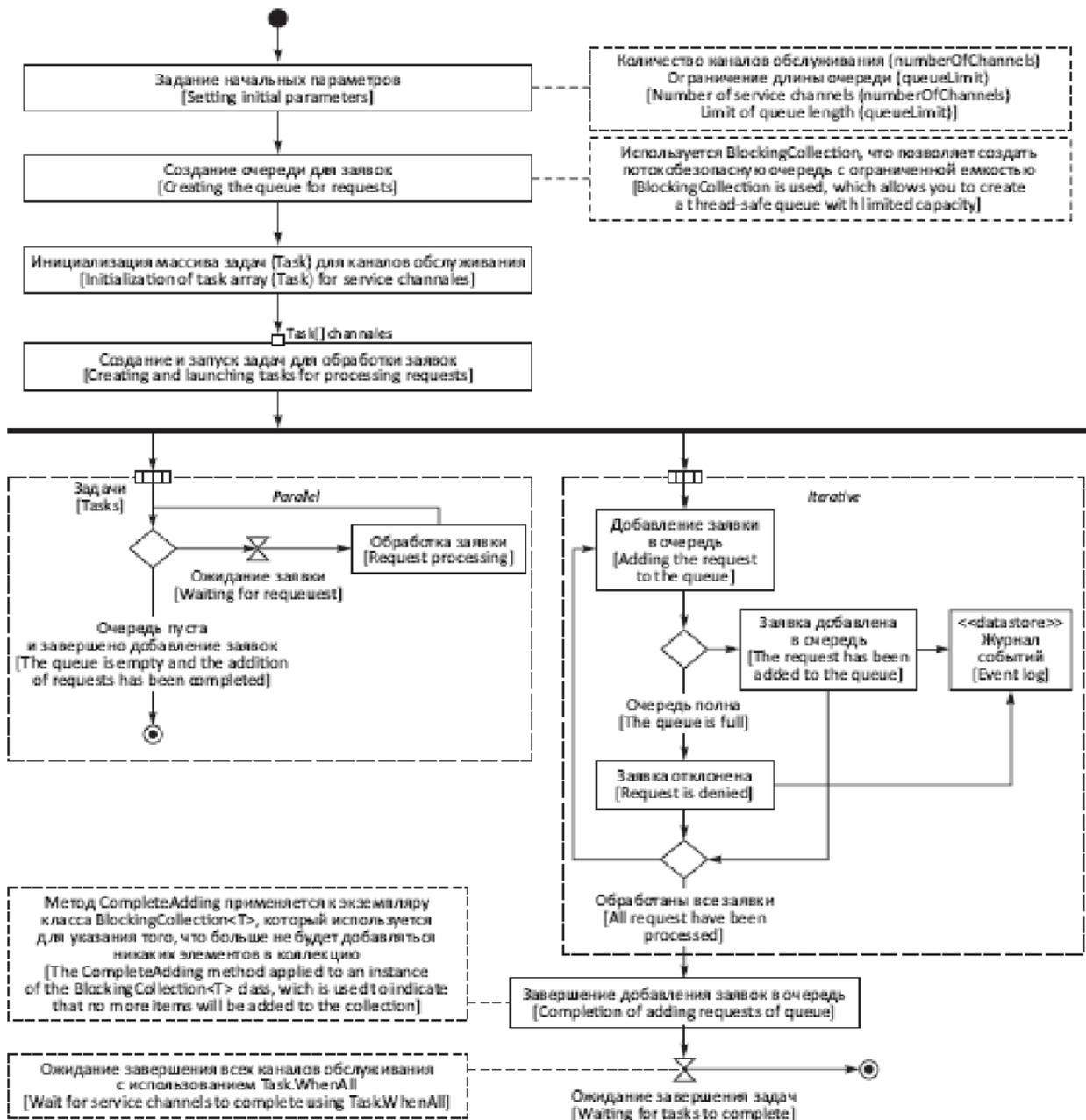


Рис. 4.2.1. Диаграмма деятельности многоканальной системы массового обслуживания с ограниченной очередью с использованием Parallel Framework

В данной системе имеется несколько каналов обслуживания и очередь с ограничением в N заявок. При добавлении заявок в очередь они обрабатываются параллельно в нескольких каналах. В случае достижения очередью своего предела новые заявки отклоняются.

Для реальных сценариев использования и оптимизации многоканальных СМО с ограниченной очередью [81; 82], можно рассмотреть несколько дополнительных аспектов:

- Балансировка нагрузки. Это включает оптимизацию распределения заявок между каналами с целью минимизации времени ожидания и обеспечения равномерной загрузки каналов.
- Динамическое масштабирование. Автоматическое изменение числа каналов в ответ на изменение интенсивности потока заявок.
- Мониторинг и логирование. Сбор статистических данных о производительности системы и времени обработки заявок для последующего анализа и оптимизации.

На рис. 4.2.2-4.2.3 приведены примеры кода с динамическим масштабированием каналов:

```

using System;
using System.Collections.Generic;
using System.Threading;
using System.Threading.Tasks;

class Program
{
    static int channelCount = 3; // Начальное количество каналов
    static int maxQueueSize = 10; // Максимальный размер очереди

    static async Task Main(string[] args)
    {
        // Запуск обработки заявок в каналах
        StartService();

        // Имитация динамического изменения нагрузки
        await Task.Delay(TimeSpan.FromSeconds(10));

        // Увеличение интенсивности потока заявок
        IncreaseLoad();

        // Остановка работы для демонстрации
        Console.ReadLine();
    }

    static void StartService()
    {
        for (int i = 0; i < channelCount; i++)
        {
            Task.Factory.StartNew(() => ProcessRequests());
        }
    }

    static async void ProcessRequests()
    {
        Random random = new Random();
        while (true)
        {
            // Имитация обработки заявки
            await Task.Delay(random.Next(100, 1000));

            // Убедимся, что очередь не превышает максимальный размер
            if (queue.Count < maxQueueSize)
            {
                // Добавление заявки в очередь
                queue.Enqueue(new Request());
                Console.WriteLine($"Заявка добавлена в очередь. Размер очереди: {queue.Count}");
            }
            else
            {
                Console.WriteLine("Очередь переполнена. Заявка отклонена.");
            }
        }
    }

    static Queue<Request> queue = new Queue<Request>();

    static void IncreaseLoad()
    {
        // Увеличение интенсивности потока заявок и динамическое масштабирование
        channelCount++;
        Console.WriteLine

```

Рис. 4.2.2 Код с динамическим масштабированием каналов

```

Task monitor = Task.Factory.StartNew(() =>
{
while (!queue.IsCompleted)
{
if (queue.Count > queueLimit * 0.8 && numberOfChannels < 5)
{
// Добавление дополнительного канала
Task additionalChannel = Task.Factory.StartNew(() =>
{
// ...код обработки заявок аналогичный предыдущим каналам
});
Console.WriteLine("Дополнительный канал добавлен");
numberOfChannels++;
}
Task.Delay(5000).Wait(); // Проверка каждые 5 секунд
}
});

```

Рис. 4.2.3 Код с динамическим масштабированием каналов (конец)

Таким образом, использование Parallel Framework платформы .NET позволяет разрабатывать высокопроизводительные многоканальные системы массового обслуживания с ограниченной очередью. Средства параллельного программирования облегчают реализацию многопоточности, а применение техник балансировки нагрузки, динамического масштабирования и мониторинга позволяет оптимизировать систему для конкретных потребностей и условий работы.

4.3 Применение конкурентных коллекций

При проектировании многоканальной СМО с ограниченной очередью с использованием Parallel Framework на платформе .NET, одним из важных аспектов является управление конкуренцией за доступ к разделяемым ресурсам. В многопоточных приложениях, когда несколько потоков имеют доступ к общим данным, необходимо обеспечить синхронизацию и надежное управление данными. В этом контексте, конкурентные коллекции .NET предлагают эффективные решения. Конкурентные коллекции, такие как `ConcurrentQueue<T>` и `ConcurrentBag<T>`, предоставляют потокобезопасные структуры данных, которые могут быть эффективно использованы в многопоточных сценариях. В контексте СМО `ConcurrentQueue<T>` может быть использована для представления очереди заявок.

Она обеспечивает потокобезопасную очередь FIFO (первым пришел – первым обслужен). Parallel Framework предоставляет различные средства для распараллеливания задач. Для обработки заявок в многоканальной СМО можно использовать Parallel.For или Parallel.ForEach (рис. 4.3.1).

```
int numberOfChannels = 3; // Количество каналов обслуживания
Parallel.For(0, numberOfChannels, i =>
{
    // Код обработки заявок для каждого канала
    Request request;
    while (requestQueue.TryDequeue(out request))
    {
        // Обработка заявки
    }
});
```

Рис. 4.3.1 Код обработки заявок с использованием Parallel.For

При использовании ConcurrentQueue, можно реализовать логику для ограничения размера очереди, проверяя ее размер перед добавлением новой заявки (рис. 4.3.2).

```
if (requestQueue.Count < maxQueueSize)
{
    requestQueue.Enqueue(newRequest);
}
else
{
    // Отказ или перенаправление заявки
}
```

Рис. 4.3.2 Код обработки заявок с ограничением размера очереди

Использование конкурентных коллекций вместе с Parallel Framework на платформе .NET позволяет эффективно реализовывать многоканальные СМО с ограниченной очередью. Конкурентные коллекции обеспечивают потокобезопасное управление данными, в то время как Parallel Framework позволяет легко распараллеливать обработку заявок по множеству каналов. Это не только улучшает

производительность системы, но и обеспечивает более высокую точность моделирования в средах с высокой степенью параллелизма.

Другой важный аспект – это эффективное распределение ресурсов между каналами (рис.4.3.3). Можно использовать Parallel Framework для динамического балансирования нагрузки между каналами обслуживания.

```
Parallel.ForEach(requestQueue, new ParallelOptions { MaxDegreeOfParallelism =  
    numberOfChannels }, (request) =>  
    {  
        // Код обработки заявки  
    });
```

Рис. 4.3.3 Код обработки заявок с распределением ресурсов между каналами

Такой подход позволяет системе автоматически адаптироваться к изменениям в потоке заявок и распределять ресурсы между каналами наиболее эффективным образом. Помимо обработки заявок, важно также реализовать механизмы мониторинга и анализа производительности системы. Это может включать в себя сбор статистики о времени ожидания в очереди, времени обслуживания и доле отказов. Эти данные могут быть использованы для последующего анализа и оптимизации системы.

4.4 Результаты тестирования

Было проведено сравнительное тестирование двух реализаций системы массового обслуживания 10 заявок по сокрытию файлов (объемом ~600 КБ) с применением приложения для ассоциативной защиты файлов, представленного в статье [83]: с использованием Parallel Framework и без его использования. Тестирование проводилось на процессоре Intel Core i5-9300H с базовой частотой 2,4 ГГц под управлением ОС Windows 11 Pro. Исходя из результатов тестирования, среднее время обработки одной заявки в реализации с использованием Parallel Framework составило 69 678 миллисекунд, в то время как в реализации без использования

Parallel Framework – 240 345 миллисекунд. Это указывает на улучшение производительности в 3,45 раз при использовании Parallel Framework. Однако, следует отметить, что Parallel Framework эффективен только при определенных условиях. При небольшом числе заявок и низкой сложности обработки заявок, затраты на управление параллельными потоками могут уменьшить преимущества многопоточности. Кроме того, существует риск высокой конкуренции за ресурсы, что может привести к снижению производительности при обработке большого объема данных. Таким образом, рекомендуется применять Parallel Framework для задач с высокой вычислительной сложностью и большим объемом данных, где параллелизация может существенно уменьшить время выполнения. В то же время, для более простых задач, следует оценить возможные выгоды и определить, является ли использование многопоточности оптимальным решением.

Выводы:

Реализация многоканальной системы массового обслуживания с ограниченной очередью на платформе .NET с применением Parallel Framework и конкурентных коллекций позволяет создать высокопроизводительное и масштабируемое решение. Использование потокобезопасных структур данных обеспечивает надежность системы в многопоточной среде, в то время как возможности параллелизма позволяют эффективно использовать ресурсы и адаптировать систему к изменяющимся условиям работы. С помощью Parallel Framework предложен новый подход к распределению заявок по каналам, что способствует более эффективной обработке. В отличие от традиционных методов, которые могут быть ограничены однопоточной обработкой, этот подход позволяет распараллеливать задачи, оптимизируя использование многоядерных процессоров. Parallel Framework обеспечивает динамическое распределение нагрузки, что позволяет системе автоматически адаптироваться к изменениям в потоке заявок. Это ведет к более высокой пропускной способности и меньшим временам ожидания заявок. Дальнейшее развитие многоканальной системы массового обслуживания (СМО) с ограниченной очередью на платформе .NET может быть направлено на усовер-

шенствование производительности, гибкости и аналитических возможностей системы. Приведем несколько направлений, которые могут быть рассмотрены:

1. Адаптивные алгоритмы могут быть применены для автоматической настройки числа каналов и размера очереди на основе наблюдаемой нагрузки. Это позволит системе автоматически реагировать на изменения в потоке заявок, оптимизируя использование ресурсов.

2. Интеграция с облачными сервисами, такими как Azure, может обеспечить масштабируемость системы на новом уровне. Это может включать в себя возможность динамического масштабирования ресурсов и использования облачных сервисов для хранения и обработки данных.

3. Расширение аналитических возможностей системы путем внедрения продвинутых инструментов мониторинга и анализа данных. Это может помочь в выявлении узких мест, предсказании будущих трендов и оптимизации процессов обслуживания.

4. Применение машинного обучения для предсказания потока заявок и оптимизации производительности системы. Это может включать в себя алгоритмы прогнозирования временных рядов, обучение с подкреплением и другие техники.

5. Применение архитектуры микросервисов может обеспечить большую гибкость и расширяемость системы, позволяя добавлять новые функции и сервисы без необходимости перестраивать всю систему.

ЗАКЛЮЧЕНИЕ

Основные результаты, полученные в данной работе, опубликованы в трудах [84-95] и заключаются в следующем.

1. Разработана новая математическая модель СМО смешанного типа, отличающаяся от известных моделей сочетанием свойств СМО с ограниченным временем ожидания заявки в очереди и СМО с временем обслуживания, распределенным по закону Эрланга. Строгими аналитическими методами получены общие формулы для характеристик модели в стационарном и нестационарном режимах, позволяющие задавать временные рамки нахождения заявок в очереди и количество фаз обслуживания.

2. Предложен новый метод исследования СМО с эрланговски распределенным временем обслуживания, позволяющий найти требуемое количество фаз, обеспечивающее заданный уровень качества обслуживания, которое может быть оценено с помощью коэффициента вариации различных характеристик СМО. Выявлено, что граничные значения числа фаз обслуживания логарифмически связаны с изменением приведенной интенсивности входного потока заявок.

3. Разработана имитационная модель СМО смешанного типа с использованием инструментальной среды AnyLogic PLE, а также метамодель, аппроксимирующая ключевые характеристики СМО на основе методов машинного обучения. Метамодель позволяет оперативно оценивать показатели эффективности системы по заданным входным параметрам без проведения ресурсоёмких имитационных экспериментов, что значительно ускоряет процесс анализа.

4. Разработан комплекс специализированных программных средств для решения задач по расчету ключевых характеристик СМО и оценки её эффективности. Проведено сравнительное тестирование двух реализаций СМО 10 заявок по сокрытию файлов (объемом ~600 КБ) с применением приложения для ассоциативной защиты файлов с использованием библиотеки Parallel Framework и без ее использования. Показано, что при использовании Parallel Framework производительность системы повышается в 3,45 раз.

С точки зрения теоретического аспекта, предложенный метод исследования СМО может быть расширен на другие немарковские модели СМО, что представляет собой перспективное направление для развития теоретических основ данной области.

С практической точки зрения, полученные результаты открывают ряд перспективных возможностей для управления сложными производственными системами. Использование этих результатов позволит более точно оценивать характеристики очередей и временные задержки в таких системах.

СПИСОК ЛИТЕРАТУРЫ

1. Вентцель Е.С. Теория вероятностей / Е.С. Вентцель. - М.: Наука, 1969. - 576 с.
2. Саати Т.Л. Элементы теории массового обслуживания и ее приложения / Т.Л. - М.: Советское радио, 1971.- 510 с.
3. Вентцель Е.С. Исследование операций / Е.С. Вентцель. - М.: Советское радио, 1972. - 552 с.
4. Ивченко Г.И. Теория массового обслуживания / Г.И. Ивченко, В.А. Каштанов, И.Н. Коваленко. – М.: Высшая школа, 1982. - 256 с.
5. Климов Г.П. Теория массового обслуживания / Г.П. Климов. - М.: Изд-во Московского университета, 2011.- 312 с.
6. Феллер В. Введение в теорию вероятностей и её приложения / В. Феллер. - М.: Мир, 1964. - 498 с.
7. Д. Кендалл, “Стохастические процессы, встречающиеся в теории очередей, и их анализ методом вложенных цепей Маркова”, Математика, 3:6 (1959), 97–112; Ann. Math. Statistics, 24 (1953), 338–354
8. Лихтциндер Б. Я. О классификации Кендалла //Инфокоммуникационные технологии. – 2020. – Т. 18. – №. 3. – С. 328-332.
9. Clark P. J., Evans F. C. Distance to nearest neighbor as a measure of spatial relationships in populations //Ecology. – 1954. – Т. 35. – №. 4. – С. 445-453.
10. Karlin S. et al. Coincidence probabilities //Pacific Journal of Mathematics. – 1959. – Т. 9. – №. 4. – С. 1141-1164.
11. Cohen J.W., Harkema P. A study of the delay encountered in telegraph timerelay switching, *Commun. News*, **15**, (1954), p. 47.
12. Cohen J.W. Berechnung der verkehrsgrossen in wartezeitsystem aus den verkehrsgrossen eines verlustsystems, *NTZ – Nachrtech. Z.*, **8**, (1955), p. 139.
13. Cohen J.W. Das warteproblem fur das vollkommene bundle mit einer endlichen quellenzahl, *NTZ – Nachrtech. Z.*, **8**, (1955), 641–645.

14. Cohen J.W. Some examples in the use of implication in switching algebra, *Commun. News*, **16**, No. 1 (1955), 1 – 10.
15. Cohen J.W. On the queueing process of lanes, *Philips Tech. Rept.*, (1956).
16. Cohen J.W. Certain delay problems for a full availability trunk group loaded by two sources, *Communication News*, **16**, No. 3 (1956), 105–113.
17. Cohen J.W. Basic problems of telephone traffic theory and the influence of repeated calls, *Philips Telecommun. Rev.*, **18**, No. 2 (1957), 49-100.
18. Cohen J.W. The full availability group of trunks with an arbitrary distribution of the interarrival times and a negative exponential holding time distribution, *Natuurk. Tydschr. (Ghent)*, **26**, No. 4 (1957), 169 – 181.
19. Cohen J.W. A survey of queueing problems occurring in telephone and telegraph traffic theory, *In Proc. 1st Int. Conf. Operat. Res., Oxford. English Universities Press, London*, (1957). 138–146
20. Cohen J.W. The generalized Engset formulae, *Philips Telecommun. Rev.*, **18**, No. 4 (1957), 158 – 170.
21. Cohen J.W. On the fundamental problem of telephone traffic theory and the influence of repeated calls, *Philips Telecommun. Rev.*, **18**, (1957), 49 – 100.
22. Cohen J.W., Beukelman B.J. Call congestion of transposed multiples, *Philips Telecommun. Rev.*, **17**, No. 4 (1957), 145 – 154.
23. Takagi H., Explicit delay distribution in first-come first-served M/M/m/K and M/M/m/K/n queues and mixed loss-delay system, *International Journal of Pure and Applied Mathematics*, **40** No. 2 (2007), 185–200.
24. Takagi H., Explicit delay distribution in first-come first-served M/M/m/K and M/M/m/K/n queues and mixed loss-delay system, *Proc. Asia-Pacific Symposium on Queuing Theory and Its Application to Telecommunication Networks*, (2006) 1–11.
25. Алиев Т.И., Муравьева-Витковская Л.А. Приоритетные стратегии управления трафиком в мультисервисных компьютерных сетях // Известия высших учебных заведений. Приборостроение. 2011. Т. 54. № 6. С. 44-48.
26. Алиев Т.И. Моделирование ядра мультисервисной сети с относительной приоритизацией неоднородного трафика / Т.И. Алиев, И.Е. Никульский,

В.О. Пяттаев // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики. 2009. № 4 (62). С. 88-96.

27. Задорожный В.Н. Минимизация потерь в сетях с фрактальным трафиком / В.Н. Задорожный, Т.Р. Захаренкова, М.П. Маркова // В книге: Математическое и компьютерное моделирование сборник материалов VI Международной научной конференции, посвященной памяти Б.А. Рогозина. Омск, 2018. С. 54-57.

28. Zadorozhnyi V.N. Estimation of prioritized disciplines efficiency based on the metamodel of multi-flows queueing systems / V.N. Zadorozhnyi, T.R. Zakharenkova, D.A. Tulubaev // Communications in Computer and Information Science. 2018. T. 912. . 290-304.

29. Назаров А.А., Моисеева С.П. Метод асимптотического анализа в теории массового обслуживания. Томск: Изд-во НТЛ, 2006. 109 с.

30. Самуйлов К.Е. К разработке исходных данных для сценария приоритетного управления доступом в мультисервисной сети LTE / К.Е. Самуйлов, И.А. Гудкова, Е.В. Маркова // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 10. С. 22-27.

31. Ботвинко А.Ю., Самуйлов К.Е. Математическая модель работы меж-сетевого экрана для мультимедийного трафика // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 12. С. 56-60.

32. Тарасов В.Н. Математическая модель телетрафика на основе системы HE2/M/1 / В.Н. Тарасов, Н.Ф. Бахарева, О. Када // Информационные технологии. 2019. Т. 25. № 4. С. 205-210.

33. Тарасов В.Н. Новые модели массового обслуживания для анализа трафика с широким диапазоном изменения параметров // В сборнике: Перспективные информационные технологии (ПИТ 2018) Труды Международной научно-технической конференции. Под редакцией С.А. Прохорова. 2018. С. 1106-1110.

34. Тарасов В.Н. Модели телетрафика на основе современной теории массового обслуживания / В.Н. Тарасов, Н.Ф. Бахарева, Э.Г. Ахметшина // Инфокоммуникационные технологии. 2018. Т. 16. № 1. С. 68-74.

35. Тарасов В.Н. Моделирование телетрафика в случае широкого диапазона изменения его параметров на основе теории массового обслуживания / В.Н. Тарасов, Н.Ф. Бахарева, Л.В. Липилина // В сборнике: Новые информационные технологии и системы сборник научных статей XIV Международной научно-технической конференции, посвященной 70-летию кафедры «Вычислительная техника» и 30-летию кафедры «Системы автоматизированного проектирования». 2017. С. 88-91.

36. Тарасов В.Н., Бахарева Н.Ф. Модели сетевого трафика для широкого диапазона изменения его параметров // В сборнике: II научный форум телекоммуникации: теория и технологии ТТТ-2017. Проблемы техники и технологий телекоммуникаций ПТиТТ-2017 материалы XVIII Международной научно-технической конференции. 2017. С. 90-92.

37. Шнепс-Шнеппе М.А. Телекоммуникации и умный дом для модернизации экономики России // Электросвязь. 2010. №5. С. 1–5.

38. Шнепс М.А. О применении цепей Маркова для изучения телефонных систем с потерями // Проблемы передачи информации: сборник трудов. 1963. № 12. С. 124 – 134.

39. Adan I. Queueing Systems / I. Adan, J. Resing. – Eindhoven, The Netherlands: Eindhoven University of Technology, 2015. – 182 p.

40. Courtois P. J., Scheys G. Minimization of the total loss rate for two finite queues in series //IEEE transactions on communications. – 1991. – Т. 39. – №. 11. – С. 1651-1661.

41. Jayasuriya A., Green D., Asenstorfer J. Modelling service time distribution in cellular networks using phase-type service distributions //ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01CH37240). – IEEE, 2001. – Т. 2. – С. 440-444.

42. Neuts, Marcel F. Matrix-geometric solutions in stochastic models: an algorithmic approach. Courier Corporation, 1994.

43. Jain S. Estimation in M/Er/1 queueing systems //Communications in Statistics-Theory and Methods. – 1991. – Т. 20. – №. 5-6. – С. 1871-1879.

44. Кирпичников А.П. Прикладная теория массового обслуживания / А.П. Кирпичников. - Казань, Изд-во КГУ, 2008.- 112 с.
45. Бочаров П.П. Теория массового обслуживания / П.П. Бочаров, А.В. Печинкин. М.: Изд-во РУДН, 1995. 529 с.
46. Antonova P., Titovtsev A. On Real Queue Length in a Queueing System with Erlang-r service time //2020 4th Scientific School on Dynamics of Complex Networks and their Application in Intellectual Robotics (DCNAIR). – IEEE, 2020. – С. 37-41.
47. Кирпичников А.П. Методы прикладной теории массового обслуживания / А.П. Кирпичников. - Казань, Изд-во Казанского университета, 2011. – 200 с.
48. Hartman P. Tauber's theorem and absolute constants //American Journal of Mathematics. – 1947. – Т. 69. – №. 3. – С. 599-606.
49. Bingham N. H. Tauberian theorems and the central limit theorem //The Annals of Probability. – 1981. – С. 221-231.
50. Якимив А. Л. Вероятностные приложения тауберовых теорем. – Физматлит, 2005
51. Якимив А. Л. Тауберова теорема для кратных степенных рядов //Математический сборник. – 2016. – Т. 207. – №. 2. – С. 143-172.
52. Шкаликов А. А. Теоремы тауберова типа о распределении нулей голоморфных функций //Математический сборник. – 1984. – Т. 123. – №. 3. – С. 317-347.
53. Кирпичников А.П., Титовцев А.С. Характеристики систем дифференцированного обслуживания поликомпонентных потоков //Вестник Казанского технологического университета. – 2012. – Т. 15. – №. 8. С. 337-340.
54. Кирпичников А. П., Титовцев А.С. Первый и второй моменты общего времени пребывания заявки в системе с произвольным количеством источников и поликомпонентных входным потоком заявок //Вестник Казанского технологического университета. – 2016. – Т. 19. – №. 6. С. 120-124.
55. Н.И. Головкин, Дисс. д. техн. наук, ТГЭУ, Владивосток, 2007. 404 с.

56. Дёч Г. Руководство к практическому применению преобразования Лапласа и Z-преобразования. – Рипол Классик, 1971.
57. Zhou Y., Peng L., Huang Y. Duhamel's formula for time-fractional Schrödinger equations //Mathematical Methods in the Applied Sciences. – 2018. – Т. 41. – №. 17. – С. 8345-8349.
58. Bourdin L. Cauchy-Lipschitz theory for fractional multi-order dynamics: State-transition matrices, Duhamel formulas and duality theorems //Differential and Integral Equations. – 2018. – Т. 31. – №. 7/8. – С. 559-594.
59. Коган И. Л. Метод интеграла Дюамеля для обыкновенных дифференциальных уравнений с постоянными коэффициентами с точки зрения теории обобщенных функций //Вестник Самарского государственного технического университета. Серия Физико-математические науки. – 2010. – №. 1 (20).
60. Preumont A. Frequency domain analysis of time integration operators //Earthquake engineering & structural dynamics. – 1982. – Т. 10. – №. 5. – С. 691-697.
61. Droniou J., Imbert C. Fractal first-order partial differential equations //Archive for Rational Mechanics and Analysis. – 2006. – Т. 182. – №. 2. – С. 299-331.
62. Шонин М. Ю. Преобразование Лапласа при решении линейных интегро-дифференциальных уравнений //Научный поиск в современном мире. – 2016. – С. 11-14.
63. Порошина Н. И., Рябов В. М. О методах обращения преобразования Лапласа // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2011. №3.
64. Рамазанов Р. Р. Сравнительная характеристика подходов имитационного моделирования общественных процессов // Вестник УГНТУ. Наука, образование, экономика. Серия: Экономика . 2017. №2 (20). URL: <https://cyberleninka.ru/article/n/sravnitelnaya-harakteristika-podhodov-imitatsionnogo-modelirovaniya-obschestvennyh-protssesov> (дата обращения: 09.12.2021).
65. Снетков Н.Н. Имитационное моделирование экономических процессов: Учебно-практическое пособие. – М.: Изд. центр ЕАОИ, 2008. – 228 с.

66. Юданова В.В. Имитационное моделирование систем массового обслуживания // Интернет-журнал «Отходы и ресурсы», 2019 №4, <https://resources.today/PDF23INOR419.pdf> (доступ свободный).

67. Якимов И.М., Кирпичников А.П., Мокшин В.В. Моделирование сложных систем в имитационной среде AnyLogic // Вестник Казанского технологического университета. 2014. №13. URL: <https://cyberleninka.ru/article/n/modelirovanie-slozhnyh-sistem-v-imitatsionnoy-srede-anylogic> (дата обращения: 09.12.2020).

68. Мейханаджян Л. А., Зарядов И. С., Милованова Т. А. Стационарные характеристики двухузловой марковской системы массового обслуживания с обобщенным обновлением // Системы и средства информатики. – 2020. – Т. 30. – №. 3. – С. 14-31.А

69. Горбунова А. В., Лебедев А. В. Система массового обслуживания с двумя входящими потоками, абсолютным приоритетом и стохастическим сбросом // Автоматика и телемеханика. – 2020. – №. 12. – С. 111-128.

70. Нуриев Н. К., Печеный Е. А., Старыгина С. Д. Математическое моделирование системы массового обслуживания с каналами разной производительности // Современные наукоемкие технологии. – 2021. – №. 1. – С. 31-36.

71. Милованова Т. А., Разумчик Р. В. Однолинейная система массового обслуживания с инверсионным порядком обслуживания с вероятностным приоритетом, групповым пуассоновским потоком и фоновыми заявками // Информатика и её применения. – 2020. – Т. 14. – №. 3. – С. 26-34.

72. Кирпичников А.П. Вероятностные характеристики открытой многоканальной системы массового обслуживания с ограниченным средним временем пребывания в очереди / А.П. Кирпичников, Нгуен Тхань Банг, Чан Куанг Куи // Вестник Казанского технологического университета - 2016. - Т. 19. - № 8. С. 123-126.

73. Кирпичников А.П. Вероятность ожидания начала обслуживания в системе массового обслуживания с ограниченным средним временем пребывания

заявки в очереди / А.П. Кирпичников, Нгуен Тхань Банг, Чан Куанг Куи // Вестник Казанского технологического университета. - 2016. - Т. 19. - № 21. С. 144-147.

74. Кирпичников А.П. Расчёт коэффициента загрузки системы массового обслуживания с ограниченным средним временем пребывания заявки в очереди / А.П. Кирпичников, Нгуен Тхань Банг, Чан Куанг Куи // Вестник Казанского технологического университета. - 2017. - Т. 20. - № 2. С. 88-92.

75. Кирпичников А.П. Среднее число заявок в очереди на обслуживание в системе массового обслуживания с ограниченным средним временем пребывания заявки в очереди / А.П. Кирпичников, Нгуен Тхань Банг, Чан Куанг Куи // Вестник Казанского технологического университета - 2017. - Т. 20. - № 6. С. 87-92.

76. Кирпичников А.П. Общее число требований, находящихся в системе массового обслуживания с ограниченным средним временем пребывания заявки в очереди / А.П. Кирпичников, Нгуен Тхань Банг, Чан Куанг Куи // Вестник Казанского технологического университета. - 2017. - Т. 20. - № 9. С. 93-96.

77. А.С. Титовцев, Дисс. канд. техн. наук, КНИТУ, Казань, 2011. 143 с.

78. Gibadullin R.F., Nikonorov V.V. Development of the System for Automated Incident Management Based on Open-Source Software // IEEE International Russian Automation Conference (RusAutoCon). 2021. Pp. 521–525.

79. Weerasinghe D.N.H. et al. Smart UML – Assignment Management Tool for UML Diagrams // IEEE 4th International Conference on Advancements in Computing (ICAC). 2022. Pp. 114–119.

80. Garcia A.M. et al. Introducing a Stream Processing Framework for Assessing Parallel Programming Interfaces // IEEE 29th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). 2021. Pp. 84–88.

81. Осипов Г.С. Исследование систем массового обслуживания с ожиданием в AnyLogic // Бюллетень науки и практики. 2016. № 10. С. 139–151.

82. Осипов Г.С. Системы массового обслуживания с ограниченной длительностью ожидания // Бюллетень науки и практики. 2016. № 12. С. 28-36.

83. Гибадуллин Р.Ф., Вершинин И.С., Глебов Е.Е. Разработка приложения для ассоциативной защиты файлов // Инженерный вестник Дона. 2023. № 6. URL: ivdon.ru/ru/magazine/archive/n6y2023/8462/

84. Антонова П.В. Исследование систем массового обслуживания смешанного типа // Навигатор в мире науки и образования. 2021. № 4 (53). Институт программных систем им. А.К. Айламазяна РАН. - С. 381-385.

85. Титовцев А.С., Антонова П.В. Характеристики системы массового обслуживания с ограниченным временем пребывания заявки в очереди и временем обслуживания, распределенным по закону Эрланга // Научно-технический вестник Поволжья. 2021. № 8. С. 79-82.

86. Титовцев А.С., Антонова П.В. Численное исследование стабильных режимов работы систем массового обслуживания с ограниченным средним временем пребывания заявки в системе и временем обслуживания, распределенным по закону Эрланга // Научно-технический вестник Поволжья. 2021. № 8. С. 83-85.

87. Антонова П.В., Титовцев А.С. Характеристики нестационарного режима систем массового обслуживания смешанного типа с ограниченным временем пребывания заявки в очереди // Научно-технический вестник Поволжья. 2021. № 12. С. 171-173.

88. Антонова П.В. Разработка многоканальной системы массового обслуживания с ограниченной очередью с применением параллельной библиотеки платформы .NET // Computational Nanotechnology. 2023. Т. 10. № 3. С. 44-50.

89. Антонова П.В., Гиздатуллин Р.М., Титовцев А.С., Казаков М.В. Разработка метамоделей системы массового обслуживания с использованием ансамблевых методов машинного обучения // Научно-технический вестник Поволжья. 2025. № 2. С. 18-21.

90. Антонова П.В., Казаков М.В., Гиздатуллин Р.М. Разработка метамоделей системы массового обслуживания с использованием ансамблевых методов машинного обучения // Перспективные фундаментальные исследования и научные методы: сборник статей международной научной конференции. – СПб.: МИ-ПИ им.Ломоносова, 2024. С. 29-31.

91. Антонова П.В. Численное исследование стабильных режимов работы СМО // Перспективные исследования в технических и естественных науках: сборник статей международной научной конференции. – СПб.: МИПИ им.Ломоносова, 2024. С. 17-18.

92. Антонова П.В. Принципы разработки систем массового обслуживания с ограниченной очередью на платформе .NET // Программные системы и вычислительные методы. 2023. № 2. С. 15-28.

93. Антонова П.В. Исследование систем массового обслуживания смешанного типа // Навигатор в мире науки и образования. 2021. № 4 (53). С. 381-385.

94. Антонова П.В. Свидетельство о регистрации электронного ресурса №24942 «Исследование систем массового обслуживания смешанного типа». М.: ОФЭРНиО, 2021.

95. Антонова П.В., Гиздатуллин Р.М., Казаков М.В. Свидетельство о регистрации электронного ресурса №25456 «Ансамблевая метамодел для оценки параметров СМО». М.: ОФЭРНиО, 2025.

ПРИЛОЖЕНИЯ

УТВЕРЖДАЮ

Генеральный директор
ООО «Систематика Консалтинг»

В.А. Злобин



«17» мая 2021г.

АКТ ВНЕДРЕНИЯ

Настоящим актом подтверждается, что результаты диссертационного исследования Антоновой П.В. использовались при разработке федеральной государственной информационной системы территориального планирования (ФГИС ТП). в части организации и функционирования центра приема обращений службы технической поддержки. Представленная в диссертационной работе модель позволяет описать функционирование подобного рода объектов, поскольку в ней учитывается прохождение заявкой нескольких фаз в обслуживании и тот факт, что заявки могут уходить из системы, не дождавшись начала обслуживания.

Применение результатов диссертационного исследования позволило Компании ООО «Систематика Консалтинг» обеспечить стабильное функционирование технической поддержки ФГИС ТП.

Руководитель департамента
комплексных информационных систем
ООО «Систематика Консалтинг»

A handwritten signature in blue ink, appearing to be 'Спивак В.Ф.', written over a horizontal line.

Спивак В.Ф.

АКТ ВНЕДРЕНИЯ

Настоящим актом подтверждается, что результаты диссертационного исследования Антоновой П.В. использовались ООО «НПО ЛАБС» при разработке программно-аппаратного решения для мониторинга и оптимизации производственных процессов в части реализации учета оборудования предприятия, добавления неограниченного количества единиц оборудования, объединения в группы, отслеживания режимов простоя и работы.

Внедрение данного решения на практике показало высокую эффективность аналитической модели СМО, которая была разработана Антоновой П.В.

Генеральный директор
ООО «НПО ЛАБС»



Сеник А.Я



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ ИМ. А.К. АЙЛАМАЗЯНА
РОССИЙСКОЙ АКАДЕМИИ НАУК
Объединенный фонд электронных ресурсов «Наука и образование»
(основан в 1991 году)

**СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ
ЭЛЕКТРОННОГО РЕСУРСА**

№24942

**Исследование систем массового обслуживания смешанного
типа**

Дата регистрации: 17 декабря 2021 года

Автор: Антонова П.В.

Директор Института
чл.-корр. РАН  С.М.Абрамов

Руководитель ОФЭРНИО
почетный работник
науки и техники Российской
Федерации  А.И.Галкина





**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
ФЕДЕРАЛЬНАЯ ИННОВАЦИОННАЯ ПЛОЩАДКА
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ ИМ. А.К. АЙЛАМАЗЯНА
РОССИЙСКОЙ АКАДЕМИИ НАУК
Объединенный фонд электронных ресурсов «Наука и образование»
(основан в 1991 году)

**СВИДЕТЕЛЬСТВО О РЕГИСТРАЦИИ
ЭЛЕКТРОННОГО РЕСУРСА С ОЦЕНКОЙ НОВИЗНЫ**

№25456

Ансамблевая метамодель для оценки параметров СМО

Дата регистрации: 05 февраля 2025 года

Авторы: Антонова П.В., Гиздатуллин Р.М., Казаков М.М.



И.о. директора
Института _____ И.П. Тищенко

Руководитель ОФЭРНИО
почетный работник
науки и техники Российской
Федерации _____ А.И.Галкина