

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,
СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича»

На правах рукописи

Савельева Анастасия Андреевна

**РАЗРАБОТКА МОДЕЛИ, МЕТОДА И АЛГОРИТМА
УПРАВЛЕНИЯ ИНФОРМАЦИОННЫМИ ПОТОКАМИ
РАСПРЕДЕЛЕННОЙ ТЕЛЕКОММУНИКАЦИОННОЙ СИСТЕМЫ
В ИНФРАСТРУКТУРЕ ТУМАННЫХ ВЫЧИСЛЕНИЙ**

2.2.15 – Системы, сети и устройства телекоммуникаций

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
кандидат технических наук, доцент
Елагин Василий Сергеевич

Санкт-Петербург – 2026

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1 Архитектурные особенности и требования к подсистеме распределенного хранения данных в телекоммуникационных системах граничных и туманных вычислений	14
1.1 Граничные и туманные вычисления в телекоммуникационных системах	18
1.1.1 Предпосылки переноса операций доступа к данным к периферии инфраструктуры	20
1.1.2 Анализ отечественных и зарубежных источников по теме.....	21
1.2 Обоснование выбора архитектуры подсистемы хранения данных	25
1.3 Архитектура распределенной подсистемы хранения и базовые процессы доступа к данным	33
1.3.1 Архитектурная модель	33
1.3.2 Концептуальное представление данных и единиц размещения при репликации.....	38
1.3.3 Процесс записи.....	41
1.4 Подходы к балансировке нагрузки в распределенных системах.....	46
1.4.1 Классификация по месту принятия решения	47
1.4.2 Классификация по степени использования информации о состоянии	49
1.4.3 Балансировка по времени прохождения запросов	50
1.4.4 Модели обслуживания потоков и распределение ресурсов как аналитическая основа балансировки	51
1.4.5 Подходы к балансировке нагрузки	53
1.5 Выводы	59

2	Разработка модели управления информационными потоками распределенной подсистемы хранения данных туманного уровня.....	61
2.1	Метод оценки готовности выполнения операции записи как ограничение при оптимизации производительности	62
2.1.1	Расчет коэффициента готовности	65
2.2	Выбор показателей производительности	85
2.3	Выбор ключевых метрик	89
2.3.1	Модель обслуживания и динамика очередей.....	91
2.3.2	Сетевые ограничения и межузловой обмен	93
2.3.3	Вывод по разделу	93
2.4	Управляющие воздействия и допустимая область решений.....	94
2.4.1	Состав управляющих воздействий.....	95
2.4.2	Ограничения на управляющие воздействия.....	97
2.4.3	Формирование допустимой области решений.....	99
2.5	Постановка задачи оптимизации	100
2.5.1	Показатели качества и целевая функция	101
2.5.2	Переменные управления	102
2.5.3	Ограничения	102
2.6	Выводы.....	105
3	Разработка алгоритма выбора узлов хранения при выполнении операции записи с репликацией данных	107
3.1	Алгоритм выбора узлов хранения и регулирования служебной нагрузки при выполнении операции записи	108
3.2	Реализуемость, наблюдаемость и вычислительная сложность.....	116
3.3	Анализ свойств алгоритма и ожидаемого эффекта	120
3.3.1	Механизм повышения производительности	125
3.4	Сравнительная оценка стратегий выбора узлов хранения и регулирования служебной нагрузки	127
	<i>Равномерное распределение</i>	129
	<i>Диспетчеризация по минимальной пользовательской очереди</i>	130

<i>Пороговое ограничение служебной нагрузки</i>	131
<i>Предложенный алгоритм интегрального выбора узлов</i>	132
3.5 Методика сопоставительного эксперимента.....	136
3.5.1 Ожидаемые результаты	140
3.6 Выводы	141
4 Экспериментальное исследование и проверка предложенного подхода	144
4.1 Постановка задачи экспериментального исследования.....	145
4.2 Модель операции записи с репликацией данных	147
4.3 Методика экспериментального исследования	157
4.4 Результаты экспериментальной оценки	159
4.4.1 Оценка допустимой интенсивности входного потока	172
4.5 Анализ результатов и область применимости алгоритма.....	175
4.6 Выводы	179
ЗАКЛЮЧЕНИЕ	182
СПИСОК ЛИТЕРАТУРЫ.....	187
ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	206
ПРИЛОЖЕНИЕ (обязательное) Акты о внедрении результатов	212

ВВЕДЕНИЕ

Актуальность темы исследования. Развитие современных телекоммуникационных систем связано с ростом числа сервисов, формирующих и обрабатывающих значительные объемы данных на периферии сети: сервисов Интернета вещей, промышленной автоматизации, интеллектуального видеонаблюдения, транспортных систем, распределенных приложений реального времени и сервисов на основе искусственного интеллекта. По оценкам IDC, объем создаваемых, обрабатываемых и реплицируемых данных продолжает быстро увеличиваться: если в 2023 г. он оценивался на уровне около 129 зеттабайт, то к 2027 г. должен более чем удвоиться, а более поздние прогнозы указывают на рост до порядка 393,9 зеттабайт к 2028 г. В этих условиях централизованная модель обработки и хранения данных исключительно в облачных центрах обработки данных не всегда обеспечивает требуемые показатели задержки, доступности и эффективного использования транспортной сети. Международные документы в области граничных и туманных вычислений рассматривают вычислительные, сетевые и запоминающие ресурсы как взаимосвязанные компоненты инфраструктуры, размещаемые ближе к пользователю и источникам данных. В частности, ITU-T Y.3540 определяет инфраструктуру граничных вычислений через ресурсы обработки, сети и хранения, а IEEE 1934-2018 закрепляет представление о туманных вычислениях как о распределенной архитектуре системного уровня, обеспечивающей размещение вычислительных, сетевых и запоминающих ресурсов на промежуточных уровнях между облачными центрами обработки данных и конечными устройствами.

Перенос операций доступа к данным на уровень туманных вычислений позволяет сократить зависимость времени отклика от состояния магистральных и транспортных сегментов сети, однако приводит к усложнению подсистемы хранения. Распределенное хранение с репликацией требует межузлового обмена, синхронизации реплик, подтверждения операций записи и восстановления избыточности при отказах. Данные процессы используют коммуникационные и вычислительные ресурсы той же инфраструктуры, что и пользовательские операции, вследствие чего возникает необходимость согласованного управления информационными потоками, представленными пользовательским и служебным трафиком. В настоящей работе под информационными потоками понимаются пользовательский и служебный трафик, а термин «нагрузка» используется для характеристики их воздействия на узлы хранения, межузловые соединения и обслуживающие ресурсы системы.

Особую значимость приобретает операция записи, поскольку при репликации включает передачу данных на несколько узлов хранения, фиксацию данных на репликах, получение подтверждений и выполнение служебных процедур согласования. Увеличение уровня избыточности повышает готовность и устойчивость хранения, но одновременно увеличивает объем межузлового обмена и может повышать время завершения операции. Поэтому актуальность исследования определяется необходимостью разработки модели, метода и алгоритма управления информационными потоками распределенной подсистемы хранения данных телекоммуникационной системы в инфраструктуре туманных вычислений при соблюдении требований к готовности выполнения операций записи и ограничений на использование коммуникационных и вычислительных ресурсов.

Степень разработанности темы. Исследованию вопросов построения и функционирования распределенных телекоммуникационных систем, анализа информационных потоков, телетрафика, производительности, качества обслуживания, отказоустойчивости и репликации данных посвящен значительный объем научных работ. Наряду с этим активно развиваются исследования в области

перспективных сетей и систем, облачных, туманных и граничных вычислений, распределения ресурсов и организации такой инфраструктуры.

Существенный вклад в формирование и развитие указанных научных направлений внесли В.М. Вишневский, А.Н. Волков, Ю.В. Гайдамака, А.Б. Гольдштейн, Б.С. Гольдштейн, В.Г. Карташевский, А.Е. Кучерявый, Е.А. Кучерявый, Р.В. Киричек, М.А. Маколкина, А.И. Парамонов, К.Е. Самуйлов, С.Н. Степанов, В.О. Тихвинский, А.А.А. Ateya, F. Bonomi, E. Brewer, R. Buyya, M. Chiang, D. Gifford, J. Gray, L. Kleinrock, L. Lamport, M. Satyanarayanan и другие ученые. Вместе с тем вопросы управления пользовательским и служебным трафиком распределенной подсистемы хранения, размещаемой на уровне туманных вычислений, при одновременном учете репликации, сетевых задержек и ограничений по готовности требуют дальнейшего исследования.

Научная задача. В диссертации решается научная задача разработки модели, метода и алгоритма управления информационными потоками распределенной подсистемы хранения данных телекоммуникационной системы в инфраструктуре туманных вычислений, направленных на повышение производительности операций записи с репликацией как механизмом обеспечения отказоустойчивости и качества обслуживания при соблюдении ограничений по готовности, размещению реплик и служебному межузловому обмену.

Предмет и объект исследования. *Объектом исследования* является распределенная подсистема хранения данных телекоммуникационной системы, функционирующей в инфраструктуре туманных вычислений. *Предметом исследования* являются модели, методы и алгоритмы управления информационными потоками при выполнении операций записи с репликацией данных, обеспечивающие повышение производительности подсистемы хранения при ограничениях по готовности и использованию коммуникационных ресурсов.

Цель работы и задачи исследования. *Целью работы* является разработка модели, метода и алгоритма управления информационными потоками распределенной подсистемы хранения данных телекоммуникационной системы, размещаемой на уровне туманных вычислений, обеспечивающих повышение

допустимой интенсивности входного потока операций записи с репликацией данных, измеряемой в заявках в секунду, при соблюдении заданных ограничений на время завершения операции записи, 95-й и 99-й процентиля задержки, готовность выполнения операции записи и объем служебного межузлового обмена.

Задачи работы:

1) выполнить анализ архитектурных особенностей распределенной подсистемы хранения данных, размещаемой на уровне туманных вычислений, и определить факторы, влияющие на выполнение операций записи с репликацией данных,

2) обосновать выбор показателей производительности распределенной подсистемы хранения данных и определить основной показатель оценки, – допустимую интенсивность входного потока операций записи, с учетом ограничений по времени завершения операции, верхним процентилем задержки, готовности и служебному межузловому обмену,

3) разработать модель управления информационными потоками распределенной подсистемы хранения данных, учитывающую пользовательские операции записи, служебные операции репликации и восстановления избыточности, очереди обслуживания на узлах, ограничения обслуживающих ресурсов и пропускной способности межузловых соединений,

4) разработать метод оценки готовности выполнения операции записи с репликацией данных, учитывающий фактор репликации, порог подтверждения записи и доступность элементов цепочки обслуживания операции,

5) разработать алгоритм выбора узлов хранения при выполнении операции записи с репликацией данных, учитывающий пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности выполнения операции записи,

6) провести экспериментальную оценку предложенного алгоритма в сравнении с базовыми стратегиями распределения нагрузки и определить его влияние на допустимую интенсивность входного потока операций записи, время завершения операции и верхние процентиля задержки.

Теоретическая значимость исследования заключается в развитии модельно-методического аппарата информационными потоками распределенной подсистемы хранения данных, размещаемой на уровне туманных вычислений. В работе предложено представление операции записи с репликацией данных как процесса обслуживания, в котором репликация рассматривается одновременно как механизм обеспечения отказоустойчивости и как источник служебной межузловой нагрузки. Разработанная модель позволяет формализовать взаимосвязь между пользовательскими операциями записи, служебными операциями поддержания репликации, очередями обслуживания, сетевыми задержками, ограничениями межузлового обмена и готовностью выполнения операции записи. Полученные результаты дополняют существующие подходы к анализу распределенных подсистем хранения за счет совместного учета пользовательской и служебной нагрузки, сетевых задержек и готовности выполнения операции записи.

Практическая значимость исследования заключается в возможности применения полученных результатов при проектировании, настройке и экспериментальной оценке распределенных программно-определяемых подсистем хранения данных, функционирующих в составе телекоммуникационной инфраструктуры сетей 5G/6G, периферийных и туманных вычислений. Результаты работы могут быть использованы операторами связи и провайдерами инфраструктурных сервисов при размещении сервисов хранения и обработки данных на периферийных и туманных узлах сети, операторами центров обработки данных при построении распределенных подсистем хранения, а также организациями, эксплуатирующими корпоративные сети, промышленные системы Интернета вещей, системы интеллектуального видеонаблюдения и транспортные цифровые сервисы. Разработанная модель управления информационными потоками, метод оценки готовности и алгоритм выбора узлов хранения могут применяться для обоснования параметров репликации, режима подтверждения записи и распределения операций записи при ограничениях по задержке, готовности и служебному межузловому обмену.

Методология и методы исследования. В работе использованы методы математического и имитационного моделирования телекоммуникационных систем и сетей, теория массового обслуживания, методы расчета коэффициента готовности, моделирования процессов репликации, управления потоками заявок в распределенных сетях с очередями и сравнительной оценки сценариев распределения нагрузки.

Научная новизна:

1) Разработана модель управления информационными потоками распределенной подсистемы хранения данных телекоммуникационной системы, размещенной в инфраструктуре туманных вычислений, отличающаяся от известных тем, что репликация представлена одновременно как механизм обеспечения отказоустойчивости хранения и как источник служебного межузлового трафика, конкурирующего с пользовательским трафиком за коммуникационные и вычислительные ресурсы. Что в свою очередь позволяет формализовать управление информационными потоками, представленными пользовательским и служебным трафиком и формирующими нагрузку на узлы хранения и межузловые соединения, при ограничениях по очередям обслуживания, пропускной способности соединений и готовности выполнения операции записи.

2) Разработан метод оценки готовности выполнения операции записи с репликацией данных, отличающийся учетом фактора репликации, порога подтверждения записи и доступности элементов цепочки обслуживания операции записи. Метод позволяет обосновывать параметры репликации как механизма отказоустойчивости не изолированно, а с учетом их влияния на качество обслуживания, время завершения операции записи и объем служебного межузлового обмена.

3) Разработан алгоритм интегрального выбора узлов хранения при выполнении операции записи с репликацией данных, отличающийся учетом пользовательской очереди, служебной нагрузки, сетевой задержки и ограничения по готовности при формировании подмножества узлов размещения реплик. Алгоритм обеспечивает выбор узлов с учетом двойственной роли репликации: как

средства отказоустойчивого хранения и как фактора, влияющего на задержки, служебную нагрузку и качество обслуживания операций записи.

Положения, выносимые на защиту:

1) Модель управления информационными потоками распределенной подсистемы хранения данных туманного уровня телекоммуникационной системы, отличающаяся от известных тем, что в ней отдельно учитываются пользовательские операции записи и служебные операции поддержания репликации, что позволяет формализовать их распределение при ограничениях по очередям обслуживания, пропускной способности межузловых соединений и требуемому коэффициенту готовности выполнения операции записи.

2) Метод оценки готовности выполнения операции записи с репликацией данных в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений, позволяет обосновать выбор режима подтверждения записи с учетом числа реплик, порога подтверждения и общей части цепочки обслуживания. Для исследуемой конфигурации при $R = 3$, $MTTR = 30$ мин и $A_{\text{общ}} = 0,99999$ переход от строгого режима подтверждения пороговому снижает расчетную ожидаемую недоступность функции записи с 22,334 до 5,260 мин/год.

3) Алгоритм выбора узлов хранения при выполнении операции записи с репликацией данных, отличающийся от известных тем, что выбор подмножества узлов выполняется на основе интегральной оценки, учитывающей пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности выполнения операции записи, что позволяет увеличить допустимую интенсивность входного потока операций записи на 48,3 % по сравнению с равномерным циклическим распределением при соблюдении заданного ограничения на время завершения операции. При этом дополнительное преимущество относительно стратегии, учитывающей пользовательскую и служебную очереди, составляет 2,4 %.

Степень достоверности и апробация результатов. Достоверность основных результатов работы подтверждается корректным применением

математического аппарата, обсуждением результатов работы на конференциях и семинарах, в том числе международных, публикацией основных результатов в рецензируемых журналах (БАК, Scopus, Web of Science).

Основные результаты работы докладывались и обсуждались на следующих международных и российских конференциях и семинарах: III Международной молодежной конференции по информационной безопасности (2025), международной конференции International Scientific and Technical Conference Modern Computer Network Technologies (MoNeTeC 2024), международной конференции Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO 2021, 2022), международной конференции International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT 2020, 2021), международная научно-техническая и научно-методическая конференция «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (АПИНО 2019, 2020, 2021, 2025, 2026).

Савельева А.А. также является исполнителем государственного контракта № ПЗ3-1-26/5 от «26» февраля 2021 г. на выполнение НИР по теме: «Разработка методов, моделей и средств повышения производительности и отказоустойчивости программно-определяемых СХД с использованием средств автоматизированного контроля и анализа результатов на базе нейронной сети» (шифр «Мульти-ЦОД», 2021-2023, Минцифры России).

Результаты исследований, связанных с тематикой диссертационной работы, получили внешнюю конкурсную апробацию. Соискатель являлась победителем конкурса грантов Комитета по науке и высшей школе Санкт-Петербурга для аспирантов в 2019 и 2021 гг. по проектам, связанным с автоматизированным тестированием производительности и отказоустойчивости программно-определяемых распределенных систем хранения данных, а также с исследованием новых путей оценки качества предоставления цифровых услуг. Авторский коллектив с участием соискателя стал победителем конкурса лучших инновационных проектов в сфере науки и высшего образования Санкт-Петербурга в 2021 г. за проект, направленный на разработку методов, моделей и средств

повышения производительности и отказоустойчивости программно-определяемых систем хранения данных. В 2025 г. соискателю присуждена премия Правительства Санкт-Петербурга «Молодежная премия Санкт-Петербурга» в номинации «В области информационных технологий» за результаты деятельности, связанной с тематикой исследования.

Публикации по теме диссертации. Материалы работы изложены в полной мере в 27 публикациях, из них: 4 статьи в изданиях, включенных в перечень рецензируемых научных изданиях (перечень ВАК при Минобрнауки России), 6 статей в изданиях, включенных в международные базы цитирования; 3 результата интеллектуальной деятельности; 8 отчетов о НИР; 6 статей в других изданиях и материалах конференций.

Личный вклад автора. Результаты диссертационной работы получены автором самостоятельно. В работах, выполненных в соавторстве, автор принимала непосредственное участие в проведении исследований, анализе и обобщении полученных результатов.

Соответствие специальности. Работа соответствует пп. 1, 3, 4, 5, 12 паспорта специальности 2.2.15, поскольку направлена на разработку моделей и методов исследования и проектирования элементов телекоммуникационных систем и сетей (п. 1), исследование процессов хранения информации (п. 3) и архитектуры распределенной подсистемы хранения данных как элемента телекоммуникационной системы (п. 4), а также на управление информационными потоками (п. 5) и распределение коммуникационных ресурсов в инфраструктуре туманных вычислений (п. 12).

1 Архитектурные особенности и требования к подсистеме распределенного хранения данных в телекоммуникационных системах граничных и туманных вычислений

В современных телекоммуникационных системах, ориентированных на обслуживание потоковых и интерактивных сервисов, а также массовых сервисов Интернета вещей, ключевыми показателями качества обслуживания являются задержка, пропускная способность и непрерывность предоставления услуг. Рост объемов данных на периферии сети и повышение требований к оперативности обработки приводят к тому, что централизованная обработка и хранение данных исключительно в облачных центрах обработки данных во многих сценариях увеличивает задержку и повышает нагрузку на транспортные сегменты сети. Развитие граничных и туманных вычислений, а также стандартизация размещения приложений на границе сети связи как элемента телекоммуникационной инфраструктуры рассматриваются в отраслевых и международных рекомендациях как один из путей снижения задержек, повышения качества обслуживания и обеспечения требований к обработке и защите данных за счет приближения вычислительных ресурсов к источникам и потребителям данных [1–3].

Перенос функций обработки на граничный уровень и уровень туманных вычислений сопровождается перераспределением функций хранения данных между облачными, периферийными и промежуточными ресурсами инфраструктуры. На периферийных узлах требуется поддержка операций кэширования, буферизации, журналирования, хранения промежуточных

результатов и локальных состояний приложений. В результате подсистема хранения на периферийном уровне становится одним из критических элементов, влияющих на интегральные характеристики телекоммуникационного сервиса: задержки операций чтения/записи и эффективная пропускная способность подсистемы хранения начинают определять качество обслуживания наряду с характеристиками каналов связи и протокольных процедур. При этом операции хранения в распределенном исполнении порождают межузловой обмен служебными операциями, формируют дополнительный трафик и требуют согласованного распределения коммуникационных ресурсов в граничной и туманной инфраструктуре [2,4,5].

Одновременно с требованиями к производительности предъявляются требования к непрерывности обслуживания и доступности данных. Для их выполнения в распределенных подсистемах хранения применяются механизмы избыточности, а также процедуры согласования и синхронизации, обеспечивающие сохранность и доступность данных при отказах элементов инфраструктуры. Указанные механизмы повышают устойчивость хранения, однако требуют дополнительных ресурсов на межузловой обмен и обработку служебных запросов, что может приводить к увеличению задержек и снижению доступной пропускной способности. Так, возникает компромисс между доступностью и производительностью, требующий формализованного анализа и обоснованного выбора архитектурных и режимных параметров распределенного хранения с учетом ограниченности коммуникационных ресурсов периферийной инфраструктуры и требований телекоммуникационных сервисов [2,4,6].

Понятие телекоммуникационной системы целесообразно рассматривать в двух взаимосвязанных плоскостях: нормативно-правовой и инженерно-архитектурной. В международной стандартизации телекоммуникационная система трактуется как совокупность средств, обеспечивающих предоставление телекоммуникационных услуг между несколькими точками, в которых оборудование предоставляет доступ к таким услугам. Данное понимание важно тем, что оно фиксирует системный характер объекта: речь идет не об отдельном

устройстве или линии связи, а о комплексе взаимодействующих средств, обеспечивающих передачу и обслуживание информации в распределенной среде.

В российской нормативной базе близкие по содержанию определения закреплены через категории «сеть связи» и «информационно-телекоммуникационная сеть». Федеральный закон № 126-ФЗ «О связи» [7] определяет сеть связи как технологическую систему, включающую средства и линии связи и предназначенную для электросвязи либо почтовой связи. В свою очередь, Федеральный закон № 149-ФЗ «Об информации, информационных технологиях и о защите информации» [8] рассматривает информационно-телекоммуникационную сеть как технологическую систему, предназначенную для передачи по линиям связи информации, доступ к которой осуществляется с использованием средств вычислительной техники. Тем самым уже на уровне законодательства подчеркивается, что современная среда передачи данных не сводится исключительно к каналам связи, а предполагает использование вычислительных средств как неотъемлемой части функционирования такой системы.

Вместе с тем для целей инженерного анализа и проектирования одной лишь правовой дефиниции недостаточно. Современные телекоммуникационные системы, особенно в контексте сетей нового поколения, граничных и туманных вычислений, строятся как интеграция сетевых, вычислительных и запоминающих ресурсов. Указанное прямо отражено в международных стандартах. Так, в архитектуре Multi-access Edge Computing (*MEC*) узел *MEC* включает платформу и виртуализированную инфраструктуру, предоставляющую вычислительные, сетевые и запоминающие ресурсы для приложений [1]. Аналогично Рекомендация ITU-T Y.3540 определяет инфраструктуру граничных вычислений как ресурсы, обеспечивающие обработку, сетевую составляющую и хранение (*от англ. «processing, network and storage» из документа*) вблизи пользователя [2]. Следовательно, в современной стандартизированной архитектуре хранение данных рассматривается не как внешняя или факультативная функция, а как один из базовых ресурсных компонентов телекоммуникационной инфраструктуры.

Дополнительное подтверждение данному подходу содержится в документах, посвященных развитию сетей будущего. В Рекомендации ITU-R M.2160 [9], определяющей рамочные представления об ИМТ-2030, указано, что вычислительные и сервисы данных должны стать интегральной частью системы ИМТ-2030, включая обработку данных на краю сети вблизи источника данных. Из этого следует, что функциональность по размещению, сохранению, обработке и передаче данных становится характеристикой телекоммуникационной системы нового поколения. Иными словами, передача данных больше не образует единственную сущность телекоммуникационной инфраструктуры, равную значимость приобретают механизмы хранения и локальной обработки, непосредственно влияющие на задержку, масштабируемость, устойчивость системы обслуживания и доступность сервиса.

С учетом изложенного утверждение о том, что система хранения данных является элементом телекоммуникационной системы, представляется методологически обоснованным, если речь идет не об узком юридическом толковании сети связи, а о функционально-архитектурном понимании телекоммуникационной системы как совокупности средств, обеспечивающих предоставление услуги. В таком контексте распределенная подсистема хранения данных может рассматриваться как функциональный элемент телекоммуникационной системы или телекоммуникационного узла, поскольку она обеспечивает сохранение состояния данных, поддержку операций записи и чтения, устойчивость к отказам и тем самым участвует в достижении целевых показателей качества обслуживания. Особенно это справедливо для туманных и граничных архитектур, где приближение хранения к источнику и потребителю данных является частью самой сервисной модели.

В связи с указанным выше в настоящей работе используется следующая интерпретация: телекоммуникационная система представляет собой совокупность взаимосвязанных средств связи, вычислительных и программно-технических компонентов, обеспечивающих передачу, обработку, хранение и предоставление данных и сервисов пользователю. При таком подходе система хранения данных

выступает не внешним по отношению к телекоммуникационной системе элементом, а одной из ее функциональных подсистем. Такая трактовка согласуется с международной стандартизацией телекоммуникационных систем, а также отвечает современному направлению эволюции сетей в сторону интеграции коммуникаций, вычислений и данных в единой инфраструктурной среде.

Таким образом, рассматриваемая проблематика относится к области телекоммуникаций, поскольку затрагивает методы исследования, моделирования и проектирования сетей и систем связи, процессы передачи и хранения данных как составные части функционирования телекоммуникационной системы, архитектурные решения и структурные параметры распределенной инфраструктуры, а также методы распределения коммуникационных ресурсов в граничных и туманных вычислениях. При таком рассмотрении распределенная подсистема хранения данных выступает как элемент телекоммуникационной инфраструктуры, влияющий на показатели качества обслуживания через межузловые взаимодействия, протоколы согласования и режимы синхронизации [1,2,4].

В рамках настоящей главы рассматриваются предпосылки применения граничных и туманных вычислений в телекоммуникационных системах, анализируются подходы к организации распределенного хранения на периферийном уровне, а также описываются архитектурные особенности и базовые процессы доступа к данным, формирующие основу для последующего построения моделей и разработки методов повышения производительности подсистем распределенного хранения данных [10].

1.1 Граничные и туманные вычисления в телекоммуникационных системах

Граничные и туманные вычисления рассматриваются как взаимосвязанные компоненты телекоммуникационной инфраструктуры, однако дальнейшее исследования, моделирование и разработки в рамках настоящей работы выполняются для туманного уровня, где сосредоточены механизмы

распределенного хранения и возникающие ограничения по готовности и производительности [11]. При этом граничный уровень в рамках работы используется как контекст источника запросов и точка формирования пользовательской нагрузки, тогда как постановка последующих задач относится к туманному уровню, на котором реализована распределенная подсистема хранения и принимаются управленческие решения.

Туманные вычисления представляют собой горизонтальную архитектуру системного уровня, в которой обслуживающие ресурсы узла и функции (вычисления, хранение данных и сетевое взаимодействие) распределяются между облачной средой и периферийными узлами. В референсной архитектуре OpenFog, принятой в качестве стандарта IEEE 1934-2018 [4], туманные вычисления представлены как распределенная модель организации вычислительных, сетевых и запоминающих ресурсов. В такой модели межузловые взаимодействия и процедуры согласования являются штатными процессами. При переносе функций на периферийный уровень изменяется распределение вычислительных и коммуникационных ресурсов, а также структура служебного и пользовательского трафика между уровнями системы [1,5].

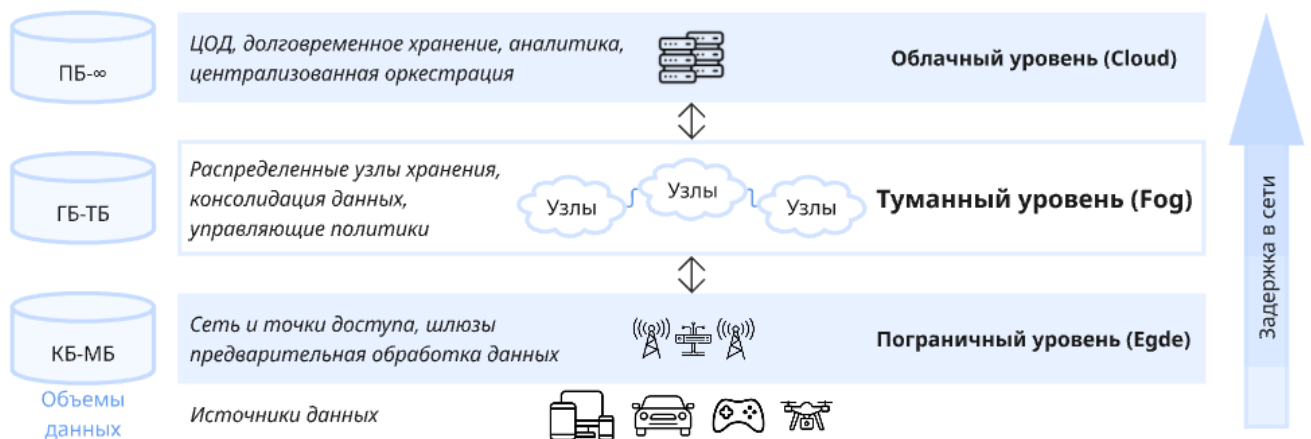


Рисунок 1.1 – Расположение облачного, туманного и граничного уровней в рассматриваемой архитектуре

1.1.1 Предпосылки переноса операций доступа к данным к периферии инфраструктуры

В научной литературе и обзорных работах по управлению данными в подобных средах обычно выделяют два базовых класса подходов к доступу к данным [12,13]. Первый класс основан на централизованном хранении и обработке в облаке, когда операции доступа к данным, прежде всего запись, обслуживаются на облачном уровне, а периферийный уровень используется преимущественно как источник данных и точка выполнения ограниченного набора функций предварительной обработки [13,14]. Второй класс опирается на приближение данных к потребителю за счет кэширования и копирования (репликации) на периферии: наиболее распространенный вариант – ускорение операций чтения (в том числе за счет копий контента на периферии) при сохранении централизованного управления исходным состоянием данных и операциями записи [15–17].

Сети доставки контента *CDN* (от англ. Content Delivery/Distribution Network) являются характерным примером второго класса, поскольку повышение качества обслуживания достигается за счет размещения копий объектов контента на распределенных периферийных узлах и обслуживания запросов чтения из ближайшей точки [16,18]. При этом операции записи и актуализации исходного состояния традиционно выполняются централизованно на исходных серверах, после чего изменения распространяются по сети в рамках механизмов обновления и кэширования [16]. Такая модель эффективна для сценариев с доминирующим чтением и относительно редкими обновлениями, однако не полностью покрывает телекоммуникационные сценарии граничных и туманных вычислений, где существенную долю могут составлять операции записи на периферии: буферизация, журналирование, накопление телеметрии, хранение промежуточных результатов и локальных состояний приложений [10,13].

1.1.2 Анализ отечественных и зарубежных источников по теме

Отечественные исследования туманной инфраструктуры отмечают, что размещение сервисов на периферии обусловлено требованиями к задержкам и устойчивости обслуживания при ограничениях транспортной сети, при этом ключевыми остаются вопросы маршрутизации, распределения нагрузки между узлами туманного уровня и управления потоками обслуживания в условиях переменной сетевой обстановки [19,20]. В работах, посвященных общим принципам туманных вычислений, показано, что их применение связано с требованиями сетей электросвязи, поддержкой сервисов реального времени и распределенным размещением вычислительных, сетевых и запоминающих ресурсов [21]. В исследованиях сервисов, критичных к задержкам, дополнительно подчеркивается необходимость выносить функции обработки ближе к пользователю и согласованно учитывать сетевые ограничения при проектировании и эксплуатации сервисов [22,23].

В работах по системам хранения данных показано, что распределенные архитектуры хранения и сети хранения данных реализуют доступ к данным как сетевой сервис. Классификация распределенных систем хранения данных, их архитектурных вариантов и критериев выбора представлена в работе [24], что позволяет обосновать рассмотрение распределенной СХД как отдельного класса инфраструктурных решений. Поэтому эксплуатационные характеристики операций ввода-вывода зависят от задержек и пропускной способности сети, а также от служебных межузловых процессов, связанных с репликацией, согласованием и восстановлением данных. Для распределенной подсистемы хранения, размещаемой на уровне туманных вычислений, это означает, что сеть входит в цепочку обслуживания операций записи и должна учитываться при моделировании их производительности и доступности.

Зарубежные исследования закрепляют туман как распределенную парадигму, в которой вычисления и хранение являются инфраструктурными функциями сетевого уровня и должны совместно проектироваться с учетом сетевой

связности и распределенного размещения узлов. Одним из ранних концептуальных обоснований туманных вычислений является работа Flavio Bonomi и соавторов [25], в которой туман рассматривается как расширение облачной парадигмы к периферии сети, характеризуемое малой задержкой, географической распределенностью, большим числом узлов и неоднородностью инфраструктуры. В работе Mung Chiang и Tao Zhang [26] туманные вычисления определяются как архитектура, в которой функции вычислений, хранения, управления и сетевого взаимодействия размещаются ближе к конечным пользователям и устройствам, не сосредотачиваются исключительно на облачном уровне. В прикладных работах по организации сервисов и распределенному размещению данных в туманной инфраструктуре, включая механизмы репликации, акцентируется влияние размещения ресурсов, реплик и режимов согласованности на задержки и доступность данных, однако нагрузочные эффекты от потока служебных операций поддержания избыточности и их конкуренция с пользовательскими операциями нередко описываются упрощенно либо рассматриваются отдельно от задач управления потоками обслуживания [27,28]. Систематизация Redowan Mahmud, Ramamohanarao Kotagiri и Rajkumar Buyya [29] также показывает, что для туманных вычислений характерны нерешенные задачи размещения ресурсов, управления задержкой и учета распределенного характера инфраструктуры. Обобщенное представление о граничных и туманных вычислениях как о совокупности принципов, архитектур и технологий распределенной обработки данных представлено в монографии под редакцией R. Buyya [30].

Таким образом, при наличии значительного массива исследований по туманной инфраструктуре, распределенному хранению, репликации и балансировке нагрузки сохраняется актуальная задача формализованного управления совокупной нагрузкой распределенной подсистемы хранения на туманном уровне, включающей пользовательские и служебные операции. Постановка такой задачи требуется для количественной оценки влияния параметров избыточности и режимов подтверждения записи на показатели

качества обслуживания в условиях ограниченных коммуникационных ресурсов и сценариев отказа [31].

При выполнении операций чтения и записи с обращением к облачному уровню формируется межуровневый трафик, а транспортная сеть становится частью цепочки обслуживания запросов. В результате возрастают задержки и повышается чувствительность качества обслуживания к перегрузкам и ограничениям коммуникационных ресурсов. Кроме того, обзорные и прикладные исследования по периферийному хранению показывают, что именно операции записи (особенно при высокой конкурентности и малых блоках данных) становятся фактором снижения производительности и требуют специализированных механизмов организации доступа и консолидации [12,32].

В настоящей работе рассматривается архитектурная организация потоков чтения и записи, предусматривающая размещение операций доступа к данным на уровне туманных вычислений. В этой схеме чтение и запись выполняются распределенными узлами, расположенными ближе к источникам и потребителям данных, а облачный уровень используется преимущественно для консолидации результатов записи и долговременного хранения [2,4,12]. Такое перераспределение потоков доступа к данным сокращает число межуровневых обращений при обслуживании запросов, снижает зависимость времени отклика от состояния транспортной сети и уменьшает нагрузку на магистральные сегменты сети за счет переноса части операций чтения и записи ближе к источникам и потребителям данных.

Отличительная черта рассматриваемого подхода состоит в том, что уровень туманных вычислений рассматривается не как вспомогательный механизм кэширования, ориентированный преимущественно на операции чтения, а как среда выполнения операций чтения и записи в распределенной подсистеме хранения, функционирующей в составе телекоммуникационной инфраструктуры. В дальнейшем анализе основное внимание уделяется параметрам избыточности, процедурам согласования и синхронизации, а также управлению нагрузкой, поскольку эти факторы определяют доступность хранения и дополнительную

служебную нагрузку, влияющую на задержки и доступную пропускную способность. В дальнейшем в настоящей работе основное внимание уделяется цепочке обслуживания операции записи, поскольку передача данных к репликам и получение подтверждений формируют основную долю дополнительных ресурсных затрат, связанных с репликацией.

Предлагаемая организация размещения и обработки данных предусматривает перенос операций записи и чтения на уровень туманных вычислений при сохранении за облачным уровнем функций долговременного хранения, централизованной аналитики и оркестрации. Схема взаимодействия источников данных, пограничного уровня, распределенной подсистемы хранения на уровне туманных вычислений и облачного уровня приведена на рисунке 1.2.

В рассматриваемой архитектуре туманного уровня нагрузка подсистемы распределенного хранения определяется не только интенсивностью пользовательских операций чтения/записи, но и объемом служебного межузлового обмена, порождаемого репликацией, синхронизацией и процедурами согласования. Следовательно, повышение производительности телекоммуникационного сервиса требует управления совокупной нагрузкой (пользовательской и служебной) за счет выбора параметров и режимов функционирования подсистемы хранения, а также правил распределения запросов и коммуникационных ресурсов в распределенной инфокоммуникационной (телекоммуникационной) сети туманного уровня, обслуживающей операции хранения.

При этом корректная оценка производительности распределенной подсистемы хранения требует явного учета профиля нагрузки и параметров тестирования, поскольку игнорирование особенностей распределенной архитектуры и инфраструктуры межузлового взаимодействия приводит к искажению результатов и неверным выводам о производительности и устойчивости [33,34].

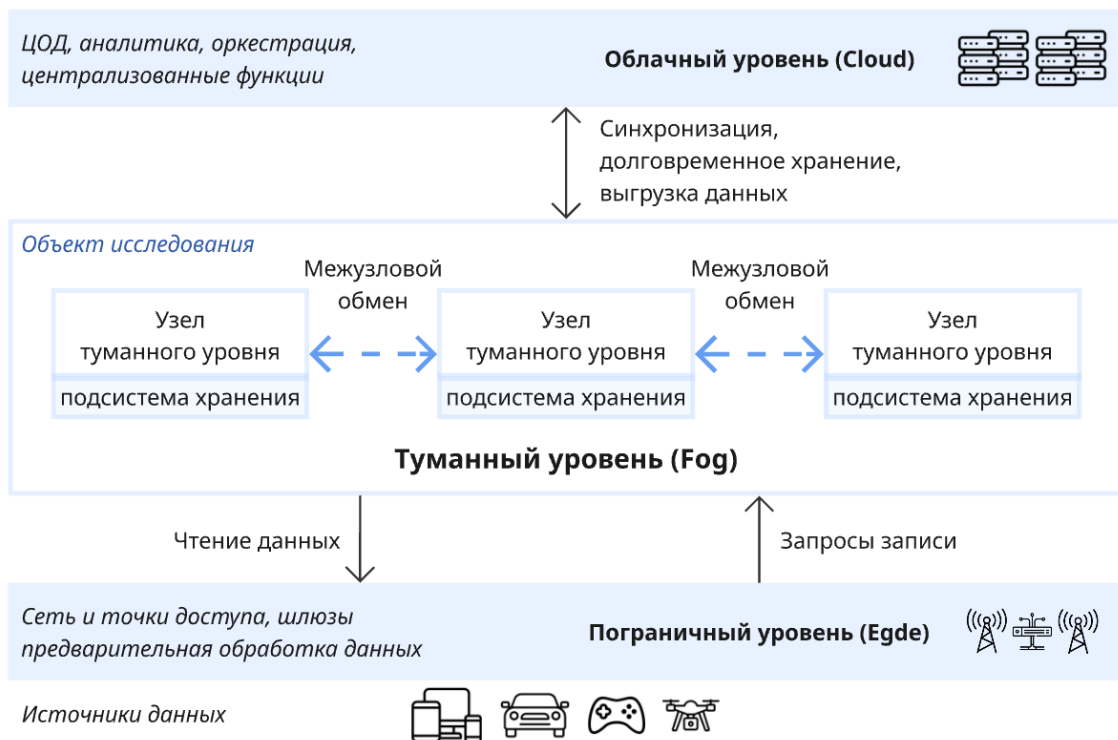


Рисунок 1.2 – Обобщенная схема взаимодействия уровней распределенной инфраструктуры при выполнении операций чтения и записи данных

1.2 Обоснование выбора архитектуры подсистемы хранения данных

Для обсуждения архитектурных особенностей и эксплуатационных факторов используется более общее понятие «система хранения данных» (СХД), тогда как рассматриваемая подсистема хранения трактуется как функциональная часть СХД, обеспечивающая предоставление услуг хранения и обслуживания операций ввода-вывода. Используемая терминология согласуется с нормативными документами, устанавливающими термины и определения в области систем хранения данных и подсистем хранения [35–37].

С учетом изложенных принципов построения инфраструктуры туманных вычислений и особенностей ее функционирования в составе телекоммуникационной инфраструктуры при выборе архитектуры СХД в настоящей работе учитываются следующие факторы:

- масштабируемость по числу узлов и контроллеров,

- возможности управления производительностью в условиях многоклиентского доступа,
- поддержка построения отказоустойчивых и катастрофоустойчивых конфигураций,
- эксплуатационная и организационная сложность решения.

Указанные факторы определяют не только характеристики хранения как такового, но и объем межузлового обмена и нагрузку на коммуникационные ресурсы при репликации и синхронизации данных, что критично для туманных вычислений. Вопросы влияния характеристик инфраструктуры хранения данных на организацию и производительность прикладных сервисов также рассматривались в ранее выполненных исследованиях [38,39]. Теоретические, модельные и системотехнические аспекты повышения эффективности программно-определяемых СХД также исследовались в рамках выполненной НИР [40–42]

Для обоснования выбора решения для дальнейшей реализации выполнен анализ существующих архитектур СХД, результаты сравнения по качественным характеристикам представлены в таблице 1.1. Набор качественных характеристик согласуется с общепринятыми моделями качества программных и информационных систем (ISO/IEC 25010), что обеспечивает сопоставимость терминов и корректность интерпретации сравнительных оценок, при этом сопоставление выполнено в логике многокритериальной оценки, допускающей использование как количественных, так и качественных критериев при фиксированных правилах присвоения оценок [43,44].

Таблица 1.1 – Сравнение типов СХД (качественные характеристики)

Критерий	Классическая СХД	Распределенная программно-определяемая СХД
Относительная стоимость владения	высокая	средняя
Относительные возможности управления производительностью	высокие	средние
Относительные возможности масштабирования	низкие	высокие
Относительная сложность обслуживания	низкая	высокая
Возможность построения катастрофоустойчивых кластеров	есть	есть
Относительная наработка на отказ	высокая	высокая
Относительная сложность организации	низкая	высокая
Максимальная производительность [45]	21002561 iops	5120099 iops
Количество контроллеров	1-2	225 [46]

Из данных таблицы 1.1 следует, что распределенные программно-определяемые СХД являются наиболее сложными в организации и сопровождении, однако при корректном проектировании обеспечивают наибольшую масштабируемость и более высокую потенциальную отказоустойчивость за счет распределения функций хранения и управления по множеству узлов. При этом оценка предельной производительности для распределенных архитектур требует осторожности: результаты ряда испытаний приводятся для ограниченной конфигурации и/или сценариев с одним клиентом, тогда как в многоклиентском режиме и при масштабировании по числу узлов совокупная производительность системы хранения может возрастать за счет параллельного обслуживания запросов и распределения нагрузки по группам дисков и контроллерам.

С учетом результатов анализа (таблица 1.1) для целей настоящей работы принято рассматривать распределенные программно-определяемые СХД [47]. Данный выбор обусловлен, в частности, необходимостью поддержки значительного числа контроллеров и распределенных узлов управления, что требуется для функционирования инфраструктуры туманных вычислений и реализации режима, при котором операции чтения и записи обслуживаются на туманном уровне с последующей консолидацией результатов записи на верхнем уровне хранения.

В ходе анализа работы СХД в различных условиях применения определены следующие принципы построения распределенной СХД:

- высокая надежность хранения данных,
- аппаратная составляющая архитектуры должна базироваться на общедоступных вычислительных компонентах,
- конфигурация всех узлов и компонентов должна быть типизирована и определена явным образом на всех узлах,
- конфигурация должна быть изначально подобрана таким образом, чтобы обеспечивать производительность для всех типов задач, но при этом допускать приоритезацию одних задач над другими в целях соответствия задачам применения туманных вычислений.

Следование данным принципам должно обеспечить построение единого решения СХД, на базе которого возможны дальнейшие разработки по обеспечению избыточности при построении распределенных систем для инфраструктуры туманных вычислений. При этом одним из ключевых требований к подсистеме хранения туманного уровня является обеспечение доступности данных и непрерывности обслуживания при отказах отдельных узлов, каналов связи и компонентов хранения, что в распределенных архитектурах достигается за счет механизмов избыточности. Избыточность необходима для полноценного функционирования системы в случае выхода из строя каких-либо элементов СХД.

Под избыточностью в настоящей работе понимается преднамеренное введение в распределенную систему хранения дополнительных обслуживающих ресурсов узла, компонентов или информационных фрагментов сверх минимально необходимого объема, обеспечивающее сохранение работоспособности системы и доступности данных при отказе отдельных элементов [48]. То есть избыточность в инженерном контексте используется прежде всего для повышения надежности и доступности, а в телекоммуникационной системе может реализовываться, в частности, за счет дублирования элементов и путей либо за счет хранения дополнительных кодовых фрагментов.

В классических СХД возможность объединения всех дисков в единый избыточный массив называется *RAID* (от англ. Redundant Array of Independent Disks), который в зависимости от конфигурации может продолжать обеспечивать доступ к данным при выходе из строя от одного до половины всех используемых дисков (данный параметр определяется уровнем *RAID*). В распределенных программно-определяемых СХД возможность объединения всех узлов в единый избыточный массив называется *RAIN* (от англ. Reliable Array of Independent Nodes), который в зависимости от архитектуры распределенной СХД может обеспечивать работоспособность при выходе из строя любого определенного количества узлов (зависит от способа реализации избыточности). В рамках настоящей работы термины *RAID* и *RAIN* используются как обозначения классов архитектур избыточного размещения данных, различающихся уровнем домена отказа и способом распределения избыточности. Под *RAID* (массив независимых дисков с избыточностью) понимается организация хранения, при которой избыточность создается и используется преимущественно на уровне дисковых устройств внутри одного вычислительного узла (или в пределах одной локальной системы хранения), то есть доменом отказа в первую очередь выступают отдельные диски и локальные контроллеры. Под *RAIN* (массив независимых узлов с избыточностью) понимается организация хранения, при которой избыточность формируется за счет распределения данных и их реплик (либо иных избыточных представлений) между несколькими узлами хранения, соединенными сетью, и в этом случае домен отказа расширяется до уровня узла и сетевого сегмента. Такое различие важно для дальнейшего анализа, поскольку распределение избыточности между узлами определяет объем межузлового обмена при записи и восстановлении, а также требования к размещению реплик по доменам отказа в туманной инфраструктуре.

Для определения уровня избыточности в распределенных программно-определяемых СХД существует два основных механизма: обеспечение избыточности с помощью кодирования, в том числе кодов коррекции ошибок и кодов стирания, и обеспечение избыточности с помощью создания

дополнительных копий, так же называемых репликами, а сам процесс – репликацией [49].

Обеспечение избыточности с помощью кодирования основывается на различных вариантах исправления ошибок, включая применение кодов Хэмминга и Рида-Соломона. Применение кодов избыточности в СХД позволяет снизить объем ресурсов хранения, требуемых для обеспечения отказоустойчивости, по сравнению с полной репликацией данных. Уровень избыточности при этом определяется выбранной схемой кодирования и может настраиваться в зависимости от числа информационных и проверочных фрагментов. При масштабировании подсистемы хранения относительная доля ресурсов, выделяемых под избыточную информацию, возрастает умеренно, что обеспечивает более рациональное использование емкости накопителей. Вместе с тем данный подход имеет ряд ограничений (таблица 1.2).

Таблица 1.2 – Сравнение методов кодирования как механизма обеспечения избыточности в системах хранения

Код/класс	Принцип и параметризация	Избыточность	Масштабирование при увеличении числа дисков	Основные ограничения в СХД
Коды Хэмминга	Блочные коды с исправлением одиночной ошибки и обнаружением двукратной ошибки (для заданного размера блока)	Определяется числом проверочных разрядов. Задается выбором кода и длины блока	При увеличении длины блока относительная доля проверочных разрядов уменьшается, однако уровень исправления ошибок остается ограниченным	Ограниченная исправляющая способность (обычно одиночная битовая ошибка), для отказов дисков/узлов (стираний) в распределенных СХД применимость ограничена
Коды Рида-Соломона (коды стирания)	Параметры (k, m): k фрагментов данных и m фрагментов избыточности. Восстановление возможно при потере до m фрагментов	Избыточность m/k Уровень избыточности задается выбором (k, m)	При росте k при фиксированном m относительная доля избыточности m/k уменьшается, что обеспечивает экономию дискового пространства при большом числе дисков	Высокая ресурсная стоимость восстановления: декодирование и межузловой обмен при реконструкции данных увеличивают задержки и нагрузку на сеть

Продолжение таблицы 1.2

Код/класс	Принцип и параметризация	Избыточность	Масштабирование при увеличении числа дисков	Основные ограничения в СХД
Коды с локальным восстановлением (LRC)	Модификация кодов стирания: дополнительно вводятся локальные проверочные соотношения для сокращения объема данных, считываемых при ремонте	Избыточность, как правило, несколько выше, чем у кода Рида-Соломона; уровень задается параметрами схемы	Сохраняется эффект снижения относительной доли избыточности при увеличении k. Дополнительно уменьшается объем ремонтного трафика в типовых сценариях отказов	Усложнение схемы размещения и управления группами, зависит от распределения отказов и выбранной топологии/доменов отказа

Проектирование высокодоступных СХД усложняется необходимостью отказоустойчивого размещения данных и метаданных избыточности. В частности, если требуется обеспечить продолжение работы распределенной СХД при недоступности сервера с дисками (при наличии достаточного числа узлов) либо при отказе серверной стойки (при наличии достаточного числа стоек), архитектура должна исключать ситуации, при которых в пределах одного домена отказа одновременно размещаются пользовательские данные и соответствующие им контрольные (проверочные) данные, например, контрольные суммы или блоки избыточности. Иными словами, данные и элементы избыточности должны распределяться по различным доменам отказа, что снижает вероятность их совместной утраты при единичном событии отказа.

Применение методов кодирования предполагает вычисление проверочных значений, что увеличивает вычислительную нагрузку на процессор и потребление оперативной памяти, тем самым повышая требования к обслуживающим ресурсам узла. Кроме того, при модификации ранее записанных блоков возникает дополнительная нагрузка на подсистему ввода-вывода: для поддержания согласованности необходимо пересчитывать и обновлять проверочные данные, что приводит к дополнительным операциям чтения и записи.

Для распределенной программно-определяемой СХД указанный подход, как правило, означает, что подтверждение успешной записи формируется после

завершения полного цикла вычисления и фиксации проверочных данных на постоянных носителях. Если же конфигурация системы допускает отправку подтверждения клиенту сразу после попадания данных в оперативную память, но до завершения вычислений и записи всех необходимых блоков на диски, то при отказе СХД в этот промежуток времени возникает риск потери данных, несмотря на ранее выданное подтверждение операции записи.

Дополнительные затраты возникают и в сетевой подсистеме, поскольку поддержание кодированной избыточности сопровождается межузловым обменом данными при обновлении проверочных блоков, деградированном доступе и процедурах восстановления, что увеличивает сетевой трафик и может повышать требования к пропускной способности внутренних соединений СХД [50].

Альтернативным подходом является обеспечение избыточности за счет создания дополнительных копий – репликации. В этом случае отказоустойчивость достигается размещением копий блоков данных на иных объектах хранения – в зависимости от архитектуры такими объектами могут выступать как физически разные диски, так и отдельные логические сущности, относящиеся к различным доменам отказа. Репликация, как правило, обеспечивает более высокую производительность записи за счет отсутствия вычислительных затрат на формирование проверочных данных, а также за счет сокращения числа дополнительных операций ввода-вывода, связанных с перерасчетом контрольной информации. Ключевым недостатком данного подхода является увеличенный расход дискового пространства, пропорциональный коэффициенту репликации.

В рамках настоящего исследования механизмы избыточности на основе кодов стирания не рассматриваются отдельно, поскольку данное направление представляет собой самостоятельную и широко разработанную область исследований распределенных систем хранения. В научной литературе детально изучены как теоретические основы кодированной избыточности [51–54], так и практические компромиссы между отказоустойчивостью, затратами на восстановление, сетевым трафиком и вычислительной сложностью. Поэтому включение данного класса механизмов потребовало бы отдельной постановки

задачи и вывело бы исследование за рамки настоящей работы, сосредоточенной на анализе репликационных механизмов и их влияния на характеристики доступности и обслуживания.

1.3 Архитектура распределенной подсистемы хранения и базовые процессы доступа к данным

В разделах 1.1-1.2 показано, что в телекоммуникационных системах с граничной и туманной инфраструктурой перенос операций доступа к данным на периферийный уровень является одним из ключевых способов снижения задержек и уменьшения межуровневого трафика при сохранении верхнего уровня для консолидации результатов записи и долговременного хранения. Для реализации указанного режима требуется подсистема хранения, способная функционировать в распределенной среде с большим числом узлов и контроллеров, поддерживать согласованные режимы работы и обеспечивать требуемую доступность данных при ограниченности коммуникационных ресурсов периферийной инфраструктуры.

Цель настоящего подраздела заключается в описании архитектурной модели распределенной программно-определяемой подсистемы хранения, размещаемой на уровне туманных вычислений в составе телекоммуникационной инфраструктуры, а также выделение процессов доступа к данным, влияющих на производительность подсистемы хранения и нагрузку на сеть.

1.3.1 Архитектурная модель

В настоящей работе распределенная программно-определяемая подсистема хранения данных рассматривается как элемент телекоммуникационной инфраструктуры, размещаемый на уровне туманных вычислений и влияющий на качество обслуживания прикладных сервисов через сетевые взаимодействия, служебный обмен и процедуры согласования. В рамках исследования такой объект рассматривается прежде всего как компонент инфраструктуры, функционирование

которого определяется характеристиками сети передачи данных и требованиями *QoS* прикладных сервисов. Поэтому далее акцент делается на телекоммуникационных ограничениях и потоках межузлового обмена, а вычислительная интерпретация используется только как удобная форма описания.

Архитектурно подсистема хранения представляется совокупностью территориально распределенных узлов, соединенных телекоммуникационной сетью, и реализует функции хранения, управления размещением данных, согласования доступа и восстановления при отказах. При этом подсистема хранения функционирует в условиях гетерогенности периферийных ресурсов и ограниченной пропускной способности межузловых соединений, что требует учета дополнительной нагрузки, возникающей при сетевом обмене между узлами.



Рисунок 1.3 – Функциональная архитектура распределенной подсистемы хранения туманного уровня

В общем виде в архитектуре выделяются следующие функциональные компоненты (рисунок 1.3):

- клиентский уровень доступа к данным – программные компоненты прикладных сервисов (или сервисный слой), формирующие операции чтения

и записи и взаимодействующие с подсистемой хранения по заданному интерфейсу. На данном уровне задается профиль нагрузки (интенсивность операций, доля чтения/записи, распределение размеров запросов и конкуренция потоков), который далее определяет требования к производительности подсистемы хранения [55],

- уровень управления доступом и координации операций – контроллеры (координаторы), обеспечивающие обработку запросов, маршрутизацию операций к узлам хранения, выбор реплик, а также выполнение процедур согласования и синхронизации. В условиях туманной инфраструктуры требуется поддержка значительного числа контроллеров, поскольку подсистема включает множество узлов, размещенных в разных сегментах сети, а обслуживание запросов должно выполняться близко к источникам и потребителям данных. Распределенное размещение контроллеров позволяет снизить задержки управления и уменьшить концентрацию служебного трафика в отдельных точках сети,

- уровень хранения данных – узлы хранения, предоставляющие обслуживающие ресурсы узла (вычислительные и дисковые ресурсы) для размещения данных и избыточности (реплики или кодовые фрагменты). На узлах хранения выполняются операции чтения/записи фрагментов данных, локальная проверка целостности, а также процедуры ремонта (восстановления) при отказах. В зависимости от режима работы системы часть операций может сопровождаться фоновой передачей данных между узлами (например, для синхронизации реплик и восстановления избыточности),

- метаданные и управление конфигурацией – механизмы, обеспечивающие хранение и актуализацию служебной информации о размещении данных, состоянии узлов, доменах отказа, политике репликации/кодирования и правилах выбора маршрутов доступа. Для программно-определяемой архитектуры принципиально, что конфигурация подсистемы задается и изменяется программными средствами и должна быть типизирована и однозначно определена на всех узлах, что обеспечивает воспроизводимость режимов работы и корректность дальнейшего моделирования.

Для иллюстрации взаимодействия выделенных компонентов рассмотрим типовой сценарий выполнения операции записи с синхронной репликацией. Координатор принимает запрос записи от клиента, выполняет обращение к подсистеме метаданных для выбора реплик и маршрутизации, инициирует передачу данных на выбранные узлы хранения и формирует подтверждение клиенту после получения подтверждений фиксации от всех реплик. Таким образом, сетевое взаимодействие включает как пользовательские операции (передача данных), так и служебные операции (выбор реплик, подтверждения фиксации и при необходимости обновление статуса/журнала). На рисунке 1.4 представлена диаграмма последовательностей, отражающая общий порядок обмена сообщениями между клиентом, координатором, подсистемой метаданных и узлами хранения при выполнении операции записи с синхронной репликацией.

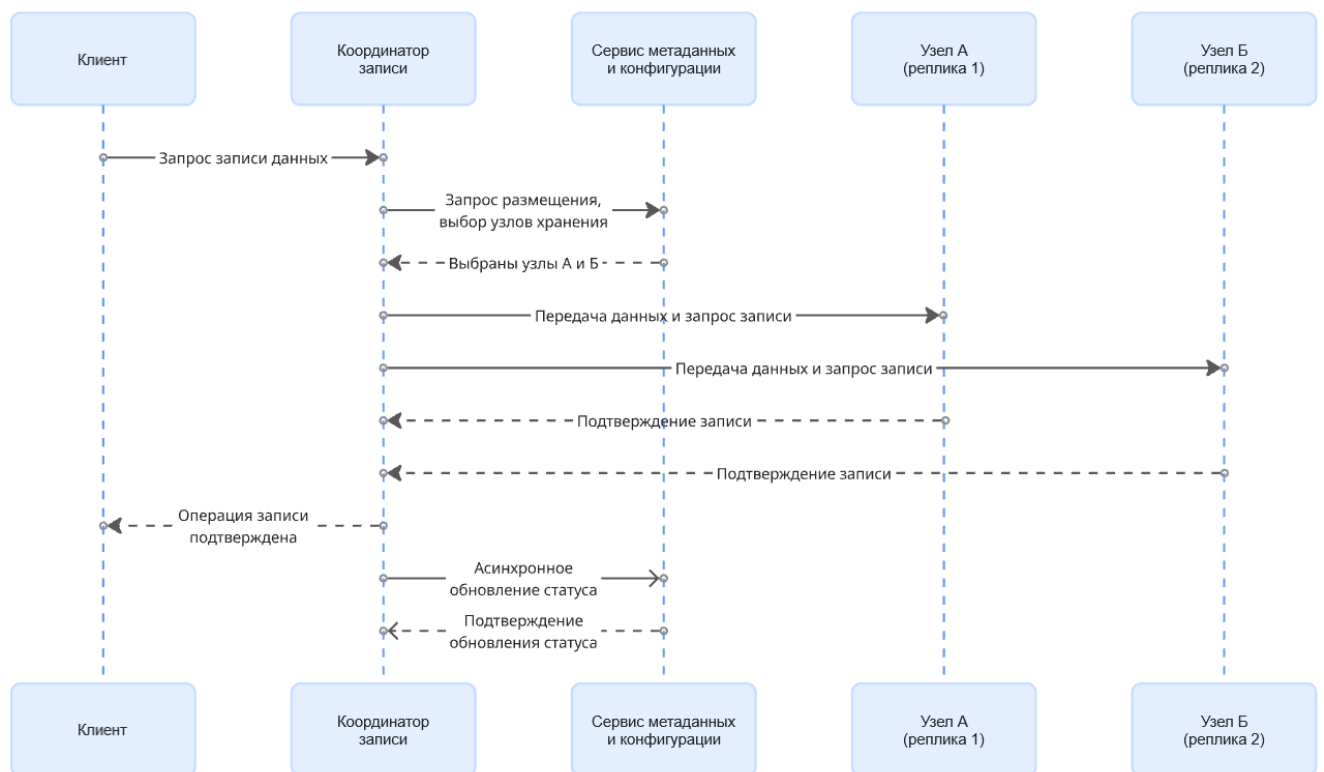


Рисунок 1.4 – Диаграмма последовательностей выполнения операции записи с синхронной репликацией (общая схема)

Рассматриваемая архитектура предполагает двухуровневую интеграцию с облачным центром обработки данных. На уровне туманных вычислений обслуживаются запросы прикладных сервисов на чтение и запись в пределах периферийной инфраструктуры, а облачный уровень используется для консолидации результатов записи, долговременного хранения и последующей обработки данных, что сокращает число межуровневых обращений и снижает зависимость времени отклика от состояния транспортных сегментов сети.

Для последующего анализа надежности и доступности, а также для корректного задания механизмов избыточности, в архитектурной модели целесообразно также фиксировать домены отказа. Указание доменов отказа необходимо для обоснования правил размещения реплик и для определения условий, при которых отказ части инфраструктуры приводит к недоступности данных или снижению качества обслуживания.

Схема архитектурных компонентов распределенной программно-определяемой подсистемы хранения туманного уровня и ее взаимодействия с верхним уровнем хранения приведена на рисунке (рисунок 1.4).



Рисунок 1.5 – Структурная схема взаимодействия с верхним уровнем хранения

В последующих подпунктах раздела рассматриваются процессы чтения и записи, процедуры согласования и синхронизации, а также источники дополнительной служебной нагрузки, через которые параметры избыточности влияют на производительность подсистемы хранения и нагрузку на телекоммуникационную сеть.

1.3.2 Концептуальное представление данных и единиц размещения при репликации

Для последующего анализа механизмов избыточности и оценки влияния параметров СХД на производительность необходимо определить представление данных и единицы, на уровне которых выполняются операции размещения, репликации, согласования и восстановления. В распределенных системах данные, как правило, рассматриваются как совокупность фрагментов заданного размера, что обеспечивает параллелизм доступа, упрощает балансировку нагрузки и позволяет регулировать объем сетевого обмена и служебных метаданных.

В рамках настоящей работы информационный массив пользователя (логический объект) представляется последовательностью блоков данных, которые разбиваются на фрагменты хранения, являющиеся минимальной единицей размещения на узлах хранения. Размер фрагмента выбирается с учетом соотношения между параллелизмом доступа, объемом метаданных и числом сетевых операций. Меньший размер фрагмента повышает гибкость распределения нагрузки, но увеличивает объем служебной информации и количество операций; больший размер фрагмента сокращает служебную нагрузку, однако ограничивает параллелизм и может снижать качество балансировки.

Избыточность в рассматриваемой подсистеме хранения в настоящей работе обеспечивается репликацией. Для каждого фрагмента хранения формируется набор реплик, размещаемых на различных узлах хранения, а число реплик определяется фактором репликации R . Репликация позволяет обеспечить доступность данных при отказах отдельных компонентов и упрощает процедуры восстановления,

поскольку такая возможность сводится к копированию данных с доступной реплики. Репликация увеличивает объем хранимых данных пропорционально числу реплик R и требует дополнительного межузлового обмена при записи и синхронизации данных, что приводит к росту задержек операций и увеличению нагрузки на коммуникационные ресурсы сети.

$$\tau_{\text{зап}}(R) = \tau_0 + \max_{r=1, \dots, R} (\tau_{\text{сеть}, r} + \tau_{\text{диск}, r}) + \tau_{\text{согл}}(R), \quad (1.1)$$

где τ_0 — постоянная составляющая времени обслуживания операции, включающая обработку запроса и формирование набора реплик, $\tau_{\text{сеть}, r}$ и $\tau_{\text{диск}, r}$ — задержки доставки и фиксации на r -й реплике, $\tau_{\text{согл}}(R)$ — дополнительная временная составляющая, обусловленная процедурами согласования реплик, принимается монотонно возрастающей функцией числа реплик R .

Корректность размещения реплик определяется правилами, учитывающими домены отказа. Под доменом отказа понимается группа компонентов, совместная деградация которых возможна вследствие единой причины. В системах классической архитектуры или в программных системах таким отказоустойчивым доменом является либо диск, либо контроллер, выход из строя, которого тоже может повлечь за собой остановку в работе, но такие системы не учитывают особенности своего расположения и связности с другими компонентами, среди которых могут быть следующие:

- HDD или SSD накопители,
- сервер,
- серверная стойка,
- ряд серверных стоек,
- серверная комната,
- центр обработки данных.

Для обеспечения заданной доступности данных реплики должны размещаться таким образом, чтобы отказ одного домена не приводил

к одновременной недоступности всех реплик соответствующего фрагмента. Следовательно, модель размещения должна учитывать топологию туманной инфраструктуры и возможные сценарии нарушения связности между сегментами телекоммуникационной сети. Распределенный способ хранения данных имеет преимущество перед классическими системами – в нем изначально заложена возможность обеспечения отказоустойчивости, связанная распределенной структурой такой системы [56]. Основная идея данного подхода заключается в том, что дополнительные реплики каждой группы данных находятся на различных узлах хранения и физически разнесены. Данные локации могут быть настроены как при запуске системы, так и добавлены в процессе ее эксплуатации. При проведении тестирования этот фактор стоит учитывать, так как значительное разнесение хранения данных может негативно сказываться на производительности, несмотря на обеспечение повышенной отказоустойчивости (рисунок 1.6). Подобное влияние топологии размещения, доменов отказа и параметров межузлового обмена на результаты испытаний отмечается при анализе особенностей моделирования нагрузки распределенных систем хранения, что требует фиксировать принятые допущения и режимы испытаний при интерпретации результатов [34,57].

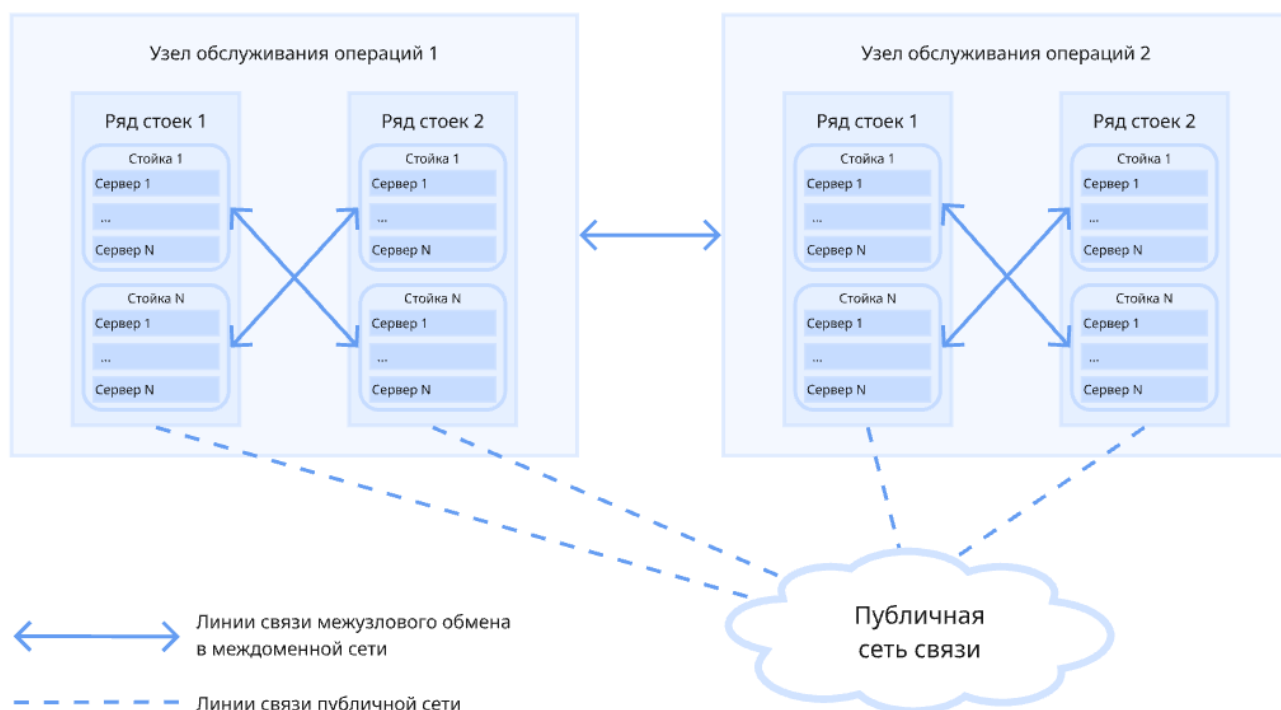


Рисунок 1.6 – Взаимодействие отказоустойчивых доменов

При синхронной репликации задержка обслуживания заявки записи определяется временем выполнения сетевого обмена и фиксации данных на репликах в соответствии с принятым режимом подтверждения, поэтому увеличение R приводит к росту объема служебных операций и, как следствие, влияет на задержки операций и использование коммуникационных ресурсов туманного уровня. Механизм доступа к данным в распределенной СХД опирается на метаданные, содержащие информацию о соответствии логических объектов и фрагментов хранения, а также сведения о расположении реплик и текущем состоянии узлов. Для каждого логического объекта формируется карта размещения, используемая контроллерами при маршрутизации операций чтения и записи и при выборе целевых узлов для размещения реплик.

1.3.3 Процесс записи

В рамках настоящей работы основное внимание уделяется операциям записи данных как наиболее ресурсоемким операциям в условиях репликации и синхронизации на туманном уровне [32]. В задачах нагрузочного тестирования распределенных систем хранения это проявляется особенно существенно: параметры записи и связанные с ними служебные операции определяют как итоговую пропускную способность, так и риск некорректной интерпретации результатов при неполной модели нагрузки [33,58]. Операции чтения в настоящем разделе подробно не рассматриваются, поскольку они не являются определяющими для выбранной постановки задачи и для дальнейшей формализации избыточности.

По результатам аналитического обзора установлено, что среди распространенных механизмов обеспечения избыточности в распределенных СХД (репликация и избыточное кодирование) в рамках настоящего исследования целесообразно рассматривать репликацию как основной механизм. Использование репликации обусловлено возможностью регулировать уровень отказоустойчивости путем изменения числа копий, а также относительной простотой выполнения

операций записи. По сравнению с кодовыми методами избыточности репликация, как правило, требует меньшего объема вычислительной обработки при записи, что важно для приложений, чувствительных к задержкам [59]. За счет того, что репликация позволяет определить количество копий блоков данных, записываемых на устройства, появляется возможность регулировать уровень отказоустойчивости. Для проводимого исследования будет использоваться изменение значения фактора репликации от 1 до 10 (количество копий блоков данных от 0 до 9).

При этом следует отметить особенность, возникающую при выборе фактора репликации, отличающуюся при факторе репликации 2 и факторах репликации 3 и более. Далее для иллюстрации обслуживания операции записи рассматривается типовой сценарий репликации с параметрами $R=2$ и 3, при котором служба записи, реализуемая первичным узлом или координатором записи, размещает реплики на удаленных узлах хранения. При этом под «диском» далее понимается либо локальный носитель, либо узел хранения, участвующий в выполнении операции записи.

При записи данных при факторе репликации $R=3$ (рисунок 1.7) клиент отправляет запрос на запись службе, отвечающей за запись данных на диск 1 (операция 1). При поступлении этого блока данных одновременно начинаются две операции – передача данного блока другим дискам (операция 2 и 3) и запись данного блока на диск, за который служба отвечает. Служба, отвечающая за запись на диск 2, записывает данные на диск 2, и только поступления результата синхронизации кэша `O_SYNC` возвращает подтверждение операции службе диска 1 (операция 4). То же самое происходит со службой диска 3 (операция 5). При поступлении подтверждений от служб всех дисков, служба диска 1 отправляет клиенту ответ о том, что блок данных успешно записан (операция 6).

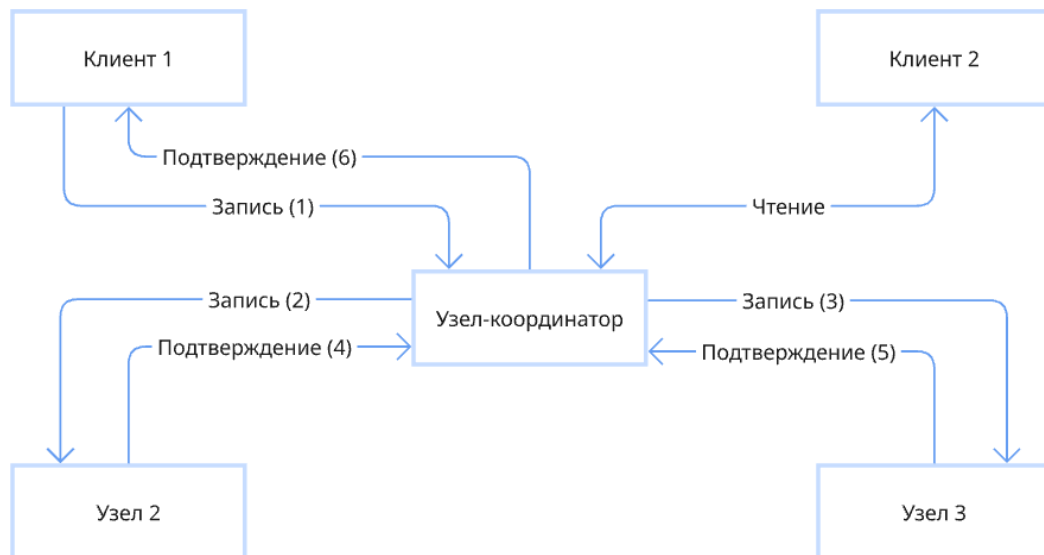


Рисунок 1.7 – Процесс записи на диск при факторе репликации 3

В случае выхода из строя какого-либо диска (или канала связи) актуальные будут считаться те данные, которые есть на большинстве дисков, что будет означать постоянную консистентность данных, даже в случае единовременного выхода из строя всех узлов системы (например, в случае аварийного отключения электропитания) [60,61].

При записи данных при факторе репликации $R=2$ (рисунок 1.8) клиент отправляет запрос на запись службе, отвечающей за запись данных на диск 1 (операция 1). При поступлении этого блока данных она одновременно начинает две операции – передача данного блока другому диску (операция 2) и записи данного блока на диск, за который она отвечает. Служба, отвечающая за запись на диск 2, записывает данные на диск 2, и только поступления результата синхронизации кэша O_SYNC возвращает подтверждение операции службе диска 1 (операция 3).



Рисунок 1.8 – Запись при факторе репликации 2

Если в этот момент происходит отказ компонента системы, обслуживающего диск 1, клиент не получает подтверждения выполнения операции, при этом на диске 2 изменение уже могло быть зафиксировано. В результате возникает неопределенность относительно исхода операции, и система хранения не может однозначно подтвердить клиенту факт ее успешного завершения. В зависимости от сценария восстановления и последующей работы распределенной СХД возможны различные исходы; наиболее характерным является рассогласование состояний реплик, при котором часть данных остается в неактуальном состоянии, а требуемая версия может отсутствовать на носителе, выбранном системой для дальнейшего обслуживания запросов. То есть, при факторе репликации $R=2$ отказ координатора до формирования подтверждения повышает риск рассогласования реплик и в отдельных сценариях допускает потерю данных даже при одиночном отказе, что необходимо учитывать при оценке вероятности потери данных.

Распределенная запись при репликации требует определения условий, при которых операция считается завершенной, и порядка распространения обновлений на реплики. Такие условия задаются режимом подтверждения записи и процедурами согласования. В общем виде режим подтверждения определяет минимальное число реплик, на которых данные должны быть зафиксированы до

возврата результата операции клиенту. Чем выше требуемое число подтверждений, тем выше устойчивость к отказам и тем ближе система к строгим требованиям согласованности, однако тем больше задержка обслуживания заявки записи и тем выше чувствительность к состоянию сети и доступности отдельных узлов. При синхронном подтверждении операция записи завершается после фиксации данных на заданном числе реплик, включая, при необходимости, все реплики [62]. Данный режим обеспечивает более строгую согласованность результата записи, однако увеличивает время завершения операции из-за межузлового обмена и ожидания подтверждений от нескольких узлов, потенциально расположенных в разных сегментах сети. При асинхронном подтверждении операция может считаться завершенной после фиксации данных на минимально достаточном наборе узлов, тогда как обновление остальных реплик выполняется в фоновом режиме, что сокращает время ответа, но повышает требования к фоновой синхронизации и восстановлению согласованности при отказах или потере связности. Процедуры восстановления при отказе узлов хранения или рассогласовании реплик формируют отдельную составляющую служебной нагрузки, включают выбор источника актуальной реплики, передачу данных на восстанавливаемый узел и обновление метаданных. В инфраструктуре туманных вычислений такой межузловой обмен использует коммуникационные ресурсы, конкурирует с прикладными потоками записи и при групповых отказах либо высокой интенсивности записи может увеличивать время обслуживания операций.

Таким образом, цепочка обслуживания операции записи в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений, включает (рисунок 1.8):

- маршрутизацию запроса и выбор целевых узлов размещения,
- передачу данных на реплики по сети,
- фиксацию данных на носителях,
- получение подтверждений согласно выбранному режиму согласования.



Рисунок 1.9 – Схема обслуживания операции записи с репликацией на уровне туманных вычислений

Указанные этапы формируют как пользовательскую нагрузку на узлы хранения, так и служебную нагрузку на сеть и контроллеры.

Следовательно, задача управления нагрузкой в данной системе сводится к формированию правил распределения операций доступа к данным и связанных с ними служебных процессов таким образом, чтобы сократить время завершения операций записи при соблюдении требований к доступности их выполнения и ограничении служебной межузловой нагрузки.

1.4 Подходы к балансировке нагрузки в распределенных системах

Поскольку в распределенной подсистеме хранения туманного уровня операция записи реализуется как композиция параллельных действий (передача данных, фиксация на репликах, подтверждение, служебные операции репликации/восстановления), качество обслуживания определяется не только внутренними механизмами согласования, но и тем, как поток операций

распределяется по доступным узлам и сетевым путям. Поэтому после рассмотрения процессов записи целесообразно перейти к обзору подходов балансировки нагрузки как ключевого инструмента управления потоками, очередями, задержками обслуживания заявок и конкуренцией пользовательскими и служебными операциями при ограничениях по коммуникационными и вычислительным ресурсам.

Балансировка нагрузки в распределенных подсистемах хранения рассматривается как совокупность методов, обеспечивающих управляемое распределение потока операций чтения/записи и связанных с ними служебными операциями (репликации, восстановления избыточности, перераспределения данных) между узлами хранения и коммуникационными ресурсами. В условиях туманной инфраструктуры данная задача осложняется выраженной неоднородностью обслуживающих ресурсов, вариативностью сетевых характеристик и высокой чувствительностью качества обслуживания к конкуренции служебных и пользовательских потоков операций. Следовательно, балансировка должна трактоваться не как «равномерное распределение запросов», а как механизм обеспечения целевых метрик *QoS* при наличии ограничений и структурных особенностей процесса записи.

Поскольку балансировка нагрузки в распределенных подсистемах хранения туманного уровня реализуется на разных уровнях архитектуры и опирается на неоднородные источники информации (локальные показатели узлов, сетевые характеристики, ограничения репликации и т.д.), единая универсальная типология оказывается недостаточной. В связи с этим далее используются несколько взаимодополняющих классификаций, позволяющих последовательно описать подходы.

1.4.1 Классификация по месту принятия решения

Балансировка на стороне клиента предполагает, что решение о выборе узла обслуживания принимается на стороне клиента (или клиентского компонента

доступа). Данный подход масштабируем и не требует единой управляющей точки, однако его эффективность зависит от качества и актуальности информации о состоянии системы (загрузке узлов, задержках сети), а также от устойчивости принятого правила к запаздыванию телеметрии.

Централизованная балансировка использует выделенный управляющий компонент (координатора или диспетчера), который принимает решения по распределению нагрузки, имея более целостное представление о состоянии распределенной СХД [63]. Потенциальным преимуществом является возможность учитывать не только локальную загрузку узлов, но и сетевую топологию, домены отказа и текущую интенсивность служебных операций. К ограничениям такого подхода относятся служебный обмен, необходимый для сбора информации о состоянии узлов, а также необходимость обеспечения отказоустойчивости управляющего компонента.

Децентрализованные схемы формируют решения распределенно, опираясь на локальные оценки состояния и детерминированные правила выбора (в том числе на основе размещения данных). Подобные схемы типичны для больших распределенных хранилищ, поскольку обеспечивают масштабируемость и устойчивость к частичным отказам, однако усложняют формальную оценку их влияния на хвостовые задержки и согласованность поведения в нестационарных условиях. В таких схемах решение принимается без единой управляющей точки, например, – в *JIQ* (от англ. Join-Idle-Queue) серверы передают диспетчерам информацию о состоянии простоя, что позволяет сократить служебный обмен и отказаться от сбора полной информации о текущей нагрузке [64]. Другой вариант децентрализации основан на детерминированном назначении объектов/запросов на узлы по функции размещения, который обеспечивает распределенный выбор узла без центрального координирующего компонента и уменьшает перестановки при изменении состава при изменении состава узлов распределенной СХД [65].

1.4.2 Классификация по степени использования информации о состоянии

По характеру используемой информации о текущем состоянии системы различают статические методы, не опирающиеся на оперативную обратную связь, и динамические методы, адаптирующие решения на основе наблюдаемых показателей нагрузки и производительности.

Статические методы не используют текущую обратную связь и распределяют запросы по фиксированному правилу. Их преимуществами являются малый объем служебного обмена и предсказуемость реализации. Однако на уровне туманных вычислений такая модель плохо учитывает изменение состояния узлов и межузловых соединений: восстановление, перераспределение данных и неоднородность компонентов могут увеличивать задержки обслуживания, включая значения верхних перцентилей распределения [66], изменения сетевых задержек и локальные перегрузки приводят к значительным колебаниям задержек, что делает статические схемы недостаточными для обеспечения стабильного QoS [67].

Динамические методы используют измеряемые показатели состояния (длины очередей, загрузку диска или CPU, сетевые задержки, оценку времени ожидания) и адаптируют распределение потока, стремясь минимизировать задержки и вероятность локальной перегрузки. Ключевым вопросом при использовании таких методов является выбор состава телеметрической информации. Учет большего числа параметров позволяет точнее описывать состояние системы, однако увеличивает объем служебного обмена при мониторинге и может снижать устойчивость управляющих решений при запаздывании измерений [68]. Отдельно выделяют динамические схемы с ограниченной телеметрией, в которых обратная связь задается частичными или квантованными сигналами (например, порогом перегрузки либо выборкой состояния ограниченного числа кандидатов). Поэтому на практике востребованы компромиссные решения, при которых объем служебного обмена при мониторинге остается ограниченным, а собираемая телеметрическая информация сохраняет достаточную информативность.

1.4.3 Балансировка по времени прохождения запросов

Для туманной инфраструктуры характерно наличие нескольких альтернативных путей обслуживания, которые с функциональной точки зрения могут быть эквивалентны, но различаются по задержкам передачи и текущей загрузке. Так задача балансировки сводится к выбору между сопоставимыми сценариями обслуживания на основе оценки полного времени прохождения запроса по каждому сценарию.

В работе [69] предложен принцип эквивалентности для адаптивной балансировки, в рамках которого решение формулируется через сравнение времен прохождения альтернативных сценариев обслуживания с учетом сетевых задержек и задержек ожидания в очередях. Рассматриваемый пример отражает характерный для уровня туманных вычислений компромисс: перераспределение запросов на альтернативный узел может уменьшить время ожидания при перегрузке одного из ресурсов, но одновременно увеличить сетевую составляющую задержки. Поэтому выбор узла должен основываться на оценке совокупного времени обслуживания или показателя QoS , а не только на локальном признаке загрузки отдельного ресурса. Для распределенных подсистем хранения это особенно существенно, поскольку операция записи включает межузловой обмен и фиксацию данных на удаленных репликах, а сетевые задержки становятся частью цепочки обслуживания операции. Актуальность учета сетевой составляющей дополнительно определяется тем, что в распределенных системах сведения о текущем состоянии элементов, в том числе о длинах очередей, поступают к управляющему механизму с ненулевой задержкой. Запаздывание такой информации может изменять свойства политик балансировки, учитывающих состояние системы (*от англ. state-aware policies*), включая устойчивость стратегии присоединения к кратчайшей очереди – JSQ (*от англ. Join-the-Shortest-Queue*). Влияние коммуникационных задержек на устойчивость JSQ и последствия использования устаревающей информации о состоянии системы рассматриваются в [70].

1.4.4 Модели обслуживания потоков и распределение ресурсов как аналитическая основа балансировки

Существенная часть исследований рассматривает балансировку нагрузки в тесной связи с моделями систем обслуживания и распределения ресурса, поскольку выбор узла обслуживания изменяет входные потоки в очереди и, тем самым, определяет задержки и вероятность деградации качества обслуживания.

В работе [71] представлена модель обслуживания трафика в облачной вычислительной системе с разделением процессорного ресурса (*от англ. Processor Sharing*) и аналитическим расчетом показателей качества обслуживания для потоков запросов. Важный методологический вывод для задач балансировки заключается в следующем: корректная оценка эффекта перераспределения нагрузки требует учета не только средних показателей, но и влияния изменившейся интенсивности потоков на режим обслуживания и мультиклассовую структуру заявок. Для распределенных подсистем хранения данный подход можно рассматривать как теоретическое обоснование необходимости связывать правила балансировки с моделью обслуживания, отражающей конкуренцию различных типов активности (пользовательские операции, служебные операции согласования/восстановления) и ограничения по обслуживающим ресурсам узла.

При размещении подсистемы хранения на уровне туманных вычислений ее функционирование определяется не только характеристиками узлов хранения, но и сетевыми условиями: ограничениями пропускной способности, особенностями маршрутизации и конкуренцией между пользовательским и служебным трафиком. В этой связи балансировка нагрузки должна учитывать не только локальные очереди узлов хранения, но и ограничения каналов межузлового обмена, особенно в режимах синхронной репликации, где сетевое взаимодействие определяет время завершения операции.

Работа [72] рассматривает обобщенную модель передачи мультисервисного трафика реального времени в сетях связи с учетом требований различных потоков, резервирования обслуживающих ресурсов и условий допустимости обслуживания.

Для задач распределенного хранения ценность данного направления состоит в том, что качество обслуживания операции записи зависит от способности сети обеспечить требуемую полосу и приемлемую задержку для межузловых обменов, включая служебные операции репликации и восстановления. Следовательно, балансировка нагрузки в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений, должна выполняться с учетом состояния межузловой сети. При выборе узлов хранения и регулировании служебных операций необходимо учитывать пропускную способность и задержки межузловых соединений, поскольку локальное выравнивание очередей без учета сетевых ограничений может приводить к перегрузке каналов и ухудшению показателей качества обслуживания.

Отличительной особенностью распределенных подсистем хранения является наличие двух различных составляющих нагрузки:

- пользовательская нагрузка, связанная с операциями чтения и записи, для которых существенное значение имеют задержка обслуживания и стабильность показателей QoS ,
- служебная нагрузка, связанная с репликацией, восстановлением избыточности и перераспределением данных – процессам, необходимыми для обеспечения надежности и согласованности хранения, однако использующими те же ресурсы узлов и межузловых соединений, что и пользовательские операции, вследствие чего могут увеличивать задержки обслуживания, в том числе в области верхних перцентилей распределения.

С практической точки зрения эффективная балансировка нагрузки должна включать не только диспетчеризацию пользовательских запросов по узлам, но и механизмы управления интенсивностью служебных операций (приоритезация, ограничение параллелизма и др.), поскольку иначе поток служебных операций может поглощать значимую долю обслуживающих ресурсов узла в моменты деградации или после отказов. Теоретически такой подход согласуется с представлением системы как сети обслуживания, в которой перераспределение потоков должно оцениваться по совокупному влиянию на показатели QoS всей

цепочки обслуживания, а не только отдельного ресурса или участка [69], в связи с этим правила управления должны быть согласованы с моделью обслуживания и распределения ресурсов [71].

1.4.5 Походы к балансировке нагрузки

В рамках дальнейшего исследования под сценарием (стратегией) балансировки понимается формализованное правило, определяющее назначение операций чтения/записи на узлы/реплики и регулирование потоков служебных операций, исходя из доступной информации о состоянии системы. В отличие от общих классификаций подходов, представленной выше, сценарии фиксируют конкретные механизмы выбора и, тем самым, задают сопоставимые базовые варианты для экспериментального сравнения.

Далее приведены типовые сценарии балансировки нагрузки:

1) сценарии без обратной связи (статические):

- равномерное распределение (*от англ. Round-Robin, RR*) – запросы циклически назначаются на узлы без учета текущего состояния, часто используется как базовый вариант сравнения [73];
- взвешенное распределение (*от англ. Weighted RR/Weighted Random*) – назначение производится пропорционально заданным весам, отражающим неоднородность обслуживающих ресурсов или доли ответственности узлов [74];
- детерминированное размещение (*от англ. Hash-based*) – выбор узла определяется функцией хеширования по ключу объекта/запроса, при этом основной эффект заключается в стабильности назначения и множества узлов распределенной подсистемы хранения, а баланс достигается «в среднем», но без реакции на перегрузку [65];

2) сценарии на основе локальной нагрузки (*от англ. State-aware*):

- минимальная очередь или минимальная загрузка (*от англ. Least Queue или Least Loaded*) – выбор узла выполняется по минимальной длине очереди,

загрузке CPU, загрузке дисковой подсистемы либо по комбинированному индексу состояния. Метод позволяет учитывать локальную перегрузку узлов, однако требует сбора и обработки телеметрической информации и чувствителен к запаздыванию и неточности измерений [75];

- присоединение к кратчайшей очереди (*от англ. Join-the-Shortest-Queue, JSQ*) – частный случай выбора по минимальной очереди, при котором запрос направляется на узел с наименьшей длиной очереди обслуживания. Подход широко используется в теории систем массового обслуживания как базовая модель анализа распределения нагрузки, однако для распределенных подсистем хранения требует уточнения понятия «очередь», поскольку операция записи включает дисковое обслуживание, межузловой обмен и получение подтверждений от узлов о создании реплик [70,76,77];

- оценка ожидаемого времени ответа (*от англ. Shortest Expected Delay*) – узел выбирается по минимуму прогнозной задержки, что дает более устойчивые результаты, чем только по длине очереди, если времена обслуживания неоднородны [78,79];

3) сценарии «низкой телеметрии» (вероятностные):

- выбор через d кандидатов (*от англ. Power-of-d Choices*) – стратегия, при которой из множества узлов случайно выбирается d кандидатов, после чего запрос направляется на узел, минимизирующий выбранный критерий (например, длину очереди, загрузку обслуживающего ресурса узла или оценку ожидаемой задержки обслуживания). По сравнению с полным вариантом выбора по минимальной очереди (*JSQ*), данный подход обеспечивает сопоставимое распределение нагрузки при меньшем объеме телеметрического обмена, поскольку состояние оценивается только для ограниченного числа узлов-кандидатов [80,81];

- случайное распределение с ограничением перегрузки (*от англ. Random Threshold*) – стратегия, при которой узел первоначально выбирается случайным образом, однако при превышении заданного порога загрузки (или длины очереди) выполняется повторный выбор узла либо запрос перенаправляется в

альтернативный (резервный) пул, при этом подход реализует ограниченную адаптацию к перегрузкам при минимальных требованиях к телеметрии и вычислительным ресурсам [82];

4) сценарии с учетом сети и топологии (*от англ. Network-/Locality-aware*):

- выбор по сетевой близости (*от англ. Min-RTT/Min-hop*) – узел выбирается среди допустимых реплик по минимуму сетевой задержки или по принадлежности к близкому домену (сегменту сети), за счет чего важен для туманных инфраструктур, где межузловая сеть часто ограничивает время операции [83,84];

- комбинированный критерий (*от англ. Latency-aware score*) – выбор по интегральной оценке вида: прогноз очереди/обслуживания и сетевой RTT/потери, является естественной формой сценария выбора по времени прохождения, пригодной для аккуратного теоретического описания [85];

5) сценарии, специфичные для записи с репликацией (*от англ. Replication-aware*):

- назначение через первичный координатор (*от англ. Primary-Based*) – стратегия, при которой для каждого набора данных (шарда, партии) назначается первичный узел (*от англ. primary*) или фиксированный координатор, через который проходят операции записи и координация репликации, затем в рамках балансировки решается задача распределения ролей между узлами кластера (переназначение лидеров/партий) с целью выравнивания нагрузки и соблюдения ограничений по отказоустойчивости и размещению данных [86];

- кворумные (*от англ. Quorum-Aware/ALL-aware*) – выбор набора реплик для записи (или маршрута записи) производится так, чтобы минимизировать время достижения условия подтверждения (например, ALL или k-of-R), учитывая реплики с повышенной задержкой отклика и сетевые ограничения [87,88];

- выбор лучшего подмножества реплик (*от англ. Subset Selection*) – если архитектура допускает вариативность набора реплик набора реплик, участвующих в завершении операции при соблюдении требований к кворуму и

отказоустойчивости (при соблюдении доменных ограничений), балансировка может дополнительно учитывать задержку и текущую нагрузку при формировании такого подмножества [89,90];

б) сценарии совместного управления пользовательской и служебной нагрузкой:

- фиксированное ограничение фоновой активности (*от англ. Background Throttling*) – служебных операций (восстановление, перебалансировка) выполняются с заранее заданными лимитами по пропускной способности и (или) уровню параллелизма, что позволяет снизить интерференцию с пользовательскими операциями и стабилизировать показатели качества обслуживания (*QoS*) [91];

- адаптивное ограничение фоновой активности (*от англ. Adaptive Throttling*) – параметры ограничения служебных операций (восстановление, перебалансировка) изменяются на основе наблюдаемых признаков деградации *QoS* пользовательских операций, таких как рост очередей, увеличение p95/p99 задержек и признаки перегрузки сети. Данный класс сценариев особенно актуален для туманной инфраструктуры, где доступные коммуникационные и обслуживающие ресурсы узла, выделяемые под поток служебных операций, могут существенно и быстро изменяться [92].

Для систематизации рассмотренных подходов к балансировке нагрузки и фиксации взаимосвязей между классификационными признаками и типовыми сценариями далее приведена матрица соответствия. На рисунке 1.9 представлены сценарии балансировки и их отнесение к классам по месту принятия решения, степени использования информации о состоянии, объекту балансировки и учитываемым факторам (сетевые условия, репликация/кворум, требования *QoS*). Цветовая маркировка отражает объект балансировки.

Обзор показывает, что универсальной стратегии балансировки для распределенных подсистем хранения туманного уровня не существует: результат зависит от места принятия решения, объема и актуальности телеметрии, объекта управления и факторов среды, прежде всего сетевых условий, репликации и требований к хвостовым процентилям *QoS*. Для тумана характерны

нестационарность и неоднородность обслуживающих ресурсов узла, обусловленные служебными операциями восстановления и перераспределения, а также колебаниями сетевых задержек, что приводит к значительной вариативности времени ответа.

В этих условиях правила выбора узла, основанные только на одном локальном показателе, например длине очереди, загрузке *CPU* или дисковой подсистемы, имеют ограниченную применимость, требуют своевременного получения телеметрической информации и не отражают сетевую составляющую, входящую в цепочку обслуживания операции записи при репликации. Поэтому более обоснованными являются динамические подходы, использующие ограниченный, но информативный набор параметров состояния и формирующие решение на основе оценки времени завершения операции с учетом сетевых задержек и условий получения подтверждений от реплик.

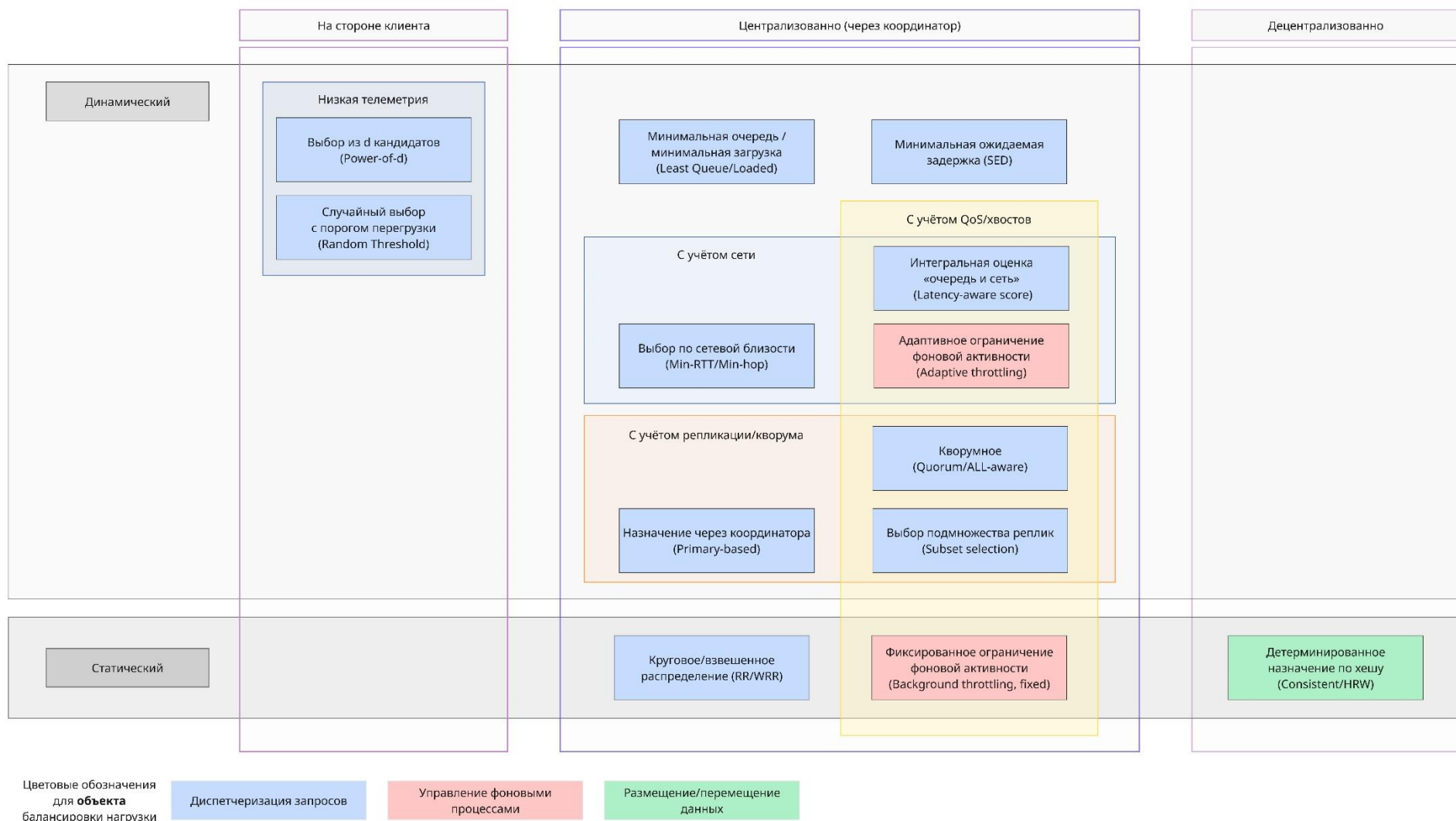


Рисунок 1.10 – Карта сценариев балансировки нагрузки и их соответствие классификациям

1.5 Выводы

В настоящей главе:

- показано, что в современных телекоммуникационных системах функции хранения данных становятся одним из инфраструктурных компонентов, влияющих на качество обслуживания наряду с вычислительными и сетевыми ресурсами,
- обосновано, что перенос операций доступа к данным ближе к источникам и потребителям позволяет сократить зависимость времени отклика от состояния транспортных сегментов сети и уменьшить объем межуровневого трафика. При этом уровень туманных вычислений в настоящей работе рассматривается не как вспомогательный уровень кэширования, ориентированный преимущественно на чтение, а как среда выполнения операций чтения и записи в распределенной подсистеме хранения,
- установлено, что для рассматриваемой архитектуры наиболее существенное значение имеет операция записи, поскольку при репликации включает передачу данных на несколько узлов хранения, фиксацию данных на репликах, получение подтверждений и выполнение служебных операций согласования. Именно операция записи формирует значительную часть межузлового обмена и определяет необходимость дальнейшего анализа задержек, готовности и служебной нагрузки,
- выполнено обоснование выбора распределенной программно-определяемой подсистемы хранения как архитектурной основы исследования. Такой тип подсистемы хранения соответствует требованиям инфраструктуры туманных вычислений, поскольку допускает распределенное размещение узлов, масштабирование по числу элементов, программное управление конфигурацией и поддержку отказоустойчивых режимов работы,
- рассмотрены основные механизмы обеспечения избыточности в распределенных подсистемах хранения и показано, что в рамках настоящего

исследования целесообразно сосредоточиться на репликации. Репликация позволяет явно задавать число копий данных, учитывать домены отказа и анализировать влияние параметров избыточности на доступность выполнения операций записи, задержки обслуживания и объем служебного межузлового обмена,

- показано, что увеличение уровня избыточности повышает устойчивость хранения, однако одновременно увеличивает объем передаваемых данных, число подтверждений, нагрузку на коммуникационные ресурсы и время завершения операции записи. Следовательно, выбор параметров репликации и режима подтверждения записи должен выполняться с учетом компромисса между доступностью, производительностью и ограничениями сетевой инфраструктуры,

- проведен анализ подходов к балансировке нагрузки в распределенных подсистемах хранения и установлено, что статические схемы распределения запросов недостаточны для условий инфраструктуры туманных вычислений, поскольку не учитывают неоднородность узлов, изменение сетевых задержек, накопление пользовательских очередей и влияние служебных операций репликации, синхронизации и восстановления,

- обоснована необходимость перехода от общего рассмотрения архитектуры подсистемы хранения к формализованной модели управления нагрузкой, в которой должны учитываться пользовательские операции записи, служебные операции поддержания репликации, ограничения межузловой сети и требуемая готовность выполнения операции записи. Выявленные особенности определяют содержание следующей главы, в которой разрабатывается модель распределения нагрузки в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений.

2 Разработка модели управления информационными потоками распределенной подсистемы хранения данных туманного уровня

В системе рекомендаций ITU-T понятие готовности системы рассматривается как составная часть понятийного аппарата телекоммуникаций и качества функционирования сетей связи, что подтверждается классификацией серии E.800-E.899, которая отнесена к области качества телекоммуникационных услуг, соответствующих моделей, целевых показателей и планирования эксплуатационной устойчивости. В рекомендации ITU-T E.800 готовность (*от англ. availability*) включена в состав более общего понятия эксплуатационной надежности и устойчивости функционирования (*от англ. dependability*), наряду с безотказностью (*от англ. reliability*), ремонтпригодностью (*от англ. maintainability*) и обеспеченностью технического обслуживания (*от англ. maintenance support*). Следовательно, в логике ITU-T готовность выступает не как вспомогательная общеинженерная характеристика, а как нормативно значимый показатель, применяемый при анализе способности телекоммуникационной системы сохранять состояние, необходимое для выполнения требуемых функций при оказании услуг связи.

Наиболее прямое подтверждение отраслевой принадлежности данного показателя содержится в рекомендации ITU-T E.862 «Планирование эксплуатационной надежности телекоммуникационных сетей» (*от англ. Dependability planning of telecommunication networks*). В рекомендации указано, что предметом рассмотрения являются модели и методы планирования

эксплуатационных характеристик телекоммуникационных сетей, а эксплуатационная надежность и устойчивость функционирования описывается через показатели готовности (*от англ. availability performance parameters*), безотказности, ремонтпригодности и обеспеченности обслуживания. Дополнительное подтверждение данной позиции содержится в рекомендации ITU-T Y.1540, где готовность прямо включена в состав параметров, используемых для задания и оценки характеристик передачи IP-пакетов и соответствующих услуг передачи данных. Следовательно, применение показателя готовности к распределенным подсистемам хранения, функционирующим в инфраструктуре туманных вычислений, согласуется с терминологической и методической базой ITU-T, поскольку такие подсистемы рассматриваются как элементы телекоммуникационного сервиса.

В ранее опубликованной работе [93] предложен метод оценки готовности распределенной подсистемы хранения туманного уровня к обслуживанию операций записи, учитывающий параметры репликации, порог подтверждения записи и состояние инфраструктурных элементов цепочки обслуживания операции. В настоящей главе данный результат используется не как самостоятельная оценка надежности подсистемы хранения, а как ограничение при построении модели управления нагрузкой. Такой подход позволяет связать выбор управляющих воздействий с требованием сохранения возможности завершения операции записи в заданном режиме подтверждения.

2.1 Метод оценки готовности выполнения операции записи как ограничение при оптимизации производительности

В распределенной подсистеме хранения туманного уровня повышение производительности неизбежно связано с компромиссами между задержками обслуживания заявок в системе, объемом межузлового обмена и устойчивостью системы обслуживания при отказах. Механизмы избыточности и согласования (включая репликацию и режим подтверждения) повышают устойчивость сервиса,

однако одновременно увеличивают нагрузку на коммуникационные и вычислительные ресурсы и тем самым могут ухудшать показатели качества обслуживания.

Распределенная подсистема хранения предоставляет набор целевых функций, каждая из которых является значимой с точки зрения эксплуатации:

- 1) запись (фиксация новых данных и обновлений),
- 2) чтение (получение ранее зафиксированных данных),
- 3) хранение (поддержание данных с заданным уровнем сохранности/целостности в условиях отказов и восстановлений).

С учетом указанного работоспособность сервиса целесообразно задавать в виде вектора функциональной готовности:

$$A^{\text{сервиса}} = (A^{\text{сеть}}, A^{\text{запись}}, A^{\text{чтение}}, A^{\text{хранение}}). \quad (2.1)$$

где $A^{\text{сеть}}$ – готовность сетевой инфраструктуры, обеспечивающей связность между компонентами сервиса и возможность передачи пользовательского и служебного трафика с учетом доступности каналов связи, сетевых узлов и маршрутов обслуживания, $A^{\text{запись}}$ и $A^{\text{чтение}}$ – готовность выполнения операций записи и чтения в заданном режиме обслуживания, а $A^{\text{хранение}}$ – готовность обеспечения функции хранения в принятом профиле требований (включая сохранение требуемой избыточности и целостности данных). Запись в виде вектора подчеркивает, что готовность сервиса рассматривается как совокупность функциональных компонент, каждая из которых должна учитываться при задании эксплуатационных ограничений.

Поскольку все перечисленные функции являются обязательными, система считается готовой только при выполнении требований по каждой компоненте вектора. Для задания эксплуатационного ограничения вводится агрегированная характеристика готовности сервиса по правилу конъюнкции (все функции должны быть доступны):

$$A^{\text{сервиса}} = \min\{A^{\text{сеть}}, A^{\text{запись}}, A^{\text{чтение}}, A^{\text{хранение}}\}. \quad (2.2)$$

Такое определение является консервативным и не требует дополнительных допущений о независимости функциональных отказов, при этом напрямую отражает требование обязательности всех функций.

Далее устойчивость системы обслуживания используется как эксплуатационное ограничение при оптимизации производительности: параметры и режимы управления, потенциально повышающие производительность (например, локализация размещения данных, изменение правил выбора реплик, перераспределение обслуживающих ресурсов узла между пользовательскими заявками и служебными операциями), допустимы только при выполнении порогового условия

$$A^{\text{сервиса}} \geq A^{\min}. \quad (2.3)$$

Тем самым область допустимых решений определяется пересечением требований по производительности и требований по готовности сервиса хранения.

В настоящей работе показатель устойчивости обслуживания формализуется через коэффициент готовности A , используемый в отечественной традиции надежности [94]. В англоязычной литературе близкий по смыслу показатель обычно обозначают как Availability, при этом различия и сопоставление терминов Availability и Reliability подробно обсуждаются, в частности, в работе [22]. В дальнейшем обозначение A сохраняется, чтобы обеспечить согласованность с принятой терминологией и используемыми расчетными соотношениями.

При этом также важно различать две группы характеристик надежности распределенного хранения: готовность выполнения функций (возможность выполнить чтение/запись и получить однозначный результат в заданной семантике подтверждения) и сохранность данных (риск необратимой утраты/повреждения данных при отказах). В настоящей главе показатели сохранности рассматриваются

как отдельная характеристика схемы избыточности и не подменяют собой показатели готовности, при этом компонент $A^{\text{хранение}}$ интерпретируется как готовность подсистемы обеспечивать функцию хранения в заданном профиле требований, а оценка вероятности потери данных используется как самостоятельная метрика контроля устойчивости хранения.

В рамках дальнейшего анализа в качестве лимитирующей компоненты вектора рассматривается функция записи, поскольку запись является технологически наиболее сложной и ресурсоемкой операцией – требует координации реплик и достижения условия подтверждения, а также порождает дополнительный поток служебных операций (репликация, журналирование, восстановление согласованности). Кроме того, при частичных отказах и деградациях типичным эксплуатационным режимом является «только чтение» (*от англ. read-only*), при котором чтение может сохраняться, тогда как запись становится недоступной или не может быть подтверждена в требуемом режиме [95]. В связи с этим далее используется гипотеза $A^{\text{запись}} \leq A^{\text{чтение}}$, а выполнение ограничения (2.3) проверяется по более строгой компоненте $A^{\text{запись}}$ при фиксированных требованиях к функции хранения.

Относительно функции хранения, необходимо отметить, что требования к данной функции обеспечиваются выбранной схемой избыточности и процедурами поддержания реплик на рассматриваемом интервале, то есть $A^{\text{хранение}} \geq A^{\min}$ и в этих условиях выполнение (2.3) сводится к проверке $A^{\text{запись}} \geq A^{\min}$ при учете гипотезы $A^{\text{запись}} \leq A^{\text{чтение}}$.

2.1.1 Расчет коэффициента готовности

В настоящем разделе коэффициент готовности A интерпретируется как готовность выполнения операции записи в распределенной подсистеме хранения туманного уровня при репликации [96]. Рассматривается режим репликации с пороговым подтверждением записи: операция считается завершенной после

фиксации данных на W целевых репликах из R , где $1 \leq W \leq R$. Параметр W задает требуемое число подтверждений, необходимых для завершения операции записи. Предельный случай $W=R$ соответствует режиму полного подтверждения записи, при котором данные должны быть зафиксированы на всех R целевых репликах, далее этот режим используется как частный случай для сопоставления.

Доступность узла хранения для выполнения операции записи определяется доступностью элементов локальной цепочки обслуживания, включающей серверную платформу, дисковый накопитель и журнальное устройство. Указанные компоненты рассматриваются как последовательные зависимости, то есть отказ любого из них приводит к недоступности узла для записи на период восстановления. Особенности влияния подсистемы хранения и процессов синхронизации на функционирование распределенной инфраструктуры рассмотрены [62]. Для конфигурации с журналированием дополнительно учитывается, что отказ журнального устройства приводит к недоступности связки «журнал-диск» для записи до восстановления, поскольку фиксация выполняется через журнальное устройство.

Далее вводится коэффициент готовности узла хранения $A^{\text{узел}}$ с учетом отказов и восстановления указанных компонентов, после чего формируется зависимость коэффициента готовности выполнения операции записи $A(R, W, \dots)$, используемая для задания ограничения (2.3) при выборе допустимых значений фактора репликации.

Операция записи считается выполнимой тогда и только тогда, когда одновременно доступны не менее W целевых узлов размещения реплик и на каждом из них обеспечивается возможность фиксации данных (получение подтверждения).

Далее вводится коэффициент готовности узла хранения $A^{\text{узел}}$, характеризующий долю времени, в течение которого узел доступен для выполнения записи.

Коэффициент готовности узла как ремонтпригодного объекта определяется классическим соотношением (2.4):

$$A_{\text{узел}} = \frac{MTBF_{\text{узел}}}{MTBF_{\text{узел}} + MTTR_{\text{узел}}}, \quad (2.4)$$

где $MTTR_{\text{узел}}$ – среднее время восстановления узла до возвращения в штатную работу подсистемы хранения. При этом $MTTR$ в работе понимается как среднее время восстановления после одного отказа, оцененное по совокупности отказов на выбранном интервале наблюдения либо заданное как параметр расчетной модели. Величина $MTTR_{\text{узел}}$ зависит от регламентов эксплуатации, доступности запасных частей, удаленности туманных узлов и других факторов.

Поскольку узел хранения в эксплуатации восстанавливается после отказа (замена/восстановление компонентов и возврат в обслуживание), он рассматривается как ремонтпригодный объект. В связи с этим исходные значения $MTBF$ компонентов, приведенные далее, используются как оценка средней наработки до отказа на рассматриваемом интервале, а для неремонтируемых компонентов $MTBF$ интерпретируется как приближение $MTTF$ при допущении постоянной интенсивности отказов.

При допущении однородности узлов и статистической независимости их недоступности коэффициент готовности выполнения операции записи при факторе репликации R и пороге подтверждения W определяется как вероятность доступности не менее W узлов из R (2.5):

$$A_{\text{запись}}(R, W) = \sum_{k=W}^R \binom{R}{k} A_{\text{узел}}^k (1 - A_{\text{узел}})^{R-k}, \quad (2.5)$$

где $A_{\text{узел}}$ – коэффициент готовности узла хранения к выполнению операции записи, определяемый готовностью элементов локальной цепочки обслуживания, включая

серверную платформу, дисковый накопитель и журнальное устройство. Тем самым в дальнейшем ограничение (2.3) применяется к $A(R, W)$, а не к агрегированной характеристике кластера «по всем узлам».

Следует отметить, что влияние увеличения R на $A^{\text{запись}}$ определяется выбранным порогом подтверждения W , а в режиме полного подтверждения записи ($W=R$) рост R повышает требования к одновременной доступности всех реплик и может снижать готовность выполнения записи при прочих равных. При пороговом подтверждении увеличение R расширяет множество допустимых сочетаний доступных реплик и может повышать $A^{\text{запись}}$. Однако рост числа реплик сопровождается увеличением межузлового обмена и объема служебных операций, связанных с поддержанием избыточности. Средняя наработка до отказа узла определяется по данным о надежности его составных компонентов и структуре их влияния на доступность узла для записи. Значения $MTBF$, приведенные в таблице 3, используются как исходные данные для расчета вероятности отказа компонентов на годовом интервале при допущении постоянной интенсивности отказов. Следует учитывать, что $MTBF$ зависит от условий эксплуатации и методики получения, поэтому в работе фиксируются принятые значения и выполняется сопоставимый пересчет на годовой интервал.

В рамках модели узел хранения рассматривается как ремонтпригодный объект (после отказа выполняется восстановление или замена, затем – возврат в обслуживание), поэтому $MTBF$ в расчетах используется как оценка средней наработки до отказа на рассматриваемом интервале. При этом для неремонтируемых элементов $MTBF$ интерпретируется как приближение $MTTF$ при указанных допущениях. Методическая корректность использования $MTBF$ или интенсивностей отказов в прогнозных расчетах обосновывается применением стандартных подходов к прогнозированию надежности электронной аппаратуры [97,98]. Для отдельных подсистем значения $MTBF$ дополнительно подтверждаются открытыми техническими спецификациями производителей [99].

При интерпретации значений $MTBF$ следует учитывать, что интенсивность отказов аппаратных компонентов во времени, как правило [100], имеет

характерную «ваннообразную» форму [101]: повышенная интенсивность возможна на раннем этапе эксплуатации (ранние отказы), затем наблюдается длительный участок сравнительно стабильной интенсивности (нормальная эксплуатация), после чего интенсивность возрастает по мере износа (рисунок 2.1). В настоящей работе расчеты коэффициента готовности выполняются для этапа нормальной эксплуатации, то есть без учета эффекта ранних отказов и без учета износа на завершающем этапе жизненного цикла, что позволяет использовать допущение постоянной интенсивности отказов и применять $MTBF/MTTF$ как исходные параметры прогнозной оценки доступности.

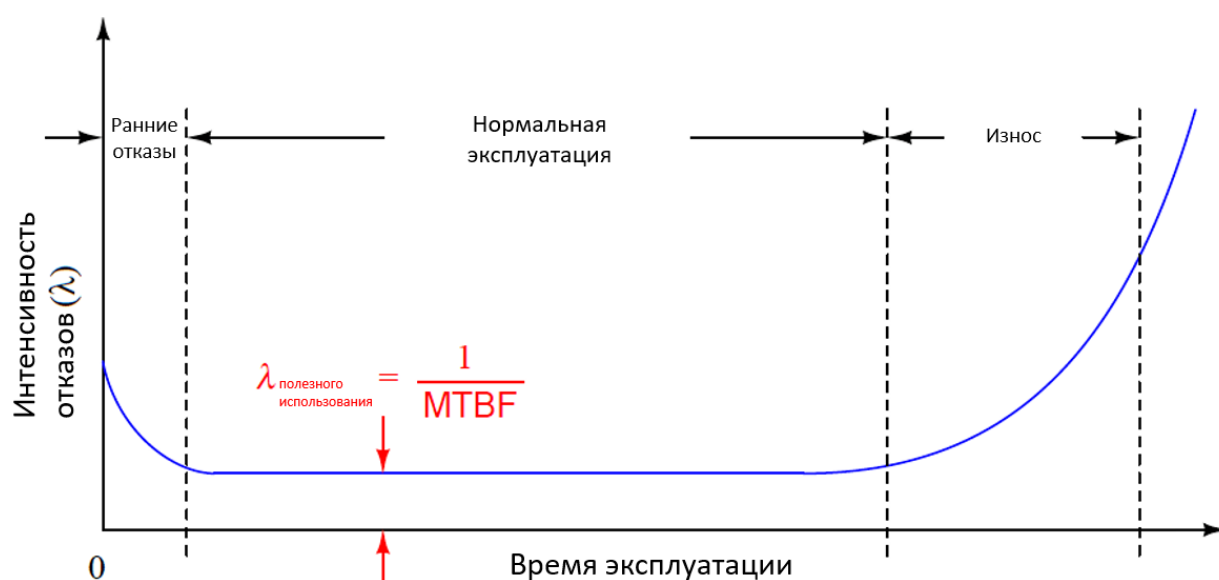


Рисунок 2.1 – Типовая зависимость интенсивности отказов компонентов от времени эксплуатации («ваннообразная» кривая)

Таблица 2.1 – Среднее время между отказами компонентов сервера

Компонент	MTBF (часов)	MTBF (лет)
Блок питания	100 000	11,42
Системная плата	300 000	34,25
Процессор	1 000 000	114,16
Оперативная память	1 000 000	114,16
Вентилятор	100 000	11,42
Контроллер HDD	300 000	34,25
Сетевой адаптер	300 000	34,25

В дальнейшем при анализе режимов управления рассматривается ремонтпригодная распределенная подсистема хранения, функционирующая на уровне туманных вычислений, состояние которой оценивается с учетом эксплуатационной готовности узлов и возможности выполнения операции записи в заданном режиме репликации и подтверждения. В данном разделе коэффициент готовности трактуется как характеристика доступности сервиса (способности узла/системы выполнять операции) и не отождествляется с вероятностью необратимой утраты данных (сохранностью). Сохранность рассматривается как отдельная характеристика, поскольку репликация по-разному влияет на доступность выполнения операций и на риск потери данных.

Принятые допущения:

- для оценки отказов компонентов на интервале эксплуатации используется допущение о постоянной интенсивности отказов (экспоненциальный закон распределения времени до отказа),
- отказы разных компонентов на интервале t считаются независимыми, допущение применяется на уровне агрегирования подсистем узла и используется для получения итоговых оценок готовности,
- под отказом узла понимается потеря способности узла выполнять операции записи (в принятом режиме подтверждения, включая деградированные состояния, при которых чтение может сохраняться, однако запись запрещена или не может быть подтверждена), а не факт физического обесточивания. Для двухсокетной платформы узел содержит два процессорных домена, отказ любого домена приводит к недоступности узла для штатной работы подсистемы хранения и требует регламентной остановки и переконфигурации или замены компонентов, поэтому в рамках модели трактуется как отказ узла.

Для компонента с известным значением $MTBF_i$ вероятность отказа на интервале t задается выражением (2.6):

$$P_i(t) = 1 - \exp\left(-\frac{t}{MTBF_i}\right), \quad (2.6)$$

где далее используется $t = 8766$ ч (1 год).

Для типовых структур применяются следующие правила.

Последовательная структура (отказ подсистемы при отказе любого из n одинаковых элементов) (2.7):

$$P_{\text{серия}}(t) = 1 - (1 - P(t))^n. \quad (2.7)$$

Резервирование (отказ подсистемы при отказе обоих элементов) (2.8):

$$P_{\text{резерв}}(t) = (P(t))^2. \quad (2.8)$$

Две независимые резервированные группы (отказ подсистемы наступает при отказе любой группы) (2.9):

$$P_{\text{группы}}(t) = 1 - (1 - P_{\text{группы}}(t))^2, P_{\text{группы}}(t) = P_{\text{резерв}}(t). \quad (2.9)$$

Отказ платформы узла при отказе любой из неоднородных независимых подсистем (2.10):

$$P_{\text{платф}}(t) = 1 - \prod_j (1 - P_j(t)). \quad (2.10)$$

Для дальнейших вычислений используется конфигурация узла с двумя процессорными доменами, двумя независимыми группами охлаждения и двумя независимыми парами сетевых адаптеров. В таблице приведены выражения и численные значения вероятностей отказа подсистем на интервале 1 год (8766 часов). Следует учитывать, что в практических источниках при пересчете показателей надежности к годовому интервалу могут использоваться различные значения длительности года (например, 8766 часов или 8760 [97]). Поэтому при

проведении расчетов необходимо явно фиксировать принятый коэффициент пересчета и проверять его соответствие исходным допущениям, принятым в используемых источниках.

Вероятность отказа каждого узла зависит от вероятности отказа каждого из перечисленных компонентов в таблице 3, для каждого из этих компонентов рассчитана вероятность отказа в течение года (таблица 2.2) с учетом фактического количества однотипных компонентов и характера их включения в структуру отказов узла.

Таблица 2.2 – Вероятности отказа подсистем сервера за год

Подсистема узла	Принятая структура	Вероятность отказа подсистемы за год
Питание	2 блока питания, резервирование, отказ обоих блоков	$P_{\text{БП}} = p_{\text{бп}}^2 = 0,007044$
Системная плата	одиночный компонент	$P_{\text{СП}} = p_{\text{сп}} = 0,028797$
Вычислительная подсистема	2 домена, состоящих из процессора и 4 модулей памяти, отказ любого домена, отказ домена наступает при отказе ЦПУ или любого ОЗУ данного домена	$P_{\text{ВП}} = 1 - (1 - P_{\text{ВП}})^2 = 0,083928$
Охлаждение	2 группы по 2 вентилятора, резервирование, отказ любой группы	$P_{\text{ОХ}} = 1 - (1 - p_{\text{вент}}^2)^2 = 0,014038$
Контроллер HDD	одиночный компонент	$P_{\text{контр}} = p_{\text{контрл}} = 0,028797$
Сетевая подсистема	2 пары сетевых плат, резервирование, отказ любой пары	$P_{\text{СЕТЬ}} = 1 - (1 - p_{\text{адапт}}^2)^2 = 0,001658$

Расчет выполнен в два этапа. На первом этапе по заданному значению *MTBF* для каждого типа компонента определялась вероятность отказа одного экземпляра в течение года при допущении постоянной интенсивности отказов и фиксированной длительности года, принятой в работе. На втором этапе учитывалось количество однотипных компонентов в составе сервера. При этом различались случаи, когда наличие нескольких элементов образует резервирование и отказ подсистемы наступает только при отказе всех установленных элементов

данного типа за рассматриваемый интервал (например, блоки питания, вентиляторы, сетевые адаптеры), и случаи, когда увеличение числа элементов означает увеличение числа потенциальных точек отказа и отказ подсистемы трактуется как отказ любого элемента (в частности, для подсистемы оперативной памяти отказ любого модуля рассматривается как отказ подсистемы памяти).

В расчетах вероятностей отказа подсистем на годовом интервале восстановление/замена компонентов в пределах года не учитывались, восстановление работоспособности узла далее учитывается отдельно через среднее время восстановления при расчете коэффициента готовности.

Итоговая вероятность отказа платформы узла за год определяется объединением событий отказа подсистем (2.11):

$$P_{\text{платф}}(t) = 1 - \prod_j (1 - P_j(t)) = 0,155462 \quad (2.11)$$

В рамках экспоненциальной модели отказов (2.12):

$$P_{\text{узел}}(t) = 1 - \exp(-\lambda_{\text{узел}} t). \quad (2.12)$$

Отсюда интенсивность отказов узла (2.13):

$$\lambda_{\text{узел}} = - \frac{\ln(1 - P_{\text{узел}}(t))}{t}, \quad (2.13)$$

и средняя наработка на отказ (2.14):

$$MTBF_{\text{узел}} = \frac{1}{\lambda_{\text{узел}}}. \quad (2.14)$$

Диски являются одним из ключевых компонентов распределенных систем хранения данных и одной из наиболее частых причин отказов, поэтому показатели отказоустойчивости распределенной подсистемы хранения в существенной степени зависят от надежности дисковых накопителей. Для накопителей в практических источниках широко используются два связанных показателя: годовой коэффициент отказов *AFR* (от англ. Annualized Failure Rate), представляющий долю накопителей, вышедших из строя за год, и показатель средней наработки до отказа (*MTTF*) либо средней наработки между отказами (*MTBF*), применяемый в прогнозных оценках при допущении постоянной интенсивности отказов.

Как отражено на типовом графике зависимости интенсивности отказов (рисунок 2.2), в настоящей работе дальнейшие расчеты выполняются для участка нормальной эксплуатации оборудования, то есть без учета повышенной интенсивности ранних отказов и старения на завершающем этапе жизненного цикла, что позволяет использовать приближение постоянной интенсивности отказов и сопоставимо применять показатели *AFR* и *MTBF/MTTF* на годовом интервале. Связь между *AFR* и показателем средней наработки (в часах) при указанном приближении задается выражением (2.15):

$$AFR = 1 - e^{-8766/MTBF}, \quad (2.15)$$

где 8766 – количество часов в году.

В качестве эмпирической базы для оценки *AFR* дисковых накопителей использованы публичные эксплуатационные данные, публикуемые сервисом Backblaze. Согласно указанным данным средний показатель *AFR* по совокупности накопителей по итогам IV квартала 2025 года составил 1,13%. Для используемых при исследовании накопителей Seagate ST8000DM002 вероятность отказа в течение года составляет 1,7% (рисунок 2.2).

Backblaze Hard Drive Failure Rates for Q4 2025

Reporting period October 1, 2025–December 31, 2025 inclusive

Drive models with drive count > 100 as of December 31, 2025 and drive days > 10,000 in Q4 2025.

MFG	Model	Size (TB)	Drive Count	Avg. Age (Months)	Drive Days	Failures	AFR (%)
HGST	HMS5C4040BLE640	4	187	103.3	17,151	1	2.13
HGST	HUH728080ALE600	8	1,073	90.4	95,800	27	10.29
HGST	HUH721212ALE600	12	2,610	72.5	239,745	4	0.61
HGST	HUH721212ALE604	12	13,371	53.5	1,221,499	85	2.54
HGST	HUH721212ALN604	12	9,877	78.0	900,963	76	3.08
Seagate	ST8000DM002	8	8,960	109.9	818,034	38	1.70
Seagate	ST8000NM000A	8	247	35.1	22,700	0	0.00
Seagate	ST8000NM0055	8	13,300	98.6	1,217,828	56	1.68
Seagate	ST10000NM0006	10	1,000	95.2	90,651	13	5.23
Seagate	ST12000NM0007	12	993	72.9	90,539	9	3.63
Seagate	ST12000NM0008	12	10,771	67.7	1,715,709	112	2.38
Seagate	ST12000NM000J	12	1,066	16.4	96,136	1	0.38
Seagate	ST12000NM001G	12	13,248	57.8	1,214,910	31	0.93
Seagate	ST14000NM000J	14	412	14.2	36,688	2	1.99
Seagate	ST14000NM001G	14	10,557	57.3	968,495	34	1.28
Seagate	ST14000NM0138	14	1,264	59.6	115,388	10	3.16
Seagate	ST16000NM000J	16	112	16.6	10,088	1	3.62
Seagate	ST16000NM001G	16	34,410	38.6	3,149,031	35	0.41
Seagate	ST16000NM002J	16	465	35.9	42,714	0	0.00
Seagate	ST24000NM002H	24	8,467	7.6	676,414	65	3.51
Toshiba	MG07ACA14TA	14	37,345	61.2	3,428,534	76	0.81
Toshiba	MG07ACA14TEY	14	994	37.1	89,854	6	2.44
Toshiba	MG08ACA16TA	16	40,115	27.8	3,670,933	68	0.68
Toshiba	MG08ACA16TE	16	6,238	46.3	568,599	9	0.58
Toshiba	MG08ACA16TEY	16	4,938	47.5	449,560	51	4.14
Toshiba	MG09ACA16TE	16	350	6.0	28,207	2	2.59
Toshiba	MG10ACA20TE	20	10,031	10.1	1,656,127	28	0.62
Toshiba	MG11ACA24TE	24	4,806	2.6	349,939	7	0.73
WDC	WUH721414ALE6L4	14	8,650	58.3	794,489	14	0.64
WDC	WUH721816ALE6L0	16	2,994	47.5	274,524	11	1.46
WDC	WUH721816ALE6L4	16	26,715	31.9	2,445,841	45	0.67
WDC	WUH722222ALE6L4	22	44,425	13.6	3,945,245	25	0.23
WDC	WUH722626ALE6L4	26	1,201	2.5	91,690	1	0.40
Totals					30,534,025	943	1.13%

Рисунок 2.2 – Изображение с сайта сервиса backblaze.com [102] со статистикой наработки на отказ для используемых дисков

В расчетах AFR диска интерпретируется как вероятность отказа за год, для согласования с $MTBF$ других компонентов вычисляется эквивалентный $MTBF_{\text{диск}}$ при экспоненциальном приближении.

$$MTBF_{\text{диск}} \approx -\frac{8766}{\ln(1 - 0.017)} \approx 5.11 \cdot 10^5 \text{ ч} (\approx 58.3 \text{ года}).$$

Также при выполнении расчетов следует учитывать особенности работы устройства журналирования. При отказе устройства журнала данные, как правило,

не утрачиваются, однако становится недоступным и соответствующий диск, поскольку запись на него осуществлялась через журнал. Следовательно, при оценке доступности необходимо учитывать не только вероятность отказа диска, но и вероятность отказа устройства журналирования: его отказ приводит к временной недоступности диска на период восстановления работоспособности связки «журнал-диск» (включая замену и/или восстановление журнала и последующую синхронизацию). Журнальное устройство учитывается как отдельный элемент локальной цепочки обслуживания операции записи, если журнал размещается на выделенном SSD- или NVMe-накопителе. Такая конфигурация применяется при хранении основных данных на HDD и позволяет уменьшить задержки журналирования и нагрузку на накопитель данных.

В исследовании надежности SSD в корпоративных системах хранения [103] для показателя *ARR* (от англ. Annual Replacement Rate) отмечено среднее значение порядка 0,22% и разброс по моделям порядка 0,07-1,2%. В отсутствие статистики отказов конкретной модели журнального SSD/NVMe параметр $P_{\text{журн}}$ задается сценарно. В базовом сценарии принимается $P_{\text{журн}} = 0,0036$ (0,36 % за год) как умеренно консервативная оценка относительно среднего *ARR*, при этом для анализа чувствительности рекомендуется использовать диапазон $P_{\text{журн}} \in [0,0007; 0,012]$.

Таким образом, годовая вероятность отказа узла хранения при выполнении операции записи определяется отказами элементов локальной цепочки обслуживания: серверной платформы, дискового накопителя и журнального устройства, если журнал размещается на отдельном накопителе (2.16):

$$P_{\text{узел}} = 1 - (1 - P_{\text{платф}}) (1 - P_{\text{диск}}) (1 - P_{\text{журн}}), \quad (2.16)$$

где $P_{\text{платф}}$ вычисляется по таблице 4, $P_{\text{диск}}$ задается по *AFR* (рисунок 2.2), а $P_{\text{журн}}$ учитывается только для конфигурации с отдельным журнальным устройством.

В отсутствие статистики восстановления для конкретной инфраструктуры $MTTR$ также задается сценарно. Данный параметр используется для перехода от вероятности отказа узла на заданном интервале к коэффициенту готовности узла $A_{\text{узл}}$, то есть к доле времени, в течение которого узел способен выполнять операцию записи. Расчет выполняется последовательно, сначала по значениям компонентов $MTBF$ определяется вероятность отказа отдельных элементов на годовом интервале, затем с учетом структуры узла вычисляется вероятность отказа серверной платформы. Далее учитываются отказ дискового накопителя и журнального устройства, что позволяет получить вероятность недоступности узла хранения для выполнения операции записи. После этого с учетом заданного значения $MTTR$ определяется коэффициент готовности одного узла $A_{\text{узл}}$.

Полученное значение $A_{\text{узл}}$ далее используется как исходный параметр для расчета коэффициента готовности выполнения операции записи $A(R, W)$. Если операция считается завершенной после получения W подтверждений из R целевых реплик, то $A(R, W)$ определяется как вероятность того, что из R выбранных узлов доступно не менее W , необходимых для подтверждения операции записи. Именно эта величина далее представлена в таблице 2.2 и на рисунке 2.3 для различных значений $MTTR$, R и W . При расчете готовности выполнения операции записи доступность требуемого числа реплик не исчерпывает всю цепочку обслуживания. Операция записи также зависит от общих элементов инфраструктуры: координатора записи, подсистемы метаданных, механизмов выбора размещения реплик и сетевой связности, необходимой для передачи данных и получения подтверждений. Недоступность этих элементов может привести к невозможности завершения записи даже при доступности требуемого числа узлов хранения.

Поэтому при интерпретации расчетов дополнительно учитывается готовность общей части цепочки обслуживания операции записи. В настоящей работе для нее принята модельная оценка $A_{\text{общ}} = 0,99999$. Такой учет позволяет получить более консервативную оценку готовности функции записи и избежать

завышенной интерпретации результата, рассчитанного только по условию доступности реплик.

Таблица 2.3 – Фрагмент таблицы с результатами расчета коэффициентов готовности узла и операции записи в разных режимах

$MTTR$, мин	$A_{\text{узел}}$	Недоступность одного узла, мин/год	$A_{\text{запись}}^* (3,3)$	Недоступность записи при $W = 3$, мин/год	$A_{\text{запись}}^* (3,2)$	Недоступность записи при $W = 2$, мин/год
24	0,999991343	4,553	0,999964030	18,919	0,999990000	5,260
25	0,999990982	4,743	0,999962947	19,488	0,999990000	5,260
26	0,999990622	4,933	0,999961865	20,057	0,999990000	5,260
...	-	-	-	-	-	-
30	0,999989179	5,692	0,999957537	22,334	0,999990000	5,260
...	-	-	-	-	-	-
48	0,999982686	9,106	0,999938060	32,578	0,999989999	5,260
49	0,999982325	9,296	0,999936978	33,147	0,999989999	5,260
50	0,999981965	9,486	0,999935896	33,716	0,999989999	5,260
...	-	-	-	-	-	-
60	0,999978358	11,383	0,999925075	39,408	0,999989999	5,260
...	-	-	-	-	-	-
70	0,999974751	13,280	0,999914255	45,098	0,999989998	5,261
71	0,999974390	13,470	0,999913173	45,667	0,999989998	5,261
72	0,999974030	13,659	0,999912092	46,236	0,999989998	5,261
...	-	-	-	-	-	-
90	0,999967536	17,075	0,999892613	56,481	0,999989998	5,261

При интерпретации результатов следует учитывать принятую точность округления. Например, в пороговом режиме подтверждения (3,2) изменение $MTTR$ в диапазоне от 24 до 30 мин практически не влияет на итоговую ожидаемую недоступность функции записи. Так, при $MTTR = 24$ мин расчетное значение недоступности составляет 5,259718 мин/год, а при $MTTR = 30$ мин – 5,259785 мин/год. Разность между указанными значениями равна 0,000067 мин/год, (приблизительно 0,004 с/год), поэтому при округлении оба значения принимают вид 5,260 мин/год, а коэффициент готовности операции записи при округлении отображается как 0,999990000. Такое совпадение округленных значений не означает отсутствия влияния $MTTR$, а показывает, что в данном режиме вклад

недоступности реплик существенно меньше вклада общей части цепочки обслуживания операции записи.

Из таблицы 2.3 видно, что учет общей части цепочки обслуживания снижает расчетную готовность функции записи по сравнению с оценкой, основанной только на доступности реплик. В пороговом режиме $W = 2$ вероятность невыполнения репликационного условия мала, поэтому итоговая недоступность функции записи в большей степени определяется готовностью общих элементов цепочки обслуживания. Полученные значения являются расчетной модельной оценкой при принятых допущениях о независимости отказов и восстановлении компонентов и не рассматриваются как заявленный уровень *SLA* (от англ. Service Level Agreement, соглашение об уровне обслуживания) телекоммуникационной системы в целом.

Таблица 2.4 – Фрагмент таблицы с результатами расчета коэффициентов готовности узла и операции записи для MTTR 30 мин

R	$A^{\text{запись}*}(R, W)$	$A^{\text{запись}*}(R, R)$
1	0,999979179	0,999979179
2	0,999968358	0,999968358
3	0,999990000	0,999957538
4	0,999989999	0,999946717
5	0,999990000	0,999935897
6	0,999990000	0,999925076
7	0,999990000	0,999914256

В таблице 2.4 сопоставлены два режима подтверждения записи: пороговый режим с минимальным порогом и строгий режим $W = R$. Значения приведены с учетом общей части цепочки обслуживания операции записи, поэтому характеризуют не только выполнение условия доступности требуемого числа реплик, но и доступность общих инфраструктурных элементов, необходимых для завершения записи. В пороговом режиме подтверждения, начиная с $R = 3$, операция записи может быть завершена при недоступности одной из реплик, поэтому ожидаемая недоступность функции записи существенно ниже, чем в строгом режиме $W = R$. При дальнейшем увеличении R выигрыш по готовности становится

ограниченным, поскольку итоговое значение определяется не только доступностью реплик, но и готовностью общей части цепочки обслуживания.

С учетом требований, формируемых для перспективных телекоммуникационных систем IMT-2030/6G, показатели надежности и готовности целесообразно рассматривать не только на фиксированном интервале наблюдения, но и применительно к полному периоду функционирования сервиса. Такой подход согласуется с положениями ITU-R M.2160 [9], в которых надежность, безопасность и устойчивость отнесены к значимым характеристикам IMT-2030, а также с отраслевыми материалами NGMN по требованиям к 6G [104].

В материалах проекта Неха-X-II [105], посвященных управлению и обеспечению *SLA* в сетях 6G, предусматривается контроль соответствия сервисов установленным показателям на протяжении всего срока их функционирования. В настоящем исследовании годовой интервал принят в качестве базового расчетного интервала, что обусловлено формой представления исходных статистических характеристик отказов и принятым допущением о постоянной интенсивности отказов на участке нормальной эксплуатации. Поэтому полученное значение ожидаемой недоступности 5,260 мин/год следует интерпретировать как нормированную расчетную оценку для исследуемой конфигурации, используемую для сопоставления режимов репликации и подтверждения записи. При предъявлении более жесткого ограничения на недоступность за полный срок функционирования сервиса исследуемая конфигурация должна дополнительно проверяться относительно соответствующего целевого значения готовности.

Полученные результаты также показывают, что при повышении готовности репликационной части дальнейшее снижение недоступности функции записи определяется прежде всего готовностью общей части цепочки обслуживания. В этих условиях общая инфраструктурная часть становится лимитирующей по показателю готовности, поэтому переход к более жестким требованиям на полном сроке эксплуатации требует комплексного повышения готовности инфраструктуры, а не только увеличения избыточности подсистемы хранения. В дальнейших исследованиях разработанный метод может быть расширен за счет

учета резервирования общих элементов цепочки обслуживания, разнесения реплик и маршрутов передачи по независимым доменам отказа, адаптивного выбора параметров репликации и режима подтверждения, а также применения иных механизмов обеспечения избыточности. Для оценки готовности на полном жизненном цикле дополнительно целесообразно учитывать коррелированные отказы и изменение характеристик отказов и восстановления во времени.

Также следует отметить, что если $MTTR$ характеризует продолжительность одного восстановления, то простой за год отражает суммарное ожидаемое время недоступности узла на годовом интервале и определяется не только длительностью восстановления, но и частотой возникновения отказов. Следовательно, $MTTR$ выступает исходным параметром расчета коэффициента готовности, тогда как годовой простой является производной характеристикой доступности. Ожидаемый простой узла характеризует недоступность одного узла хранения как элемента системы. Ожидаемый простой кластера характеризует недоступность функции записи на кластерном уровне, то есть ситуации, когда число доступных реплик становится меньше порога подтверждения W . Поэтому кластерный простой при кворумной записи, как правило, существенно меньше простоя отдельного узла.

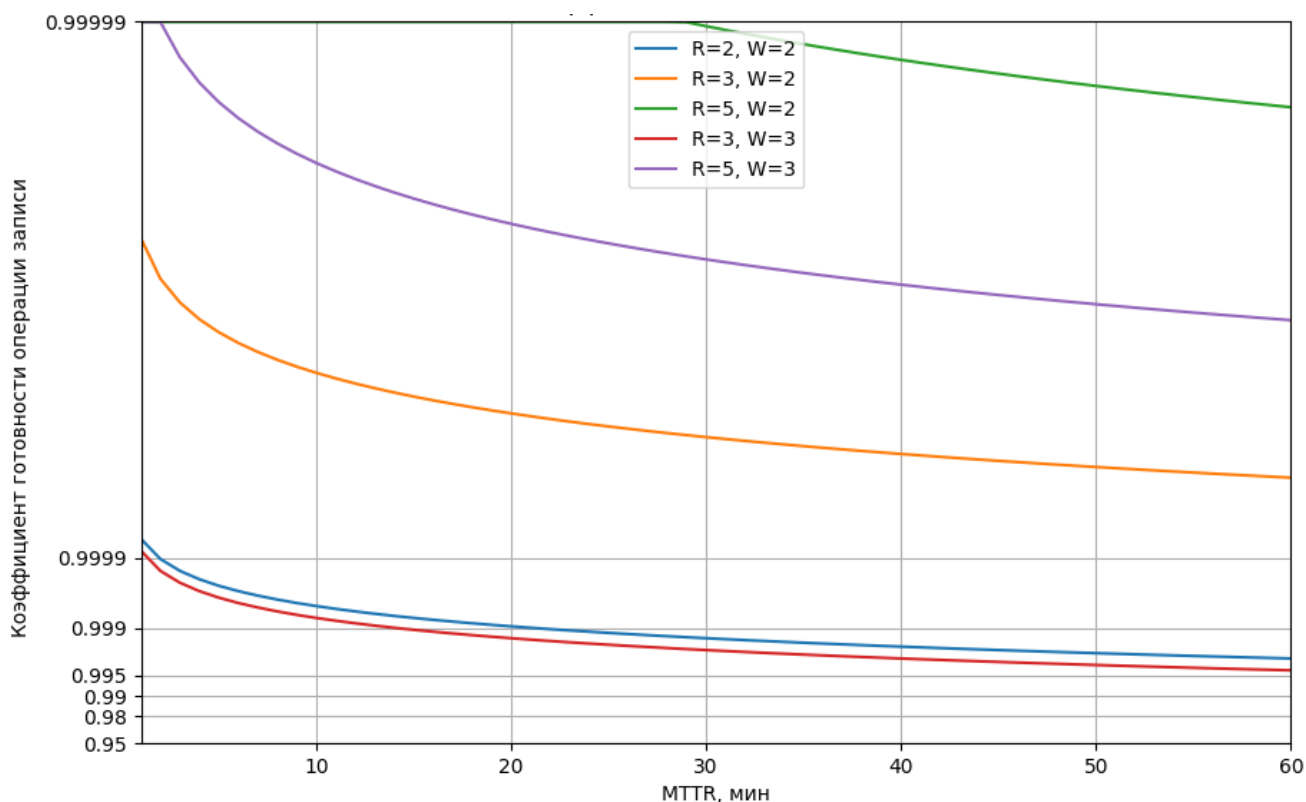


Рисунок 2.3 – Зависимость коэффициента готовности выполнения операции записи от среднего времени восстановления

Следует подчеркнуть, что близость $A^{\text{запись}}$ к 1 в режиме $R = 3, W = 2$ обусловлена кворумной семантикой подтверждения: операция записи завершается при получении W подтверждений из R , т.е. при доступности не менее двух реплик, что не означает «абсолютную готовность» системы. При этом для сопоставления необходимо отметить, что для компонента $A^{\text{хранения}}$ промышленным ориентиром целевого уровня сохранности данных является значение порядка 0,999999999999 (или также используется обозначение «11 девяток»), декларируемое в объектных хранилищах облачного класса [106].

Разработанный метод оценки готовности позволяет рассчитывать коэффициент готовности выполнения операции записи при различных значениях фактора репликации и порога подтверждения. Анализ результатов показывает, что при факторе репликации $R=1$ (отсутствие репликации) коэффициент готовности принимает низкие значения, что обуславливает неприемлемые риски с точки зрения сохранности и доступности данных. При увеличении фактора репликации

до $R=2$ коэффициент готовности возрастает, однако прирост оказывается ограниченным вследствие особенностей функционирования механизма репликации, кроме того, репликация с $R=2$ сопровождается повышенными рисками рассогласования (неконсистентности) реплик при отказах и нарушениях связности. Начиная с $R=3$ коэффициент готовности приближается к 1, обеспечивая приемлемый уровень доступности. Значения фактора репликации $R=3$ достаточно для обеспечения надежного хранения данных, при этом отсутствует необходимость в дополнительном резервировании дискового пространства, характерном для конфигураций с $R>3$. Вместе с тем, с учетом особенностей функционирования распределенной программно-определяемой СХД увеличение фактора репликации сверх трех приводит к росту задержек операций чтения и записи, что нежелательно с точки зрения требований к высокой производительности и поддержанию высокоскоростной передачи и обработки данных. Выбор параметров R и W может уточняться в зависимости от класса обслуживаемых данных и требований к задержке, сохранности и допустимости потери отдельных записей, соответствующая классификация трафика и данных предложена в [93].

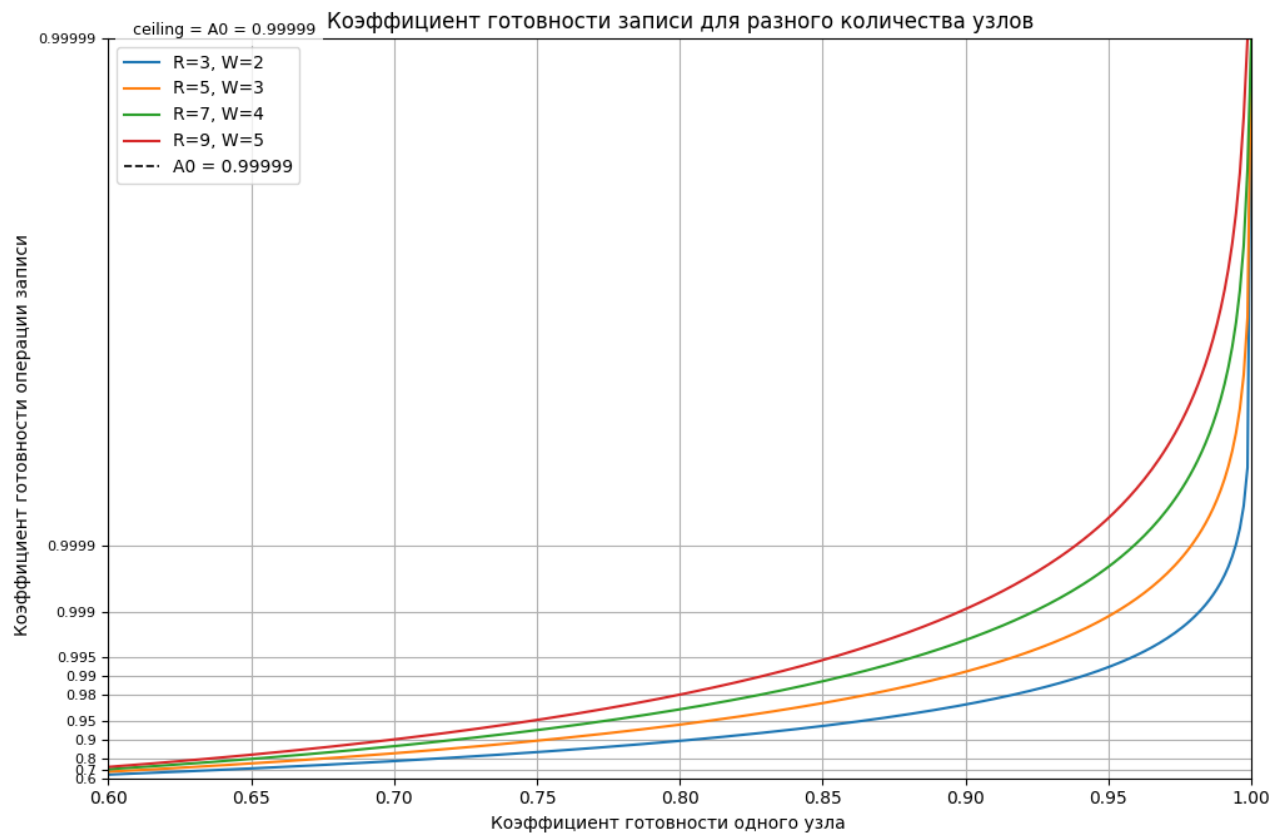
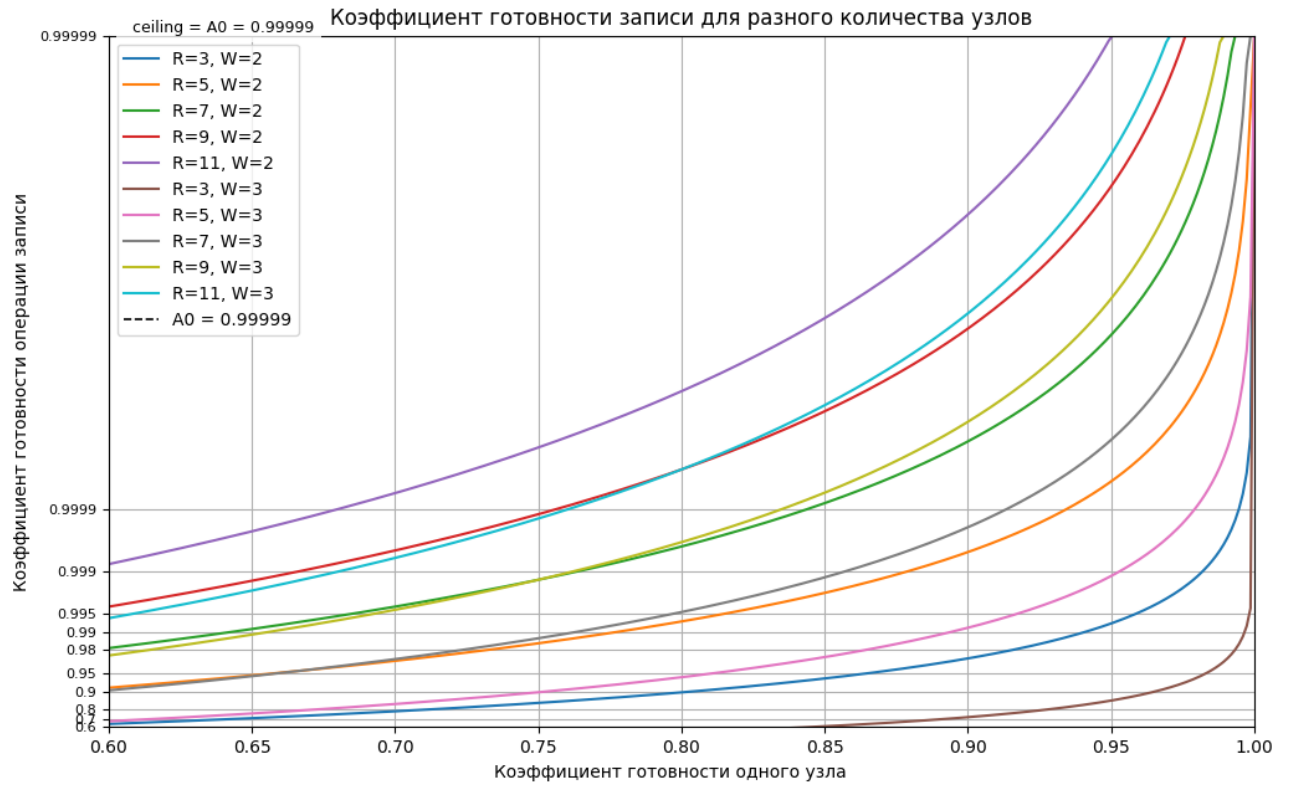


Рисунок 2.4 – Зависимость коэффициента готовности выполнения операции записи от параметров репликации и режима подтверждения

В настоящей главе расчет коэффициента готовности используется как критерий допустимости управляющих воздействий: стратегия распределения пользовательских операций и регулирования служебной нагрузки считается допустимой, если при заданных параметрах репликации и подтверждения сохраняется возможность завершения операции записи. В этом смысле показатель A связывает метод готовности с моделью управления информационными потоками.

2.2 Выбор показателей производительности

Для количественной проверки утверждения о повышении производительности системы необходимо выбрать показатели, которые:

- напрямую отражают качество обслуживания операций доступа к данным в распределенной подсистеме хранения,
- чувствительны к влиянию нагрузки от потока служебных операций (репликации, согласования, восстановления избыточности и перераспределения данных),
- сопоставимы с практикой экспериментальной оценки распределенных систем туманного уровня.

Подход к выбору показателей на основе их привязки к целям управления и типовые наборы метрик для облачных, туманных и периферийных вычислений представлены в обзорной работе, систематизирующей показатели производительности и методы их применения [107,108].

В качестве основного показателя производительности целесообразно использовать пропускную способность операций записи (2.17)

$$P_{\text{зап}} = \frac{N_{\text{заверш.записей}}}{T}, \quad (2.17)$$

где $N_{\text{заверш.записей}}$ – число успешно завершенных операций записи за интервал наблюдения T . В дальнейшем, с учетом дискретного представления времени

(см. ниже), интервал наблюдения T в (2.17) трактуется как число шагов моделирования, за которое ведется подсчет $N_{\text{заверш.записей}}$, при необходимости переход к физическим единицам времени выполняется с учетом длительности одного шага дискретизации. В экспериментальной части показатель $P_{\text{зап}}$ конкретизируется через допустимую интенсивность входного потока операций записи $\lambda_{\text{доп}}$. При устойчивом режиме обслуживания интенсивность завершенных операций соответствует интенсивности поступления заявок, поэтому рост $\lambda_{\text{доп}}$ отражает увеличение предельной производительности подсистемы хранения, позволяет учитывать не только число обслуживаемых операций, но и выполнение ограничений по времени завершения операции записи, задержкам верхних процентилях и готовности сервиса записи.

Данный показатель непосредственно характеризует способность подсистемы хранения обслуживать пользовательский поток запросов и традиционно используется при исследовании распределенных хранилищ [33]. Практика оценки производительности и масштабируемости СХД, построенного поверх распределенного объектного хранилища, опирается именно на измерение пропускной способности и задержек операций [34,109].

Процент повышения производительности по основному показателю определяется как (2.18)

$$E = \frac{\lambda_{\text{нов}} - \lambda_{\text{баз}}}{\lambda_{\text{баз}}} \times 100\%, \quad (2.18)$$

где $\lambda_{\text{баз}}$ – значение показателя для сравниваемого способа управления (базового варианта), $\lambda_{\text{нов}}$ – значение для предлагаемой модели.

Одной пропускной способности недостаточно для корректной интерпретации результатов, поскольку увеличение пропускной способности может сопровождаться ростом задержек, поэтому целесообразно фиксировать задержки выполнения операций записи на высоких процентилях (например, 95-м и 99-м). Контроль распределения задержек в крупномасштабных распределенных системах

обусловлен тем, что при параллельной обработке запросов итоговое время выполнения часто определяется неблагоприятными реализациями задержек отдельных этапов и взаимодействий, а влияние редких, но существенных задержек становится заметным на уровне качества обслуживания. Данная особенность обсуждается в работах, посвященных влиянию высоких процентилей распределения задержек на функционирование масштабируемых сервисов [66].

Для показателей задержки, где «меньше – лучше», улучшение вычисляется как (2.19)

$$E = \frac{L_{\text{баз}} - L_{\text{нов}}}{L_{\text{баз}}} \times 100\%, \quad (2.19)$$

где L – выбранный процентиль задержки операции записи (например, 99-й).

Связь роста задержек с ростом очередей в системе может быть дополнительно обоснована законом Литтла, связывающим среднее число заявок в системе, интенсивность поступления и среднее время пребывания, что дает теоретическое объяснение того, почему управление нагрузкой через очереди и распределение обслуживающих ресурсов узла приводит к изменению задержек.

Поскольку межузловой обмен и служебные операции влияют на время обслуживания и использование ресурсов подсистемы хранения, рост $P_{\text{зап}}$ следует интерпретировать с учетом показателя, отражающего служебную составляющую нагрузки:

- интенсивность межузлового обмена (объем/скорость служебного сетевого трафика) за интервал T ,
- доля служебных операций (например, отношение объема служебных операций к объему пользовательских данных, либо доля времени, затрачиваемая на восстановление избыточности и перераспределение данных).

Такие показатели прямо соответствуют логике выбора метрик «в соответствии с целью управления и архитектурными особенностями» для туманного уровня и рекомендуются в систематизациях показателей для облачных, туманных и периферийных сред [107].

Если предлагаемая модель включает регулирование восстановления избыточности и перераспределения данных, необходимо дополнительно фиксировать время восстановления избыточности (или зависимую характеристику скорости восстановления), чтобы исключить ситуацию, когда выигрыш по пользовательским метрикам достигается недопустимым ухудшением восстановительных процессов. Практическая значимость такого контроля согласуется с работами по оценке поведения распределенных хранилищ под нагрузкой и при масштабировании/стрессовых режимах, где анализируется снижение характеристик и влияние служебных операций на производительность [110].

В рамках экспериментальной проверки предлагается использовать:

- основной показатель: допустимая интенсивность входного потока операций записи, определяемая как максимальное значение интенсивности поступления заявок, при котором сохраняется заданный критерий допустимости по времени завершения операции записи,
- обязательные подтверждающие: 95-й и 99-й процентиля задержки операций записи,
- показатели нагрузки от потока служебных операций: интенсивность межузлового обмена и/или доля служебных операций,
- контроль корректности: время восстановления избыточности (при наличии соответствующего механизма управления).

Использование допустимой интенсивности, задержек и профиля нагрузки как взаимосвязанных характеристик соответствует практике экспериментальной оценки распределенных систем хранения и облачных систем обслуживания данных [111,112], где нагрузочный эксперимент строится через задание профилей операций и последующее сопоставление задержки и достигнутой производительности.

Сбор и обработка указанных метрик могут быть реализованы с использованием разработанного программного средства, зарегистрированного

в качестве программы для ЭВМ [113]. В свою очередь, сбор результатов нагрузочных испытаний и сопоставимых метрик компонентов распределенной подсистемы хранения реализуется с использованием программного средства [114], что обеспечивает воспроизводимость профилей нагрузки и единый формат представления результатов для последующего сравнения базового и предлагаемого способов управления.

Такой набор обеспечивает интерпретацию результатов: рост пропускной способности подтверждается отсутствием ухудшения высоких перцентилей задержки и объясняется изменением соотношения пользовательской и нагрузки от потока служебных операций, что соответствует принятой концепции исследования.

2.3 Выбор ключевых метрик

В настоящем разделе формализуется функционирование распределенной подсистемы хранения туманного уровня как распределенной вычислительной сети, обслуживающей поток пользовательских операций доступа к данным и порождающей служебные операции, связанные с репликацией, согласованием, восстановлением избыточности и перераспределением данных. Необходимость разделения пользовательской и служебной составляющих нагрузки и фиксации параметров испытаний обусловлена особенностями моделирования нагрузки распределенных систем хранения и рисками искажения результатов при неполном учете архитектурных факторов. Для описания динамики нагрузки применяется аппарат теории очередей и стохастической оптимизации сетей как инструмент для связи управляемых режимов с QoS и ограничениями межузлового обмена, позволяющий рассматривать управляемые воздействия с пропускной способностью и задержками обслуживания заявок в системе [115].

Рассмотрим множество узлов хранения $\mathcal{N} = \{1, \dots, N\}$, объединенных сетью передачи данных. Сетевую инфраструктуру представим ориентированным графом $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, где $\mathcal{V}$ – множество вершин (узлы хранения и коммуникационные узлы),

$\mathcal{E}$ – множество каналов связи. Для каждого канала $e \in \mathcal{E}$ задана пропускная способность N_e и (при необходимости) параметр задержки распространения τ_e .

Функционирование подсистемы хранения рассматривается в дискретном времени. Обозначим через $k = 0, 1, 2, \dots$ номер шага управления, на котором фиксируется состояние узлов хранения и формируется управляющее воздействие. Физический момент времени, соответствующий шагу k , обозначим через t_k , а длительность шага дискретизации – через Δt . На шаге k в систему поступает случайное число пользовательских запросов $C(k)$, которые могут относиться к операциям записи. В рассматриваемой архитектуре операции записи с репликацией и согласованием являются основным источником дополнительной служебной нагрузки, поэтому дальнейшее моделирование сосредоточено на операциях записи, тогда как модель чтения строится по аналогичным принципам.

Для каждого узла хранения $i \in \mathcal{N}$ вводятся две компоненты нагрузки:

- $Q_i(k)$ – длина пользовательской очереди, определяемая как число пользовательских операций записи, ожидающих обслуживания на i -м узле на шаге управления k ,
- $B_i(k)$ – длина служебной очереди, определяемая как число служебных операций репликации, согласования, восстановления или перераспределения данных, ожидающих обслуживания на i -м узле на шаге управления k .

Такое разбиение позволяет различать пользовательскую и служебную составляющие нагрузки – пользовательские операции определяют характеристики обслуживания, а служебные операции обеспечивают избыточность и согласованность данных, используя при этом те же вычислительные, дисковые и коммуникационные ресурсы узла. Управление нагрузкой реализуется через два класса воздействий, согласующихся с архитектурой распределенного хранения:

- 1) распределение пользовательских запросов по узлам хранения. На каждом шаге k принимается решение о направлении поступивших запросов в очереди конкретных узлов. Обозначим через $c_i(k)$ число пользовательских операций, направленных на узел i на шаге k , при этом $\sum_{i \in \mathcal{N}} c_i(k) = C(k)$,

2) регулирование интенсивности поступления служебных операций. Для каждого узла i вводится управляющий параметр $b_i(k) \in [0,1]$, определяющий долю обслуживающего ресурса узла, выделяемую под служебные операции. Тогда доступная интенсивность обслуживания пользовательских заявок уменьшается пропорционально $b_i(k)$, а интенсивность обслуживания служебных операций увеличивается.

Такое управление необходимо, поскольку служебные операции восстановления и перераспределения, с одной стороны, критичны для поддержания избыточности, а с другой – способны существенно снижать пропускную способность пользовательских операций за счет конкуренции за обслуживающий ресурс узла и межузлового обмена. Подобные эффекты отражены в системных работах по распределенному хранению, механизмам размещения и перераспределения данных [116].

Условие (2.3) включается в систему ограничений и далее рассматривается как ограничение на множество допустимых управлений $\{c_i(k), b_i(k)\}$. Далее A рассматривается не как критерий оптимизации, а как ограничение корректности, исключающее управления, при которых прирост производительности достигается за счет недопустимого снижения доступности и предсказуемости результата операции.

2.3.1 Модель обслуживания и динамика очередей

Пусть μ_i – нормированная производительность узла i , определяемая доступным обслуживающим ресурсом и выражающая объем обслуживаемой нагрузки за один дискретный шаг моделирования. Тогда на шаге k интенсивности обслуживания распределяются между пользовательской и служебной составляющими (2.20):

$$\mu_i^{(u)}(k) = \mu_i (1 - b_i(k)), \quad \mu_i^{(s)}(k) = \mu_i b_i(k), \quad (2.20)$$

где $\mu_i^{(u)}(k)$ – интенсивность пользовательских операций, $\mu_i^{(s)}(k)$ – интенсивность для служебных операций, здесь и далее $i \in \mathcal{N}$, $k = 0, 1, 2, \dots$.

Вводимая очередь рассматривается как прикладная модель перегрузки узла хранения в туманной инфокоммуникационной инфраструктуре: рост очередей отражает ухудшение *QoS* операций чтения/записи и обусловлен конкуренцией за вычислительные, дисковые и коммуникационные ресурсы (в том числе из-за служебных операций репликации и восстановления).

Обозначим через $S_i^{(u)}(k)$ и $S_i^{(s)}(k)$ фактически обслуженные объемы работ (завершенные операции) на шаге k , причем (2.21)

$$0 \leq S_i^{(u)}(k) \leq \mu_i^{(u)}(k), \quad 0 \leq S_i^{(s)}(k) \leq \mu_i^{(s)}(k). \quad (2.21)$$

Тогда динамика очередей задается рекуррентными соотношениями (2.22):

$$\begin{aligned} Q_i(t+1) &= \max\{Q_i(k) - S_i^{(u)}(k), 0\} + c_i(k), \\ B_i(t+1) &= \max\{B_i(k) - S_i^{(s)}(k), 0\} + g_i(k), \end{aligned} \quad (2.22)$$

где $g_i(k)$ – приток служебных операций на узле i на шаге k .

Величина $g_i(k)$ зависит от:

- отказов и деградаций, требующих восстановления избыточности,
- политики размещения и перераспределения данных,
- выбранного режима репликации и согласования, и, в общем случае, может моделироваться как стохастический процесс, коррелированный с пользовательской нагрузкой.

Связь между средними значениями очередей и средним временем пребывания операций в системе обосновывается законом Литтла, что позволяет использовать $Q_i(k)$ (или $\sum_i Q_i(k)$) как показатель задержек и формализовать цель управления в терминах качества обслуживания.

2.3.2 Сетевые ограничения и межузловой обмен

Операции записи с репликацией порождают межузловой обмен: передачу пользовательских данных на выбранные узлы хранения, а также служебные сообщения согласования и подтверждения на шаге k . В рамках настоящей работы межузловой обмен трактуется как служебный сетевой трафик телекоммуникационной инфраструктуры туманного уровня, непосредственно влияющий на задержки и пропускную способность операций хранения [117]. Поэтому вводимое ограничение по каналу отражает эксплуатационное ограничение сети передачи данных распределенной подсистемы хранения и используется для разработки правил управления.

Для учета сетевых ограничений введем функцию сетевой нагрузки $\Phi_e(k)$ – объем данных, переданный по каналу e на шаге k . Тогда выполняется ограничение (2.23):

$$\Phi_e(k) \leq N_e, \quad \forall e \in \mathcal{E}. \quad (2.23)$$

Функция $\Phi_e(k)$ зависит от маршрутизации пользовательских операций и от интенсивности служебных операций. При необходимости может быть введена агрегированная характеристика межузлового обмена $Net(k)$, например суммарный объем переданных данных или взвешенная сумма загрузки каналов, которая далее используется в целевой функции или ограничениях задачи управления. Использование ограниченных сетей очередей и подходов к управлению, ориентированных на стабилизацию очередей при ограничениях обслуживающих ресурсов, соответствует классическим результатам для ограниченных сетей очередей и политик, максимизирующих область устойчивости [118].

2.3.3 Вывод по разделу

Построенная модель описывает систему как совокупность узлов с двумя конкурирующими компонентами нагрузки (пользовательской и служебной),

связанными общими обслуживающими ресурсами и ограничениями сети, что создает формальную основу для постановки задачи управления нагрузкой как задачи стохастической оптимизации: требуется выбрать правила распределения пользовательских запросов и регулирования служебных операций так, чтобы повысить пропускную способность и улучшить характеристики задержки при выполнении ограничений по готовности и коммуникационными ресурсам. Теоретический аппарат стохастической оптимизации сетей на основе метода Ляпунова обеспечивает корректную основу для синтеза таких правил управления в условиях случайных поступлений и изменяющихся состояний системы. Указанный теоретический аппарат используется как основа для построения правила управления распределенной подсистемой хранения, размещаемой на уровне туманных вычислений, с учетом межузлового обмена и требований *QoS*. Научный вклад раздела связан с прикладной адаптацией модели к задаче управления подсистемой хранения данных и ее применением в рамках рассматриваемой системы, при этом вопросы развития общей теории стохастической оптимизации выходят за рамки настоящего исследования.

2.4 Управляющие воздействия и допустимая область решений

В соответствии с концепцией исследования (глава 1), повышение производительности подсистемы хранения туманного уровня достигается за счет управления совокупной нагрузкой, включающей пользовательские операции доступа к данным и служебные операции (репликация, согласование, восстановление избыточности, перераспределение данных).

В рамках математической модели управление формализуется как выбор набора управляющих воздействий, определяющих, куда направлять пользовательские операции и какую долю обслуживающих ресурсов выделять служебным операциям при выполнении системных ограничений по ресурсам и коэффициенту готовности. Для формализации данного подхода могут использоваться методы стохастического управления и оптимизации. В настоящей

работе они применяются к распределенной подсистеме хранения, размещаемой на уровне туманных вычислений, с учетом межузлового обмена, ограничений пропускной способности каналов и требований QoS к операциям хранения [115,118].

2.4.1 Состав управляющих воздействий

Пусть на шаге k формируется управляющее решение (2.24)

$$u(k) = \{ c_i(k), b_i(k), \pi(k) \}, \quad (2.24)$$

где компоненты имеют следующий смысл.

В первую очередь, необходимо рассмотреть распределение пользовательских операций по узлам хранения. Для поступившего на шаге k объема пользовательских операций $C(k)$ выбираются значения $c_i(k) \geq 0$, определяющие число операций, направляемых на узел i (2.25):

$$\sum_{i \in \mathcal{N}} c_i(k) = C(k). \quad (2.25)$$

Данный класс воздействий соответствует задачам балансировки нагрузки и диспетчеризации в распределенных системах, где производительность определяется тем, насколько равномерно и своевременно запросы распределяются между обслуживающими узлами. Эффективность решений, ориентированных на сравнение очередей на ограниченном числе кандидатов (например, выбор из нескольких узлов), подтверждена в классических работах по вероятностной балансировке нагрузки [80].

Для последующего сравнения используются референсные алгоритмы распределения запросов, характеризующиеся ограниченным объемом служебного обмена. К таким подходам относится схема «выбор свободной очереди», исследованная применительно к масштабируемым сервисам [64].

Далее необходимо рассмотреть регулирование интенсивности поступления служебных операций. Для каждого узла i вводится параметр $b_i(k) \in [0,1]$, задающий долю доступной обслуживающей способности узла, выделяемую служебным операциям. В модели обслуживания (раздел 2.3) это приводит к разбиению производительности узла μ_i на пользовательскую и служебную составляющие (2.20).

Практическая необходимость данного воздействия обусловлена тем, что восстановление и перераспределение данных (в том числе процессы «заполнения» и «восстановления» после изменений состава устройств) способны существенно снижать характеристики пользовательских операций ввода-вывода.

В технической документации и материалах эксплуатации распределенных хранилищ отдельно подчеркивается, что миграция данных для восстановления баланса может заметно ухудшать рабочую производительность, а регулирование параметров служебных операций позволяет задавать компромисс между скоростью восстановления и реактивностью к пользовательской нагрузке [119].

Дополнительно наличие выделенного служебного межузлового сетевого обмена (репликация, восстановление, перераспределение) фиксируется в руководствах по проектированию инфраструктуры, что подтверждает необходимость учитывать и управлять этой составляющей нагрузки [120].

Также необходимо принимать во внимание политику размещения и переразмещения данных (долгосрочное воздействие, при необходимости). Если предполагается управление не только распределением операций, но и структурой размещения реплик, вводится воздействие $\pi(k)$, описывающее выбор схемы размещения/переразмещения (например, веса устройств, правила выбора доменов отказа, состав наборов реплик для групп объектов).

В современных распределенных хранилищах размещение реплик реализуется детерминированными функциями распределения, обеспечивающими балансировку и минимизацию избыточных перемещений данных при изменении состава множества узлов распределенной подсистемы хранения. Указанные

принципы подробно рассматриваются в работах, посвященных функциям размещения реплик и объектным хранилищам [121].

В рамках данной работы $\pi(k)$ целесообразно рассматривать как управляющее воздействие, изменяемое на более длительном временном горизонте – от минут до часов, тогда как $c_i(k)$ и $b_i(k)$ относятся к оперативному управлению и могут изменяться с интервалом порядка секунд и менее.

2.4.2 Ограничения на управляющие воздействия

Допустимая область решений формируется системой ограничений четырех типов.

Балансовые и неотрицательные ограничения

При необходимости $c_i(k)$ трактуются как целочисленные величины (число операций), однако для аналитического исследования допустима непрерывная аппроксимация (интенсивности потоков) (2.26):

$$c_i(k) \geq 0, \quad \sum_{i \in \mathcal{N}} c_i(k) = C(k), \quad 0 \leq b_i(k) \leq 1 \quad (2.26)$$

Ограничения вычислительных ресурсов узлов

Фактическая интенсивность обслуживания не может превышать доступную производительность системы (2.27):

$$S_i^{(u)}(k) \leq \mu_i(1 - b_i(k)), \quad S_i^{(s)}(k) \leq \mu_i b_i(k). \quad (2.27)$$

Данные ограничения обеспечивают физическую реализуемость управления и связывают решение $b_i(k)$ с конкуренцией пользовательской и служебной операциями.

Сетевые ограничения на межузловой обмен

Поскольку запись с репликацией и процедуры восстановления генерируют трафик в межузловой сети подсистемы хранения, управляющие решения должны удовлетворять ограничениям пропускной способности каналов (2.23). Указанные ограничения интерпретируются как ограничения на управление информационными потоками (пользовательскими и служебными) в телекоммуникационной сети межузлового обмена подсистемы хранения и непосредственно определяют достижимые значения QoS .

Учет сетевых ограничений при управлении нагрузкой соответствует классическим моделям ограниченных сетей массового обслуживания. В таких моделях правила планирования выбираются на основе состояния очередей и обеспечивают устойчивость системы обслуживания при допустимых значениях интенсивности поступления заявок [118].

Ограничение по коэффициенту готовности

Ключевым ограничением, согласующим управление нагрузкой с требованиями к доступности выполнения операции записи, является условие (2.3). Практически это означает, что множество допустимых решений $\{c_i(k), b_i(k), \pi(k)\}$ должно исключать такие режимы распределения и служебных операций, которые приводят к недопустимому снижению готовности (например, вследствие перегрузки критических узлов, деградации сетевого обмена для подтверждения репликации, либо чрезмерного вытеснения пользовательских операций служебными). Тем самым ограничение по коэффициенту готовности A задает границы допустимых режимов работы подсистемы хранения с учетом сетевой связности и межузлового обмена, а для телекоммуникационного сервиса, использующего данную подсистему, – отражает требование к доступности и предсказуемости завершения операций записи и используется при выборе управляющих решений по распределению нагрузки.

2.4.3 Формирование допустимой области решений

Обозначим через $\mathcal{U}(k)$ множество допустимых управляющих воздействий на шаге k . Тогда (2.28)

$$u(k) \in \mathcal{U}(k), \quad (2.28)$$

где $\mathcal{U}(k)$ определяется пересечением:

- балансовых и интервальных ограничений на $\{c_i(k), b_i(k)\}$,
- ресурсных ограничений узлов на обслуживание,
- сетевых ограничений на межузловой обмен,
- ограничения $A \geq A_{\min}$,
- технологических ограничений (например, допустимых доменов отказа для размещения реплик, правил согласования и подтверждения записи), заданных архитектурой подсистемы хранения.

Такое представление позволяет далее (в разделе 2.5) сформулировать задачу управления как задачу оптимизации при ограничениях, а в разделе 2.6 – построить алгоритм, который на каждом шаге выбирает $u(k)$ на основе наблюдаемого состояния системы, оставаясь внутри $\mathcal{U}(k)$. Теоретический аппарат оптимизации динамических сетей дает формальные основания для выбора управлений по текущим очередям с обеспечением ограничений по средним значениям показателей [115]. Следует подчеркнуть, что указанный аппарат применяется в работе как прикладной методический инструмент для выбора управлений в пределах допустимой области, задаваемой телекоммуникационными ограничениями межузловой сети подсистемы хранения (пропускная способность межузловых соединений, служебные операции репликации/восстановления). Научный результат данного раздела состоит в формализации допустимой области решений для подсистемы хранения туманного уровня, а не в развитии общей теории оптимизации.

Для практической применимости модели важно, чтобы компоненты решения опирались на измеримые характеристики: длины очередей (или косвенные показатели загрузки), интенсивность служебных операций, а также параметры межузлового сетевого обмена. Разделение сетей клиентского трафика и служебного межузлового обмена в распределенных хранилищах подчеркивает, что потоки служебных операций могут быть измерены и учитываться как отдельный объект управления [120]. Тем самым алгоритм интегрального выбора узлов ориентирован на практическую реализацию в инфокоммуникационной инфраструктуре, поскольку использует измеримые сетевые параметры для управления пользовательскими и служебными информационными потоками.

2.5 Постановка задачи оптимизации

Целью управления нагрузкой в распределенной подсистеме хранения туманного уровня является повышение производительности обслуживания пользовательских операций (прежде всего операций записи) при сохранении требуемого уровня готовности A и соблюдении ресурсных ограничений вычислительных узлов и сети. Для формализации данной цели используется постановка задачи оптимизации по средним (временным) показателям в стохастической системе с очередями, что является стандартным подходом для распределенных систем хранения туманного уровня как элемента телекоммуникационной инфраструктуры со случайными поступлениями и изменяющимися состояниями [115]. В настоящей работе указанная цель рассматривается в контексте телекоммуникационной инфраструктуры туманного уровня: межузловой обмен при репликации и восстановлении формирует служебный сетевой трафик, влияющий на QoS операций хранения. Поэтому далее оптимизационная постановка используется как инженерный способ разработки правил управления информационными потоками в межузловой сети подсистемы хранения, а не как самостоятельное развитие методов оптимизации.

2.5.1 Показатели качества и целевая функция

В качестве основной характеристики производительности рассматривается пропускная способность операций записи $P_{\text{зап}}$, определяемая как средняя интенсивность успешно завершенных операций записи. Дополнительно фиксируется качество обслуживания по задержке, включая показатели на высоких процентилях распределения задержки (например, 95-й и 99-й процентиля), поскольку именно «хвост» задержек способен определять воспринимаемую производительность и устойчивость сервиса в крупномасштабных распределенных системах [66]. Высокие процентиля задержки трактуются как ключевой показатель *QoS* инфокоммуникационного сервиса на туманном уровне и зависят не только от дискового обслуживания, но и от интенсивности межузлового сетевого обмена, обусловленного репликацией и процедурами восстановления. Обоснование выбора и типовые наборы показателей для облачных, туманных и периферийных вычислений приведены в обзоре метрик оценки производительности [107].

Поскольку управление в данной работе направлено на согласование пользовательской и служебной составляющих нагрузки, целевая функция должна учитывать как состояние очередей и задержки обслуживания пользовательских операций, так и интенсивность служебных процессов, связанных с восстановлением избыточности, перераспределением данных и межузловым обменом.

При введенных ранее $Q_i(k)$, $B_i(k)$ и $\text{Net}(k)$ задача может формулироваться как минимизация средневременного функционала (2.29):

$$\min_{\{u(k)\}} \limsup_{K \rightarrow \infty} \frac{1}{K} \sum_{k=0}^{K-1} \left(\sum_{i \in \mathcal{N}} w_Q Q_i(k) + \sum_{i \in \mathcal{N}} w_B B_i(k) + \sum_{i \in \mathcal{N}} w_N \text{Net}(k) \right), \quad (2.29)$$

где $w_Q, w_B, w_N \geq 0$ — веса, задающие приоритет качества обслуживания пользовательских операций относительно интенсивности служебных операций и объема сетевого обмена.

Предлагаемый подход основан на применении аппарата оптимизации динамических сетей к задаче управления пользовательскими и служебными информационными потоками в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений. Учет ограничений межузлового обмена и требований *QoS* позволяет интерпретировать данную задачу в прикладном телекоммуникационном контексте.

2.5.2 Переменные управления

Управляющее решение на шаге k задается набором (2.24), введенным ранее. Такое управление отражает практическую необходимость ограничивать восстановление и перераспределение данных для сохранения эксплуатационной производительности: в документации одного из ключевых производителей СХД прямо указывается, что процессы восстановления выполняются с ограничениями, чтобы система работала приемлемо в деградированном состоянии [122]. Тем самым переменные управления интерпретируются как настраиваемые режимы работы инфраструктурного сервиса хранения, влияющие на баланс между пользовательским *QoS* и служебным трафиком (нагрузкой) распределенной подсистемы хранения. При этом указанное в выражении $\pi(k)$, означает воздействие, связанное с размещением/переразмещением данных и реплик. Принципы управляемого и масштабируемого размещения реплик в динамически изменяющемся составе узлов распределенных хранилищ описаны в работах, посвященных функциям размещения и объектным распределенным хранилищам [116,123].

2.5.3 Ограничения

При постановке задачи оптимизации управляющие воздействия выбираются не произвольно, а из множества допустимых решений, определяемого как физическими ограничениями модели системы, так и требованиями к готовности

выполнения операций записи, при этом для любого шага k должно выполняться $u(k) \in \mathcal{U}(k)$. Допустимая область решений задается ограничениями, имеющими инженерную природу и отражающими характеристики инфокоммуникационной инфраструктуры туманного уровня, такое задание ограничений обеспечивает применимость получаемых управлений в реальной телекоммуникационной среде.

Ограничения реализуемости

Множество $\mathcal{U}(k)$ включает ограничения физической реализуемости, заданные в разделе 2.3.2, то есть ограничения на распределение поступивших операций, диапазоны управляющих параметров, а также ресурсные и сетевые ограничения, вытекающие из ограниченности вычислительных, дисковых и коммуникационных ресурсов. В дальнейшем эти ограничения рассматриваются как «модельные» и не повторяются в явном виде, поскольку они полностью определяют базовую реализуемость выбранного управления.

Ограничение по коэффициенту готовности

Ключевым ограничением задачи оптимизации является требование обеспечить готовность выполнения операций записи не ниже заданного уровня (2.3). Данное условие интерпретируется как ограничение на множество допустимых управлений: решения, которые приводят к снижению готовности ниже $A_{\min}$, исключаются из $\mathcal{U}(k)$, что на практике означает, что управление не должно переводить подсистему хранения в режимы, при которых операция записи не может быть завершена с заданными гарантиями (например, из-за недопустимой деградации критических узлов, недоступности требуемого числа реплик либо нарушения допустимого режима подтверждения записи).

Ограничения на восстановление избыточности

Если управление включает регулирование интенсивности служебных операций восстановления и перераспределения данных, необходимо обеспечить, чтобы повышение пользовательской производительности не достигалось ценой

недопустимого замедления восстановления избыточности. Для этого вводится дополнительное ограничение на служебные операции, которое может быть задано в одной из эквивалентных форм:

- ограничение на средневременной уровень очереди служебных операций,
- ограничение на среднее (или гарантированное) время восстановления избыточности,
- ограничение на суммарную интенсивность (объем) служебного межузлового обмена, обусловленного восстановительными процессами.

Конкретная форма ограничения выбирается в соответствии с методикой оценки в экспериментальной части, при этом само наличие такого ограничения фиксирует допустимый компромисс между пользовательской и служебной составляющими нагрузки.

Если управление включает регулирование служебных операций, необходимо обеспечить, чтобы сокращение времени обслуживания пользовательских операций не достигалось за счет недопустимого накопления служебной нагрузки, связанной с репликацией, синхронизацией, восстановлением избыточности и перерасмещением данных. В качестве условия допустимости управления вводится ограничение на среднее по времени значение суммарной служебной очереди $B_i(k)$ (2.30):

$$\limsup_{K \rightarrow \infty} \frac{1}{K} \sum_{k=0}^{K-1} \sum_{i \in \mathcal{N}} B_i(k) \leq B_{\max}, \quad (2.30)$$

В настоящей работе данное ограничение используется как модельное условие допустимости управления: служебная нагрузка не должна накапливаться неограниченно и вытеснять пользовательские операции записи. В экспериментальной части влияние служебной нагрузки оценивается через ее

воздействие на время завершения операции записи, верхние процентиля задержки и допустимую интенсивность входного потока операций записи.

Итоговое определение допустимой области решений

С учетом изложенного множество допустимых управлений представим как пересечение (2.31):

$$\mathcal{U}(k) = \mathcal{U}_{\text{модель}}(k) \cap \mathcal{U}_A(k) \cap \mathcal{U}_{\text{восст}}(k) \quad (2.31)$$

где $\mathcal{U}_{\text{модель}}(k)$ – множество решений, удовлетворяющих ограничениям реализуемости математической модели (раздел 2.3), $\mathcal{U}_A(k)$ – множество решений, удовлетворяющих условию $A \geq A_{\min}$, $\mathcal{U}_{\text{восст}}(k)$ – множество решений, обеспечивающих допустимый уровень (интенсивность) восстановительных процессов (вводится при наличии соответствующего механизма управления).

Такое представление позволяет далее строить правило управления, которое выбирает $u(k)$ на основе наблюдаемого состояния системы, оставаясь внутри $\mathcal{U}(k)$ и, следовательно, соблюдая как ресурсные ограничения модели, так и требования к готовности выполнения операций записи.

2.6 Выводы

В настоящей главе:

- построена математическая модель распределенной подсистемы хранения туманного уровня, в которой информационные потоки представлены пользовательским и служебным трафиком, формирующим нагрузку на узлы хранения и межузловые соединения,
- определен набор показателей для оценки повышения производительности: основным показателем принята интенсивность обслуживания операций записи, дополнительно контролируются задержки на высоких процентилях и показатели служебных операций,

- сформулирована задача управления информационными потоками и задана допустимая область решений $\mathcal{U}(k)$, включающая ресурсные и сетевые ограничения, а также ограничение по коэффициенту готовности $A \geq A_{\min}$,
- разработан метод оценки готовности выполнения операции записи с репликацией данных, учитывающий фактор репликации, порог подтверждения и доступность элементов цепочки обслуживания операции.

3 Разработка алгоритма выбора узлов хранения при выполнении операции записи с репликацией данных

Результаты второй главы позволяют перейти от анализа функционирования распределенной подсистемы хранения к разработке алгоритмического решения задачи управления нагрузкой. Для рассматриваемой архитектуры ключевое значение имеет не только наличие избыточных узлов хранения, но и порядок их использования при выполнении операции записи с репликацией данных.

Модель, сформированная во второй главе, далее используется как основа для построения алгоритма выбора узлов хранения. Переменные состояния характеризуют пользовательскую и служебную составляющие нагрузки, а допустимая область решений $U(k)$ задает ограничения по готовности, восстановлению избыточности и использованию ресурсов инфокоммуникационной инфраструктуры. На этой основе в третьей главе формируется практическое правило выбора управляющего воздействия $u(k) \in U(k)$, ориентированное на обслуживание операции записи с репликацией данных.

Предлагаемый алгоритм основан на учете текущего состояния распределенной инфраструктуры и предполагает выбор узлов хранения с учетом пользовательских очередей, служебной межузловой нагрузки, сетевых задержек и ограничений по готовности. Такой подход позволяет снизить вероятность направления реплик на перегруженные или недостаточно надежные элементы системы, а также учитывать влияние служебного трафика на качество обслуживания пользовательских операций.

В настоящей главе формулируется задача управления информационными потоками, уточняются входные параметры и ограничения алгоритма, описываются

процедура выбора узлов хранения и порядок регулирования служебной межузловой нагрузки. Разработанный алгоритм используется в четвертой главе как основа для экспериментальной оценки при различных параметрах репликации, режимах подтверждения записи и условиях неоднородности телекоммуникационной инфраструктуры.

3.1 Алгоритм выбора узлов хранения и регулирования служебной нагрузки при выполнении операции записи

В настоящей работе управление информационными потоками в распределенной подсистеме хранения, размещаемой на уровне туманных вычислений, рассматривается как выбор управляющего воздействия $u(k) \in U(k)$ на каждом шаге функционирования системы. В отличие от статических правил распределения, при которых поток операций назначается на узлы по заранее заданной схеме, предлагаемый алгоритм использует наблюдаемое состояние инфраструктуры и формирует решение с учетом пользовательской нагрузки, служебной нагрузки, сетевой составляющей и ограничений по готовности [124]. Необходимость учета нескольких факторов состояния при распределении нагрузки в средах туманных вычислений согласуется с работами, где задача распределения вычислительной нагрузки рассматривается как многокритериальная и зависящая от характеристик ресурсов, требований задачи и состояния среды [125].

В дальнейшем управляющее воздействие рассматривается как совокупность двух взаимосвязанных решений (3.1):

$$u(k) = \{R(k), b_i(k)\}, \quad (3.1)$$

где $R(k)$ – множество узлов хранения, выбранных для размещения реплик при выполнении операции записи на шаге k ,

$b_i(k)$ – вектор коэффициентов регулирования служебной нагрузки. Коэффициент $b_i(k) \in [0,1]$ задает долю обслуживающего ресурса i -го узла,

выделяемую служебным операциям репликации, синхронизации, восстановления и перераспределения данных. Соответственно, доля ресурса, доступная для обслуживания пользовательских операций, составляет $1 - b_i(k)$.

Коэффициенты $b_i(k)$ непосредственно связаны с моделью распределения обслуживающего ресурса, введенной в разделе 2.3.1. При производительности узла μ_i доля $b_i(k)$ выделяется служебным операциям, а доля $1 - b_i(k)$ остается доступной для пользовательских операций. Поэтому интенсивности обслуживания пользовательской и служебной составляющих определяются соотношениями (2.20) То есть, $b_i(k)$ является не интенсивностью поступления операций, а управляющим коэффициентом распределения обслуживающего ресурса узла между пользовательской и служебной нагрузкой.

Первое решение определяет, на какие узлы хранения будут направлены реплики пользовательской операции записи. Второе решение определяет, в какой степени служебные операции репликации, синхронизации, восстановления и перераспределения данных могут использовать обслуживающий ресурс узлов без недопустимого ухудшения качества обслуживания пользовательских операций. Так, алгоритм обеспечивает совместное решение задач выбора узлов хранения и регулирования служебной межузловой нагрузки.

Пусть на шаге управления k для выполнения операции записи требуется сформировать R реплик. Из общего множества узлов хранения $\mathcal{N} = \{1, 2, \dots, N\}$ необходимо выбрать подмножество узлов хранения (3.2)

$$R(k) = \{i_1(k), i_2(k), \dots, i_R(k)\} \subset \mathcal{N}, \quad (3.2)$$

где $i_1(k), i_2(k), \dots, i_R(k)$ – номера узлов, выбранных для размещения реплик на шаге управления k .

Операция записи считается завершенной после получения W подтверждений от выбранных узлов, где $1 \leq W \leq R$. При $W = R$ реализуется строгий режим подтверждения, при котором время завершения операции определяется ветвью

обслуживания с наибольшим временем завершения. При $W < R$ используется пороговый режим подтверждения, при котором операция завершается после получения заданного числа подтверждений. Следовательно, выбор множества $R(k)$ влияет не только на распределение нагрузки между узлами, но и на временные характеристики выполнения операции записи.

Состояние каждого узла хранения на шаге управления k задается вектором наблюдаемых параметров (3.3)

$$S_i(k) = (Q_i(k), B_i(k), d_i(k), A_i(k)), \quad (3.3)$$

где $Q_i(k)$ – длина пользовательской очереди, определяемая как число пользовательских операций записи, ожидающих обслуживания на i -м узле на шаге управления k , $B_i(k)$ – длина служебной очереди, определяемая как число служебных операций репликации, синхронизации, восстановления или перераспределения данных, ожидающих обслуживания на i -м узле на шаге управления k , $d_i(k)$ – оценка сетевой задержки взаимодействия с данным узлом, $A_i(k)$ – показатель готовности узла или соответствующей цепочки обслуживания операции записи.

На первом этапе алгоритма формируется множество допустимых узлов хранения (3.4):

$$\mathcal{N}_{\text{доп}}(k) = \{i \in \mathcal{N} : A_i(k) \geq A_{\min}, Q_i(k) \leq Q_{\max}, B_i(k) \leq B_{\max}\}. \quad (3.4)$$

В данное множество включаются только те узлы, которые удовлетворяют ограничениям по готовности, пользовательской нагрузке и служебной нагрузке. Тем самым алгоритм исключает решения, при которых снижение времени обслуживания могло бы быть достигнуто за счет выбора узлов, не обеспечивающих требуемую готовность или находящихся в состоянии перегрузки.

Обозначим через $N_{\text{доп}}(k)$ число узлов, входящих в множество $\mathcal{N}_{\text{доп}}(k)$. Если $N_{\text{доп}}(k) < R$, то в текущем состоянии инфраструктуры невозможно сформировать требуемое число реплик без нарушения заданных ограничений. В этом случае алгоритм фиксирует отсутствие допустимого управляющего решения и переводит подсистему в режим ограниченной функциональности: новые назначения пользовательских операций на узлы, не удовлетворяющие заданным ограничениям, ограничиваются, а служебные операции восстановления допустимого состояния получают повышенный приоритет.

Если число допустимых узлов достаточно для формирования требуемого числа реплик, то для каждого узла $i \in N_{\text{доп}}(k)$ на основе параметров состояния $S_i(k)$ рассчитывается интегральная оценка выбора узла (3.5):

$$C_i(k) = \alpha \hat{Q}_i(k) + \beta \hat{B}_i(k) + \gamma \hat{d}_i(k), \quad (3.5)$$

где $\hat{Q}_i(k)$, $\hat{B}_i(k)$, $\hat{d}_i(k)$ – нормированные значения пользовательской очереди, служебной нагрузки и сетевой задержки соответственно, α , β , γ – весовые коэффициенты, задающие относительную значимость указанных факторов. Показатель $A_i(k)$ не включается в интегральную оценку $C_i(k)$, а используется как ограничение допустимости при формировании множества $N_{\text{доп}}(k)$.

Нормирование параметров выполняется для приведения величин различной размерности к сопоставимому виду (3.6):

$$\begin{aligned} \hat{Q}_i(k) &= \frac{Q_i(k)}{Q_{\max}}, \\ \hat{B}_i(k) &= \frac{B_i(k)}{B_{\max}}, \\ \hat{d}_i(k) &= \frac{d_i(k)}{d_{\max}}. \end{aligned} \quad (3.6)$$

Узлы хранения упорядочиваются по возрастанию значения $C_i(k)$, после чего для размещения реплик выбираются первые R узлов (3.7):

$$R(k) = \{i_1, \dots, i_R\}: C_{i_1}(k) \leq C_{i_2}(k) \leq \dots \leq C_{i_R}(k). \quad (3.7)$$

Такой принцип выбора позволяет снизить вероятность включения в операцию записи узлов с высокой пользовательской очередью, значительной служебной нагрузкой или неблагоприятной сетевой задержкой. В отличие от выбора только по минимальной пользовательской очереди, предложенный критерий учитывает влияние служебных операций и сетевого взаимодействия, что особенно существенно для туманной инфраструктуры, где репликация и восстановление данных формируют межузловой служебный трафик.

После выбора множества $R(k)$ используется механизм регулирования служебной нагрузки, направленный на предотвращение ситуации, при которой служебные процессы занимают ресурсы, необходимые для обслуживания пользовательских операций. Для каждого узла задается параметр $b_i(k) \in [b_{\min}, b_{\max}]$, где $b_{\min}$ исключает полную остановку процессов восстановления и поддержания избыточности, а $b_{\max}$ предотвращает чрезмерное вытеснение пользовательских операций служебной активностью.

Значение $b_i(k)$ изменяется в зависимости от текущего состояния пользовательской и служебной очередей. Если пользовательская очередь превышает допустимый порог, интенсивность служебных операций ограничивается, чтобы сохранить качество обслуживания пользовательского потока. Если служебная очередь накапливается сверх допустимого уровня, доля ресурса, выделяемая служебным операциям, увеличивается, чтобы не допустить недопустимого замедления восстановления избыточности.

В общем виде правило регулирования может быть представлено следующим образом (3.8):

$$b_i(k+1) = \min[b_{\max}, \max(b_{\min}, b_i(k) + \eta_B \Delta B_i(k) - \eta_Q \Delta Q_i(k))], \quad (3.8)$$

где η_B и η_Q – коэффициенты чувствительности к росту служебной и пользовательской очередей соответственно,

$\Delta B_i(k)$ и $\Delta Q_i(k)$ – нормированные превышения служебной и пользовательской очередей над заданными порогами.

Данное правило отражает компромисс между двумя требованиями. С одной стороны, при росте пользовательской нагрузки алгоритм должен ограничивать служебную активность, чтобы не ухудшать задержки операций записи. С другой стороны, при накоплении служебной очереди алгоритм не должен полностью подавлять восстановительные процессы, поскольку это может привести к нарушению требований к избыточности и готовности.

Последовательность работы алгоритма включает следующие этапы.

- 1) получение текущего состояния узлов хранения: $Q_i(k)$, $B_i(k)$, $d_i(k)$, $A_i(k)$,
- 2) формирование множества допустимых узлов $N_{\text{доп}}(k)$ с учетом ограничений по готовности, пользовательской и служебной нагрузке,
- 3) проверка возможности выбора требуемого числа узлов для размещения реплик,
- 4) расчет интегральной оценки $C_i(k)$ для каждого допустимого узла,
- 5) выбор множества $R(k)$, включающего R узлов с минимальными значениями интегральной оценки,
- 6) направление реплик операции записи на выбранные узлы,
- 7) ожидание W подтверждений в соответствии с заданным режимом подтверждения записи.
- 8) обновление пользовательских и служебных очередей,
- 9) коррекция параметров $b_i(k)$, регулирующих долю ресурса, выделяемую служебным операциям,

10) проверка принадлежности сформированного управления допустимой области $U(k)$.

Таким образом, предлагаемый алгоритм реализует согласованный выбор узлов хранения и регулирование служебной нагрузки. Его отличие от базовых стратегий распределения состоит в том, что решение принимается не только по пользовательской очереди, но и с учетом служебной нагрузки, сетевой составляющей и ограничения по готовности, что позволяет рассматривать алгоритм как средство управления пользовательскими и служебными информационными потоками в распределенной подсистеме хранения туманного уровня.

В дальнейшем экспериментальная проверка сосредоточена на выборе множества узлов хранения $R(k)$, поскольку данное решение непосредственно определяет время завершения операции записи с репликацией данных. Регулирование служебной нагрузки рассматривается как составная часть алгоритма и задает ограничения, обеспечивающие согласование сокращения времени завершения операции записи с требованиями к допустимому состоянию служебных процессов.

Ниже на схеме (рисунок 9) приведено описание алгоритма, реализующего выбор $u(k) = \{c_i(k), b_i(k)\}$ на шаге k .

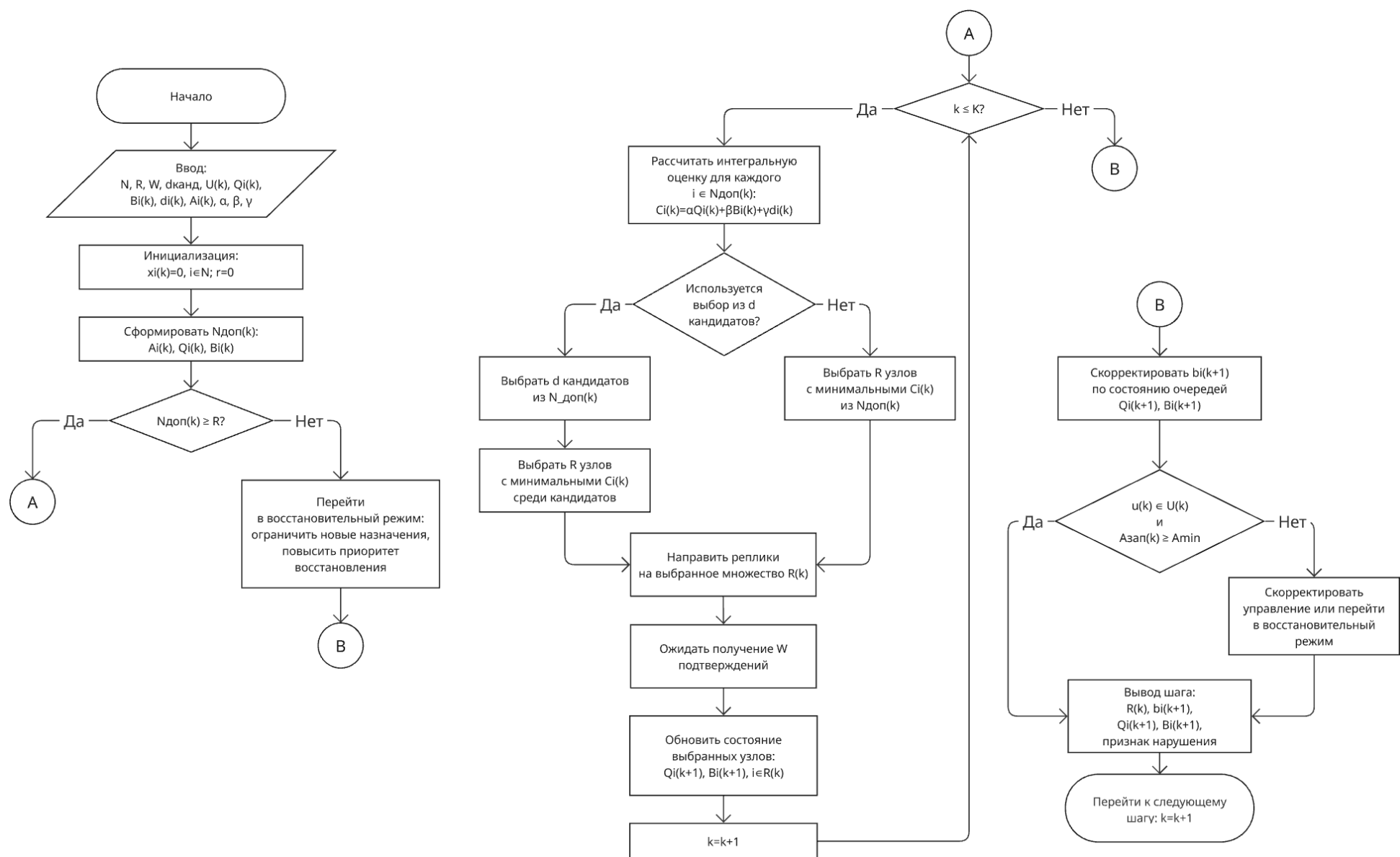


Рисунок 3.1 – Алгоритм выбора узлов хранения и регулирования нагрузки при выполнении операции записи с репликацией данных

3.2 Реализуемость, наблюдаемость и вычислительная сложность

Предложенный алгоритм интегрального выбора узлов является реализуемым в распределенной подсистеме хранения туманного уровня, поскольку использует управляющие воздействия, соответствующие штатным механизмам функционирования такой подсистемы: выбор узлов хранения для размещения реплик операции записи и регулирование интенсивности служебных операций, связанных с репликацией, синхронизацией, восстановлением и перераспределением данных.

Процедура выбора узлов хранения реализуется средствами диспетчеризации операций записи, размещаемыми на уровне координатора, управляющего компонента или клиентского слоя доступа к данным, программная реализация функций организации доступа к управлению данными может быть обеспечена средствами разработанной программы для ЭВМ [126]. При поступлении операции записи координатор формирует множество допустимых узлов $N_{\text{доп}}(k)$, рассчитывает оценку предпочтительности узлов и выбирает подмножество $R(k)$, на которое направляются реплики. Такой механизм соответствует архитектуре распределенной подсистемы хранения, в которой маршрутизация операций к узлам хранения и выбор реплик выполняются на основании служебной информации подсистемы метаданных и конфигурации: сведений о размещении данных, политике репликации, доменных ограничениях и текущем состоянии инфраструктуры.

Вторая составляющая алгоритма связана с регулированием служебной нагрузки и реализуется за счет задания допустимой доли обслуживающего ресурса узла, используемой служебными операциями. Коэффициент $b_i(k)$ задает долю ресурса i -го узла, выделяемую восстановительным, синхронизационным и перераспределительным процессам. В реальных системах такое воздействие может быть реализовано через ограничение параллелизма фоновых процессов, задание предельной интенсивности служебного обмена или изменение приоритета служебных операций относительно пользовательских операций записи. При этом

значения $b_i(k)$ ограничиваются диапазоном $[b_{\min}; b_{\max}]$, что исключает как полную остановку восстановительных процессов, так и чрезмерное вытеснение пользовательских операций служебной нагрузкой.

Проверка принадлежности управления допустимой области $U(k)$ выполняется через контроль ограничений, введенных во второй главе. В частности, учитываются ограничения по готовности выполнения операции записи, допустимым состояниям очередей, интенсивности служебных операций и использованию коммуникационных ресурсов межузловой сети подсистемы хранения. Поэтому алгоритм не рассматривает выбор узлов хранения только как задачу минимизации задержки, а формирует решение в пределах эксплуатационно допустимых режимов функционирования подсистемы хранения.

Для применения алгоритма требуется доступ к следующим наблюдаемым характеристикам состояния системы (3.3). Указанные в (3.3) параметры являются измеримыми или оцениваемыми в процессе эксплуатации распределенной подсистемы хранения. В формальной модели $Q_i(k)$ определяется как число пользовательских операций записи, ожидающих обслуживания на i -м узле. В практической реализации при отсутствии прямого доступа к длине очереди допускается использование эксплуатационных оценок, коррелирующих с очередью, например среднего времени ожидания или загрузки обслуживающего ресурса узла, однако в расчетной модели далее используется именно $Q_i(k)$ как длина очереди. Служебная нагрузка может оцениваться по интенсивности операций восстановления, синхронизации, репликации и перераспределения данных. Сетевая составляющая может учитываться через задержку передачи, время возврата подтверждения или агрегированную характеристику межузлового обмена, например объем переданных данных либо загрузку каналов. Готовность может определяться на основании состояния узла, доступности его компонентов, состояния сетевой связности и принятого режима подтверждения записи.

Алгоритм не требует априорного знания статистических характеристик входного потока и не предполагает точного прогнозирования будущих состояний системы. Решение принимается на основании текущего наблюдаемого состояния,

что соответствует условиям туманной инфраструктуры, где интенсивность пользовательских запросов, состояние сетевых соединений и объем служебных операций могут изменяться во времени.

При этом практическая применимость алгоритма зависит от актуальности телеметрической информации о состоянии узлов. В модели принимается, что задержка обновления телеметрической информации τ_{tel} мала по сравнению с интервалом пересчета управляющего воздействия Δt , данное условие задается в виде $\tau_{tel} \leq \varepsilon_{tel} \Delta t$, где $\varepsilon_{tel} \in (0; 1)$ – допустимая доля запаздывания телеметрии относительно интервала пересчета управляющего воздействия.

При выполнении данного условия значения пользовательских очередей, служебной нагрузки и сетевых задержек, используемые при выборе множества узлов хранения, считаются актуальными для текущего шага управления. Если задержка обновления телеметрии становится сопоставимой с Δt , алгоритм может использовать устаревшую информацию о состоянии инфраструктуры, что снижает качество выбора узлов. В таком случае для повышения устойчивости решения целесообразно использовать сглаженные оценки состояния, увеличить интервал пересчета управляющего воздействия либо ограничить число узлов-кандидатов, для которых требуется оперативное обновление телеметрии.

Вычислительная сложность алгоритма определяется числом узлов хранения n , числом операций записи, поступивших на шаге управления k , числом формируемых реплик R и, при использовании вероятностного выбора, числом кандидатов d .

Формирование множества допустимых узлов требует проверки ограничений для каждого узла хранения и имеет сложность $O(n)$. Расчет интегральной оценки $C_i(k)$ для всех допустимых узлов также выполняется за $O(n)$. Если выбор R узлов осуществляется путем полной сортировки множества допустимых узлов по значению $C_i(k)$, сложность составляет $O(n \log n)$.

При использовании процедуры выбора только R минимальных элементов без полной сортировки сложность может быть снижена до $O(n \log R)$.

Если применяется вариант выбора из ограниченного числа кандидатов d , где $d \ll n$, то расчет и выбор выполняются только для подмножества кандидатов, а сложность выбора для одной операции записи может быть оценена как $O(d \log R)$.

Такой вариант снижает вычислительные затраты и объем требуемой телеметрии, сохраняя принцип предпочтения менее загруженных и более благоприятных по сетевым условиям узлов.

Для множества операций записи, поступивших на шаге управления k , вычислительная сложность выбора узлов хранения определяется как $O(C(k)n \log R)$ при полном рассмотрении множества допустимых узлов и $O(C(k)d \log R)$ при выборе из ограниченного числа кандидатов.

Коррекция параметров $b_i(k)$, регулирующих служебную нагрузку, выполняется независимо для каждого узла и имеет сложность $O(n)$.

Таким образом, общий порядок вычислительной сложности алгоритма определяется главным образом процедурой выбора узлов хранения для операций записи. В практической реализации вычислительная нагрузка может быть снижена за счет периодического пересчета оценок $C_i(k)$, пакетной обработки поступивших операций, ограничения числа кандидатов d и использования локальных оценок состояния узлов.

Предложенный алгоритм интегрального выбора узлов является масштабируемым в том смысле, что его основная вычислительная нагрузка возрастает линейно или квазилинейно с увеличением числа узлов хранения. При этом возможность выбора из ограниченного числа кандидатов позволяет применять алгоритм в распределенных подсистемах хранения с большим числом узлов без необходимости полного перебора всей инфраструктуры для каждой операции записи.

Следовательно, реализуемость алгоритма обеспечивается тем, что он опирается на штатные механизмы диспетчеризации операций записи и регулирования служебных процессов; наблюдаемость обеспечивается использованием измеримых показателей очередей, служебной нагрузки, сетевых

задержек и готовности; вычислительная сложность допускает практическую реализацию как при полном рассмотрении множества узлов, так и при использовании ограниченного числа кандидатов.

3.3 Анализ свойств алгоритма и ожидаемого эффекта

Предложенный алгоритм интегрального выбора узлов относится к классу правил управления, использующих наблюдаемое состояние распределенной подсистемы хранения и формирующих управляющее воздействие $u(k)$ в пределах допустимой области решений $U(k)$. В отличие от статических стратегий распределения нагрузки, алгоритм не задает фиксированное соответствие между входным потоком операций и узлами хранения, а на каждом шаге учитывает текущие значения пользовательских очередей, служебной нагрузки, сетевой задержки и готовности элементов инфраструктуры. Такой подход согласуется с результатами теории стохастической оптимизации сетей и моделей ограниченных сетей очередей, где управление по текущему состоянию очередей и ресурсов используется для поддержания устойчивости обслуживания и допустимых средневременных характеристик без априорного знания распределений входных потоков [115,118].

Основное свойство алгоритма состоит в том, что выбор узлов хранения выполняется не по одному локальному признаку, а по совокупности эксплуатационно значимых параметров. Пользовательская очередь $Q_i(k)$ отражает число операций записи, ожидающих обслуживания на i -м узле, и тем самым характеризует локальное накопление пользовательской нагрузки. Служебная очередь $B_i(k)$ характеризует нагрузку, связанную с репликацией, синхронизацией, восстановлением и перераспределением данных. Сетевая задержка $d_i(k)$ позволяет учитывать распределенный характер туманной инфраструктуры, в которой узлы хранения могут находиться в разных сегментах сети. Показатель готовности $A_i(k)$ используется как ограничение допустимости выбора узла для обслуживания операции записи.

За счет предварительного формирования множества допустимых узлов $N_{\text{доп}}(k)$ алгоритм исключает из рассмотрения те элементы инфраструктуры, которые не удовлетворяют требованиям по готовности, пользовательской или служебной нагрузке, что означает, что алгоритм не сводится к минимизации текущей задержки любой ценой: выбираемое управляющее воздействие должно оставаться реализуемым и не должно нарушать ограничения, заданные во второй главе.

Важным свойством предложенного алгоритма является разделение двух взаимосвязанных составляющих управления. Первая составляющая связана с выбором множества узлов хранения $R(k)$, на которые направляются реплики при выполнении операции записи. Вторая составляющая связана с регулированием доли ресурса $b_i(k)$, выделяемой служебным операциям. Такое разделение позволяет управлять временем завершения пользовательской операции записи и одновременно ограничивать накопление служебных операций, необходимых для поддержания избыточности и согласованности данных.

Выбор узлов хранения влияет прежде всего на временные характеристики операции записи. При выполнении записи с репликацией одна пользовательская операция порождает несколько параллельных ветвей обслуживания. Поэтому включение в множество $R(k)$ перегруженного узла либо узла, доступ к которому характеризуется повышенной сетевой задержкой, может увеличить время завершения операции, особенно в строгом режиме подтверждения $W = R$. В пороговом режиме подтверждения $W < R$ зависимость времени ответа от ветвей обслуживания с наибольшим временем завершения снижается, однако соответствующие межузловые операции продолжают формировать служебную нагрузку и использовать коммуникационные ресурсы уровня туманных вычислений.

Регулирование служебной нагрузки влияет на устойчивость функционирования подсистемы хранения во времени, а также позволяет ограничивать конкуренцию служебных процессов с пользовательскими операциями за ресурсы узлов и межузловых соединений, а также предотвращать

накопление восстановительных операций, способное увеличить время завершения последующих операций записи.

Если служебные операции не ограничиваются, то при восстановлении избыточности или перерасмещении данных могут вытеснять пользовательские операции и увеличивать задержки записи. Если, напротив, служебные операции чрезмерно подавляются, возникает риск накопления невыполненных восстановительных процессов и ухудшения состояния избыточности. Поэтому параметр $b_i(k)$ используется для задания компромисса между качеством обслуживания пользовательского потока и необходимостью поддержания работоспособного состояния подсистемы хранения.

С точки зрения теории систем обслуживания алгоритм реализует state-aware-подход, при котором решение принимается на основе текущего состояния очередей и ограничений ресурсов. Данный подход согласуется с моделями управления динамическими сетями, в которых состояние очередей используется для выбора решений, ограничивающих перегрузку и поддерживающих допустимые средневременные характеристики. В настоящей работе данный аппарат используется не как самостоятельное развитие теории оптимизации, а как прикладной инструмент синтеза правила управления для распределенной подсистемы хранения туманного уровня.

Ожидаемый эффект от применения алгоритма связан с несколькими механизмами.

Во-первых, использование пользовательской очереди $Q_i(k)$ позволяет снизить вероятность направления новых операций записи на уже перегруженные узлы. В результате уменьшается локальное накопление очередей и снижается вероятность резкого роста времени ожидания обслуживания.

Во-вторых, учет служебной нагрузки $B_i(k)$ позволяет отличать узлы, свободные с точки зрения пользовательского потока, но занятые восстановлением, синхронизацией или перерасмещением данных, что особенно важно для распределенной подсистемы хранения, поскольку служебные операции

используют те же вычислительные, дисковые и сетевые ресурсы, что и пользовательские операции записи.

В-третьих, учет сетевой задержки $d_i(k)$ позволяет избежать выбора узлов, обращение к которым сопровождается повышенным временем передачи данных и возврата подтверждений. В туманной инфраструктуре данный фактор имеет принципиальное значение, поскольку узлы хранения могут находиться на различных участках сети, а межузловой обмен при репликации входит в цепочку обслуживания операции записи.

В-четвертых, использование ограничения по готовности $A_i(k) \geq A_{\min}$ позволяет исключить режимы, при которых улучшение временных характеристик достигается за счет выбора узлов, не обеспечивающих требуемую доступность выполнения операции записи. Тем самым алгоритм сохраняет связь между управлением производительностью и эксплуатационными требованиями к готовности сервиса.

Ожидаемый выигрыш алгоритма должен проявляться прежде всего в нестационарных и неоднородных режимах функционирования: при росте входной нагрузки, накоплении служебных операций, различии сетевых задержек между ветвями обслуживания, частичной деградации отдельных узлов или наличии альтернативных узлов хранения $N > R$ [127]. В этих условиях статические схемы распределения не учитывают фактическое состояние инфраструктуры и могут направлять реплики на неблагоприятные ветви обслуживания. Предложенный алгоритм интегрального выбора узлов, напротив, использует наблюдаемые параметры состояния и перераспределяет операции записи в пользу менее загруженных и более предпочтительных узлов.

При этом ожидаемый эффект не следует трактовать как универсальное улучшение во всех режимах. При малой входной нагрузке, однородных узлах, стабильных сетевых задержках и отсутствии выраженной служебной активности равномерное распределение может давать результаты, близкие к результатам алгоритма интегрального выбора, поскольку отсутствуют существенные различия между альтернативными ветвями обслуживания.

Вместе с тем даже при невысокой входной нагрузке алгоритм может давать преимущество в неоднородной инфраструктуре, где узлы хранения различаются по производительности, типу накопителей, характеристикам сетевого доступа или доступному обслуживаемому ресурсу. Например, различия между узлами с HDD- и SSD-накопителями могут приводить к неодинаковому времени локального обслуживания операции записи даже при отсутствии выраженного накопления очередей. В этом случае выбор узлов с меньшим ожидаемым временем обслуживания позволяет направлять пользовательские операции на более производительные ветви, тогда как узлы с менее благоприятными характеристиками могут преимущественно использоваться для фоновых служебных операций, восстановления избыточности или перераспределения данных. Методологически данный вывод согласуется с подходами к балансировке нагрузки в гетерогенных кластерах, где различия производительности серверов рассматриваются как самостоятельный фактор выбора обслуживающего узла [128]. Следовательно, наибольший эффект ожидается при появлении неоднородности, конкуренции пользовательской и служебной нагрузки, различий сетевых задержек и наличии альтернативных узлов хранения для выбора множества реплик.

Таким образом, предложенный алгоритм обладает следующими свойствами:

- реализует выбор управляющего воздействия на основе наблюдаемого состояния системы,
- учитывает пользовательскую и служебную составляющие нагрузки,
- использует сетевую составляющую как фактор выбора узлов хранения,
- сохраняет ограничение по готовности как условие допустимости решения,
- объединяет выбор узлов хранения и регулирование служебной активности,
- допускает снижение вычислительной сложности за счет выбора из ограниченного числа кандидатов,
- учитывает неоднородность узлов хранения по производительности и сетевым характеристикам при выборе ветвей обслуживания операции записи,

– ориентирован на повышение устойчивости временных характеристик операции записи в неоднородной туманной инфраструктуре.

Следовательно, применение алгоритма способствует снижению среднего времени обслуживания операции записи и повышению устойчивости системы к локальным перегрузкам, служебным операциям и неблагоприятным сетевым условиям. Данные свойства проверяются в четвертой главе на примере операции записи с репликацией данных, где сопоставляются равномерное распределение и выбор узлов хранения по алгоритму интегрального выбора при различных режимах подтверждения и неоднородности ветвей обслуживания.

3.3.1 Механизм повышения производительности

Механизм повышения производительности в предлагаемом алгоритме связан с управляемым выбором ветвей обслуживания операции записи и регулированием конкуренции между пользовательской и служебной нагрузкой.

В распределенной подсистеме хранения операция записи с репликацией не обслуживается одним узлом, а разделяется на несколько параллельных обращений к узлам хранения. Поэтому итоговое время выполнения операции определяется не только средней загрузкой подсистемы, но и состоянием выбранных ветвей обслуживания. Если в набор реплик включается узел с высокой пользовательской очередью, значительной служебной нагрузкой или увеличенной сетевой задержкой, то время завершения операции записи может существенно возрасти. Указанное особенно выражено при строгом режиме подтверждения, когда операция считается завершенной только после получения подтверждений от всех выбранных узлов.

Предложенный алгоритм интегрального выбора узлов снижает вероятность выбора таких неблагоприятных ветвей за счет интегральной оценки (3.5).

Минимизация данной оценки при выборе множества $R(k)$ приводит к тому, что реплики преимущественно направляются на узлы с меньшей пользовательской очередью, меньшей служебной нагрузкой и меньшей сетевой задержкой. Сходная

по направленности идея используется в работах по выбору реплик в распределенных хранилищах данных [129], где учет текущего состояния и вариативности времени ответа серверов рассматривается как способ снижения задержек верхних процентилей. В результате снижается вероятность формирования операции записи, время завершения которой определяется перегруженной или удаленной по задержке ветвью обслуживания.

Дополнительный эффект связан с регулированием параметра $b_i(k)$. При росте пользовательской очереди алгоритм ограничивает долю ресурса, выделяемую служебным операциям, что позволяет уменьшить влияние фоновой активности на задержки пользовательских операций записи. При накоплении служебной очереди алгоритм, напротив, увеличивает ресурс, выделяемый служебным операциям, чтобы не допустить недопустимого замедления восстановления избыточности и ухудшения состояния подсистемы хранения.

Таким образом, производительность повышается за счет трех взаимосвязанных действий:

- 1) исключения из выбора узлов, не удовлетворяющих ограничениям допустимости,
- 2) направления реплик на менее загруженные и более благоприятные по сетевым условиям узлы,
- 3) регулирования служебной активности для снижения конкуренции с пользовательскими операциями записи.

Получаемый эффект должен проявляться в снижении среднего времени обслуживания операции записи, уменьшении верхних процентилей задержки и более устойчивом поведении системы при росте входной нагрузки. При этом контроль ограничения по готовности не позволяет рассматривать повышение производительности изолированно от эксплуатационной устойчивости подсистемы хранения. Именно поэтому предложенный алгоритм оценивается не только по временным характеристикам, но и с учетом принадлежности принимаемых решений допустимой области $U(k)$.

Ожидаемый прирост производительности формируется за счет двух механизмов, заложенных в структуру управления:

- снижение неравномерности распределения пользовательских операций по узлам (предотвращение локальных перегрузок) за счет выбора узла обслуживания по текущей загрузке. Эффект выбора среди нескольких альтернатив как средства уменьшения дисбаланса нагрузки и сокращения очередей обоснован в классических работах по вероятностной балансировке нагрузки [80],
- регулирование служебных операций для уменьшения конкуренции за ресурсы в периоды повышенной пользовательской нагрузки при сохранении требований к восстановлению избыточности.

Практическая значимость такого компромисса подтверждается документацией распределенных хранилищ [123]: увеличение параллелизма восстановления ускоряет восстановление, но повышает нагрузку на распределенную СХД и снижает производительность пользовательского ввода-вывода, поэтому применяются механизмы ограничения восстановления и перераспределения.

Для распределенных систем принципиально важно контролировать задержки на высоких процентилях (например, 95-м и 99-м): именно «хвост» задержек влияет на воспринимаемое качество обслуживания и на суммарное время выполнения операций при параллельной обработке [66], поэтому вывод о повышении производительности верифицируется не только ростом пропускной способности, но и отсутствием ухудшения высоких процентилей задержки при выполнении ограничений.

3.4 Сравнительная оценка стратегий выбора узлов хранения и регулирования служебной нагрузки

Сравнительная оценка вводится для проверки практической применимости предложенного алгоритма и определения его отличий от базовых правил распределения нагрузки, используемых в распределенных системах обслуживания.

В обзорных работах по распределению потоков данных Интернета вещей в облачных и туманных средах также подчеркивается, что статические и динамические алгоритмы различаются по объему используемой информации о состоянии системы и по способности реагировать на изменение нагрузки [130].

В рамках настоящей работы сравнение выполняется для стратегий, которые могут быть описаны в терминах введенной ранее модели управления: выбора множества узлов хранения $R(k)$ для размещения реплик операции записи и регулирования доли ресурса $b_i(k)$, выделяемой служебным операциям.

Необходимость такого сравнения обусловлена тем, что повышение производительности распределенной подсистемы хранения туманного уровня может быть достигнуто различными способами. Одни стратегии используют фиксированное распределение входного потока и не требуют информации о текущем состоянии узлов. Другие стратегии опираются на отдельные виды информации о состоянии подсистемы, например пользовательскую очередь, локальную загрузку узла или наличие служебной активности. Отдельно для сравнения рассматривается пороговое регулирование служебной нагрузки, возникающей при репликации, синхронизации, восстановлении и перераспределении данных. Алгоритм интегрального выбора отличается от указанных вариантов тем, что объединяет выбор узлов хранения и регулирование служебной нагрузки в едином управляющем воздействии $u(k)$.

Для сопоставления используются четыре класса стратегий:

- равномерное распределение операций записи без учета текущего состояния узлов,
- диспетчеризация по минимальной пользовательской очереди,
- выбор с учетом пользовательской и служебной очередей,
- предложенный алгоритм интегрального выбора узлов хранения и регулирования служебной нагрузки.

Такой набор стратегий позволяет последовательно выделить вклад отдельных механизмов управления: использования информации о пользовательской очереди, учета служебной нагрузки, учета сетевой

составляющей и контроля ограничений по готовности. При этом для всех стратегий ограничение по готовности рассматривается как условие допустимости выбранного множества узлов, а отличие предложенного алгоритма состоит в использовании данного ограничения уже на этапе формирования множества допустимых узлов.

Равномерное распределение

Равномерное циклическое распределение используется как базовая стратегия, не использующая оперативную телеметрию состояния узлов. При таком подходе операции записи назначаются узлам хранения по фиксированному правилу, например циклически. Если на шаге k поступает $C(k)$ операций записи, то в простейшем случае поток распределяется между узлами приблизительно равномерно (3.9):

$$x_i(k) \approx \frac{C(k)}{n}, \quad (3.9)$$

где $x_i(k)$ – число операций, назначенных на i -й узел на шаге k , n – общее число узлов хранения.

Для операции записи с репликацией равномерная стратегия может быть представлена как циклический выбор множества $R(k)$ из доступных узлов без учета текущих значений $Q_i(k)$, $B_i(k)$, $d_i(k)$ и $A_i(k)$. Достоинство такого подхода состоит в минимальных вычислительных затратах и отсутствии необходимости в сборе телеметрии. Однако в условиях туманной инфраструктуры равномерное распределение не учитывает неоднородность сетевых задержек, накопление пользовательских очередей и конкуренцию со служебными операциями. Поэтому при одинаковом входном потоке $C(k)$ отдельные узлы могут переходить в состояние перегрузки, что приводит к росту времени завершения операции записи и увеличению хвостовых задержек.

В дальнейшем равномерное распределение рассматривается как базовый вариант сравнения, позволяющая оценить эффект от использования наблюдаемого состояния инфраструктуры.

Диспетчеризация по минимальной пользовательской очереди

Диспетчеризация по минимальной пользовательской очереди (или также выбор по минимальной пользовательской очереди) относится к классу динамических стратегий, использующих текущее состояние узлов. В этом случае при выборе узла учитывается значение $Q_i(k)$, отражающее накопление пользовательских операций, ожидающих обслуживания. Для одной операции выбор может быть записан как (3.10)

$$i^*(k) = \arg \min_{i \in N_{\text{доп}}(k)} Q_i(k). \quad (3.10)$$

Для операции записи с репликацией необходимо выбрать не один узел, а множество узлов $R(k)$. В таком случае стратегия выбирает R узлов с минимальными значениями пользовательской очереди (3.11):

$$R(k) = \{i_1, \dots, i_R\}: Q_{i_1}(k) \leq Q_{i_2}(k) \leq \dots \leq Q_{i_R}(k). \quad (3.11)$$

Преимущество данной стратегии состоит в том, что она реагирует на локальную перегрузку и снижает вероятность направления новых операций на узлы с большой пользовательской очередью. Вместе с тем выбор только по $Q_i(k)$ не учитывает служебную нагрузку $B_i(k)$, сетевую задержку $d_i(k)$ и ограничения по готовности. Поэтому узел с малой пользовательской очередью может оказаться неблагоприятным для записи, если в данный момент он занят восстановлением избыточности, участвует в интенсивном межузловом обмене или находится в сетевом сегменте с повышенной задержкой.

Данная стратегия используется как промежуточный вариант сравнения: она позволяет отделить эффект учета пользовательской очереди от эффекта более полного учета состояния инфраструктуры.

Пороговое ограничение служебной нагрузки

Пороговое ограничение фоновой служебной нагрузки (или также выбор с учетом пользовательской и служебной очередей) рассматривается как базовая эксплуатационная стратегия, при которой интенсивность служебных процессов изменяется не по интегральной оценке состояния узлов, а по факту достижения заранее заданных порогов пользовательской или служебной очереди. В отличие от предложенного алгоритма, данная стратегия не формирует множество узлов хранения для размещения реплик, а только регулирует долю ресурса, выделяемую служебным операциям. Фоновая активность ограничивается в периоды роста пользовательской нагрузки, поскольку служебные операции используют общие ресурсы с пользовательскими операциями записи и могут увеличивать время их обслуживания. К таким операциям относятся восстановление избыточности, синхронизация реплик, перераспределение данных, обновление служебных метаданных и иные процессы, использующие те же вычислительные, дисковые и коммуникационные ресурсы, что и пользовательские операции записи.

В рамках данной стратегии параметр $b_i(k)$, определяющий долю ресурса, выделяемую служебным операциям, изменяется в зависимости от состояния пользовательской и служебной очередей. В простейшем виде правило может быть представлено следующим образом (3.12):

$$b_i(k+1) = \begin{cases} b_{\min}, & Q_i(k) > Q_{\text{порог}}, \\ b_{\max}, & B_i(k) > B_{\text{порог}}, \\ b_i(k), & \text{в остальных случаях.} \end{cases} \quad (3.12)$$

Если пользовательская очередь превышает заданный порог $Q_{\text{порог}}$, служебная активность ограничивается для предотвращения дальнейшего роста задержек

пользовательских операций. Если служебная очередь превышает порог $B_{\text{порог}}$, доля ресурса, выделяемая служебным операциям, увеличивается для предотвращения чрезмерного накопления восстановительных процессов.

Достоинство данной стратегии состоит в простоте реализации и прямой связи с реальной практикой ограничения фоновой активности. Однако пороговое регулирование не решает задачу выбора множества узлов $R(k)$ для операции записи и не учитывает сетевую составляющую при направлении реплик. Кроме того, результат существенно зависит от выбора порогов $Q_{\text{порог}}$, $B_{\text{порог}}$, $b_{\min}$ и $b_{\max}$.

Предложенный алгоритм интегрального выбора узлов

Предложенный алгоритм интегрального выбора узлов отличается от рассмотренных базовых стратегий тем, что формирует управляющее воздействие на основе нескольких параметров состояния инфраструктуры. На первом этапе формируется множество допустимых узлов (3.13)

$$N_{\text{доп}}(k) = \{i \in N: A_i(k) \geq A_{\min}, Q_i(k) \leq Q_{\max}, B_i(k) \leq B_{\max}\}. \quad (3.13)$$

Затем для каждого допустимого узла рассчитывается интегральная оценка (3.5), после чего для размещения реплик выбираются R узлов с минимальными значениями $C_i(k)$. Параллельно выполняется регулирование параметра $b_i(k)$, определяющего долю ресурса, выделяемую служебным операциям.

Таким образом, предложенный алгоритм одновременно учитывает пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности, что позволяет рассматривать его как развитие базовых стратегий: от равномерного распределения он отличается использованием текущего состояния системы, от выбора по минимальной очереди – учетом служебной и сетевой составляющих, а от порогового регулирования – согласованным выбором узлов хранения и служебной активности в пределах допустимой области $U(k)$.

Сопоставление стратегий приведено в таблице 3.1. Из таблицы видно, что базовые стратегии используют только отдельные виды информации о состоянии элементов подсистемы. Равномерное распределение используется как исходный базовый вариант сравнения, поскольку не требует сбора телеметрии и не учитывает текущее состояние инфраструктуры. Выбор по минимальной очереди учитывает пользовательскую перегрузку, но не отражает влияние служебных операций и сетевых задержек. Пороговое регулирование служебной нагрузки позволяет ограничивать фоновую активность, но не определяет порядок выбора узлов для размещения реплик, при этом для стратегии порогового регулирования учет пользовательской очереди означает не выбор узлов по $Q_i(k)$, а изменение доли ресурса, выделяемой служебным операциям, при превышении заданного порога пользовательской очереди. Предложенный алгоритм интегрального выбора узлов объединяет указанные механизмы и применяет их в рамках единой допустимой области решений.

В качестве метрик сопоставления используются показатели, согласованные с постановкой задачи управления нагрузкой:

- T_{write} – время завершения операции записи,
- P_{write} – пропускная способность операций записи,
- L_{95}, L_{99} – 95-й и 99-й процентиля задержки операций записи,
- $V_{\text{svc}}(k)$ – объем или интенсивность служебного межузлового обмена,
- $\Delta A(k) = A_{\text{write}}(k) - A_{\text{min}}$ – запас по ограничению готовности выполнения операции записи.

Основной акцент при сопоставлении делается на времени завершения операции записи и верхних процентилях задержки, поскольку именно эти показатели отражают влияние перегруженных или неблагоприятных ветвей обслуживания. При этом показатели служебной нагрузки и соблюдения ограничения по готовности используются как контрольные: они необходимы для проверки того, что улучшение пользовательских метрик не достигается за счет

недопустимого подавления восстановительных процессов или перехода в режимы, не удовлетворяющие требованиям доступности.

Таким образом, сравнительная оценка задает основу для экспериментальной проверки предложенного алгоритма. В четвертой главе рассматривается не только итоговое сопоставление алгоритма интегрального выбора с равномерным распределением, но и промежуточные варианты управления, позволяющие оценить вклад отдельных факторов в изменение времени завершения операции записи. За счет этого экспериментальная часть позволяет последовательно показать, как на результат влияют учет пользовательской очереди, учет служебной нагрузки и включение сетевой составляющей в правило выбора узлов хранения.

Таблица 3.1 – Матрица сопоставления стратегий

Стратегия	Учитывает $Q_i(k)$	Учитывает $B_i(k)$	Учитывает межузловой обмен	Контроль $A_i(k) \geq A_{min}$	Формирует $R(k)$	Регулирует $b_i(k)$
Равномерное циклическое распределение	нет	нет	нет	нет	да, по фиксированному правилу	нет
Выбор по минимальной пользовательской очереди	да	нет	нет	частично, при внешней проверке	да	нет
Выбор с учетом пользовательской и служебной очередей	да, через порог Q_{max}	да	нет	частично, при внешней проверке	нет	да
Предложенный алгоритм интегрального выбора узлов	да	да	да	да	да	да

3.5 Методика сопоставительного эксперимента

Сопоставительный эксперимент предназначен для проверки того, как различные правила формирования управляющего воздействия $u(k)$ влияют на обслуживание операций записи в распределенной подсистеме хранения туманного уровня. Методика эксперимента строится таким образом, чтобы различия в результатах определялись именно стратегией управления, а не изменением внешних условий, состава узлов или параметров нагрузки. Методические подходы к формированию профилей нагрузки, генерации нагрузочных воздействий и организации экспериментальных исследований программно-определяемых систем хранения данных ранее прорабатывались в рамках НИР [131–133]

В рамках эксперимента сопоставляются стратегии, рассмотренные в разделе 3.4: равномерное распределение операций записи, выбор узлов по минимальной пользовательской очереди, пороговое регулирование служебной нагрузки и предложенный алгоритм выбора узлов хранения с регулированием служебной активности. Все стратегии рассматриваются в единой модели состояния и используют одинаковые исходные параметры распределенной подсистемы хранения.

Для всех вариантов фиксируются число доступных узлов хранения N , число формируемых реплик R , порог подтверждения W , профиль входного потока $C(k)$, начальные состояния пользовательских и служебных очередей, а также параметры сетевых задержек и готовности элементов инфраструктуры. В ходе одного эксперимента изменяется только правило формирования управляющего воздействия $u(k)$, то есть порядок выбора множества узлов $R(k)$ и, если это предусмотрено стратегией, порядок регулирования параметра $b_i(k)$.

На каждом шаге моделирования задается входной поток операций записи и текущее состояние инфраструктуры, при этом формирование нагрузочного профиля и сохранение результатов тестирования могут выполняться средствами разработанной программы для ЭВМ [114]. Далее в соответствии с выбранной стратегией формируется управляющее воздействие, определяется множество узлов

хранения для размещения реплик, выполняется проверка принадлежности решения допустимой области $U(k)$, после чего обновляется состояние пользовательских и служебных очередей. Для операции записи учитывается режим подтверждения: при $W = R$ завершение зависит от всех выбранных ветвей обслуживания, при $W < R$ – от момента получения требуемого числа подтверждений.

Показатели сопоставления используются в соответствии с разделами 2.2 и 3.4. Основное внимание уделяется временным характеристикам операции записи и устойчивости их изменения при росте нагрузки, сетевой неоднородности и накоплении служебных операций. Контроль ограничений допустимой области необходим для исключения ситуаций, когда улучшение пользовательских показателей достигается за счет нарушения требований к готовности или недопустимого замедления восстановления избыточности.

Для проверки поведения стратегий рассматриваются следующие сценарии.

Сценарий 1 – изменение интенсивности входного потока

Интенсивность $C(k)$ изменяется ступенчато или по заданному профилю. Сценарий позволяет оценить, как стратегии реагируют на рост пользовательской нагрузки и в какой момент начинают формироваться очереди на отдельных узлах хранения.

Сценарий 2 – неоднородность пользовательских и служебных очередей

Для части узлов задаются повышенные значения $Q_i(k)$ или $B_i(k)$. Сценарий используется для проверки способности алгоритма избегать выбора узлов, находящихся в состоянии пользовательской или служебной перегрузки.

Сценарий 3 – неоднородность сетевых задержек

Один или несколько узлов характеризуются увеличенной задержкой $d_i(k)$. Данный сценарий отражает специфику туманной инфраструктуры, где узлы хранения могут находиться в различных сегментах сети, а межузловой обмен входит в цепочку обслуживания операции записи.

Сценарий 4 – изменение режима подтверждения записи

Сопоставляются строгий режим $W = R$ и пороговый режим $W < R$. Сценарий позволяет определить, как режим подтверждения влияет на чувствительность операции записи к неблагоприятным ветвям обслуживания.

Сценарий 5 – выбор R узлов из множества $N > R$

Данный сценарий используется для анализа ситуации, когда подсистема хранения располагает избыточным числом доступных узлов, а стратегия выбора подмножества реплик становится самостоятельным механизмом управления временем записи и распределением нагрузки.

Сопоставительный эксперимент выполняется в два этапа. На первом этапе проводится расчетная оценка, позволяющая определить ожидаемый характер изменения времени записи и выявить области перегрузки. На втором этапе используется имитационное моделирование, позволяющее проверить поведение системы при изменении входной нагрузки, накоплении очередей и неоднородности ветвей обслуживания.

Результаты интерпретируются с учетом области применимости алгоритма. При малой нагрузке, однородных узлах и стабильных сетевых задержках различия между стратегиями могут быть незначительными, поскольку в таких условиях отсутствует выраженная необходимость в интегральном выборе узлов. Преимущество предложенного алгоритма должно проявляться прежде всего при росте нагрузки, накоплении служебных операций, неоднородности сетевых задержек и наличии альтернативных узлов хранения.

Таким образом, методика сопоставительного эксперимента задает порядок проверки предложенного алгоритма в условиях, согласованных с моделью главы 2 и сопоставительной базой раздела 3.4. В четвертой главе данная методика конкретизируется для экспериментальной оценки операции записи с репликацией данных при различных значениях R , W , N , сетевых задержек и стратегий выбора узлов хранения.

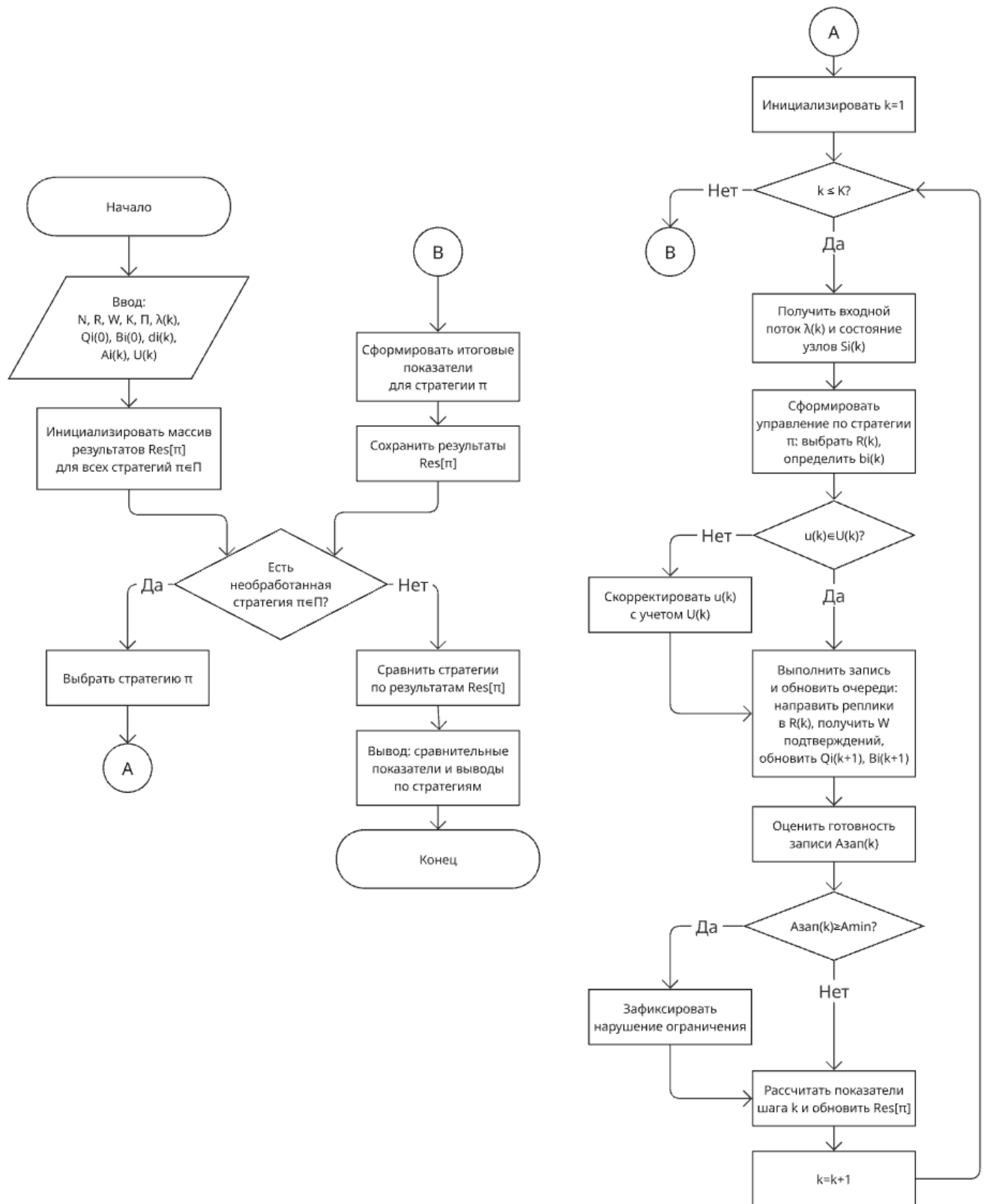


Рисунок 3.2 – Сценарий сопоставительного эксперимента

3.5.1 Ожидаемые результаты

Ожидаемые результаты сопоставления формулируются как проверяемые положения, вытекающие из структуры предложенного алгоритма и принятой модели управления нагрузкой.

Во-первых, при низкой входной нагрузке и однородных характеристиках узлов различия между стратегиями должны быть незначительными, поскольку очереди не оказывают существенного влияния на время обслуживания операции записи, а сетевые задержки близки по значениям.

Во-вторых, при росте интенсивности входного потока преимущество стратегий, использующих текущее состояние узлов, должно возрасти. В этих условиях равномерное распределение может приводить к локальному накоплению очередей, тогда как выбор узлов с учетом $Q_i(k)$ снижает вероятность назначения новых операций на перегруженные элементы подсистемы хранения. Значимость учета длины очереди и неоднородности обслуживающих узлов подтверждается результатами исследований [134] для крупномасштабных гетерогенных систем.

В-третьих, при наличии служебной нагрузки учет только пользовательской очереди является недостаточным. Ожидается, что алгоритм, использующий $B_i(k)$, позволит снизить влияние операций репликации, синхронизации и восстановления на пользовательские операции записи по сравнению со стратегиями, не учитывающими служебную составляющую. Аналогичная проблема конкуренции фоновых операций с пользовательской обработкой рассматривается для систем баз данных с отдельным размещением вычислительных ресурсов и ресурсов хранения: показано [135], что фоновые операции обслуживания данных могут приводить к росту задержек верхних перцентилей.

В-четвертых, при неоднородности сетевых задержек преимущество предложенного алгоритма должно проявляться за счет менее частого включения в множество $R(k)$ узлов, взаимодействие с которыми характеризуется повышенной задержкой. Данный эффект особенно значим для строгого режима подтверждения

$W=R$, при котором время завершения операции записи определяется максимальным временем получения подтверждения от выбранных узлов.

В-пятых, контроль допустимой области $U(k)$ и ограничения по готовности должен предотвращать получение выигрыша по времени обслуживания за счет перехода в эксплуатационно недопустимые режимы. Поэтому улучшение временных характеристик рассматривается как значимое только при сохранении требований к готовности и восстановлению избыточности.

Таким образом, ожидается, что предложенный алгоритм интегрального выбора узлов даст наибольший эффект не в стационарных однородных условиях, а при росте нагрузки, накоплении служебных операций, неоднородности сетевых задержек и наличии альтернативных узлов хранения $N > R$. Указанные положения далее проверяются в четвертой главе при экспериментальной оценке операции записи с репликацией данных.

3.6 Выводы

В настоящей главе разработан алгоритм выбора узлов хранения при выполнении операции записи с репликацией данных в распределенной подсистеме хранения туманного уровня.

Основные результаты главы заключаются в следующем.

- на основе модели, сформированной во второй главе, предложен алгоритм, выбирающий управляющее воздействие $u(k) \in U(k)$ по наблюдаемому состоянию системы. Управляющее воздействие включает выбор множества узлов хранения $R(k)$ для размещения реплик операции записи и регулирование доли ресурса $b_i(k)$, выделяемой служебным операциям,
- формализована процедура выбора допустимых узлов хранения с учетом ограничений по готовности, пользовательской нагрузке и служебной нагрузке, что позволяет исключать из рассмотрения узлы, использование которых

может привести к нарушению требований к готовности или к перегрузке элементов подсистемы хранения,

- предложен критерий выбора узлов хранения, учитывающий пользовательскую очередь $Q_i(k)$, служебную нагрузку $B_i(k)$ и сетевую задержку $d_i(k)$. Такой подход позволяет учитывать не только локальную загрузку узлов, но и влияние межузлового обмена, возникающего при репликации, синхронизации и восстановлении данных,

- показано, что алгоритм предусматривает две взаимосвязанные составляющие управления: выбор узлов хранения для операции записи и регулирование служебной нагрузки, при этом совместная реализация указанных составляющих позволяет сокращать время обслуживания пользовательских операций без нарушения допустимого режима восстановления избыточности,

- рассмотрены реализуемость, наблюдаемость и вычислительная сложность алгоритма. Установлено, что алгоритм использует измеримые или оцениваемые эксплуатационные параметры распределенной подсистемы хранения и может быть реализован как при полном рассмотрении множества узлов, так и при выборе из ограниченного числа кандидатов,

- выполнено сопоставление предложенного алгоритма с базовыми стратегиями: равномерным распределением, выбором по минимальной пользовательской очереди и пороговым регулированием служебной нагрузки. Показано, что базовые стратегии учитывают только отдельные компоненты состояния системы, тогда как предложенный алгоритм интегрального выбора узлов объединяет пользовательскую, служебную, сетевую и эксплуатационную составляющие управления,

- сформирована методика сопоставительного эксперимента, определяющая порядок проверки алгоритма при изменении входной нагрузки, неоднородности очередей, неоднородности сетевых задержек, различных режимах подтверждения записи и выборе R узлов из множества доступных узлов $N > R$.

Таким образом, в третьей главе выполнен переход от математической модели управления информационными потоками к алгоритму интегрального выбора узлов хранения и регулирования служебной нагрузки. Экспериментальная проверка предложенного алгоритма и оценка его влияния на время выполнения операции записи с репликацией данных выполняются в четвертой главе.

4 Экспериментальное исследование и проверка предложенного подхода

В предыдущих главах сформированы модель управления информационными потоками распределенной подсистемы хранения туманного уровня и алгоритм интегрального выбора узлов хранения при выполнении операции записи с репликацией данных. В рамках предложенного алгоритма управляющее воздействие $u(k)$ включает выбор множества узлов хранения $R(k)$, на которые направляются реплики, и регулирование параметра $b_i(k)$, определяющего долю ресурса узла, выделяемую служебным операциям. При этом выбор управления выполняется в пределах допустимой области $U(k)$, заданной ограничениями по готовности, ресурсам и состоянию пользовательской и служебной нагрузки.

Целью настоящей главы является экспериментальная проверка предложенного алгоритма на примере операции записи с репликацией данных в распределенной подсистеме хранения, функционирующей на уровне туманных вычислений. В рамках проверки основное внимание уделяется выбору множества узлов хранения $R(k)$, поскольку это управляющее решение определяет время завершения операции записи, распределение нагрузки между ветвями обслуживания и чувствительность подсистемы к неоднородности сетевых условий.

Экспериментальная оценка проводится в сопоставлении четырех стратегий выбора узлов хранения. В качестве базового варианта используется равномерное циклическое распределение, не учитывающее текущее состояние узлов. В качестве промежуточных вариантов рассматриваются выбор узлов по минимальной пользовательской очереди и выбор с учетом пользовательской и служебной очередей. В качестве исследуемого варианта используется алгоритм интегрального выбора узлов хранения, учитывающий пользовательскую очередь, служебную

нагрузку, сетевую составляющую и ограничения допустимости управления. Такое сопоставление позволяет последовательно выделить вклад отдельных факторов в снижение времени завершения операции записи.

Операция записи рассматривается как параллельный процесс, в котором завершение операции определяется получением заданного числа подтверждений записи. Для ее описания используется представление на основе модели Fork-Join: входящая операция разделяется на несколько ветвей обслуживания, соответствующих выбранным узлам хранения, после чего исходная операция завершается при получении заданного числа подтверждений W . В строгом режиме подтверждения $W = R$ время завершения зависит от всех выбранных ветвей, а в пороговом режиме $W < R$ – от момента получения требуемого числа подтверждений.

Экспериментальное исследование направлено на проверку того, позволяет ли предложенный алгоритм снизить чувствительность операции записи к локальным перегрузкам, накоплению служебной нагрузки и неоднородности сетевых задержек по сравнению с базовыми стратегиями. Полученные результаты используются для определения области применимости алгоритма и уточнения условий, при которых выбор узлов по алгоритму интегрального выбора дает наибольший эффект.

4.1 Постановка задачи экспериментального исследования

Целью экспериментального исследования является проверка влияния стратегии выбора узлов хранения на временные характеристики операции записи с репликацией данных в распределенной подсистеме хранения туманного уровня.

В качестве объекта экспериментального исследования рассматривается операция записи данных с репликацией, выполняемая в распределенной телекоммуникационной инфраструктуре. Предметом исследования является зависимость времени завершения операции записи от стратегии выбора узлов хранения, режима подтверждения, числа доступных узлов, состояния

пользовательских и служебных очередей, а также неоднородности сетевых задержек.

Экспериментальная оценка построена таким образом, чтобы разделить проверку расчетной модели операции записи и сопоставление стратегий выбора узлов хранения. Сначала рассматривается представление операции записи с репликацией как параллельного процесса с ожиданием подтверждений и анализируется влияние режима подтверждения на время завершения операции. Затем проводится сравнение четырех стратегий выбора узлов хранения, такой порядок позволяет оценить не только итоговое преимущество предложенного алгоритма, но и вклад отдельных учитываемых факторов: пользовательской очереди, служебной нагрузки и сетевой составляющей.

При этом ограничение по готовности рассматривается как общее условие допустимости управляющего решения для всех сравниваемых стратегий. При этом равномерное циклическое распределение, выбор по минимальной пользовательской очереди и выбор с учетом пользовательской и служебной очередей не используют $A_i(k)$ как признак ранжирования узлов, проверка готовности выполняется после формирования множества выбранных узлов. В предложенном алгоритме интегрального выбора показатель $A_i(k)$ используется на более раннем этапе – при формировании множества допустимых узлов, внутри которого затем выполняется ранжирование по пользовательской очереди, служебной нагрузке и сетевой задержке.

Операция записи рассматривается как параллельный процесс, в котором завершение исходной операции определяется получением заданного числа подтверждений записи от выбранных узлов хранения: одна пользовательская операция порождает несколько ветвей обслуживания, соответствующих выбранным узлам хранения, а завершение операции определяется моментом получения требуемого числа подтверждений W . Такой подход позволяет оценить влияние строгого режима подтверждения $W=R$, порогового режима $W<R$, выбора R узлов из множества $N>R$, а также изменения характеристик отдельных ветвей обслуживания.

Основным показателем оценки является время завершения операции записи T_{write} . Дополнительно анализируются среднее время записи, верхние процентиля задержки и относительное изменение времени записи при переходе от равномерного распределения к интегральному выбору узлов. При интерпретации результатов учитывается, что улучшение временных характеристик должно рассматриваться только в пределах допустимых режимов функционирования, заданных ограничениями по готовности и состоянию служебной нагрузки.

Задача экспериментального исследования состоит в сравнительной оценке стратегий выбора узлов хранения при выполнении операции записи с репликацией данных в условиях локальной перегрузки, служебной нагрузки и неоднородности сетевых задержек.

Таблица 4.1 – Основные используемые обозначения

Обозначения	Комментарий
R	уровень избыточности, определяющий количество формируемых реплик (резервных копий данных)
W	порог подтверждения
N	количество доступных узлов хранения туманного уровня
T_{write}	время завершения операции
ветвь обслуживания операции записи	обслуживание одной реплики данных на выбранном узле хранения, включая передачу данных, локальную запись и возврат подтверждения
цепочка обслуживания операции	совокупность элементов инфраструктуры, участвующих в выполнении операции записи в целом, включая координатор, сетевые соединения и выбранные узлы хранения
$Q_i(k), B_i(k)$	пользовательская и служебная очереди, учитываемые при выборе узлов хранения
$S_i(k)$	вектор состояния i -го узла хранения
$C_i(k)$	интегральная оценка выбора i -го узла, используемая для ранжирования допустимых узлов

4.2 Модель операции записи с репликацией данных

Для экспериментальной оценки алгоритма интегрального выбора операция записи с репликацией данных представляется как параллельный процесс

с ожиданием подтверждений записи. Такое представление соответствует модели Fork-Join: поступившая операция записи разделяется на несколько ветвей обслуживания, каждая из которых соответствует передаче реплики на выбранный узел хранения, выполнению локальной записи и возврату подтверждения координатору. Математическое описание операции записи с репликацией данных, используемое в настоящем разделе, основано на расчетном аппарате, ранее представленных в работах [136,137]. В рамках настоящей диссертации соответствующий математический аппарат используется при построении экспериментальной модели для оценки разработанного алгоритма интегрального выбора узлов хранения, обеспечивая получение и анализ основных результатов исследования.

Пороговый режим подтверждения записи может быть интерпретирован как частный случай Fork-Join-модели, в которой исходная задача считается завершенной после выполнения k из n параллельных подзадач [138]. Подходы, основанные на формировании избыточных параллельных запросов и завершении исходной операции без ожидания всех сформированных ветвей, также рассматриваются как механизм снижения задержки обслуживания [139]. Применение AnyLogic для построения имитационных моделей сложных систем соответствует практике использования данной среды для дискретно-событийного и агентного моделирования процессов с несколькими взаимодействующими компонентами [140].

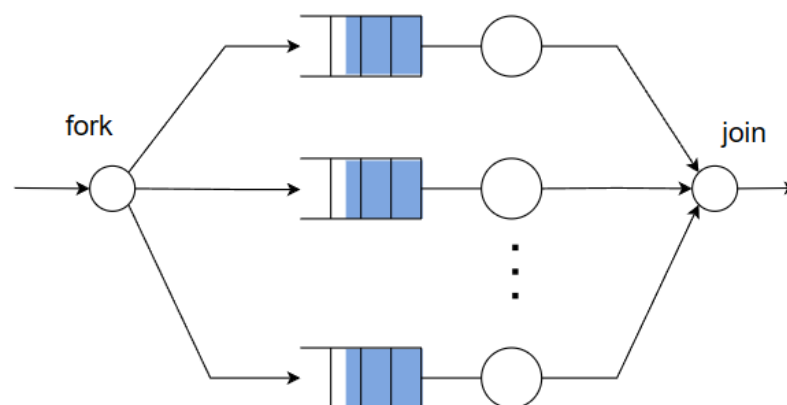


Рисунок 4.1 – Представление операции записи с последующей репликацией в виде Fork-Join-модели

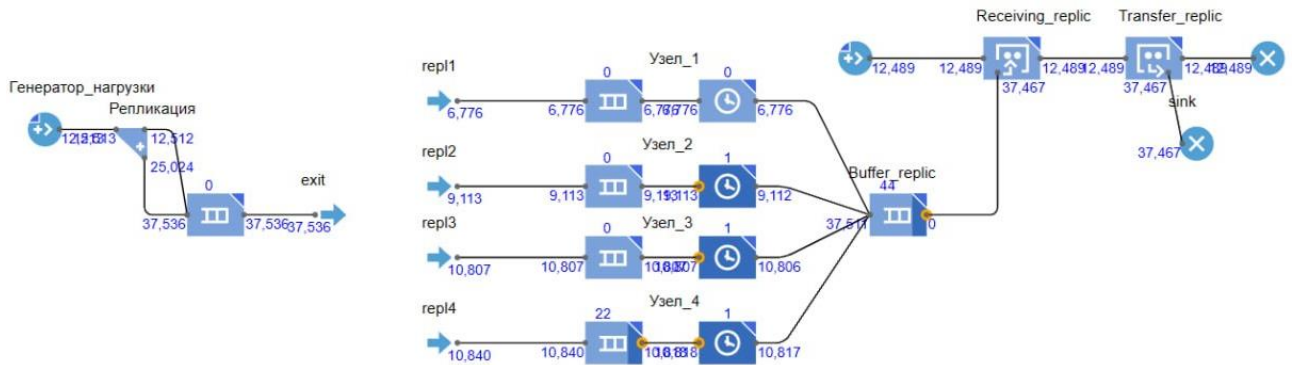


Рисунок 4.2 – Модель в программе AnyLogic

В разделе 3.1 множество узлов хранения, выбранных для размещения реплик на шаге управления k , было обозначено как $R(k)$. В настоящем разделе данное множество рассматривается как результат применения одной из сравниваемых стратегий выбора узлов: равномерного циклического распределения, выбора по минимальной пользовательской очереди, выбора с учетом пользовательской и служебной очередей либо алгоритма интегрального выбора узлов хранения.

Для операции записи с репликацией множество $R(k)$ включает R узлов хранения, на которые направляются реплики операции записи. Параметр R определяет число параллельных ветвей обслуживания. Операция записи считается завершенной после получения W подтверждений от выбранных узлов хранения, где $1 \leq W \leq R$. Параметр W задает условие завершения исходной пользовательской операции, то есть при $W = R$ реализуется строгий режим подтверждения, а при $W < R$ – пороговый режим подтверждения.

Время обслуживания операции записи на i -й ветви представляется как сумма сетевой и локальной составляющих (4.1):

$$T_i(k) = d_i(k) + s_i(k), \quad (4.1)$$

где $d_i(k)$ – задержка передачи данных к i -му узлу хранения и возврата подтверждения, $s_i(k)$ – время локального обслуживания операции записи на узле.

Локальное обслуживание включает прием данных, запись на носитель, выполнение внутренних процедур фиксации и формирование подтверждения.

С учетом состояния очередей локальное время обслуживания может быть представлено как функция пользовательской и служебной нагрузки (4.2):

$$s_i(k) = f(Q_i(k), B_i(k), \mu_i), \quad (4.2)$$

где $Q_i(k)$ – пользовательская очередь на i -м узле, $B_i(k)$ – служебная очередь, связанная с репликацией, синхронизацией и восстановлением данных, μ_i – интенсивность обслуживания узла.

Для расчетной оценки отдельная ветвь обслуживания может быть аппроксимирована системой массового обслуживания $M/M/1$. При условии устойчивости $\lambda_i < \mu_i$ среднее время пребывания заявки в ветви обслуживания записывается как (4.3):

$$\mathbb{E}[T_i] = d_i + \frac{1}{\mu_i - \lambda_i}, \quad (4.3)$$

где λ_i – интенсивность поступления операций на i -й узел. Данное выражение используется как приближенная оценка для сопоставления сценариев, а не как полное описание всех процессов распределенной подсистемы хранения.

Для сравнения с параллельной организацией записи рассматривается последовательная интерпретация операции, при которой обращения к выбранным узлам выполняются одно за другим (4.4):

$$T_{\text{seq}}(k) = \sum_{i \in R(k)} T_i^{(k)}. \quad (4.4)$$

При однородных ветвях обслуживания данное выражение может быть записано в виде (4.5):

$$T_{\text{seq}}^R \approx \frac{R}{\mu - \lambda}, \quad (4.5)$$

где λ – интенсивность входного потока, μ – интенсивность обслуживания одной ветви. Последовательная интерпретация используется только как базовый вариант сравнения, поскольку фактическая операция записи с репликацией выполняется параллельно.

В строгом режиме подтверждения $W=R$ операция записи завершается только после получения подтверждений от всех выбранных узлов, поэтому время завершения операции записи равно максимальному времени обслуживания среди ветвей, соответствующих выбранным узлам (4.6):

$$T_{\text{write}}(k) = \max_{i \in R(k)} T_i^{(k)}. \quad (4.6)$$

Если времена завершения обслуживания по ветвям рассматриваются как независимые случайные величины с функциями распределения $F_i(\tau)$, то функция распределения времени завершения операции записи в строгом режиме имеет вид (4.7)

$$F_{T_{\text{write}}}(\tau) = \prod_{i \in R(k)} F_i(\tau). \quad (4.7)$$

Из данного выражения следует, что увеличение задержки или накопление очереди хотя бы на одном из выбранных узлов может определять время завершения всей операции записи.

Для расчетной оценки среднего времени завершения операции записи в однородном случае используется аппроксимация среднего времени отклика Fork-Join-системы с R параллельными ветвями обслуживания (4.8):

$$T_{FJ}^{R \times N} \approx \frac{1}{\mu} \left(H_R + \frac{\rho}{2(1-\rho)} (S_{R,\rho} + (1-2\rho) \cdot S_{R,1-\rho}) \right), \quad (4.8)$$

где μ – интенсивность обслуживания одной ветви, ρ – загрузка ветви обслуживания, H_R – гармоническое число порядка R (4.9):

$$H_R = \sum_{j=1}^R \frac{1}{j}, \quad (4.9)$$

а $S_{R,\rho}$ и $S_{R,1-\rho}$ – вспомогательные суммы (4.10):

$$\begin{aligned} S_{R,\rho} &= \sum_{j=1}^R \frac{1}{j-\rho}, \\ S_{R,1-\rho} &= \sum_{j=1}^R \frac{1}{j-(1-\rho)}. \end{aligned} \quad (4.10)$$

Данная аппроксимация применяется для оценки параллельной организации записи при однородных ветвях обслуживания и используется как расчетная база для сопоставления с последовательной интерпретацией операции записи.

В пороговом режиме подтверждения $W < R$ операция завершается после получения W -го подтверждения из R возможных. Если времена завершения обслуживания по ветвям упорядочены по возрастанию (4.11):

$$T_{(1)}(k) \leq T_{(2)}(k) \leq \dots \leq T_{(R)}(k), \quad (4.11)$$

то время завершения операции записи определяется W -й порядковой статистикой $T_{\text{write}}(k) = T_{(W)}(k)$.

Для однородных ветвей с функцией распределения $F(\tau)$ функция распределения W -й порядковой статистики может быть записана как (4.12)

$$F_{T_{(W)}}(\tau) = \sum_{j=W}^R \binom{R}{j} F(\tau)^j (1 - F(\tau))^{R-j}. \quad (4.12)$$

Такое представление позволяет сопоставить строгий и пороговый режимы подтверждения. В строгом режиме время завершения операции определяется максимальным временем получения подтверждения от выбранных узлов, тогда как в пороговом режиме подтверждения отдельные ветви с увеличенным временем обслуживания могут не определять пользовательское время ответа. Вместе с тем они продолжают использовать ресурсы узлов и формировать служебную нагрузку, что учитывается через состояние $B_i(k)$.

Если число доступных узлов хранения превышает число формируемых реплик $N > R$, то выбор множества $R(k)$ становится самостоятельным механизмом управления временем записи. При равномерном циклическом распределении подмножества узлов выбираются без учета текущего состояния инфраструктуры. В этом случае средняя интенсивность поступления подзадач на один узел может быть оценена как (4.13)

$$\lambda_{\text{sub}} = \lambda \frac{R}{N}. \quad (4.13)$$

Для однородных узлов соответствующая загрузка ветви обслуживания определяется выражением (4.14)

$$\rho_{\text{sub}} = \frac{\lambda_{\text{sub}}}{\mu}. \quad (4.14)$$

Тогда для случая выбора R ветвей обслуживания из N доступных узлов расчетная аппроксимация среднего времени отклика Fork-Join-системы может быть записана в виде (4.15)

$$T_{FJ}^{R < N} \approx \frac{1}{\mu} \left(H_R + \frac{\rho_{\text{sub}}}{2(1 - \rho_{\text{sub}})} (S_{R, \rho_{\text{sub}}} + (1 - 2\rho_{\text{sub}})S_{R, 1 - \rho_{\text{sub}}}) \right). \quad (4.15)$$

При этом рассматривается устойчивый режим $\rho_{\text{sub}} < 1$.

Данное выражение используется как приближенная расчетная оценка для сценариев, в которых число доступных узлов превышает число реплик.

В предложенном алгоритме выбор множества $R(k)$ выполняется не по фиксированному циклическому правилу, а на основе интегральной оценки состояния узлов (3.5), рассчитываемой по параметрам состояния $S_i(k)$, для размещения реплик выбираются R узлов с минимальными значениями $C_i(k)$ и (3.7).

Тем самым модель операции записи связывается с алгоритмом главы 3: изменение множества $R(k)$ изменяет состав ветвей обслуживания и, следовательно, значение $T_{\text{write}}(k)$.

Для анализа неоднородной инфраструктуры рассматриваются сценарии, в которых отдельные ветви обслуживания отличаются по сетевой задержке или интенсивности обслуживания. В случае повышенной сетевой задержки для одного из узлов выполняется условие $d_s > d_f$, где d_s – задержка ветви с ухудшенными сетевыми характеристиками, d_f – задержка типовой ветви. Тогда время обслуживания такой ветви определяется как $T_s(k) = d_s + s_s(k)$, а для типовой ветви $T_f(k) = d_f + s_f(k)$.

В строгом режиме подтверждения включение ветви с повышенной задержкой в множество $R(k)$ может привести к увеличению времени всей операции записи, поскольку (4.16)

$$T_{\text{write}}(k) = \max\{T_s(k), T_{f_1}(k), \dots, T_{f_{R-1}}(k)\}. \quad (4.16)$$

Для сценария со сниженной интенсивностью обслуживания одного узла вводятся обозначения μ_s и μ_f , где μ_s – интенсивность обслуживания узла с ограниченной производительностью, μ_f – интенсивность обслуживания типового узла, причем $\mu_s < \mu_f$.

Для частного случая выбора одного узла со сниженной интенсивностью обслуживания и двух типовых узлов при $R = 3$ обобщенная оценка интенсивности обслуживания выбранного набора может быть представлена через гармоническое среднее (4.17):

$$\mu_{\text{eff}}^{(3)} = \frac{3}{\frac{1}{\mu_s} + \frac{2}{\mu_f}}. \quad (4.17)$$

В общем виде для одной ветви со сниженной интенсивностью обслуживания и $R - 1$ типовых ветвей (4.18):

$$\mu_{\text{eff}}^{(R)} = \frac{R}{\frac{1}{\mu_s} + \frac{R-1}{\mu_f}}. \quad (4.18)$$

Данное выражение используется как расчетная оценка для сопоставления конфигураций, в которых дополнительный узел имеет ухудшенные характеристики обслуживания. В рамках эксперимента оценивается, всегда ли узел со сниженной производительностью должен исключаться из множества $R(k)$ либо при высокой нагрузке его включение может сократить время записи за счет распределения потока между большим числом ветвей обслуживания.

Для описания выбора узлов используется принцип уравнивания загрузки ветвей обслуживания. Загрузка i -й ветви определяется как (4.19)

$$\rho_i(k) = \frac{\lambda_i(k)}{\mu_i}. \quad (4.19)$$

В простейшем виде условие уравнивания загрузки между выбранными ветвями можно записать как (4.20)

$$\rho_i(k) \approx \rho_j(k), i, j \in R(k). \quad (4.20)$$

Эквивалентно (4.21):

$$\frac{\lambda_i(k)}{\mu_i} \approx \frac{\lambda_j(k)}{\mu_j}. \quad (4.21)$$

Данное условие отражает стремление алгоритма снизить вероятность включения в операцию записи перегруженной ветви обслуживания и перераспределить поток в пользу более благоприятных узлов.

Для определения области, в которой включение узла со сниженной интенсивностью обслуживания становится оправданным, вводится критическое значение интенсивности входного потока λ_{crit} , которое определяется из условия равенства расчетных времен записи для сравниваемых конфигураций (4.22):

$$T_{\text{write}}^{(N=3, R=3)}(\lambda_{\text{crit}}) = T_{\text{write}}^{(N=4, R=3, \mu_s)}(\lambda_{\text{crit}}). \quad (4.22)$$

При $\lambda < \lambda_{\text{crit}}$ включение узла со сниженной интенсивностью обслуживания преимущественно увеличивает время записи, поскольку такая ветвь обслуживания имеет повышенную вероятность стать определяющей для времени завершения операции. При $\lambda > \lambda_{\text{crit}}$ его включение может снижать время записи за счет разгрузки остальных узлов и распределения входного потока между большим числом ветвей обслуживания.

Для количественного сопоставления базовой и исследуемой стратегий используется относительное изменение времени записи (4.23):

$$G_T(s) = \frac{T_{RR} - T_s}{T_{RR}} \times 100\%, \quad (4.23)$$

где T_{RR} – время завершения операции записи при равномерном циклическом распределении, T_s – время завершения операции записи при использовании сравниваемой стратегии s . В качестве s могут рассматриваться выбор по минимальной пользовательской очереди, выбор с учетом пользовательской и служебной очередей либо предложенный алгоритм интегрального выбора узлов хранения [69]. Положительное значение $G_T(s)$ соответствует снижению времени записи по сравнению с равномерным циклическим распределением, отрицательное – его увеличению.

4.3 Методика экспериментального исследования

Экспериментальное исследование строится как последовательная проверка расчетной модели операции записи и алгоритма выбора узлов хранения при различных условиях функционирования распределенной подсистемы хранения туманного уровня. В отличие от общей методики сопоставительного эксперимента, сформированной в разделе 3.5, в настоящем разделе задаются конкретные сценарии, используемые для получения результатов главы 4.

В сценариях, направленных на оценку выбора множества $R(k)$, сопоставляются четыре стратегии формирования множества узлов хранения:

- равномерное циклическое распределение,
- выбор узлов по минимальной пользовательской очереди,
- выбор узлов с учетом пользовательской и служебной очередей,
- алгоритм интегрального выбора узлов хранения.

Все стратегии проверяются при одинаковых параметрах входного потока, числа доступных узлов, числа реплик, режима подтверждения и характеристик ветвей обслуживания. Различие между вариантами состоит только в составе

информации, используемой при выборе узлов. Экспериментальная проверка включает следующие сценарии (таблица 4.2).

Таблица 4.2 – Сценарии исследования

Сценарий	Изменяемый фактор	Назначение
Последовательная и параллельная запись	способ представления операции записи	обоснование представления операции записи на основе модели Fork-Join
Строгий режим подтверждения	$W=R$	оценка влияния повышенной задержки в одной из ветвей обслуживания
Пороговый режим подтверждения	$W < R$	анализ влияния порогового подтверждения на время завершения операции
Выбор R узлов из $N > R$	состав подмножества $R(k)$	оценка влияния выбора подмножества узлов на время записи
Неоднородность сетевых задержек	увеличение $d_i(k)$ для одной или нескольких ветвей	оценка влияния неоднородной задержки на одном из направлений связи на время записи и область перегрузки
Частичное снижение производительности одного узла	уменьшение μ_i для одного узла	анализ промежуточного состояния между штатной работой и отказом узла

Во всех сценариях операция записи рассматривается как процесс формирования нескольких параллельных ветвей обслуживания с последующими подтверждениями. Основное внимание уделяется тому, как изменение состава выбранных узлов влияет на время завершения операции записи при различных значениях R , W , N , сетевой задержки и интенсивности обслуживания.

Последовательность сценариев выбрана таким образом, чтобы сначала подтвердить корректность параллельной интерпретации операции записи, затем оценить влияние режима подтверждения, после чего перейти к проверке собственно алгоритма выбора узлов хранения. Сценарии неоднородной задержки и сниженной производительности узла используются для определения условий, при которых выбор узлов по алгоритму интегрального выбора дает наибольший эффект по сравнению с равномерным распределением. Опыт проведения экспериментальных исследований, тестирования и обобщения результатов для

программно-определяемых систем хранения данных, полученный в рамках заключительных этапов НИР, использовался при формировании методического подхода к экспериментальной проверке в настоящей работе [141,142].

4.4 Результаты экспериментальной оценки

Результаты экспериментальной оценки представлены в виде зависимостей времени завершения операции записи от интенсивности входного потока, режима подтверждения, стратегии выбора узлов хранения, неоднородности сетевых задержек и интенсивности обслуживания отдельных узлов. Сопоставление выполнялось для равномерного циклического распределения, выбора по минимальной пользовательской очереди, стратегии с учетом пользовательской и служебной очередей, а также предложенного алгоритма, учитывающего состояние очередей и сетевую составляющую. Экспериментальная оценка влияния репликации и режима согласованности на производительность распределенных хранилищ согласуется с работами [143], в которых сравниваются стратегии репликации и согласованности на примере HBase и Cassandra.

На первом этапе сопоставлены последовательная и параллельная интерпретации операции записи. Последовательный вариант рассматривается как базовый вариант сравнения, при которой обращения к узлам хранения выполняются одно за другим. Параллельный вариант соответствует Fork-Join-представлению, при котором передача реплик, локальная запись и возврат подтверждений выполняются одновременно по нескольким ветвям обслуживания.

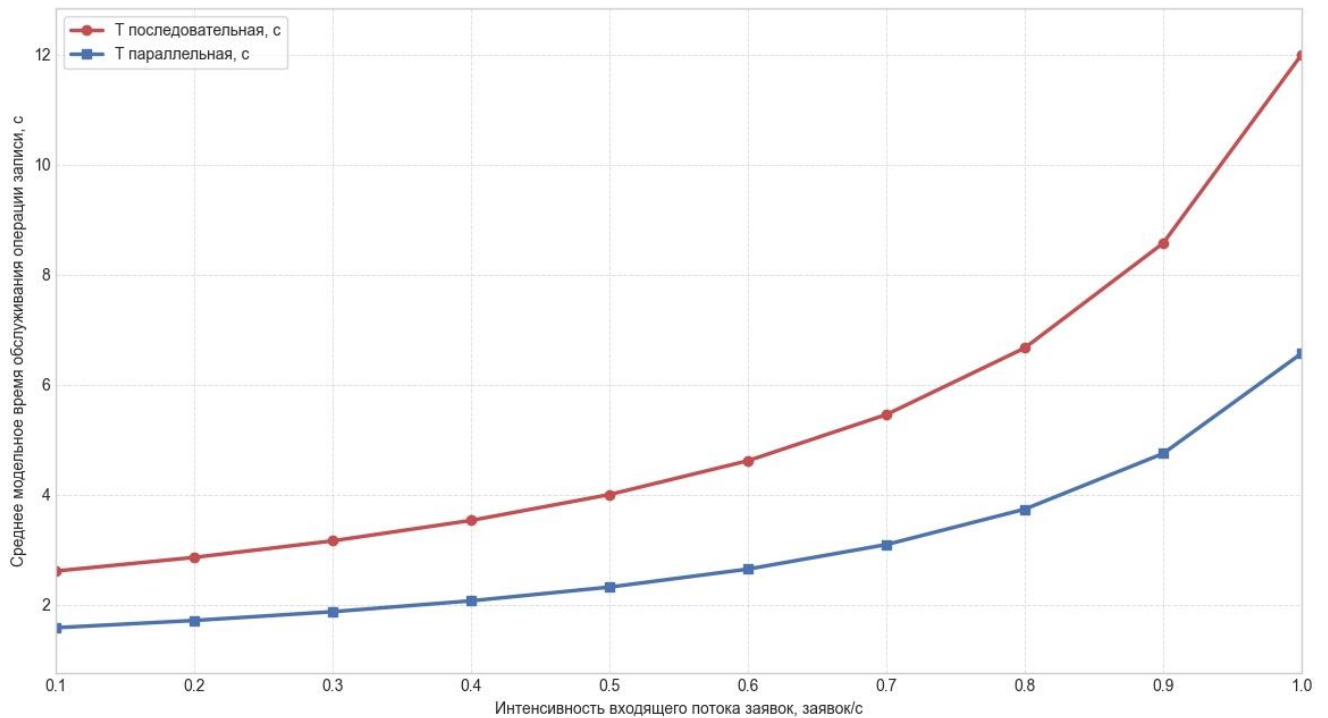


Рисунок 4.6 – Сравнение последовательной и параллельной организации операции записи

Полученные результаты показывают, что последовательная интерпретация приводит к завышенной оценке времени записи. В последовательном случае время выполнения операции определяется суммой времен обслуживания выбранных ветвей (4.4), тогда как при параллельной организации в строгом режиме подтверждения операция завершается после получения подтверждений от всех выбранных узлов, поэтому время записи определяется максимальным временем обслуживания среди выбранных ветвей (4.24):

$$T_{\text{write}}(k) = \max_{i \in R(k)} T_i(k). \quad (4.24)$$

Следовательно, для дальнейшего анализа операции записи с репликацией целесообразно использовать параллельное представление с получением подтверждений.

Для проверки применимости расчетной модели выполнено сопоставление расчетных (ММ в легенде графика) и имитационных (ИМ в легенде графика)

значений времени записи при одинаковых параметрах входного потока и ветвей обслуживания.

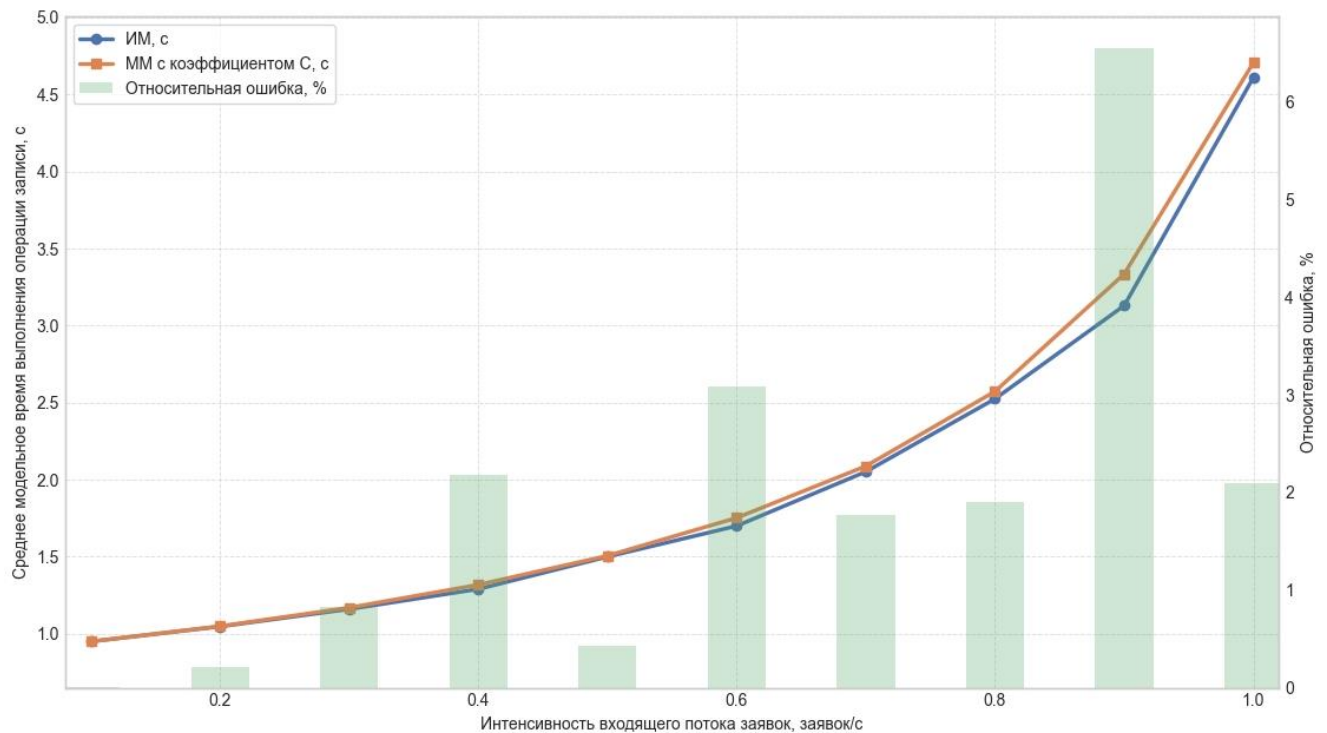


Рисунок 4.7 – Сопоставление расчетной и имитационной оценок времени записи

Результаты показывают согласованность расчетной и имитационной моделей в области низкой и средней загрузки. При приближении к области высокой загрузки расхождение возрастает, что связано с нелинейным накоплением очередей и усилением влияния случайных задержек обслуживания. При этом расчетная модель сохраняет применимость для сопоставительной оценки сценариев, поскольку в настоящем исследовании основное значение имеют не абсолютные значения времени записи, а относительные изменения при изменении стратегии выбора узлов.

Далее выполнено сопоставление стратегий при росте интенсивности входного потока. Для количественной оценки изменения времени записи используется показатель G_T по (4.23).

При малой интенсивности входного потока различия между стратегиями выражены слабо, поскольку очереди на узлах хранения не оказывают

существенного влияния на время обслуживания, а вероятность выбора перегруженной ветви остается низкой. В этих условиях равномерное распределение может давать результаты, близкие к стратегиям, использующим текущее состояние системы.

При увеличении интенсивности входного потока различия между стратегиями становятся более заметными. Равномерное распределение приводит к более раннему накоплению очередей на отдельных узлах, поскольку не учитывает текущее состояние инфраструктуры. Выбор по минимальной пользовательской очереди снижает время записи относительно равномерного распределения, однако его эффект ограничен отсутствием учета служебной нагрузки и сетевой составляющей. Стратегия, учитывающая одновременно $Q_i(k)$ и $B_i(k)$, демонстрирует более устойчивое поведение при наличии фоновых операций. Наиболее устойчивые результаты показывает предложенный алгоритм интегрального выбора узлов, поскольку при выборе множества $R(k)$ учитываются пользовательская очередь, служебная нагрузка и задержка ветви обслуживания.

Следующий сценарий связан с анализом строгого и порогового режимов подтверждения записи. В строгом режиме $W = R$ операция завершается после получения подтверждений от всех выбранных узлов. В пороговом режиме $W < R$ операция завершается после получения W -го подтверждения, то есть $T_{\text{write}}(k) = T_{(W)}(k)$.

Результаты показывают (рисунок 4.9), что строгий режим подтверждения обладает большей чувствительностью к локальным перегрузкам и неоднородности сетевых задержек. В данном режиме даже одна ветвь с повышенным временем обслуживания может определять время завершения всей операции. Пороговый режим снижает влияние ветвей с наибольшим временем завершения на пользовательское время ответа, однако эти ветви продолжают использовать ресурсы узлов и формировать служебную нагрузку. Поэтому снижение времени ответа в пороговом режиме должно интерпретироваться совместно с состоянием служебных очередей $B_i(k)$.

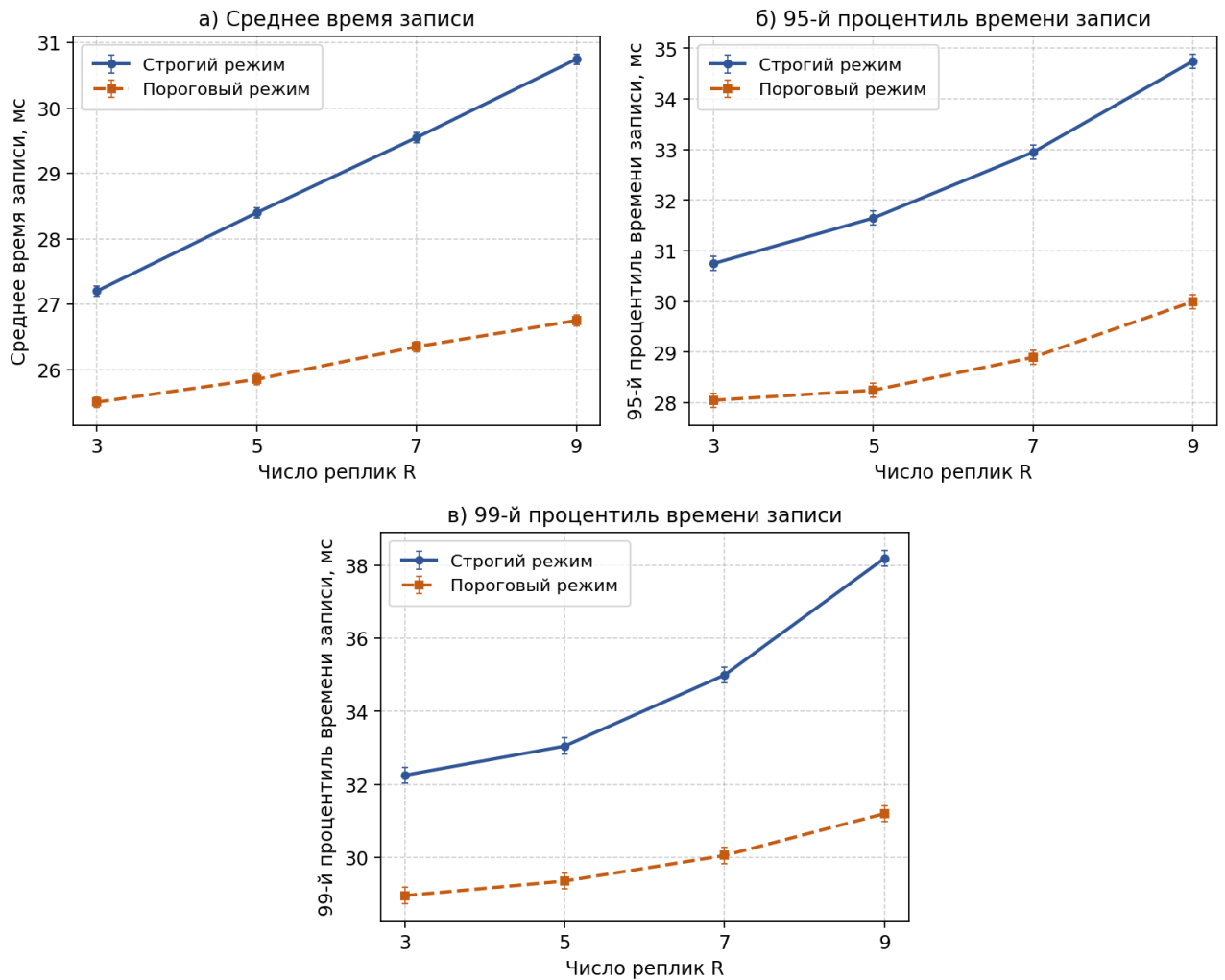


Рисунок 4.9 – Сравнение времени записи при строгом и пороговом режимах подтверждения

Далее рассмотрен сценарий выбора R узлов из множества $N > R$. Данный сценарий является ключевым для проверки алгоритма, поскольку при наличии избыточного множества доступных узлов стратегия выбора подмножества реплик становится самостоятельным механизмом управления временем записи.

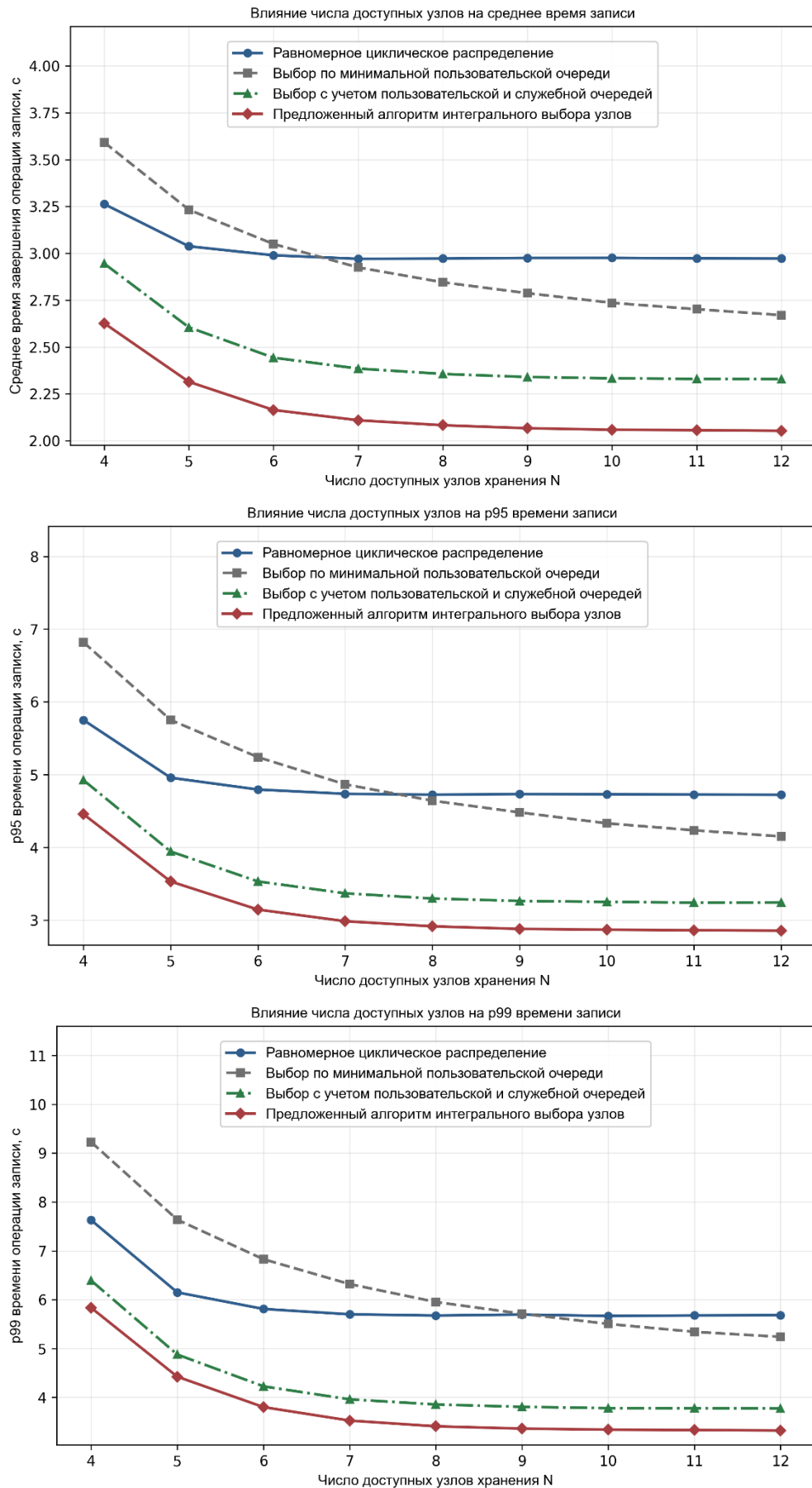


Рисунок 4.10 – Влияние выбора R узлов из $N > R$ на время завершения операции записи при различных стратегиях

При равномерном циклическом распределении подмножества узлов выбираются без учета текущего состояния инфраструктуры. Поэтому узлы с повышенной пользовательской очередью, служебной нагрузкой или сетевой задержкой могут регулярно включаться в состав ветвей обслуживания. Выбор по минимальной пользовательской очереди уменьшает вероятность включения узлов с высокой $Q_i(k)$, но не исключает выбор узлов, занятых служебными операциями или имеющих повышенную сетевую задержку. Предложенный алгоритм интегрального выбора узлов формирует множество $R(k)$ на основе интегральной оценки выбора $C_i(k)$, рассчитываемой по параметрам состояния узлов, поэтому снижает вероятность включения неблагоприятных ветвей в операцию записи.

Особое значение имеет сценарий неоднородности сетевых задержек. В данном случае одна из ветвей обслуживания характеризуется увеличенной задержкой передачи данных и возврата подтверждений $d_s > d_f$.

В строгом режиме включение такой ветви в множество выбранных узлов приводит к увеличению времени всей операции по (4.16).

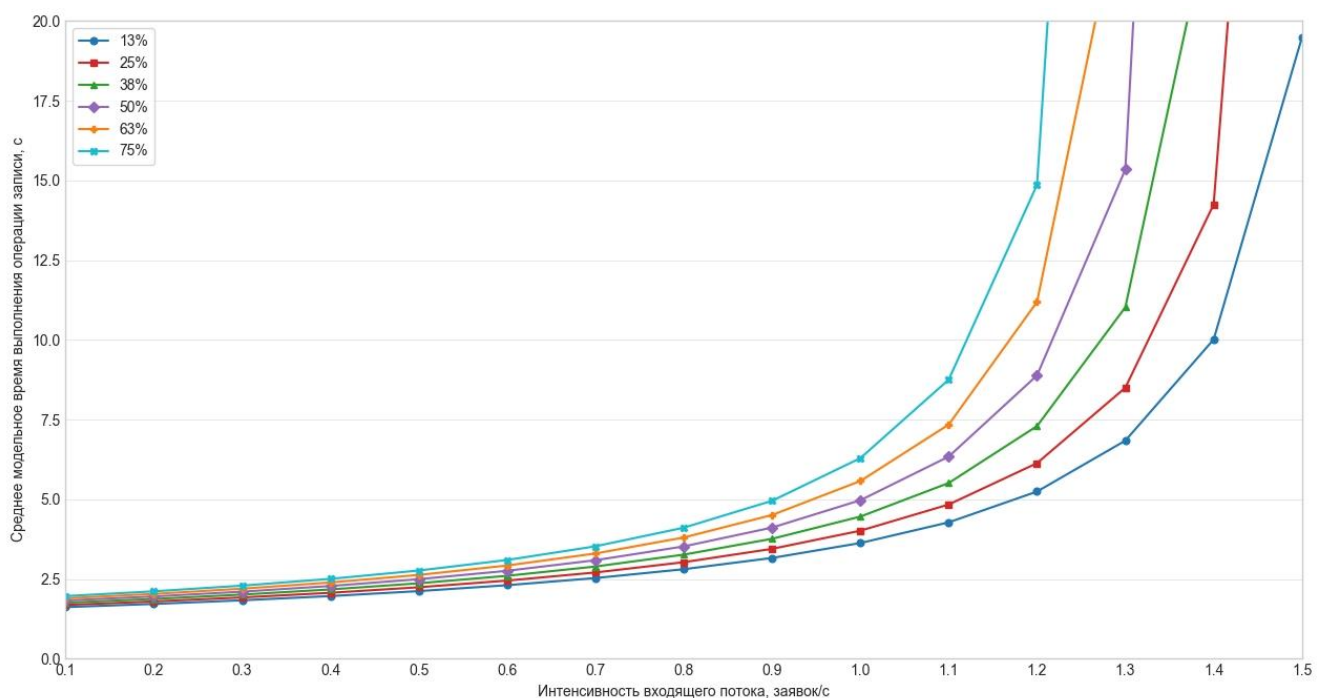


Рисунок 4.11 – Влияние повышенной сетевой задержки в одной из ветвей обслуживания при равномерном циклическом распределении

Результаты для равномерного распределения показывают, что увеличение задержки в одной из ветвей приводит к более раннему переходу системы в область резкого роста времени записи. При низкой нагрузке различия между вариантами задержки выражены умеренно, однако при росте входного потока влияние неблагоприятной ветви усиливается.

Для сопоставления стратегий дополнительно рассматривается влияние неоднородной задержки при разных правилах выбора узлов.

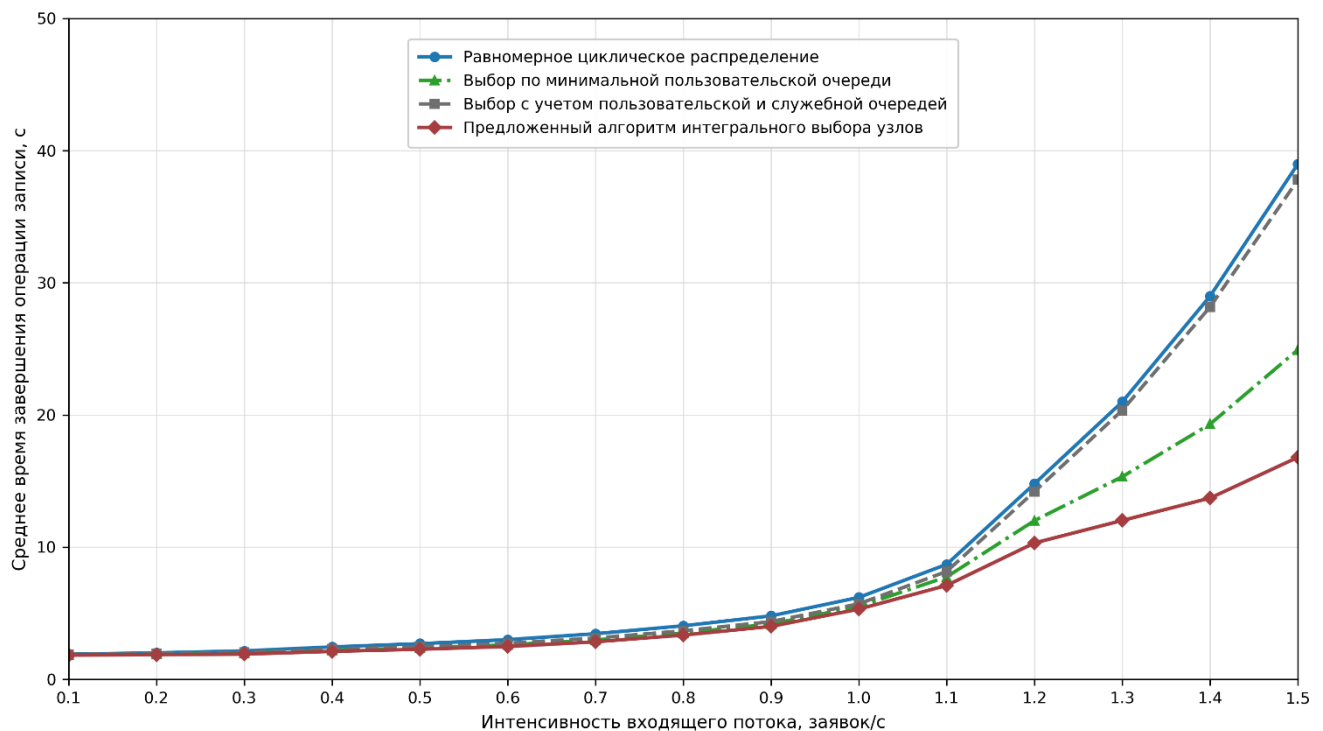


Рисунок 4.12 – Сравнение стратегий распределения операций записи при увеличении задержки одной ветви обслуживания на 75 %

Равномерное распределение наиболее чувствительно к увеличению задержки на отдельном направлении связи. Стратегия выбора по минимальной пользовательской очереди не всегда устраняет этот эффект, поскольку узел с повышенной сетевой задержкой может иметь малую очередь и поэтому попадать в множество выбранных реплик. Предложенный алгоритм снижает чувствительность операции записи к сетевой неоднородности за счет учета $d_i(k)$

при формировании $R(k)$. Наибольший эффект проявляется при наличии альтернативных узлов хранения, когда $N > R$.

Следующий сценарий связан с влиянием служебной нагрузки. В этом случае часть узлов характеризуется повышенным значением $B_i(k)$, обусловленным репликацией, синхронизацией, восстановлением или перераспределением данных.

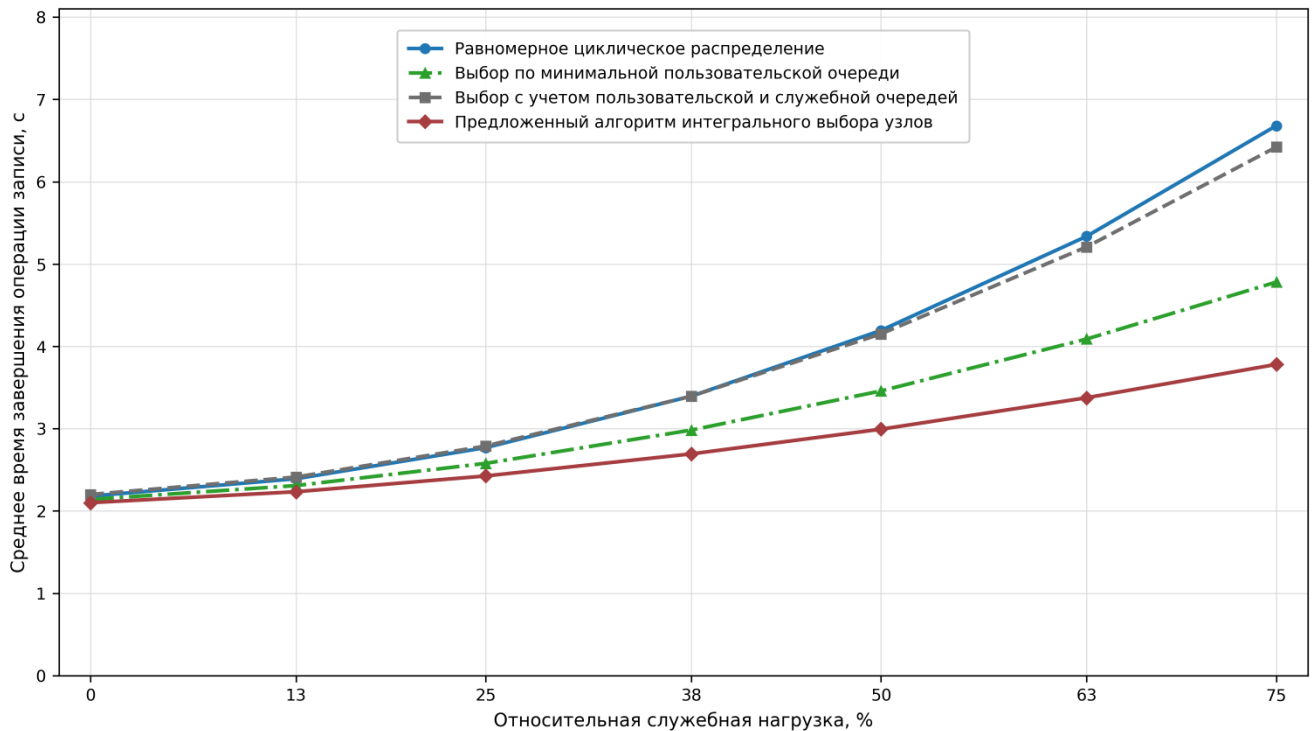


Рисунок 4.13 – Влияние служебной нагрузки на время завершения операции записи при различных стратегиях выбора узлов при $\lambda = 1,0$ заявок/с

Равномерное распределение не реагирует на рост служебной нагрузки и может направлять реплики на узлы, значительная часть ресурсов которых занята фоновыми процессами. Выбор по минимальной пользовательской очереди также не полностью устраняет данный эффект, поскольку узел может иметь малую пользовательскую очередь, но высокую служебную нагрузку. Учет $B_i(k)$ позволяет снизить вероятность выбора таких узлов. Предложенный алгоритм интегрального выбора узлов обеспечивает более сбалансированное поведение, поскольку при выборе узлов учитывает одновременно пользовательскую очередь, служебную нагрузку и сетевую составляющую.

Отдельно рассмотрен сценарий частичного снижения производительности одного из узлов хранения. В отличие от полного отказа, такой узел сохраняет возможность обслуживания операций, но характеризуется меньшей интенсивностью обслуживания $\mu_s < \mu_f$.

Для оценки влияния такого узла используются расчетные времена записи для конфигураций с тремя типовыми узлами и с четырьмя узлами, один из которых имеет сниженную производительность. Граница смены характера влияния определяется условием (4.25)

$$T_{FJ}^{(3)}(\lambda_{\text{crit}}) = T_{FJ}^{(4)}(\lambda_{\text{crit}}, \mu_s). \quad (4.25)$$

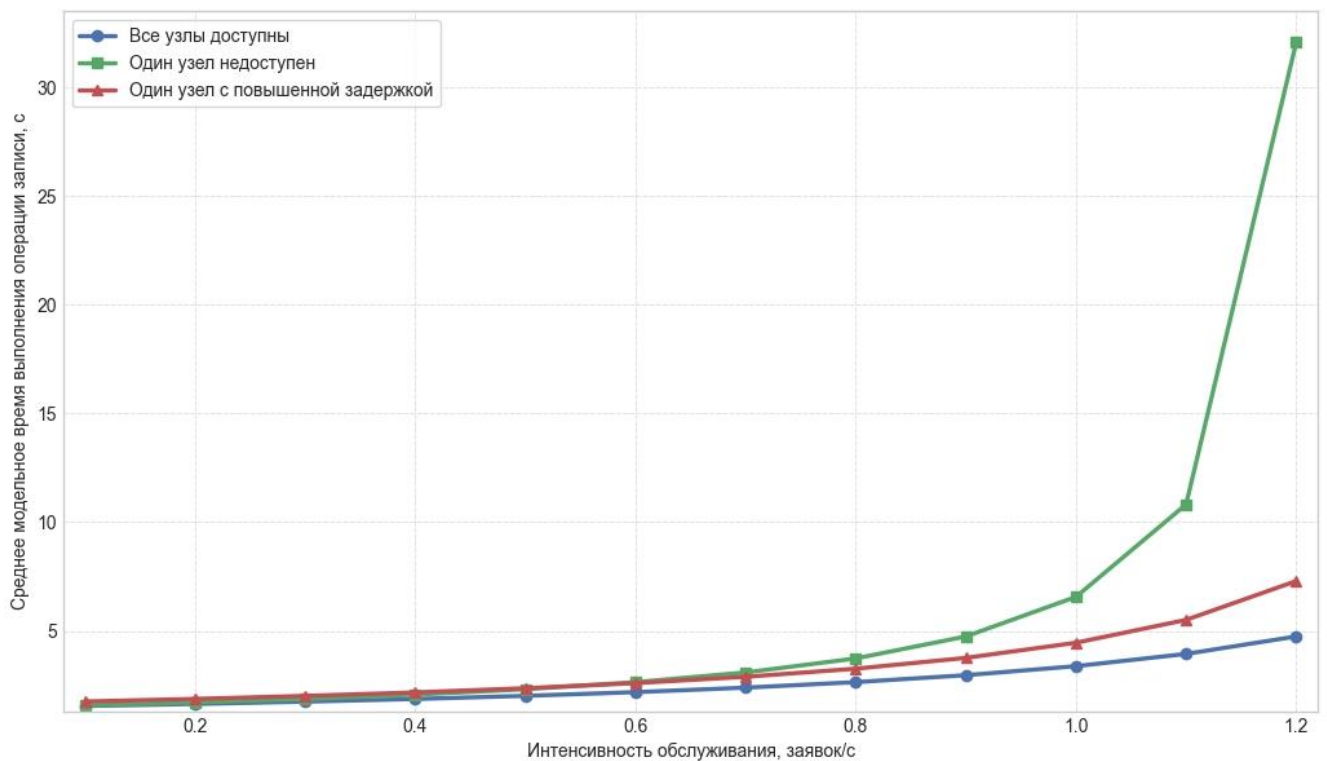


Рисунок 4.14 – Влияние исключения узла и снижения интенсивности обслуживания одного узла на среднее время выполнения операции записи

Полученные результаты показывают, что узел со сниженной интенсивностью обслуживания не всегда должен исключаться из множества доступных ресурсов. При низкой входной нагрузке его включение может увеличивать время записи, поскольку ветвь обслуживания, связанная с таким узлом, чаще определяет время

завершения операции. При высокой нагрузке дополнительный узел может снижать время записи за счет перераспределения входного потока и разгрузки типовых узлов.

Значение λ_{crit} определяет границу, при которой эффект разгрузки типовых узлов за счет включения дополнительной ветви становится сопоставимым с ухудшением времени обслуживания, вызванным сниженной производительностью одного из узлов. При $\lambda < \lambda_{crit}$ включение такого узла преимущественно ухудшает временные характеристики, а при $\lambda > \lambda_{crit}$ может быть оправдано за счет перераспределения нагрузки между большим числом ветвей обслуживания.

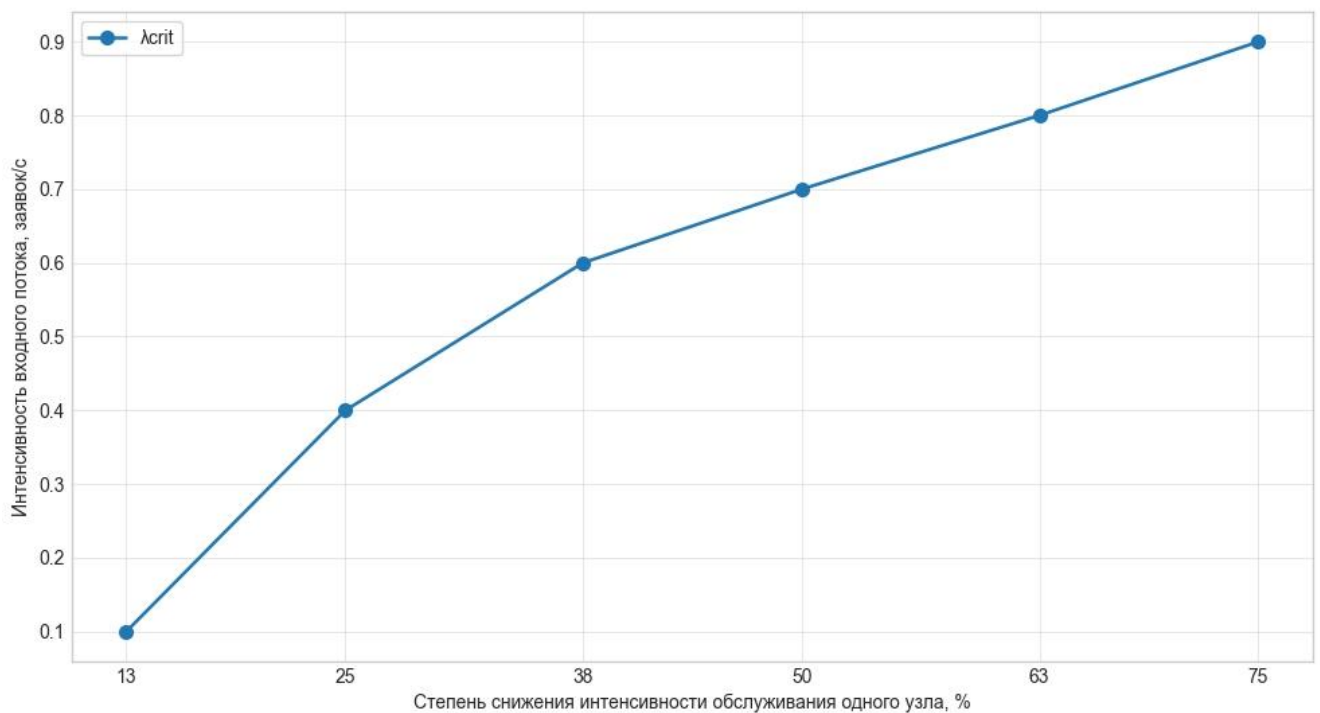


Рисунок 4.15 – Зависимость интенсивности входного потока от степени снижения интенсивности обслуживания одного узла

Сопоставление стратегий по совокупности сценариев показывает, что преимущество предложенного алгоритма проявляется не во всех режимах одинаково. При низкой нагрузке и однородных характеристиках узлов различия между стратегиями незначительны. В таких условиях усложнение правила выбора узлов не приводит к существенному выигрышу, поскольку отсутствуют

выраженные локальные перегрузки и ветви обслуживания с повышенной задержкой либо сниженной интенсивностью обслуживания.

Наибольший эффект достигается при росте нагрузки, накоплении служебных операций, неоднородности сетевых задержек и наличии альтернативных узлов хранения. В этих режимах алгоритм интегрального выбора узлов уменьшает вероятность включения в операцию записи перегруженных или неблагоприятных ветвей обслуживания, что приводит к снижению среднего времени записи и верхних процентилей задержки.

Обобщение поведения стратегий в экспериментальных сценариях приведено в таблице 4.3. Результаты экспериментальной оценки подтверждают, что выбор узлов хранения является значимым механизмом управления временными характеристиками операции записи с репликацией данных. Равномерное распределение может использоваться как простая базовая стратегия в однородных условиях, однако при неоднородности нагрузки и сетевых задержек оно не обеспечивает устойчивого поведения времени записи. Алгоритм интегрального выбора узлов обеспечивает более согласованный выбор ветвей обслуживания за счет учета пользовательской очереди, служебной нагрузки и сетевой составляющей.

Таблица 4.3 – Обобщенная оценка поведения стратегий в экспериментальных сценариях

Стратегия	Низкая нагрузка	Высокая нагрузка	Служебная нагрузка	Неоднородные задержки	Узел со сниженной интенсивностью обслуживания
Равномерное циклическое распределение	результаты близки к другим стратегиям	ранний рост очередей	не учитывается	высокая чувствительность	может регулярно выбирать неблагоприятную ветвь
Выбор по минимальной пользовательской очереди	снижает локальную пользовательскую перегрузку	лучше равномерного распределения	не учитывает $B_i(k)$	не учитывает $d_i(k)$	эффект зависит от текущей очереди
Выбор с учетом пользовательской и служебной очередей	эффект умеренный	снижает влияние служебной активности	учитывает $B_i(k)$	не учитывает сетевую составляющую	эффект зависит от загрузки узла
Предложенный алгоритм интегрального выбора узлов	эффект ограничен	снижает рост времени записи	учитывает $B_i(k)$	снижает чувствительность к $d_i(k)$	учитывает промежуточное состояние узла

4.4.1 Оценка допустимой интенсивности входного потока

Для количественного подтверждения влияния стратегий выбора узлов хранения на область допустимых нагрузок выполнен расчет допустимой интенсивности входного потока $\lambda_{\text{доп}}$. Под допустимой интенсивностью понимается последняя проверенная точка входной нагрузки, при которой сохраняется выполнение заданного критерия допустимости режима обслуживания.

В качестве критерия допустимости использовалось одновременное выполнение трех условий:

- $\bar{T}_{\text{write}} \leq 1550$ мс для ограничения среднего времени завершения операции записи,
- $p99(T_{\text{write}}) \leq 3000$ мс (значение 99-го перцентиля задержки),
- дополнительно точка нагрузки признавалась допустимой только при условии, что для каждой операции записи в рамках имитационного эксперимента могло быть сформировано множество из R узлов хранения, удовлетворяющих ограничению по готовности $A_i(k) \geq A_{\min}$.

Значение 1550 мс используется в эксперименте как модельное ограничение среднего времени завершения операции записи, характеризующее принятый в исследовании профиль требований к производительности, и не рассматривается как универсальное нормативное требование.

Операция записи моделировалась как кворумная: из $R = 3$ выбранных реплик завершение операции определялось получением $W = 2$ подтверждений.

Для каждой точки нагрузки и каждой стратегии выполнялось 30 независимых реализаций имитационного эксперимента. В одной реализации моделировалось 10 000 операций записи, поэтому на одну точку графика по одной стратегии приходилось 300 000 смоделированных операций записи.

Результаты оценки допустимой интенсивности входного потока приведены в таблице 4.4.

Таблица 4.4 – Фрагмент оценки допустимой интенсивности входного потока при строгом критерии допустимости

Стратегия выбора узлов хранения	Учитываемые признаки состояния	$\lambda_{\text{доп}}$, заявок/с	$\bar{T}_{\text{write}}$ при $\lambda_{\text{доп}}$, мс	p_{99} при $\lambda_{\text{доп}}$, мс	Прирост относительно RR, %
Равномерное циклическое распределение	Текущее состояние узлов не учитывается	1,45	1542,49	2973,78	-
Выбор по минимальной пользовательской очереди	$Q_i(k)$	2,05	1290,37	2953,75	41,4
Выбор с учетом пользовательской и служебной очередей	$Q_i(k), B_i(k)$	2,10	1292,99	2982,12	44,8
Предложенный алгоритм интегрального выбора узлов	$Q_i(k), B_i(k), d_i(k), A_i(k) \geq A_{\min}$	$\geq 2,15$	1254,21	2991,68	не менее 48,3

Из данных таблицы 4.4 следует, что равномерное циклическое распределение достигает границы допустимого режима при $\lambda_{\text{доп}} = 1,45$ заявок/с. Стратегии, использующие информацию о состоянии узлов, обеспечивают расширение области допустимых нагрузок. Для стратегии с учетом пользовательской и служебной очередей последняя допустимая точка составляет 2,10 заявок/с, а для предложенного алгоритма интегрального выбора узлов – 2,15 заявок/с.

Графическое отображение полученных зависимостей представлено на рисунке 4.16.

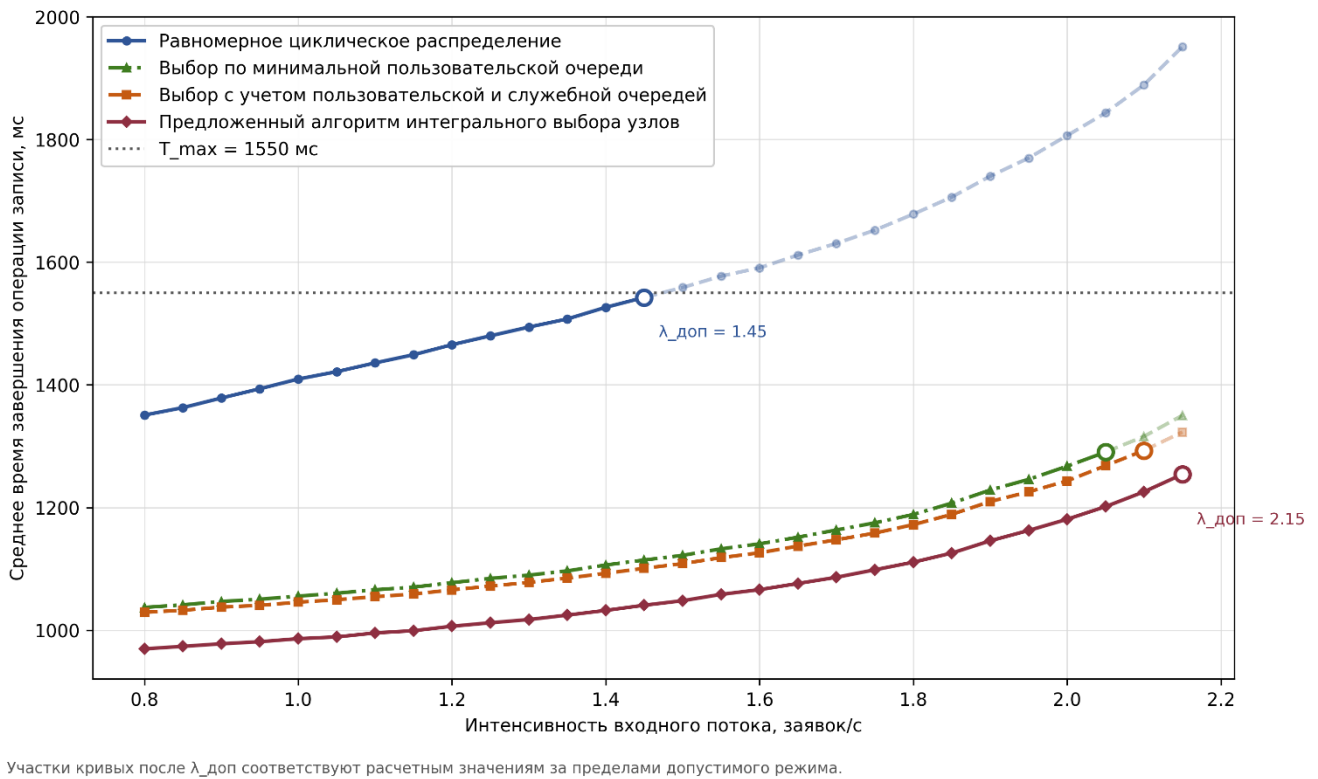


Рисунок 4.16 – Зависимость среднего времени завершения операции записи от интенсивности входного потока при строгом критерии допустимости

Как видно из рисунка 4.16, при увеличении интенсивности входного потока равномерное циклическое распределение быстрее достигает границы допустимого режима. Для предложенного алгоритма интегрального выбора узлов рост среднего времени завершения операции записи остается более пологим в исследованном диапазоне нагрузок, что обеспечивает увеличение $\lambda_{\text{доп}}$ до 2.15 заявок/с.

Относительный прирост допустимой интенсивности входного потока по сравнению с равномерным циклическим распределением определяется выражением (2.18). С учетом полученных значений:

$$E_{\lambda} = \frac{2,15 - 1,45}{1,45} \times 100\% = 48,3 \, \%.$$

Следовательно, предложенный алгоритм интегрального выбора узлов позволяет увеличить допустимую интенсивность входного потока операций записи на 48,3% по сравнению с равномерным циклическим распределением при

соблюдении ограничений на среднее время завершения операции записи, значение 99-го перцентиля задержки и готовность выбранного множества узлов.

Дополнительное преимущество предложенного алгоритма относительно стратегии, учитывающей пользовательскую и служебную очереди, составляет:

$$E_{\lambda} = \frac{2,15 - 2,10}{2,10} \times 100\% = 2,4 \, \%.$$

Полученный результат показывает, что основной вклад в расширение области допустимых нагрузок обеспечивает учет состояния узлов, а сетевая составляющая в предложенном алгоритме выполняет роль дополнительного уточняющего фактора при выборе множества реплик. При этом предложенный алгоритм сохраняет наибольшее значение $\lambda_{\text{доп}}$ среди рассмотренных стратегий в заданной конфигурации эксперимента.

4.5 Анализ результатов и область применимости алгоритма

Результаты экспериментальной оценки показывают, что стратегия выбора узлов хранения оказывает существенное влияние на временные характеристики операции записи с репликацией данных. При этом эффект от применения алгоритма проявляется не во всех режимах одинаково: в однородных условиях и при малой входной нагрузке различия между стратегиями минимальны, тогда как при росте нагрузки, появлении служебных операций и неоднородности сетевых задержек преимущество стратегий, использующих информацию о состоянии узлов, становится более выраженным.

Равномерное циклическое распределение демонстрирует приемлемые результаты только при относительно стабильном состоянии узлов и близких значениях сетевых задержек. Основное ограничение данной стратегии состоит в отсутствии реакции на текущее состояние инфраструктуры. В результате при накоплении очередей или ухудшении характеристик одной из ветвей

обслуживания равномерное распределение продолжает включать неблагоприятные узлы в операции записи, что приводит к росту времени завершения операции и более раннему достижению границы допустимого режима.

Выбор по минимальной пользовательской очереди улучшает результаты по сравнению с равномерным распределением, поскольку снижает вероятность назначения новых операций на уже перегруженные узлы. Однако данная стратегия учитывает только одну составляющую состояния системы. Малая пользовательская очередь не всегда означает предпочтительность узла: он может быть занят служебными операциями восстановления, синхронизации или перераспределения данных либо находиться в сегменте сети с повышенной задержкой. Поэтому учет только $Q_i(k)$ недостаточен для устойчивого управления записью в инфраструктуре туманных вычислений.

Учет служебной нагрузки позволяет точнее оценивать фактическую доступность обслуживающего ресурса узла для пользовательских операций, что особенно важно для распределенной подсистемы хранения, где служебные процессы используют те же вычислительные, дисковые и коммуникационные ресурсы, что и операции записи. Полученные результаты подтверждают, что игнорирование служебной составляющей может приводить к выбору узлов, формально свободных по пользовательской очереди, но фактически занятых фоновыми процессами.

Наиболее полное правило выбора реализует предложенный алгоритм интегрального выбора узлов, поскольку он объединяет несколько признаков состояния инфраструктуры: пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности. За счет этого алгоритм снижает вероятность включения в множество реплик узлов, которые могут стать неблагоприятными ветвями обслуживания. Данный эффект особенно значим при наличии избыточного множества доступных узлов, когда $N > R$, поскольку в этом случае существует реальная свобода выбора состава $R(k)$.

Отдельное значение имеет учет неоднородности сетевых задержек. Для инфраструктуры туманных вычислений данный фактор является существенным,

поскольку узлы хранения могут размещаться в разных сегментах сети, а межузловой обмен входит в цепочку обслуживания операции записи. Стратегия, не учитывающая сетевую составляющую, может выбирать узлы с повышенной задержкой даже при низкой пользовательской очереди. Поэтому включение сетевого признака в критерий выбора узлов является обоснованным с точки зрения модели обслуживания операции записи и учета телекоммуникационной составляющей межузлового обмена.

Расчет допустимой интенсивности входного потока $\lambda_{\text{доп}}$ обобщает результаты сопоставления стратегий в предельной области нагрузок. В отличие от отдельных сценариев, где анализируется влияние числа доступных узлов, сетевой неоднородности или состояния отдельных ветвей обслуживания, расчет $\lambda_{\text{доп}}$ показывает границу режима, в котором одновременно выполняются ограничения на среднее время завершения операции записи, значение p_{99} и готовность выбранного множества узлов. Поэтому данный расчет является наиболее строгой проверкой влияния стратегии выбора узлов на область допустимых нагрузок.

Согласно полученным результатам, равномерное циклическое распределение достигает границы допустимого режима при $\lambda_{\text{доп}} = 1,45$ заявок/с. Для стратегии, учитывающей пользовательскую и служебную очереди, последняя допустимая точка составляет 2,10 заявок/с, а для предложенного алгоритма интегрального выбора узлов заявок/с. Следовательно, относительный прирост допустимой интенсивности входного потока для предложенного алгоритма по сравнению с равномерным циклическим распределением составляет 48,3%. Дополнительное преимущество относительно стратегии, учитывающей пользовательскую и служебную очереди, составляет 2,4 %.

Полученный результат показывает, что основной вклад в расширение области допустимых нагрузок обеспечивает переход от статического распределения к стратегиям, учитывающим состояние узлов. При этом сетевая составляющая в предложенном алгоритме выполняет роль дополнительного уточняющего фактора при выборе множества реплик. Тем самым результаты по

$\lambda_{\text{доп}}$ не противоречат графикам, построенным для отдельных сценариев: в сценариях с изменением числа доступных узлов или неоднородностью ветвей обслуживания преимущество алгоритма проявляется через снижение времени записи и верхних процентилей задержки, а в предельной области нагрузок – через увеличение допустимой интенсивности входного потока.

Сценарий частичного снижения производительности одного узла показывает, что состояние узла не следует рассматривать только как бинарное: «доступен» или «недоступен». В ряде режимов узел со сниженной интенсивностью обслуживания ухудшает время записи, однако при высокой нагрузке его использование может быть оправдано за счет разгрузки остальных узлов. Следовательно, решение об исключении или использовании такого узла должно приниматься с учетом текущей нагрузки, числа доступных узлов и режима подтверждения записи.

Область применимости предложенного алгоритма определяется условиями, при которых состояние узлов и сетевых направлений существенно влияет на время выполнения операции записи. Алгоритм наиболее целесообразен для распределенных подсистем хранения туманного уровня, где операции записи выполняются с репликацией, доступно больше узлов, чем требуется для формирования реплик, а пользовательская и служебная нагрузки изменяются во времени.

Предложенный алгоритм не следует рассматривать как решение, обеспечивающее преимущество во всех режимах работы. При низкой нагрузке, однородных сетевых задержках и отсутствии выраженной служебной активности равномерное распределение может давать сопоставимые результаты, поскольку требует меньшего объема телеметрической информации и более простой обработки состояния узлов. Поэтому применение интегрального алгоритма наиболее обосновано в условиях нестационарности, неоднородности и конкуренции пользовательских и служебных потоков.

Ограничения применимости алгоритма связаны прежде всего с актуальностью наблюдаемой информации о состоянии системы. При

существенном запаздывании телеметрии качество выбора узлов может снижаться, поскольку решение принимается по устаревшим значениям очередей и задержек. Кроме того, в настоящей главе основное внимание уделено операции записи; распространение подхода на смешанные профили чтения и записи, коррелированные отказы и сложные сценарии восстановления требует дополнительного исследования.

Таким образом, результаты экспериментальной оценки подтверждают, что алгоритм интегрального выбора узлов целесообразно применять как механизм управления нагрузкой в неоднородной распределенной подсистеме хранения туманного уровня. Его практическая значимость состоит в согласованном выборе узлов хранения с учетом пользовательской нагрузки, служебного межузлового обмена, сетевых задержек и ограничений готовности. В исследованной конфигурации это позволило увеличить допустимую интенсивность входного потока операций записи на 48,3 % по сравнению с равномерным циклическим распределением при соблюдении ограничений на среднее время завершения операции записи, значение p_{99} и готовность выбранного множества узлов [137].

4.6 Выводы

В настоящей главе выполнена экспериментальная оценка предложенного алгоритма выбора узлов хранения при выполнении операции записи с репликацией данных в распределенной подсистеме хранения туманного уровня.

Основные результаты главы заключаются в следующем.

- показано, что операция записи с репликацией данных должна рассматриваться как параллельный процесс с ожиданием подтверждений записи. Последовательная интерпретация приводит к завышенной оценке времени записи и не отражает фактическую структуру обслуживания, при которой реплики передаются на несколько узлов хранения одновременно,
- установлено, что режим подтверждения существенно влияет на временные характеристики операции записи. В строгом режиме подтверждения

время завершения операции определяется максимальным временем обслуживания среди выбранных ветвей, поэтому система становится чувствительной к локальным перегрузкам, росту служебной нагрузки и увеличению сетевой задержки на отдельных направлениях. В пороговом режиме подтверждения ветви обслуживания с увеличенным временем завершения могут не определять пользовательское время ответа, однако продолжают использовать ресурсы узлов и межузловых соединений до завершения соответствующих операций,

- подтверждено, что при наличии избыточного множества доступных узлов $N > R$ стратегия выбора узлов хранения становится самостоятельным механизмом управления временем выполнения операции записи. В таких условиях выбор состава множества $R(k)$ влияет не только на распределение пользовательской нагрузки, но и на интенсивность служебного межузлового обмена,

- сопоставление базовых стратегий показало, что равномерное циклическое распределение может быть приемлемым в однородных условиях и при низкой нагрузке, однако при росте интенсивности входного потока, накоплении очередей и неоднородности сетевых задержек оно приводит к более раннему росту времени записи,

- показано, что выбор узлов только по минимальной пользовательской очереди улучшает результаты по сравнению с равномерным распределением, но остается недостаточным для распределенной подсистемы хранения туманного уровня, поскольку не учитывает служебную нагрузку и сетевую составляющую операции записи,

- установлено, что учет служебной нагрузки позволяет снизить вероятность выбора узлов, занятых фоновыми процессами репликации, синхронизации, восстановления или перераспределения данных, что подтверждает необходимость совместного рассмотрения пользовательской и служебной составляющих нагрузки при управлении операциями записи,

– показано, что учет сетевой задержки особенно значим для туманной инфраструктуры, где узлы хранения могут располагаться в разных сегментах сети. Включение сетевой составляющей в критерий выбора узлов позволяет снизить чувствительность операции записи к неблагоприятным коммуникационным направлениям,

– рассмотрен сценарий частичного снижения производительности одного из узлов хранения. Установлено, что такой узел не всегда должен исключаться из обслуживания: при низкой нагрузке его использование может снижать время записи, однако при высокой нагрузке он может снижать общее время выполнения операции за счет разгрузки остальных узлов,

– определена область применимости предложенного алгоритма. Наибольший эффект достигается в нестационарных и неоднородных условиях функционирования: при росте входной нагрузки, наличии служебных операций, различии сетевых задержек и возможности выбора R узлов из множества $N > R$. В однородной системе при низкой нагрузке преимущество алгоритма может быть ограниченным.

Таким образом, экспериментальная оценка подтвердила, что алгоритм интегрального выбора узлов обеспечивает более обоснованный выбор узлов хранения при выполнении операции записи с репликацией данных. Его применение позволяет учитывать пользовательскую нагрузку, служебный межузловой обмен и сетевые задержки, что снижает чувствительность времени записи к локальным перегрузкам и неоднородности туманной телекоммуникационной инфраструктуры.

ЗАКЛЮЧЕНИЕ

В диссертационной работе решена научная задача разработки модели и алгоритма распределения нагрузки в распределенной подсистеме хранения данных, функционирующей на уровне туманных вычислений в составе телекоммуникационной системы. Полученные результаты направлены на повышение производительности операций записи с репликацией данных при соблюдении требований к готовности выполнения операции, ограничениях межузлового обмена и конкуренции пользовательских и служебных операций за ресурсы узлов хранения и телекоммуникационной сети.

В результате выполненного исследования обосновано, что распределенная подсистема хранения данных в инфраструктуре туманных вычислений должна рассматриваться как функциональный элемент телекоммуникационной системы, что обусловлено тем, что операции записи, репликации, согласования и восстановления данных формируют межузловой трафик, используют коммуникационные ресурсы и непосредственно влияют на задержку, пропускную способность и непрерывность предоставления сервиса. Следовательно, управление такой подсистемой должно учитывать как параметры хранения, так и состояние сетевого взаимодействия между узлами.

В работе обоснован выбор распределенной программно-определяемой подсистемы хранения как архитектурной основы исследования. Показано, что такой тип подсистемы соответствует требованиям инфраструктуры туманных вычислений, поскольку допускает распределенное размещение узлов, масштабирование по числу элементов, программное управление конфигурацией и реализацию отказоустойчивых режимов работы. Установлено, что наиболее

существенное влияние на производительность рассматриваемой подсистемы оказывает операция записи с репликацией данных, поскольку она включает передачу данных на несколько узлов хранения, фиксацию данных на репликах, получение подтверждений и выполнение служебных операций согласования.

Разработана модель управления нагрузкой распределенной подсистемы хранения данных туманного уровня, в которой отдельно учитываются пользовательские операции записи и служебные операции поддержания репликации. В модели введены пользовательские и служебные очереди, ограничения по обслуживающим ресурсам узлов, ограничения пропускной способности межузловых соединений, а также требование сохранения заданного коэффициента готовности выполнения операции записи, что позволило формализовать управление совокупной нагрузкой подсистемы хранения как выбор допустимых управляющих воздействий в условиях ограниченных коммуникационных и вычислительных ресурсов.

Разработан метод оценки готовности выполнения операции записи с репликацией данных. Метод учитывает фактор репликации, порог подтверждения записи и доступность элементов цепочки обслуживания операции. Показано, что режим подтверждения существенно влияет на готовность выполнения записи: для исследуемой конфигурации переход от строгого режима подтверждения к пороговому снижает расчетную ожидаемую недоступность функции записи с 22,334 до 5,260 мин/год. Полученный результат подтверждает необходимость выбора параметров репликации и подтверждения записи с учетом не только временных характеристик, но и требований к доступности выполнения операции.

Предложен алгоритм выбора узлов хранения при выполнении операции записи с репликацией данных. Алгоритм основан на интегральной оценке состояния узлов, включающей пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности. В отличие от равномерного циклического распределения и выбора только по минимальной пользовательской очереди, предложенный алгоритм учитывает несколько факторов, определяющих

время завершения операции записи в распределенной инфраструктуре, что позволяет снижать вероятность выбора перегруженных узлов, узлов с повышенной служебной активностью или неблагоприятными сетевыми характеристиками.

Показана реализуемость предложенного алгоритма в распределенной подсистеме хранения туманного уровня. Алгоритм опирается на измеримые или оцениваемые параметры состояния инфраструктуры: длины пользовательских и служебных очередей, сетевые задержки, состояние узлов и доступность цепочки обслуживания операции записи. Рассмотрена вычислительная сложность алгоритма и показано, что он может применяться как при полном рассмотрении множества доступных узлов, так и при выборе из ограниченного числа кандидатов, что обеспечивает возможность его использования в масштабируемых распределенных конфигурациях.

Экспериментальная проверка предложенного подхода выполнена на модели операции записи с репликацией данных. Операция записи представлена как параллельный процесс с ожиданием подтверждений, что позволяет учитывать влияние отдельных ветвей обслуживания и режима подтверждения записи. Показано, что последовательная интерпретация операции записи приводит к завышенной оценке времени выполнения, тогда как параллельная модель более корректно отражает особенности записи с репликацией в распределенной подсистеме хранения.

Результаты экспериментального исследования подтвердили, что стратегия выбора узлов хранения оказывает существенное влияние на время завершения операции записи. При малой нагрузке и однородных характеристиках узлов различия между стратегиями выражены слабо. При росте интенсивности входного потока, наличии служебной нагрузки, неоднородности сетевых задержек и частичном снижении производительности отдельных узлов преимущество предложенного алгоритма становится более выраженным. Равномерное циклическое распределение в таких условиях может регулярно включать в операцию записи неблагоприятные ветви обслуживания, тогда как алгоритм интегрального выбора снижает вероятность их выбора.

Для исследуемой конфигурации установлено, что применение алгоритма интегрального выбора узлов позволяет увеличить допустимую интенсивность входного потока операций записи на 48,3 % по сравнению с равномерным циклическим распределением при соблюдении заданного ограничения на время завершения операции. Данный результат подтверждает, что учет пользовательской очереди, служебной нагрузки и сетевой составляющей позволяет повысить эффективность использования ресурсов распределенной подсистемы хранения без выхода за пределы допустимого эксплуатационного режима.

К основным научным результатам диссертационной работы относятся:

- модель управления информационными потоками распределенной подсистемы хранения данных туманного уровня телекоммуникационной системы, отличающаяся раздельным учетом пользовательских операций записи и служебных операций поддержания репликации,
- метод оценки готовности выполнения операции записи с репликацией данных, позволяющий учитывать параметры репликации, порог подтверждения и доступность элементов цепочки обслуживания,
- алгоритм интегрального выбора узлов хранения при выполнении операции записи с репликацией данных, учитывающий пользовательскую очередь, служебную нагрузку, сетевую задержку и ограничение по готовности,
- экспериментальная оценка предложенного алгоритма, подтвердившая увеличение допустимой интенсивности входного потока операций записи на 48,3 % по сравнению с равномерным циклическим распределением в исследуемой конфигурации, при этом дополнительное преимущество относительно стратегии, учитывающей пользовательскую и служебную очереди, составляет 2,4 %.

Разработанные модель управления информационными потоками, метод оценки готовности и алгоритм интегрального выбора узлов хранения могут применяться для выбора параметров репликации и режима подтверждения записи, выбора узлов хранения и оценки допустимых режимов функционирования

распределенной подсистемы хранения с учетом требований к производительности и готовности.

Полученные результаты подтверждают достижение цели диссертационной работы и решение поставленных задач. Дальнейшее развитие исследования может быть связано с расширением модели на смешанные профили операций чтения и записи, учетом коррелированных отказов, детализацией сценариев восстановления избыточности и анализом влияния запаздывания телеметрической информации на качество выбора управляющих воздействий.

СПИСОК ЛИТЕРАТУРЫ

1. ETSI GS MEC 003 V3.1.1 (2022-03). Multi-access Edge Computing (MEC); Framework and Reference Architecture. – Sophia Antipolis: ETSI, 2022. – URL: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/03.01.01_60/gs_mec003v030101p.pdf (дата обращения: 05.06.2026).
2. Recommendation ITU-T Y.3540 (10/2023). Edge computing – Overview and high-level requirements. – Geneva: International Telecommunication Union, 2023. – URL: <https://www.itu.int/epublications/publication/itu-t-y-3540-2023-10-edge-computing-overview-and-high-level-requirements> (дата обращения: 23.01.2026).
3. Recommendation ITU-T X.1648 (04/2025). Guideline on edge computing data security. – Geneva: International Telecommunication Union, 2025. – URL: <https://www.itu.int/epublications/publication/itu-t-x-1648-2025-04-guideline-on-edge-computing-data-security> (дата обращения: 23.01.2026).
4. IEEE 1934-2018. IEEE Standard for Adoption of OpenFog Reference Architecture for Fog Computing. – New York: IEEE, 2018. – URL: <https://standards.ieee.org/ieee/1934/7137/> (дата обращения: 23.01.2026).
5. ETSI GS MEC-IEG 006 V1.1.1 (2017-01). Mobile Edge Computing; Market Acceleration; MEC Metrics Best Practice and Guidelines. – Sophia Antipolis: ETSI, 2017. – URL: https://www.etsi.org/deliver/etsi_gs/mec-ieg/001_099/006/01.01.01_60/gs_mec-ieg006v010101p.pdf (дата обращения: 05.06.2026).
6. ETSI GS MEC 045 V3.2.1 (2025-08). Multi-access Edge Computing (MEC); QoS Measurement API. – Sophia Antipolis: ETSI, 2025. – URL: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/045/03.02.01_60/gs_MEC045v030201p.pdf (дата обращения: 05.06.2026).

7. Федеральный закон от 07.07.2003 № 126-ФЗ «О связи»: в ред. от 25.05.2026 // Собрание законодательства Российской Федерации. – 2003. – № 28. – Ст. 2895.
8. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации»: в ред. от 29.12.2025 // Собрание законодательства Российской Федерации. – 2006. – № 31, ч. 1. – Ст. 3448.
9. Recommendation ITU-R M.2160-0 (11/2023). Framework and overall objectives of the future development of IMT for 2030 and beyond. – Geneva: International Telecommunication Union, 2023. – URL: <https://www.itu.int/rec/R-REC-M.2160-0-202311-I/en> (дата обращения: 05.06.2026).
10. 3GPP TS 22.261. Service requirements for the 5G system. – URL: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3107> (дата обращения: 23.01.2026).
11. Iorga M., Feldman L., Barton R., Martin M. J., Goren N., Mahmoudi C. Fog Computing Conceptual Model: NIST Special Publication 500-325. – Gaithersburg: National Institute of Standards and Technology, 2018. – DOI: 10.6028/NIST.SP.500-325.
12. Meruje Ferreira L. M., Coelho F., Pereira J. Databases in Edge and Fog Environments: A Survey // ACM Computing Surveys. – 2024. – Vol. 56, No. 11. – Article 285. – P. 1-40. – DOI: 10.1145/3666001
13. Gkonis P., Giannopoulos A., Trakadas P., Masip-Bruin X., D’Andria F. A Survey on IoT-Edge-Cloud Continuum Systems: Status, Challenges, Use Cases, and Open Issues // Future Internet. – 2023. – Vol. 15, No. 12. – Article 383. – DOI: 10.3390/fi15120383.
14. Mahbub M., Shubair R. M. Contemporary advances in multi-access edge computing: A survey of fundamentals, architecture, technologies, deployment cases, security, challenges, and directions // Journal of Network and Computer Applications. – 2023. – Vol. 219. – Article 103726. – DOI: 10.1016/j.jnca.2023.103726.
15. Maia A., Boutouchent A., Kardjadja Y. [et al.]. A survey on integrated computing, caching, and communication in the cloud-to-edge continuum // Computer

Communications. – 2024. – Vol. 219. – P. 128-152. – DOI: 10.1016/j.comcom.2024.03.005.

16. Pathan A.-M. K., Buyya R. A Taxonomy and Survey of Content Delivery Networks: Technical Report GRIDS-TR-2007-4. – Melbourne: University of Melbourne, 2007. – URL: <https://clouds.cis.unimelb.edu.au/~rbuyya/gridbus/reports/CDN-Taxonomy.pdf> (дата обращения: 05.06.2026).

17. Shvidkiy A. A., Savelieva A. A., Zarubin A. A. Caching Methods Analysis for Improving Distributed Storage Systems Performance // 2021 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO). – IEEE, 2021. – P. 1-5.

18. Stocker V., Smaragdakis G., Lehr W., Bauer S. The growing complexity of content delivery networks: Challenges and implications for the Internet ecosystem // Telecommunications Policy. – 2017. – Vol. 41, No. 10. – P. 1003-1016.

19. Волков А. Н. Задача маршрутизации в сети динамических туманных вычислений // Труды учебных заведений связи. – 2024. – Т. 10, № 4. – С. 27-37.

20. Волков А. Н. Динамические туманные вычисления и бессерверная архитектура: на пути к зеленым ИКТ // Труды учебных заведений связи. – 2024. – Т. 10, № 3. – С. 24-34.

21. Пинчук А. В., Соколов Н. А., Фрейнкман В. А. Общие принципы туманных вычислений // Первая миля. – 2018. – № 3 (72). – С. 38-45. – DOI: 10.22184/2070-8963.2018.72.3.38.45.

22. Кашкаров Д. В. Разработка и исследование моделей и методов обеспечения требований к характеристикам надежности для сетей связи с ультра малыми задержками : дис. канд. техн. наук. – Санкт-Петербург: СПбГУТ, 2022. – 121 с.

23. Плотников П. В., Владыко А. Г. Анализ подходов к оптимизации V2X-систем: кластеризация, граничные и туманные вычисления // Труды учебных заведений связи. – 2024. – Т. 10, № 3. – С. 7-22.

24. Тормасов А., Лысов А., Мазур Э. Распределенные системы хранения данных: анализ, классификация и варианты выбора // Труды Института системного

программирования РАН. – 2015. – Т. 27, № 6. – С. 225-252. – DOI: 10.15514/ISPRAS-2015-27(6)-15.

25. Bonomi F., Milito R., Zhu J., Addepalli S. Fog computing and its role in the internet of things // Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing. – New York: Association for Computing Machinery, 2012. – P. 13-16. – DOI: 10.1145/2342509.2342513.

26. Chiang M., Zhang T. Fog and IoT: An Overview of Research Opportunities // IEEE Internet of Things Journal. – 2016. – Vol. 3, No. 6. – P. 854-864. – DOI: 10.1109/IIOT.2016.2584538.

27. Шорин О. А., Асланян В. А. Подходы к интеграции технологии NB-IoT с сетью 5G // Экономика и качество систем связи. – 2024. – № 3 (33). – С. 56-62.

28. Saleh S. S., Alansari I., Hamiaz M. K. [et al.]. iFogRep: An intelligent consistent approach for replication and placement of IoT based on fog computing // Egyptian Informatics Journal. – 2023. – Vol. 24, No. 2. – P. 327-339. – DOI: 10.1016/j.eij.2023.05.003.

29. Mahmud R., Kotagiri R., Buyya R. Fog Computing: A Taxonomy, Survey and Future Directions // Internet of Everything: Algorithms, Methodologies, Technologies and Perspectives. – Singapore: Springer, 2018. – P. 103-130.

30. Fog and Edge Computing: Principles and Paradigms / ed. by R. Buyya, S. N. Srirama. – Hoboken: Wiley, 2019. – 496 p.

31. Koucheryavy A., Kirichek R., Makolkina M. [et al.]. Novel Prospects of Scientific Research in the Field of Telecommunications Networks for the 2021-2024 Years // Telecom IT. – 2020. – Vol. 8, No. 3. – P. 1-19. – DOI: 10.31854/2307-1303-2020-8-3-1-19.

32. Qian L., Qu Z., Cai M. [et al.]. FastCache: A write-optimized edge storage system via concurrent merging cache for IoT applications // Journal of Systems Architecture. – 2022. – Vol. 131. – Article 102718. – DOI: 10.1016/j.sysarc.2022.102718.

33. Зарубин А. А., Савельева А. А., Швидкий А. А. Моделирование нагрузки на распределенную систему хранения данных // Вестник связи. – 2018. – № 7. – С. 13-16.
34. Зарубин А. А., Швидкий А. А., Савельева А. А. Особенности моделирования нагрузки на РСХД // Вестник связи. – 2018. – № 8. – С. 13-17.
35. Storage subsystem // SNIA Dictionary. – URL: <https://www.snia.org/education/online-dictionary/term/storage-subsystem> (дата обращения: 02.02.2026).
36. ПНСТ 807–2022. Информационные технологии. Центры обработки данных. Термины и определения: предварительный национальный стандарт Российской Федерации: дата введения 2023-06-01. – Москва: Российский институт стандартизации, 2023.
37. ГОСТ Р 57700.27–2020. Высокопроизводительные вычислительные системы. Термины и определения : национальный стандарт Российской Федерации: дата введения 2021-06-01. – Москва: Стандартинформ, 2020.
38. Долгун В. О., Швидкий А. А., Тарлыков А. В., Савельева А. А. Подходы к построению инфраструктуры виртуальных рабочих столов на базе гиперконвергентной инфраструктуры // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2019): VIII Международная научно-техническая и научно-методическая конференция. – Санкт-Петербург: СПбГУТ, 2019. – С. 158-163.
39. Казаков Д. Б., Тарлыков А. В., Швидкий А. А., Савельева А. А. Оценка производительности работы виртуальных рабочих столов на базе гиперконвергентной инфраструктуры // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2019): VIII Международная научно-техническая и научно-методическая конференция. – Санкт-Петербург: СПбГУТ, 2019. – С. 170-174.
40. Теоретические исследования решений для повышения эффективности программно-определяемой СХД: отчет о научно-исследовательской работе / Зарубин А. А., Швидкий А. А., Елагин В. С., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2021. – 119 с. – НИР шифр «Мульти-ЦОД».

Этап 2. – Государственный контракт № П33-1-26/5. – № гос. регистрации 121032500023-5.

41. Разработка моделей и вариантов построения программно-определяемой СХД: отчет о научно-исследовательской работе / Зарубин А. А., Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2021. – 182 с. – НИР шифр «Мульти-ЦОД». Этап 3. – Государственный контракт № П33-1-26/5. – № гос. регистрации 121032500023-5.

42. Разработка системотехнических решений по построению ЭО программно-аппаратной платформы программно-определяемой СХД: отчет о научно-исследовательской работе / Зарубин А. А., Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2022. – 179 с. – НИР шифр «Мульти-ЦОД». Этап 4. – Государственный контракт № П33-1-26/5. – № гос. регистрации 121032500023-5.

43. Băcescu Ene G.-V., Stoia M.-A., Cojocaru C., Todea D. A. SMART Multi-Criteria Decision Analysis (MCDA) – One of the Keys to Future Pandemic Strategies // Journal of Clinical Medicine. – 2025. – Vol. 14, No. 6. – Article 1943. – DOI: 10.3390/jcm14061943.

44. Jato-Espino D., Martín-Rodríguez Á., Martínez-Corral A., Sañudo-Fontaneda L. A. Multi-expert multi-criteria decision analysis model to support the conservation of paramount elements in industrial facilities // Heritage Science. – 2022. – Vol. 10. – Article 68. – DOI: 10.1186/s40494-022-00712-7.

45. Rethink Data: Put More of Your Business Data to Work – from Edge to Cloud: A Seagate Technology Report. – Seagate, 2020. – URL: <https://www.seagate.com/gb/en/our-story/rethink-data/> (дата обращения: 23.01.2026).

46. SPC-1 and SPC-1E Benchmark Results // Storage Performance Council. – URL: <https://www.spcresults.org/benchmarks/results/spc1-spc1e> (дата обращения: 23.01.2026).

47. Зарубин А. А., Савельева А. А., Швидкий А. А. Анализ характеристик открытых систем хранения данных // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2022): XI Международная

научно-техническая и научно-методическая конференция. – Санкт-Петербург: СПбГУТ, 2022.

48. ISO/IEC TS 22237-31:2026. Information technology – Data centre facilities and infrastructures – Part 31: Key performance indicators for resilience. – URL: <https://www.iso.org/standard/88711.html> (дата обращения: 19.03.2026).

49. Storage Policies // OpenStack Docs. – URL: https://docs.openstack.org/swift/stein/overview_policies.html (дата обращения: 19.03.2026).

50. Shen Z., Li X., Lee P. P. C. Fast Predictive Repair in Erasure-Coded Storage // 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN). – Portland: IEEE, 2019. – P. 556-567. – DOI: 10.1109/DSN.2019.00062.

51. Kløve T., Korzhik V. Error Detecting Codes: General Theory and Their Application in Feedback Communication Systems. – Berlin: Springer, 2012. – 257 p.

52. Rashmi K. V., Shah N. B., Gu D., Kuang H., Borthakur D., Ramchandran K. A “hitchhiker’s” guide to fast and efficient data reconstruction in erasure-coded data centers // Proceedings of the 2014 ACM Conference on SIGCOMM. – New York: ACM, 2014. – P. 331-342. – DOI: 10.1145/2619239.2626325.

53. Balaji S. B., Krishnan M. N., Vajha M. [et al.]. Erasure coding for distributed storage: an overview // Science China Information Sciences. – 2018. – Vol. 61, No. 10. – Article 100301. – DOI: 10.1007/s11432-018-9482-6.

54. Shen Z., Li X., Lee P. P. C. [et al.]. A Survey of the Past, Present, and Future of Erasure Coding for Storage Systems // ACM Transactions on Storage. – 2025. – Vol. 21, No. 1. – P. 1-39. – DOI: 10.1145/3708994.

55. Shvidkiy A. A., Savelieva A. A., Zarubin A. A. Approaches Analysis to Determining the Characteristics of the Load on the Disk Subsystem Created by User Applications // 2022 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO). – IEEE, 2022. – P. 1-8.

56. Shvidkiy A. A., Savelieva A. A., Zarubin A. A. Analysis of fault tolerance improvement in distributed over network data storage // 2021 13th International Congress

on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). – IEEE, 2021. – P. 214-218.

57. Зарубин А. А., Елагин В. С., Швидкий А. А., Савельева А. А. Подходы к моделированию нагрузки на распределенную систему хранения данных // Системы синхронизации, формирования и обработки сигналов. – 2018. – Т. 9, № 3. – С. 90-95.

58. Zarubin A. A., Elagin V. S., Shvidkiy A. A., Savelieva A. A. Approaches to modeling the load on a distributed storage system // 2018 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO). – IEEE, 2018. – P. 1-5.

59. Pfandzelter T., Japke N., Schirmer T., Hasenburg J., Bermbach D. Managing data replication and distribution in the fog with FReD // Software: Practice and Experience. – 2023. – Vol. 53, No. 10. – P. 1958-1981. – DOI: 10.1002/spe.3237.

60. Whittaker M., Charapko A., Hellerstein J. M., Howard H., Stoica I. Read-Write Quorum Systems Made Practical // Proceedings of the 8th Workshop on Principles and Practice of Consistency for Distributed Data. – New York: ACM, 2021. – P. 1-8. – DOI: 10.1145/3447865.3457962.

61. Replication and Synchronization Modes // Oracle Documentation. – URL: <https://docs.oracle.com/cd/E19359-01/819-6148-10/chap2.html> (дата обращения: 23.01.2026).

62. Shvidkiy A. A., Spirkina A. V., Savelieva A. A., Tarlykov A. V. Evaluation of the impact the hyper-converged infrastructure storage subsystem synchronization on the overall performance // 2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). – Brno: IEEE, 2020. – P. 248-252. – DOI: 10.1109/ICUMT51630.2020.9222420.

63. Mukherjee D. Scalable Load Balancing Algorithms in Networked Systems: PhD thesis. – Eindhoven: Technische Universiteit Eindhoven, 2018. – 377 p. – ISBN 978-90-386-4558-2.

64. Lu Y., Xie Q., Kliot G., Geller A., Larus J. R., Greenberg A. Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services //

Performance Evaluation. – 2011. – Vol. 68, No. 11. – P. 1056-1071. – DOI: 10.1016/j.peva.2011.07.015.

65. Karger D., Lehman E., Leighton T. [et al.]. Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the World Wide Web // Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing. – New York: ACM, 1997. – P. 654-663. – DOI: 10.1145/258533.258660.

66. Dean J., Barroso L. A. The Tail at Scale // Communications of the ACM. – 2013. – Vol. 56, No. 2. – P. 74-80. – DOI: 10.1145/2408776.2408794.

67. Mouradian C., Naboulsi D., Yangui S., Glitho R. H., Morrow M. J., Polakos P. A. A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges // IEEE Communications Surveys & Tutorials. – 2018. – Vol. 20, No. 1. – P. 416-464. – DOI: 10.1109/COMST.2017.2771153.

68. Fischer S., Vöcking B. Adaptive routing with stale information // Theoretical Computer Science. – 2009. – Vol. 410, No. 36. – P. 3357-3371. – DOI: 10.1016/j.tcs.2009.05.014.

69. Elagin V. S., Redrugina N. M., Saveleva A. The equivalence principle for adaptive load balancing in networks and systems // 2024 International Scientific and Technical Conference Modern Computer Network Technologies (MoNeTeC). – Moscow: IEEE, 2024. – P. 1-6. – DOI: 10.1109/MoNeTeC60984.2024.10768123.

70. Mahdian S., Zhou Z., Bambos N. Robustness of Join-the-Shortest-Queue Scheduling to Communication Delay // 2018 Annual American Control Conference (ACC). – Milwaukee: IEEE, 2018. – P. 3708-3713.

71. Volkov A., Stepanov S. Development of a Model and Algorithms for Servicing Traffic in a Cloud Computing System // 2021 28th Conference of Open Innovations Association (FRUCT). – Moscow: IEEE, 2021. – P. 510-515.

72. Степанов С. Н., Саламех Н. Построение и анализ обобщенной модели передачи мультисервисного трафика реального времени в сетях связи // Наукоемкие технологии в космических исследованиях Земли. – 2016. – Т. 8, № 4. – С. 37-44.

73. Шуляк А. В. Сравнительный анализ алгоритмов балансировки нагрузки в среде облачных вычислений // Научный журнал. – 2021. – № 6 (61). – С. 6-11.
74. Mohanty A., Chiranjeevi G. N. Comparative Analysis of Round Robin Arbiter and Weighted Round Robin Arbiter // 2024 Asia Pacific Conference on Innovation in Technology (APCIT). – Mysuru: IEEE, 2024. – P. 1-4. – DOI: 10.1109/APCIT62007.2024.10673654.
75. Cao J., Nyberg C. An approximate analysis of load balancing using stale state information for servers in parallel // Proceedings of the Second IASTED International Conference on Communications, Internet, and Information Technology. – Calgary: ACTA Press, 2003.
76. Gupta V., Harchol-Balter M., Sigman K., Whitt W. Analysis of join-the-shortest-queue routing for web server farms // Performance Evaluation. – 2007. – Vol. 64, No. 9-12. – P. 1062-1081. – DOI: 10.1016/j.peva.2007.06.012.
77. Hurtado-Lange D., Maguluri S. T. Load balancing system under Join the Shortest Queue: Many-Server-Heavy-Traffic Asymptotics // Queueing Systems. – 2022. – Vol. 101, No. 3-4. – P. 353-391.
78. Selen J., Adan I., Borst S., van Leeuwen J. S. H. Steady-state analysis of shortest expected delay routing // Queueing Systems. – 2016. – Vol. 84, No. 3. – P. 309-354.
79. Harchol-Balter M. Performance Modeling and Design of Computer Systems: Queueing Theory in Action. – Cambridge: Cambridge University Press, 2013.
80. Mitzenmacher M. The Power of Two Choices in Randomized Load Balancing // IEEE Transactions on Parallel and Distributed Systems. – 2001. – Vol. 12, No. 10. – P. 1094-1104.
81. Van Houdt B. On the performance evaluation of distributed join-idle-queue load balancing with and without token withdrawals // Performance Evaluation. – 2024. – Vol. 165. – Article 102427. – DOI: 10.1016/j.peva.2024.102427.
82. Horváth I. A., Scully Z., Van Houdt B. Mean Field Analysis of Join-Below-Threshold Load Balancing for Resource Sharing Servers // Proceedings of the ACM on

Measurement and Analysis of Computing Systems. – 2019. – Vol. 3, No. 3. –Article 57. – P. 1-21. – DOI: 10.1145/3366705.

83. Krishnan R., Madhyastha H. V., Srinivasan S. [et al.]. Moving beyond end-to-end path information to optimize CDN performance // Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement. – New York: ACM, 2009. – P. 190-201.

84. Panigrahy N. K., Vasantam T., Basu P., Towsley D., Swami A., Leung K. K. On the Analysis and Evaluation of Proximity-Based Load-Balancing Policies // ACM Transactions on Modeling and Performance Evaluation of Computing Systems. – 2022. – Vol. 7, No. 2-4. – Article 6. – P. 1-27. – DOI: 10.1145/3549933.

85. Jiang W., Fang L., Xie H., Zhou X., Wang J. Tars: Timeliness-Aware Adaptive Replica Selection for Key-Value Stores // 2017 26th International Conference on Computer Communication and Networks (ICCCN). – Vancouver: IEEE, 2017. – P. 1-9.

86. Ongaro D., Ousterhout J. In Search of an Understandable Consensus Algorithm // Proceedings of the 2014 USENIX Annual Technical Conference. – Philadelphia: USENIX Association, 2014. – P. 305-319.

87. Schultz W., Avitabile T., Cabral A. Tunable Consistency in MongoDB // Proceedings of the VLDB Endowment. – 2019. – Vol. 12, No. 12. – P. 2071-2081. – DOI: 10.14778/3352063.3352125.

88. DeCandia G., Hastorun D., Jampani M. [et al.]. Dynamo: Amazon's Highly Available Key-Value Store // ACM SIGOPS Operating Systems Review. – 2007. – Vol. 41, No. 6. – P. 205-220. – DOI: 10.1145/1323293.1294281.

89. Howard H., Malkhi D., Spiegelman A. Flexible Paxos: Quorum Intersection Revisited // Proceedings of the 20th International Conference on Principles of Distributed Systems (OPODIS 2016) / ed. by P. Fatourou, E. Jiménez, F. Pedone. – Dagstuhl: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. – Vol. 70. – P. 25:1-25:14. – DOI: 10.4230/LIPIcs.OPODIS.2016.25.

90. Yadav R., Rahut A. FlexiRaft: Flexible Quorums with Raft // Conference on Innovative Data Systems Research (CIDR). – Amsterdam, 2023. – URL: <https://vldb.org/cidrdb/papers/2023/p83-yadav.pdf> (дата обращения: 05.06.2026).
91. Wang Z., Zhang G., Wang Y., Yang Q., Zhu J. Dayu: Fast and Low-Interference Data Recovery in Very-Large Storage Systems // Proceedings of the 2019 USENIX Annual Technical Conference. – Renton: USENIX Association, 2019. – P. 993-1007.
92. Misra P. A., Wang Z., Zhang G., Wang Y., Yang Q., Zhu J. Dayu [et al.]. Managing Tail Latency in Datacenter-Scale File Systems Under Production Constraints // Proceedings of the Fourteenth EuroSys Conference 2019. – New York: ACM, 2019. – Article 17. – P. 17:1-17:15. – DOI: 10.1145/3302424.3303973.
93. Савельева А. А. Метод оценки готовности распределенной подсистемы хранения уровня туманных вычислений к обслуживанию операций в телекоммуникационной сети // Труды учебных заведений связи. – 2026. – Т.12, №4. – С. 105-117.
94. ГОСТ Р 27.010–2019 (МЭК 61703:2016). Надежность в технике. Математические выражения для показателей безотказности, готовности, ремонтпригодности: национальный стандарт Российской Федерации: дата введения 2020-09-01. – Москва : Стандартинформ, 2019.
95. Kirsten D. Efficient and failure-aware replication of an XML database: Master thesis. – Konstanz: University of Konstanz, 2014. – URL: <https://files.basex.org/publications/Kirsten%20%5B2014%5D%2C%20Efficient%20and%20failure-aware%20replication%20of%20an%20XML%20database.pdf> (дата обращения: 05.06.2026).
96. Иваничкина Л. В., Непорада А. П. Модель надежности распределенной системы хранения данных в условиях явных и скрытых дисковых сбоев // Труды Института системного программирования РАН. – 2015. – Т. 27, № 6. – С. 253-274. – DOI: 10.15514/ISPRAS-2015-27(6)-16.
97. Reliability Prediction Procedure for Electronic Equipment: Telcordia SR-332 Issue 3 // ALD Service. – URL: <https://aldservice.com/SR332-Telcordia-Issue-3.html> (дата обращения: 20.02.2026).

98. IEC 61709:2017. Electric components – Reliability – Reference conditions for failure rates and stress models for conversion. – URL: <https://webstore.iec.ch/en/publication/28554> (дата обращения: 20.02.2026).
99. FIT/MTBF/MTTF/failure rate of CPU I7-12700 TE CM8071504619317 // Intel Community. – 2024. – URL: <https://community.intel.com/t5/Embedded-Intel-Core-Processors/FIT-MTBF-MTTF-failure-rate-of-CPU-I7-12700-TE-CM8071504619317/m-p/1580419#M5043> (дата обращения: 20.02.2026).
100. O'Connor P. D. T., Kleyner A. Practical Reliability Engineering. – 5th ed. – Chichester: Wiley, 2012.
101. Woo S. Modern Definitions in Reliability Engineering // Reliability Design of Mechanical Systems. – Cham: Springer, 2017. – P. 35-59. – DOI: 10.1007/978-3-319-50829-0_3.
102. Drive Stats Team. Backblaze Drive Stats for 2025 // Backblaze Blog. – 2026. – URL: <https://www.backblaze.com/blog/backblaze-drive-stats-for-2025/> (дата обращения: 05.06.2026).
103. Maneas S., Mahdavian K., Emami T., Schroeder B. A Study of SSD Reliability in Large Scale Enterprise Storage Deployments // 18th USENIX Conference on File and Storage Technologies (FAST 20). – Santa Clara: USENIX Association, 2020. – P. 137-149. – ISBN 978-1-939133-12-0. – URL: <https://www.usenix.org/conference/fast20/presentation/maneas> (дата обращения: 05.06.2026).
104. NGMN Alliance. 6G Requirements and Design Considerations. Version 1.0. – 2023. – URL: https://www.ngmn.org/wp-content/uploads/NGMN_6G_Requirements_and_Design_Considerations.pdf (дата обращения: 01.08.2026).
105. Hexa-X-II Deliverable D6.3. Initial Design of 6G Smart Network Management Framework. – 2024. – URL: <https://hexa-x-ii.eu/hexa-x-ii-deliverable-d6-3-focuses-on-the-initial-design-of-6g-smart-network-management-framework/> (дата обращения: 03.08.2026).
106. Data protection in Amazon S3 // Amazon Simple Storage Service Documentation. – URL: <https://docs.aws.amazon.com/AmazonS3/latest/userguide/DataDurability.html> (дата обращения: 03.03.2026).

107. Aslanpour M. S., Gill S. S., Toosi A. N. Performance evaluation metrics for Cloud, Fog and Edge computing: A review, taxonomy, benchmarks and standards for future research // Internet of Things. – 2020. – Vol. 12. – Article 100273. – DOI: 10.1016/j.iot.2020.100273.
108. Зарубин А. А., Кызыюров О. Е., Савельева А. А. Цифровое качество программно-определяемых приложений инфокоммуникационных сетей. Формирование подходов к разработке моделей и методов его оценки // Информационные технологии и телекоммуникации. – 2017. – Т. 5, № 2. – С. 56-61.
109. Wang F., Nelson M., Oral S., Atchley S., Settlemeyer B. W. [et al.]. Performance and scalability evaluation of the Ceph parallel file system // Proceedings of the 8th Parallel Data Storage Workshop. – New York: ACM, 2013. – P. 14-19. – DOI: 10.1145/2538542.2538562.
110. Gudu D., Hardt M., Streit A. Evaluating the performance and scalability of the Ceph distributed storage system // 2014 IEEE International Conference on Big Data. – Washington: IEEE, 2014. – P. 177-182. – DOI: 10.1109/BigData.2014.7004229.
111. Садов О. Л., Власов Д. В., Грудинин В. А., Каирканов А. Б., Сомс Л. Н., Титов В. Б., Хоружников С. Э., Чугреев Д. А., Шевель А. Е., Шкребец А. Е. Исследование сети хранения данных, построенной с использованием программно-конфигурируемых сетей OpenFlow // Современные проблемы науки и образования. – 2013. – № 4. – 7 с.
112. Cooper B. F., Silberstein A., Tam E., Ramakrishnan R., Sears R. Benchmarking cloud serving systems with YCSB // Proceedings of the 1st ACM Symposium on Cloud Computing. – New York: Association for Computing Machinery, 2010. – P. 143-154. – DOI: 10.1145/1807128.1807152.
113. Швидкий А. А., Зарубин А. А., Савельева А. А. Система сбора и обработки метрик производительности распределенной системы хранения данных: свидетельство о государственной регистрации программы для ЭВМ № 2022611763, Российская Федерация. – 2022.
114. Швидкий А. А., Зарубин А. А., Савельева А. А. Система генерации нагрузки на распределенную систему хранения данных с сохранением

комбинированных результатов нагрузочного тестирования: свидетельство о государственной регистрации программы для ЭВМ № 2022611764, Российская Федерация. – 2022.

115. Neely M. J. Stochastic Network Optimization with Application to Communication and Queueing Systems. – Cham: Springer International Publishing, 2010.

116. Weil S. A., Leung A. W., Brandt S. A., Maltzahn C. RADOS: A scalable, reliable storage service for petabyte-scale storage clusters // Proceedings of the 2nd International Workshop on Petascale Data Storage. – New York: ACM, 2007. – P. 35-44.

117. Васюткин А. В., Власов Д. В., Швидкий А. А., Савельева А. А. Анализ методов отслеживания влияния нагрузки сетевой подсистемы гиперконвергентной инфраструктуры на производительность вычислительного кластера // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2021): X Международная научно-техническая и научно-методическая конференция. – Санкт-Петербург: СПбГУТ, 2021. – С. 380-384.

118. Tassiulas L., Ephremides A. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks // IEEE Transactions on Automatic Control. – 1992. – Vol. 37, No. 12. – P. 1936-1948.

119. Ceph: Throttling down or up backfill and recovery and rebalance // Red Hat Customer Portal. – URL: <https://access.redhat.com/articles/1422313> (дата обращения: 28.01.2026).

120. General principles for selecting hardware // Red Hat Ceph Storage 5. Hardware Guide. – URL: https://docs.redhat.com/en/documentation/red_hat_ceph_storage/5/html/hardware_guide/general-principles-for-selecting-hardware (дата обращения: 28.01.2026).

121. Weil S. A., Brandt S. A., Miller E. L., Maltzahn C. CRUSH: Controlled, Scalable, Decentralized Placement of Replicated Data // SC'06: Proceedings of the 2006 ACM/IEEE Conference on Supercomputing. – Tampa: IEEE, 2006. – P. 31. – DOI: 10.1109/SC.2006.50.

122. Handling a node failure // Red Hat Ceph Storage 4. Operations Guide.
– URL: https://docs.redhat.com/en/documentation/red_hat_ceph_storage/4/html/operations_guide/handling-a-node-failure (дата обращения: 28.01.2026).
123. Weil S. A., Brandt S. A., Miller E. L., Long D. D. E., Maltzahn C. Ceph: A Scalable, High-Performance Distributed File System // Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI '06). – Berkeley: USENIX Association, 2006. – P. 307-320.
124. Мутханна А. С. А. Интегральное решение проблемы размещения контроллеров и балансировки нагрузки // Труды учебных заведений связи. – 2023. – Т. 9, № 2. – С. 81-93. – DOI: 10.31854/1813-324X-2023-9-2-81-93.
125. Клименко А. Б., Сафроненкова И. Б. Решение задачи распределения вычислительной нагрузки в средах туманных вычислений на базе онтологий // Известия ЮФУ. Технические науки. – 2018. – № 8. – С. 83-94.
126. Швидкий А. А., Христофоров Р. О., Савельева А. А. Система организации доступа к управлению данными гиперконвергентных центров обработки данных: свидетельство о государственной регистрации программы для ЭВМ № 2023615289, Российская Федерация. – 2023.
127. Кучерявый А. Е., Окунева Д. В., Парамонов А. И., Хоанг Ф. Н. Методы распределения трафика в гетерогенной сети Интернета вещей высокой плотности // Труды учебных заведений связи. – 2024. – Т. 10, № 2. – С. 67-74.
128. Gandhi A., Zhang X., Mittal N. HALO: Heterogeneity-Aware Load Balancing // 2015 IEEE 23rd International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. – Atlanta: IEEE, 2015. – P. 242-251. – DOI: 10.1109/MASCOTS.2015.14.
129. Suresh L., Canini M., Schmid S., Feldmann A. C3: Cutting Tail Latency in Cloud Data Stores via Adaptive Replica Selection // 12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15). – Oakland: USENIX Association, 2015. – P. 513-527.
130. Глушак Е. В., Михайлова П. Д. Обзор адаптивных алгоритмов распределения потоков данных Интернета вещей в облачных и туманных средах

// Инфокоммуникационные технологии. – 2024. – Т. 22, № 4 (88). – С. 15-22. – DOI: 10.18469/ikt.2024.22.4.02.

131. Разработка программно-алгоритмического решения для модуля генерации профилей нагрузки на подсистему ввода-вывода программно-аппаратной платформы программно-определяемой СХД: отчет о научно-исследовательской работе / Зарубин А. А., Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2022. – 121 с. – НИР шифр «Мульти-ЦОД». Этап 5. – Государственный контракт № ПЗ3-1-26/5. – № гос. регистрации 121032500023-5.

132. Разработка программно-алгоритмического решения для ЭО модуля повышения производительности на базе нейронной сети программно-аппаратной платформы программно-определяемой СХД: отчет о научно-исследовательской работе / Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М., Тарабанов И. Ф. [и др.]; СПбГУТ. – Санкт-Петербург, 2022. – 142 с. – НИР шифр «Мульти-ЦОД». Этап 6. – Государственный контракт № ПЗ3-1-26/5. – № гос. регистрации 121032500023-5.

133. Разработка программно-методического обеспечения экспериментальных исследований ЭО программной-аппаратной платформы программно-определяемой СХД: отчет о научно-исследовательской работе / Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М., Тарабанов И. Ф. [и др.]; СПбГУТ. – Санкт-Петербург, 2023. – 188 с. – НИР шифр «Мульти-ЦОД». Этап 7. – Государственный контракт № ПЗ3-1-26/5. – № гос. регистрации 121032500023-5.

134. Abdul Jaleel J., Doroudi S., Gardner K. Queue-length-aware dispatching in large-scale heterogeneous systems // Queueing Systems. – 2024. – Vol. 108, No. 1-2. – P. 125-184. – DOI: 10.1007/s11134-024-09918-x.

135. Pang X., Wang J. Reducing Tail Latency in Storage-Disaggregated Database Systems // Proceedings of the ACM on Management of Data. – 2026. – Vol. 4, No. 1. – Article 74. – 26 p. – DOI: 10.1145/3786688.

136. Алексеева Н. Н., Редругина Н. М., Савельева А. А., Тарабанов И. Ф. Математическая модель записи информации на узлы распределенной системы хранения данных // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2022): XI Международная научно-техническая и научно-методическая конференция, Санкт-Петербург. – Санкт-Петербург: СПбГУТ, 2022. – С. 65-70.

137. Елагин В. С., Редругина Н. М., Савельева А. А. Экспериментальная оценка алгоритма выбора узлов хранения при выполнении операции записи с репликацией данных в распределенной телекоммуникационной инфраструктуре // Электросвязь. – 2026. – № 7. – С. 24-35.

138. Wang H., Li J., Shen Z., Zhou Y. Approximations and Bounds for (n, k) Fork-Join Queues: A Linear Transformation Approach // 2018 18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). – Washington: IEEE, 2018. – P. 422-431. – DOI: 10.1109/CCGRID.2018.00069.

139. Gardner K., Zbarsky S., Doroudi S., Harchol-Balter M., Hyttiä E., Scheller-Wolf A. Reducing Latency via Redundant Requests: Exact Analysis // ACM SIGMETRICS Performance Evaluation Review. – 2015. – Vol. 43, No. 1. – P. 347-360. – DOI: 10.1145/2796314.2745873.

140. Якимов И. М., Кирпичников А. П., Мокшин В. В. Моделирование сложных систем в имитационной среде AnyLogic // Вестник Казанского технологического университета. – 2014. – Т. 17, № 13. – С. 352-357.

141. Экспериментальные исследования разработанных решений. Обобщение и оценка результатов исследований в части ЭО модуля повышения производительности: отчет о научно-исследовательской работе / Окунева Д. В., Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2023. – 125 с. – НИР шифр «Мульти-ЦОД». Этап 8. – Государственный контракт № ПЗ3-1-26/5. – № гос. регистрации 121032500023-5.

142. Экспериментальные исследования разработанных решений. Обобщение и оценка результатов исследований в части ЭО модуля экспертной системы. Разработка технических предложений по разработке платформы СХД:

отчет о научно-исследовательской работе / Окунева Д. В., Швидкий А. А., Гольдштейн А. Б., Савельева А. А., Редругина Н. М. [и др.]; СПбГУТ. – Санкт-Петербург, 2023. – 154 с. – НИР шифр «Мульти-ЦОД». Этап 9. – Государственный контракт № П33-1-26/5. – № гос. регистрации 121032500023-5.

143. Wang H., Li J., Zhang H., Zhou Y. Benchmarking Replication and Consistency Strategies in Cloud Serving Databases: HBase and Cassandra // Big Data Benchmarks, Performance Optimization, and Emerging Hardware / ed. by J. Zhan, R. Han, C. Weng. – Cham: Springer International Publishing, 2014. – P. 71-82. – DOI: 10.1007/978-3-319-13021-7_6.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей работе применяются следующие основные определения, обозначения и сокращения:

- 5G – (Fifth Generation) пятое поколение систем мобильной связи, соответствующее концепции международной подвижной электросвязи IMT-2020
- 6G – (Sixth Generation) шестое поколение систем мобильной связи, развиваемое в рамках концепции международной подвижной электросвязи IMT-2030
- AFR – (Annualized Failure Rate) годовой коэффициент отказов, используется как вероятность отказа накопителя на годовом интервале
- ARR – (Annual Replacement Rate) годовая доля замен устройств, применяется при оценке надежности SSD/NVMe-накопителей, используемых как журнальные устройства
- CDN – (Content Delivery/Distribution Network) сеть доставки контента, пример архитектуры, ориентированной преимущественно на обслуживание запросов чтения из ближайшей точки
- CPU – (Central Processing Unit) центральный процессор, компонент узла, влияющий на доступность и производительность локальной цепочки обслуживания.

ETSI	– (European Telecommunications Standards Institute) Европейский институт телекоммуникационных стандартов
HDD	– (Hard Disk Drive) жесткий магнитный диск, тип накопителя, используемый в составе узла хранения
IDC	– (International Data Corporation) аналитическая организация
IEEE	– (Institute of Electrical and Electronics Engineers) международная организация стандартизации
IMT-2030	– (International Mobile Telecommunications-2030) рамочное представление о системах связи следующего поколения, включающее интеграцию коммуникаций, вычислений и данных
IOPS	– (Input/Output Operations Per Second) число операций ввода-вывода в секунду, показатель предельной или измеряемой производительности СХД
IoT	– (Internet of Things) Интернет вещей, один из классов сервисов, формирующих значительные потоки данных на периферии сети
ISO/IEC	– (International Organization for Standardization / International Electrotechnical Commission) международные организации стандартизации
ITU-R	– (Radiocommunication Sector of the International Telecommunication Union) сектор радиосвязи МСЭ
ITU-T	– (Telecommunication Standardization Sector of the International Telecommunication Union) сектор стандартизации электросвязи МСЭ
IQ	– (Join-Idle-Queue) стратегия распределения, при которой сведения о простаивающих серверах используются для снижения служебного обмена при диспетчеризации

JSQ	– (Join-the-Shortest-Queue) присоединение к кратчайшей очереди, динамическая стратегия выбора узла по минимальной длине очереди обслуживания
MEC	– (Multi-access Edge Computing) многоабонентские граничные вычисления, архитектурный подход к размещению вычислительных, сетевых и запоминающих ресурсов ближе к пользователю
MTBF	– (Mean Time Between Failures) средняя наработка между отказами, используется для оценки надежности ремонтпригодных компонентов
MTTF	– (Mean Time To Failure) средняя наработка до отказа, для неремонтируемых компонентов используется как близкий по смыслу показатель при допущении постоянной интенсивности отказов
MTTR	– (Mean Time To Repair/Recovery) среднее время восстановления после отказа до возвращения узла или компонента в штатную работу
NIST	– (National Institute of Standards and Technology) Национальный институт стандартов и технологий США
NVMe	– (Non-Volatile Memory Express) интерфейс доступа к энергонезависимой памяти, рассматривается как возможная основа журнального устройства
O_SYNC	– режим синхронной фиксации данных при операции ввода-вывода, подтверждение возвращается после завершения записи на соответствующем носителе
QoS	– (Quality of Service) качество обслуживания, совокупность эксплуатационных характеристик сервиса, включая задержку, пропускную способность и доступность

RAID	– (Redundant Array of Independent Disks) массив независимых дисков с избыточностью, избыточность формируется преимущественно на уровне дисков внутри локальной системы хранения
RAIN	– (Reliable Array of Independent Nodes) массив независимых узлов с избыточностью, избыточность формируется за счет распределения данных и реплик между несколькими узлами хранения
RR	– (Round-Robin) равномерное циклическое распределение запросов по узлам без учета текущего состояния инфраструктуры
RTT	– (Round-Trip Time) время двусторонней передачи, используется как показатель сетевой задержки при взаимодействии с узлом
SLA	– (Service Level Agreement) соглашение об уровне обслуживания
SSD	– (Solid-State Drive) твердотельный накопитель, может использоваться как накопитель данных или журнальное устройство
МСЭ	– Международный союз электросвязи
НИР	– научно-исследовательская работа
СХД	– система хранения данных, в работе рассматривается как инфраструктурный компонент, включающий подсистему хранения и механизмы обслуживания операций ввода-вывода
Ветвь обслуживания операции записи	– обслуживание одной реплики данных на выбранном узле хранения, включая передачу данных, локальную запись и возврат подтверждения

Готовность операции записи	– вероятность того, что при заданных R и W доступно не менее W целевых узлов и операция записи может быть завершена в принятом режиме подтверждения
Избыточность	– наличие в системе дополнительных элементов или данных, не требуемых для ее функционирования в безотказном режиме, но используемых для поддержания работы и сохранения доступа к данным при отказе
Координатор операции записи	– компонент, принимающий запрос записи, выбирающий целевые узлы хранения, инициирующий передачу реплик и формирующий подтверждение клиенту
Коэффициент готовности	– показатель способности узла или сервиса находиться в состоянии, необходимом для выполнения требуемой функции, с учетом отказов и восстановления
Логический объект	– информационный массив пользователя, представляемый в модели как совокупность фрагментов хранения
Межузловой обмен	– сетевой обмен между узлами хранения, возникающий при передаче данных, репликации, возврате подтверждений, синхронизации и восстановлении
Метаданные	– служебная информация о размещении данных, состоянии узлов, доменах отказа, политике репликации и правилах выбора маршрутов доступа
Пользовательская очередь	– число пользовательских операций записи, ожидающих обслуживания на конкретном узле хранения
Реплика	– копия фрагмента хранения, размещаемая на отдельном узле хранения для обеспечения доступности и отказоустойчивости
Репликация	– механизм избыточности, при котором создаются дополнительные копии данных на разных узлах хранения

Служебная нагрузка	– нагрузка, формируемая операциями репликации, синхронизации, восстановления избыточности, перераспределения данных и обновления служебных метаданных
Служебная очередь	– число служебных операций репликации, синхронизации, восстановления или перераспределения, ожидающих обслуживания на конкретном узле хранения
Служебный межузловой трафик	– часть сетевого трафика, связанная с поддержанием репликации, согласованности, избыточности и восстановления данных.
Сохранность данных	– характеристика риска необратимой утраты или повреждения данных (в работе отделяется от готовности выполнения операций)
Узел хранения	– элемент распределенной подсистемы хранения, предоставляющий вычислительные, дисковые и сетевые ресурсы для размещения данных и выполнения операций чтения/записи
Узел хранения	– элемент распределенной подсистемы хранения, предоставляющий вычислительные, дисковые и сетевые ресурсы для размещения данных и выполнения операций чтения/записи
Фактор репликации	– параметр, определяющий число копий данных, формируемых при выполнении операции записи
Фрагмент хранения	– минимальная единица размещения данных на узлах хранения, на уровне которой выполняются репликация, согласование и восстановление
Цепочка обслуживания операции	– совокупность элементов инфраструктуры, участвующих в выполнении операции записи в целом: координатор, сетевые соединения и выбранные узлы хранения

ПРИЛОЖЕНИЕ

(обязательное)

Приложение содержит скан-копии документов, подтверждающих внедрение и практическое использование результатов, полученных в рамках настоящего исследования. Документы приведены последовательно в соответствии с приведенным ниже перечнем:

- Акт о внедрении научных результатов диссертационного исследования в учебный процесс СПбГУТ (от 19.06.2026),
- Акт о внедрении научных результатов диссертационного исследования в ООО «ВАС Экспертс» (от 27.08.2026),
- Акт о внедрении научных результатов диссертационного исследования в АО «ЦВКС «МСК-IX» (от 02.09.2026).

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,
СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ
ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)

Юридический адрес: набережная реки Мойки,
д. 61, литера А, Санкт-Петербург, 191186

Почтовый адрес: пр. Большевиков, д. 22, корп. 1,
Санкт-Петербург, 193232

Тел. (812) 3263156, Факс: (812) 3263159

<http://sut.ru>

E-mail: rector@sut.ru

ОКПО 01179934 ОГРН 1027809197635

ИНН 7808004760 КПП 784001001

ОКТМО 40909000

19.06.2026 № 105/54-99
на № _____ от _____

Утверждаю

Проректор по
научной работе



Рабин

Алексей Владимирович

Акт

о внедрении научных результатов,
полученных Савельевой Анастасией Андреевной

Комиссия в составе Маколкиной М.А., заведующей кафедрой инфокоммуникационных систем, Гольдштейна А.Б., профессора кафедры инфокоммуникационных систем и Лаюшки Н.А., заведующей лабораторией кафедры инфокоммуникационных систем составила настоящий акт о том, что научные результаты, полученные Савельевой Анастасией Андреевной, использованы:

- 1) При чтении лекций и проведении практических занятий по курсу «Основы построения инфокоммуникационных систем и сетей» (Рабочая Программа № 23.05/87-Д, утверждена Первым проректором-проректором по учебной работе А.В. Абиловым 23.06.2023), разделы программы:
 - Раздел 1. Базовые принципы инфокоммуникационных сетей
 - Раздел 3. IP-сети (в части сетей передачи данных)
 - Раздел 4. Общие принципы построения сетей (в части концепции построения сетей связи, NGN, сетей операторов связи и CDN).
- 2) При чтении лекций и проведении практических занятий по курсу «Сети связи и системы коммутации» (Рабочая Программа № 24.05/14-Д, утверждена Первым проректором-проректором по учебной работе А.В. Абиловым 02.04.2024), разделы программы:
 - Раздел 11. Понятие о качестве обслуживания
 - Раздел 16. Система с повторными вызовами
 - Раздел 18. Оценка надежности сетевых элементов
 - Раздел 20. Класс обслуживания сетевых элементов
- 3) При выполнении НИР и написании отчетов о НИР на тему «Разработка методов, моделей и средств повышения производительности и отказоустойчивости программно-определяемых СХД с использованием

средств автоматизированного контроля и анализа результатов на базе нейронной сети» (шифр «Мульти-ЦОД») по государственному контракту № ПЗ3-1-26/5 от «26» февраля 2021 г., заключенного с Россвязью (Минцифры России) в рамках национальной программы «Цифровая экономика Российской Федерации», дата начала 26.02.2021, дата окончания 13.12.2023.

В рамках выполнения НИР и в указанных дисциплинах используются следующие новые научные результаты, полученные Савельевой Анастасией Андреевной в диссертационной работе:

- модель распределения пользовательской и служебной нагрузки в распределенной подсистеме хранения данных инфраструктуры туманных вычислений, позволяющая учитывать состояние очередей, служебную нагрузку и сетевую неоднородность при выполнении операций записи,

- метод оценки готовности выполнения операции записи с репликацией данных, учитывающий фактор репликации, порог подтверждения записи и доступность элементов цепочки обслуживания операции,

- алгоритм выбора узлов хранения для выполнения операций записи, основанный на интегральной оценке состояния узлов и позволяющий повысить допустимую интенсивность потока операций записи по сравнению с базовыми стратегиями распределения нагрузки.

Зав. кафедрой ИКС,
д-р. техн. наук, доцент



М.А. Маколкина

Профессор кафедры ИКС,
д-р. техн. наук, доцент



А.Б. Гольдштейн

Зав. лабораторией кафедры ИКС



Н.А. Лаюшка



ООО "ВАС Экспертс"
191028, Санкт-Петербург,
Литейный пр. д. 26 Лит. А,
БЦ «Преображенский», офис 513
+7 (812) 313-88-54
info@vasexperts.ru



Утверждаю
Исполнительный директор
Терещенко А.А.

27.08.2026

АКТ

о внедрении результатов диссертационной работы
Анастасии Андреевны Савельевой, представленной на соискание ученой степени
кандидата технических наук

Настоящим подтверждается внедрение научных результатов диссертационной работы Савельевой А.А. на тему «Разработка модели, метода и алгоритма управления информационными потоками распределенной телекоммуникационной системы в инфраструктуре туманных вычислений» при разработке и совершенствовании программных решений ООО «ВАС Экспертс», предназначенных для обработки и хранения телекоммуникационных данных, в рамках таких задач, как распределение информационных потоков, организация распределенной обработки данных и выбор вариантов взаимодействия узлов инфраструктуры.

При выполнении указанных работ использованы следующие научные результаты диссертационной работы:

- модель управления информационными потоками пользовательской и служебной нагрузки в распределенной подсистеме хранения данных, позволяющая учитывать состояние очередей, объем служебной нагрузки и неоднородность сетевых характеристик при выполнении операций записи;

— алгоритм выбора узлов хранения для выполнения операций записи на основе интегральной оценки их состояния, учитывающий совокупность параметров функционирования узлов при распределении потоков.

Использование указанных результатов позволило осуществлять комплексный учет текущего состояния элементов распределенной телекоммуникационной инфраструктуры при анализе вариантов обработки и хранения данных и обосновании технических решений по распределению нагрузки между узлами, что подтверждает практическую значимость полученных результатов.

Директор по продукту



Белобородов С. Г.

Начальник группы
технической поддержки и внедрения



Чехутский В. С.



**Акционерное общество
«Центр взаимодействия компьютерных
сетей «МСК-IX»**

Российская Федерация,
127083, город Москва, улица 8 Марта, дом 1,
строение 12, пом. XXXV, ком. 19
Тел.: +7 (495) 737-92-95
Email: info@msk-ix.ru

№ МСК-03/884/1 от 02.09.2026

на № _____

Утверждаю
Технический директор
АНО «ЦВКС» «МСК-IX»

Ильин А.С.



АКТ

о внедрении научных результатов диссертационной работы
Савельевой Анастасии Андреевны

Настоящим актом подтверждаем, что научные результаты диссертационной работы Савельевой Анастасии Андреевны на тему «Разработка модели, метода и алгоритма управления информационными потоками распределенной телекоммуникационной системы в инфраструктуре туманных вычислений», представленной на соискание ученой степени кандидата _____ технических наук, в настоящее время используются АО «ЦВКС «МСК-IX» при выполнении работ по анализу и обоснованию технических решений в области развития географически распределенной телекоммуникационной инфраструктуры и сервисных платформ Общества, включая задачи размещения функций хранения и обработки данных и распределения информационных потоков между узлами инфраструктуры.

В частности, используются результаты моделирования пользовательской и служебной нагрузки в распределенных подсистемах хранения данных, а также положения метода оценки готовности выполнения операции записи с репликацией данных. Полученные результаты применены при анализе влияния параметров репликации, служебного межузлового обмена и характеристик элементов инфраструктуры на производительность и готовность выполнения операций записи, что позволило учитывать указанные факторы при выборе режима подтверждения записи и обосновании соответствующих технических решений.

Практическая значимость указанных результатов заключается в возможности их применения при проектировании и настройке распределенных программно-определяемых подсистем хранения данных, в том числе для обоснования параметров репликации, режима подтверждения и распределения операций записи.

Акт выдан для представления в диссертационный совет.

Руководитель DNS проектов,
к.т.н., доцент

П.Б.Храмцов