

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

Дмитриева Юлия Сергеевна

**РАЗРАБОТКА МОДЕЛИ И МЕТОДИКИ РАСПРЕДЕЛЕНИЯ
ТРАФИКА В МУЛЬТИКОНТРОЛЛЕРНЫХ СЕТЯХ SDN С УЧЁТОМ
МУЛЬТИПАРАМЕТРИЧЕСКИХ ТРЕБОВАНИЙ КАЧЕСТВА
ОБСЛУЖИВАНИЯ**

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.....	3
ГЛАВА 1. Анализ методов балансировки нагрузки в сетях.....	11
1.1. Архитектура SDN.....	11
1.2. Подходы к качеству обслуживания.....	16
1.2.1. Параметры SLA	16
1.2.2. Требования к сетевым показателям качества для служб, основанных на протоколе IP.....	18
1.2.3. Стандартизированное сопоставление 5QI с характеристиками QoS.....	21
1.2.4. Примеры договоров российских поставщиком услуг.....	24
1.3. Анализ методов балансировки трафика на основе параметров QoS.....	27
1.4. Обоснование выбора алгоритма управления ресурсами на основе параметров QoS.....	34
1.5. Выводы по главе 1.....	39
ГЛАВА 2. РАЗРАБОТКА МЕТОДА КОМПЛЕКСНОЙ ОЦЕНКИ В ПАРАМЕТРОВ QoS В СЕТЯХ SDN.....	42
2.1. Протоколы взаимодействия.....	42
2.2. Сценарий взаимодействия коммутаторов и контроллеров.....	45
2.3. Исследуемые сетевые характеристики в параметрах контроллера.....	50
2.4. Исследуемые сетевые характеристики в параметрах коммутатора.....	51
2.5. Модель расчета параметра комплексной оценки QoS.....	57
2.6. Выводы по главе 2.....	65
ГЛАВА 3. МОДЕЛЬ БАЛАНСИРОВКИ НАГРУЗКИ В МУЛЬТИКОНТРОЛЛЕРНОЙ СЕТИ SDN С УЧЕТОМ ПАРАМЕТРОВ QoS.....	67
3.1. Структурная схема балансировки нагрузки контроллеров	67
3.2. Разработка математической модели балансировки трафика.....	68
3.3. Математическая модель балансировки по параметру нагрузки.....	72
3.4. Математическая модель балансировки по параметрам QoS.....	75
3.5. Выводы по главе 3.....	84
ГЛАВА 4. МЕТОДИКА МНОГОКРИТЕРИАЛЬНОЙ БАЛАНСИРОВКИ ТРАФИКА В SDN.....	87
4.1. Принципы работы с комбинированным параметром QoS.....	87
4.2. Методика балансировки трафика коммутаторов по комбинированному параметру QoS.....	88
4.3. Аprobация методики.....	91
4.4. Экспериментальная апробация.....	97
4.5. Выводы по главе 4.....	103
ЗАКЛЮЧЕНИЕ.....	105
СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ.....	107
СПИСОК ЛИТЕРАТУРЫ.....	110
ПРИЛОЖЕНИЯ	121

ВВЕДЕНИЕ

Актуальность темы диссертации. Развитие телекоммуникационных систем характеризуется стремительным ростом объёмов, передаваемых данных, появлением новых классов сервисов и ужесточением требований к качеству их обслуживания. Особенно это касается приложений реального времени, таких как голосовая и видеосвязь, потоковое вещание, онлайн-игры, промышленный Интернет вещей и системы беспилотного транспорта, которые критически чувствительны к сетевой задержке, вариации сетевой задержки и потерям пакетов. Параллельно с этим происходит фундаментальная трансформация сетевых архитектур в сторону программно-конфигурируемых сетей (SDN), предлагающих централизованное, гибкое и программируемое управление. Однако внедрение SDN, особенно в виде мультиконтроллерных архитектур, необходимых для обеспечения масштабируемости и отказоустойчивости, выявило новые сложные задачи в области управления трафиком. Огромное количество трафика базируется сегодня на идеологии SDN сетей и, потому, вопрос изучения балансировки нагрузки, позволяющий повысить контролируемость и качество обслуживания в этих сетях является актуальной задачей.

Существующие методы и алгоритмы балансировки нагрузки в сети SDN фокусируются на улучшении лишь одного или нескольких параметров, таких как загрузка каналов или время отклика, не обеспечивая комплексного учёта всего спектра параметров качества обслуживания (QoS). Это приводит к ситуациям, когда формально нагрузка распределена равномерно, но ключевые для пользователей показатели — сетевая задержка, вариация сетевой задержки и процент потерь — остаются неудовлетворительными. Более того, разнородность этих параметров, измеряемых в разных единицах и имеющих различную природу, создаёт методологическую проблему их совместного использования в рамках единого алгоритма принятия решений. Отсутствие эффективных механизмов

нормализации и взвешивания разнородных метрик QoS является существенным научно-техническим пробелом.

Важнейшим практическим требованием, усугубляющим актуальность проблемы, является необходимость гарантированного выполнения соглашений об уровне обслуживания, заключаемых между поставщиками услуг и их клиентами. Анализ договоров крупнейших российских операторов связи показывает чёткую дифференциацию параметров QoS для различных классов трафика: Real Time, Best Effort. Современные алгоритмы балансировки, как правило, не адаптированы для работы с такими многоуровневыми политиками качества, что не позволяет операторам в полной мере выполнять свои обязательства и эффективно управлять ресурсами. В условиях жёсткой конкуренции на телекоммуникационном рынке это ведёт не только к снижению лояльности абонентов, но и к прямым финансовым потерям.

Таким образом, возникает необходимость в разработке новой модели и методики распределения трафика в мультиконтроллерных сетях SDN, которая интегрирует многопараметрический подход к оценке QoS, механизмы приведения разнородных параметров к единому виду, ориентированная на выполнение SLA. Решение этой задачи позволит существенно повысить эффективность использования современной и перспективной сетевой инфраструктуры, обеспечить требуемое качество обслуживания для современных приложений и создать технологический задел для сетей следующего поколения (5G/6G) и гетерогенных сетей на базе SDN. Всё это подтверждает высокую научную и практическую актуальность темы диссертационного исследования, направленного на создание комплексной методики распределения трафика с учётом мультипараметрических требований качества обслуживания.

В диссертационной работе разрабатывается модель и методика распределения трафика в мультиконтроллерных сетях SDN, позволяющая за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей качества обслуживания по сравнению с традиционной нагрузочной балансировкой.

Степень разработанности темы. В настоящее время проблемам, связанным с анализом функционирования, управления качеством обслуживания в сетях SDN, посвящено значительное количество научных работ. Большое внимание уделяется вопросам, связанным с разработкой имитационных моделей ПКС. Существенный вклад в решение задач анализа, проектирования и развития ПКС внесли ученые отечественные ученые Кучерявый А.Е., Киричек Р.В., Смелянский Р.Л., Самуйлов К.Е., Гайдамака Ю.В., Гольдштейн Б.С., Кучерявый Е.А., Пармонов А.И., А.С.А. Мутханна, Волков А.Н., Бахарева Н.Ф., Владыко А.Г., Елагин В.С., Карташевский В.Г., Леохин Ю.Л., Перепелкин Д.А., Соколов Н.А., Степанов С.Н., Тарасов В.Н., Шалимов И.А., Шелухин О.И. Среди зарубежных следует отметить работы ученых Ian F. Akyildiz, Anderson T., Leland W.E., McKeown N., Min Luo, Norros I., Parulkar G., Peterson L., Rexford J., Salman O., Sherwood R., Tootoonchian A., Pu Wang, Willinger W, Zhao Y.

Работы отмеченных выше авторов внесли весомый вклад в исследования и управление качеством обслуживания в сетях.

Таким образом задача разработки методов анализа параметров качества обслуживания трафика различных приложений в ПКС является весьма актуальной.

Объектом исследования является мультиконтроллерная программно-конфигурируемая сеть.

Предметом исследования являются сетевые характеристики и параметры распределения трафика в мультиконтроллерной программно-конфигурируемая сети.

Цель и задачи диссертации. Целью диссертации является разработка модели и методики распределения трафика в мультиконтроллерной сети SDN с учетом нормализации разнородных параметров QoS и их комбинированной оценки с целью улучшения показателей качества сервиса.

Цель диссертационной работы достигается путем последовательного решения следующих задач:

– Анализ современных методов балансировки нагрузки в сетях SDN с акцентом на учёт параметров качества сервиса, включая обзор российских и

зарубежных исследований, рекомендаций (ITU-T E.860, ITU-T Y.1541, 3GPP TS 23.501) и SLA российских операторов связи;

– Выявление недостатков существующих методов балансировки, такие как отсутствие многопараметрического подхода к QoS, игнорирование нормализации разнородных метрик и требований SLA для различных классов трафика;

– Разработка модели балансировки нагрузки для мультиконтроллерных сетей SDN, учитывающую параметры QoS на уровне контроллеров (задержка обработки запроса, потерянные пакеты) и коммутаторов (круговая сетевая задержка, вариация сетевой задержки, потери пакетов);

– Разработка методики многокритериальной балансировки трафика на основе комбинированного параметра QoS, включая механизм нормализации разнородных параметров и систему весовых коэффициентов, адаптированных к классам обслуживания (Real Time, Best Effort);

– Апробация разработанной модели путём имитационного моделирования и сравнения эффективности балансировки по параметру нагрузки и на базе параметров QoS;

– Экспериментальная апробация методики с оценкой улучшения комбинированного параметра QoS, а также подтвердить возможность выполнения строгих требований QoS для чувствительных к качеству сервиса приложений.

Научная задача состоит в повышении эффективности балансировки трафика за счет разработки модели и методики распределения трафика в мультиконтроллерных сетях SDN на базе нормализации разнородных параметров QoS и их комбинированной оценки.

Научная новизна полученных результатов исследования работы состоит в следующем:

– Благодаря предложенному методу комплексной оценки параметров QoS в сетях SDN, основанному на сведении разнородных параметров качества контроллеров и коммутаторов к единому интегральному показателю с использованием нормализующих весовых коэффициентов, обеспечивается

улучшение параметров качества при балансировке в ПКС по сравнению с традиционными методами.

– Разработанная модель балансировки нагрузки в мультиконтроллерной архитектуре сети SDN позволяет за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей QoS по сравнению с традиционной нагрузочной балансировкой — на 22,4 % на уровне контроллеров и на 47,9 % на уровне коммутаторов.

– Применение методики динамической балансировки трафика в сетях SDN с интеграцией комбинированного параметра QoS обеспечивает распределение нагрузки и улучшает комбинированный показатель качества на 46,0 % по сравнению с традиционной нагрузочной балансировкой, что подтверждено с использованием разработанного и зарегистрированного программного обеспечения имитационного моделирования.

Теоретическая значимость диссертации заключается в существенном развитии научного фундамента в области качества обслуживания в программно-конфигурируемых сетях и, в частности, в решении задачи многокритериальной балансировки нагрузки. Работа вносит вклад в теорию телетрафика и распределения сетевых ресурсов, предлагая новые концептуальные подходы и формальные модели, которые обобщают и расширяют существующие представления.

Важнейшим теоретическим результатом является разработка модели и методики распределения трафика в мультиконтроллерных сетях SDN, позволяющая за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить минимизацию показателей QoS по сравнению с традиционной нагрузочной балансировкой. Модель преодолевает фрагментарность традиционных подходов, и устанавливает теоретические принципы оптимизации балансировки трафика SDN на основе разнородных параметров QoS. Тем самым закладывается теоретический базис для проектирования более сложных и эффективных мультиконтроллерных ПКС,

способных адаптироваться к комплексным требованиям современных приложений.

Разработанная методика и её программная реализация формируют теоретико-экспериментальную платформу для дальнейших исследований. Позволяют верифицировать гипотезы, моделировать сложные сценарии и изучать поведение сетей в контролируемых условиях, тем самым способствуя развитию методов компьютерных экспериментов и имитационного моделирования в телекоммуникационной науке.

Практическая значимость диссертации заключается в повышении качества предоставления клиентских сервисов за счёт обоснованного выбора сетевых элементов для маршрутизации.

Диссертационное исследование заключается в создании законченного прикладного инструментария, позволяющего операторам связи, системным интеграторам и разработчикам сетевого оборудования повысить эффективность использования сетевых ресурсов и обеспечить гарантированное качество обслуживания для абонентов и корпоративных клиентов. Полученные результаты доведены до уровня, пригодного для непосредственного внедрения в существующую и перспективную сетевую инфраструктуру.

Разработанная методика балансировки трафика на уровне управления в сети SDN может быть интегрирована в существующие контроллеры или использована как основа для создания собственных приложений сетевой оркестрации. Её применение позволяет операторам связи автоматизировать процесс управления трафиком и гарантировать соблюдение соглашений об уровне обслуживания для различных классов трафика.

Использование механизма нормализации и весовых коэффициентов параметров качества сервиса предоставляет простой и понятный инструмент для настройки приоритетов балансировки. Администратор сети имеет возможность гибко изменять "вес" каждого параметра (потери пакетов, круговой сетевой задержки, вариации сетевой задержки) в зависимости от текущих бизнес-задач или

заключаемых SLA-договоров, адаптируя поведение сети под конкретного заказчика или тип услуги.

Разработанная и зарегистрированная программа для ЭВМ «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования» представляет собой готовый инструмент, который может использоваться проектными институтами и операторами связи для: планирования сетевой инфраструктуры; аудита и оптимизации существующих сетей; обучения персонала.

Таким образом, практическая значимость работы подтверждается созданием конкретных инженерных решений и программных продуктов, ориентированных на решение актуальных задач операторов связи по обеспечению качества обслуживания в современных гетерогенных сетях на базе технологии SDN.

Методология и методы диссертационного исследования. Для решения поставленных в диссертации задач использовались методы теории телетрафика и теории массового обслуживания, теории вероятностей, линейной алгебры.

Положения, выносимые на защиту:

1. Метод комплексной оценки параметров QoS в сетях SDN, основанный на сведении разнородных параметров QoS контроллеров и коммутаторов к единому интегральному показателю с использованием нормализующих весовых коэффициентов.

2. Модель балансировки нагрузки в мультиконтроллерной архитектуре сети SDN позволяет за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей QoS по сравнению с традиционной нагрузочной балансировкой — на 22,4 % на уровне контроллеров и на 47,9 % на уровне коммутаторов.

3. Методика динамической балансировки трафика в сетях SDN с интеграцией комбинированного параметра QoS, обеспечивает распределение нагрузки и улучшает комбинированный показатель качества на 46,0 % по сравнению с традиционными подходами, что подтверждено экспериментальными исследованиями на модельной сети.

Степень достоверности основных результатов подтверждена результатами аналитических расчётов и имитационного моделирования, обсуждением результатов диссертационной работы на международных конференциях и семинарах, публикацией основных результатов диссертации в ведущих рецензируемых журналах.

Апробация работы. Основные результаты диссертационной работы докладывались и обсуждались на 10-ой международной научно-технической и научно-методической конференции «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (Санкт-Петербург, 24–25 февраля 2021), 11-ой международной научно-технической и научно-методической конференции «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (Санкт-Петербург, 15–16 февраля 2022), 12-ой международной научно-технической и научно-методической конференции «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (Санкт-Петербург, 28 февраля–01 марта 2023), 13-ой международной научно-технической и научно-методической конференции «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (Санкт-Петербург, 27–28 февраля 2024), на семинарах кафедры инфокоммуникационных систем СПбГУТ.

Публикации. Всего по теме диссертации опубликовано 12 работ, из них 8 статей в рецензируемых научных журналах, входящих в перечень изданий, рекомендуемых ВАК Министерства высшего образования и науки Российской Федерации. Получены 1 патент на промышленный образец и 2 свидетельства о регистрации программы для ЭВМ.

Соответствие диссертации паспорту научной специальности. Диссертационная работа соответствует следующим пунктам паспорта научной специальности 2.2.15 Системы, сети и устройства телекоммуникаций: п.2, п.5, п.10. Проведенные исследования соответствуют формуле специальности.

Личный вклад автора. Все основные результаты диссертационной работы получены автором самостоятельно. Экспериментальные исследования проведены под научным руководством автора при его непосредственном участии.

ГЛАВА 1 АНАЛИЗ МЕТОДОВ БАЛАНСИРОВКИ НАГРУЗКИ В СЕТЯХ

1.1 Архитектура SDN

Первые разработки в области стандартизации программно-конфигурируемых сетей (SDN от англ. Software-Defined Networking) в Международном Союзе Электросвязи были начаты 2009 году. В 2014 году МСЭ-Т Y.3300 была выпущена рекомендация «Framework of Software-Defined Networking» [1], в ней приведено описание эталонной архитектуры программно-конфигурируемых сетей. Архитектура SDN была предложена МСЭ-Т в документе [1] и содержащая следующие уровни: уровень ресурсов (данных) или инфраструктуры (Resource layer), уровень управления (Control layer) и уровень приложений (Application layer).

В 2011 году с участием крупнейших компаний Google, Verizon, Deutsche Telekom, Microsoft, Facebook, и Yahoo и др. [2], организована некоммерческая организация Open Networking Foundation (ONF). И уже в 2014 году эта организация определила в документе ONF TR-502: SDN Architecture [3], свои перспективы развития и видение архитектуры ПКС. Надо отметить, что протокол OpenFlow достаточно долго рассматривался как экспериментальная площадка.

В 2015 году IRTF (Internet Research Task Force) предложила представление архитектуры программно-конфигурируемых сетей в документе RFC 7426 «Software-defined networking (SDN): Layers and Architecture Terminology» [4]. В ПКС предусмотрена концепция мониторинга и измерений. Это обусловлено тем, что данные функции являются встроенными и сама технология сети SDN включает данные функции для возможности сетевого управления контроллером [3]. Полученная в результате измерений и мониторинга информация может быть использована для программной обработки и получения данных о состоянии сети.

Кроме того, в ПКС легко реализуется функция зеркалирования трафика, которая возможна на плоскость управления и на какое-либо внешнее устройство.

Программно-конфигурируемая сеть — это технология в сфере сетевых технологий, которая предоставляет администраторам сети доступ к инициализации, контролю, управлению, динамическому изменению поведения сети с помощью открытых интерфейсов и абстракции более низкого уровня [5]. Статическая архитектура традиционных сетей не обеспечивала поддержку динамических или масштабируемых вычислений, а также требований к хранению данных в большинстве современных вычислительных систем. В сети SDN благодаря разделению уровня управления от уровня данных появляется возможность централизованного, гибкого и программно управляемого контроля за сетевыми потоками, что позволяет динамически перераспределять ресурсы, оптимизировать маршрутизацию и адаптировать топологию под изменяющиеся требования трафика и приложений.

Основной целью SDN является управление сетью с помощью программного обеспечения, работающего при централизованной настройке. Сетевые коммутаторы и маршрутизаторы могут локально управлять собственными таблицами маршрутизации. Управление переадресацией трафика осуществляется с помощью протоколов плоскости управления. Традиционные сетевые протоколы не адаптируются динамически к происходящим изменениям. Сетевые компоненты в домене пересылки работают сообща и следуют аналогичным правилам в соответствии с общими стандартами. Уровень управления отделен от уровня данных. На уровне управления расположен централизованный контроллер, управляющий соединениями с хостами, доступный повсеместно, позволяет сетевым инженерам использовать эффективные передовые методы, управляя на нем программным обеспечением [6, 7].

Браун В. и Мент М. в 2014 году объявили, что SDN привлекла внимание благодаря способности решать проблему отсутствия программируемости в традиционных сетевых архитектурах, что, в свою очередь, также позволило использовать простые и быстрые методы сетевых инноваций [8]. Поскольку ПКС

обеспечивает разделение плоскости данных от плоскости управления что облегчает программную реализацию сложных сетевых приложений. Управление менее специфичным и более дешевым оборудованием с помощью программных приложений с использованием стандартизированных интерфейсов стало реальностью. Более того, появилась возможность динамического добавления новых функций в сеть в виде приложений для достижения большей гибкости. Такая же концепция применяется в операционных системах мобильных телефонов, поскольку приложения могут добавляться в систему динамически [9-11].

В 2012-2014 годах ONF, основанный в 2011, выпустил множество документов, определяющих архитектуру SDN [2].

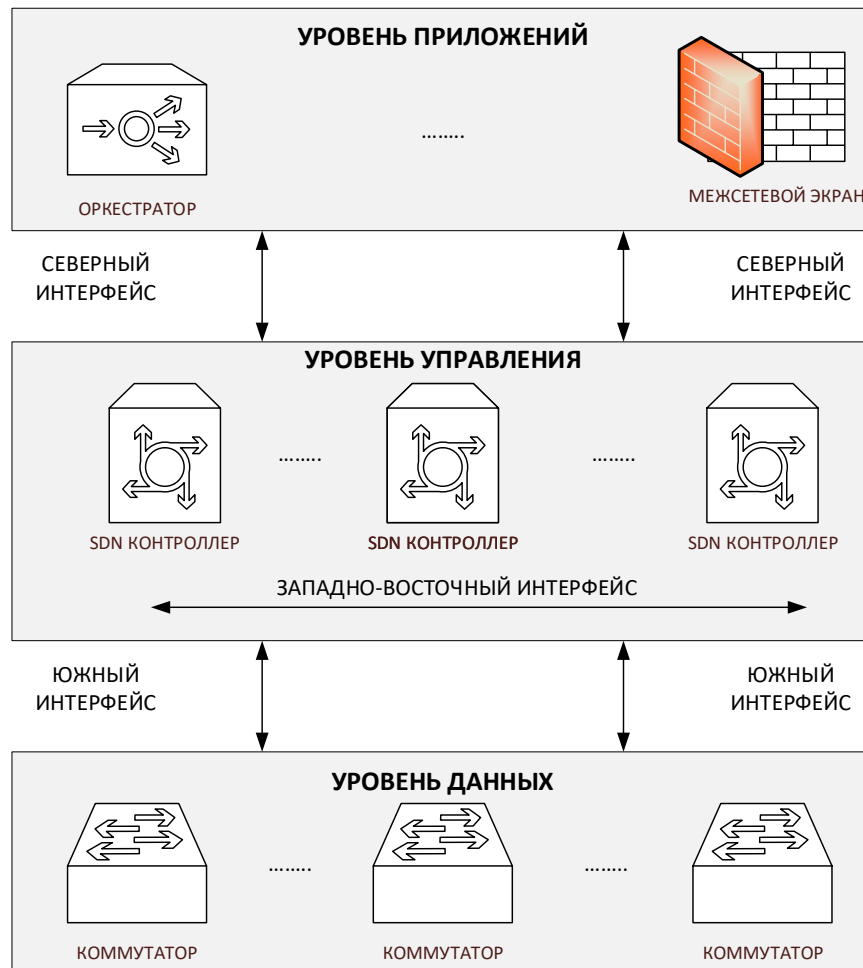


Рисунок 1.1 – Архитектура программно-конфигурируемой сети

Архитектура SDN сети изображена на рисунке 1.1, состоящая из трех уровней. Уровень данных является самым нижним уровнем, также известным как уровень инфраструктуры. Этот уровень состоит из оборудования маршрутизации: коммутаторы, маршрутизаторы. Уровень данных отвечает за пересылку данных, а также за мониторинг всей информации, хранящейся локально и сбор статистики.

Уровень над уровнем данных — это уровень управления, который также называется уровень управления. Состоит из одного или нескольких контроллеров SDN, его роль заключается в контроле и управлении инфраструктурным оборудованием коммутаторы и контроллеры через интерфейс, называемый «южным API» или южный интерфейс.

Уровень данных управляется и программируется уровнем управления. Информация, предоставляемая уровнем данных, используется для определения сетевых операций и маршрутизации. Он состоит из одного или нескольких программных контроллеров, которые могут взаимодействовать с элементами сети пересылки с помощью стандартизированных интерфейсов, также известных как южные интерфейсы.

Уровень приложений представляет собой приложения, которые позволяют добавлять новые сетевые функции, такие как управление трафиком, QoS, безопасность и т.д. [12]. Приложения создаются с помощью программного интерфейса, называемого «северным API» или северным интерфейсом [13-18].

Западно-восточный интерфейс (East-West Interface) в SDN — это механизм взаимодействия между несколькими контроллерами SDN или доменами управления, позволяющий им координировать работу по управлению распределённой или крупномасштабной сетью. Это важный компонент для промышленных сетей SDN, где один контроллер не может управлять всей сетью из-за масштаба, географического распределения или требований отказоустойчивости.

Согласно [1] цели ПКС заключаются в реализации следующих функций:

- 1) сокращение бизнес-циклов;
- 2) ускорение реализаций инноваций в ПКС;

- 3) быстрая адаптация к требованиям пользователей ПКС;
- 4) высокая доступность ресурсов и эффективность использования ПКС;
- 5) гибкая настройка сетевых ресурсов.

Преимущества сетей SDN:

1. Централизованная и программируемая сеть SDN [19-20] предоставляет необходимые инструменты для централизованного динамического управления и конфигурирования сети в соответствии с меняющимися потребностями предприятий [21-23].

2. В сети SDN задачи управления инфраструктурой, конфигурацией и развёртыванием сетевых сервисов решаются закрытыми (производитель оборудования конкретных вендоров) или открытыми инструментами автоматизации и оркестрации: OpenStack, Puppet и Chef [24].

3. В программном обеспечении контроллера SDN сетевая интеллектуальная информация централизована, то есть поддерживает глобальное сетевое представление, которое отображается для приложений и механизмов политик в виде одного логического коммутатора. [25].

4. SDN помогает организациям быстро развёртывать новые приложения, сервисы и инфраструктуру, чтобы они могли оперативно соответствовать меняющимся бизнес-целям и задачам [26].

5. SDN позволяет организациям настраивать новые типы приложений, сервисов и бизнес-моделей, которые предлагают новые источники дохода и большую сетевую ценность [27-29].

В сети SDN централизованный контроллер – это единый объект, который управляет всеми устройствами сети, представляющий единую точку отказа и имеет ограничения в масштабировании. Для управления сетью с большим количеством элементов плоскости данных одного контроллера недостаточно. Мультиконтроллерная SDN [30] в отличие от централизованной может быть масштабирована в соответствии с требованиями любой сети различного размера от небольших до крупномасштабных. При использовании распределенных

контроллеров возникает понятие западного-восточного интерфейса, позволяющих взаимодействовать элементам плоскости управления [31].

1.2 Подходы к качеству обслуживания

В условиях многоуровневых телекоммуникационных систем и взаимодействия множества поставщиков услуг обеспечение согласованного качества обслуживания (QoS) становится сложной задачей. Для её решения используются международные рекомендации, такие как ITU-T E.860 [32], вводящие принцип «единой ответственности», а также ITU-TY.1541 [33], определяющие классы QoS для IP-сетей. В контексте бурного развития сетей 5G ключевую роль играет спецификация 3GPP TS 23.501 [34], устанавливающая соответствие между идентификаторами QoS (5QI) и конкретными характеристиками обслуживания. Данные документы формируют основу для создания SLA, гарантирующих пользователям заданный уровень качества даже в сложных многопровайдерских средах.

Рассмотрим рекомендации определяющие нормы параметров качества обслуживания клиентов.

1.2.1 Параметры SLA

Рассмотрим параметры SLA на примере рекомендации ITU-T E.860 [32]. В среде с несколькими поставщиками услуг поставщик услуг SP (service provider) предоставляющий услугу клиенту использует сервисы, предоставляемые другими поставщиками услуг SP, и, становится намного сложнее обеспечить уровень QoS, указанный в SLA. Поэтому необходимо определить обязанности всех организаций,

участствующих в предоставлении услуг и скоординировать все действия для достижения согласованных уровней качества обслуживания.

Для упрощения решения упомянутой выше проблемы, применяется концепция «единой ответственности» в EURESCOM P806-GI [35] Концепция единой ответственности основана на соглашении об уровне обслуживания, заключенном между двумя организациями об условиях QoS, называемая соглашением о QoS или соглашением об уровнях обслуживания (SLA, от англ. Service Level Agreement). Обозначение используется в МСЭ-Т E.800 (09/2008) [36]. Содержание соглашения об общей ответственности и уровнях обслуживания проиллюстрировано на рисунке 1.2.

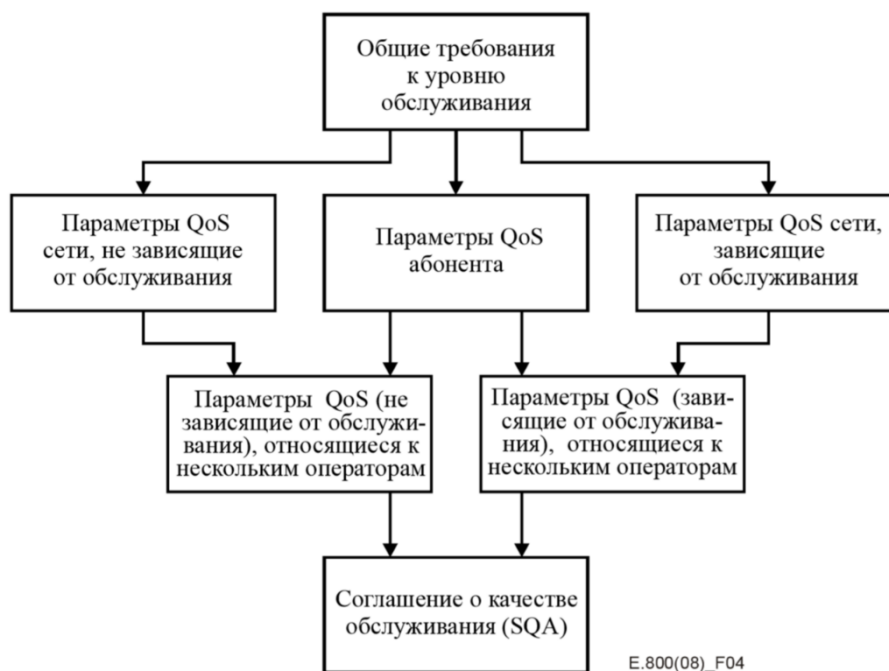


Рисунок 1.2 - Соглашение об общей ответственности и уровнях обслуживания

Единая ответственность, согласованная поставщиком услуг с клиентом в рамках соглашения об уровне обслуживания, позволяет пользователю сохранить основного поставщика услуг (с которым он согласовал соглашение об уровне обслуживания) в качестве единственного ответственного за общее получаемое качество обслуживания. В свою очередь, основной поставщик, поскольку

возникающие проблемы зависят от услуг, получаемых другими поставщиками услуг, может применить ту же единую ответственность к своим субпоставщикам. Применяя единую ответственность рекурсивным образом ко всем субъектам (поставщику и субпоставщикам), которые принимают участие в предоставлении услуги (рисунок 1.2), гарантируется услуга, согласованная с конечным пользователем.

Благодаря единой ответственности и рекурсивному применению соглашений об уровне обслуживания сложная проблема предоставления услуг в среде с несколькими провайдерами сводится к элементарным отношениям только между двумя объектами (пара пользователь/поставщик). Однако это означает, что основной поставщик услуг должен проявлять гибкость в отношении качества обслуживания своих клиентов, поскольку качество услуг (предоставляемых его субподрядчиками) может колебаться в пределах согласованных диапазонов.

1.2.2 Требования к сетевым показателям качества для служб, основанных на протоколе IP

Требования к рабочим характеристикам приложений достаточно хорошо изучены, но некоторые ключевые факторы часто находятся вне области контроля поставщиков сетевых услуг (например, домашние и локальные вычислительные сети, шлюзы приложений, оконечное оборудование, хосты и другие устройства потребителей). МСЭ-Т учитывает, что требования к рабочим характеристикам оборудования потребителей являются доступными, например, рекомендация МСЭ-Т Р.1010 [37] для оконечного оборудования и шлюзов VoIP. Соединение данных требований с определенными уровнями рабочих характеристик сетей (как показано в дополнениях к данной рекомендации) дает возможность непосредственно связать рабочие характеристики приложения и сети. В ответ на это поставщики услуг согласовали уровни рабочих характеристик сети для обеспечения совместной

работы, чтобы систематизировать и привести в соответствие свою работу многочисленным требованиям данной рекомендации. Соглашение относительно уровней рабочих характеристик сетей является весьма полезным, потому что оно ограничивает появление критических и часто преобладающих факторов в работе приложений [33].

В рекомендации МСЭ-Т Y.1541 указываются значения сетевых показателей качества (интерфейс UNI-UNI) протокола IP для каждого из параметров рабочих характеристик, определенных в рекомендации МСЭ-Т Y.1540 [38]. Конкретные значения рабочих характеристик изменяются в зависимости от классов сетевых QoS.

В рекомендации определены восемь классов сетевых QoS, два из которых являются условными. Данная рекомендация применяется к международным каналам передачи данных сетей IP (интерфейс UNI-UNI). Определенные классы сетевых QoS предназначены для того, чтобы быть основой соглашений между конечными пользователями и поставщиками сетевых услуг, а также между поставщиками сетевых услуг. Классы можно использовать, когда статические соглашения предоставляют возможность выполнения динамических запросов, которые поддерживаются протоколами спецификации QoS.

Классы QoS поддерживают очень широкий диапазон приложений, включая следующие: голосовую телефонию, мультимедийные конференции, цифровое видео и интерактивную передачу данных. Для других приложений могут потребоваться новые или пересмотренные классы, но любая потребность в новых классах должна быть сопоставлена с требованиями осуществимости реализации, также следует учитывать, что количество классов должно быть небольшим для реализации масштабирования в глобальных сетях.

В рекомендации представлены классы сетевых QoS необходимые для поддержки ориентированных на пользователя категорий QoS. Таким образом, данная рекомендация соответствует общей структуре определения качества услуг связи, описанной в рекомендации МСЭ-Т G.1000 [39], а также категориям

мультимедийного QoS конечного пользователя, необходимых для поддержки пользовательских приложений, представленных в рекомендации МСЭ-Т G.1010.

Каждый класс сетевого QoS создает определенную комбинацию ограничений для значений рабочих характеристик. Правила указывают, когда каждый из классов сетевого QoS может быть использован, но не накладывают требований на использование какого-либо конкретного класса сетевого QoS в каком-либо конкретном окружении. Классы сетевого QoS представлены в таблице 1.1.

Таблица 1.1 — Определения классов сетевого QoS протокола IP и требования к рабочим характеристикам сети

Параметр рабочей характеристики сети	Сущность требования к рабочей характеристике сети	Классы QoS					
		Класс 0	Класс 1	Класс 2	Класс 3	Класс 4	Класс 5 Неопределенный
Задержка	Верхнее ограничение значения IPTD	100 мс	400 мс	100 мс	400 мс	1 с	Н
Джиттер	Верхнее ограничение для 1 – 10 ⁻³ квантиля значения IPTD за вычетом минимального значения IPTD	50 мс	50 мс	Н	Н	Н	Н
Потери пакетов	Верхнее ограничение для вероятности потери пакета	1×10^{-3}	1×10^{-3}	1×10^{-3}	1×10^{-3}	1×10^{-3}	Н
Коэффициент ошибочных пакетов	Верхнее ограничение	1×10^{-4}					Н

1.2.3 Стандартизированное сопоставление 5QI с характеристиками QoS

В спецификации 3GPP TS 23.501 V15.4.0 (2018-12) представлено сопоставление стандартизированных значений 5QI (5G QoS Identifier) с характеристиками QoS 5G указано в таблице 1.2 [34].

Таблица 1.2 — Стандартизированное сопоставление характеристик 5QI и QoS

5QI Значение	Тип Источника	Уровень приоритета по умолчанию	Предел сетевой задержки	Норма ошибочных пакетов	Размер пакета по умолчанию	Среднее окно по умолчанию	Примеры служб
1	Гарантированная скорость передачи данных (GBR)	20	100 мс	10^{-2}	Н/Д	2000 мс	Разговорная речь
2		40	150 мс	10^{-3}	Н/Д	2000 мс	Разговорное видео (прямая трансляция)
3		30	50 мс	10^{-3}	Н/Д	2000 мс	Онлайн игры, Сообщения V2X, распределение электроэнергии—среднее напряжение, автоматика - мониторинг
4		50	300 мс	10^{-6}	Н/Д	2000 мс	Неразговорное видео (поток данных с буферизацией)
65		7	75 мс	10^{-2}	Н/Д	2000 мс	Режим рации для спецслужб (например, МСРГТ)
66		20	100 мс	10^{-2}	Н/Д	2000 мс	Режим рации для гражданских пользователей
67		15	100 мс	10^{-3}	Н/Д	2000 мс	Видео для спецслужб
75							

5	Негара нтиров анная скорос ть переда чи данных	10	100 мс	10^{-6}	Н/Д	Н/Д	Сигнализация IMS
6		60	300 мс	10^{-6}	Н/Д	Н/Д	Видео (вещание с буферизацией) на основе TCP (например, www, электронная почта, чат, ftp, p2p обмен файлами, потоквое видео и т. д.)
7		70	100 мс	10^{-3}	Н/Д	Н/Д	Голос, видео (прямая трансляция), Интерактив- ные игры
8		80	300 мс	10^{-6}	Н/Д	Н/Д	Видео (вещание с буферизацией) на основе TCP (например, www, электронная почта, чат, ftp, p2p обмен файлами, потоквое видео и т. д.)
9		90					
69		5	60 мс	10^{-6}	Н/Д	Н/Д	Сигнализация , чувствительна я к задержкам для спецслужб (например, сигнализация МС-РТТ)
70		55	200 мс	10^{-6}	Н/Д	Н/Д	Данные для спецслужб (например, примеры услуг 5QI 6/8/9)
79		65	50 мс	10^{-2}	Н/Д	Н/Д	Сообщения V2X

80		68	10 мс	10^{-6}	Н/Д	Н/Д	Приложения с низкой задержкой eMBB, дополненная реальность
82		19	10 мс	10^{-4}	255 байт	2000 мс	Дискретная автоматизация (см. TS 22.261 [40])
83		22	10 мс	10^{-4}	1354 байт	2000 мс	Дискретная автоматизация (см. TS 22.261 [40])
84		24	30 мс	10^{-5}	1354 байт	2000 мс	Дискретная автоматизация (см. TS 22.261 [40])
85		21	5 мс	10^{-5}	255 байт	2000 мс	Дискретная автоматизация (см. TS 22.261 [40])

Условные обозначения:

GBR (Guaranteed Bit Rate) — гарантированная скорость передачи данных.

Н/Д – нет данных.

Real Time Gaming – программное обеспечение для казино

V2X (Vehicle-to-Everything) — система обмена данными между автомобилем и другими объектами дорожной инфраструктуры.

MCPTT (Mission Critical Push-to-Talk) — стандарт, позволяющий организовать функционал профессиональной подвижной радиосвязи поверх сети LTE.

Packet Delay Budget (PDB) — это максимальная допустимая задержка для передачи пакетов в сети от источника к месту назначения.

1.2.4 Примеры договоров российских поставщиков услуг

Рассмотрим примеры параметров QoS в договорах крупнейших российских поставщиков услуг связи: Ростелеком, Билайн, Транстелеком, Связьтранснефть.

Приведем параметры, которые выносятся в обычный SLA и договоры с операторами связи.

- Ростелеком

Оператор гарантирует следующие эксплуатационные характеристики услуги:

- услуга предоставляется 24 часа в сутки 7 дней в неделю;
- совокупная доступность (работоспособность) магистральных сетей и оборудования узлов оператора, задействованных в предоставлении услуги, составляет не менее 99,5 % в месяц.

— гарантии качества в зоне ответственности оператора на наземном канале оператора Ростелеком представлены в таблице 1.3.

Таблица 1.3 — Гарантии качества в зоне ответственности оператора на наземном канале оператора Ростелеком

Класс сервиса	Доступность сети за месяц	Процент потерянных пакетов (PE-to-PE) в среднем за месяц	Сетевые задержки на наземных каналах (PE-to-PE) в среднем за месяц	Вариация сетевой задержки на наземных каналах (PE-to-PE) (джиттер)
Real Time	не менее 99,5%	не более 0,2%	не более 150 мс	не более 50 мс
Business Critical	не менее 99,5%	не более 0,5%	не более 200 мс	-
Best Effort	не менее 99,5%	не более 1%	более 250 мс	-

- Билайн

Параметры качества услуги оператора Билайн представлены в таблице 1.4.

Таблица 1.4 — Параметры качества услуги

Параметр	Best Effort	Business Critical	Real Time
Круговая сетевая задержка, мс	200	160	80
Вариация сетевой задержки, мс	—	—	20
Процент потери пакетов, %	1	0,2	0,1

Класс трафика Best Effort негарантированный, трафик, некритичный к задержкам и потерям пакетов, например, для просмотра веб-страниц.

Класс трафика Business Critical для данных, некритичных к задержке и незначительной потере пакетов: передача файлов, работа с электронной почтой, видеонаблюдения, видеотрансляции, обращения к базам данных, интерактивные офисные приложения (корпоративные WEB-порталы, терминальные сервисы).

Класс трафика Real Time трафик реального времени имеет абсолютный приоритет над классами Business Critical и Best Effort, за счёт чего обеспечивается минимальная задержка и неравномерность сетевой задержки.

- Транстелеком

- «круговая» сетевая задержка прохождения IP-пакета (Round trip network delay) - время, которое потребуется зондирующему ICMP (Internet Control Message Protocol)-пакету для того, чтобы достичь точки назначения и вернуться обратно между двумя узлами сети исполнителя – не более 20 мс;

- неравномерность сетевой задержки (Jitter) – разница между значениями сквозного времени задержек, возникающая при передаче серии зондирующих пакетов за определенный промежуток времени – не более 10 мс;

- процент потерь IP-пакетов (Packet Loss) – отношение разницы общего количества переданных и правильно принятых пакетов к общему количеству переданных пакетов за определенный промежуток времени. Потери выражаются в процентах - не более 0,5 %.

Загрузка порта заказчика не должна превышать 80 %. Превышение указанного значения может привести к ухудшению параметров качества канала доступа.

- Связьтранснефть

Техническое задание по параметрам услуги предоставления каналов связи:

- Односторонняя сетевая задержка, не более – 100 мс;
- Jitter односторонней задержки, не более – 35 мс;
- Потеря пакетов (в каждом направлении), не более – 0,1 %.

Проанализировав рекомендации Международного союза электросвязи и договоры российских операторов связи уровень обслуживания представляет собой соглашение между абонентом и провайдером услуг связи и может быть представлен в общем виде на рисунке 1.3.



Рисунке 1.3 — Общий вид соглашения об уровне обслуживания SLA

1.3 Анализ методов балансировки трафика на основе параметров QoS

Произведем обзор российских и зарубежных релевантных работ, посвященных вопросам балансировки трафика в сетях SDN, на предмет используемых для этого авторами параметров QoS.

В работе [41] проанализировано влияние архитектуры программно-конфигурируемых сетей и архитектуры плоскости управления на эффективность методов балансировки нагрузки. Представлен подробный анализ алгоритмов и метрик балансировки нагрузки в централизованных и распределенных архитектурах SDN. Рассмотрены механизмы балансировки нагрузки, применяемые, в частности, в сетях 5G. Проведенные исследования показывают, что балансировка нагрузки улучшает распределение нагрузки между контроллерами SDN. Конечные пользователи получают высокое качество обслуживания и удобство использования за счет сокращения задержек, времени отклика и увеличения пропускной способности.

Авторы в работе [42] заложили основу для алгоритма балансировки нагрузки нескольких распределенных контроллеров (MDCLB, от англ. multiple distributed controller load balancing) в масштабной SDN-IoT для умных городов. Исследователи предлагают алгоритм для дисбаланса нагрузки с использованием нескольких контроллеров на основе параметра загрузки центрального процессора в плоскости управления. Переменная балансировка нагрузки для контроллеров в плоскости управления минимизирует сетевую задержку и ограничивает дисбаланс нагрузки трафика. Пороговое значение используется для сравнения с нагрузкой на серверах. Если пороговое значение превышает нагрузку, то конкретный коммутатор на сервере выполняет миграцию пакетов из внутрикластерного в межкластерный контроллер.

В исследовании [43] предложен улучшенный алгоритм балансировки нагрузки и предоставления QoS. Основные результаты: коммутатор обнаруживает большие потоки данных и следует подходу с несколькими путями на основе весов

контейнеров, сгруппированных на основе пропускной способности по классам трафика QoS. Задержка и отказоустойчивость уникальны для любого класса QoS на каждом интерфейсе коммутатора. В случае, если трафик без класса QoS идет по тому же пути, что и трафик с классом QoS, трафик без класса QoS отправляется по нескольким путям на основе весового подхода вместо того, чтобы направлялся по тому же пути, который использует трафик с классом QoS.

В работе [44] предложен метод SDN-DVFS (dynamic voltage frequency scaling), для балансировки нагрузки трафика на серверах и улучшения QoS в сети, который не только предотвращает перегрузку сети, но и снижает частоту синхронизаций между контроллером и серверами.

В [45] авторы предложили подход к балансировке нагрузки для мониторинга текущих показателей нагрузки серверов и применения многопараметрических метрик для планирования подключений с целью максимально эффективной балансировки нагрузки на серверах, позволяющий эффективно выполнять балансировку нагрузки независимо от типа трафика.

В исследовании [46] предлагается механизм динамической балансировки нагрузки с несколькими SDN контроллерами, который сокращает время балансировки.

В [47] разработан механизм маршрутизации на основе многоагентного обучения с подкреплением (MARL, от англ. Multi-Agents Reinforcement Learning), использующий распределенный подход обучения на основе DQN (Deep Q-Networks) и обеспечивающий динамическую обработку QoS и эффективное управление междоменными ресурсами в крупномасштабных сетях SDN.

В обзоре статей [48] описаны свойства комбинированных сетей Интернета вещей и SDN, а также архитектура и роль SDN в IoT для удовлетворения требований к качеству обслуживания в различных направлениях. Представлен обзор литературы по балансировке нагрузки в сетях SD-IoT и их характеристик, доступные решения с их преимуществами и недостатками, соответствующие параметры QoS, а также инструменты и испытательные стенды.

Предложена интеллектуальная кластерная сеть, основанная на алгоритме k-средних в [49], которая позволяет эффективно снизить частоту потери пакетов или задержку.

Для сокращения времени отклика сервисов и равномерного распределения нагрузки между обслуживаемыми узлами в статье [50] рассматриваются методы реактивной и проактивной балансировки нагрузки в SDN в контексте сетях «машина-машина» (M2M).

Предложен новый метод QoS-маршрутизации HMLR-AC (Hybrid Multi-constraint Lagrangian Relaxation based Aggregated Cost) для SDN [51], который: оптимизирует показатели (выше пропускная способность, меньше задержка, потери пакетов и лучше масштабируемость), работает в реальном времени, адаптируясь к изменениям сети. Метод доказал эффективность в симуляциях и перспективен для Industrial IoT и 5G-сетей.

Гибридный метод PPO (Proximal Policy Optimization) [52] обеспечивает значительное сокращение сквозной задержки до 100 мс, снижение потери пакетов до 2 % и улучшенную балансировку нагрузки с использованием канала связи от 60 % до 62 %. Эффективное распределение трафика и снижение перегрузок обеспечивают более плавную масштабируемость.

Предложен улучшенный подход к управлению потоками и балансировке нагрузки для промышленного Интернета вещей с поддержкой SDN (EFLB-IoT, от англ. Enhanced Flow Control and Load Balancing) [53]. EFLB-IoT — это основанный на аппроксимации метод, который эффективно поддерживает сетевую активность между распределенными контроллерами. Результаты моделирования показывают, что предложенная стратегия снижает максимальную нагрузку на каналы связи на 76 % и максимальное время отклика контроллера на 85 % по сравнению с существующими методами, демонстрируя превосходную производительность по сравнению с современными методами.

В данном исследовании [54] проанализировано несколько работ, в которых приложения высокопроизводительных вычислений и обработки больших объемов данных используют программно-конфигурируемую сеть для планирования своих

задач и заданий. Выявлено ряд проблем, решению которых следует посвятить новые исследования, связанные с динамическим и специализированным обеспечением качества обслуживания для приложений высокопроизводительных вычислений и обработки больших объемов данных.

В статье [55] выполнен обзор научных работ российских и зарубежных авторов для выявления применяемых в них решений, а также их достоинств и недостатков. Исходя из сравнительного анализа работ сделано предположение, что качественное новое решение может быть найдено в сети на основе намерений, работающей на сетевом, логическом и физическом уровнях сети SDN до наступления инцидента. В такой сети можно непрерывно проверять соответствие между фактическим и ожидаемым состояниями сети с помощью мониторинга ключевых показателей эффективности сети в режиме реального времени, что позволит корректировать параметры сети для бесперебойной работы и обеспечения высокого качества предоставляемых сервисов.

В работе [56] рассматривается возможность балансировки нагрузки в программно-конфигурируемых сетях путем кластеризации контролеров. Результаты моделирования показали, что представленный метод балансировки нагрузки не только эффективен, но и помогает оптимизировать ресурсы сети, а также улучшать QoS.

В исследовании [57] рассмотрены методы обеспечения QoS с помощью протокола OpenFlow, выделены проблемы при работе с оборудованием, имеющим различные сценарии конфигурирования. Рассмотрен контроллер SDN Ryu и REST API. Предложено решение на основе модуля контроллера и драйверов для соответствующих сетевых устройств, работа с которым осуществляется с помощью REST API. Выделены достоинства и недостатки подхода.

В работе [58] предложена потоковая модель адаптивной маршрутизации для программно-конфигурируемых сетей с балансировкой нагрузки по двум критериям. Установлено, что использование предложенной модели позволяет улучшить значения многопутевой задержки в среднем от 10–19 до 35–48 %. Вероятность доставки пакетов при решении задачи маршрутизации в рамках

предложенной модели оставалась практически на уровне значений полученных в рамках ранее известных моделей. Кроме этого, полученные в результате решения задачи маршрутизации одинаковые значения средних задержек вдоль путей способствуют минимизации джиттера пакетов, обусловленного реализацией многопутевой стратегии маршрутизации. Выигрыш по средней многопутевой задержке при вероятности своевременной доставки, равной 0,999, полученный при решении задачи маршрутизации в рамках предлагаемой модели, по сравнению с решением в рамках других моделей составил 12-38 % в зависимости от интенсивности поступающего в сеть трафика.

В таблице 1.5 приведены результаты сравнительного анализа подходов к балансировке нагрузки в сетях SDN

Таблица 1.5 — Результаты сравнительного анализа подходов к балансировке нагрузки в сетях SDN

Ссылка	Подходы к балансировке нагрузки	Метрики оценки
[41]	- Статические (Random, Round Robin, Weighted Round Robin) — простые, но не адаптивные; - Динамические (Least Connection, на основе времени отклика, искусственного интеллекта) — эффективнее, но более ресурсоемкие.	Задержка (latency), время отклика (response time), потеря пакетов (packet loss), пропускная способность (throughput), использование ресурсов
[42]	Алгоритм MDCLB: - Динамическая балансировка на основе порогового значения нагрузки; - Миграция нагрузки из перегруженного кластера в менее загруженный; - Учитывается задержка передачи и обработки пакетов; - Используется концепция Nash bargaining для минимизации задержек.	Задержка, потери пакетов.
[43]	Выбор непересекающихся узлов для вычисления резервных путей. В случае с большими потоками данных перегрузка одного пути предотвращается за счет распределения трафика между несколькими путями.	Пропускная способность, задержка и надежность, эффективная маршрутизация узлов и каналов, а также обработка различных требований QoS с учетом максимальных усилий (best-effort).
[44]	Метод SDN-DVFS (dynamic voltage frequency scaling). Учитывает перегрузку виртуальной машины, эффективность хост-машины и нагрузку, прилагаемую пользователем.	Задержка освобождения трафика и пропускной способности контроллеров как частица в алгоритме

		PSO (Particle Swarm Optimization).
[45]	Объединили проактивные потоки, которые оставляют большую часть управления компьютерной сетью традиционным механизмам, с добавленными реактивных потоков, используемых для дополнительных функций SDN. Применили многопараметрические метрики для выбора сервера для маршрутизации.	- Загрузка CPU - I/O Read / Write - Link Upload / Download
[46]	Динамической балансировки нагрузки для контроллера с несколькими SDN.	Время отклика, пропускная способность, потребление ресурсов.
[47]	Механизм маршрутизации на основе многоагентного обучения с подкреплением MARL (Multi-Agents Reinforcement Learning), использующий Deep Q-Network (DQN)	Пропускная способность, джиттер, потери пакетов
[48]	Интеграция SDN и IoT (SD-IoT).	Доступность, отказоустойчивость, надёжность.
[49]	Интеллектуальная кластерная сеть, основанная на алгоритме k-средних.	Потери пакетов, задержка.
[50]	Методы реактивной и проактивной балансировки нагрузки в SDN.	Сокращение времени отклика сервисов и равномерное распределение нагрузки между обслуживаемыми узлами.
[51]	Гибридная маршрутизация сегментов на основе многоограниченной лагранжевой релаксации (HMLR-AC, Hybrid Multi-constraint Lagrangian Relaxation-based Aggregated Cost) для маршрутизации с учетом качества обслуживания (QoS).	Пропускная способность, задержка, надёжность.
[52]	Метод адаптивной балансировки нагрузки на основе гибридной оптимизации проксимальных политик для решения проблем динамических моделей трафика и угроз безопасности в SDN-IoT.	Использование канала, длина очереди, задержка, потеря пакетов.
[53]	EFLB-IIoT (Enhanced Flow Control and Load Balancing Approach for SDN-Enabled Industrial Internet-of-Things) улучшенный подход к управлению потоками и балансировке нагрузки для промышленного Интернета вещей с поддержкой SDN.	Нагрузка на каналы связи, время отклика контроллера.
[54]	Q-Ctrl, OpenQoS, Flow QoS, QoS control, VSDN (Video over Software-Defined Networking), CECT (Computationally Efficient Congestion-avoidance and Traffic Engineering), Amoeba Net (сетевой сервис с поддержкой SDN для анализа больших данных).	Пропускная способность, задержка, джиттер.

[55]	СМО – оригинальная авторская модель на базе СМО, CPN (аббр. от англ. Coloured Petri Net) – модель графа раскрашенной сети Петри, К-CP – модель K-Center Problem, ТИ – модели из теории игр, ГМ* – оригинальная авторская модель на основе теории графов, RR – Round Robin.	-
[56]	Кластеризации контроллеров.	Параметры QoS не конкретизированы.
[57]	Метод управления качеством обслуживания в SDN с учетом проблем неполной поддержки протоколов со стороны производителей оборудования и специфичности его настройки.	Параметры QoS не конкретизированы.
[58]	Потоковая модель адаптивной маршрутизации для SDN с балансировкой нагрузки по двум критериям, ориентированной на улучшение показателей QoS.	Коэффициент максимальной загрузки каналов, критерий адаптации процесса балансировки нагрузки, задержка, доставка пакетов.

Проанализировав существующие подходы к балансировке трафика в сети SDN выделены следующие недостатки:

- Отсутствует многопараметрический подход к QoS, где для контроллеров учитываются параметры: задержка обработки запроса, потерянные пакеты [48], для коммутаторов: потери пакетов, круговая сетевая задержка, вариация сетевой задержки [33]. В исследованиях, посвященных балансировке, где рассматривается QoS, качество сервиса — это приоритизация трафика или ее отсутствие.
- Рассмотренные подходы не учитывают нормализацию разнородных параметров QoS и не решают проблему разных единиц измерения параметров QoS через систему нормализации и весовых коэффициентов параметров QoS.
- Рассмотренные подходы не учитывают критерии SLA к разным классам сервиса: Real Time, Business Critical, Best Effort.
- В рассмотренных исследованиях описаны подходы к балансировке на стороне сервис-провайдеров и дата-центров, т.е. рассматривают только серверную инфраструктуру, не принимая во внимание абонентские сети SDN.

Для решения вышеперечисленных проблем предлагается метод комплексной оценки параметров QoS в сетях SDN на основе приведения параметров QoS

различных типов трафика к комбинированному показателю для применения в алгоритмах балансировки.

1.4 Обоснование выбора алгоритма управления ресурсами на основе параметров QoS

Балансировка трафика в сетях SDN по параметрам QoS возможна в алгоритмах, которые используют веса параметров.

Рассмотрим основные особенности и этапы балансировки трафика в SDN и проанализируем критерии и алгоритмы, применяемые в современных исследованиях, а также произведем сравнительную оценку для выбора наиболее эффективного алгоритма балансировки [59].

Ключевые особенности балансировки в SDN:

- **Централизованный Интеллект:** решения принимаются на основе глобального состояния, а не локального на каждом устройстве.
- **Гибкость:** критерии легко комбинируются и меняются программно.
- **Динамичность:** быстрая реакция на изменения [60, 61].
- **Программируемость:** алгоритмы балансировки — это программное обеспечение, которое можно разрабатывать, тестировать и развертывать независимо от оборудования [62].
- **Адаптивность:** возможность использования сложных адаптивных алгоритмов, включая машинное обучение.

Рассмотрим этапы балансировка трафика в SDN:

1. **Мониторинг:** контроллер (или специализированные приложения на нем) собирает статистику (через OpenFlow [63], NetFlow/sFlow, SNMP, gRPC) по всем ключевым критериям (загрузка линков, задержка, потери и т.д.).

2. Анализ: на основе глобального представления анализируется состояние сети и требования приложений/политик.

3. Принятие решения: алгоритм балансировки (реализованный как приложение контроллера) вычисляет оптимальные пути для потоков (flows) или групп потоков на основе заданных критериев и их приоритетов. Алгоритмы распределения трафика:

- Простые эвристики (наименьшая загрузка, наименьшая задержка);
- Сложные алгоритмы (на основе теории графов, линейного программирования).
- Алгоритмы машинного обучения (предсказание нагрузки, проактивная балансировка).

4. Программирование: контроллер устанавливает или изменяет правила потоков (Flow Entries) на коммутаторах через протоколы Южного интерфейса (например, OpenFlow), перенаправляя трафик согласно вычисленному плану балансировки [64].

5. Адаптация: процесс непрерывный. При изменении состояния сети (отказ линка, всплеск трафика) контроллер пересчитывает балансировку.

В SDN (Software-Defined Networking) балансировка трафика — это динамический процесс, управляемый контроллером, который принимает решения на основе глобального представления о состоянии сети. Критерии балансировки более гибкие и комплексные, чем в традиционных сетях. Произведен анализ релевантных работ, посвященных вопросам управления ресурсами, на предмет используемых для этого авторами методов и алгоритмов балансировки нагрузки [65-70]. Произведем аналитический обзор работ по критериям.

Рассмотрим основные критерии, применяемые при балансировке трафика в SDN:

1. Критерии производительности (Performance Metrics):

- Загрузка каналов (Link Utilization) – основной критерий. Контроллер стремится минимизировать перегрузку на отдельных линках, распределяя трафик по менее загруженным путям.

- Задержка (Latency) критична для приложений (VoIP, онлайн-игры, биржевые торги) чувствительных к задержкам приложений. Трафик направляется по путям с наименьшей задержкой.

- Джиттер (Jitter) – вариация задержки. Минимизируется для потокового трафика (видео, аудио).

- Потери пакетов (Packet Loss) сигнализируют о перегрузке или проблемах.

- Пропускная способность (Bandwidth) – обеспечение выделенной или гарантированной полосы для определенных потоков или классов трафика.

2. Критерии качества обслуживания (QoS – Quality of Service):

- Тип трафика (Traffic Class): разные алгоритмы и пути для разных классов трафика (Best Effort, Business Critical, Real Time).

- Соглашения об уровне обслуживания (SLA, Service Level Agreement): Балансировка должна обеспечивать выполнение SLA (минимальная задержка, потери, вариация сетевой задержки и т.д.) для конкретных клиентов или приложений.

3. Критерии эффективности системы (System Efficiency):

- Загрузка контроллера (Controller Load): Избегание сценариев, где слишком частая перебалансировка или обработка большого числа мелких потоков перегружают контроллер.

- Загрузка коммутаторов (Switch Load): распределение нагрузки не только по линкам, но и по вычислительным ресурсам коммутаторов.

- Масштабируемость (Scalability): алгоритм балансировки должен эффективно работать в больших сетях, не создавая чрезмерной нагрузки на контроллер и управляющую плоскость.

- Время реакции:

- Real time трафик – трафик приложений, чувствительных ко времени. Обычно это голосовой или видеотрафик.

— Not real time трафик – трафик, пересылаемый без специальной расстановки приоритетов. Это приоритет «по умолчанию» для трафика, который не классифицирован как другой тип.

- Ресурсоёмкость зависит от архитектуры сети, алгоритмов балансировки, масштаба развертывания и используемых технологий.
- Сложность реализации зависит от выбранного подхода, масштаба сети и требований к производительности.

Выбор конкретных критериев и их весов зависит от целей сети, типа приложений, политик администратора и возможностей контроллера и приложений SDN. Часто используется комбинация нескольких критериев для достижения оптимального результата.

Проанализированы релевантные нашей задаче алгоритмы балансировки (Round Robin (RR), Weighted Round Robin (WRR), Enhanced Weighted Round Robin (EWRR), Least Connections, W Least Connections (Weighted Least Connections), Deep QN (Deep Q-Network)) трафика, которые используют веса параметров. Рассмотрены актуальные публикации российских и зарубежных авторов. Результаты сравнительного анализа представлены в таблице 1.6, где 5 – максимальный балл оценки критерия, 0 – минимальный балл оценки критерия. (Таблица 1.6, Таблица 1.7, Таблица 1.8)

Таблица 1.6 — Результаты сравнительного анализа алгоритмов балансировки трафика по критерию производительность

Название алгоритм	Критерии балансировки трафика				
	Производительность				
	Загрузка каналов	Круговая сетевая задержка	Вариация сетевой задержки	Потери пакетов	Пропускная способность
Round Robin	0	0	0	0	0
Weighted Round Robin	5	5	5	5	5
Enhanced Weighted Round Robin	5	5	5	5	5

Least Connections	0	0	0	0	0
Weighted Least Connections	0	0	0	0	0
Deep QN	5	5	0	5	5

Таблица 1.7 — Результаты сравнительного анализа алгоритмов балансировки трафика по критерию качество обслуживания

Название алгоритм	Критерии балансировки трафика	
	Качество обслуживания	
	Тип трафика	SLA
Round Robin	0	0
Weighted Round Robin	5	5
Enhanced Weighted Round Robin	5	5
Least Connections	0	0
Weighted Least Connections	0	0
Deep QN	0	0

Таблица 1.8 — Результаты сравнительного анализа алгоритмов балансировки трафика по критерию эффективность системы

Название алгоритм	Критерии балансировки трафика					
	Эффективность алгоритма					
	Загрузка Контроллера	Загрузка коммутаторов	Масштабируемость	Время реакции (Real time или Not Real time)	Ресурсоёмкость	Сложность реализации
Round Robin	1	1	5	Non real time	5	5
Weighted Round Robin	4	4	5	Non Real time	5	5
Enhanced Weighted Round Robin	5	5	5	Real time	5	5
Least Connections	0	0	5	Real time	0	5

Weighted Least Connections	0	0	5	Real time	1	5
Deep QN	5	5	5	Non real time	0	0

Результаты сравнительного табличного анализа позволяют сделать следующие выводы.

В исследовании был проведен обзор научных работ на предмет различных подходов к балансировке сетей SDN. Как результат, произведено сравнение предлагаемых решений по трем критериям: производительность, качество обслуживания, эффективность алгоритма. Исходя из сравнительного анализа алгоритмов балансировки, а также учитывая достоинства и недостатки описанных в них алгоритмов, сделан вывод, что балансировка трафика в сети SDN должна быть основана на лучшем алгоритме - EWRR, позволяющий учитывать параметры QoS.

1.5 Выводы по главе 1

Концепция «единой ответственности» ITU-T E.860 позволяет упростить управление качеством, возлагая ответственность на основного поставщика, который координирует взаимодействие с субпоставщиками. Это обеспечивает выполнение QoS-требований для конечного пользователя через систему соглашений о SLA.

Рекомендации ITU-T Y.1541 определяют классы сетевого QoS с конкретными значениями потери пакетов, круговой сетевой задержки, вариации сетевой задержки и коэффициента ошибок. Эти классы обеспечивают поддержку широкого спектра приложений (голос, видео, интерактивные сервисы) и служат основой для соглашений между пользователями и поставщиками услуг.

Рекомендация 3GPP TS 23.501 вводит идентификатор 5QI, который связывает классы трафика с гарантированными параметрами QoS (потери пакетов,

круговая сетевая задержка). Это позволяет эффективно обслуживать разнородные приложения — от критичных к задержкам (например, управление автономными системами) до фоновых задач.

Анализ SLA крупнейших российских провайдеров («Ростелеком», «Билайн», Транстелеком, «Связьтранснефть») показал, что параметры QoS (потери пакетов, круговая сетевая задержка, вариация сетевой задержки) дифференцируются по классам сервиса: Real-Time, Business-Critical, Best-Effort. Это подтверждает соответствие российских практик международным рекомендациям.

Обзор современных работ выявил тенденцию к использованию интеллектуальных и адаптивных алгоритмов балансировки, учитывающих многопараметрические метрики QoS. Однако существующие решения не обеспечивают комплексного учёта разнородных параметров QoS, их нормализации и адаптации к различным классам сервиса в соответствии с SLA.

К ключевым проблемам существующих подходов к балансировке относятся:

- Отсутствие интеграции в алгоритмы балансировки многопараметрического подхода к QoS;
- Отсутствие нормализации разнородных метрик QoS и системы весовых коэффициентов на основе параметров качества;
- Игнорирование специфики SLA для различных классов трафика (Real Time, Business Critical, Best Effort).

Сравнительный анализ алгоритмов балансировки показал, что перспективное направление – EWRR демонстрирует наилучшие результаты по критериям производительности, параметров QoS и эффективности алгоритма. Он поддерживает динамическую балансировку с учётом весовых коэффициентов, что делает его пригодным для реализации многопараметрического подхода к управлению QoS в сетях SDN.

Рекомендации для дальнейшего исследования:

1. Разработка методики комплексной оценки параметров QoS с применением нормализации и весовых коэффициентов для различных классов трафика.

2. Реализация алгоритма балансировки на основе EWRR, адаптированного к требованиям трафика с разными SLA.

Это позволит создать адаптивную систему управления трафиком, обеспечивающую выполнение требований QoS для разнородных приложений в современных сетях связи.

ГЛАВА 2 РАЗРАБОТКА МЕТОДА КОМПЛЕКСНОЙ ОЦЕНКИ ПАРАМЕТРОВ QoS В СЕТЯХ SDN

2.1 Протоколы взаимодействия

Software-Defined Networking (SDN) использует набор протоколов для разделения управления (control plane) и передачи данных (data plane), а также для взаимодействия между компонентами. Протоколы SDN делятся на три ключевые категории: южные (southbound), северные (northbound) и восточно-западные (east-west) интерфейсы. Южные интерфейсы (Southbound Protocols) обеспечивают связь между SDN-контроллером и сетевыми устройствами (коммутаторами, маршрутизаторами), как показано на рисунке 1.1).

На уровне инфраструктуры (Data Plane) протоколы определяют, как контроллер взаимодействует с сетевыми устройствами.

Протоколы сети SDN между контроллерами и устройствами:

- OpenFlow — эталонный стандарт самый известный протокол SDN. Важной особенностью протокола OpenFlow является возможность программной реализации на физических (реализованных аппаратно) устройствах, например, на коммутаторах, маршрутизаторах и т.д. для их использования в качестве устройств программно-конфигурируемой сети [71, 72]. OpenFlow определяет, как контроллер программно управляет таблицами потоков на коммутаторах [73].

Протокол OpenFlow управляет таблицами потоков (flow tables) на устройствах. Поддерживает версии (1.0, 1.3, 1.5). Пример использования: контроллер отправляет Flow-Mod сообщение коммутатору, чтобы добавить правило для перенаправления трафика [74, 75].

Протокол OpenFlow разработан учёными из Стэнфорда для управления инфраструктурой сети SDN-контроллера. Работа OpenFlow заключается в модификации содержания таблицы передачи пакетов внутри маршрутизатора или

коммутатора (forwarding table). Архитектура сетевых устройств представляется в виде трех плоскостей: администрирования (management plane), управления (control plane) и передачи данных (data plane), как показано на рисунке 2.1.



Рисунок 2.1 — Работа OpenFlow

Плоскость администрирования (Management Plane) устанавливает режимы работы устройства, модифицирует версии программного обеспечения, «зашитого» в сетевое устройство, выполняет команды протокола администрирования сети SNMP (Simple Network Management Protocol) от внешней системы администрирования сети (NMS, от англ. Network Management System), а также команды внешней конфигурации устройства через интерфейс командной строки CLI (Command Line Interface), которые вручную вводит системный администратор. Плоскость передачи данных (Data Plane) передает пакеты данных со входа устройства (IN) на его выход (OUT) в соответствии с таблицей передачи (Forwarding Table). На плоскости управления (Control Plane) работают протоколы маршрутизации и коммутации.

Плоскость управления использует таблицу маршрутизации (routing table) для составления таблицы передачи пакетов (forwarding table), которую использует плоскость передачи данных (data plane). Таблицы передачи пакетов доставляется на плоскость передачи от плоскости администрирования. В соответствии с

таблицей передачи, пакет, прибывающий на входной порт коммутатора (IN), передается плоскостью передачи данных на выходной порт (OUT).

OpenFlow – это протокол управления потоками (flow) в сети.

Сетевые устройства (маршрутизаторы, коммутаторы и другие) занимаются передачей пакетов и фреймов в соответствии с протоколами маршрутизации, однако, бизнес-приложения не используют определённые пакеты для предоставления услуг. Они обмениваются данными между сервером и клиентом и создают потоки пакетов от источника к получателю, который называется «Flow» (поток) [71].

- OF-CONFIG – протокол, который позволяет удаленно настраивать коммутаторы. С помощью протокола OF-CONFIG реализуется конфигурация логического коммутатора с тем, чтобы он мог управляться контроллером через протокол OpenFlow [76].

- P4 Runtime (P4RT) — это открытый протокол управления плоскостью данных для программируемых сетевых устройств, работающих под управлением программ, написанных на языке P4. Это ключевой компонент современной архитектуры SDN, обеспечивающий полную независимость от конкретного оборудования. P4 Runtime предоставляет унифицированный API для контроллера SDN, позволяющий динамически управлять таблицами потоков (forwarding pipeline) в коммутаторах, маршрутизаторах или программируемых сетевых картах, не зависящий от конкретной архитектуры чипа или программы P4. Контроллер "видит" именно таблицы, соответствия и действия, которые описаны в программе P4, загруженной в устройство [77].

- SNMP (Simple Network Management Protocol) — стандартный протокол для управления и мониторинга устройств в IP-сетях. Работает на прикладном уровне модели TCP/IP, используется для обмена служебной информацией между сетевыми устройствами и системами администрирования.

- gNMI (gRPC Network Management Interface) — это высокопроизводительный протокол управления и телеметрии для сетевых устройств, основанный на gRPC и YANG моделях данных. Является ключевым

протоколом в SDN-архитектуре, особенно в подходах Model-Driven Programmability.

Основные характеристики gNMI:

- Основан на gRPC. Использует фреймворк удалённого вызова процедур от Google, что обеспечивает высокую производительность, бинарную сериализацию и поддержку потоковой передачи.

- Использует YANG модели. Конфигурационные и операционные данные структурированы согласно YANG-моделям (могут быть стандартными от IETF/OpenConfig или вендорскими).

- Единый интерфейс для всего. Объединяет функции конфигурации, оперативного управления, телеметрии в реальном времени и RPC-вызовов.

- NETCONF — это стандартизированный протокол для конфигурационного управления в SDN. Хотя для телеметрии в реальном времени и высокопроизводительных сценариев чаще используют gNMI, NETCONF используется для: начальной настройки устройств; конфигурации сложных сервисов (MPLS, VPN); гибридных сред, где SDN-контроллер управляет традиционным оборудованием; сценариев, где требуется строгая транзакционность и валидация.

В SDN-архитектуре NETCONF часто используется в сочетании с другими протоколами: OpenFlow для данных, gNMI для телеметрии, а NETCONF — для базовой конфигурации и управления жизненным циклом устройств.

2.2 Сценарий взаимодействия коммутаторов и контроллеров

Канал OpenFlow — это интерфейс, соединяющий логический коммутатор OpenFlow с контроллером OpenFlow. Через интерфейс OpenFlow контроллер настраивает и управляет коммутатором, получает события от коммутатора и отправляет пакеты из коммутатора. Канал управления коммутатора поддерживает

один канал OpenFlow с одним контроллером или несколько каналов OpenFlow и позволяет нескольким контроллерам совместно управлять коммутатором. Между каналом данных и каналом OpenFlow интерфейс зависит от реализации, но сообщения канала OpenFlow отформатированы в соответствии с протоколом коммутатора OpenFlow. Канал OpenFlow обычно шифруется с использованием TLS, а может работать напрямую по TCP.

Протокол коммутатора OpenFlow поддерживает три типа сообщений: контроллер-коммутатор, асинхронный и симметричный, каждый из которых имеет несколько подтипов. Сообщения контроллер-коммутатор инициируются контроллером и используются для непосредственного управления или проверки состояния коммутатора. Асинхронные сообщения инициируются коммутатором и используются для обновления контроллера о сетевых событиях и изменениях состояния коммутатора. Симметричные сообщения инициируются либо коммутатором, либо контроллером и отправляются без запроса.

Типы сообщений, используемые OpenFlow:

1) Симметричные

Симметричные сообщения отправляются без запроса в любом направлении.

Когда соединение OpenFlow устанавливается впервые, каждая сторона соединения отправляет сообщение OFPT_HELLO с полем версии, установленным на самую высокую версию протокола коммутатора OpenFlow, поддерживаемую отправителем. Сообщение Hello включает элементы OpenFlow, для настройки соединения. После получения сообщения Hello получатель вычисляет версию протокола коммутатора OpenFlow, которая будет использоваться. Если и отправленное сообщение Hello, и полученное сообщение Hello содержат приветствия OFPHET_VERSIONBITMAP, то согласованная версия должна быть самой высокой версией, установленной в обоих устройствах. В противном случае согласованная версия должна быть меньшей из отправленных и полученных в полях версий. Если согласованная версия поддерживается получателем, то соединение продолжается. В противном случае получатель должен ответить сообщением об ошибке Hello Failed, а затем разорвать соединение.

После того, как коммутатор и контроллер обменялись сообщениями OFPT_HELLO и успешно согласовали номер версии, настройка соединения завершена, и стандартные сообщения OpenFlow обмениваются по соединению. Одно из первых действий, которое должен выполнить контроллер, — это отправить сообщение OFPT_FEATURES_REQUEST, чтобы получить идентификатор Datapath ID коммутатора.

Контроллер устанавливает параметры конфигурации в коммутаторе и запрашивает их с помощью сообщений OPT_SET_CONFIG и OFPT_GET_CONFIG_REQUEST соответственно.

2) Асинхронные

Асинхронные сообщения отправляются от коммутатора без запроса контроллера. Коммутаторы отправляют асинхронные сообщения контроллерам, чтобы обозначить получение пакета или изменение состояния коммутатора.

Сообщение OFPT_PACKET_IN используется, когда пакеты отправляются на контроллер, сообщение содержит причину отправки пакета - Reason packet (одна из OFPR_*):

OFPR_TABLE_MISS = 0, /* Нет соответствующего потока flow (запись потока flow пропущена в таблице)*/

OFPR_APPLY_ACTION = 1, /*Вывод на контроллер в apply-actions*/

OFPR_INVALID_TTL = 2, /*Пакет имеет недопустимый TTL*/

OFPR_ACTION_SET = 3, /* Вывод на контроллер набора действий action set*/

OFPR_GROUP = 4, /*Вывод на контроллер в групповом контейнере group bucket*/

OFPR_PACKET_OUT = 5, /*Вывод на контроллер в packet-out*/

Если контроллер запросил уведомление об истечении времени записи потока или об их удалении из таблиц, используется сообщение OFPT_FLOW_REMOVED.

При добавлении, изменении и удалении портов из тракта данных контроллер информируется сообщением OFPT_PORT_STATUS.

3) Контроллер-коммутатор

Сообщения контроллера/коммутатора инициируются контроллером и могут требовать или не требовать ответа от коммутатора.

Контроллер использует сообщение OFPT_PACKET_OUT для отправки пакета через канал передачи данных.

С помощью запроса OFPT_FLOW_MOD записи потока изменяются в таблице потоков flow table. Контроллер настраивает динамическое состояние в таблице потоков с помощью запроса OFPT_TABLE_MOD.

С помощью сообщения OFPT_GROUP_MOD выполняются изменения в групповой таблице из контроллера.

Сообщение OFPT_PORT_MOD контроллер использует для изменения режима порта.

С помощью сообщения OFPT_TABLE_MOD устанавливается структура таблиц на коммутаторе [72].

Сценарий взаимодействия между коммутатором и контроллером представлен на рисунке 2.2.

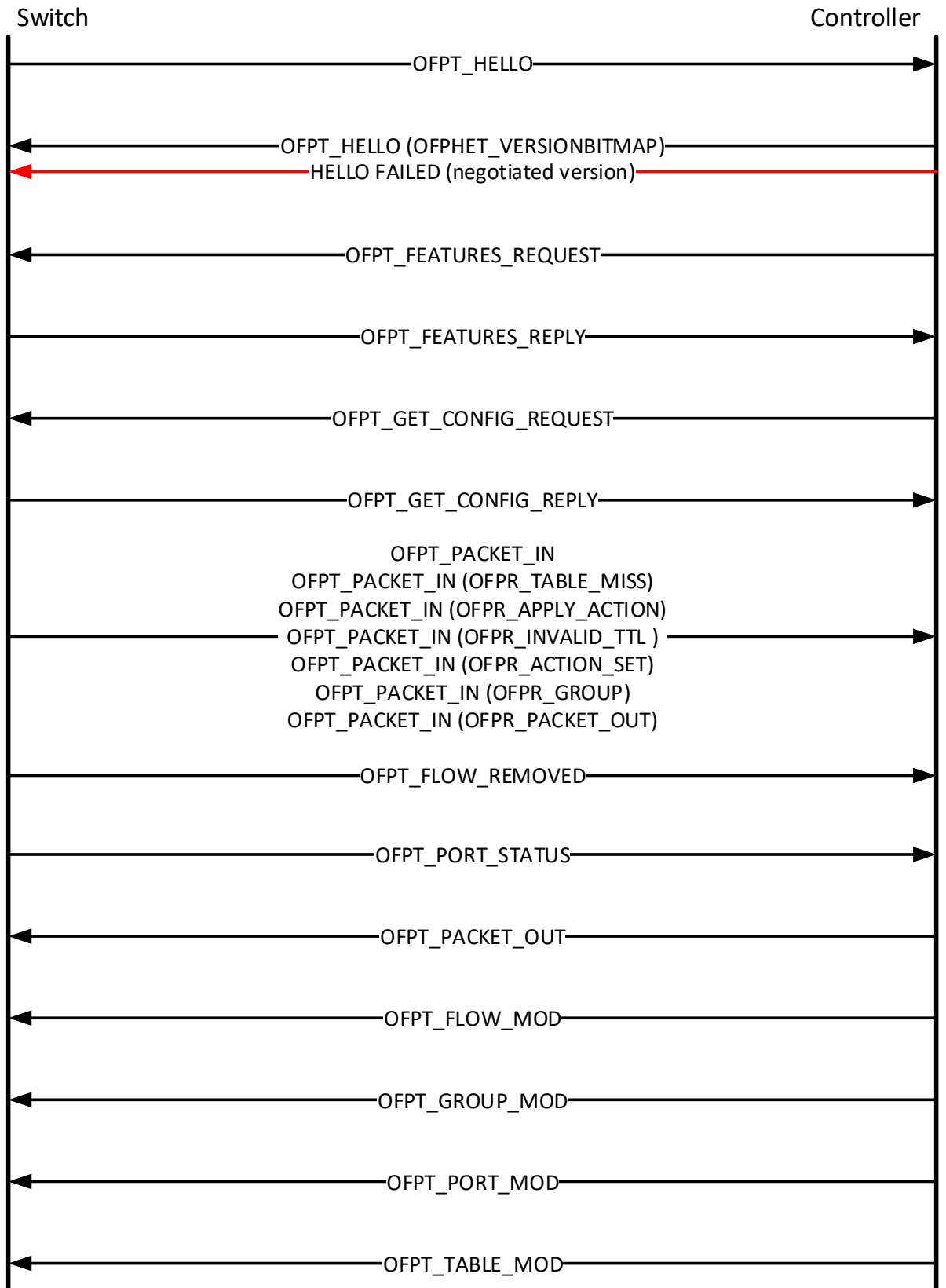


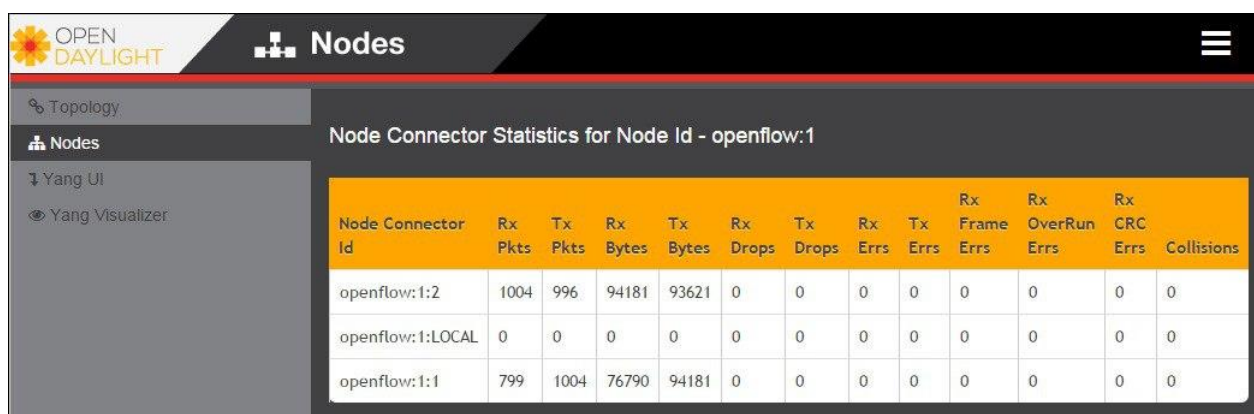
Рисунок 2.2 — Сценарий взаимодействия между коммутатором и контроллером

2.3. Исследуемые сетевые характеристики в параметрах контроллера

Контроллер SDN отвечает за балансировку нагрузки в сети и улучшение параметров QoS. Параметры QoS зависят от типа сетей, типа трафика. В исследовании [48] проанализировано множество параметров QoS, применяемых для балансировки нагрузки контроллеров в сети SDN. При балансировке наиболее часто используемыми параметрами QoS являются:

- Задержка обработки запроса — время обработки запроса, поступающего от коммутатора с учетом поступающей нагрузки d ;
- Потерянные пакеты — пакеты, откинутае ввиду невозможности передачи из-за перегрузки сети t .

На примере контроллера OpenDayLight параметр потерянные пакеты представлены на рисунке 2.3.



Node Connector Id	Rx Pkts	Tx Pkts	Rx Bytes	Tx Bytes	Rx Drops	Tx Drops	Rx Errs	Tx Errs	Rx Frame Errs	Rx OverRun Errs	Rx CRC Errs	Collisions
openflow:1:2	1004	996	94181	93621	0	0	0	0	0	0	0	0
openflow:1:LOCAL	0	0	0	0	0	0	0	0	0	0	0	0
openflow:1:1	799	1004	76790	94181	0	0	0	0	0	0	0	0

Рисунок 2.3 — Параметры контроллера OpenDayLight

Параметр Tx Errs характеризует пропускную способность сети от\до данного контроллера.

2.4. Исследуемые сетевые характеристики в параметрах коммутатора

Рассмотрим основные сетевые характеристики в параметрах коммутатора:

1. Загрузка центрального процессора (CPU utilization)

Загрузка центрального процессора коммутатора показана на рисунке 2.4.

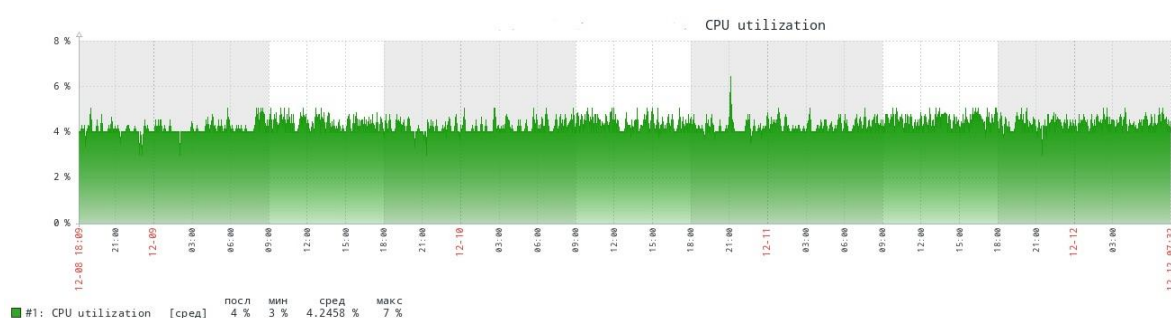


Рисунок 2.4 — Загрузка центрального процессора коммутатора

Это процент использования ресурсов процессора от его максимальной возможности вычислительных мощностей.

Загрузка центрального процессора — это показатель объема работы, выполняемой центральным процессором в течение определенного периода времени, обычно выражаемый в процентах. Он дает ценную информацию о том, насколько эффективно центральный процессор выполняет свои задачи и есть ли возможности для улучшения. Загрузка ЦП может варьироваться в зависимости от характера и интенсивности вычислительных задач, при этом некоторые процессы требуют больше процессорного времени, чем другие.

Таким образом занятость процессора мы отображаем в единицах от 1 % до 100 % занятости.

2. Используемое пространство на диске (Disk used space)

Пример используемого пространства на диске коммутатора представлено на рисунке 2.5.

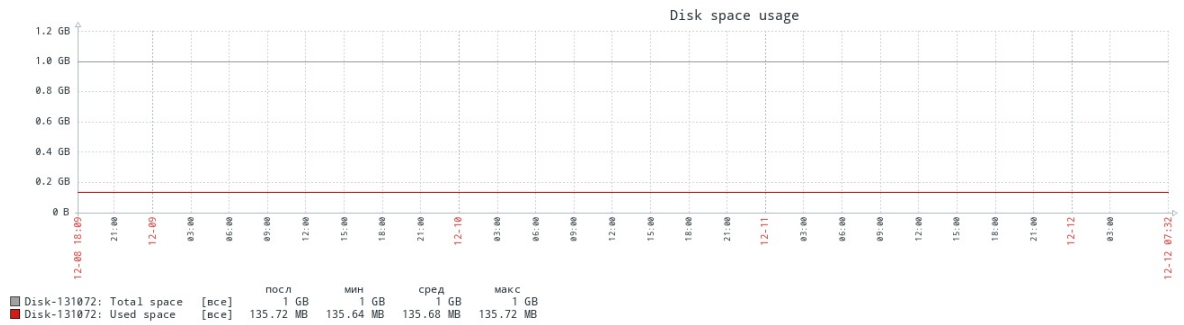


Рисунок 2.5 — Используемое пространство на диске коммутатора

Параметр характеризует количество данных, сохраняемых на диске. Виды дисков: винчестеры, твердотельные накопители SSD. Накопители SSD имеют преимущество перед винчестером, поскольку не подвержены механическому старению и удары на него не оказывают губительное воздействие.

Использование дискового пространства измеряется в Мб, для расчетов примем 1Gb (100 %), таким образом занятость дискового пространства измеряется в процентах от 1 до 100 %.

3. Сетевой трафик (Network traffic)

Коммутатор имеет параметр пропускная способность сетевого трафика, зависящий от количества и емкости портов и матрицы коммутации.

Приведем пример занятости порта коммутатора. Загрузка порта отображается на рисунке 2.6 в Мб.

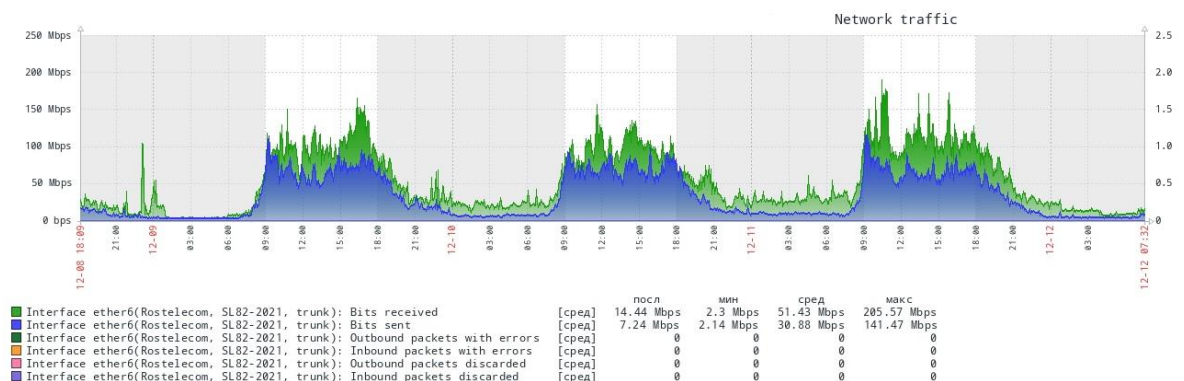


Рисунок 2.6 — Пример загрузки порта коммутатора

Загруженность порта коммутатора измеряется в Мб, для расчетов примем 1Gb (максимальная пропускная способность порта 100 %), таким образом занятость порта измеряется в процентах от 1 до 100 %.

4. Использование памяти (Memory usage)

Пример занятости памяти CPU отображается рисунке 2.7 в процентах от 1 % до 100 %.

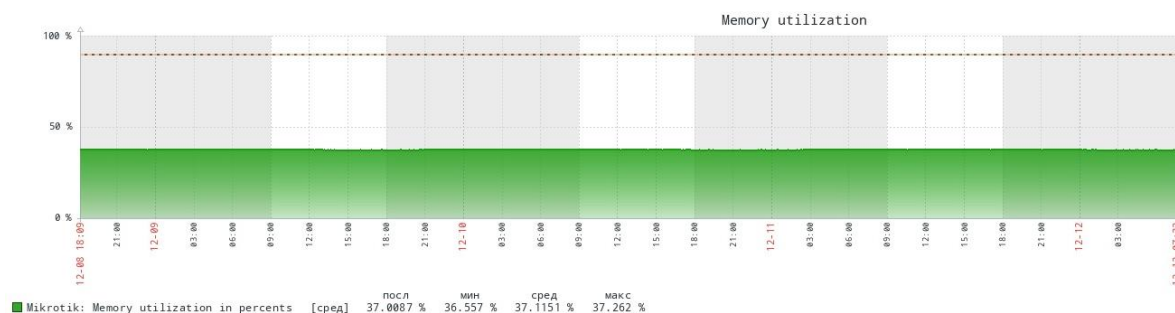


Рисунок 2.7 — Занятость памяти CPU

5. Потери пакетов (Loss of packet)

Коэффициент потери IP-пакетов (Internet Protocol packet Loss Ratio, IPLR) – это отношение общего количества результатов потерянных IP-пакетов к общему количеству переданных IP – пакетов в представляющей интерес совокупности. [38].

Loss of packet - это измерение между узлами А и В в двунаправленном канале, которое позволяет измерить две величины, связанные с каналом: A_TxLoss (transmit loss, потери при передаче): количество пакетов, переданных точкой А по каналу, но не принятых точкой В. и A_RxLoss (receive loss): количество пакетов, переданных В по каналу, но не принятых в А [78].

6. Задержка при передаче IP-пакетов (Ping)

Задержка передачи IP-пакетов (IPTD, Internet Protocol packet Transfer Delay) определена для всех результатов "успешная передача пакетов" и "ошибочный пакет" в базовой секции или совокупность сетевых секций NSE (Network Section Ensemble). Параметр IPTD представляет собой период времени ($t_2 - t_1$) между

появлением двух соответствующих эталонных событий передачи IP-пакетов – входным эталонным событием переноса (передачи) IP-пакетов $IPRE_1$ (Internet Protocol packet transfer Reference Event) в момент t_1 и выходным эталонным событием переноса (передачи) IP-пакетов $IPRE_2$ в момент t_2 , где $(t_2 > t_1)$ и $(t_2 - t_1) \leq T_{max}$. Если пакет фрагментируется в пределах NSE, t_2 – это время последнего соответствующего выходного события. Задержка сквозной передачи IP-пакетов – это однонаправленная задержка между пунктами измерения MP (Measurement Point) в хост-источник SRC (Source host) и хост-получатель DST (Destination host), как показано на рисунке 2.8, где Exchange Link (EL) – звено обмена, Network Section (NS) сетевая секция [38].

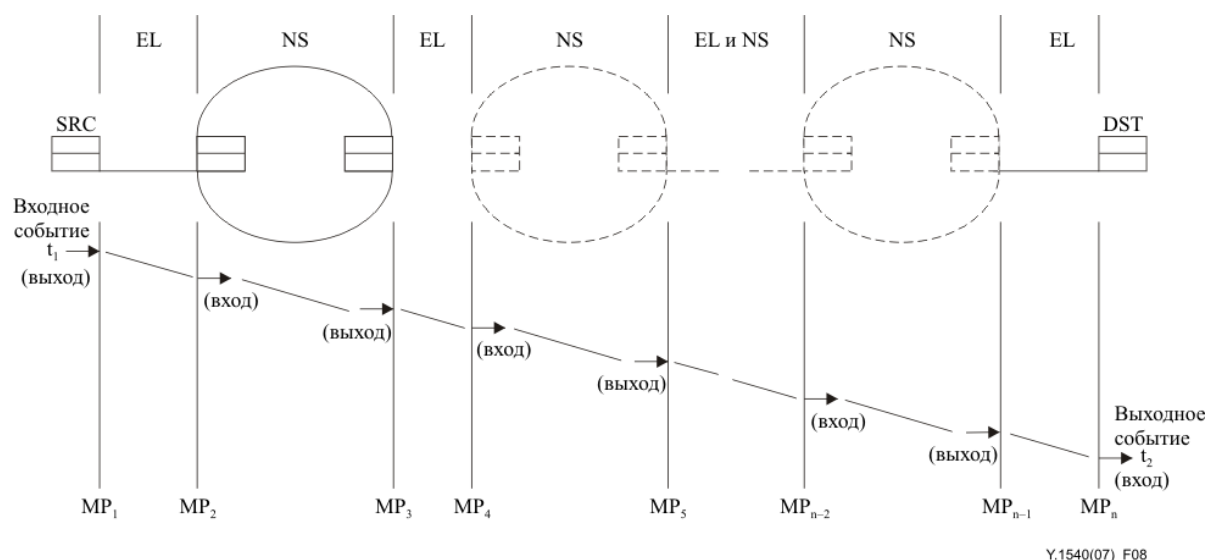


Рисунок 2.8 – События задержки при передаче IP-пакетов (показана сквозная передача одного IP-пакета)

Звено обмена соединяет:

- 1) хост-источник или хост-получатель со смежным хостом (например, маршрутизатором), возможно находящимся в другом подчинении, который иногда называют звеном доступа, входным звеном или выходным звеном; либо
- 2) маршрутизатор одной сетевой секции с маршрутизатором другой сетевой секции.

Следует отметить, что ответственность за звено обмена, его пропускную способность и его рабочие характеристики распределяется между соединяемыми сторонами.

Сетевая секция – это набор хостов, которые в совокупности образуют часть IP-службы между SRC и DST и находятся под одной (или совместной) полномочной ответственностью. Некоторые сетевые секции содержат один хост без соединительных звеньев. Частными случаями сетевых секций являются NS-источник и NS-получатель. Две сетевые секции соединяются с помощью звеньев обмена.

7. Джиттер (Jitter)

Вариации задержек сквозной двухточечной передачи IP-пакетов – Packet Delay Variation (PDV).

Вариации задержек передачи IP-пакетов также являются важным фактором. Поточковые приложения могут использовать информацию обо всем диапазоне вариаций задержек IP-пакетов для исключения недогрузки и перегрузки буферов. Экстремальные вариации задержек IP-пакетов могут привести к увеличению порогов таймера повторных передач TCP, а также задержать повторные передачи IP-пакетов или вызвать повторную передачу пакетов.

Вариации задержек IP-пакетов (PDV) при сквозной двухточечной передаче определяются на основе наблюдений поступления, соответствующих IP-пакетов на входные и выходные пункты измерения MP (Measurement Point) (например, MP_{DST}, MP_{SRC}). Эти наблюдения характеризуют варьированность в комбинации событий поступления IP-пакетов на выходных MP относительно комбинации соответствующих событий на входном MP по отношению к эталонной задержке.

PDV (v_k) при двухточечной передаче IP-пакета k между SRC и DST представляет собой разницу между абсолютной задержкой передачи IP-пакетов (x_k) пакета k и определенной эталонной задержкой передачи IP-пакетов $d_{1,2}$ между теми же MP $v_k = x_k - d_{1,2}$ изображено на рисунке 2.9.

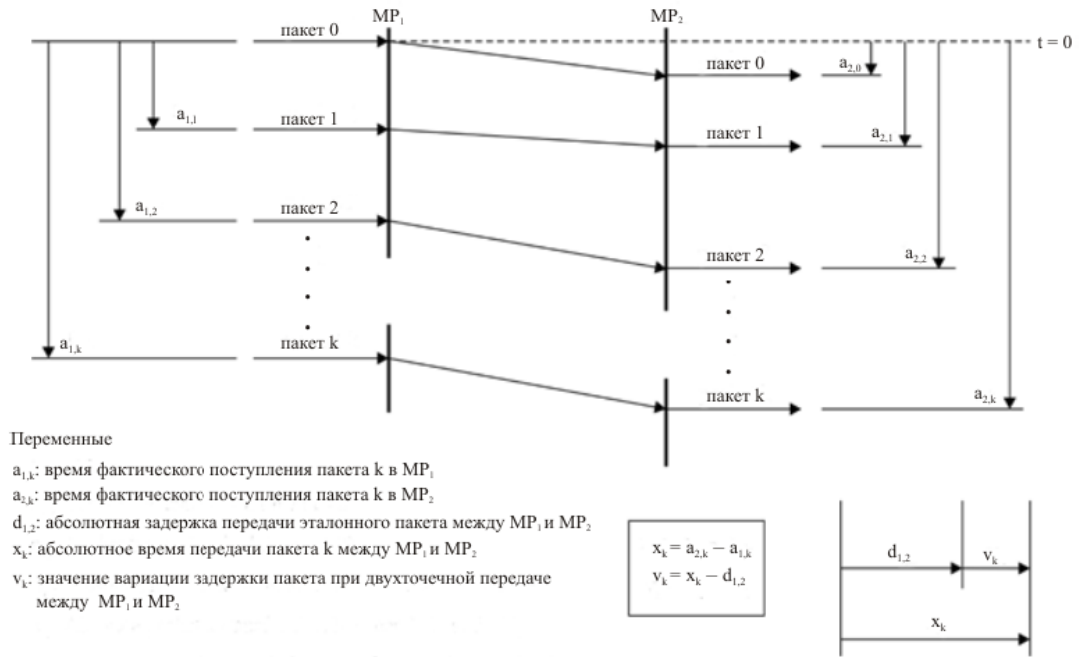


Рисунок 2.9 – Вариация задержки IP-пакетов при двухточечной передаче

Эталонная задержка передачи IP-пакетов, $d_{1,2}$, между SRC и DST – это абсолютная задержка передачи IP-пакетов, которой подвергается выбранный IP-пакет между этими двумя МР. Положительные значения вариации задержки IP-пакетов (IPDV, аббр. от англ. IP Packet Delay Variation) при двухточечной передаче соответствуют задержкам передачи IP-пакетов, превышающим те, которым подвергается эталонный IP-пакет; отрицательные значения при двухточечной передаче соответствуют задержкам передачи пакетов меньше тех, которым подвергается эталонный IP-пакет. Распределение PDV при двухточечной передаче идентично распределению абсолютных задержек передачи IP-пакетов, выражаемых постоянным значением, равным $d_{1,2}$.

Изменение задержки между пакетами (IPDV) и изменение задержки пакетов (PDV) [79] — это показатели производительности, полученные на основе одностороннего измерения задержки, и они важны в некоторых приложениях. IPDV представляет собой разницу между задержками в одну сторону для последовательных пакетов в потоке. PDV, с учетом тестового интервала измерения, представляет собой разницу между задержкой в одну сторону для пакета в этом интервале и задержкой в пакете в интервале с минимальной задержкой [38].

Рассмотренные параметры применяются в реальных сетях и, поэтому, релевантно могут быть применены для построения в нагрузочной модели для балансировки трафика.

2.5 Модель расчета параметра комплексной оценки QoS

Полагаем, что имеется несколько SDN коммутаторов, образующих сеть, обслуживающую трафик, включенных в них пользователей как показано на рисунке 2.10.

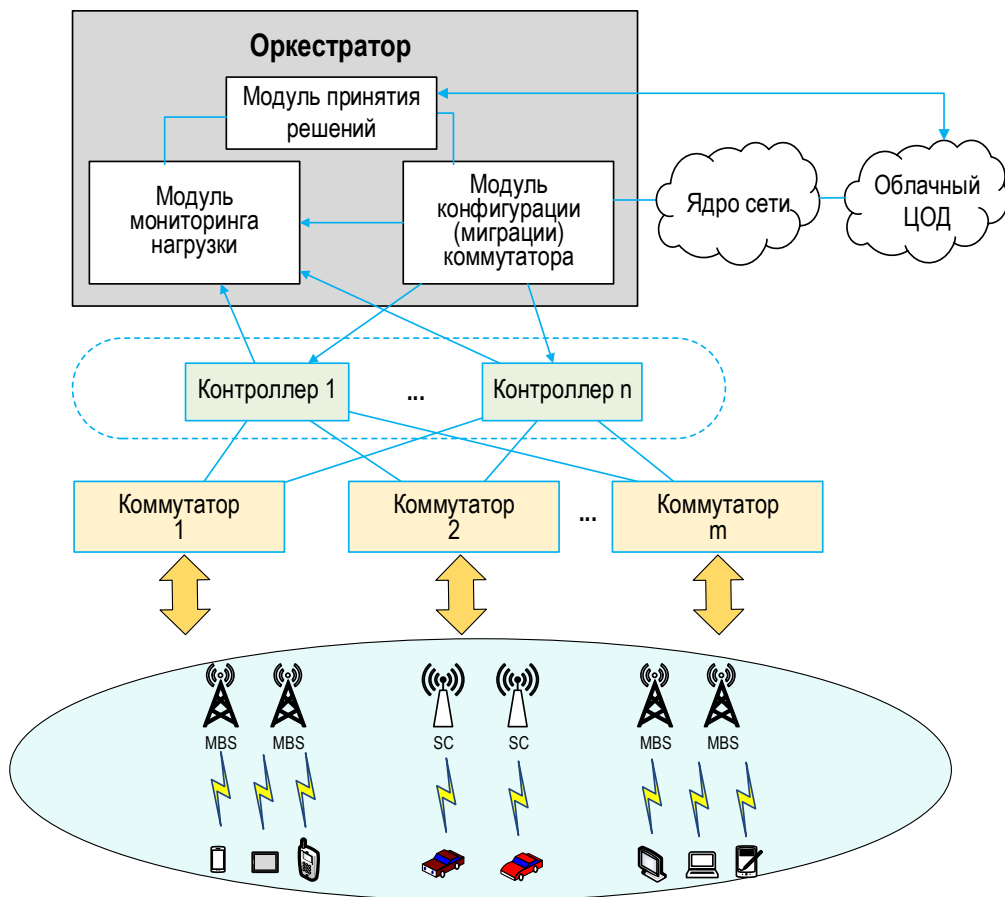


Рисунок 2.10 — Архитектура сети SDN

Для балансировки нагрузки рассмотрим модель распределения нагрузки на уровне контроллеров и коммутаторов.

Модель балансировки нагрузки должна учитывать параметры QoS контроллеров и коммутаторов по параметрам, которые значимы в данной сети. Для повышения качества клиентских сервисов необходимо при балансировке нагрузки учитывать не только нагрузку на контроллер, но и параметры QoS.

Контроллер SDN отвечает за балансировку нагрузки в сети и улучшение параметров QoS. Параметры QoS зависят от типа сетей, типа трафика. В исследовании [48] проанализировано множество параметров QoS, применяемых для балансировки нагрузки в сети SDN. Для балансировки наиболее часто используемыми параметрами QoS являются:

- задержка – время обработки запроса, поступающего от коммутатора с учетом поступающей нагрузки d , мс;
- потерянные пакеты — пакеты откинутае ввиду невозможности передачи из-за перегрузки сети t , %.

Чтобы решить, на какой контроллер перенаправить соединение, оркестратору необходимо сравнить значения соответствующих параметров для каждого контроллера. Для этого необходимо сформировать из собранных параметров матрицу A (2.2) с a_{ij} , где i - порядковый номер параметра ($i=1, \dots, l$), а j - номер контроллера ($j = 1, \dots, n$).

1. Нагрузка контроллера – Y .
2. Задержка обработки запроса – время обработки запроса, поступающего от коммутатора с учетом поступающей нагрузки a_d ;
3. Потерянные пакеты – количество потерянных пакетов a_t ;

При балансировке сетевого трафика с использованием параметров QoS учитываются параметры с разными единицами измерения и возникает вопрос о нормализации параметров QoS, так как они различны по своим единицам измерения [80].

Для повышения качества клиентских сервисов модель балансировки нагрузки должна учитывать не только нагрузку на элементы сети, но и параметры QoS и использовать EWRR, учитывающий параметры QoS для контроллеров и коммутаторов.

Применим алгоритм EWRR для балансировки нагрузки контроллеров с учетом параметров QoS.

Разное влияние на сервис оказывает каждый из параметров. Поэтому необходимо оценить влияние каждого параметра, определяем нормализующий коэффициент C_{ij} для каждого параметра.

$$\sum_{j=1}^n \sum_{i=1}^l C_{ij} = 1 \quad (2.1)$$

$$A = \begin{pmatrix} C_{11}a_{w_1} & C_{21}a_{w_2} & \dots & C_{n1}a_{w_n} \\ C_{12}a_{d_1} & C_{22}a_{d_2} & \dots & C_{n2}a_{d_n} \\ C_{13}a_{t_1} & C_{23}a_{t_2} & \dots & C_{n3}a_{t_n} \end{pmatrix}, \quad (2.2)$$

где a_{w_n} — нормализованное значение нагрузки на контроллер;

a_{d_n} — нормализованное значение задержки обработки запроса;

a_{t_n} — нормализованное значение потери пакетов.

Элемент a_{ij} в матрице A (2.2) умножается на нормализующий коэффициент C_{ij} (2.1).

Для каждого параметра необходимо найти один контроллер с одинаковым, минимальным значением и присвоить ему значение в соответствии с формулой 2.3:

$$b_{ij} = \begin{cases} 1, & a_{ij} = \min_{j=1, \dots, n} (a_{ij}) \\ 0, & a_{ij} > \min_{j=1, \dots, n} (a_{ij}) \end{cases} \quad (2.3)$$

Применяя это соотношение, получаем матрицу B (2.4) с элементами b_{ij} , представляющими значения, которые присваиваются контроллерам для каждого параметра:

$$B = \left\| b_{ij} \right\|_{(3 \times n)} \quad (2.4)$$

Строка матрицы В содержит элементы b_{ij} , значение которых отлично от нуля (несколько контроллеров) имеют одинаковое, минимальное значение a_{ij} для наблюдаемого параметра.

Задача состоит в том, чтобы найти контроллер с наименьшей суммой нормализованных параметров QoS умноженных на нормализующие коэффициенты получив комбинированный параметр элемента сети в определенный момент времени.

Рассмотрим метод вычисления комбинированного параметра QoS контроллера. Произведем поэтапно расчет нормализованной суммы параметров QoS для контроллера.

Сумма нормализованных значений будет выглядеть так:

$$Y_{C_QoS} = Y_d + Y_t, \quad (2.5)$$

где Y_d - нормализованный «вес» значения задержки d ,

Y_t - нормализованный «вес» значения потерянных пакетов t .

Для нормализации необходимо привести значения в разных единицах измерения к одному единому виду, который позволит сравнивать их между собой и использовать для расчёта.

1) Введем ограничения U для параметров задержка d и потерянные пакеты t :

d от 1 до U_d , мс,

t от 0 до U_t , %.

2) Произведем нормализацию, т.е. приведем к единому виду параметры, имеющие разные единицы измерения:

d разделим на U_d ;

t разделим на U_t .

3) Получим комбинированный параметр QoS контроллера (2.6), выполнив действия 1-3 для (2.5) получаем:

$$Y_{C_QoS} = \frac{d}{U_d} \times C_1 + \frac{t}{U_t} \times C_2, \quad (2.6)$$

где C_1, C_2 нормализующий коэффициент соответствующего параметра.

Получили комбинированный параметр QoS контроллера Y_{C_QoS} для двух наиболее часто используемых параметров при балансировке согласно [48]. Данный алгоритм может быть использован для оценки других параметров QoS. При применении на конкретной сети весовые коэффициенты конкретных параметров могут корректироваться для учета особенностей данной сети.

Руководствуясь алгоритмом распределения нагрузки EWRR определяем, что контроллер с наименьшим значением комбинированного параметра QoS - это контроллер на который распределяется нагрузка. Поэтому необходимо вычислить значение комбинированного параметра QoS для каждого контроллера по формуле:

$$S_j = \sum_{i=1}^3 b_{ij}, j=1, \dots, n \quad (2.7)$$

Применяя формулу (2.7), получаем вектор: $S_j = [S_1, S_2, \dots, S_n]$. Необходимо провести минимизацию, найдя элемент S_j (индекс контроллера).

Таким образом, получаем информацию о минимальном комбинированном параметре QoS контроллера. Направляя запрос на контроллер с минимальным комбинированным параметром QoS получаем максимальные параметры качества QoS для данного запроса.

Алгоритм EWRR балансировки нагрузки на примере коммутаторов с учетом параметров QoS будет выглядеть так: чтобы решить, на какой коммутатор перенаправить соединение, оркестратору необходимо сравнить значения нагрузки соответствующих параметров для каждого коммутатора. Для этого необходимо сформировать из параметров матрицу E (2.6) с e_{hf} , где h - порядковый номер параметра ($h=1, \dots, r$); f - номер коммутатора ($f=1, \dots, m$).

Разное влияние на сервис оказывает каждый из параметров. Поэтому необходимо оценить влияние каждого параметра, определяем нормализующий коэффициент D_{hf} для каждого параметра.

$$\sum_{f=1}^m \sum_{h=1}^r D_{hf} = 1$$

Элемент e_{hf} в матрице E умножается на нормализующий коэффициент D_{hf} (2.8).

$$E = \begin{pmatrix} D_{11}e_{cpu_1} & D_{21}e_{cpu_2} & \dots & D_{m1}e_{cpu_m} \\ D_{12}e_{dus_1} & D_{22}e_{dus_2} & \dots & D_{m2}e_{dus_m} \\ D_{13}e_{nt_1} & D_{23}e_{nt_2} & \dots & D_{m3}e_{nt_m} \\ D_{14}e_{mu_1} & D_{24}e_{mu_2} & \dots & D_{m4}e_{mu_m} \\ D_{15}e_{lp_1} & D_{25}e_{lp_2} & \dots & D_{m5}e_{lp_m} \\ D_{16}e_{pn_1} & D_{26}e_{pn_2} & \dots & D_{m6}e_{pn_m} \\ D_{17}e_{jt_1} & D_{27}e_{jt_2} & \dots & D_{m7}e_{jt_m} \end{pmatrix}, \quad (2.8)$$

где e_{cpu_m} — нормализованное значение загрузка центрального процессора;

e_{dus_m} — нормализованное значение используемое пространство на диске;

e_{nt_m} — нормализованное значение сетевой трафик;

e_{mu_m} — нормализованное значение использование памяти;

e_{lp_m} — нормализованное значение потери пакетов;

e_{pn_m} — нормализованное значение круговая сетевая задержка;

e_{jt_m} — нормализованное значение вариация сетевой задержки.

Для каждого параметра необходимо найти один коммутатор или несколько с одинаковым, минимальным значением и присвоить ему значение в соответствии с формулой:

$$z_{hf} = \begin{cases} 1, & e_{hf} = \min_{f=1, \dots, m} (e_{hf}) \\ 0, & e_{hf} > \min_{f=1, \dots, m} (e_{hf}) \end{cases} \quad (2.9)$$

Применяя формулу (2.9), получаем матрицу Z (2.10) с элементами z_{hf} , представляющими значения, которые присваиваются коммутаторам для каждого параметра:

$$Z = \parallel z_{hf} \parallel_{(7*m)} \quad (2.10)$$

Строка матрицы Z содержит один или несколько элементов z_{hf} , значение которых отлично от нуля (несколько коммутаторов) имеют одинаковое, минимальное значение z_{hf} для наблюдаемого параметра.

Задача состоит в том, чтобы найти коммутатор с наименьшей суммой нормализованных параметров QoS умноженных на нормализующие коэффициенты получив комбинированный параметр элемента сети в определенный момент времени.

Рассмотрим метод вычисления комбинированного параметра QoS коммутатора. Произведем поэтапно расчет нормализованной суммы параметров QoS для коммутатора. Сумма нормализованных значений будет выглядеть следующим образом:

$$y_{S_QoS} = y_{lp} + y_{pn} + y_{jt}, \quad (2.11)$$

где y_{lp} – нормализованный «вес» значения потерянные пакеты lp ,

y_{pn} – нормализованный «вес» значения круговая сетевая задержка pn ,

y_{jt} – нормализованный «вес» значения вариация сетевой задержки jt .

Для нормализации необходимо привести значения в разных единицах измерения к одному единому виду, который позволит сравнивать их между собой или использовать для расчёта, для это введем пределы параметров U_{lp} , U_{pn} и U_{jt} и выполним следующие действия:

1) Введем ограничения для параметров потери пакетов lp , круговая сетевая задержка pn , вариация круговой сетевой задержки jt .

lp от 0 % до U_{lp} %;

pn от 1 мс до U_{pn} мс;

jt от 0 мс до U_{jt} мс.

2) Произведём нормализацию, т.е. приведем к единому виду параметры, имеющие разные единицы измерения: lp разделим на U_{lp} ; pn разделим на U_{pn} ; jt разделим на U_{jt} .

3) Получим комбинированный параметр QoS коммутатора, выполнив действия 1-3 для (2.11):

$$w_{S_QoS} = \frac{lp}{U_{lp}} \times C_1 + \frac{pn}{U_{pn}} \times C_2 + \frac{jt}{U_{jt}} \times C_3, \quad (2.12)$$

где C_1, C_2, C_3 нормализующий коэффициент соответствующего параметра.

Получили комбинированный параметр качества сервиса коммутатора w_{S_QoS} для трех параметров QoS, которые оцениваются в клиентских сервисах. При применении на конкретной сети весовые коэффициенты параметров могут корректироваться для учета особенностей сети.

Руководствуясь алгоритмом распределения нагрузки определяем, что коммутатор с наименьшим значением комбинированного параметра QoS - это коммутатор на который распределяется нагрузка. Поэтому необходимо вычислить значение комбинированного параметра QoS для каждого коммутатора по соотношению 2.13:

$$G_f = \sum_{h=1}^7 z_{hf}, f=1, \dots, m \quad (2.13)$$

Применяя соотношение (2.10), получаем вектор: $G_f = [G_1, G_2, \dots, G_m]$. Необходимо провести минимизацию, найдя элемент G_f (индекс коммутатора).

Ограничения метода вычисления комбинированного параметра QoS: нагрузка на коммутатор, контроллер не превосходит 80 %, при выходе за данные

пределы значения параметров QoS не будут отражать реальное состояние сети. Превышение заданных ограничений параметров качества, т.е. превышение значения ограничения 80 % будет означать, что сеть близка к перегрузке, что не позволит корректно балансировать трафик [81].

Таким образом, получаем информацию о минимальном комбинированном параметре QoS коммутатора. Направляя запрос на коммутатор с минимальным комбинированным параметром QoS получаем максимальные параметры качества QoS для данного запроса.

2.6 Выводы по главе 2

В данной главе проведено исследование протоколов и механизмов взаимодействия в программно-конфигурируемых сетях (SDN), а также разработана модель балансировки нагрузки с учётом параметров качества обслуживания (QoS) для мультиконтроллерной архитектуры.

Рассмотрены ключевые протоколы южного (OpenFlow, NETCONF/YANG, OVSDB, OpFlex), северного (REST API, RESTCONF, gRPC) и восточно-западного интерфейсов, обеспечивающих взаимодействие между контроллерами.

Проанализирован алгоритм обмена сообщениями между коммутаторами и контроллерами по протоколу OpenFlow, включая симметричные, асинхронные и управляющие сообщения. Определена роль контроллера в динамическом управлении сетевыми потоками и обеспечении гибкой маршрутизации.

Выделены ключевые параметры QoS, влияющие на балансировку нагрузки: задержка, потери пакетов, загрузка центрального процессора, использование памяти, дискового пространства и сетевого трафика. Показана необходимость учёта как аппаратных характеристик коммутаторов, так и параметров сети для обеспечения качества обслуживания.

Предложена модель балансировки нагрузки, учитывающая параметры QoS как на уровне контроллеров, так и на уровне коммутаторов. Введён метод комплексной оценки параметров QoS в сетях SDN, основанный на сведении разнородных параметров качества контроллеров и коммутаторов к единому интегральному показателю с использованием нормализующих весовых коэффициентов. Модель включает матричное представление параметров, расчёт нормализующих коэффициентов и определение минимальных комбинированных параметров QoS для принятия решений о распределении нагрузки.

Разработанная модель позволяет улучшить качество обслуживания за счёт динамического перераспределения нагрузки с учётом актуальных сетевых условий. Модель может быть адаптирована для различных сценариев использования, включая разнородный трафик гетерогенных сетей на базе SDN и сети операторского уровня.

Таким образом, проведённое исследование создаёт теоретическую основу для реализации эффективных механизмов балансировки нагрузки в мультиконтроллерных сетях SDN с учётом требований QoS, что способствует повышению надёжности, производительности и адаптивности современных сетевых инфраструктур

ГЛАВА 3 МОДЕЛЬ БАЛАНСИРОВКИ НАГРУЗКИ В МУЛЬТИКОНТРОЛЛЕРНОЙ СЕТИ SDN С УЧЕТОМ ПАРАМЕТРОВ QoS

3.1 Структурная схема балансировки нагрузки контроллеров

Для выполнения задач по балансировке нагрузки актуализируется информация о нагрузке контроллеров и коммутаторов. Информация о нагрузке, используется для расчета и мониторинга состояния контроллеров и коммутаторов. Определяет находится ли контроллер или коммутатор в состоянии перегрузки требующей миграции в соответствии с заданным порогом. Если контроллер или коммутатор находятся в состоянии перегрузки и требуется миграция, оркестратор выбирает контроллер или коммутатор, подлежащий миграции. Для информационного взаимодействия между контроллерами и коммутаторами, информацию о нагрузке собирает и хранит оркестратор (рисунок 3.1).

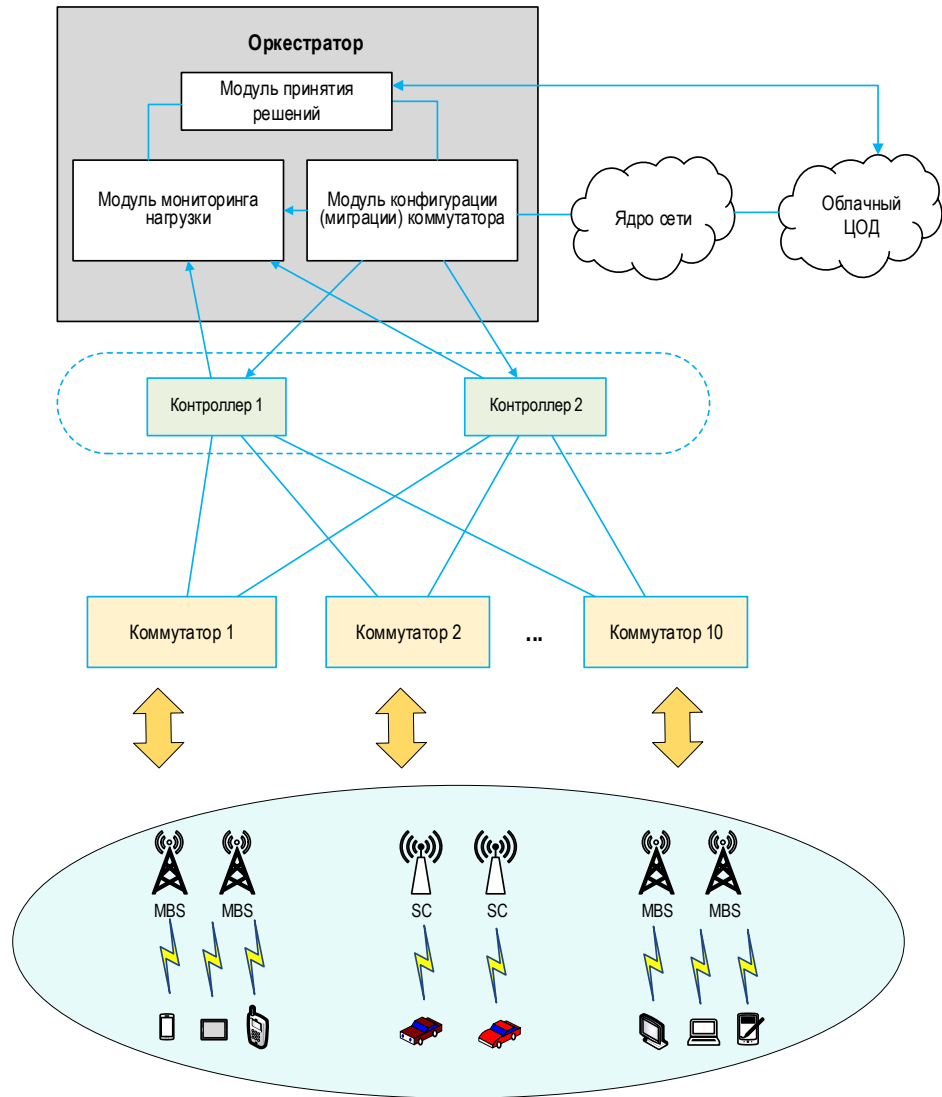


Рисунок 3.1 — Модельная сеть SDN

3.2 Разработка математической модели балансировки трафика

При анализе договоров российских операторов связи выделяется три класса клиентского трафика: класс 1 - Real Time, класс 2 – Best Effort.

Real Time трафик для передачи особенно критичных к задержкам и потерям данных. Примеры приложений: голосовой трафик (VoIP), видео ТВ-сигналов высокого качества, online игры.

Best Effort трафик для передачи некритичных данных. Пример: интернет-трафик.

Класс 1. Real Time все параметры трафика одинаково важны (довольно строгий), таким образом для модели в данном типе трафика корректно применить для каждого параметра коэффициенты нормализации $D_{lp}=0,33$; $D_{pn}=0,34$; $D_{jt}=0,33$.

Класс 2 - Best Effort параметры более «лояльны», чем в классе два и параметр вариация сетевой задержки не определяется, таким образом для третьего класса трафика можно применить коэффициенты нормализации $D_{lp}=0,5$; $D_{pn}=0,5$; $D_{jt}=0$ [33].

Пользователи генерируют потоки данных, каждый поток данных пользователей создает нагрузку на контроллер Y_c и коммутатор Y_s . Примем:

- максимальная нагрузка на контроллер равна 50 % (всего 2 контроллера), итого $2 \times 50 \% = 100 \%$;
- максимальная нагрузку коммутаторов равна 10 % (всего 10 коммутаторов), итого $10 \times 10 \% = 100 \%$;
- условно максимальная нагрузка, генерируемая абонентами равна 80 % от максимально возможной. Итого максимальная нагрузка равна 80 % согласно «СП 519.1325800.2023. Свод правил. Сети связи. Правила проектирования» (утв. Приказом Минстроя России от 17.03.2023 N 200/пр) [81] для предотвращения перегрузки рекомендуется учесть при расчетах (оценке) максимально допустимую загрузку линии связи на направлении на уровне не более 80 % максимальной [82].

Запросы с весом Y генерируются с нагрузкой задаваемой случайным образом от 0,01 % до 0,09 %. Макс $Y \leq 80 \%$.

Для учета параметров QoS контроллеров задаем случайным образом параметры QoS для контроллеров. Согласно исследованию [83] заданы ограничения для параметров d , t . Итого в каждом запросе будет 2 варианта параметров для контроллеров:

1. Нагрузка контроллера Y_1 , Y_2 , в диапазоне от 0,01 % до 0,09 %.

2. Параметры QoS d_1, d_2 – параметр задержка обработки запросов, мс. В диапазоне от 1 до 10 мс.

3. Параметры QoS t_1, t_2 – параметр потерянных пакетов, %. В диапазоне от 0 до 0,1 %.

Генерация случайных величин для математической модели контроллеров представлена в таблице 3.1.

Таблица 3.1 — Обозначения величин параметров для контроллеров

Параметр	Порядковый номер контроллера	
	1	2
Нагрузка W_c	Y_{c_1}	Y_{c_2}
Задержка d	d_1	d_2
Потерянные пакеты t	t_1	t_2

Для математической модели принимаем, что вес запроса для всех контроллеров будет равен между собой, так как каждый контроллер затрачивает одинаковое количество ресурсов на обработку потока данных ($Y_{c_1} = Y_{c_2}$).

В расчетах математической модели учитываем общий параметр физических характеристик коммутатора – нагрузку коммутатора Y_s , состоящую из загрузки центрального процессора, используемого пространства на диске, сетевого трафика, использования памяти. Нагрузка коммутатора в инженерных системах рассчитывается исходя из максимальной производительности.

Каждый поток данных клиента создает нагрузку на параметры 1-4. Очевидно, что при превышении предельных значений этих параметров: загрузка центрального процессора (CPU utilization); используемое пространство на диске (Disk used space); сетевой трафик (Network traffic); использование памяти (Memory usage) коммутатор аппаратно не может участвовать в балансировке нагрузки по причине перегрузки. Таким образом вводим единый параметр нагрузки, генерируемой клиентом на коммутатор (Y_s). Для балансировки трафика по параметрам QoS вводим единый комбинированный параметр качества обслуживания QoS,

базирующийся на сетевых характеристиках. Введем единый комбинированный параметр качества обслуживания, базирующийся на сетевых характеристиках.

1. Нагрузка, генерируемая клиентом на коммутатор Y_s , %;
2. Потери пакетов lp , %;
3. Круговая сетевая задержка pn , мс.
4. Вариация круговой задержки jt , мс.

В модели f - номер коммутатора ($f = 1, \dots, 10$).

Приведение параметров QoS для использования в модели выполнено на базе значений и рекомендаций из документов ITU G114, ITU-T Y.1540, ITU-T Y.1541 [84, 38, 33] и реальных примеров договоров операторов связи (таблица 3.2).

Таблица 3.2 — Параметры QoS математической модели

Параметр	Best Effort	Real Time
Процент потери пакетов, %	1	0,1
Круговая сетевая задержка, мс	200	80
Вариация сетевой задержки, мс	-	20

Для учета параметров QoS коммутаторов задаем случайным образом параметры QoS. Итого в каждом запросе будет 10 вариантов параметров для коммутаторов.

1. Загрузка коммутатора Y_s , в диапазоне от 0,01 % до 0,09 %;

Параметры QoS из технического задания договора оператора связи Билайн:

2. Потери пакетов lp в диапазоне от 0 % до 0,1 %;
3. Круговая сетевая задержка pn в диапазоне от 1 мс до 80 мс;
4. Вариация сетевой задержки jt в диапазоне от 0 мс до 20 мс.

Таким образом, генерации случайных величин для математической модели коммутаторов представлена в таблице 3.3.

Таблица 3.3 — Обозначения величин параметров для коммутаторов

Параметр	Порядковый номер коммутатора									
	1	2	3	4	5	6	7	8	9	10
Нагрузка Y_s	Y_{s1}	Y_{s2}	Y_{s3}	Y_{s4}	Y_{s5}	Y_{s6}	Y_{s7}	Y_{s8}	Y_{s9}	Y_{s10}
Потери пакетов lp	lp_1	lp_2	lp_3	lp_4	lp_5	lp_6	lp_7	lp_8	lp_9	lp_{10}
Круговая сетевая задержка pn	pn_1	pn_2	pn_3	pn_4	pn_5	pn_6	pn_7	pn_8	pn_9	pn_{10}
Вариация круговой сетевой задержки jt	jt_1	jt_2	jt_3	jt_4	jt_5	jt_6	jt_7	jt_8	jt_9	jt_{10}

Для математической модели принимаем, что нагрузка Y_s для всех коммутаторов одинаковая, так как каждый коммутатор потратит одинаковое количество ресурсов на обработку потока данных.

3.3 Математическая модель балансировки по параметру нагрузки

На первом этапе равномерно распределяем нагрузку на контроллеры не учитывая параметры QoS, используя алгоритм EWRR. То есть оркестратор направляет запрос на наиболее свободный контроллер. Нормализованные параметры QoS суммируем для каждого запроса потока данных контроллера.

Далее приступаем ко второму этапу распределяем нагрузку на коммутаторы. Приступаем к равномерному распределению нагрузки на коммутаторах не учитывая параметры QoS, применяя алгоритм EWRR. То есть оркестратор направляет запрос на наиболее свободный коммутатор. Нормализованные параметры QoS суммируем для каждого коммутатора модели.

Распределение параметров нагрузки на контроллеры, используя алгоритм EWRR. Результаты представим в таблице 3.4, где указаны сумма параметров Y_s ; сумма нормализованных параметров QoS d и t контроллеров для 100 000 запросов.

Таблица 3.4 — Расчет сумм нормализованных параметров для контроллеров

Параметр	Порядковый номер контроллера	
	1	2
Нагрузка Y_C	$\sum_j Y_{C_1}$	$\sum_j Y_{C_2}$
Нормализованная задержка d	$\sum_j d_1$	$\sum_j d_2$
Нормализованные потерянные пакеты t	$\sum_j t_1$	$\sum_j t_2$
Комбинированный параметр QoS	$\sum_j QoS_1$	$\sum_j QoS_2$

Распределение параметров нагрузки на коммутаторы, используя алгоритм EWRR. Результаты представим в таблице 3.5, где указаны сумма параметров Y_s ; сумма нормализованных параметров QoS круговая сетевая задержка, потери пакетов и вариация сетевой задержки контроллеров для 100 000 запросов.

Таблица 3.5 — Расчет сумм нормализованных параметров для коммутаторов

Параметр	Порядковый номер коммутатора									
	1	2	3	4	5	6	7	8	9	10
Нагрузка Y_s	$\sum_j Y_{s1}$	$\sum_j Y_{s2}$	$\sum_j Y_{s3}$	$\sum_j Y_{s4}$	$\sum_j Y_{s5}$	$\sum_j Y_{s6}$	$\sum_j Y_{s7}$	$\sum_j Y_{s8}$	$\sum_j Y_{s9}$	$\sum_j Y_{s10}$
Нормализованная потеря пакетов lp	$\sum_j lp_1$	$\sum_j lp_2$	$\sum_j lp_3$	$\sum_j lp_4$	$\sum_j lp_5$	$\sum_j lp_6$	$\sum_j lp_7$	$\sum_j lp_8$	$\sum_j lp_9$	$\sum_j lp_{10}$
Нормализованная круговая сетевая задержка pn	$\sum_j pn_1$	$\sum_j pn_2$	$\sum_j pn_3$	$\sum_j pn_4$	$\sum_j pn_5$	$\sum_j pn_6$	$\sum_j pn_7$	$\sum_j pn_8$	$\sum_j pn_9$	$\sum_j pn_{10}$
Нормализованное вариация сетевой задержки jt	$\sum_j jt_1$	$\sum_j jt_2$	$\sum_j jt_3$	$\sum_j jt_4$	$\sum_j jt_5$	$\sum_j jt_6$	$\sum_j jt_7$	$\sum_j jt_8$	$\sum_j jt_9$	$\sum_j jt_{10}$
Комбинированный параметр QoS	$\sum_j QoS_1$	$\sum_j QoS_2$	$\sum_j QoS_3$	$\sum_j QoS_4$	$\sum_j QoS_5$	$\sum_j QoS_6$	$\sum_j QoS_7$	$\sum_j QoS_8$	$\sum_j QoS_9$	$\sum_j QoS_{10}$

Задаем случайные параметры в таблице 3.1 и 3.3.

Для математического расчета сформируем 100 000 запросов на обслуживание со случайными параметрами значений, которые указаны в таблице 3.1 и 3.3.

Пример запроса со случайно сгенерированными параметрами нагрузки и QoS для коммутаторов и контроллеров представлены в таблице 3.6.

Таблица 3.6 — Пример запроса со случайно сгенерированными параметрами нагрузки и QoS для коммутаторов и контроллеров

Контроллеры			Коммутаторы										
	1	2		1	2	3	4	5	6	7	8	9	10
Wc	5	5	Ws	8	8	8	8	8	8	8	8	8	8
QoS_d	5	8	QoS_lp	3	9	5	2	2	1	1	2	8	5
QoS_t	8	7	QoS_pn	24	56	70	59	55	37	11	45	45	61
Σ QoS	72	83	QoS_jt	5	5	14	16	3	14	17	6	17	12
			Σ QoS	30	59	70	58	35	40	34	139	73	60

По результатам обработки 100 000 случайно сгенерированных запросов и балансировки нагрузки на коммутаторы и контроллеры используя алгоритм EWRR получили значения, которые представлены в таблице 3.7.

Таблица 3.7 — Значения сумм комбинированных параметров QoS для контроллеров и коммутаторов в результате балансировки по параметру нагрузки

Контроллеры											
	1	2									
Wc	248	251									
QoS_d	243	252									
QoS_t	264	281									
Σ	2817	2961									
Коммутаторы											
	1	2	3	4	5	6	7	8	9	10	
Ws	57	51	51	54	54	51	52	53	57	51	
QoS_lp	62	45	45	39	42	53	41	49	56	45	
QoS_pn	414	366	467	244	346	345	311	388	457	485	
QoS_jt	114	108	77	74	145	81	93	98	123	90	
Σ	563	478	468	351	521	451	417	483	576	497	

3.4 Математическая модель балансировки по параметрам QoS

На первом этапе распределяем нагрузку на контроллерах по параметрам QoS, используя алгоритмом EWRR. Оркестратор направляет запрос на контроллер с наименьшим комбинированным параметром QoS, то есть оркестратор выбирает контроллер с минимальным комбинированным параметром QoS. В данном случае будет действовать ограничение $Y_c \leq 50\%$. Нормализованные параметры QoS суммируем для определения комбинированной оценки параметров QoS контроллера.

Чтобы решить, на какой контроллер перенаправить запрос, оркестратору SDN необходимо сравнить значения комбинированного параметра QoS для каждого контроллера. Для этого необходимо сформировать из собранных параметров матрицу A (3.1) с a_{ij} , где i - порядковый номер параметра ($i=1, \dots, 3$).

Разное влияние на сервис оказывает каждый из параметров QoS. Поэтому необходимо оценить влияние каждого параметра, определяем нормализующий коэффициент C_{ij} для каждого параметра равный 0,5, так как задержка в обслуживании и потери пакетов одинаково критичны для качества сервисов.

$$\sum_{j=1}^2 \sum_{i=1}^3 C_{ij} = 1$$

Элемент a_{ij} в матрице A умножается на нормализующий коэффициент C_{ij} .

$$A = \begin{pmatrix} C_{11}a_{w_1} & C_{21}a_{w_2} \\ C_{12}a_{d_1} & C_{22}a_{d_2} \\ C_{13}a_{t_1} & C_{23}a_{t_2} \end{pmatrix} \quad (3.1)$$

Для каждого параметра находим один контроллер с одинаковым, минимальным значением и присваиваем ему значение в соответствии с формулой:

$$b_{ij} = \begin{cases} 1, & a_{ij} = \min_{j=1, \dots, 2} (a_{ij}) \\ 0, & a_{ij} > \min_{j=1, \dots, 2} (a_{ij}) \end{cases} \quad (3.2)$$

Применяя формулу 3.2, получаем матрицу B (3.3) с элементами b_{ij} , представляющими значения, которые присваиваются контроллерам для каждого параметра:

$$B = \|b_{ij}\|_{(3 \times 2)} \quad (3.3)$$

Строка матрицы B содержит один или несколько элементов b_{ij} , значение которых отлично от нуля. Задача состоит в том, чтобы найти контроллер с наименьшим комбинированным параметром QoS в определенный момент времени.

Руководствуясь алгоритмом распределения нагрузки определяем, что контроллер с наименьшим средневзвешенным значением - это контроллер на который распределяется нагрузка. Поэтому необходимо вычислить значение средневзвешенной оценки для каждого контроллера по соотношению:

$$S_j = \sum_{i=1}^3 b_{ij}, j=1, 2 \quad (3.4)$$

Применяя соотношение (3.4), получаем вектор: $S_j = [S_1, S_2]$. Необходимо провести минимизацию, найдя элемент S_j (индекс контроллера).

Таким образом, получаем информацию о минимальной средневзвешенной оценке контроллера. Направляя запрос на контроллер с минимальным комбинированным параметром получаем максимальные параметры качества QoS для данного запроса.

Далее приступаем ко второму этапу распределению нагрузки на коммутаторах на базе параметров QoS, используя алгоритм EWRR. Оркестратор направляет запрос на коммутатор с наименьшим комбинированным параметром QoS, то есть оркестратор выбирает коммутатор с минимальной нормализованной суммой значений потерянных пакетов, круговой сетевой задержки и вариации сетевой задержки. В данном случае будет действовать ограничение максимальное $Y_s \leq 10\%$. Нормализованные параметры QoS суммируем для каждого запроса контроллера для нашей модели.

Чтобы решить, на какой коммутатор перенаправить запрос, оркестратору SDN необходимо сравнить значения комбинированного параметра QoS для каждого коммутатора. Для этого необходимо сформировать из собранных параметров матрицу E (3.5) с e_{hf} , где h - порядковый номер параметра ($h=1, \dots, 4$), f -номер коммутатора.

Параметр нагрузки Y_s не нормализуется в расчётах, так как имеет только два состояния: коммутатор не загружен на 100 %, то коэффициент $D_{1f}=1$, если

коммутатор перегружен, то коэффициент $D_{1f}=0$ (коммутатор не может участвовать в балансировке нагрузки данного запроса).

$$\sum_{f=1}^{10} \sum_{h=1}^4 D_{hf} = 1$$

В математической модели для трафика Real Time зададим нормализующий коэффициент D_{hf} для каждого параметра QoS: $D_{2f} = 0,33$, $D_{3f} = 0,34$, $D_{4f} = 0,33$ и нагрузки D_{1f} $[0,1]$ и в общем случае может быть изменено для корректировки влияния параметров:

$$D_{1f} = \begin{cases} 0, & \text{коммутатор перегружен} \\ 1, & \text{коммутатор не загружен} \end{cases}$$

$$E = \begin{pmatrix} D_{11}e_{w_1} & D_{21}e_{w_2} & \dots & D_{10_1}e_{w_{10}} \\ D_{12}e_{lp_1} & D_{22}e_{lp_2} & \dots & D_{10_2}e_{lp_{10}} \\ D_{13}e_{pn_1} & D_{23}e_{pn_2} & \dots & D_{10_3}e_{pn_{10}} \\ D_{14}e_{jt_1} & D_{24}e_{jt_2} & \dots & D_{10_4}e_{jt_{10}} \end{pmatrix} \quad (3.5)$$

Для каждого параметра необходимо найти один коммутатор или несколько с одинаковым (запрос распределяется на первый по счету), минимальным значением и присвоить ему значение в соответствии с формулой:

$$z_{hf} = \begin{cases} 1, & e_{hf} = \min_{f=1, \dots, 10} (e_{hf}) \\ 0, & e_{hf} > \min_{f=1, \dots, 10} (e_{hf}) \end{cases} \quad (3.6)$$

Применяя формулу 3.6, получаем матрицу Z (3.7) с элементами z_{hf} , представляющими значения, которые присваиваются коммутаторам для каждого параметра:

$$Z = \|z_{hf}\|_{(4*10)} \quad (3.7)$$

Строка матрицы Z содержит один или несколько элементов z_{hf} , значение которых отлично от нуля. Задача состоит в том, чтобы найти коммутатор с наименьшим комбинированным параметром QoS в определенный момент времени. Разное влияние на сервис оказывает каждый из параметров: потери пакетов, круговая сетевая задержка, вариация сетевой задержки.

Руководствуясь алгоритмом распределения нагрузки определяем, что коммутатор с наименьшим значением комбинированного параметра QoS – это коммутатор, на который распределяется нагрузка. Поэтому необходимо вычислить значение комбинированного параметра QoS для каждого коммутатора по соотношению:

$$G_f = \sum_{h=2}^4 z_{hf}, h=2, \dots, 4; f=1, \dots, 10, \quad (3.8)$$

Применяя соотношение (3.8), получаем вектор: $G_f = [G_1, G_2, \dots, G_{10}]$. Необходимо провести минимизацию, найдя элемент G_f (индекс коммутатора). Вычислив коммутатор с минимальным индексом получаем целевой коммутатор с минимальным комбинированным параметром QoS [85].

Для математического расчета сформируем 100 000 запросов на обслуживание со случайными параметрами значений, которые указаны в таблице 3.1 и таблице 3.3.

Пример запроса со случайно сгенерированными параметрами нагрузки и QoS для коммутаторов и контроллеров представлено в таблице 3.8.

Таблица 3.8 — Пример запроса со случайно сгенерированными параметрами нагрузки и QoS для коммутаторов и контроллеров

Контроллеры			Коммутаторы											
	1	2		1	2	3	4	5	6	7	8	9	10	
Wc	2	2	Ws	8	8	8	8	8	8	8	8	8	8	
QoS_d	2	3	QoS_lp	7	2	2	6	8	1	5	5	2	8	
QoS_t	4	3	QoS_pn	64	20	6	73	56	47	59	57	53	3	
Σ QoS	33	33	QoS_jt	13	8	5	4	11	9	5	13	3	19	
			Σ QoS	73	29	18	57	68	39	47	63	35	60	

По результатам обработки 100 000 случайно сгенерированных запросов и балансировки нагрузки на коммутаторы и контроллеры с использованием алгоритма EWRR получили значения, которые представлены в таблице 3.9.

Таблица 3.9 — Значения сумм комбинированных параметров QoS для контроллеров и коммутаторов в результате балансировки на базе параметров QoS

Контроллеры														
	1	2												
Wc	259	242												
QoS_d	216	228												
QoS_t	172	199												
Σ	2156	2372												
Коммутаторы			1	2	3	4	5	6	7	8	9	10		
Ws	46	65	78	72	56	19	69	25	45	51				
QoS_lp	26	36	21	34	32	16	33	25	28	23				
QoS_pn	105	218	341	300	242	87	167	86	160	135				
QoS_jt	53	56	60	89	41	13	64	23	30	59				
Σ	217	301	309	383	273	110	283	156	208	229				

С использованием математической модели произведены расчёты, осуществлена балансировка трафика. Отобразим результаты балансировки

трафика по средневзвешенному комбинированному параметру QoS для контроллеров на рисунке 3.2.

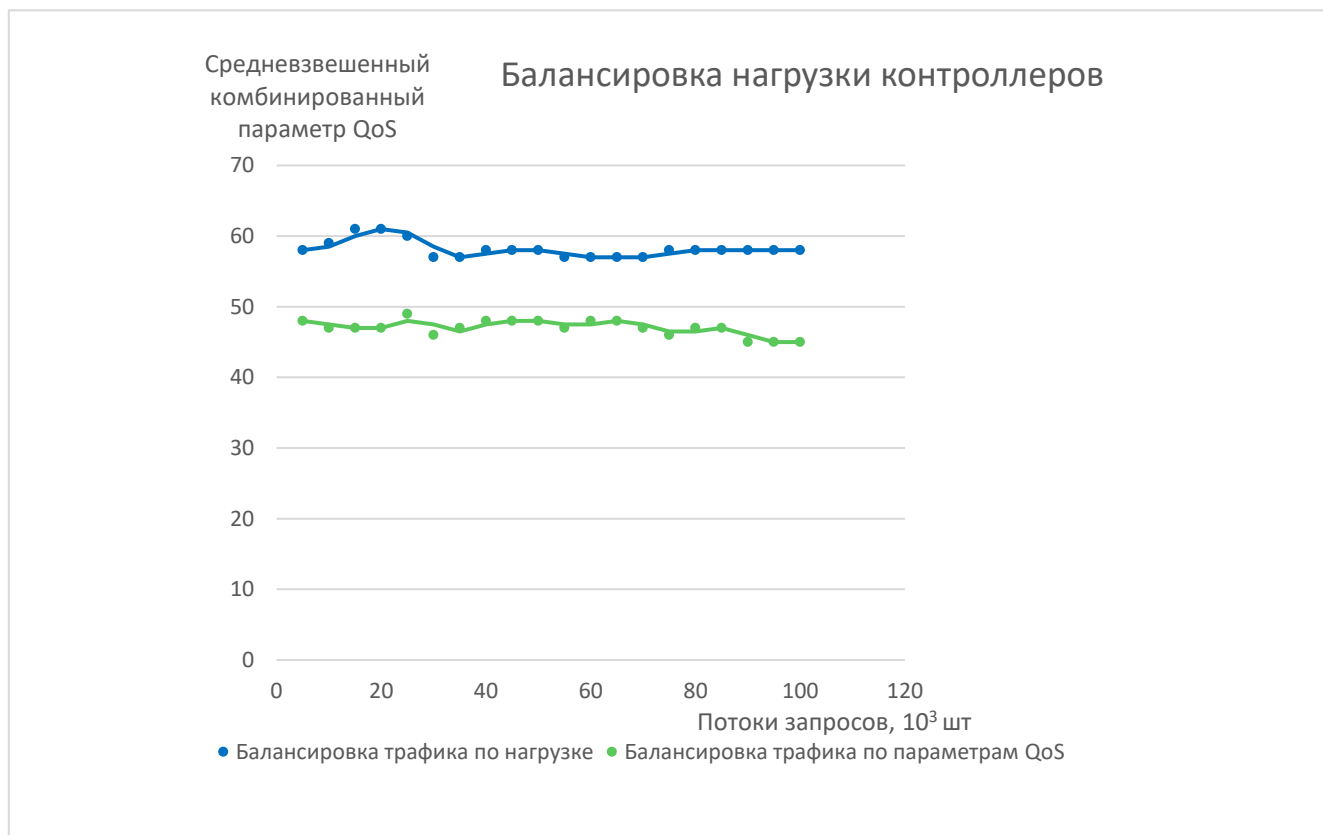


Рисунок 3.2 — Балансировка нагрузки контроллеров

График балансировки трафика контроллеров по параметру нагрузки представлен на рисунке 3.2 изображен синим цветом. Значение средневзвешенного комбинированного параметра QoS при балансировке по нагрузке равно 58.

График балансировки трафика контроллеров по средневзвешенному комбинированному параметру QoS на рисунке 3.2 изображен зеленым цветом и равен 45. Значение средневзвешенного комбинированного параметра QoS уменьшилось при балансировке трафика по параметрам QoS на 22,4 %.

Отообразим результаты балансировки трафика по каждому из параметров QoS коммутаторов (потери пакетов, вариация сетевой задержки и круговая сетевая задержка) на рисунке 3.3.

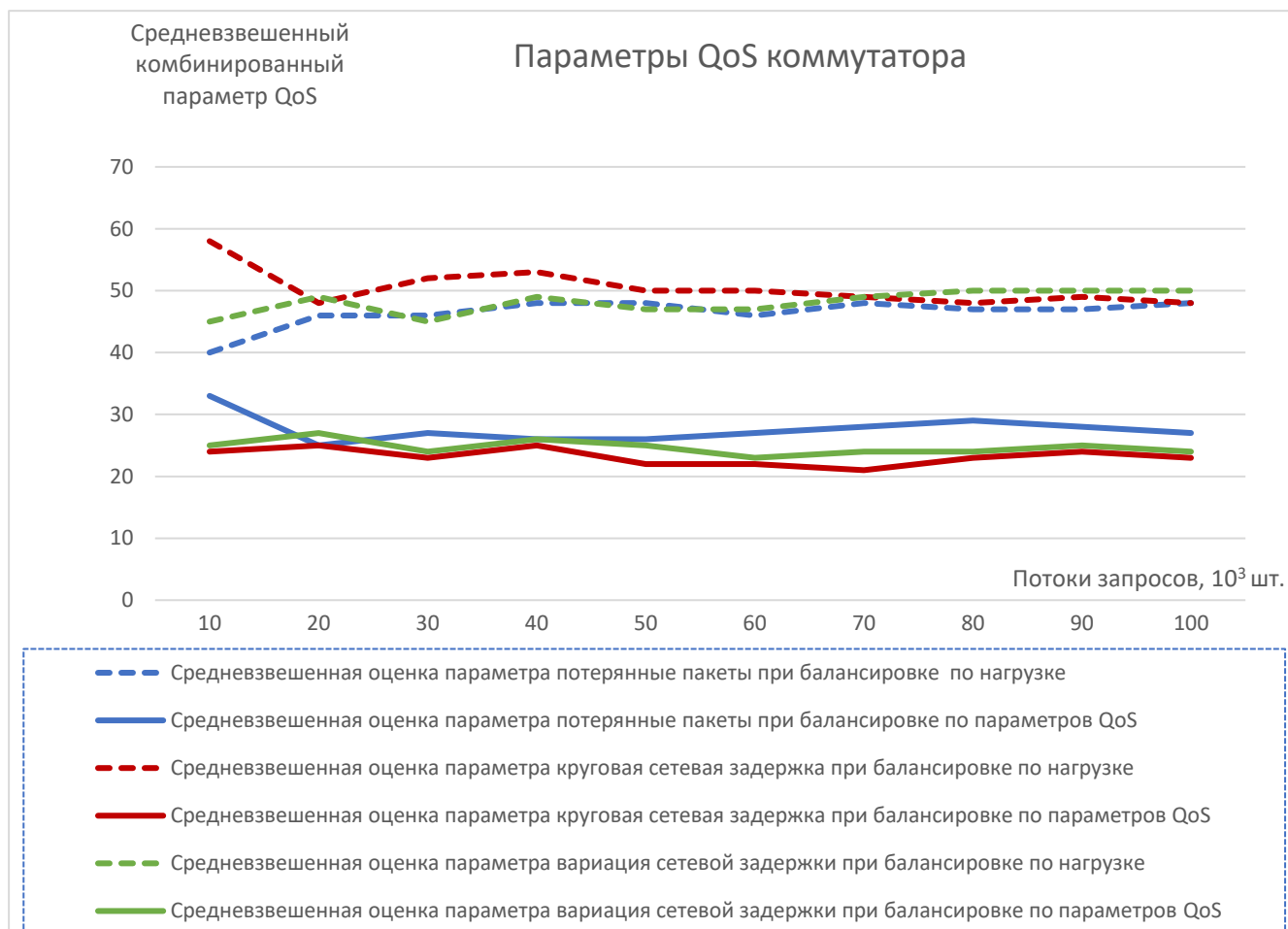


Рисунок 3.3 — Параметры QoS балансировки нагрузки коммутаторов

График балансировки трафика коммутаторов по нагрузке представлен на рисунке 3.3 и изображен прерывистой синей линией для параметра потери пакетов, прерывистой красной линией для параметра круговая сетевая задержка, прерывистой зеленой линией для параметра вариация сетевой задержки. Балансировка трафика по параметрам QoS изображен сплошной синей линией для параметра потери пакетов, сплошной красной линией для параметра круговая сетевая задержка, сплошной зеленой линией для параметра вариация сетевой задержки.

Отообразим результаты балансировки трафика по средневзвешенному комбинированному параметру QoS для коммутаторов на рисунке 3.4.

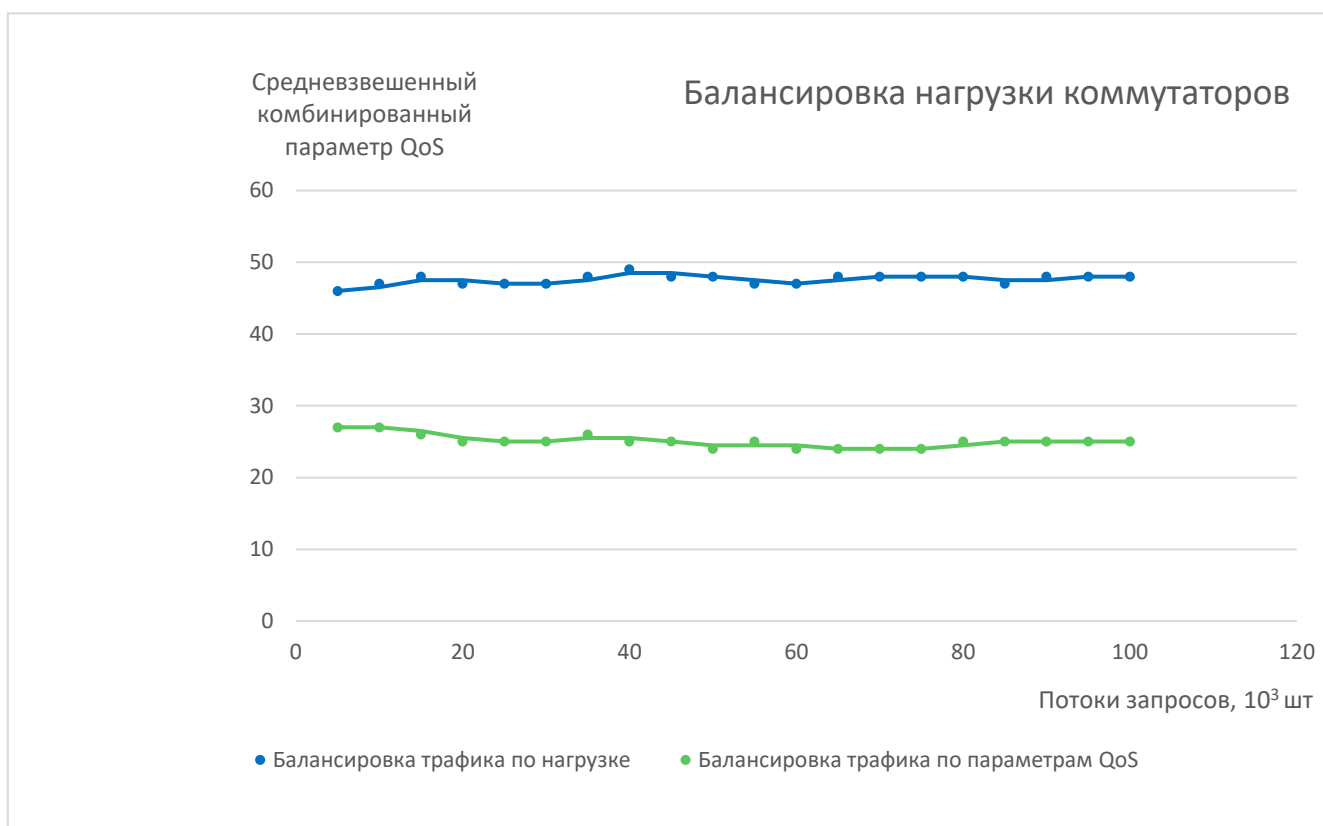


Рисунок 3.4 — Балансировка нагрузки коммутаторов

График балансировки трафика коммутаторов по нагрузке представлен на рисунке 3.4 изображен синим цветом. Значение средневзвешенного комбинированного параметра QoS при балансировке по нагрузке равно 48.

График балансировки трафика коммутаторов по комбинированному параметру QoS на рисунке 3.4 изображен зеленым цветом и равен 25. Значение средневзвешенного комбинированного параметра QoS уменьшилось при балансировке трафика по параметрам QoS на 47,9 %.

3.5 Выводы по главе 3

Проведенное исследование и разработанная модель балансировки нагрузки на уровне контроллеров и уровне коммутаторов демонстрируют высокую эффективность предложенного подхода для мультиконтроллерных сетей SDN. Ключевым результатом работы является количественное подтверждение того, что учет параметров QoS не просто является дополнительной функцией, а становится критически важным фактором, значительно повышающим общую производительность и отказоустойчивость сети.

Основные результаты и выводы по главе 3 можно сформулировать следующим образом:

1) Доказана эффективность комплексного подхода на уровне контроллеров и уровне коммутаторов. Предложенная модель, которая последовательно балансирует нагрузку сначала на уровне контроллеров, а затем на уровне коммутаторов, показала свою жизнеспособность. Такой подход позволяет избежать узких мест как в плоскости управления, так и в плоскости данных, обеспечивая согласованное управление ресурсами во всей сети.

2) Подтверждено преимущество QoS-ориентированной балансировки перед нагрузочной.

Сравнение двух сценариев балансировки только по нагрузке и балансировки с учетом комбинированного параметра QoS выявило существенный выигрыш в качестве:

- На уровне контроллеров средневзвешенный комбинированный параметр QoS улучшен на 22,4 %. Это доказывает, что даже для критически важных элементов управления учет задержки обработки запроса и потерь пакетов позволяет более разумно распределять управляющие запросы, снижая время задержки обработки запросов и потери пакетов.

- На уровне коммутаторов достигнут еще более значительный выигрыш — 47,9 %. Столь существенное улучшение объясняется большим количеством

учитываемых сетевых параметров (потери пакетов, круговая сетевая задержка, вариация сетевой задержки) и их прямым влиянием на качество обслуживания конечного пользователя. Это свидетельствует о том, что для коммутаторов, непосредственно обрабатывающих трафик данных, оптимизация QoS является первостепенной задачей.

3) Алгоритм EWRR с нормализацией параметров QoS показал высокую практическую применимость. Предложенный математический аппарат, основанный на построении матриц параметров, их нормализации с помощью коэффициентов и последующем вычислении комбинированной оценки параметров QoS, оказался эффективным инструментом для принятия обоснованных решений оркестратором. Метод позволяет корректно сравнивать разнородные параметры с разными единицами измерения и гибко настраивать приоритеты для различных классов трафика (Real Time, Best Effort).

4) Практическая значимость для операторов связи. Результаты работы имеют прямое прикладное значение. Внедрение подобной модели позволит операторам связи:

- Выполнять требования стандартов (таких как ITU-T G114, ITU-T Y.1540, ITU-T Y.1541) по качеству обслуживания для разных классов трафика;
- Повысить удовлетворенность абонентов за счет снижения потери пакетов, круговой сетевой задержки, вариации сетевой задержки в критичных приложениях, таких как VoIP, онлайн-видео и финансовые сервисы;
- Более эффективно использовать сетевые ресурсы, предотвращая перегрузки не только по объему трафика, но и по качественным показателям, что заложено в нормативах (например, СП 519.1325800.2023).

Перспективы дальнейших исследований видятся в развитии данной модели в нескольких направлениях: реализация и тестирование алгоритма в реальной SDN-среде (например, на базе контроллеров ONOS или OpenDaylight), разработка механизма динамической адаптации нормализующих коэффициентов в зависимости от текущего состояния сети и типа трафика, а также интеграция

методов машинного обучения для прогнозирования нагрузки и проактивной балансировки.

В заключение можно утверждать, что разработанная модель балансировки нагрузки на уровне контроллеров и уровне коммутаторов на базе параметров QoS представляет собой целостное и эффективное решение актуальной проблемы управления в сложных мультиконтроллерных сетях SDN.

ГЛАВА 4 МЕТОДИКА МНОГОКРИТЕРИАЛЬНОЙ БАЛАНСИРОВКИ ТРАФИКА В SDN

4.1 Принципы работы с комбинированным параметром QoS

Методика динамического перераспределения нагрузки функционирует по принципу непрерывного циклического процесса, который можно представить, как последовательность этапов: мониторинг, анализ, принятие решения, выполнение действия, мониторинг. Цикличность позволяет системе постоянно адаптироваться к изменяющимся условиям сети, реагируя на возникающие или прогнозируемые перегрузки.

Основные принципы методики:

1. Оптимизация ресурсов: главная цель — равномерно распределить нагрузку по всем доступным коммутаторам, избегая концентрации трафика на отдельных коммутаторах, что может привести к их перегрузке и деградации сервиса.
2. Автономность: после запуска алгоритм функционирует без вмешательства, автоматически принимая решения на основе собранных данных.
3. Простота и вычислительная эффективность: базовая реализация алгоритма EWRR [67-68] ориентирована на непрерывную обработку данных и принятие решений, что важно для высоконагруженных систем. Цикличность обеспечивает непрерывную обработку данных, где каждое принятое решение приводит к изменению состояния сети, которое затем снова анализируется для следующего цикла принятия решений.

4.2 Методика балансировки трафика коммутаторов по комбинированному параметру QoS

Рассмотрим этапы методики балансировки трафика коммутаторов используя алгоритм EWRR (рисунок 4.1, где X – максимальный порог нагрузки на коммутатор).



Рисунок 4.1 — Методика балансировки трафика коммутаторов по комбинированному параметру QoS

Основой для принятия решения является актуальная и точная информация о состоянии сети. В разработанной модели реализован непрерывный механизм сбора и первичного анализа параметров коммутаторов (нагрузка, параметры QoS) [86].

Параметры коммутатора анализируются с заданной частотой (например, несколько раз в секунду или через фиксированные интервалы указанного модельного времени). Это обеспечивает актуальность данных, что критически важно для оперативного реагирования на быстрые изменения нагрузки и параметров QoS.

1. Поступает поток запросов на обслуживание.
2. Нормализация параметров QoS запроса и вычисление комбинированного параметра QoS для коммутаторов S_1, \dots, S_m .

Для балансировки нагрузки необходимо учитывать параметры QoS коммутаторов по параметрам, которые значимы в данной сети. Для повышения качества клиентских сервисов необходимо при балансировке нагрузки учитывать не только нагрузку на коммутаторе, но и параметры QoS для запроса на коммутаторе [33, 36].

Как образец требований к качеству клиентских сервисов возможно взять общедоступные договора Российских операторов связи, в них указываются конкретные параметры QoS для разного типа трафика и уровня SLA.

Проанализируем параметры QoS в договорах крупных поставщиков услуг: Ростелеком, Билайн, Транстелеком, Связьтранснефть.

Проанализировав параметры QoS в договорах операторов связи. Делаем вывод, что все операторы связи указывают три параметра QoS: вариация сетевой задержки, круговая сетевая задержка, потери пакетов. Значения данных параметров варьируется от уровня обслуживания – SLA.

Таким образом, для балансировки трафика на коммутаторах необходимо учитывать и улучшать параметры QoS: вариация сетевой задержки, круговая сетевая задержка, потери пакетов. Для повышения качества клиентских сервисов модель балансировки нагрузки должна учитывать не только нагрузку на узлы сети, но и параметры QoS.

При балансировке сетевого трафика на базе параметров QoS учитываются параметры с разными единицами измерения и возникает вопрос о нормализации параметров QoS, так как они различны по своим единицам измерения.

Каждый из параметров QoS оказывает разное влияние на сервис, поэтому необходимо оценить «вес» каждого параметра при балансировке трафика. Для этого определяем нормализующий коэффициент C_{ij} для каждого параметра. Проанализировав договоры операторов связи, ITU-T Recommendation Y.1541 [33].

Методика балансировки нагрузки на примере коммутаторов на базе параметров QoS будет выглядеть так: чтобы решить, на какой коммутатор перенаправить соединение, оркестратору необходимо сравнить значения соответствующих параметров для каждого коммутатора. Для этого необходимо сформировать из собранных параметров матрицу E (4.1) с e_{hf} , где h - порядковый номер параметра ($h=1, \dots, r$), $r=3$; f - номер коммутатора ($f=1, \dots, m$).

- 1) Потери пакетов – ℓp ;
- 2) Круговая сетевая задержка – p_n ;
- 3) Вариация сетевой задержки – j_t .

Разное влияние на сервис оказывает каждый из параметров. Поэтому необходимо оценить влияние каждого параметра, определяем нормализующий коэффициент D_{hf} для каждого параметра.

$$\sum_{f=1}^m \sum_{h=1}^3 D_{hf} = 1$$

3. Элемент e_{hf} в матрице E (4.1) умножается на нормализующий коэффициент D_{hf} .

$$E = \|e_{hf}\|_{(3 \times m)} = \begin{pmatrix} D_{11}e_{\ell p_1} & D_{21}e_{\ell p_2} & \dots & D_{m1}e_{\ell p_m} \\ D_{12}e_{p_{n_1}} & D_{22}e_{p_{n_2}} & \dots & D_{m2}e_{p_{n_m}} \\ D_{13}e_{j_{t_1}} & D_{23}e_{j_{t_2}} & \dots & D_{m3}e_{j_{t_m}} \end{pmatrix} \quad (4.1)$$

Таким образом, получаем информацию о коммутаторе с максимальной средневзвешенной оценкой параметров QoS.

4. Определение перегрузки: определение, не превышает ли значение нагрузки с данным запросом на коммутаторе максимально возможную. $Y_s + Y$ запроса $< X$.

5. Запрос направляется на выбранный коммутатор с максимальным QoS или направление запроса в оркестратор для повторной обработки (при перегрузке коммутаторов).

Методика обеспечивает в режиме реального времени мониторинг состояния всех моделируемых коммутаторов сетевой инфраструктуры, который служит основой для принятия решения.

4.3 Апробация методики

Для проверки методики балансировки нагрузки по комбинированному параметру QoS создана программа для ЭВМ под названием «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования» [87, 88, 89].

Программа для ЭВМ позволяет пользователю взаимодействовать с серверной системой через интуитивно понятный интерфейс, обеспечивая визуализацию результатов имитационного моделирования в виде графиков. Программа для ЭВМ обеспечивает настройку параметров эксперимента с помощью удобных ползунков и панели управления, позволяя гибко изменять условия моделирования. Программа для ЭВМ способствует ускоренному анализу данных, предоставляя пользователю возможность наблюдать динамическое поведение моделей в реальном времени. Программа для ЭВМ осуществляет установку соединения с сервером и передачу параметров эксперимента, обеспечивая точную настройку и мониторинг работы имитационной модели.

Разработанная имитационная модель представляет собой программную реализацию, предназначенную для воспроизведения поведения сетевой инфраструктуры в контролируемой среде. Архитектура модели построена на

принципе дискретно-событийного моделирования, что позволяет эффективно моделировать прохождение трафика через типовые сегменты сети.

Сетевая инфраструктура моделируется как совокупность независимых узлов сети, характеристики которых настраиваются индивидуально. Взаимодействие между узлами осуществляется через механизмы динамического перенаправления трафика, имитирующие изменение маршрутизации в реальных сетях.

Ключевое внимание уделяется моделированию источников нагрузки, которые характеризуются интенсивностью генерации трафика и возможностью динамического переключения целевого узла сети. Это позволяет методике балансировки перенаправлять потоки трафика между узлами в реальном времени.

Для обеспечения управления системой непрерывно собираются ключевые метрики состояния каждого узла (нагрузка, параметры QoS). На основе этих данных работает алгоритм динамического перераспределения нагрузки, функционирующий по циклическому принципу: мониторинг, анализ, принятие решения, выполнение действия. Этот процесс позволяет системе адаптироваться к изменяющимся условиям сети и минимизировать перегрузки.

Программная реализация системы построена по клиент-серверной архитектуре. Клиентская часть отвечает за взаимодействие с пользователем, настройку экспериментов и визуализацию данных. Серверная часть реализует логику имитационного моделирования, расчёт метрик и работу методики балансировки.

Основные принципы проектирования включают модульность, расширяемость и интерактивность. Взаимодействие между компонентами обеспечивает управление моделированием в реальном времени, визуализацию состояния сети и сохранение результатов для последующего анализа.

В левой части экрана находится скрываемая панель управления экспериментом (рисунок 4.2).

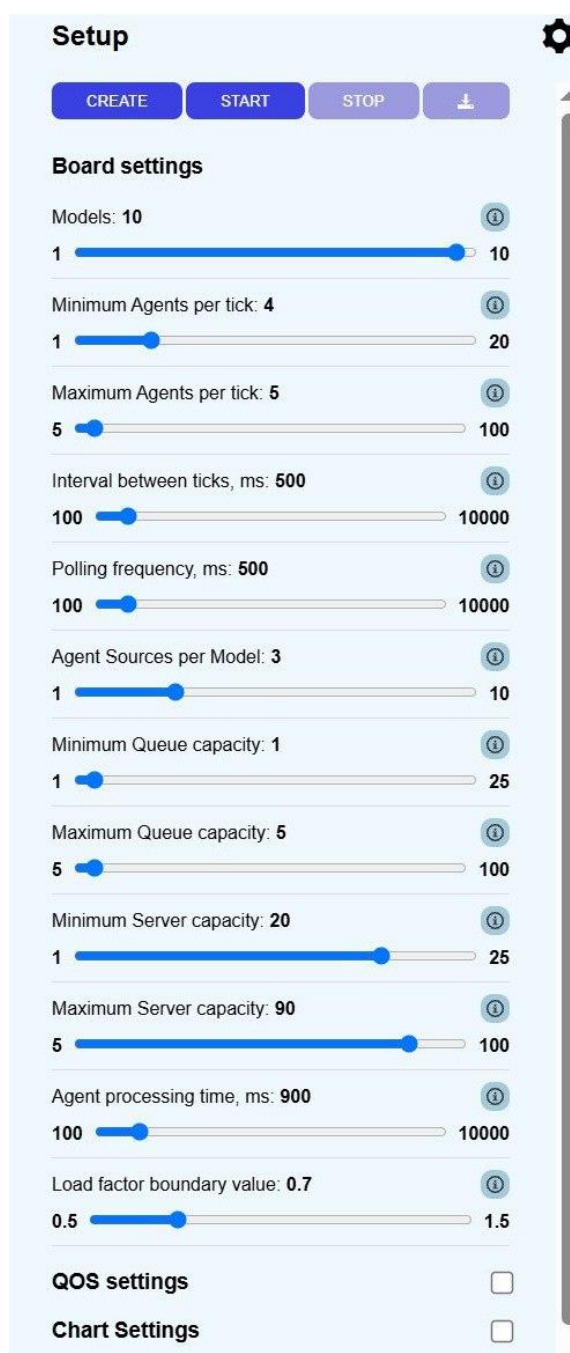


Рисунок 4.2 — Панель управления и конфигурации параметров эксперимента

В верхней части меню «Setup» части меню расположены кнопки команд для управления экспериментом (рисунок 4.3):

1. «create» - создание эксперимента с текущими характеристиками;
2. «start» - запуск эксперимента;

3. «stop» - завершение эксперимента;
4. «download» - загрузка результатов эксперимента в формате Excel.

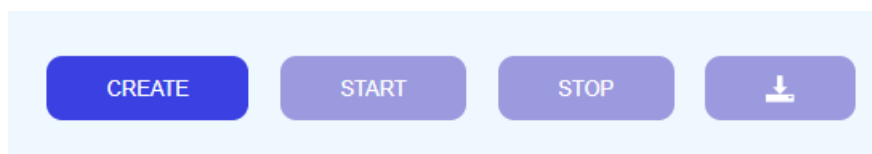


Рисунок 4.3 — Панель команд для контроля над экспериментом.

Часть меню «Board settings» позволяет пользователю управлять характеристиками эксперимента. При изменении любого из параметров необходимо пересоздать эксперимент, чтобы настройки стали актуальными для сервера.

Для настройки моделей эксперимента используются слайдеры настройки:

- количества моделей в эксперименте (минимальное значение = 1, максимальное значение = 10);
- минимального значения диапазона количества приходящих в модель агентов в эксперименте (минимальное значение = 1, максимальное значение = 20);
- максимального значения диапазона кол-ва приходящих в модель агентов в эксперименте (минимальное значение = 5, максимальное значение = 100);
- интервала времени в мс прихода агентов в модели в эксперименте (минимальное значение = 100, максимальное значение = 10000);
- интервала времени в мс взятия статистики с моделей в эксперименте (минимальное значение = 100, максимальное значение = 10000);
- количества Source элементов в каждой модели в эксперименте (минимальное значение = 1, максимальное значение = 10);
- минимального значения диапазона количества мест в Queue элементе у моделей в эксперименте (минимальное значение = 1, максимальное значение = 25);

- максимального значения диапазона количества мест в Queue элементе у моделей в эксперименте (минимальное значение = 5, максимальное значение = 100);
- минимального значения диапазона количества мест в Delay элементе у моделей в эксперименте (минимальное значение = 1, максимальное значение = 25);
- максимального значения диапазона количества мест в Delay элементе у моделей в эксперименте (минимальное значение = 5, максимальное значение = 100);
- значения времени обслуживания агента в мс в Delay элементе у моделей в эксперименте (минимальное значение = 100, максимальное значение = 10000);
- значения коэффициента загрузки, при котором происходит принятие решения о переводе потока агентов в эксперименте (минимальное значение = 0.1, максимальное значение = 1).

Балансировка нагрузки на коммутаторах по параметрам QoS включается при помощи флага. При активном флаге предоставляется возможность управлять параметрами контроля по QoS в части меню «QoS settings» (рисунок 4.4).

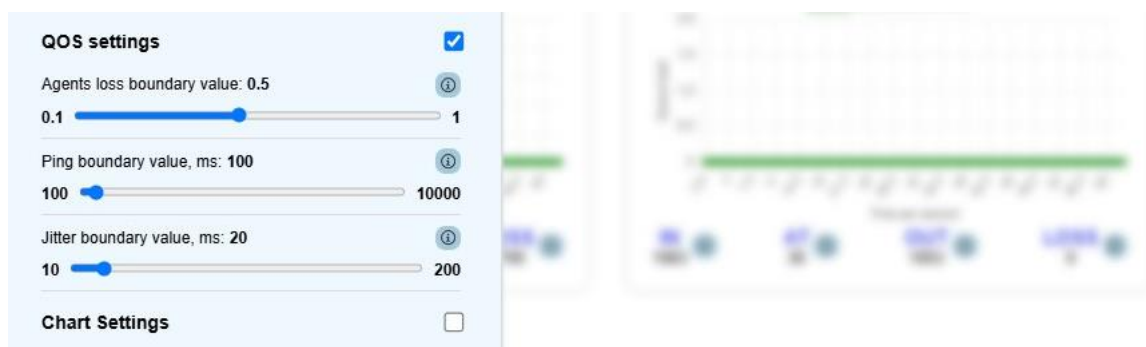


Рисунок 4.4 — Параметры QoS эксперимента

Для настройки QoS в эксперименте используются слайдеры настройки для агентов:

– значения потери пакетов (loss), которые задают значение QoS потери пакетов (минимальное значение находится в диапазоне от 0 до 0,1 %, максимальное – от 0 до 1 %);

– значения круговой сетевой задержки (ping), которая задает значение QoS круговой сетевой задержки (минимальное значение находится в диапазоне от 0 до 100 мс, максимальное – от 0 до 10000 мс);

– значения вариация сетевой задержки (jitter), которое задает значение QoS вариация сетевой задержки (минимальное значение находится в диапазоне от 0 до 10 мс, максимальное значение – от 0 до 200 мс);

– количества хранимых и отображаемых на графиках данных эксперимента.

Если флаг настроек системы визуализации статистики не активен, то все данные будут отображаться на графиках и не будут сохранены в файл ModelsParametersStatistic.exsx. При активном флаге появляется слайдер для контроля за сохраняемыми и отображаемыми данными на графиках эксперимента.

Область эксперимента (рисунок 4.5). На ней расположены блоки информации о каждой модели. Отображаются два графика: нагрузка на моделируемый коммутатор и очередь на обслуживание. Область обновляется в реальном времени, интервал обновления задаётся в панели управления.

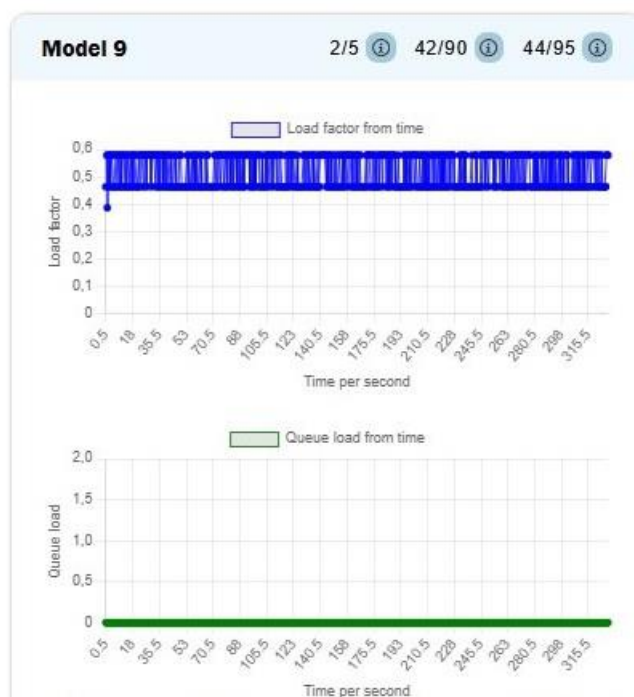


Рисунок 4.5 — Область эксперимента на примере модели 9

4.4 Экспериментальная апробация

Проведем эксперимент с помощью программы «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования» для проверки алгоритма.

Зададим параметры имитационного моделирования в программе. Эксперимент проводится на 10 коммутаторах.

1. Диапазоны значений параметров QoS:

- L1, L2, L3, ..., L10 – параметр потери пакетов (Loss) в диапазоне от 0 до 0,1 %;
- P1, P2, P3, ..., P10 – параметр круговая сетевая задержка (Ping) в диапазоне от 1 до 100 мс;

• $J_1, J_2, J_3, \dots, J_{10}$ – параметр вариация сетевой задержки (Jitter) в диапазоне от 0 до 20 мс;

2. Количество потоков данных 1000 в секунду.

3. Максимальная загрузка коммутаторов 70 %.

С указанными параметрами в панели эксперимента (рисунок 4.6) провели эксперимент.

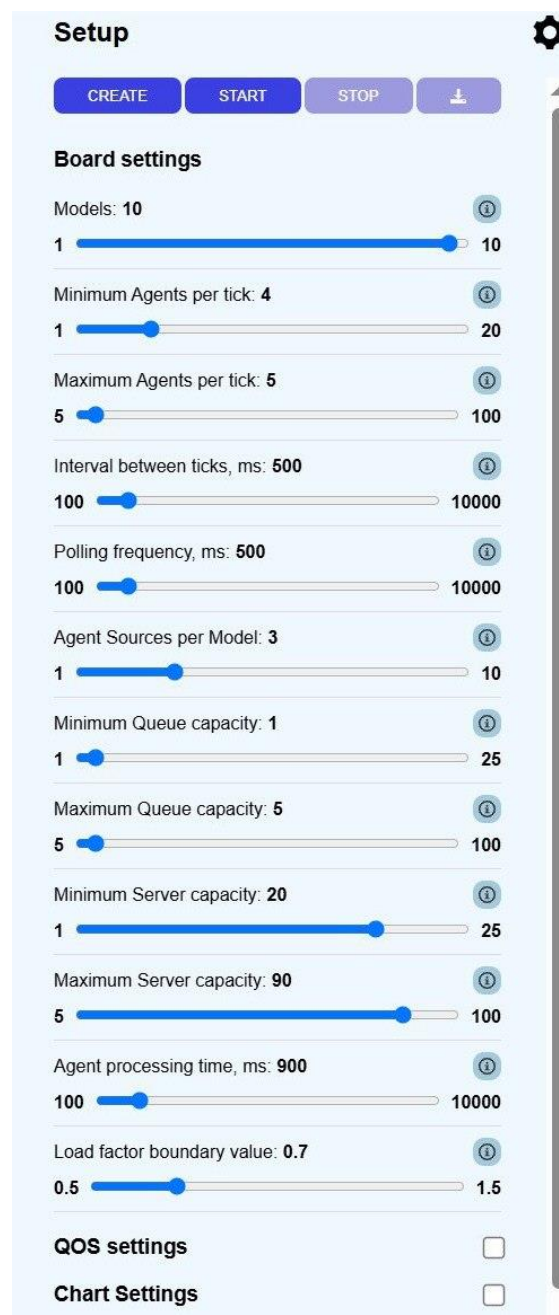


Рисунок 4.6 — Панель с конфигурацией параметров эксперимента

Графическое изображение эксперимента представлено на рисунке 4.7.

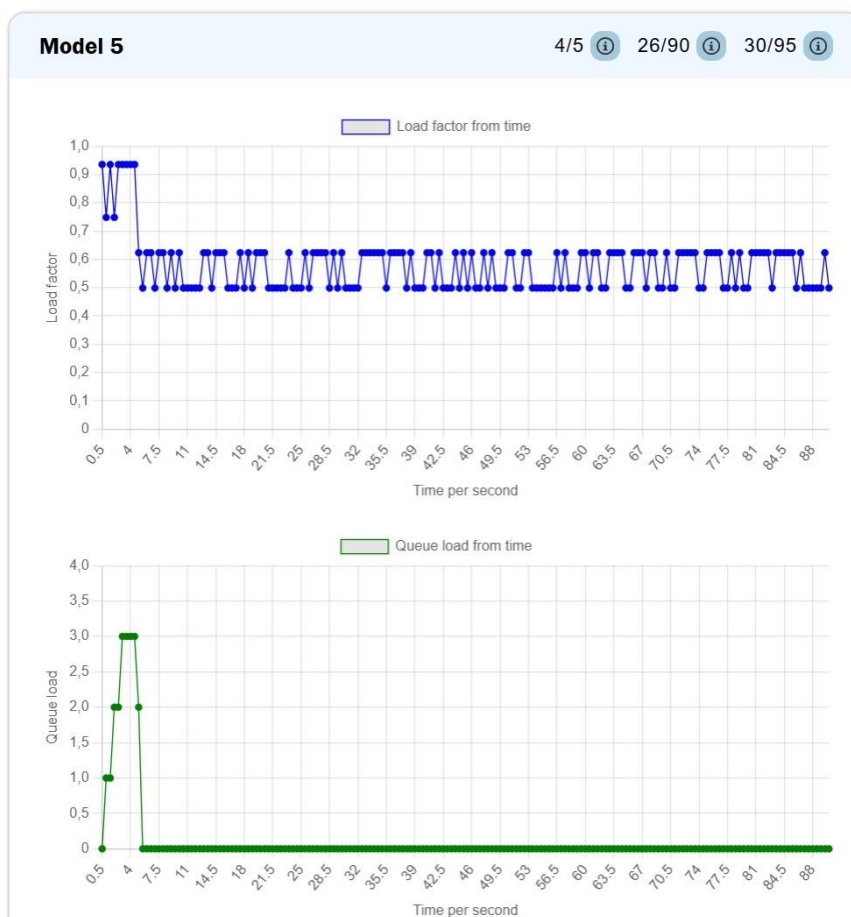


Рисунок 4.7 — Графическое изображение эксперимента на примере модели 5

На снимке экрана (рисунок 4.7) продемонстрирована балансировка трафика для модели 5. Видно, что на коммутатор номер 5 в начале эксперимента поступает большое количество заявок на обслуживание и образуется очередь из агентов на обслуживание. Нагрузка на модель 5 равна 0,95 до 4 секунды и далее становится равномерной на всем этапе эксперимента. С 5 секунды эксперимента алгоритм равномерно распределяет нагрузку (агентов) по моделям и график нагрузки становится равномерным и однородным для всего остального времени эксперимента. В тоже время с 5 секунды равномерное распределение трафика по другим агентам демонстрирует, что очередь на данную модель становится равна 0, то есть каждый поступивший запрос обрабатывается сразу, то есть нагрузка на все коммутаторы распределяется корректно и равномерно. Таким образом график, представленный на рисунке 9 отображает корректную работу

алгоритма в части балансировки трафика и распределения поступающих запросов на все модели эксперимента с равномерной нагрузкой на каждую модель из 10.

Графическое изображение эксперимента представлено на рисунке 4.8.

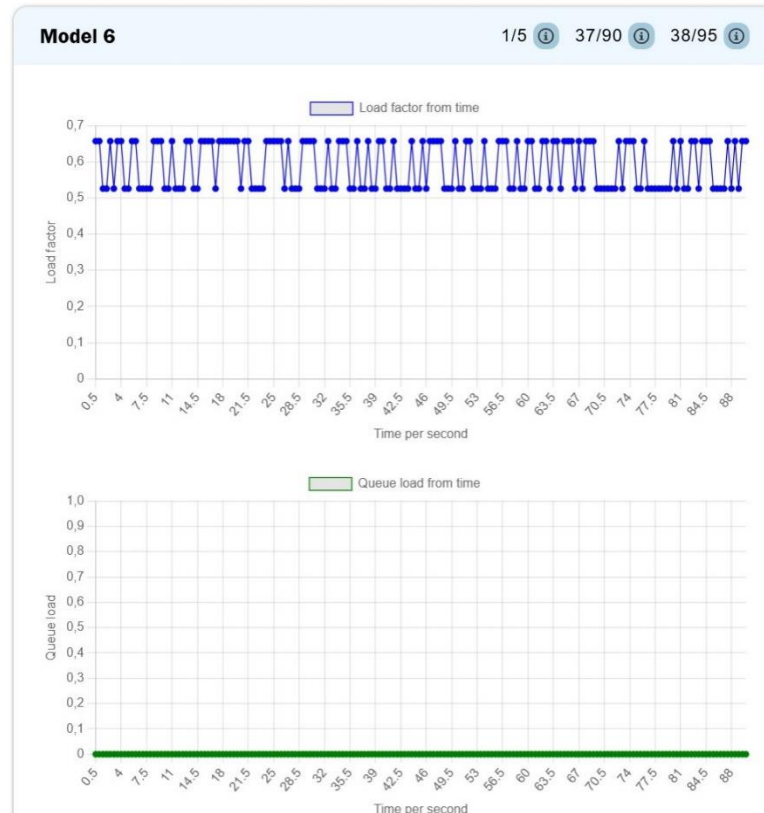


Рисунок 4.8 — Графическое изображение эксперимента на примере модели 6

На модель 6 представленную на скриншоте (рисунок 4.8) с начала эксперимента равномерно распределяется трафик, очередь из агентов (запросов) отсутствует. Параметры нагрузки на модель 6 соответствуют заданным значениям эксперимента и не выходят за пределы обозначенной максимальной нагрузки 0,7.

По результатам работы программы «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования» проведено два эксперимента по балансировке нагрузки между коммутаторами. Один эксперимент с учетом нагрузки, а второй по параметрам QoS. Создается файл с результатами статистических данных эксперимента с

названием ModelsParametersStatistic.exsx. На основании этих данных построены графики (рисунок 4.9).

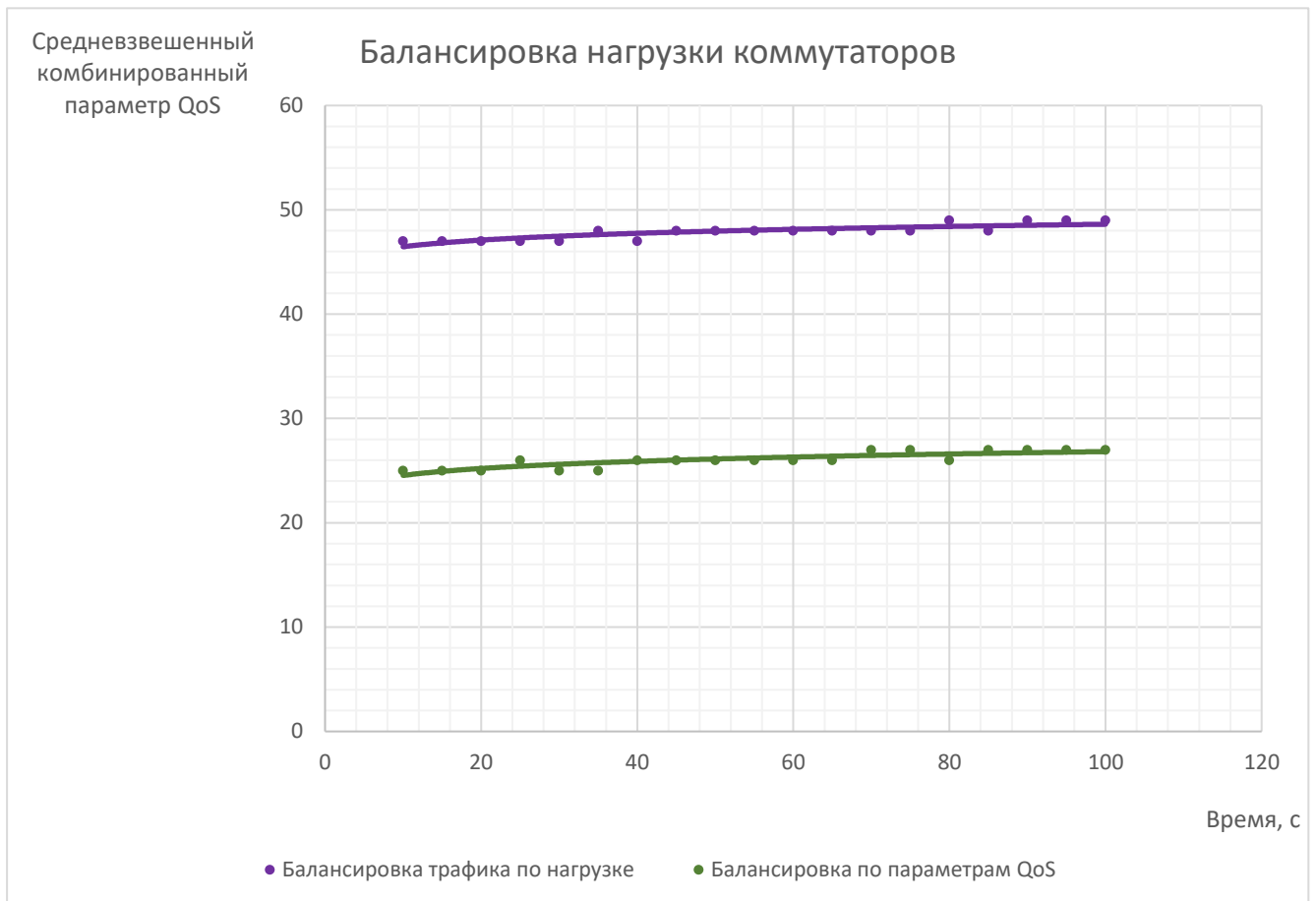


Рисунок 4.9 — Балансировка нагрузки коммутаторов

При использовании методики балансировки только по нагрузке комбинированный показатель QoS остаётся на относительно низком уровне, с заметными колебаниями, что свидетельствует о нестабильности качества обслуживания.

При применении предложенной методики балансировки на базе комбинированного параметра QoS показатель значительно повышается, демонстрируя устойчивое улучшение качества обслуживания.

Предложенная методика обеспечила улучшение комбинированного показателя QoS на 46,0 % по сравнению с методом балансировки, учитывающим только нагрузку на узлы сети. Это подтверждает эффективность интеграции

параметров QoS (потери пакетов, круговой сетевой задержки, вариации сетевой задержки) при балансировке трафика в сетях SDN.

На рисунке 4.10 представлены результаты расчетного и экспериментального методов балансировки по нагрузке и по параметрам QoS.

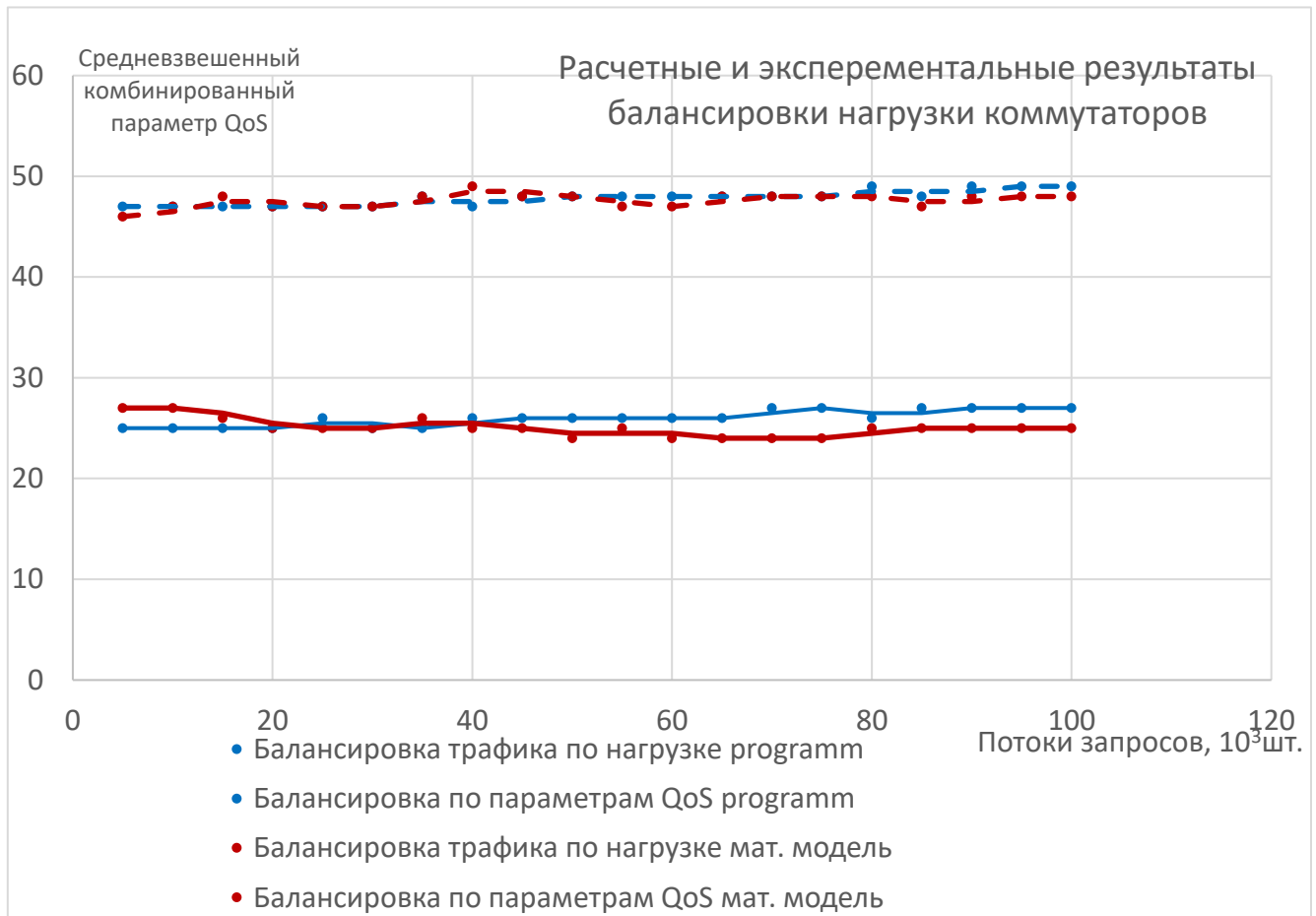


Рисунок 4.10 — Результаты расчетного и экспериментального методов балансировки по нагрузке и по параметрам QoS

Результаты балансировки нагрузки коммутаторов аналитической и расчетной модели имеют расхождение: при балансировке по нагрузке 2 %, а по параметрам QoS 8 %.

4.5 Выводы по главе 4

В ходе проведенного исследования была успешно разработана и апробирована методика балансировки трафика на базе комбинированного параметру QoS для сетей SDN.

Разработана методика балансировки, основанный на принципах цикличности (мониторинг – анализ – решение – действие), автономности и вычислительной эффективности. В его основу положен модифицированный алгоритм EWRR, дополненный механизмом учёта трёх ключевых параметров QoS: потери пакетов, круговой сетевой задержки, вариации сетевой задержки.

Предложен механизм нормализации разнородных параметров QoS, позволяющий приводить значения с разными единицами измерения к единой безразмерной шкале. На основе анализа SLA крупных российских операторов связи (Ростелеком, Билайн, Транстелеком, Связьтранснефть) были определены весовые коэффициенты для каждого параметра, что обеспечивает объективность и адаптивность принятия решений.

Разработана и зарегистрирована программа для ЭВМ – «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования». Программа предоставляет интуитивно понятный интерфейс для настройки параметров эксперимента, управления имитационными моделями и визуализации результатов в реальном времени, что существенно ускоряет процесс анализа и верификации.

Проведена экспериментальная апробация методики в контролируемых условиях. Результаты экспериментов подтвердили эффективность предложенного методика:

- Обеспечено эффективное распределение нагрузки между коммутаторами.

- Продемонстрирована способность системы оперативно реагировать на пиковые нагрузки и перераспределять трафик для предотвращения перегрузок отдельных узлов сети.
- Комбинированный показатель QoS улучшился на 46,0 % по сравнению с балансировкой, учитывающей только загрузку узлов.

Таким образом, предложенная методика балансировки трафика по комбинированному параметру QoS доказала свою практическую значимость и эффективность. Она позволяет не только оптимизировать использование сетевых ресурсов, но и гарантировать соблюдение согласованных уровней качества обслуживания (SLA), что является критически важным для операторов связи и поставщиков облачных сервисов. Дальнейшие исследования могут быть направлены на адаптацию методики для более сложных сетевых топологий и интеграцию с системами прогнозирования нагрузки.

ЗАКЛЮЧЕНИЕ

По результатам исследований в диссертационной работе были получены следующие основные результаты:

1. Исследована научная литература в области изучения и анализа трафика и методов балансировки нагрузки в SDN, выявлены их недостатки: отсутствие многопараметрического подхода к QoS, игнорирование нормализации разнородных параметров и требований SLA для разных классов трафика.
2. Обоснован выбор алгоритма EWRR как базового для построения новой методики, показавшего наилучшие результаты по критериям производительности, возможности учета разнородных параметров QoS адаптирующийся к различным классам сервиса.
3. Проанализированы рекомендации QoS (ITU-T E.860, ITU-T Y.1541, 3GPP TS 23.501) и QoS крупных российских операторов связи, что легло в основу формирования весовых коэффициентов для нормализации параметров QoS.
4. Разработана модель балансировки нагрузки для мультиконтроллерной SDN, учитывающая параметры QoS как на уровне контроллеров (задержка обработки, потерянные пакеты), так и на уровне коммутаторов (потери пакетов, круговая сетевая задержка, вариация сетевой задержки), решающая проблему сопоставления разнородных метрик QoS через систему весовых коэффициентов.
5. Предложена модель многокритериальной балансировки трафика на основе комбинированного параметра QoS, включающая: механизм нормализации разнородных параметров QoS (потери пакетов, круговая сетевая задержка, вариация сетевой задержки); алгоритм вычисления взвешенной оценки состояния сетевых элементов; процедуру динамического перераспределения трафика с учётом класса обслуживания (Real Time, Best Effort).
6. Получены расчетные значения показателей QoS в рамках разработанной модели сети: на уровне контроллеров: улучшение комбинированного параметра

QoS на 22,4 % по сравнению с балансировкой только по нагрузке; на уровне коммутаторов: улучшение комбинированного параметра QoS на 47,9 %.

7. Для апробации методики многокритериальной балансировки трафика на основе комбинированного параметра QoS выполнена программная реализация для имитационного моделирования.

8. Разработана и зарегистрирована программа для ЭВМ «Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования» (свидетельство № 2025662025 от 16.05.2025).

9. Программно апробирована методика многокритериальной балансировки трафика на основе комбинированного параметра QoS. Получены результаты, схожие с расчетными: улучшение общего показателя QoS на 46,0 % в сравнении с традиционными методами балансировки, распределяющими трафик на основе только параметра нагрузки элементов сети.

10. Подтверждена возможность выполнения строгих требований QoS для приложений (VoIP, видео, онлайн-игры) чувствительных к задержкам.

11. Результаты работы могут быть использованы: операторами связи для гарантированного выполнения требований QoS и повышения качества услуг; разработчиками SDN-решений для создания интеллектуальных систем управления трафиком; при проектировании сетей 5G/6G, промышленного IoT и облачных платформ, требующих высоких стандартов QoS.

12. В диссертации решена актуальная научная задача создания эффективного механизма балансировки нагрузки в SDN с гарантированным качеством обслуживания. Полученные результаты имеют теоретическую и практическую ценность для развития телекоммуникационных технологий.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

ГМ	Графовая модель
МСЭ-Т	Сектор стандартизации электросвязи Международного союза электросвязи (МСЭ)
ПКС	Программно-конфигурируемая сеть
СМО	Система массового обслуживания
ТИ	Теория игр
ЦП	Центральный процессор
ЭВМ	Электронная вычислительная машина
API	Application Programming Interface
CECT	Computationally Efficient Congestion-avoidance and Traffic Engineering
CLI	Command Line Interface
CPN	Coloured Petri Net
CPU	Central Processing Unit
DQN	Deep Q-Network
EFLB-IIoT	Enhanced Flow Control and Load Balancing
EWRR	Enhanced Weighted Round Robin
FTP	File Transfer Protocol
GBR	Guaranteed Bit Rate
gNMI	gRPC Network Management Interface
gRPC	Remote Procedure Calls
HMLR-AC	Hybrid Multi-constraint Lagrangian Relaxation based Aggregated Cost
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force IoT Internet of Things

IPDV	IP Packet Delay Variation
IPLR	IP packet Loss Ratio
IPTD	IP packet Transfer Delay
ITU	International Telecommunication Union
ITU-T	International Telecommunication Union Telecommunication Standardization Sector
MARL	Multi-Agents Reinforcement Learning
MCPTT	Mission Critical Push-to-Talk
MDCLB	Multiple Distributed Controller Load Balancing
MP	Measurement Point
MPLS	Multiprotocol Label Switching
NMS	Network Management System
P4RT	P4 Runtime
PDB	Packet Delay Budget
PDV	Packet Delay Variation
PPO	Proximal Policy Optimization
PSO	Particle Swarm Optimization
ONF	Open Networking Foundation
QoS	Quality of Service
RFC	Request for Comments
RR	Round Robin
SDN	Software-Defined Networking
SDN-IoT	SDN Internet of Things
SDN-DVFS	Dynamic Voltage Frequency Scaling
SLA	Service Level Agreement
SNMP	Simple Network Management Protocol
SSH	Secure Shell
TCP	Transmission Control Protocol
TLS	Transport Layer Security

V2X	Vehicle-to-Everything
VoIP	Voice over IP
VPN	Virtual Private Network
VSDN	Video over Software-Defined Networking
W Least Connections	Weighted Least Connections
WRR	Weighted Round Robin

СПИСОК ЛИТЕРАТУРЫ

1. Recommendation ITU-T Y.3300 «Framework of software-defined networking». ITU-T, Geneva, June, 2014.
2. ONF TR-502. SDN architecture. [Электронный ресурс] / Режим доступа: https://opennetworking.org/wp-content/uploads/2013/02/TR_SDN_ARCH_1.0_06062014.pdf. 2014.
3. Ефимушкин В.А. Обзор решений SDN/NFV зарубежных производителей / В.А. Ефимушкин, Т.В. Ледовских, Д.М. Корабельников, Д.Н. Языков // Т-Comm: Телекоммуникации и транспорт. – 2015. – Т. 9. №8. – С. 5-13.
4. RFC 7426 Software-Defined Networking (SDN): Layers and Architecture Terminology [Электронный ресурс] / IETF.org. – Режим доступа: <https://tools.ietf.org/html/rfc7426>.
5. McKeown, N. OpenFlow: Enabling Innovation in Campus Networks / Anderson T., Balakrishnan H., Parulkar G., Peterson L., Rexford J., Shenker S., Turner J. // ACM SIGCOMM Computer Communication Review. — 2008. — Vol. 38, No. 2. — Pp. 69–74.
6. SDN Architecture Overview. Version 1.0. Open Networking Foundation. TR-521. 2013. https://opennetworking.org/wp-content/uploads/2014/10/TR-521_SDN_Architecture_overview.pdf (Accessed 20.01.2026)
7. Дмитриева Ю.С. Модель принятия решения при миграции виртуальных систем в SDN / Ю.С. Дмитриева // Актуальные проблемы инфотелекоммуникаций в науке и образовании: сборник научных статей: в 4х томах, Санкт-Петербург, 27 февраля – 28 февраля 2024 года / Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича. Том 1. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2024. – С. 282-287.

8. Braun, W. Software-Defined Networking Using OpenFlow: Protocols, Applications and Architectural Design Choices / Braun, W., Menth, M. // *Future Internet*. 2014. 6, 302-336.
9. Kreutz, D. Software-Defined Networking: A Comprehensive Survey / Kreutz, D., Ramos, F. M., Verissimo, P., Rothenberg, C. E., Azodolmolky, S., Uhlig, S. // *Proceedings of the IEEE*. 2015. 103(1), 14-76.
10. Göransson, P., Black, C., Culver, T. Software Defined Networks: A Comprehensive Approach / Göransson, P., Black, C., Culver, T. Elsevier Science. 2014. 352 pp.
11. Feamster, N. The road to SDN: an intellectual history of programmable networks / Feamster, N., Rexford, J., Zegura, E. // *ACM SIGCOMM Computer Communication Review*. 2014. 44(2), pp. 87-98.
12. Tijare, P. The Northbound APIs of Software Defined Networks / Tijare, P., Deepika V. // *International Journal of Engineering Sciences & Research Technology*. 2016. 5(10). Pp. 501-513.
13. Tony-Mayeko, A. SDN introduction and prospect / Tony-Mayeko A. // *The Journal of Scientific and Engineering Research*. 2024. 11(1). Pp. 231-237.
14. Alsmadi, I.M. A Systematic Literature Review on Software-Defined Networking / I.M. Alsmadi, I. AlAzzam, M. Akour // *Information Fusion for Cyber-Security Analytics. Studies in Computational Intelligence*. – 2017. – Vol. 691. – P. 333-369.
15. Benzekki, K. Software-defined networking (SDN): a survey / K. Benzekki, A. El Fergougui, A. Elbelrhiti Elalaoui // *Security and Communication Networks*. – 2016. – V. 9 (18). – P. 5803-5833.
16. Ksentini, A. Toward enforcing network slicing on RAN: Flexibility and resources abstraction / A. Ksentini, N. Nikaein // *IEEE Communications Magazine*. – 2017. – Vol. 55. – No. 6. – P. 102-108.
17. Xia, W. A Survey on Software-Defined Networking / W. Xia, Y. Wen, C.H, Foh, D. Niyato, H. Xie // *IEEE Communications Surveys & Tutorials*. – 2015. – Vol.17. – No. 1. – P. 27-51.

18. Ryait, D.K., Sharma, M. / Performance Evaluation of SDN Controllers // Inventive Communication and Computational Technologies / Proceedins of ICICCT 2023. Lecture Notes in Networks and Systems, vol. 757. Springer, Singapore. P. 1009-1021.
19. Rathee, S. Analysis and Performance Evaluation of Different Methods to Achieve Way-Point Enforcement in Hybrid SDN / Rathee S., Haribabu K., Patel, P., Bhatia A., Gandhi, U. // Advances in Intelligent Systems and Computing. pp 190–200. 2020. DOI: 10.1007/978-3-030-44038
20. Sahep, S. Enhancing Virtual Network Performance Using Software-Defined Networking (SDN) / S. Sahep // Muthanna Journal of Pure Science 12(2). 2025. DOI: 10.52113/2/12.02.2025/188-196.
21. Владыко А.Г. Программируемые сети SDN / А.С. Мутханна, Р.В. Киричек, А.Н. Волков, М.А. Маколкина, А.И. Парамонов // Учебное пособие. Издательство «Лигр», Санкт-Петербург. – 2019. — С. 120.
22. Rupani, K., Punjabi, N., Shamdasani, M., Chaudhari, S. / Dynamic Load Balancing in Software-Defined Networks Using Machine Learning // Proceeding of International Conference on Computational Science and Applications. — 2020. — P. 283–292.
23. Astutoz, B.N. A survey of software-defined networking: past, present, and future of programmable networks / B.N. Astutoz, M. Mendonca, X.N. Nguyen, K. Obraczka, T. Turletti // IEEE Communications Surveys & Tutorials. 2014. – Vol. 16. – No. 3. – P. 1617-1634.
24. Krishnan, P. OpenStackDP: a scalable network security framework for SDN-based OpenStack cloud infrastructure / P. Krishnan, K. Jain, A. Aldweesh, P. Prabu, R. Buyya // Journal of Cloud Computing 12, 26. 2023. <https://doi.org/10.1186/s13677-023-00406-w>.
25. Tapus, N. Comparative Analysis of Centralized and Distributed SDN Environments for IoT Networks / N. Tapus, K. Yalda, Diyar J. Hamad. // Control Engineering and Applied Informatics. 2024. Vol.26, No. 3, pp. 84-91. DOI: 10.61416/ceai.v26i3.9164.

26. Лапони́на О.Р., Сухомли́н В.А. / Способы трансформации сетей к SDN-архитектуре // International journal of information technologies. – 2015. – Vol. 3, № 4.
27. Salsano, S. SDN Architecture and Southbound APIs for IPv6 Segment Routing Enabled Wide Area Networks / P. L. Ventre, M.M. Tajiki, S. Salsano, C. Filsfils // IEEE Transactions on Network and Service Management. 2018. Vol. 15, NO. 4. PP (99). DOI: 10.1109/TNSM.2018.2876251.
28. Ghosh, A. T. A study on load balancing techniques in SDN / A. Ghosh, T. Manoranjitham // International Journal of Engineering & Technology. 2018. 7(2.4). Pp. 174-177.
29. Open Networking Foundation. Software-Defined Networking: The New Norm for Networks: White Paper. April 13, 2012. 12 p.
URL: <https://opennetworking.org/sdn-resources/white-papers/software-defined-networking-the-new-norm-for-networks/> (дата обращения: 23.02.2026).
30. Blial, O. An Overview on SDN Architectures with Multiple Controllers / O. Blial, B.M. Mamoun, B. Redouane // Journal of Computer Networks and Communications. 2016. Pp. 1-8.
31. Semenovych, A. Comparative analysis of SDN controllers / A. Semenovych, O. Laponina // International Journal of Open Information Technologies. 2018. Vol 6, No 7. P. 50-56.
32. ITU-T Recommendation E.860. Framework of a service level agreement = Рекомендация МСЭ-Т E.860. Структура соглашения об уровне услуг: дата введения 2002–06. Женева: Международный союз электросвязи, 2002. 26 с. URL: <https://www.itu.int/rec/T-REC-E.860/en> (дата обращения: 23.02.2026).
33. ITU-T Recommendations Y.1541 «Internet protocol aspects – Quality of service and network performance. Network performance objectives for IP-based services». ITU-T, Geneva, December 2011.
34. 3GPP TS 23.501 V15.4.0 (2018-12). 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Release 15).

35. Espvik, O. The EURESCOM Quality of Service framework (P806) — theory and practice / O. Espvik // EURESCOM mess@ge. — 2000. — № 2. — С. 4—5. https://www.eurescom.eu/wp-content/uploads/2021/11/message02_2000.pdf
36. МСЭ-T E.800 (09/2008). Determination of terms related to quality of service and network performance including dependability. ITU-T, Geneva, 2008. — 20 с.
37. МСЭ-T P.1010 (11/2015). Fundamental frequency-based representation of the talking and listening characteristics of a telephone user. ITU-T, Geneva, 2015. — 22 с.
38. ITU-T Recommendations Y.1540 « Internet protocol aspects – Quality of service and network performance. Internet protocol data communication service – IP packet transfer and availability performance parameters». ITU-T, Geneva, December 2019.
39. МСЭ-T G.1000 (11/2001). Quality of service and performance: Aspects and recommendations applicable to the next generation networks. ITU-T, Geneva, 2001. — 14 с.
40. 3GPP TS 22.261. 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Service requirements for the 5G system. URL: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3107> (дата обращения: 23.02.2026)
41. Mahmoudi, M. SDN-DVFS: an enhanced QoS-aware load-balancing method in software defined networks / M. Mahmoudi, A. Avokh, B. Barekatin // Cluster Computing 25(1). Pp. 1237-1262. 2022. DOI: 10.1007/s10586-021-03522-x
42. Bharanidharan, C. An Enhanced Framework for Traffic Load Balancing and QoS Provisioning in SDN / C. Bharanidharan, S. Indira Gandhi, R. D. Devapriya // Wireless Personal Communications 121(4). Pp. 3451-3472. 2021. DOI: 10.1007/s11277-021-08886-2

43. Majdoub, M. QoS Aware Cross Domain Routing in SDN: A Comparative Study Between Competitive and Cooperative MARL Approaches / M. Majdoub, A.E. Kamel, H. Youssef // SN Computer Science. 2024. Volume 5, article number 979.
44. Chahlaoui, F. A Taxonomy of Load Balancing Mechanisms in Centralized and Distributed SDN Architectures / F. Chahlaoui, H. Dahmouni // SN Computer Science. 2020. 1(5), article number 268. DOI: 10.1007/s42979-020-00288-8
45. Babbar, H. Load Balancing Algorithm on the Immense Scale of Internet of Things in SDN for Smart Cities / H. Babbar, S. Rani, D. Gupta, H. M. Aljahdali, A. Singh, F. Al-Turjman // Sustainability. 2021. 13, 9587. DOI: 10.3390/su13179587
46. Malbasic, T. Hybrid SDN Networks: A Multi parameter Server Load Balancing Scheme / T. Malbasic, P.D. Bojović, Z. Bojovic, J. Šuh, D. Vujošević. // Journal of Network and Systems Management. 2022. Volume 30, article number 30. DOI: 10.1007/s10922-022-09642-y
47. Vellela, S. S. An Adaptive Load Balancing Technique for Multi SDN Controllers / S. S. Vellela // Proceedings of the International Conference on Augmented Intelligence and Sustainable Systems (ICAISS-2022). IEEE Xplore: CFP22CB2-ART. 2022. 1403–1408. Trichy, India. DOI: 10.1109/ICAISS55157.2022.10010881
48. Rostami, M. An overview of QoS aware load balancing techniques in SDN based IoT networks / M. Rostami, S. Goli. // Journal of Cloud Computing 2024. 13(1). DOI: 10.1186/s13677-024-00651-7
49. Mezher, L. S. Load Balancing Based Intelligent Software Defined Networking (SDN) Controller / L. S. Mezher // Journal of Information Systems Engineering & Management. 2024. 10(1s):57-68. DOI: 10.52783/jisem.v10i1s.101.
50. Baniya, P. Dynamic Load Balancing Schemes for Software Defined Networking (SDN) / P. Baniya, P. Nand, B. Bhushan // In book: Proceedings of Second International Conference on Intelligent System. 2024. pp. 493-506. DOI: 10.1007/978-981-99-8976-8_41.
51. Gopal, P. K. A hybrid multi-constraint lagrangian relaxation based aggregated cost based segment routing in QoS aware software defined networks / K.P.

Gopal, M. Sambath // Vietnam Journal of Science and Technology. 2025. Vol. 63 No. 3. DOI: <https://doi.org/10.15625/2525-2518/19667>.

52. Sandanasamy, A. Hybrid Proximal Policy Optimization based Adaptive Link Load Balancing with INT in Secure SDN-IoT Networks / A. Sandanasamy, P J. Charles // Indian Journal of Science and Technology. 2025. doi.org/10.17485/IJST/v18i24.1007. EDN: KEZCMA.

53. Kumar, S. EFLB-IIoT: Enhanced Flow Control and Load Balancing Approach for SDN-Enabled Industrial Internet-of-Things / S. Kumar, A. Malik // International Journal of Communication Systems. 2025. doi.org/10.1002/dac.70043. EDN: HEEJXN.

54. Лосано-Риск Х.Э. Качество обслуживания в программно-определяемых сетях для научных приложений: возможности и проблемы / Х.Э. Лосано-Риск, Р. Ривера-Родригес, Х.И. Ньето-Иполито, С. Вильярреаль-Рейес, А. Галавис-Москеда, М. Васкес-Брисеньо // Труды ИСП РАН. — 2021. — Т. 33. № 1. — С. 111–122. DOI:10.15514/ISPRAS-2021-33(1)-8.

55. Дмитриева, Ю.С. Сравнительный анализ методов управления сетевыми ресурсами в сетях SDN / Ю.С. Дмитриева // Труды учебных заведений связи. — 2022. — Т. 8. № 1. — С. 73–83. DOI:10.31854/1813-324X-2022-8-1-73-83.

56. Кучерявый А.Е. Исследование моделей балансировки нагрузки в программно-конфигурируемых сетях / С. Мухизи, А.С. Мутхана, Р.В. Киричек, А.Е. Кучерявый // Электросвязь. — 2019. — № 1. — С. 23–29. EDN: YTHCMX.

57. Садов О.Л. Исследование способов управления QoS в программно-конфигурируемых сетях с помощью контроллера Ryu / О.Л. Садов, Н.В. Тивиков // Современные проблемы науки и образования. — 2015. — № 1 (часть 2). — С. 104-109. <https://science-education.ru/ru/article/view?id=20104>. EDN: TXUWMD

58. Лемешко А.В. Разработка и исследование потоковой модели адаптивной маршрутизации в программно-конфигурируемых сетях с балансировкой нагрузки / А.В. Лемешко, Т.В. Вавенко // Доклады ТУСУРа. — 2013. — № 3 (29). — С. 100–108. EDN: RULWEF

59. Дмитриева Ю.С. Исследовательские аспекты в программно-конфигурируемых сетях / Ю.С. Дмитриева, В.С. Елагин // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2022): XI Международная научно-техническая и научно-методическая конференция, Санкт-Петербург, 15–16 февраля 2022 года. Том 1. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2022. – С. 407-413.

60. Дмитриева Ю.С. Моделирование сетевого ресурса в программно-конфигурируемых сетях / Ю.С. Дмитриева, В.С. Елагин // Вестник связи. — 2020. — № 6. — С. 35-40.

61. Дмитриева Ю.С. Подходы к моделированию ресурсов SDN / Ю.С. Дмитриева, В. С. Елагин // Актуальные проблемы инфотелекоммуникаций в науке и образовании: сборник научных статей: в 4х томах, Санкт-Петербург, 24–25 февраля 2021 года / Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича. Том 4. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2021. – С. 305-309.

62. Дмитриева Ю.С. Управление сетевыми ресурсами на основе намерений / Ю.С. Дмитриева // Вестник связи. — 2022. № 4. — С. 20-26.
EDN: PWEMWS

63. Дмитриева Ю.С. Параметры в SDN для прогнозирования сетевых событий / Ю.С. Дмитриева, В. С. Елагин // Актуальные проблемы инфотелекоммуникаций в науке и образовании: сборник научных статей: в 4х томах, Санкт-Петербург, 28 февраля – 1 марта 2023 года / Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича. Том 1. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2023. – С. 399-403.

64. Дмитриева Ю.С. Анализ методов идентификации трафика для управления ресурсами в SDN / Ю.С. Дмитриева, Д.В. Окунева, В.С. Елагин // Труды

учебных заведений связи. — 2023. — Т. 9, № 6. — С. 42-57. DOI: 10.31854/1813-324X-2023-9-6-42-57 EDN: VARVAY

65. Hidayat T. Load Balancing Network by using Round Robin Algorithm: A Systematic Literature Review / T. Hidayat, Y. Azzery, R. Mahardiko // *Jurnal Online Informatika*. 2020. Volume 4 No. 2. 2019: 85-89. DOI: 10.15575/join.v4i2.446. EDN: IKICMF

66. Shashidhara B. Vyakaranal. Weighted Round-Robin Load Balancing Algorithm for Software-Defined Network / S. B. Vyakaranal, J. G. Naragund // *Springer Nature*. 2019. Pp. 375–387. DOI: 10.1007/978-981-13-5802-9_35

67. Sinha, G. Enhanced Weighted Round Robin Algorithm to Balance the Load for Effective Utilization of Resource in Cloud Environment / G. Sinha, Deepak K. Sinha // *EAI Endorsed Transactions on Cloud Systems*. 2020. Volume 6. Issue 18. <http://dx.doi.org/10.4108/eai.7-9-2020.166284>

68. Sree Priya, S. Enhanced Weighted Round Robin: A New Paradigm in Cloud Load Balancing / S. Sree Priya, T. Rajendran // *Indian Journal of Science and Technology*. 2025. 18(15): 1220-1228. <https://doi.org/10.17485/IJST/v18i15.3976>

69. Данешманд Б. Исследование и обзор механизмов балансировки нагрузки на основе SDN в 5G/IMT-2020 / Б. Данешманд, Л.А. Ту // *Вестник Воронежского государственного технического университета*. — 2022. — Т. 18. № 1. — С. 102-111.

70. Owusu E., Kwame Agyemang-Prempeh Agyekum, Benneh M., Ayorna P., Justice Owusu Agyemang, George Nii Martey Colley. A transformer-based deep Q learning approach for dynamic load balancing in software-defined networks. <https://arxiv.org/abs/2501.12829>. DOI: 10.48550/arXiv.2501.12829.

71. Open Networkin Foundation, OpenFlow Switch Specification version 1.0.0, ONF TS-001, Grudzień. 2009.

72. Open Networkin Foundation, OpenFlow Switch Specification version 1.5.1, ONF TS-025. 2011.

73. Шалагинов А. SDN-контроллер. / А. Шалагинов // Сетевые технологии [электронный ресурс]. — 2021. — № 16. — Режим доступа: <https://shalaginov.com/2021/08/19/software-defined-switch/>
74. Gelberger, A. Performance Analysis of Software-Defined Networking (SDN) / A. Gelberger, N. Yemini and R. Giladi // 2013 IEEE 21st International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems. 2013. — P. 389-393. DOI: 10.1109/MASCOTS.2013.58
75. Azodolmolky S. Software Defined Networking with OpenFlow / S. Azodolmolky. — Birmingham: Published by Packt Publishing, 2013. — 153 p.
76. Open Networkin Foundation OpenFlow Management and Configuration Protocol (OF-Config 1.1.1). 2013. [Электронный ресурс] / Режим доступа: <https://www.opennetworking.org/wp-content/uploads/2013/02/of-config-1-1-1.pdf>.
77. Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., & Walker, D. P4: Programming Protocol-Independent Packet Processors. ACM SIGCOMM Computer Communication Review, 44(3). 2014. pp. 87–95. <https://doi.org/10.1145/2656877.2656890>
78. RFC 6374: Packet Loss and Delay Measurement for MPLS Networks / D. Frost, S. Bryant // Internet Engineering Task Force (IETF). — 2011. — September. — 52 p. — (Request for Comments; 6374). — URL: <https://www.rfc-editor.org/rfc/pdf/rfc6374.txt.pdf> (дата обращения: 21.02.2026)
79. RFC 5481: Packet Delay Variation Applicability Statement / A. Morton, G. Ramachandran, G. Maguluri. — IETF, March 2009. — 27 p. — URL: <https://www.rfc-editor.org/info/rfc5481> (дата обращения: 21.02.2026)
80. Дмитриева Ю.С. Метод комплексной оценки параметров QoS в сетях SDN / Ю.С. Дмитриева // Вестник связи. — 2026. — № 4. — С. 14-21.
81. СП 519.1325800.2023. Свод правил. Сети связи. Правила проектирования (утв. Приказом Минстроя России от 17.03.2023 № 200/пр).
82. Дмитриева Ю.С. Разработка модели балансировки нагрузки в мультиконтроллерной сети SDN с учетом параметров QoS / Ю.С. Дмитриева //

Электросвязь. — 2026. — № 02. — С. 55-61.
DOI: 10.34832/ELSV.2026.76.2.007 EDN: PCPWAH

83. Галич С.В. Исследование производительности ПКС-контроллера OpenDaylight на сетях разных масштабов / С.В. Галич, М.С. Деогенов, В.Г. Карташевский, А.О. Пасюк, Е.С. Семенов // Известия Южного федерального университета. Технические науки. — 2016. — № 9 (182). — С. 121–133. DOI: 10.18522/2311-3103-2016-9-121133. EDN: XIKHOR

84. Рекомендация МСЭ-Т G.114: Односторонняя передача. 2003.
<https://www.itu.int/rec/T-REC-G.114/en>

85. Дмитриева Ю.С. Мультиконтроллерная модель системы балансировки нагрузки при миграции виртуальных систем в SDN / Ю.С. Дмитриева // Вестник связи. — 2024. — № 7. — С. 16-24. EDN: BVRTSH

86. Дмитриева Ю.С. Разработка и апробация методики балансировки трафика в сетях SDN на базе комбинированного параметра QoS / Ю.С. Дмитриева, В.С. Елагин // Труды учебных заведений связи. 2026. — Т. 12. № 2. — С. 36-44. DOI: 10.31854/1813-324X-2026-12-2-36-44 EDN: FOVZUL

87. Свидетельство о государственной регистрации программы для ЭВМ № RU 2025662025. Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования / Дмитриева Ю.С., Елагин В.С., Рыбин И.П., Сербин А.А. Клиентская панель управления, настройки и визуализации результатов экспериментов имитационного моделирования. Опубл. 16.05.2025. EDN:NNHGGO

88. Патент на промышленный образец от 27.02.2026 RU 151708 S. Графический интерфейс пользователя для отображения результатов исследования / Дмитриева Ю. С., Елагин В. С., Рыбин И. П., Сербин А. А.

89. Свидетельство о государственной регистрации программы для ЭВМ № RU 2025662026. Серверная система создания, управления и балансировки нагрузки / Дмитриева Ю.С., Елагин В.С., Рыбин И.П., Сербин А.А. Опубл. 16.05.2025.

ПРИЛОЖЕНИЯ

Акт внедрения ООО «Протей ТЛ»



ООО «ПРОТЕЙ ТЛ»
197342, г. Санкт-Петербург, вн.тер.г. муниципальный
округ Ланское, ул. Белоостровская, д. 6, литер А,
помещ.30-Н, офис 615 Тел.: +7 (812) 401-63-25
<https://tl.protei.ru/>, e-mail: sales@protei.ru

УТВЕРЖДАЮ
Генеральный директор
ООО «ПРОТЕЙ ТЛ»
А.В. Александров



АКТ

о внедрении результатов диссертационной работы

Юлии Сергеевны Дмитриевой

«Разработка модели и методики распределения трафика в мультиконтроллерных сетях SDN с учётом мультипараметрических требований качества обслуживания»

Настоящим актом подтверждаем, что научные результаты диссертационной работы Юлии Сергеевны Дмитриевой «Разработка модели и методики распределения трафика в мультиконтроллерных сетях SDN с учётом мультипараметрических требований качества обслуживания», представленной на соискание ученой степени кандидата технических наук, внедрены ООО «ПРОТЕЙ ТЛ» при разработке Методики планирования корпоративных сетей передачи данных.

В частности, были применены следующие новые научные результаты из диссертации Ю.С. Дмитриевой:

- Метод комплексной оценки параметров QoS в сетях SDN на базе параметров QoS, основанный на сведении разнородных параметров QoS контроллеров и коммутаторов к единому интегральному показателю с использованием нормализующих весовых коэффициентов;
- Модель балансировки нагрузки в мультиконтроллерной архитектуре сети SDN, которая позволяет за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей QoS по сравнению с традиционной нагрузочной балансировкой.

Технический директор, к.т.н.
ООО «ПРОТЕЙ ТЛ»


Н.А. Куликов



Общество с ограниченной ответственностью
«Научно-Технический Центр АРГУС»
Почтовый адрес: 197198, г. Санкт-Петербург,
ул. Красного Курсанта, д. 25, лит. Ж
Юридический адрес: 191028,
г. Санкт-Петербург, ул. Пестеля, д. 7
ИНН 7841364714
КПП 784101001
ОГРН 1077847466597
Тел.: +7-812-333-36-60
E-mail: office@argustelecom.ru
URL: www.argustelecom.ru

Утверждаю

Генеральный директор

ООО «НТЦ АРГУС»

доктор технических наук

А.Б. Гольдштейн



АКТ

о внедрении результатов диссертационной работы Дмитриевой Юлии Сергеевны на тему «РАЗРАБОТКА МОДЕЛИ И МЕТОДИКИ РАСПРЕДЕЛЕНИЯ ТРАФИКА В МУЛЬТИКОНТРОЛЛЕРНЫХ СЕТЯХ SDN С УЧЁТОМ МУЛЬТИПАРАМЕТРИЧЕСКИХ ТРЕБОВАНИЙ КАЧЕСТВА ОБСЛУЖИВАНИЯ», представленной на соискание ученой степени кандидата технических наук по специальности 2.2.15– Системы, сети и устройства телекоммуникаций

Настоящим актом подтверждаем, что научные результаты диссертационной работы Дмитриевой Юлии Сергеевны «РАЗРАБОТКА МОДЕЛИ И МЕТОДИКИ РАСПРЕДЕЛЕНИЯ ТРАФИКА В МУЛЬТИКОНТРОЛЛЕРНЫХ СЕТЯХ SDN С УЧЁТОМ МУЛЬТИПАРАМЕТРИЧЕСКИХ ТРЕБОВАНИЙ КАЧЕСТВА ОБСЛУЖИВАНИЯ», представленной на соискание ученой степени кандидата технических наук, внедрены в разработках ООО «НТЦ АРГУС». В частности, в двух модулях учета и управления ресурсами сети SDN были использованы следующие научные результаты из диссертации Дмитриевой Ю.С.:

– Модель балансировки нагрузки в мультиконтроллерной архитектуре сети SDN позволяет за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей QoS по сравнению с традиционной нагрузочной балансировкой — на 22,4 % на уровне контроллеров и на 47,9 % на уровне коммутаторов.

– Методика динамической балансировки трафика в сетях SDN с интеграцией нормализованного комбинированного параметра QoS, обеспечивает распределение нагрузки и улучшает комбинированный показатель качества на 46,0 % по сравнению с традиционной нагрузочной балансировкой, что подтверждено с использованием разработанного и зарегистрированного программного обеспечения имитационного моделирования.

Разработанные в диссертации Дмитриевой Ю.С. модель и методика, позволили повысить эффективность систем эксплуатационного управления ООО “НТЦ АРГУС” при применении на сетях SDN.

Члены комиссии:

Заместитель генерального директора, к.т.н., Феноменов М.А.

Руководитель направления разработки Ковалев А.А.



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

03.04.2026 № 51/54-99
на № _____ от _____

Утверждаю

Проректор по научной работе,
д-р. техн. наук., доцент



А.В. Рабин



Акт

о внедрении научных результатов, полученных

Дмитриевой Юлией Сергеевной

в диссертационной работе «Разработка модели и методики распределения трафика в мультиконтроллерных сетях SDN с учётом мультипараметрических требований качества обслуживания»

Комиссия в составе и.о. декана факультета инфокоммуникационных сетей и систем В.С. Елагина, заведующей кафедрой инфокоммуникационных систем М.А. Маколкиной и заведующей лабораторией кафедры инфокоммуникационных систем Н.А. Лаюшки составила настоящий акт в том, что научные результаты, полученные Дмитриевой Юлией Сергеевной, использованы:

1. При проведении практических занятий и лабораторных работ по курсу «Технологические принципы организации инфокоммуникационных услуг» (Рабочая Программа № 25.05/236-Д, утверждена Первым проректором-проректором по учебной работе А.В. Абиловым 23.06.2025), разделы Программы:

- Глобальная информационная инфраструктура;

- Качество обслуживания и качество восприятия инфокоммуникационных услуг.

2. При проведении практических занятий и лабораторных работ по курсу «Исследовательские аспекты в программно-конфигурируемых сетях» (Рабочая Программа № 25.05/150-Д, утверждена Первым проректором-проректором по учебной работе А.В. Абиловым 23.06.2025), разделы Программы:

- Тенденции развития информационных технологий и ресурсов;
- Методики принятия решений в исследованиях.

В указанных дисциплинах используются следующие новые научные результаты, полученные Дмитриевой Юлией Сергеевной в диссертационной работе:

1. Метод комплексной оценки параметров QoS в сетях SDN на базе параметров QoS, основанный на сведении разнородных параметров QoS контроллеров и коммутаторов к единому интегральному показателю с использованием нормализующих весовых коэффициентов.

2. Модель балансировки нагрузки в мультиконтроллерной архитектуре сети SDN позволяет за счёт нормализации разнородных параметров QoS и их комбинированной оценки обеспечить улучшение показателей QoS по сравнению с традиционной нагрузочной балансировкой - на 22,42 % на уровне контроллеров и на 47,92 % на уровне коммутаторов.

3. Методика динамической балансировки трафика в сетях SDN с интеграцией нормализованного комбинированного параметра QoS, обеспечивает распределение нагрузки и улучшающая комбинированный показатель качества на 46 % по сравнению с традиционными подходами, что подтверждено экспериментальными исследованиями на модельной сети.

Зав. кафедрой ИКС

М.А. Маколкина

Доцент кафедры ИКС

А.А. Гребенщикова

Зав. лабораторией кафедры ИКС

Н.А. Лаюшка