

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

На правах рукописи



ПЕСТИН МАКСИМ СЕРГЕЕВИЧ

**Методы и алгоритмы взаимодействия программного обеспечения узлов
беспроводных децентрализованных сетей передачи данных при решении
задач маршрутизации**

Специальность 2.3.5. Математическое и программное обеспечение
вычислительных систем, комплексов и компьютерных сетей

Диссертация
на соискание ученой степени
кандидата технических наук

Научный руководитель:
кандидат технических наук, доцент
Новиков Александр Сергеевич

Тула – 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1. ОБЗОР И АНАЛИЗ МЕТОДОВ ОБЕСПЕЧЕНИЯ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРИ МАРШРУТИЗАЦИИ ПОТОКОВ ТРАФИКА В ДСПД.....	13
1.1. Задачи маршрутизации в ДСПД.....	13
1.2. Программное обеспечение сетевой маршрутизации узлов ДСПД.....	15
1.3. Анализ и выбор критериев эффективности передачи данных по ДСПД.....	17
1.4. Анализ методов взаимодействия сетевого программного обеспечения при решении задач маршрутизации трафика в ДСПД.....	20
1.5. Взаимодействие абонентов при балансировке трафика в сетях ДСПД.....	36
1.6. Анализ и выбор маршрутных метрик в ДСПД.....	40
1.7. Выводы по главе.....	48
2. МЕТОДЫ И АЛГОРИТМЫ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ УЗЛОВ ПРИ МАРШРУТИЗАЦИИ ТРАФИКА В ДСПД.....	51
2.1. Метод взаимодействия ПОСМ при подключении узла к ДСПД и поддержания стабильного сетевого соединения.....	52
2.1.1. Алгоритм взаимодействия ПОСМ при установлении и дальнейшем поддержании соединения с сетью.....	56
2.2. Метод взаимодействия ПОСМ при построении квазипараллельных маршрутов связи.....	58
2.2.1. Метод оценки маршрутов передачи данных.....	63
2.2.2. Алгоритм взаимодействия ПОСМ при запросе маршрута.....	68
2.2.3. Алгоритм взаимодействия ПОСМ при возврате маршрутов связи.....	71
2.3. Метод взаимодействия ПОСМ при восстановлении маршрутов.....	75
2.3.1. Алгоритм взаимодействия ПОСМ узлов при восстановлении маршрутов.....	79
2.4. Метод балансировки трафика по квазипараллельным маршрутам....	84
2.4.1. Алгоритм коммуникации между ПОСМ при доставке пакетов с данными.....	86
2.5. Выводы по главе.....	89
3. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СЕТЕВЫХ УЗЛОВ ДЛЯ УМЕНЬШЕНИЯ СКВОЗНОЙ ЗАДЕРЖКИ И ПОВЫШЕНИЯ НАДЕЖНОСТИ ПЕРЕДАЧИ ДАННЫХ.....	91

3.1. Программное средство для моделирования взаимодействия ПОСМ узлов ДСПД	91
3.2. Архитектура сетевого программного обеспечения отдельных абонентов ДСПД с поддержкой квазипараллельных маршрутов передачи данных и балансировки трафика в них	100
3.3. Протокол маршрутизации на основе разработанных методов и алгоритмов взаимодействия ПОСМ узлов ДСПД	102
3.4. Программная реализация протокола маршрутизации между абонентами ДСПД.....	111
3.5. Выводы по главе.....	118
4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ РАЗРАБОТАННЫХ МЕТОДОВ И АЛГОРИТМОВ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПО ПРИ МАРШРУТИЗАЦИИ ТРАФИКА	120
4.1. Экспериментальная проверка корректности функционирования методов и алгоритмов взаимодействия сетевого ПО	121
4.2. Применение и настройка алгоритмов машинного обучения для оценки качества маршрутов связи	127
4.3. Экспериментальная проверка разработанных методов и алгоритмов взаимодействия ПОСМ при формировании и восстановлении квазипараллельных маршрутов	131
4.4. Экспериментальная проверка алгоритма взаимодействия ПОСМ при восстановлении маршрутов.....	139
4.5. Экспериментальная проверка влияния метода оценки маршрутов передачи данных на объём служебного трафика.....	144
4.6. Выводы по главе.....	145
ЗАКЛЮЧЕНИЕ	148
СПИСОК ЛИТЕРАТУРЫ.....	151
ПРИЛОЖЕНИЯ.....	169
Приложение А. Свидетельства о государственной регистрации программ для ЭВМ	169
Приложение Б. Акты об использовании результатов диссертационной работы.....	173

ВВЕДЕНИЕ

Актуальность. Беспроводные децентрализованные сети передачи данных (ДСПД) являются технологией, которая позволяет обеспечивать обмен данными между мобильными узлами в условиях недоступности централизованной инфраструктуры, например, когда сеть формируется в удалении от крупных населённых пунктов или инфраструктура повреждена в результате природных или техногенных катастроф. ДСПД имеют динамически изменяющуюся ячеистую топологию и строятся на основе мобильных узлов, которые осуществляют коммуникацию между собой посредством взаимодействия программного обеспечения при функционировании сети и в процессе передачи информации. При этом каждое абонентское устройство должно участвовать в построении маршрутов передачи трафика и формировании топологии сети, являясь как конечной точкой, так и маршрутизатором для других узлов.

Решение задач формирования структуры сегментов сетей, подключения новых узлов к ним, построения и поддержания в актуальном состоянии маршрутов передачи данных, комплексирования маршрутных метрик и балансировки потоков трафика обычно обеспечивается программным обеспечением сетевой маршрутизации (ПОСМ) абонентов ДСПД. С одной стороны, ПОСМ должно обеспечивать функционирование узлов как независимых абонентов, у которых имеются свои специфичные задачи и функции. С другой, используя механизмы межпрограммного взаимодействия, ПОСМ должно осуществлять контроль и управление функционированием сети как единого объекта. Это требует наличия эффективных механизмов взаимодействия между программным обеспечением сетевой маршрутизации отдельных узлов.

Анализ современных исследований показывает, что в настоящее время высокая эффективность передачи трафика обеспечивается лишь в сетях со статичной или квазистатичной топологией. При этом связь часто существенно ухудшается в случае высокой динамики изменения структуры сети. Одним из основных факторов, которые приводят к указанной проблеме, являются недостатки существующих методов взаимодействия ПОСМ. Проактивное

взаимодействие, включающее запланированные периодические отправки оповещений и уведомлений, генерирует большие объёмы служебного трафика и сопровождается низкими темпами обновления маршрутной информации, что значительно снижает пропускную способность сетевых каналов, а также ведет к трудностям в обеспечении связности в ДСПД. В то же время, использование реактивных подходов, в которых взаимодействие инициируется в ответ на событие или запрос, приводит к возникновению существенных задержек при построении первичных маршрутов передачи трафика, а также к трудностям с их восстановлением в случае разрыва каналов. Таким образом, в настоящее время существует проблема повышения эффективности передачи данных в ДСПД, которая может быть решена за счет разработки более совершенных протоколов взаимодействия между программным обеспечением отдельных узлов сети.

В связи с вышеизложенным, разработка и исследование новых методов и алгоритмов взаимодействия программного обеспечения маршрутизации трафика в беспроводных децентрализованных сетях для организации передачи данных между мобильными узлами в условиях отсутствия централизованной инфраструктуры является актуальной **научной задачей**.

Степень разработанности темы исследования. Разработке подходов взаимодействия сетевого программного обеспечения при формировании и обслуживании маршрутов передачи данных в беспроводных децентрализованных сетях посвящены работы отечественных и зарубежных учёных Дугаева Д.А., Романова С.В., Кулакова М.С., Леонова А.В., Шаваша А, Soonki Jo, Tabatabaei S., Subramaniam. K., Singh J. Вопросами распределённой балансировки трафика в ДСПД занимались Васильев Д.С., Кайсина И.А., Periyasamy P., Pal. A., Taha A., Alghamdi S.A., Er-rouidi M, Saleh A. и др. Исследования используемых в ДС маршрутных метрик отражены в работах Махрова С.С., Базенкова Н.И., Pandey, P., Xiaoxia Qi., Venatia S.E. и др.

Анализ известных исследований позволил наметить пути совершенствования методов взаимодействия сетевого программного обеспечения, сформировать объект, предмет, цель и задачи диссертационной работы.

Объектом исследования является программное обеспечение сетевой маршрутизации узлов в беспроводных децентрализованных сетях передачи данных.

Предметом исследования являются методы и алгоритмы взаимодействия программного обеспечения узлов в децентрализованных сетях передачи данных при маршрутизации и балансировке потоков трафика.

Целью диссертационной работы является повышение эффективности передачи данных в беспроводных децентрализованных сетях за счет создания новых методов и алгоритмов организации взаимодействия сетевого программного обеспечения отдельных узлов при решении задач маршрутизации в условиях наличия множества альтернативных путей доставки трафика.

Для достижения поставленной цели сформулированы и решены следующие **задачи исследования**:

1. Разработка методов и алгоритмов взаимодействия программного обеспечения сетевой маршрутизации узлов ДСПД для обнаружения, восстановления маршрутов связи, обеспечивающих повышение эффективности передачи данных по сети, а также разработка метода балансировки потоков данных в ДСПД по квазипараллельным маршрутам, который позволит снизить сквозную задержку доставки сообщений.

2. Разработка метода оценки маршрутов, который будет использоваться при взаимодействии между программным обеспечением сетевой маршрутизации отдельных узлов, при формировании путей передачи информации и позволит снизить объемы служебного трафика между отдельными узлами.

3. Разработка программного средства для исследования взаимодействия программного обеспечения сетевой маршрутизации в ДСПД с высокой динамикой изменения топологии сети.

4. Разработка архитектуры программного обеспечения сетевой маршрутизации, обеспечивающей интеграцию предложенных методов и алгоритмов в рамках единого решения. Создание прототипа программного обеспечения

сетевой маршрутизации на основе протокола связи с использованием программного средства.

5. Экспериментальная проверка и подтверждение эффективности разработанных методов и алгоритмов взаимодействия программного обеспечения сетевой маршрутизации трафика и их сравнение со стандартными технологиями на типовых сценариях функционирования ДСПД.

Исследование обладает следующей **научной новизной**:

1. Предложены методы и алгоритмы взаимодействия программного обеспечения сетевой маршрутизации при построении и восстановлении маршрутов связи в ДСПД, которые отличаются от существующих решений совмещением реактивной и проактивной схем обмена сообщениями, совместным использованием адресных и широковещательных информационных ответов, что позволяет повысить значение коэффициента доставки сетевых пакетов и уменьшить сквозную задержку при их передаче по найденным маршрутам, а также обеспечивает возможность построения и поддержки квазипараллельных путей передачи трафика.

2. Предложен новый метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия программного обеспечения сетевых узлов, отличающийся от стандартных решений возможностью динамической фильтрации и упорядочивания информационных сообщений на промежуточных узлах сети, что позволяет повысить эффективность функционирования ДСПД за счет снижения суммарного объема служебного трафика между программным обеспечением сетевой маршрутизации отдельных абонентов.

3. Предложен метод балансировки потоков данных в ДСПД, позволяющий снизить сквозную задержку доставки сообщений и отличающийся децентрализованным взаимодействием сетевых узлов при распределении пакетов трафика по квазипараллельным маршрутам с учётом значений их динамических метрик, собираемых при информационном обмене между программным обеспечением сетевой маршрутизации.

Практической значимостью обладают следующие результаты:

1. Реализовано программное средство, которое позволяет проводить моделирование процессов функционирования сети в различных сценариях работы и предоставляет возможность исследовать, отлаживать и выполнять оценку предложенных методов и алгоритмов взаимодействия сетевого программного обеспечения узлов ДСПД.

2. Предложена архитектура программного обеспечения сетевой маршрутизации отдельных узлов ДСПД, обеспечивающая интеграцию разработанных методов и алгоритмов взаимодействия ПО в рамках единого решения и их применение на практике.

3. Предложенные в диссертации методы и алгоритмы реализованы в виде протокола связи, на основе которого созданы программные модули для выполнения процедур сетевой маршрутизации, функционирующие в рамках разработанного программного средства.

4. Проведено экспериментальное исследование созданного протокола в различных сценариях функционирования ДСПД, которое подтвердило эффективность предложенных решений, что открывает возможность его применения на практике для быстрого развертывания сетей в условиях низкой пропускной способности и нестабильности каналов связи, вызванных высокой динамикой изменения топологии сети и потребностями в передаче большого объема информации.

Методы и инструменты решения. Для решения поставленных в диссертационной работе задач использовались инструменты имитационного моделирования (OMNET++), градиентный бустинг деревьев принятия решений, фреймворк Qt, языки программирования C++, python3, библиотека CatBoost.

Положения, выносимые на защиту:

1. Методы и алгоритмы взаимодействия программного обеспечения сетевой маршрутизации при построении и восстановлении маршрутов связи в ДСПД, которые отличаются от существующих решений совмещением реактивной и проактивной схем обмена сообщениями, совместным использованием адресных и широковещательных информационных ответов, что

позволяет повысить значение коэффициента доставки сетевых пакетов и уменьшить сквозную задержку при их передаче по найденным маршрутам, а также обеспечивает возможность построения и поддержки квазипараллельных путей передачи трафика.

2. Новый метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия программного обеспечения сетевых узлов, отличающийся от стандартных решений возможностью динамической фильтрации и упорядочивания информационных сообщений на промежуточных узлах сети, что позволяет повысить эффективность функционирования ДСПД за счет снижения суммарного объема служебного трафика между программным обеспечением сетевой маршрутизации отдельных абонентов.

3. Метод балансировки потоков данных в ДСПД, позволяющий снизить сквозную задержку доставки сообщений и отличающийся децентрализованным взаимодействием сетевых узлов при распределении пакетов трафика по квазипараллельным маршрутам с учётом значений их динамических метрик, собираемых при информационном обмене между программным обеспечением сетевой маршрутизации.

Соответствие диссертации паспорту специальности. Диссертация соответствует требованиям специальности 2.3.5 – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей», а именно пункту 3 – Модели, методы, архитектуры, алгоритмы, языки и программные инструменты организации взаимодействия программ и программных систем.

Апробация результатов диссертации. Основные результаты диссертационной работы докладывались и обсуждались на 9th and 10th IEEE Mediterranean Conference of Embedded Computing (MECO) (Будва, Черногория, 2020, 2021, Scopus, WoS), 21th International scientific and technical conference «Automation-2022» (Сочи, 2022, Scopus), XVII и XVIII Международная научно-техническая конференция «Recognition-2023» (Курск, 2021, 2023), Международном конгрессе «Интеллектуальные системы и информационные

технологии (IS&IT-2023)» (Дивноморское, 2023), Всероссийской научно-технической конференции «Интеллектуальные и информационные системы (Интеллект)» (Тула, 2019, 2021), научном семинаре, проводимом ТулГУ, ТГПУ и ОИЯИ (г. Тула, 2022), II-ой межрегиональной научной конференции «Промышленная революция 4.0: Взгляд молодёжи».

Результаты диссертационной работы были включены в отчёт по НИР «Разработка протокола верхнего уровня для организации связи в беспроводных децентрализованных сетях передачи данных», выполняемой в рамках гранта правительства Тульской области в сфере науки и техники (договор ДС/286 от 25.10.2021).

Личный вклад автора заключается в выполнении основного объёма исследований при разработке методов и алгоритмов взаимодействия ПОСМ мобильных узлов для решения задач маршрутизации в ДСПД, создании программного обеспечения для исследования разработанных методов и алгоритмов и проведении их экспериментальной проверки.

Публикации. По теме диссертации опубликована 21 работа, в том числе 9 статей в рецензируемых научных журналах, рекомендованных ВАК для публикации результатов диссертационного исследования (4 – К1, 5 – К2), 3 публикации в изданиях, индексируемых в международных наукометрических базах Scopus и Web of Science, 9 публикаций в журналах и материалах конференций, индексируемых РИНЦ. На разработанное в рамках диссертационной работе программное обеспечение было получено 4 свидетельства о регистрации программы для ЭВМ.

Структура и объём диссертации. Диссертация состоит из введения, четырех глав, заключения, изложенных на 175 страницах печатного текста, списка использованной научной литературы, включающего 139 наименований научных трудов на русском и иностранных языках и 2 приложений, и содержит 67 рисунков, 13 таблиц и 33 формулы.

Во введении дана общая характеристика работы, рассмотрена ее актуальность, обоснованы цели, задачи, научная новизна, представлены основные

практические и теоретические результаты работы, положения, выносимые на защиту, сведения об апробации и публикации результатов, структура и объем диссертации.

В первой главе представлены результаты анализа современных методов, алгоритмов и протоколов взаимодействия сетевого программного обеспечения маршрутизации трафика, предназначенного для обнаружения и обслуживания маршрутов, их оценки и балансировки потоков данных по найденным путям.

Во второй главе разработаны методы и алгоритмы взаимодействия между сетевым программным обеспечением отдельных узлов при решении задач маршрутизации, включая установление узлом соединения с сетью и его дальнейшее поддержание с формированием списка смежных абонентов, процедуры построения маршрутов связи между узлами, восстановления неисправных путей, метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия ПОСМ, метод балансировки потоков трафика в ДСПД.

В третьей главе разработано программное средство для моделирования процессов взаимодействия сетевого программного обеспечения в различных сценариях работы беспроводных ДСПД, предложены архитектура программного обеспечения сетевой маршрутизации отдельных узлов ДСПД и протокол связи на основе описанных методов и алгоритмов, созданы программные модули для выполнения процедур сетевой маршрутизации, функционирующие в рамках разработанного программного средства.

Четвёртая глава посвящена исследованию предложенных в работе методов и алгоритмов взаимодействия сетевого программного обеспечения и экспериментальному подтверждению, что их практическая реализация позволяет обеспечить улучшение эффективности передачи трафика в ДСПД в различных сценариях их функционирования.

В заключении представлены итоги выполненного исследования, приведены рекомендации, и описаны перспективы дальнейших исследований по теме диссертации.

В приложении представлены свидетельства о регистрации разработанных программ, акты внедрения.

1. ОБЗОР И АНАЛИЗ МЕТОДОВ ОБЕСПЕЧЕНИЯ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРИ МАРШРУТИЗАЦИИ ПОТОКОВ ТРАФИКА В ДСПД

1.1. Задачи маршрутизации в ДСПД

Под маршрутизацией в компьютерных сетях понимается процесс определения маршрута (пути) передачи данных от абонента-отправителя (абонента-источника) к абоненту-получателю (целевому абоненту) через промежуточные узлы и последующее перенаправление пакетов трафика по построенному пути [1]. Процесс маршрутизации подразумевает решение нескольких задач:

- установление соединения с другими узлами ДСПД и дальнейшее его поддержание;
- построение маршрутов между узлом-отправителем и узлом-получателем;
- поддержание маршрутов связи в актуальном состоянии при передаче по нему данных;
- передачу пакета с данными от абонента-источника целевому узлу.

1. Задача установления соединения с узлами ДСПД и дальнейшее его поддержание решается при первичном подключении абонента к сети, установления связи с соседними абонентами и поддержания соединения с ними в время функционирования сети [2]. Узлы, с которыми имеются каналы прямой связи, называются соседними. Если два узла имеют каналы прямой связи с третьим, но не между собой, то они являются соседями 2-го уровня. К соседним узлам отсутствует необходимость запрашивать маршруты, и данные можно отправлять напрямую, в ином случае они задействуются при построении маршрутов к удалённым узлам.

2. Если между узлом-источником и узлом назначения отсутствуют актуальные маршруты, и абоненты не являются соседними относительно друг друга, решается задача построения маршрутов между узлом-отправителем и узлом-получателем [3]. Данный процесс включает изучение структуры сети

(наличие узлов и каналов связи между ними, а также определение их динамических характеристик), выбор критериев, в соответствии с которыми будет определяться эффективность передачи данных по маршруту. На основе выбранных критериев рассчитываются метрики для каждого возможного пути. Например, для уменьшения задержек передачи данных может использоваться количество транзитных узлов или длина пути. Из всех возможных маршрутов выбираются те, которые наилучшим образом соответствуют заданным критериям. Процесс построения путей включает обновление правил маршрутизации на одном или нескольких узлах.

3. Задача восстановления и обслуживания маршрутов в компьютерных сетях является критически важной для обеспечения надёжности и эффективности передачи данных [4]. Она включает в себя диагностику состояния маршрутов, обнаружение отказов (обрыв канала связи или выход из строя промежуточных узлов), перестроение маршрута, восстановление связи, обновление правил маршрутизации. После обнаружения отказа производится корректировка маршрута с учётом произошедших изменений в топологии сети. При корректировке пути обновляются таблицы маршрутизации на одном или нескольких сетевых устройствах. Скорректированные маршруты абоненты сети могут вновь их использовать для передачи данных. При этом необходимо постоянно осуществлять мониторинг состояния сети, чтобы своевременно обнаруживать потенциальные проблемы и предотвращать отказы.

4. Задача передачи трафика по сети заключается в обеспечении доставки пакетов данных от источника к получателю по найденным маршрутам [5]. Для этого пользователь или приложение инициируют запрос на передачу данных к целевому абоненту. Данные разбиваются на пакеты определенного размера, каждый из которых содержит заголовок с информацией о маршруте, узле-источнике и узле-получателе. При этом пакеты могут быть переданы как по одному единственному маршруту, так и распределены между несколькими альтернативными. Пакеты передаются через сеть, проходя через цепочки узлов и каналов связи между ними. Каждый узел проверяет заголовок пакета, чтобы

определить, куда его следует направить дальше. На стороне получателя производится сборка пакетов в исходный поток данных при помощи информации в заголовках пакетов. Чтобы гарантировать успешную доставку, получатель отправляет подтверждение абоненту-источнику о получении данных.

Решение приведённых выше задач позволяет обеспечить передачу данных в децентрализованных сетях связи, оптимизируя при этом использование сетевых ресурсов и обеспечивая эффективное соединение между пользователями.

1.2. Программное обеспечение сетевой маршрутизации узлов ДСПД

Программное обеспечение сетевой маршрутизации (ПОСМ, routing software, сетевые службы маршрутизации, сетевые демоны маршрутизации) [6] – это специализированное сетевое программное обеспечение, функционирующее на узлах с несколькими сетевыми интерфейсами, которое обеспечивает решение задач маршрутизации трафика на основе протоколов в сетях с использованием клиент-серверной или децентрализованной архитектуры. Такое ПО функционирует на каждом узле сети и является программными агентами сети [7], которые взаимодействуют между собой для решения задач маршрутизации в сетях со сложными топологиями, в которых между абонентами существуют один или несколько промежуточных маршрутизаторов. ПО маршрутизации трафика функционируют на сетевом уровне в соответствии с моделью OSI. Согласно ей, ПОСМ отвечают на запросы обслуживания от транспортного уровня и направляют запросы обслуживания на канальный уровень [8]. Такие программные сервисы реализуют протокол маршрутизации – набор соглашений, которые определяют правила обмена между аналогичными программами на разных сетевых узлах.

Программное обеспечение сетевой маршрутизации трафика предназначено для обеспечения функционирования узла в части:

– хранения информации о маршрутах передачи трафика до других абонентов, состояние известных абонентов и каналов связи между ними, найденных маршрутах и их характеристиках. На их основе формируются таблицы

или другие формы правил маршрутизации [9], которые используются протоколами сетевого уровня;

– построения и обслуживания маршрутов передачи трафика. Процесс включает себя процедуру, в ходе которой обеспечивается формирование цепочек ретрансляции из абонентов сети, применение которых позволяет передавать трафик между целевыми абонентами. В соответствии с этим формируются или обновляются правила маршрутизации. Обслуживание маршрутов является процессом, который обеспечивает поддержание работоспособности уже найденных путей в течение некоторого времени посредством замены одного или нескольких маршрутизаторов в исходных путях.

Взаимодействие между программным обеспечением сетевой маршрутизации отдельных абонентов обеспечивается за счёт обмена данными, который включает отправку инициирующих сообщений, информационных ответов и уведомлений. Взаимодействие инициируется по запросам или имеет периодический характер, что необходимо для решения общей задачи маршрутизации трафика в ДСПД. Взаимодействие между ПОСМ осуществляется при помощи служебных пакетов, которые могут включать информацию о состоянии абонентов и каналов связи между ними, запросы текущих состояний и уведомления об их изменениях.

Такое программное обеспечение может быть включено в состав ядер операционных систем [10, 11], либо распространяться в виде отдельных пакетов. Примерами последних являются Quagga, XORP, Zebra, BIRD, OpenBGPD [12].

Zebra – программный пакет, который предоставляет сервисы маршрутизации на основе протоколов стека TCP/ IP с поддержкой протоколов маршрутизации, таких как OSPF (Open Shortest Path First), RIP (Routing Information Protocol), BGP (Border Gateway Protocol). Zebra также поддерживает BGP Route Reflector, что позволяет узлу выступать в качестве сервера маршрутизации. Zebra предоставляет механизм многосерверной маршрутизации, предоставляет API для использования сторонними приложениями. Quagga является

пакетом сетевой маршрутизации, основанном на Zebra, предоставляющим реализацию протоколов OSPF (Open Shortest Path First), RIP (Routing Information Protocol), BGP (Border Gateway Protocol) и IS-IS для Unix-подобных платформ, в частности Linux, Solaris, FreeBSD и NetBSD.

XORP (eXtensible Open Router Platform – Расширяемая открытая платформа маршрутизатора) – это пакет программного обеспечения для маршрутизации с открытым исходным кодом, изначально разработанный в международном институте компьютерных наук в Беркли (Калифорния). Он поддерживает протоколы OSPF, BGP, RIP, PIM, IGMP, OLSR. Продукт разработан на основе принципов модульности и расширяемости программного обеспечения и нацелен на демонстрацию стабильности и предоставление требований к функциям для использования в производстве, а также на поддержку сетевых исследований. OpenBGPD — это реализация протокола BGPv4. Она позволяет использовать обычные машины в качестве маршрутизаторов, обменивающихся маршрутами с другими системами, использующими протокол BGP.

BIRD (рекурсивная аббревиатура – BIRD Internet Routing Daemon) – это реализация наборов протоколов с открытым исходным кодом для маршрутизации пакетов в операционных системах типа Unix. BIRD устанавливает несколько таблиц маршрутизации и использует протоколы маршрутизации BGP, RIP и OSPF, а также статически определенные маршруты. В настоящее время BIRD включен во многие дистрибутивы Linux, включая Debian, Ubuntu и Fedora.

1.3. Анализ и выбор критериев эффективности передачи данных по ДСПД

Эффективность маршрутов связи в ДСПД – способность выполнять задачу передачи трафика через цепочку транзитных абонентов с определённым качеством [13, 14, 15] в заданных условиях применения ДСПД. Эффективность маршрутов может быть охарактеризована рядом критериев, которые можно разделить на показатели производительности и надёжности. Производительность может быть измерена временными критериями, которые

оценивают задержку передачи данных по маршрутам сети, а также показатели пропускной способности, показывающие объём переданных данных в единицу времени. К показателям производительности относятся:

- сквозная задержка передачи данных;
- круговая задержка передачи данных;
- запас времени на передачу потока данных;
- мгновенная, средняя, максимальная пропускные способности сети;
- джиттер;
- бёртность.

Сквозная задержка (в англоязычной литературе – end-to-end delay) представляет величину, равную промежутку времени от начала передачи пакетов узлом-источником данных до окончания их приёма абонентом-получателем. Круговая задержка передачи данных или задержка приёма передачи – интервал времени от момента передачи пакета до получения абонентом-отправителем ответной квитанции. В большинстве случаев пакеты передаются в составе более крупных блоков данных, например, фреймов (кадров), потоков. Высокая задержка передачи хотя бы одного пакета непосредственно приводит к высоким задержкам передачи всего фрейма, поэтому часто оценивают задержки передачи фреймов данных.

Запаса времени на передачу потока данных с учётом потерь пакетов $Delay_{Reserve}$ представляющую мультипликативную свёртку коэффициента доставки пакетов по сети (packet delivery ratio, PDR) и разницу между предельным $E2EDG_{max}$ и фактическим временем сквозной задержки доставки пакетов в потоке (end-to-end delay group, E2EDG) [15]:

$$Delay_{Reserve} = PDR * (E2EDG_{max} - E2EDG) \quad (1.1)$$

К преимуществам данного показателя относится учёт потерь пакетов при передаче сообщений по маршрутам, что делает его сочетающим в себе показатели производительности и надёжности.

Пропускная способность сети – объём информации, который может быть передан в единицу времени. При этом различают мгновенную, среднюю

и максимальную пропускную способность. В первом случае измерение пропускной способности производится в отдельно выбранном коротком интервале времени (менее 1 секунды). Такой показатель слабо выражает производительность, так как в работе сети могут быть кратковременные скачки нагрузки и её спада. Средняя пропускная способность маршрута характеризует таковую на протяжении всего времени существования маршрута.

Джиттер представляет изменение задержек передачи между отдельными пакетами, которые отправляются по сети в течении короткого промежутка времени. Джиттер может повышаться ввиду различия в скорости и нагрузки передачи данных по отдельным каналам, наличия помех. В результате повышения джиттера пакеты могут приходить к адресату в произвольном порядке. Критерий показывает разницу в задержках между передачей отдельных порций данных, однако не демонстрирует производительность сети в целом. Бёрстность представляет собой нормированную максимальную битовую скорость. Это метрика позволяет оценивать качество технического оборудования, на основе которого функционирует сеть.

Надежность сети – её способность исправно функционировать в течение продолжительного периода времени. Показателями надёжности сети могут выступать коэффициенты доставки и потери пакетов данных. Коэффициент доставки пакетов – отношение количества успешно принятых сообщений на узлах к общему количеству отправленных сообщений в сеть. Второй коэффициент – противоположен первому. При этом маршруты в ДСПД должны характеризоваться непрерывной готовностью по обеспечению высокого уровня надёжности – время восстановления маршрута не должно превышать от 1 до десятков секунд в зависимости от требований к передаче трафика.

В качестве обобщенного критерия эффективности передачи информации в ДСПД принято решение использовать метрику запаса времени на передачу потока данных с учётом потерь пакетов, так как он одновременно учитывает как производительность, так и надёжность сети.

1.4. Анализ методов взаимодействия сетевого программного обеспечения при решении задач маршрутизации трафика в ДСПД

Маршрутизация трафика в беспроводных децентрализованных сетях связи является распределённой [16]. Каждый узел хранит собственную базу данных с информацией о маршрутах, которая накапливается и обновляется за счёт обмена информацией между абонентами. При этом каждый узел обладает собственным программным обеспечением сетевой маршрутизации трафика [17], которое должно позволять формировать структуру сети, обеспечивать построение, обслуживание, оценку маршрутов связи и балансировку трафика по ним для обеспечения высокой эффективности передачи данных по сети. Методы и алгоритмы взаимодействия программных ПОСМ являются частью протоколов маршрутизации сетей ДСПД.

Среди методов взаимодействия между программными компонентами абонентов ДСПД для маршрутизации трафика выделяют проактивные, реактивные и гибридные.

1.4.1. Проактивные методы взаимодействия

Проактивное (табличное) взаимодействие между ПО в сетях ДСПД предполагает, что абоненты производят периодический обмен служебными сообщениями с информацией о текущей топологии сети, и на основании получаемых данных от других узлов формируют свои таблицы маршрутизации. Табличное взаимодействие между абонентами ДСПД является адаптацией методов и алгоритмов классических проводных сетей под новые условия функционирования. Наиболее распространёнными протоколами маршрутизации, реализующие проактивное взаимодействие, являются DSDV и OLSR.

Протокол маршрутизации DSDV (Destination-Sequenced Distance-Vector Routing – маршрутизация на основе вектора расстояния до места назначения) основан на алгоритме Беллмана-Форда поиска кратчайшего пути на графе [18]. По алгоритму производится сбор данных для формирования таблицы маршрутизации (рис. 1.1), каждая запись которой содержит целевой узел, следующий промежуточный узел к целевому, расстояние до целевого узла

(количество транзитных абонентов), порядковый номер. Периодически абоненты рассылают уведомления об имеющихся изменениях в версии топологии из своей базы данных, при значительных изменениях производится рассылка таблицы маршрутизации целиком. Для поддержания согласованности таблицы маршрутам присваиваются новые порядковые номера, и информация об изменениях в топологии периодически рассылается по сети. Возможен обмен как полными дампами данных о таблицах маршрутизации, так и их изменившимися частями. Во втором случае происходит существенная экономия трафика. Записи в таблице маршрутизации содержат адреса абонента-назначения, первый транзитный узел на пути к нему, расстояние в хопах.

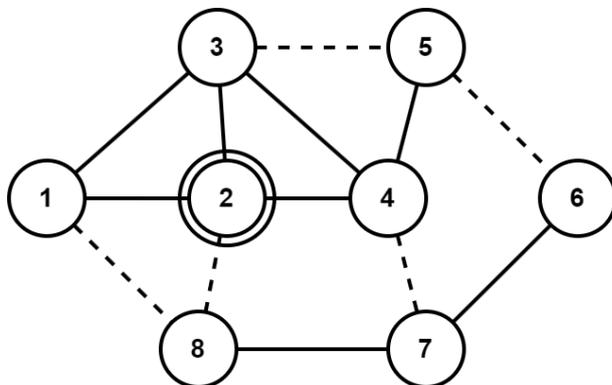


Таблица маршрутизации для абонента №2

Целевой абонент	Следующий абонент	Длина маршрута	№ пос.-ти
1	1	1	123
2	0	0	516
3	3	1	212
4	4	1	168
5	4	2	372
8	1	∞	432

Рисунок 1.1 – Таблица маршрутизации для протокола DSDV

WRP (Wireless routing protocol – протокол беспроводной маршрутизации) схож с DSDV, при этом каждый узел имеет информацию о полной топологии сети [19]. Как следствие, объем трафика выше, чем в DSDV, и эффективность межсетевого взаимодействия между программным обеспечением абонентов снижается. Протокол Babel [20] является развитием DSDV, в котором для восстановления маршрута связи до целевого абонента производится реактивный запрос восстановления маршрута к ПО маршрутизации целевого абонента (см. п. 1.1.2).

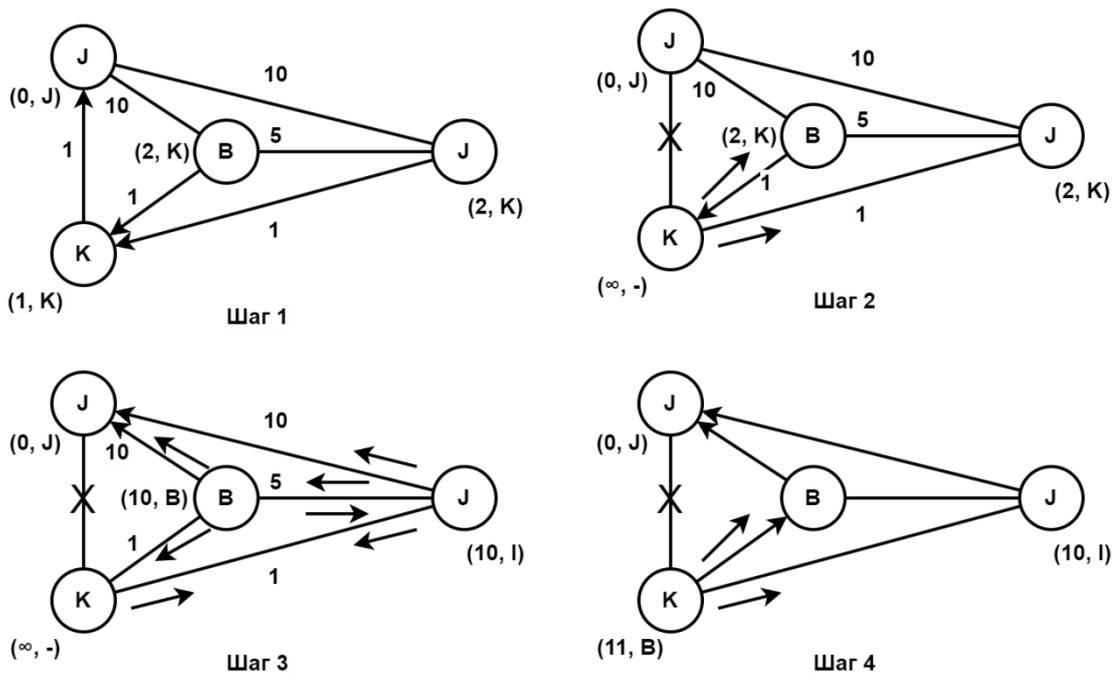


Рисунок 1.2 – Пример взаимодействия абонентов сети при восстановлении маршрутов связи в WRP [21]

OLSR [22] (Optimized Link-State Routing – оптимизация маршрутов на основе состояния связей) – проактивный протокол маршрутизации на основе состояния канала, который использует широковещательные сообщения приветствия (hello) и контроля топологии (topology control, TC) для получения информации о связях между узлами. Данная информация должна быть известна каждому узлу в сети. Обнаружение изменений в топологии сети производится при помощи принимаемых от соседей сообщений hello. hello содержит адрес узла-отправителя, список доступных соседей с меткой симметричности соединений. Каждый узел хранит информацию о соседях 1-го и 2-го уровня. Схема потоков данных в протоколе представлена на рис. 1.3, схема обмена служебными пакетами – на рис. 1.4. Кроме того, OLSR поддерживает многопутевую маршрутизацию – одновременно может быть обнаружено несколько непересекающихся маршрутов связи.

В настоящее время OLSR и его модификации являются наиболее удачными проактивными схемами взаимодействия ПО сетевой маршрутизации абонентов ДСПД. Например, основанные на OLSR протоколы адаптированы

для сниженного энергопотребления сетевых узлов [23], снижения управляющего трафика [24].

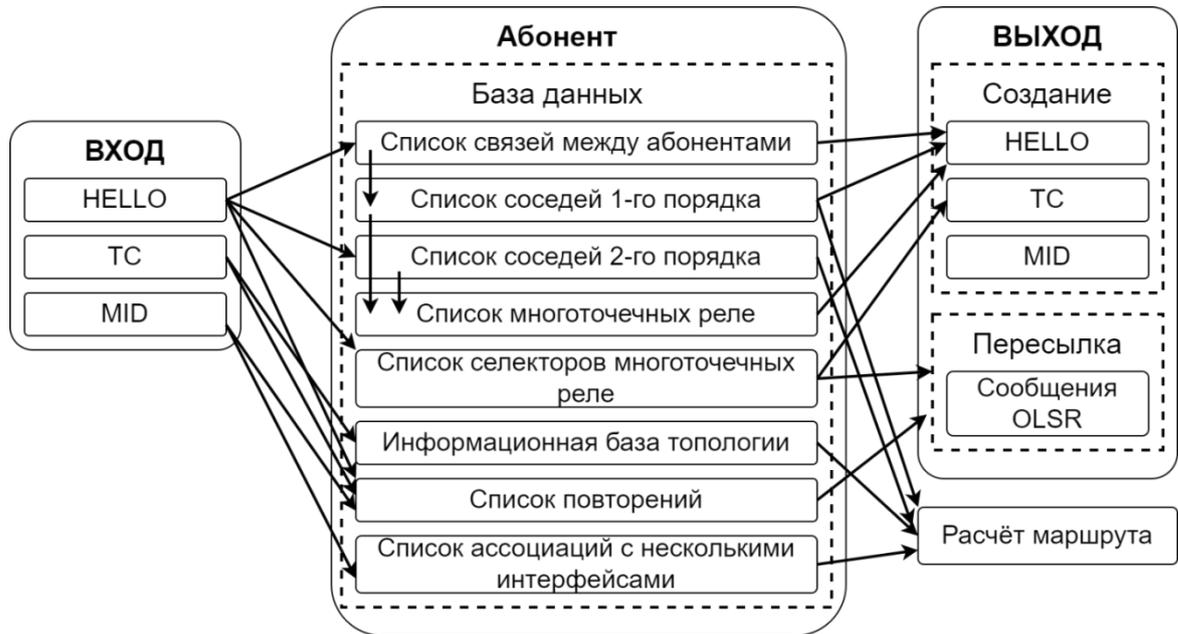


Рисунок 1.3 – Схема потоков данных протокола OLSR

Для повышения качества взаимодействия абонентов могут быть применены метаэвристические методы. В работе [25] для решения задач построения и восстановления маршрутов связи сетевые сервисы используют оценку времени жизни каналов связи на основе алгоритма роя частиц. Абоненты взаимодействуют для определения показателей мобильности, расстояние между собой, интенсивность потребления энергией при обработке и передачи данных. На основе показателей производится оценка времени жизни каналов связи. В случае ухудшения характеристик существующих маршрутов они перестраиваются с включением в них каналов и узлов с более высокими значениями динамических характеристик.

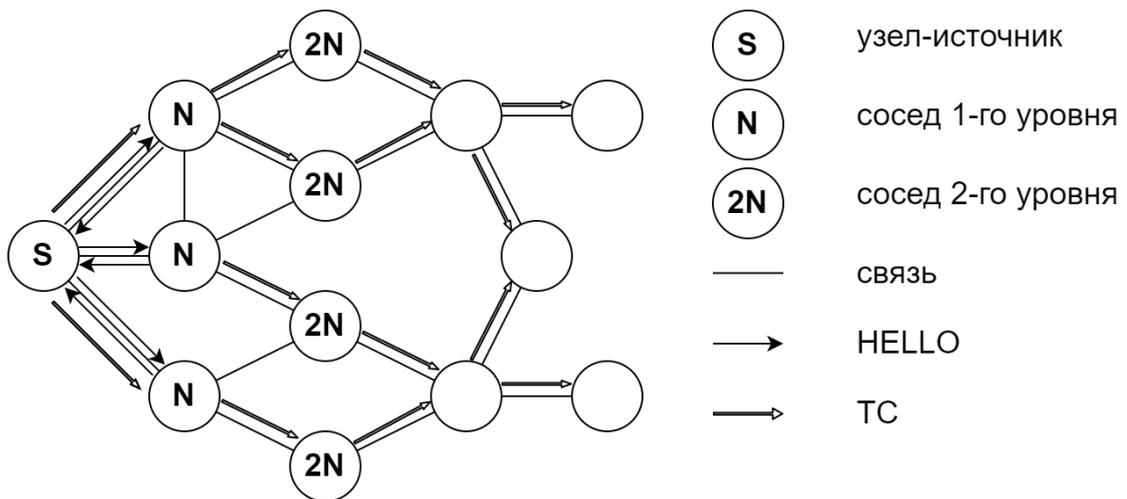


Рисунок 1.4 – Схема передачи служебных пакетов

Необходимость в постоянном обновлении таблиц маршрутизации влечёт за собой высокий объём служебного трафика, способствует расходу зарядов батарей и снижению пропускной способности даже в условиях простаивания сети (когда полезные данные не передаются). Во многих случаях для обновления неисправных маршрутов или обнаружения новых требуется несколько итераций алгоритмов поиска. Данные об актуальности маршрутов связи достаточно инертные. Эти недостатки ограничивают применение табличных протоколов маршрутизации в высокодинамичных сетях передачи данных.

1.4.2. Реактивные методы взаимодействия

Реактивная (или «по запросу») маршрутизация между абонентами ДСПД предполагает, что взаимодействие ПО сетевой маршрутизации при построении и восстановлении маршрутов производится только при необходимости. В режиме простоя сети или при отсутствии необходимости передачи данных между абонентами, процессы построения и обновления маршрутов производиться не будут. Классическими протоколами, использующие реактивное взаимодействие между сетевым программным обеспечением, являются AODV и DSR.

Один из самых известных протоколов реактивной маршрутизации – AODV [26] (Ad-hoc On-Demand Distance Vector – маршрутизация по требованию по вектору расстояний в самоорганизующихся сетях). Согласно

алгоритму его работы, если актуальный маршрут к узлу назначения отсутствует в базе данных ПОСМ, она производит рассылку сообщений RREQ соседним абонентам. Они ретранслируются промежуточными узлами, пока запросы не достигают узла-назначения. Целевой абонент отправляет ответ RREP. При прохождении RREQ через транзитные сетевые устройства абонентами производится запоминание информации об узлах, с которых поступил запрос. Собранная информация используется для построения обратного пути. При выпадении абонентов из маршрута транзитные узлы уведомляют об этом исходный узел. Схема маршрутизации представлена на рис. 1.5.

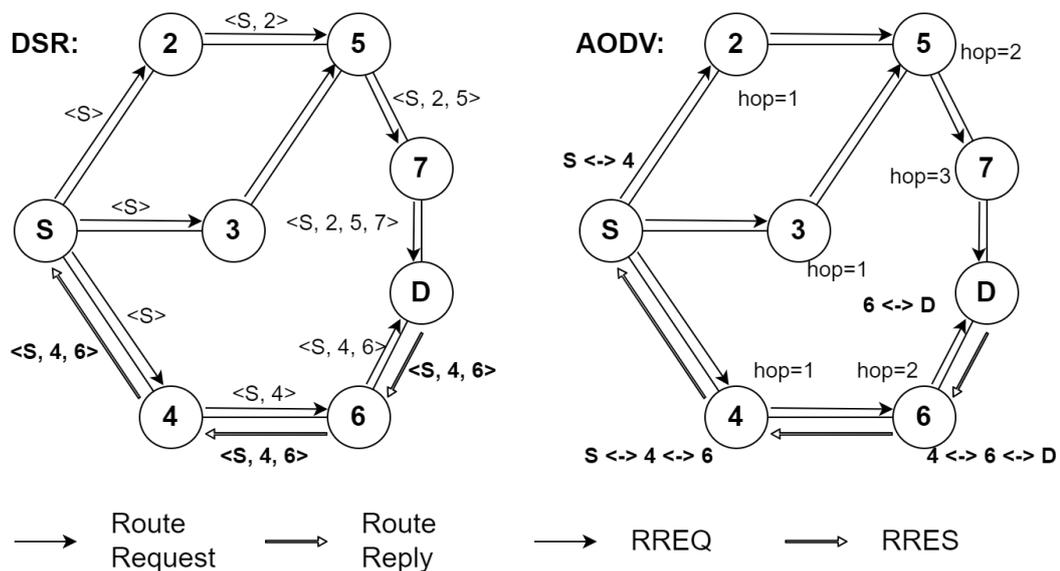


Рисунок 1.5 – Схема взаимодействия абонентов при построении маршрута связи в протоколах DSR и AODV

AODV является базовой технологией для других реактивных протоколов связи. Например, протокол DYMO (AODV 2.0) [27] является улучшенной версией AODV, в которой ПОСМ транзитных абонентов могут использовать восходящие части длинных маршрутов для передачи собственных сообщений. Протокол AOMDV [28] является многопутевой версией протокола AODV и позволяет обнаруживать сразу несколько непересекающихся путей к целевому узлу. Модификации AODV – LBAOMDV [29], EQ-AODV [30], EE-AODV [31], EA_AODV [32] повышают энергоэффективность сети за счёт использования метрик на основе данных о расходе и остаточных значениях электроэнергии.

Они предполагают снижение уровня взаимодействия с абонентами, у которых низкий уровень остаточной энергии.

Целый ряд исследований посвящено обслуживанию маршрутов связи для протокола AODV. Авторами работы [33] предлагается протокол ELRAODV, в котором возможно локальное восстановление маршрута. В отличие от базового протокола, RREP содержит два предыдущих узла в маршруте. Кроме того, вводится пакет «NHello» как альтернатива «Hello», который содержит для абонента-отправителя список соседей. Поэтому, в случае обрыва канала связи, имеется возможность оперативно восстановить маршрут, если между абонентами, потерявшими связь, имеется общий сосед.

В статье [34] предлагается подход к восстановлению маршрутов, называемый make-before-break, для протокола AODV (RSEA-AODV). ПОСМ абонентов, взаимодействия между собой, производят мониторинг состояния узлов сети – остаточный уровень энергии и размер входящих очередей пакетов. В случае их критичных показателей производится запрос HLP к ПОСМ соседних абонентов о наличии актуальных маршрутов к целевым узлам. При положительном ответе одного из ПОСМ производится стыковка нисходящей части пути с актуальным маршрутом соседнего узла. Если такой маршрут отсутствует, ПОСМ отправляет исходному абонентскому устройству уведомление RCR о необходимости повторного поиска маршрута для обнаружения данных. Все промежуточные абоненты, принимая RCR, сбрасывают исходный маршрут в своих таблицах маршрутизации.

Авторы работы [35] предлагают определять в процессе функционирования работы сети надёжные узлы с помощью механизма RAF (the reliability antecedent packet forwarding). Эти узлы могут быть использованы для надёжной передачи данных. Во время трансляции трафика по сети производится отслеживание на предмет того, какие абоненты чаще других теряют данные. Эти абоненты исключаются из маршрутов, при этом в маршрут включаются надёжные узлы. Однако метод не гарантирует, что вблизи абонента с низкими показателями динамических характеристик будет найден надёжный узел.

Авторы публикации [36] предлагают концепцию резервного узла для восстановления маршрута связи. ПОСМ хранит информацию об маршрутах соседних узлов. Они прослушивают передачу данных по маршруту на соседнем узле, а также абонентах, предшествующих и последующих соседнему узлу. Если от транзитных абонентских устройств не поступает сообщений подтверждения передачи данных – его функцию в маршруте берёт резервный узел. Схема взаимодействия агентов маршрутизации при обнаружении и задействовании резервных узлов представлены на рис. 1.6. Недостатком работы является то, что рассматриваются случаи, когда узел источник, транзитный абонент и узел назначения являются соседями резервного узла. Кроме того, метод не позволяет превентивно выявлять разрывы в маршрутах.

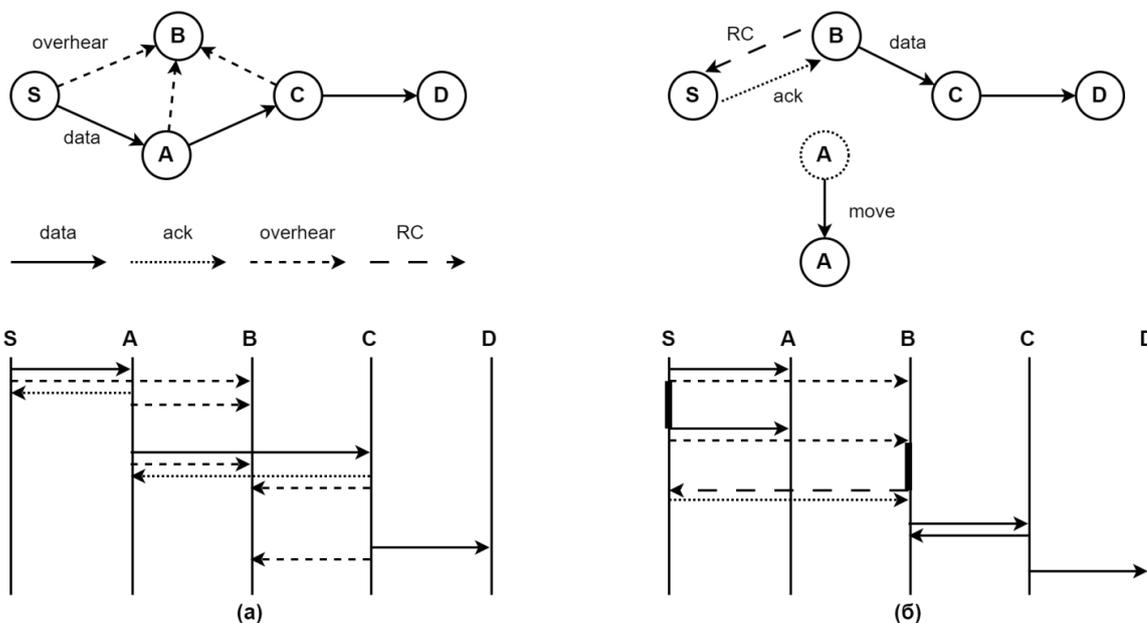


Рисунок 1.6 – Схема взаимодействия абонентов при (а) обнаружении и использовании (б) резервных абонентов в протоколе IBR-AODV

Схема поиска маршрута по протоколу DSR [37] (Dynamic Source Routing – динамическая маршрутизация от источника) похожа на AODV, однако главным отличием является накопление адресов транзитных абонентов в теле пакета. По сформированному в пакете маршруту производится его возврат узлу-инициатору, ПОСМ который сохраняет его в кэше. Схема маршрутизации протокола DSR представлена на рис. 1.5. Недостатком данного протокола по

сравнению с AODV является увеличение трафика в связи с необходимостью хранения маршрутов в теле сообщений.

Авторы статьи [38] предлагают алгоритм для DSR, при помощи которого строится альтернативный путь, где каждый узел основного маршрута знает ближайший путь к узлам альтернативного маршрута. При этом для снижения объёма взаимодействия между абонентами используется фильтр Блюма, по которому определяется вхождение абонента в параллельный маршрут.

Реактивный протокол TORA (Temporally Ordered Routing Algorithm – Алгоритм временной маршрутизации) основан на концепции реверсирования канала [39] (рис. 1.7). Его особенностью является использование метрики «высота» – удалённость целевого узла относительно текущего. Функционирование алгоритмов заключается в поиске и обновлении показателя. В этом случае маршрут к целевому узлу является последовательностью узлов с убывающим значением высоты.

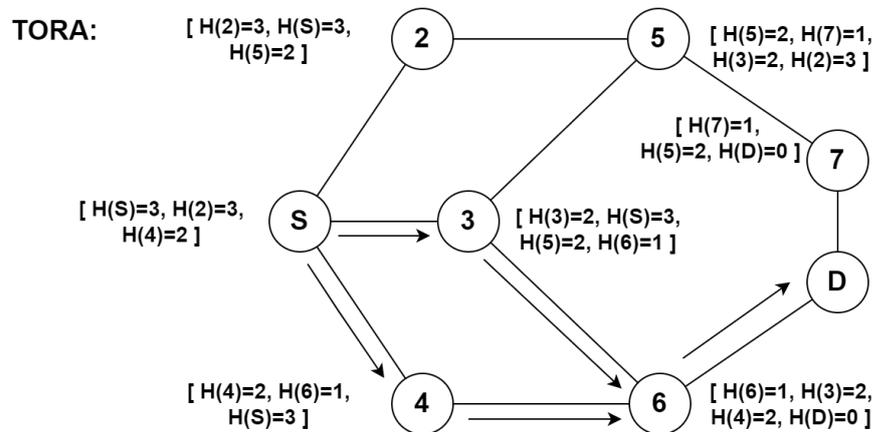


Рисунок 1.7 – Схема пересылки пакетов в протоколе TORA

Реактивный протокол маршрутизации B.A.T.M.A.N. (better approach to mobile ad-hoc networking – лучший подход к организации мобильной ДСПД сети) [40], в отличие от предыдущих технологий, работает на канальном уровне. Поэтому, по сути, при помощи данного протокола создаётся своеобразный виртуальный распределённый коммутатор. Поскольку физические адаптеры используются для связи на 2-м уровне модели OSI, то они не нуждаются в настройке IP-адресов. Поэтому в системе создаётся

виртуальный адаптер VATH, который может быть использован при необходимости передачи полезной информации.

Как и в случае с табличными протоколами маршрутизации, для организации взаимодействия ПОСМ используются метаэвристические алгоритмы. В диссертационной работе Леонова А.В. [41] взаимодействие между абонентами строится на основе алгоритма пчелиной колонии. На первом этапе производится разведка сети. До целевого абонента отправляются пакеты (пчёлы-разведчики) для определения множества доступных каналов связи до него. Затем производится пробная передача трафика – четырёх групп пакетов между абонентом-источником и абонентом-назначения. Выбор транзитивных абонентов производится согласно четырём критериям, соответствующих группам пакетов – минимальная сквозная задержка до узла, пропускная способность, максимальная продолжительность жизни сети и наибольший уровень заряда батарей. По результатам доставки пакетов определяется качество маршрутов, среди которых для дальнейшего использования выбирается лучший.

В работе [42] для обеспечения многопутевой маршрутизации используются алгоритм выбора биологического аттрактора. Подход предназначен для повышения коэффициента доставки пакетов и средней пропускной способности. В статье [43] предлагается алгоритм маршрутизации, основанный на методе оптимизации COA (cuckoo optimization algorithm). Протокол прогнозируемой энергоэффективной и надежной многоадресной маршрутизации (PEERM) [44] использует алгоритм PSO для построения надежного, энергоэффективного многоадресного дерева маршрутов. Фитнесс-функция алгоритма PSO учитывает различные параметры для увеличения срока службы узла и обеспечения стабильности пути между источником и пунктом назначения: задержки передачи данных на пути, ожидаемый расход энергии, ожидаемый коэффициент доставки на маршруте.

Метод выбора маршрута Energy-Efficient Lifetime Aware Multicast (EELAM) [45] – это схема обнаружения многоадресных маршрутов для ДСПД, которая была разработана с использованием адаптивного генетического алгоритма. EELAM функционирует на основе топологии дерева и адаптирует эволюционную стратегию вычислений (генетический алгоритм). Этот генетический алгоритм играет важную роль с точки зрения выбора оптимальных средних узлов с максимальной остаточной энергией и минимальным потреблением энергии. В статье [46] предложен механизм маршрутизации SynFAnt, основанный на нечёткой синхронизированной сети Петри (SynFPN) и алгоритме муравьиной колонии. SynFPN предназначена для моделирования функций маршрутизации и принятия решений, использующих подход синхронизированного нечеткого перехода. Алгоритм муравьиной колонии используется для поиска решения проблемы событий неопределенности в ДСПД.

Реактивные протоколы маршрутизации лучше проактивных адаптированы под функционирование высокодинамичных ДСПД, однако они обладают сопутствующими недостатками (высокой задержкой на обнаружение маршрутов, возникновением «всплесков» служебного трафика во время поиска маршрутов). Кроме того, при равномерном распределении трафика между абонентами сети эффективность реактивной маршрутизации снижается.

1.4.3. Гибридные методы взаимодействия

Гибридные протоколы маршрутизации сочетают в себя свойства реактивных и проактивных протоколов. Их концепция состоит в том, что поиск маршрута к ближайшим узлам производится по проактивной маршрутизации, а к более дальним по реактивной. Известными протоколами данной группы являются ZRP, FSR, LANMAR.

Классическим примером протоколов гибридной маршрутизации является ZRP [47] (Zone Routing Protocol – зональный протокол маршрутизации), который разработан для использования в больших масштабируемых сетях.

Протокол использует проактивный механизм для обнаружения узлов в непосредственной близости от самого узла (внутри зоны), и реактивный – для обнаружения отдалённых узлов. Для абонентов определены области, называемые зонами (рис. 1.8).

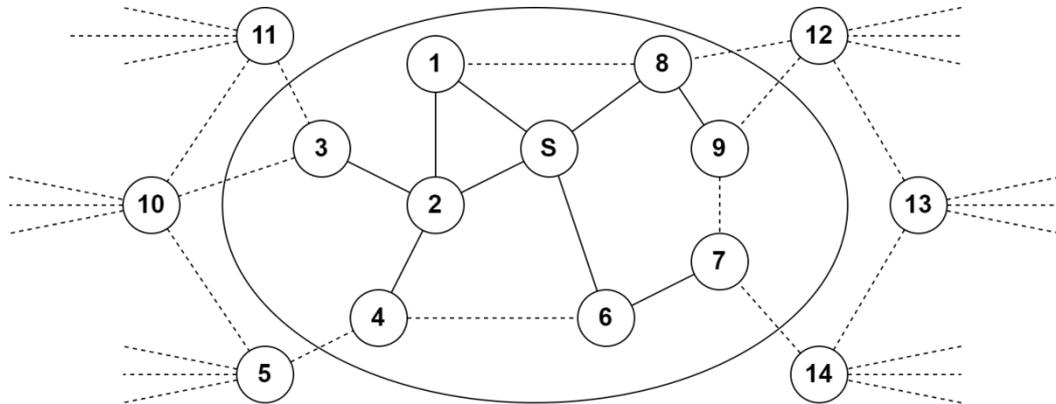


Рисунок 1.8 – Зона с радиусом 2 для ZRP

Протоколы FSR (Fisheye State Routing) и LANMAR (Landmark routing protocol) по сути являются улучшением идей, предложенных в ZRP. Однако всем им присущи недостатки – необходимость определения низкодинамичных зон проактивной маршрутизации. Выделение этих зон заранее может нарушить концепцию ДСПД, поскольку в большинстве случаев придётся искусственно поддерживать их для обеспечения маршрутизации.

1.4.4. Методы взаимодействия с учётом местоположения

Схемы взаимодействия между ПОСМ с учетом местоположения (или географические) в мобильных одноранговых сетях предполагают, что узлы осведомлены о местоположении, направлении движения других абонентов в сети. Используя информацию о положении и перемещении абонентов, производится построение маршрутов сети. Позиционирование узлов в пространстве, в частности, может осуществляться при помощи глобальных систем позиционирования, например, GPS, ГЛОНАС. Соответственно, важной задачей является обеспечение эффективного взаимодействия абонентов для определения физических координат друг друга.

Одним из распространённых протоколов с учётом местоположения является LAR (Location aware routing) [48]. Информация о местоположении

используется для минимизации пространства поиска при обнаружении маршрутов между абонентами. Позиционирование производится при помощи GPS. При вычислении местоположения целевого узла, а также при получении информации о его направлении и скорости движения, источник отправляет запросы маршрута узлам только в «ожидаемой зоне» нахождения целевого узла. Если исходный узел не имеет информации о скорости и направлении, вся сеть рассматривается как ожидаемая зона.

LAR использует две схемы для определения того, находится ли узел в зоне запроса (рис. 1.9). В первой схеме узел просто включает в себя наименьший прямоугольник, содержащий текущее местоположение исходного узла и ожидаемую зону целевого узла на основе его исходного местоположения и текущей скорости. Во второй схеме узел вычисляет расстояние между собой и конечным узлом на основе координат GPS и включает эти значения в пакеты запроса маршрутов. Промежуточный узел, получающий этот пакет, вычисляет свое расстояние от пункта назначения. Если его расстояние от пункта назначения больше, чем расстояние от источника, промежуточный узел не находится в зоне запроса и, следовательно, отбрасывает пакет. В противном случае он пересылает пакет всем своим соседям.

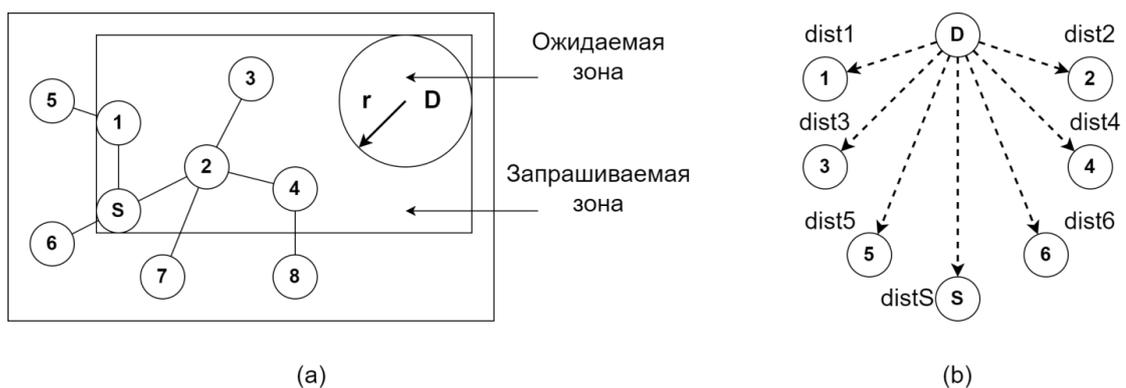


Рисунок 1.9 – Схемы определения нахождения узла в зоне опроса 1 (a) и 2 (b)

Аналогичный подход используется в работах [49–51], в которых маршрутизация предполагает возможность прогнозирования положения узла, если известны скорость и направление его перемещения. В ряде подходов [50; 52] для передачи данных используется разделение узлов на географические зоны, пересылка данных производится через ближайшую зону.

Несмотря на то, что использование информации о местоположении существенно сокращает накладные расходы на поиск и обнаружение маршрутов к узлам назначения, вся сеть попадает в зависимость от качества работы системы определения положения узла, а также от наличия самой системы. Во многих средствах связи таковые средства отсутствуют. Возникает потребность в постоянном оповещении других абонентов сети об изменении своего местоположения. Кроме того, подход не учитывает наличие препятствий и помех, возникающих вследствие высокой степени пересечённости местности, техногенных строений, радиопомех, которые препятствуют приёму сигналов. Поэтому данный подход нашёл широкое применение, в основном, в сетях FANET беспилотных летательных аппаратов.

1.4.5. Иерархическая маршрутизация

В то время как традиционная интернет-маршрутизация изначально иерархична, в первом приближении специальные сети маршрутизируются через плоскую совокупность узлов. По мере увеличения размера сетей эти подходы приводят к увеличению размеров таблиц маршрутизации и накладных расходов на пакеты управления. Протоколы иерархической маршрутизации в ДСПД создают иерархию узлов обычно с помощью методов кластеризации. Узлы на более высоких уровнях иерархии предоставляют специальные услуги, улучшая масштабируемость и эффективность маршрутизации.

Классическими протоколами иерархической маршрутизации являются H-LANMAR [53], CEDAR (Cluster-based Energy-aware Data Aggregation Routing) [54]. На рис. 1.10 представлена общая схема иерархической маршрутизации, характерная для большинства иерархических протоколов маршрутизации. Она заключается в том, что узлы объединяются в определённые кластеры, внутри которых осуществляется плоская маршрутизация, но для связи с узлами из соседнего кластера используются узлы-шлюзы. Шлюзы кластеров также могут объединяться в кластеры, организуя сложную древовидную сеть.

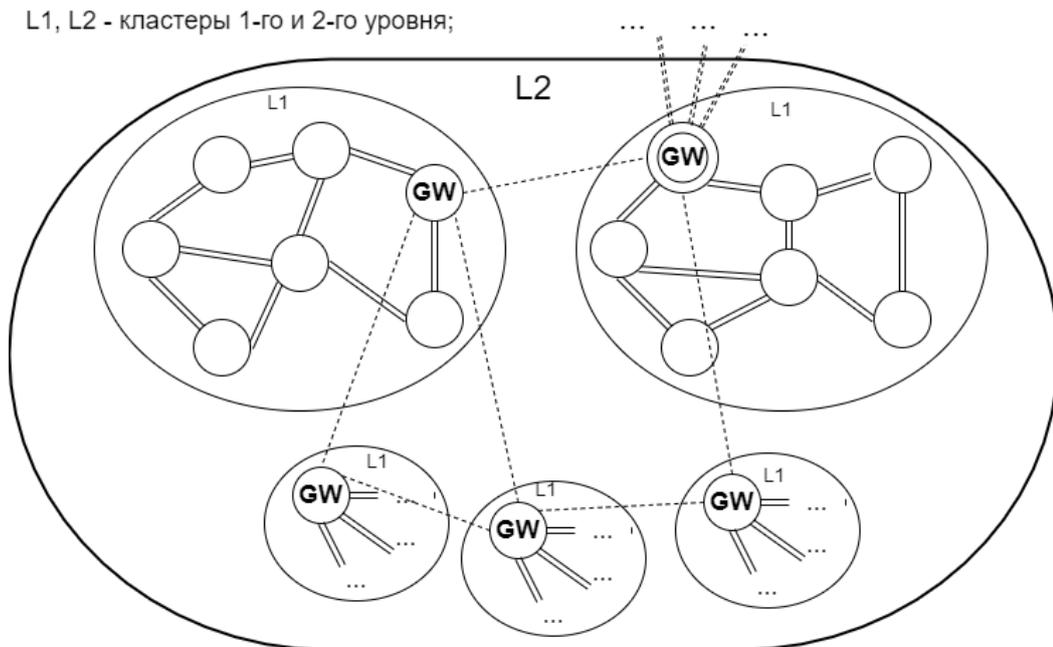


Рисунок 1.10 – Схема иерархической сети

Часто в иерархической маршрутизации используются координирующие абоненты, роль которых могут выполнять как специально вводимые в сеть устройства, так и некоторые узлы сети [55, 56].

Тема взаимодействия абонентов в иерархических сетях разрабатывалась отечественными учёными. В диссертационном исследовании «Методы иерархической маршрутизации мобильной самоорганизующейся сети доступа» Романова С.В. [57] представлен метод иерархической маршрутизации ad-hoc сети. Он включает алгоритмы иерархической, внутрикластерной и межкластерной маршрутизации, методы обработки и передачи служебных пакетов между ПО маршрутизации трафика ДСПД.

В исследовании предложена стратегия распространения информации о структуре кластеров между абонентами. Она подразумевает, что каждое ПОСМ маршрутизации является источником широковещательных сообщений и ретранслятором широковещательного трафика для абонентов своего кластера. Для внутрикластерной маршрутизации используется протокол геомаршрутизации. Обмен служебными сообщениями возникает в случае изменения узлом координат (его движения). Кластер образуется на основе данных о расположении узлов в пространстве.

Каждый кластер имеет один или несколько узлов-шлюзов, при помощи которых имеется возможность доставить данные в другие кластеры. Обладая информацией о других кластерах, шлюзы могут переправлять сообщения по нужному пути. Поэтому межкластерная маршрутизация заключается в выборе маршрута к шлюзу, через который будет иметься возможность доставить пакет к нужному узлу на других кластерах.

Преимуществом данного подхода является существенное снижение служебного трафика, особенно в сетях с низкой подвижностью узлов. Тем не менее, при высокой подвижности узлов доля служебного трафика в сети является приемлемой. Кроме того, имеется существенный недостаток, приемлемый иным методам иерархической маршрутизации: работоспособность сети зависит от надёжности ведущих (центральных) узлов кластеров и от шлюзов. Поэтому при отсутствии резервных узлов, выполняющих данные функции, могут оказаться неработоспособными целые сегменты сети.

В диссертации [58] «Использование нейронных механизмов искусственного интеллекта для кластеризации узлов и маршрутизации данных в беспроводных сенсорных сетях» Махрова С.С. предлагается разбивать сеть связи на кластеры методами, основанными на нейронной сети Кохоннена и матричном способе разбиения. В отличие от предыдущей работы, здесь используется так называемая энергетическая видимость соседних абонентов, прямо пропорциональная мощности входных сигналов о них. Узлы взаимодействуют между собой для построения матрицы радиовидимости. Содержимое матрицы является входными данными для дальнейшей кластеризации.

Существенным недостатком данных сетей является необходимость в статичности узлов и внутрикластерных топологий, в ином случае накладные сетевые расходы на восстановление сети способны существенно снизить основные показатели качества работы сети или сделать сеть полностью неработоспособной.

Иерархические методы маршрутизации имеют низкую устойчивость к высокой динамике изменения топологии сети, поскольку для восстановления

маршрутов передачи данных требуется передавать большой объём данных, включая уровни иерархии. Кроме того, выход из строя главных кластерных узлов или кластерных шлюзов может приводить к поломкам целых сегментов сети.

1.5. Взаимодействие абонентов при балансировке трафика в сетях ДСПД

Значительная часть классических методов маршрутизации в беспроводных децентрализованных сетях не предполагают механизмов балансировки трафика, вследствие чего отсутствует потребность в механизмах взаимодействия сетевого ПО для реализации этих механизмов. При этом балансировка нагрузки возможна в тех случаях, когда алгоритм маршрутизации может определять несколько непересекающихся маршрутов. Например, такие решения могут быть основаны на протоколах AOMDV [59], OLSR [60], DSR [61].

Наиболее простые методы балансировки трафика в беспроводных ДСПД основаны на алгоритме round-robin или циклическом выборе маршрутов [59, 62]. Предполагается, что трафик равномерно распределяется по найденным путям. Соответствующие подходы не требуют взаимодействия ПОСМ. Отсутствие учёта критериев производительности и их изменения является причиной низкой эффективности подхода для обеспечения балансировки нагрузки маршрутов.

В работе «Congestion-Aware Fibonacci Sequence Based Multipath Load Balancing Routing Protocol for MANETs» [63] Naseem M. и Kumar C. несколько параллельных маршрутов связи сортируют в порядке убывания количества абонентов в составе маршрута. Нумерация маршрутов начинается с «2», при этом пути равной длины нумеруют одинаковым индексом. Распределение трафика производится в пропорциях, равным числам Фибоначчи.

В статье [64] предлагается протокол LCMR (Least Common Multiple based Routing), который предполагает построение нескольких маршрутов и балансировку трафика между ними. Пакеты трафика распределяются обратно пропорционально времени, затраченного на обнаружение пути. Во всех вышеприведённых подходах ПОСМ не взаимодействуют между собой для

корректировки весов маршрутов, поэтому подход применим исключительно для сетей со стабильными параметрами работы. Более того, авторы [63] делают допущение, что за время передачи данных топология сети и параметры каналов связи не претерпевают существенных изменений.

В протоколе LOBAM, разработанного R. Ktari, F. Xue и P. Huang [60], основанном на OLSR, сообщения Hello включают информацию о загруженности (размеру очереди сообщений) собственного и соседних абонентов. Трафик между маршрутами распределяется в зависимости от загруженности следующих двух абонентов в маршруте. При этом баланс маршрута i вычисляется по следующей формуле (1.2):

$$i_bal = \frac{\left(qoc_F + \sum_1^N qoc_i \right)^2}{(N+1) \times \left(qoc_F^2 + \sum_1^N qoc_i^2 \right)}, \quad (1.2)$$

где qoc_F – занятость очереди следующего узла, qoc_i – занятость очереди соседнего узла, N – общее количество соседей следующего узла. Проведённое авторами экспериментальное исследование продемонстрировало, что описанный подход, в котором абоненты взаимодействуют для обеспечения балансировки нагрузки, позволяет повысить долю доставляемых пакетов до целевого абонента и снизить среднюю задержку очереди сообщений по сравнению со стандартным OLSR и протоколами с балансировками, основанными на случайном и циклическом выборе путей.

V. Ramesh и другие в своей работе «Performance Comparison of Congestion Aware Multi-Path Routing (with Load Balancing) and Ordinary DSR» [61] реализуют взаимодействие между абонентами для обеспечения балансировки производится путём включения в пакеты RREQ и RREP информации о загруженности маршрутов. После каждой пересылки производится перерасчёт загруженности. Метрикой эффективности маршрута является произведение загрузки маршрута и его длины. Чем меньше её значение, тем больше пересылается по этому пути. Однако ПОСМ не взаимодействуют между собой для

контроля загрузки маршрутов после завершения процесса их обнаружения. Поэтому эффективность подхода снижается с повышением динамики изменения параметров сети.

В работе С. Амхамди «Load balancing ad hoc on-demand multipath distance vector (LBAOMDV) routing protocol» [29] рассматривается протокол маршрутизации с балансировкой трафика, целью которого распределение энергетической нагрузки на узлы сети. В качестве метрики рассматривается энергопотребление абонентами маршрутов и неиспользуемая полоса пропускания. Аналогично предыдущему подходу, данные об энергопотреблении абонентов собираются на этапе обнаружения маршрутов. Маршруты с низким остаточным уровнем энергии не используются в дальнейшей балансировке. Трафик распределяется по маршрутам в соответствии с их остаточной пропускной способностью. Отличием исследования J. Pourbemanu и др. [65] от предыдущего является использование в качестве метрик средний размер очереди или среднее количество байтов в очереди абонентов, который определяются путём взаимодействия абонентов связи на этапе обнаружения маршрутов связи. Также предлагаются подходы, в которых производится оценка загрузки абонентов сети [66].

В некоторых работах имеется попытка совместить различные метрики производительности маршрутов. Так в подходе [67] описан способ и система распределенной балансировки трафика в беспроводной сенсорной сети на основе алгоритма маршрутизации от узла источника к узлу назначения, где беспроводная сенсорная сеть представляется как граф $G(N, M)$, где N узлы сети, а M грани, имеется K маршрутов (рис. 1.11).

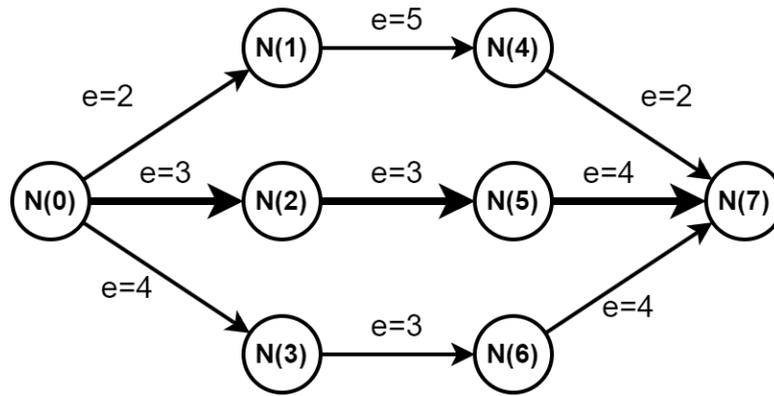


Рисунок 1.11 – Граф сенсорной беспроводной сети

Информация генерируется со скоростью Q_c и передается по каналу связи C со скоростью q_c , причем i -й узел имеет запас энергии E_i , а каждая грань ij имеет вес e_{ij} , который соответствует энергии для передачи одного пакета данных от узла i к j , а время жизни T_i каждого узла определяется как (1.3):

$$T_i = \frac{E_i}{\sum_{j \in M} q_c \sum_{j \in M} e_{ij}^e} \quad (1.3)$$

На каждом узле определяется таблица маршрутизации и выстраивается вектор передачи сообщения, проводится анализ вариантов маршрутов по наиболее оптимальным суммарным векторам, которые рассчитываются по таблице маршрутизации. Для этого определяется время жизни всей $T_{sys} = \min T_i(q_c), j \in N$. Для достижения максимального времени жизни всей сети распределяют маршруты, где их выбор основан на использовании наименее затратных передач на каждом узле, а наиболее затратные исключаются.

Общим недостатком приведённых выше методов является отсутствие механизмов актуализации данных о маршрутах после завершения процесса их поиска. ПОСМ не взаимодействуют после окончания процесса их построения для определения изменившихся параметров трафика. Ввиду высокой динамики изменения топологии ДСПД характеристики найденных маршрутов могут существенно изменяться во время передачи данных, что способствует снижению эффективности предложенных методов балансировки нагрузки.

1.6. Анализ и выбор маршрутных метрик в ДСПД

Используемые для оценки маршрутов метрики непосредственно влияют на эффективность передачи данных по сети [68]. Важными показателями эффективности являются доля доставляемых до целевого абонента пакетов (PDR – packet delivery ratio) и сквозное время передачи пакетов (E2ED – end to end delay) [69].

В классических протоколах связи сетей ДСПД для оценки маршрутов используются такие метрики, как количество переходов между абонентами (длина маршрута) или время обнаружения маршрута [64]. Их расчёт не требует сложных алгоритмов взаимодействия ПОСМ абонентов. Длина маршрута определяется количеством переходов, которое совершили поисковые пакеты при прохождении до целевого узла или обратно. Время обнаружения маршрута обычно принимается равным длине временного промежутка от момента инициализации процедуры поиска до момента, когда узел-инициатор сможет начать передачу данных по найденному маршруту.

Оценка маршрутов по приведённым выше метрикам является достаточно поверхностной, поскольку слабо учитывается загруженность абонентов, надёжность каналов связи. Такие подходы остаются эффективными в сетях с относительно стабильной топологией, небольшим объёмом трафика и низкими скачками нагрузки каналов связи. Использование более сложных метрик требует обеспечения взаимодействия программного обеспечения сетевой маршрутизации для сбора информации о загрузке абонентов и каналов связи, их надёжности.

Существуют исследования, предлагающие решения, где формирование метрик является результатом взаимодействия нескольких абонентов. В ранее рассмотренных трудах [60, 61, 65] рассчитывается загрузка абонентов маршрута. В работе [70] в качестве отдельных метрик для повышения производительности OLSR предлагаются MD (минимальная задержка), ML (минимальные потери). Для расчёта MD абоненты обмениваются временем приёма пакетов, в то время как для ML – количество принятых и отправленных соседу пакетов.

С. Ци в работе «Multi-path Routing Improved Protocol in AODV Based on Nodes Energy» [71] предлагает в качестве маршрутной метрики использовать информацию об остаточной энергии маршрута. Метрика вычисляется по формуле (1.4):

$$P_{ri} = \sqrt{\left(\frac{0.6E_{min} + 0.4E_{SUM}/S_{SUM}}{E_{initial}}\right)^2 + \left(\frac{1}{L}\right)^2}, \quad (1.4)$$

где E_{min} – минимальная энергия узла в маршруте, E_{SUM} – сумма остаточной энергии всех узлов, $E_{initial}$ – энергия узла инициатора, S_{SUM} – длина маршрута. Приоритетным является маршрут с наибольшим значением метрики P_{ri} . Аналогичные метрики для оптимизации энергоэффективности маршрутизации, требующие взаимодействия абонентов для сбора информации, предложены в [29, 67, 72].

Существуют исследования, в которых прорабатывается вопрос использования комплексных метрик [73], сбор и комплексирование которых требует взаимодействие абонентов. В [74] предлагается метрика, являющейся мультипликативной свёрткой значений загрузки абонента связи и его остаточной энергии. В [75] используется аналитическая метрика на основе энергопотребления абонентов связи, скорости обмена данными и задержек между абонентами маршрута.

В статье [76] А. Таха «Energy Efficient Multipath Routing Protocol for Mobile Ad-Hoc Network Using the Fitness Function» предложен протокол маршрутизации FF-AOMDV. Он основан на протоколе AOMDV, который в свою очередь является многопутевой версией стандартного AODV. Особенностью предлагаемого протокола является определения качества маршрутов при помощи фитнес-функции, рассчитываемой по нескольким показателям: уровень энергии каждого узла, длина маршрута, расстояние между абонентами. Оптимальный маршрут рассчитывается по следующей формуле (1.5):

$$Optimum_route_1 = \frac{\sum_{v(n) \in r^*} ene(v(n))}{\sum_{v(n) \in V^*} ene(v)}, \quad (1.5)$$

где v – вершины оптимального маршрута r , V – все вершины сети, eve – энергия. По приведённой формуле производится сравнение уровня энергии среди всех маршрутов и выбирается маршрут с её самым высоким уровнем. Расчёт кратчайшего пути с информацией о расстоянии производится следующим образом (1.6):

$$Optimum_route_2 = \frac{\sum_{e(n) \in r} * dist(v(n))}{\sum_{e \in E}}, \quad (1.6)$$

где e – рёбра (звенья) в оптимальном маршруте, а E – представляет все рёбра в сети.

После обнаружения маршрутов абонент-отправитель взаимодействует с целевым абонентом по найденным путям, чтобы проверить их эффективность. По итогу обмена данными, отбрасываются маршруты низкого качества, ошибочно определённые как оптимальные.

В диссертационном исследовании [77] «Разработка алгоритмов передачи потоковых данных на прикладном уровне в сетях беспилотных летательных аппаратов» Васильева Д.С. для повышения эффективности передачи данных по сети FANET (сети беспилотных летательных аппаратов) предложено использовать методы многопутевой избыточности и запросы повторной передачи. В качестве метрики качества используется нормированная пропускная способность η . Алгоритм выбора маршрута основан на использовании в наложенной сети 3-ёх узлов, соединённых виртуальными каналами: узла источника S , узла назначения D и узла помощника R . S может как напрямую отправлять данные узлу D (состояние SD), так и с использованием R (состояние SRD).

Для выбора направления имеются 3 критерия, использующие информацию о предыдущих передачах на данные узлы. Согласно пошаговому критерию, выбирается маршрут, по которому была осуществлена подтверждённая передача. Применение равновесного критерия заключается в равномерном распределении по маршрутам SD и SRD. Адаптивный критерий основан на использовании коэффициента доставки пакетов (PDR): используется маршрут с наибольшим показателем.

Узел R не является статичным: предложен алгоритм для его выбора на основе соревновательного, жадного и предиктивного критериев. Первый критерий определяет узел R , который первый откликнулся на специальный запрос, второй выбирает R как ближайшего соседа. В третьем случае при помощи протоколов геомаршрутизации производится прогноз положения соседа через некоторое время и выбирается в качества узла-помощника географически ближайший узел. Алгоритм взаимодействия абонентов для выбора узла помощника представлен на рис. 1.12.



Рисунок 1.12 – Алгоритм выбора узла помощника

Основным плюсом предложенного в диссертации подхода является возможность использования разработанных методов совместно с существующими протоколами, повышая их эффективность (в среднем в 1.6 от значения метрики качества без использования разработанных алгоритмов).

В диссертации [78] Дугаева Д.А. «Разработка адаптивного алгоритма маршрутизации для беспроводных многоузловых сетей передачи данных» предложена схема адаптивной маршрутизации, которая основана на математическом аппарате машинного обучения с подкреплением. При этом задача маршрутизации с задачей обучения с подкреплением связана следующим образом. С понятием «агент» ассоциирован узел-источник отправки сообщения, с целью – узел-получатель, соседи выступают в роли множества действий A , а таблица маршрутизации эквивалентна набору значений ожидаемой награды Q . Если узел-источник посылает сообщение узлу-соседу, это означает, что агент выбрал действие и выполнил его. Если узел-источник получил подтверждение приёма пакета от соседа (через пакет АСК), это значит, что агенту выдана награда за действие.

Для выбора направления дальнейшего шага в маршруте используются веса направлений, которые определяют вероятность выбора следующего шага в маршруте. Данные веса Q формируются на основе наград за выбор действий. При потере пакетов с данными или обратных сообщений АСК выбранная связь получает отрицательную награду или штраф, из-за чего вероятность её дальнейшего выбора снизится. При больших задержках получения ответного сообщения АСК формируются положительная награда с низким значением. Если принятое сообщение АСК содержит высокое значение награды, это означает, что маршрут через связанные узлы хороший, и он является предпочтительным по сравнению с остальными.

Тогда предлагаемый алгоритм выбора маршрута используют метод взвешенного действия, в основе которого лежит вероятностное распределение Гиббса-Больцмана (1.7):

$$P_t(a) = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^n \frac{Q_t(b)}{\tau}}, \quad (1.7)$$

где $P_t(a)$ – вероятность выбора действия a (выбор маршрута) на шаге t , $Q_t(a)$ – ожидаемое значение награды за выбор действия a на шаге t ,

$Q_t(b)$ – ожидаемое значение награды за другое действие на шаге t , τ – некоторый положительный параметр. Данный параметр динамически меняется в зависимости от количества потерь PLR на линии связи (1.8):

$$\tau(PLR) = \begin{cases} 10, & \text{если } PLR \leq 1 \\ 10k(PLR - 1) + 10, & \text{если } PLR > 1 \end{cases}, \quad (1.8)$$

где k – предполагаемый коэффициент роста.

Главным преимуществом предложенной схемы маршрутизации является высокая эффективность в сетях с высоким трафиком. Основным недостатком – высокая инертность определённых маршрутов: они могут существовать достаточно долго при том, что они могли потерять актуальность. Для обновления маршрутов требуется большое количество трафика для определения качества отдельных каналов связи.

При этом не учитываются надёжность маршрутов, загруженность каналов и узлов связи, расход энергии источника питания на передачу данных. Более сложные метрики, такие как прогнозируемая длина маршрута, расход энергии на передачу пакетов позволяют улучшить лишь отдельные показатели функционирования ДСПД. При этом существующие подходы предполагают сохранение надёжности R передачи данных, но не улучшают напрямую показатель. В тоже время важна оперативность доставки данных – полный пакет информации необходимо доставить до разрыва маршрута. Известные методы оптимизации протоколов связи, хотя и улучшают качество их функционирования, не решают указанных проблем.

Одними из наиболее применяемых маршрутных метрик в рассмотренных ранее работах [70, 71, 76] являются показатели, основанные на энергопотреблении узлов маршрута, а также их остаточных зарядов. Это позволяет увеличить время работы каждого абонентского устройства (с автономными источниками питания) и срок службы сети в целом. С другой стороны, в большинстве случаев такой подход приводит к снижению одного или нескольких критериев эффективности. В частности, повышаются задержки передачи трафика ввиду выбора более длинных, но при этом, более энергоэффективных

участков маршрута, снижается надёжность доставки данных из-за включения в маршрут менее надёжных каналов связи.

Поэтому для оценки маршрутов связи необходимо выбирать те, которые непосредственно могут влиять на рассмотренные критерии качества путей передачи трафика. Рассмотрим частные метрики [79] маршрутов, которые должны быть собраны в результате взаимодействия служб маршрутизации, и должны быть ими применены для комплексной оценки маршрутов связи:

1. Длина маршрута N – количество узлов в маршруте связи. Чем меньше абонентов в маршруте, тем выше вероятность доставки данных до конечного абонента и того, что время доставки будет минимальным. Однако ввиду влияния на абонентов сети и каналов связи множество факторов, как надёжность каналов связи, загрузка абонентов и их мобильность, отсутствует гарантия, что кратчайших маршрут – оптимален [80].

2. Надёжность маршрута R_{ch} определяет, какая часть данных будет доставлена по маршруту между абонентами. Он представляет собой произведение показателей надёжности отдельных каналов между абонентами маршрута (1.9):

$$R_{ch} = \prod_{i=1}^{N-1} RDL(i, i+1), \quad (1.9)$$

где $RDL(i, i+1)$ – надёжность канала связи между 2-мя последовательными абонентами $i, i+1$ маршрута. $RDL(i, i+1)$ может быть рассчитана как отношение количества пакетов, отправленное абонентом i абоненту $i+1$, к количеству пакетов, принятое абонентом $i+1$.

На значение R_{ch} влияют следующие факторы:

- наличие помех (природных и техногенных) между абонентами;
- взаимное расположение узлов в пространстве. Чем ближе географически абоненты располагаются друг к другу, тем выше качество связи. Как метрики можно выделить $Dist_{av}$, $Dist_{max}$ – среднее и максимальное расстояния между абонентами маршрута [81]. На взаимное расположение влияет скорость

перемещения абонентов относительно друг друга. V_{av} , V_{max} – средняя и максимальная скорости перемещения узлов маршрута [82];

– мощность сигнала. Чем он выше, тем выше вероятность того, что передаваемые данные будут приняты;

– перегрузка абонентов маршрута. Чем больше данных принимает абонент, тем больше коллизий возникает во время их передачи, из-за чего сообщения могут быть потеряны.

3) Загрузка каналов связи (использование полосы пропускания) маршрута показывает, какой объём трафика от общей пропускной способности проходит по каналам связи маршрута. На перегруженных каналах связи в пакетах чаще будут возникать ошибки целостности, а задержка их передачи будет расти. В качестве метрик можно использовать следующие величины [83]:

– L_{max} – максимальная загрузка на одном из каналов связи маршрута;

– L_{av} – среднюю загрузку канала связи.

4) Загрузка узлов связи.

Загрузка узлов маршрута может быть выражена следующими показателями:

– Q_{av} , Q_{max} – средний и максимальный размеры очереди пакетов на транзитных абонентах [84];

– D_{av} , D_{max} – средняя и максимальная задержки пакета на узлах маршрута. Задержки могут быть обусловлены обработкой пакета, ожиданием в очередях на обработку (входящая очередь) и на отправку [85];

– R_n – доля данных, которые не теряются при прохождении пакетов через абонентов сети. Ввиду слишком долгого пребывания пакетов в очереди они могут быть удалены, и как следствие, потеряны.

5) Количество пересекающихся актуальных маршрутов [86]. Данная метрика показывает, со сколькими другими действующими маршрутами связи происходит пересечение. Для данного показателя целесообразно использовать 2 составляющие:

– общее количество пересечений с другими маршрутами отдельными каналами связи (1.10):

$$I_{all} = \sum_{i=1}^{N-1} count_{inters}(i, i+1), \quad (1.10)$$

где $count_{inters}(i, i+1)$ – количество пересечения с активными маршрутами на канале между узлами $i, i+1$;

– максимальное количество пересечений с другими маршрутами на одном канале связи (1.11):

$$I_{max} = \max(count_{inters}(i, i+1)). \quad (1.11)$$

Перечисленные метрики являются важными для оценки критериев эффективности маршрутов, поскольку непосредственно влияют на них. Используя совокупность метрик, можно выдать комплексную оценку о степени применимости пути для маршрутизации трафика. Однако на сегодняшний день не существует эффективного механизма для вывода аналитических зависимостей от множества значений. Поэтому при разработке комплексной метрики далее целесообразно обратиться к алгоритмам машинного обучения.

1.7. Выводы по главе

В результате проведённого анализа современных методов, алгоритмов и протоколов взаимодействия программного обеспечения сетевой маршрутизации, предназначенных для обнаружения и обслуживания маршрутов передачи данных в ДСПД, балансировки трафика в них и их оценки, были сформированы следующие выводы:

1. Анализ существующих проактивных и реактивных методов и алгоритмов взаимодействия программного обеспечения сетевой маршрутизации показал, что они обеспечивают высокую эффективность построения, обслуживания маршрутов и передачи трафика по ним, в основном, в сетях с относительно низкой динамикой изменения топологии.

2. Анализ известных оптимизаций указанных базовых подходов показал, что они не позволяют достичь существенного прироста

производительности, либо улучшают лишь частные маршрутные метрики, при этом альтернативные подходы взаимодействия абонентов, как гибридные, иерархические и географические имеют узкую специализацию и пригодны для частных сценариев функционирования ДСПД;

3. Из-за высокого объёма служебного трафика в проактивном взаимодействии маршруты обновляются относительно редко, и при высоком уровне мобильности абонентов не успевают обновляться;

4. ПОСМ реактивных протоколов маршрутизации взаимодействуют косвенно через сообщения запросов маршрутов, из-за чего отсутствуют возможности по комплексной оценке путей, нахождению лучших маршрутов и оперативному их восстановлению в случае разрыва канала.

Поэтому для достижения поставленной в диссертационной работе цели необходима разработка методов обеспечения взаимодействия ПОСМ абонентов, которые отличаются от уже известных применением локализованной проактивной коммуникацией в реактивных схемах, а также использованием квазипараллельных маршрутов для повышения надёжности и снижения сквозной задержки доставки данных по сети

При этом необходимо разработать методы взаимодействия сетевого ПО при решении основных задач маршрутизации, которые ранее были описаны в главе, а именно: при установлении и дальнейшем поддержании соединения с сетью, построении квазипараллельных маршрутов связи между целевыми абонентами, восстановлении маршрутов передачи данных, передачи трафика в ДСПД.

5. При балансировке трафика целесообразно использовать так называемые квазипараллельные маршруты, для которых является допустимым, но не обязательным наличие общих промежуточных узлов, что в свою очередь требует эффективных алгоритмов взаимодействия сетевого ПО для построения таких путей.

6. Расчёт эффективности маршрутов должен быть основан на комплексировании множества маршрутных метрик, при этом в качестве инструмента

их определения целесообразно применять технологию машинного обучения. Среди частных метрик в ходе проведённого анализа были выбраны длина маршрута, потери трафика при прохождении через узлы и каналы связи, загрузка каналов маршрута, количество пересечений маршрута с альтернативными.

7. В качестве показателя, при помощи которого можно оценивать эффективность передачи трафика по сети, выбран запас времени на передачу потока данных, поскольку данный критерий учитывает как производительность, и так и надёжность передачи данных.

Таким образом, за основу разрабатываемых далее методов взаимодействия ПОСМ необходимо взять реактивный подход, поскольку он существенно лучше адаптирован для условий функционирования высокодинамичных ДСПД. Полученные в главе выводы являются основанием для проведения дальнейших исследований.

2. МЕТОДЫ И АЛГОРИТМЫ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ УЗЛОВ ПРИ МАРШРУТИЗАЦИИ ТРАФИКА В ДСПД

По результатам проведённого в первой главе анализа было установлено, что для достижения высокой эффективности передачи данных по децентрализованным сетям требуется построение многопутевых маршрутов и балансировка трафика между ними. Для решения этой задачи в диссертации предлагается использовать взаимодействие программного обеспечения сетевой маршрутизации отдельных узлов, которое обеспечивает информационный обмен о состоянии топологии сети, определение частных и обобщённых маршрутных метрик на основе полученной информации и построение новых путей передачи данных. Для этого необходимо разработать следующие методы:

- метод взаимодействия программного обеспечения сетевой маршрутизации при установлении и дальнейшем поддержании соединения с сетью, включая формирование графа соседних узлов первого и второго порядков, сбор динамических характеристик каналов связи (ДХКС) и динамических характеристик сетевых узлов (ДХСУ) для дальнейшей оценки многопутевых маршрутов;

- метод взаимодействия ПОСМ при построении квазипараллельных маршрутов связи между целевыми абонентами, включая сбор частных маршрутных метрик и определение показателя эффективности передачи данных по маршруту;

- метод взаимодействия сетевого ПО при восстановлении маршрутов передачи данных за счёт включения или исключения из пути абонентов связи;

- метод балансировки потоков данных по маршрутам в ДСПД на основе децентрализованного взаимодействия сетевых узлов при распределении пакетов трафика по квазипараллельным маршрутам.

Для практической реализации методов необходимо разработать соответствующие алгоритмы. Общая структура предлагаемого в диссертационной работе решения представлена на рис. 2.1.

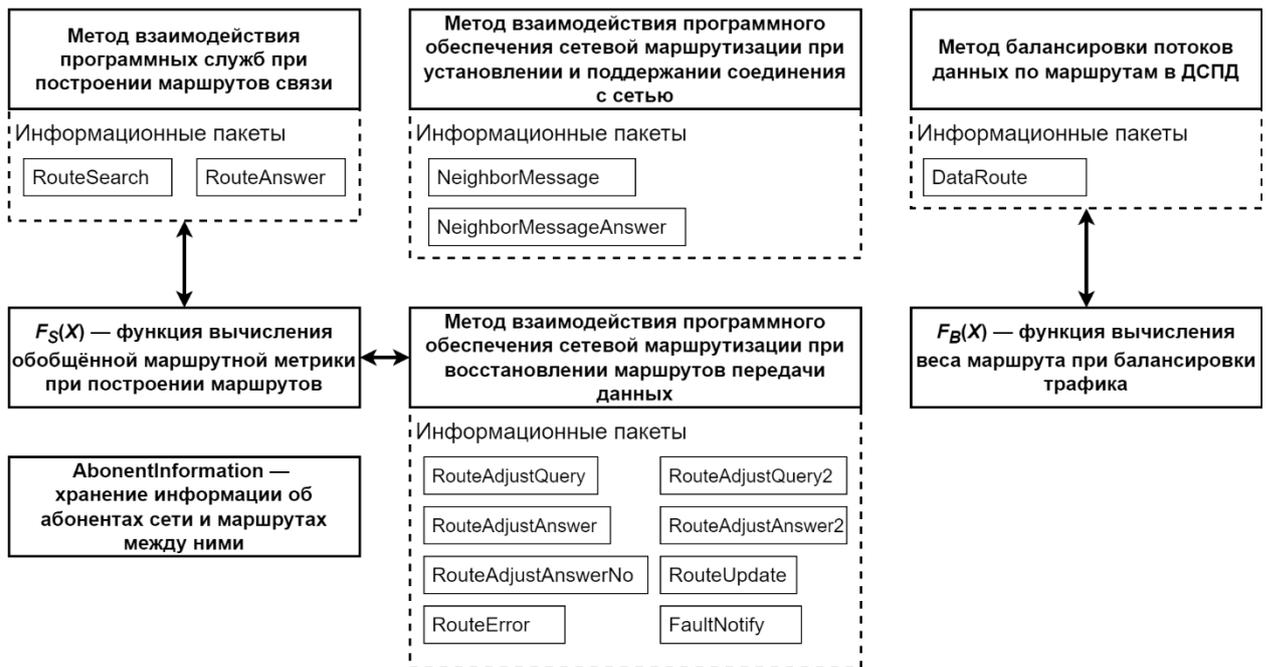


Рисунок 2.1 – Структура предлагаемого решения

Решение включает в себя разработку и реализацию методов взаимодействия ПОСМ узлов для выполнения процедур маршрутизации в ДСПД, метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия ПОСМ, метод балансировки потоков трафика в ДСПД.

2.1. Метод взаимодействия ПОСМ при подключении узла к ДСПД и поддержания стабильного сетевого соединения

Общая структура метода взаимодействия ПОСМ при подключении узла к ДСПД и поддержания стабильного сетевого соединения представлена на рис. 2.2. В процессе установления соединения абонента с ДСПД производится обнаружение узлов, с которыми обмен данными осуществляется напрямую без использования промежуточных абонентов-ретрансляторов. Такие узлы называются соседними (смежными) абонентами первого порядка [87]. Если сетевые узлы имеют общего соседнего абонента первого порядка, но сами не являются таковыми относительно друг друга, то они определяются как смежные узлы второго порядка. Метод предназначен для формирования графа соседних абонентов первого и второго порядков и поддержания его в актуальном состоянии. Наличие такого графа позволяет оперативно передать данные целевому абоненту напрямую без процедуры построения маршрута. Метод от известных

отличается тем, что широковещательный запрос на установление соединения совмещается с предыдущими ответами на запросы других узлов в одном служебном сообщении, которое содержит информацию о ДХКС и ДХСУ известных смежных абонентов первого порядка. Это обеспечивает формирование графа соседних узлов первого и второго порядков, а также сбор информации, необходимой для построения и восстановления маршрутов передачи данных.

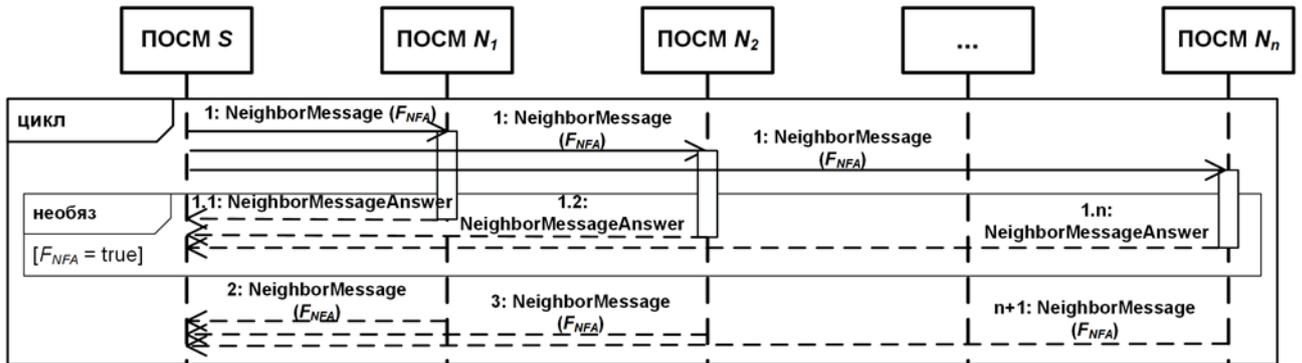


Рисунок 2.2 – Структура метода взаимодействия ПОСМ при подключении абонента к ДСПД и поддержания стабильного сетевого соединения

ПОСМ узла S должна периодически через равные промежутки времени посылать широковещательные пакеты `NeighborMessage` другим абонентам с информацией о соседних узлах первого порядка, с которыми установлено соединение, и свои динамические характеристики. При этом S может запросить подтверждение `NeighborMessageAnswer` от каждого узла получателя. Соединение с узлом S считается установленным, если от него в течение заданного периода времени поступали сообщения `NeighborMessage` или `NeighborMessageAnswer`.

При этом пакеты `NeighborMessage` или `NeighborMessageAnswer` должны содержать следующие поля:

- адрес абонента-отправителя сообщения;
- адрес получателя сообщения;
- F_{NFA} – флаг необходимости отправки ответа `NeighborMessageAnswer`.

В сообщении `NeighborMessageAnswer` не реализуются либо всегда устанавливается значение «1»;

- количество соседей у абонента-отправителя сообщения;
- список ДХСУ абонента-отправителя;
- множество соседних узлов абонента отправителя, а также списки их ДХСУ и значений динамических характеристик каналов связи между ними и абонентом-отправителем.

Пакет NeighborMessageAnswer может быть использован для подтверждения получения NeighborMessage соседними абонентами первого порядка. Это позволяет увеличить частоту отправки сообщений NeighborMessage и более оперативно получать изменения от соседних абонентов. С другой стороны, сброс флага F_{NFA} ($F_{NFA} = 0$) позволяет существенно сократить трафик сети, особенно при большом количестве соседних абонентов.

Для хранения информации о сетевых узлах, наличия связи с соседними абонентами, информации о ДХКС и ДХСУ, маршрутов передачи трафика используется ассоциативный массив AbonentInformations (рис. 2.3), который содержит информацию о структуре сети и маршрутные правила [88]. Ключом является адрес абонента A . Значение – структура AbonentInformation. AbonentInformation содержит следующие поля:

- Direct – флаг наличия прямой связи с абонентом A ;
- LastTime – время, когда в последний раз было получено NeighborMessage или NeighborMessageAnswer от абонента A ;
- LastRouteID – последний использованный идентификатор поиска маршрута к абоненту A ;
- NeighborList – ассоциативный массив, содержащий соседних абонентов первого порядка для абонента A (поле актуально при Direct = True), а также их ДХСУ и ДХКС между ними и узлом A ;
- RouteMap – ассоциативный массив, содержащий информацию о наличии маршрутов к узлу A , которые включают адрес текущего абонента.

Ключом массива RouteMap является пара значений <SourceSearch, RouteID>. SourceSearch – адрес абонента, который инициировал поиск маршрута к A . RouteID – идентификатор поиска маршрутов. Если SourceSearch

содержит адрес текущего абонента, то он является инициатором поиска маршрута к *A*. Каждой паре $\langle \text{SourceSearch}, \text{RouteID} \rangle$ соответствует сессия поиска маршрутов. Сессия представляет историю коммуникационных запросов и ответов от момента начала поиска маршрутов к *A* до их устаревания. Структура *Session* имеет следующие поля:

- *State* – состояние сессии. Всего имеется 3 состояния: запущена процедура поиска маршрута, имеются актуальные маршруты, сессия устарела;
- *ListInput* – список входящих маршрутов к *A*;
- *ListOutput* – список исходящих маршрутов к *A*;
- *InputTmpGateway* – адрес абонента, который предшествует текущему узлу во временном маршруте;
- *OutputTmpGateway* – адрес абонента, который следует за текущим узлом во временном маршруте.

Для каждого маршрута из списков *ListInput* и *ListOutput* существует резервная копия, которая используется в алгоритмах обслуживания маршрутов связи.

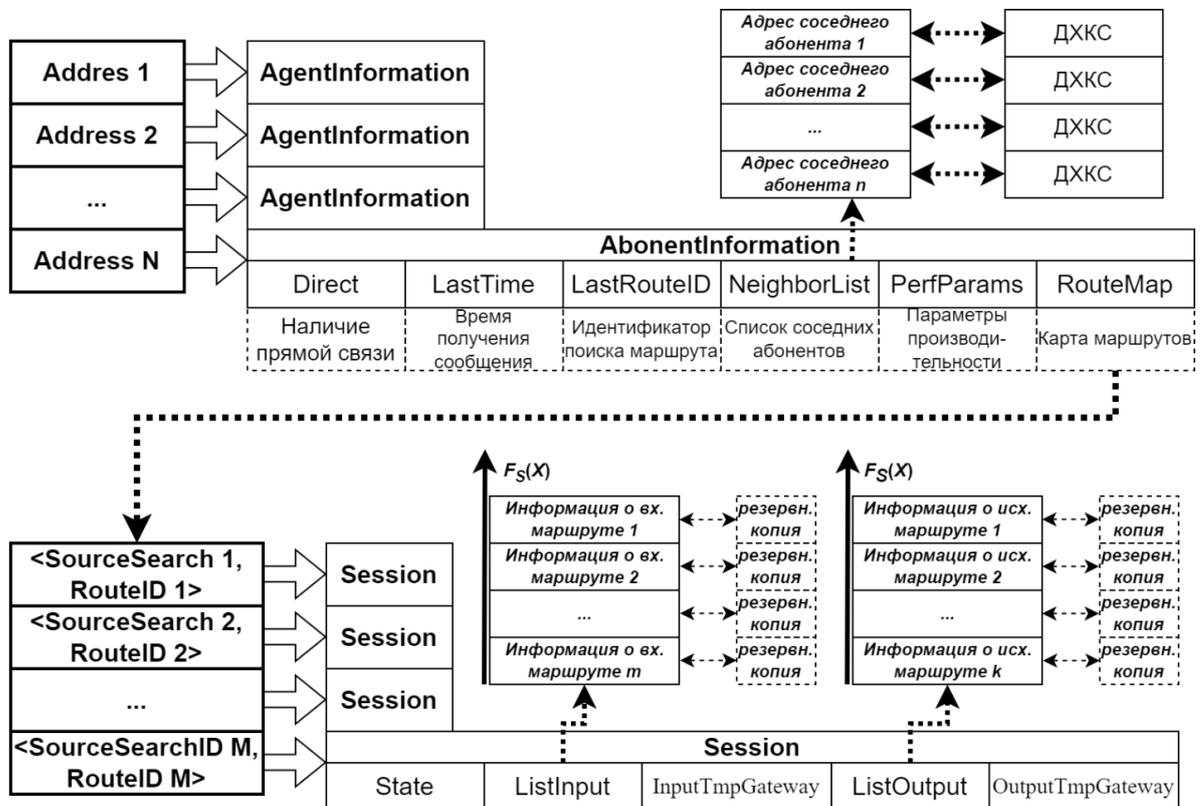


Рисунок 2.3 – Структура *AbonentInformation*

Структура AbonentInformation формируется на этапах установления соединения с сетью и обнаружения маршрутов к целевым узлам, а также после коммуникационных сессий при построении и восстановлении маршрутов. Структура содержит правила маршрутизации, которые используются для передачи и балансировки трафика по квазипараллельным маршрутам.

2.1.1. Алгоритм взаимодействия ПОСМ при установлении и дальнейшем поддержании соединения с сетью

Для реализации предложенного метода на практике был разработан специальный алгоритм (рис. 2.4). На его вход поступает текущая информация о соседних абонентах первого порядка. На выходе алгоритма формируется скорректированный граф соседних абонентов. В рамках разработанного алгоритма производится заполнение структуры AbonentInformation, которая содержит информацию о соседних абонентах и актуальных маршрутах (рис. 2.3).

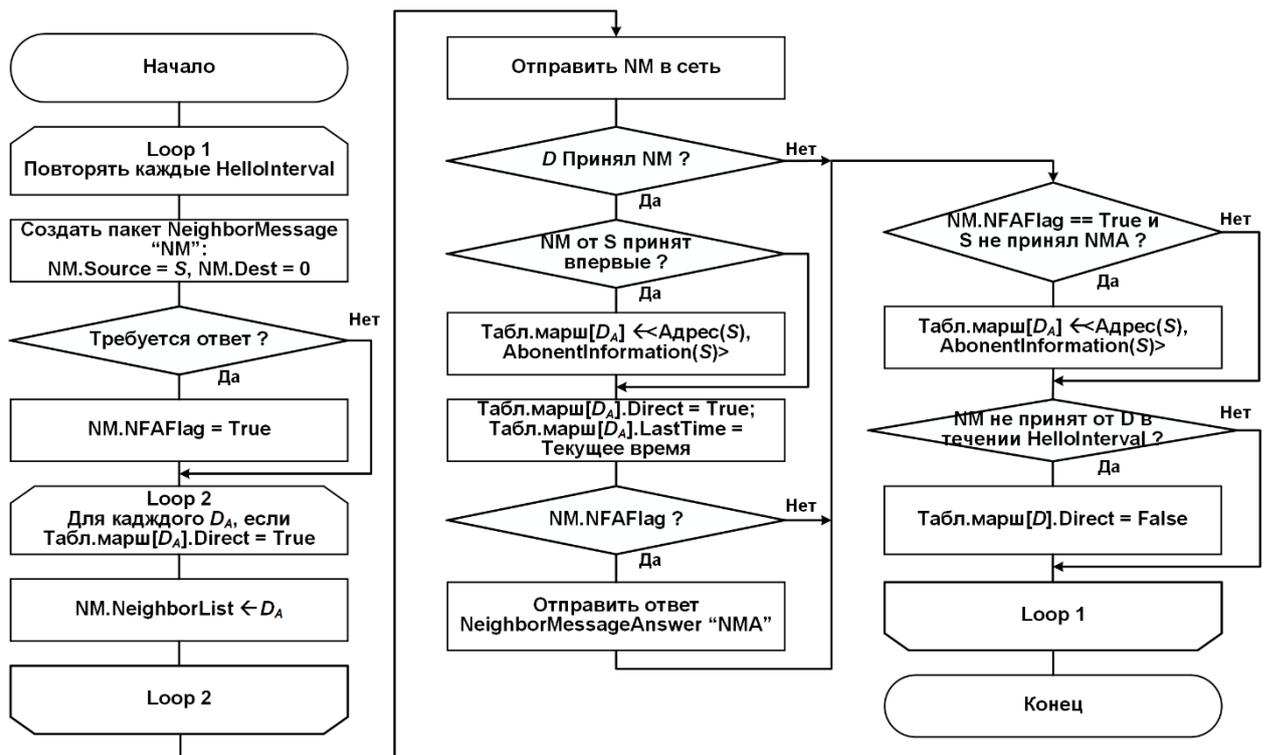


Рисунок 2.4 – Схема алгоритма взаимодействия ПОСМ

при установлении и дальнейшем поддержании соединения с сетью

Алгоритм включает в себя в себя реализацию последовательности действий:

1) Абонент S формирует сообщение NeighborMessage для информирования других абонентов сети о себе, имеющих соседей первого порядка и их ДХКС и ДХСУ, после чего отправляет пакет в сеть:

1.1) В качестве адреса отправителя устанавливает собственный адрес;
 1.2) В качестве адреса получателя устанавливает широковещательный адрес «0»;

1.3) В пакет добавляются данные о своих ДХСУ (скорость перемещения абонента, координаты относительно базовой точки, Q, D_{av}, I, R_n) и ДХСУ соседей 1-го порядка ($R_{ch}, L_{ch}, M_{Recv}, Q, D_{av}, I, R_n$). Эти показатели в дальнейшем используются для формирования частных метрик маршрутов, в состав которых входит текущий абонент;

1.4) При низкой загруженности сети может быть установлен флаг необходимости ответа F_{NFA} . Если флаг установлен, то при получении сообщения соседний абонент должен отправить ответ, используя пакет NeighborMessageAnswer, структура которого аналогична NeighborMessage;

1.5) Добавляется список абонентов, с которыми на момент отправки сообщения установлена прямая связь. Если абонент не подключён к сети, то список пуст;

2) Каждый абонент D , принявший NeighborMessage, добавляет запись в AbonentInformation об абоненте отправителе, если ранее записи не существовало, или обновляет информацию по абоненту:

2.1) В поле Direct устанавливается значение true;

2.2) В поле LastTime записывается текущее время;

2.3) В поле PerfParams записываются значения ДХСУ абонента S и ДХКС смежного с ним канала $I, L, Q, D_{av}, R_{ch}, R_n$. При этом R_{ch} обновляется на основе информации о количестве принятых пакетов M_{Recv} за последний промежуток HelloInterval. Значение M_{Recv} берётся из сообщения HelloNeighbor из поля с информацией о соседях 1-го порядка;

2.4) Обновляется поле со списком соседних абонентов;

3) Если сообщение NeighborMessage от абонента S не было принято в течение периода $2 * \text{HelloInterval}$, то в их таблицах для абонента снимается флаг о наличии прямого соединения;

4) Если в NeighborMessage установлен флаг F_{NFA} , то абонент-отправитель должен получить ответ NeighborMessageAnswer от получателя в течение интервала HelloAnswer. При этом:

4.1) Формирование сообщения производится аналогично пакету NeighborMessage;

4.2) В поле D устанавливается адрес абонента отправителя;

4.3) Для абонентов, от которых не были получены ответы, в AbonentInformation узла S снимаются флаги наличия прямого соединения с D ;

5) Абонент S периодически повторяет алгоритм через HelloInterval.

Таким образом, предложенный метод позволяет устанавливать соединение с ДСПД, обнаруживать соседних абонентов первого и второго порядков, за счёт чего формируется двухшаговый граф сети, а также осуществляется сбор их ДХСУ и ДХКС между ними. Наличие двухшагового графа позволяет производить передачу трафика соседним абонентам без запуска процедуры построения маршрута. Собранные ДХСУ и ДХКС могут быть использованы при построении маршрутов связи и их дальнейшем обслуживании.

2.2. Метод взаимодействия ПОСМ при построении квазипараллельных маршрутов связи

В качестве основы предлагаемого в работе метода взаимодействия программного обеспечения сетевой маршрутизации при определении квазипараллельных маршрутов выбран подход реактивной коммуникации. Один такой коммуникационный запрос может быть принят всеми абонентами сети ввиду «лавинообразного» распространения (flooding, [89]) и для контроля их распространения используются ограничения по количеству ретрансляций копий пакетов. При этом ПОСМ производит отслеживание образования петель в маршрутах.

Стандартные технологии маршрутизации ДСПД предполагают, что передача трафика производится по одному из множества существующих маршрутов, который предпочтительнее других по одной или нескольким метрикам. Альтернативные пути не задействуются, исключается балансировка трафика, что приводит к неравномерному его распределению по сети. Ввиду этого основной маршрут перегружается, что приводит к повышению потерь при передаче данных и увеличению временных задержек при доставке пакетов.

Для балансировки трафика по сети требуется наличие нескольких параллельных маршрутов. В существующих решениях применяется построение строго непересекающихся путей, что не всегда возможно обеспечить ввиду проблемы отсечки маршрута [90]. При этом прослеживаются случаи, когда несколько маршрутов включают общие подмаршруты, которые имеют высокую пропускную способность, сопоставимую с суммарной пропускной способностью непересекающихся частей путей. Под квазипараллельными маршрутами понимаются такие пути, для которых является допустимым, но не обязательным, наличие общих промежуточных абонентов. Пример таких маршрутов приведен на рис. 2.5.

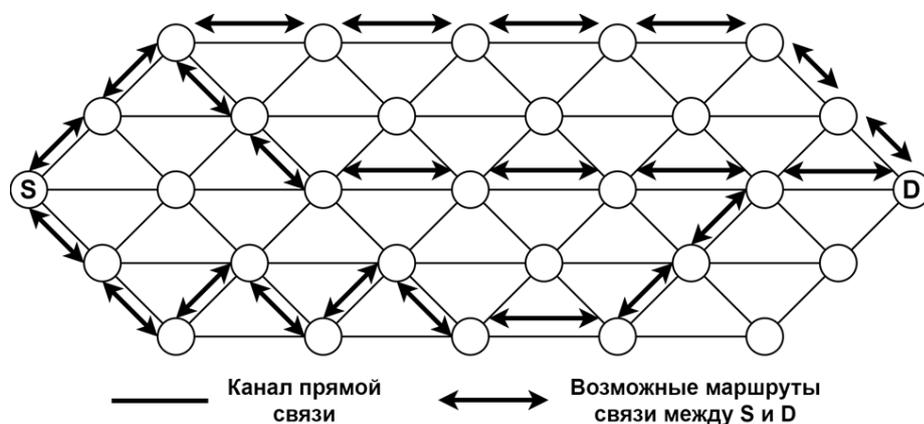


Рисунок 2.5 – Схема квазипараллельных маршрутов связи

Для построения таких путей был разработан метод взаимодействия ПОСМ при построении квазипараллельных маршрутов связи (рис. 2.6). Метод включает следующие этапы:

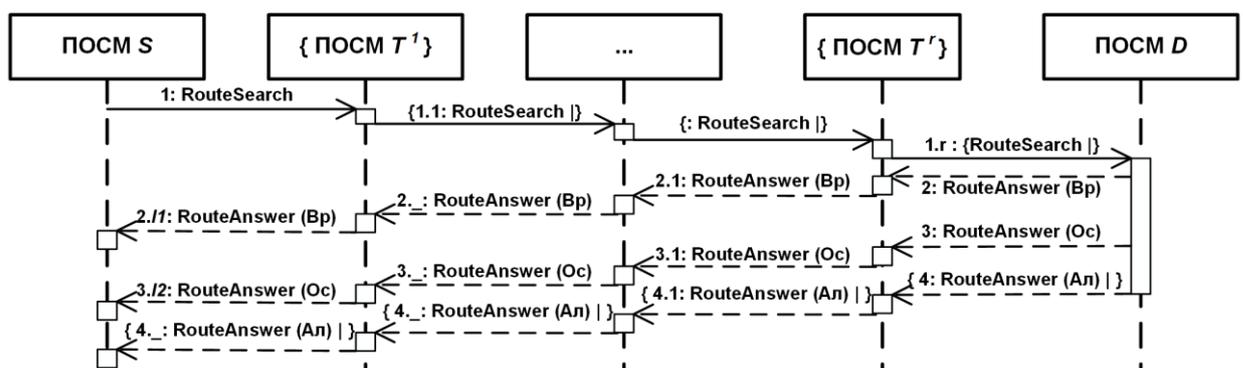
- коммуникационный запрос абонента инициатора поиска маршрута узлу назначения, в процессе которого на промежуточных узлах формируется граф переходов с приоритизацией входящих частично-сформированных маршрутов;

– коммуникационный ответ от узла назначения узлу-инициатору, содержащий временный маршрут между абонентами, который может быть использован для передачи данных до формирования основных маршрутов;

– коммуникационный ответ от узла назначения узлу-инициатору, в процессе передачи которого формируется постоянный основной маршрут между абонентами;

– широковещательный коммуникационный ответ от узла назначения узлу-инициатору, в процессе передачи которого формируются постоянные альтернативные пути передачи данных, квазипараллельные между собой и основным маршрутом.

Таким образом, метод отличается от известных тем, что для накопления и передачи частных метрик формируемых путей выполняется единый реактивный запрос от узла-инициатора с последующей цепочкой информационных ответов от промежуточных узлов, используемых для обнаружения основного маршрута, которая совмещается с широковещательными рассылками, применяемыми для построения альтернативных путей передачи данных. Это позволит обеспечивать формирование квазипараллельных маршрутов, позволяет снизить сквозную задержку передачи данных и повысить значение коэффициента доставки пакетов.



$\{ \text{ПОСМ } T^i \}$ - множество ПОСМ сетевой промежуточных абонентов между S и D, $|\{ \text{ПОСМ } T^i \}| \geq 0, i \neq j$
 $\{ \text{RouteMess} \}$ - множество сообщений типа RouteMess, пересылаемые между 2-я множествами узлов
 (Bp), (Oc), (Al) - флаги временного, основного и альтернативных маршрутов

Рисунок 2.6 – Схема метода взаимодействия ПОСМ при построении квазипараллельных маршрутов связи

Коммуникационный запрос представляет первый этап взаимодействия сетевого ПО. ПОСМ абонента-инициатора S построения маршрутов к D

выполняет коммуникационный запрос, начиная с рассылки сообщения RouteSearch соседним абонентам. В соответствующем пакете производится накопление адресов промежуточных узлов между S и D , которые образуют формируемый маршрут, также там сохраняются его частные метрики X . При получении запроса ПО маршрутизации промежуточных абонентов T добавляют в пакет собственный адрес и обновляют его частные пути. Каждый узел T , впервые принявший запрос, ретранслирует его соседним абонентам. RouteSearch сохраняются в список запросов для пар $\langle S, D \rangle$ и сортируются по показателю $F_S(X)$. $F_S(X)$ представляет обобщённую маршрутную метрику, полученную путём комплексирования его частных критериев. Рассылка прекращается при достижении сообщением абонента D , что завершает первый этап взаимодействия ПОСМ, в результате которого должен быть сформирован ациклический граф переходов (рис. 2.7) от S к D . В итоге, каждый промежуточный абонент имеет несколько частично сформированных маршрутов связи к S .

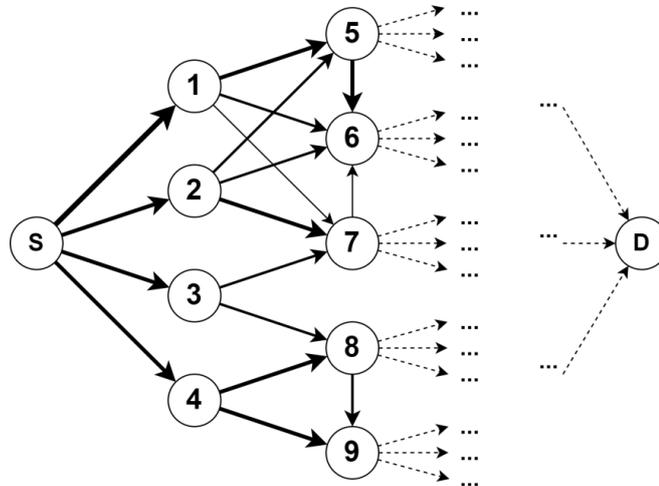


Рисунок 2.7 – Ациклический граф переходов, формируемый на первом шаге метода взаимодействия ПОСМ при построении маршрутов

Коммуникационные ответы узлу-инициатору при построении маршрутов используются на втором этапе метода взаимодействия ПОСМ. Производится обмен пакетами RouteAnswer для построения основного и альтернативных маршрутов между узлами S и D . После получения первого пакета RouteSearch ПОСМ абонента D извлекает из него маршрут, помечает

временным и возвращает его абоненту S . Временный маршрут может быть использован для передачи данных до того, как будут сформированы основной и альтернативные пути. Через заданный промежуток времени сетевое ПО абонента D отправляет RouteAnswer узлу T' , с которого был принят лучший частично-сформированный маршрут по метрике $F_S(X)$. Сетевое ПО каждого из последующих абонентов T' повторяет процесс отправки, пока RouteAnswer не достигнет узла S . Через определённый интервал времени ПО абонента D отправляет широковещательные пакеты RouteAnswer узлу S , при достижении которого в пакетах оказываются сформированными альтернативные маршруты, которые могут частично пересекаться с основным (рис. 2.8).

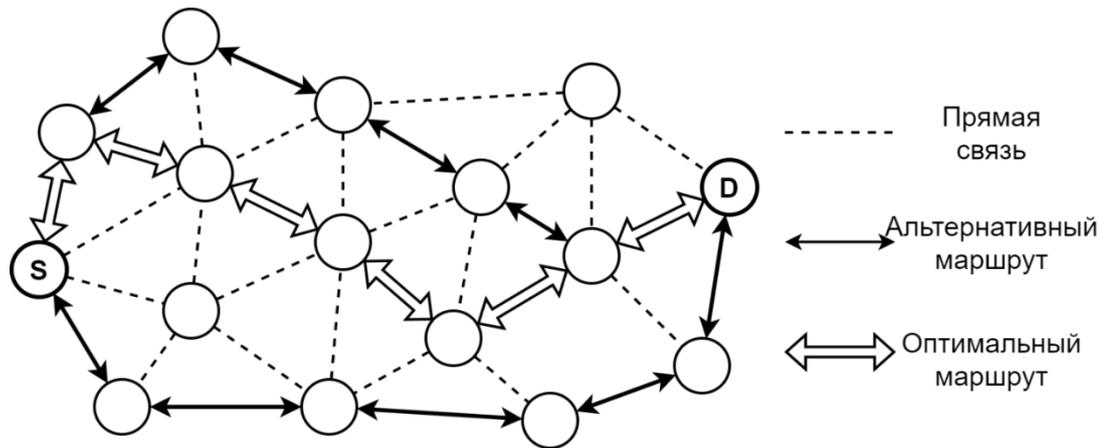


Рисунок 2.8 – Пример маршрутов связи, построенных на втором этапе взаимодействия ПОСМ

Для оценки пригодности частично или полностью сформированного маршрута для передачи трафика по сети, а также их ранжирования используется функция оценки $F_S(X)$. Маршрут R_1 предпочтительнее R_2 , если $F_S(X_{R1}) > F_S(X_{R2})$. При этом маршрут R_1 является пригодным для маршрутизации трафика, если $F_S(X) > F_S'$, где F_S' – некоторое пороговое значение.

Производить оценку маршрутов связи можно на основе множества частных показателей. Маршрут состоит из узлов и соединяющих их каналов. Эффективность маршрутизации зависит от того, насколько каждый абонент или канал связи эффективно перемещает пакет в процессе передачи данных к узлу получателю. На основе анализа, проведённого в разделе 1.4, были выбраны следующие метрики для оценки маршрутов связи:

- N – длина маршрута;
- I_{all}, I_{max} – общее и максимальное внутри узла количество пересечений маршрутов между другими узлами;
- L_{av}, L_{max} – средняя и максимальная загрузки каналов связи;
- Q_{av}, Q_{max} – средний и максимальный размеры очереди пакетов на транзитных абонентах;
- D_{av}, D_{max} – средняя и максимальная задержки пакета на узлах маршрута;
- R_{ch} – доля данных, которые не теряются при прохождении пакетов через каналы беспроводной связи;
- R_n – доля данных, которые не теряются при прохождении пакетов через абонентов сети.

Метрики маршрутов образуют его признаковое описание X . Используя этот вектор, можно выдать оценку о степени применимости текущего пути для маршрутизации трафика.

2.2.1. Метод оценки маршрутов передачи данных

В процессе взаимодействия сетевого программного обеспечения требовалось производиться динамическую фильтрацию и упорядочивания информационных сообщений, содержащих формируемые маршруты. Для этого был предложен новый метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия ПОСМ, на основе частных путевых метрик X , позволяющий снизить суммарный объем служебных сообщений при формировании маршрутов, и таким образом, повысить эффективность передачи данных. Метод включает последовательное выполнение следующих этапов:

1. При обработке информационных сообщений (RouteSearch, RouteAnswer) производится обновление содержащихся в них частных маршрутных метрик X на основе ДХСУ принявшего их абонента и ДХКС ассоциированного канала. Среди используемых частных критериев можно выделить среднюю и максимальную загрузки каналов связи, средний и максимальный размеры очереди пакетов на транзитных абонентах, среднюю и максимальную задержки

пакета при прохождении через узел маршрута, коэффициент доставки пакетов на каналах связи.

2. В качестве оцениваемой маршрутной метрики выбрана взвешенная мультипликативная свёртка значений коэффициента доставки данных (packed delivered ratio, PDR) и сквозной задержки передачи пакета (end-to-end delay, E2ED), делённой на нормировочный коэффициент $E2ED_{norm}$, представляющий оценочное значение максимального времени доставки пакета:

$$F_S = \frac{PDR^{K_{S1}}}{\left(\frac{E2ED}{E2ED_{norm}} \right)^{K_{S2}}} . \quad (2.1)$$

где K_{S1} и K_{S2} – значения весовых коэффициентов, устанавливаемых на этапе сбора выборки для обучения моделей в зависимости от того, какой приоритет оператор сети отдаёт показателям PDR и E2ED. Исходя из 2.1, маршрут R_1 предпочтительнее R_2 , если $F_S^{R1} > F_S^{R2}$. При этом маршрут R_1 является пригодным для маршрутизации трафика, если $F_S^{R1} > F_S'$, где F_S' – некоторое пороговое значение. Для минимизации ошибки прогнозирования производится вычисление не отдельных составляющих свёртки, а её непосредственное значение F_S . Пусть имеется обучающий набор $\{X_i, y_i\}_{i=1}^n$, где X_i и y_i – вектор частных метрик и истинное значение прогнозируемой переменной соответственно, n – размер обучающей выборки.

Задача подбора аналитических зависимостей рассмотренных показателей от заданного набора метрик является многокритериальной с большим числом параметров настройки, поэтому для ее решения целесообразно обратиться к алгоритмам машинного обучения, которые позволят более точно учитывать параметры и характеристики конкретной сети связи и процессы передачи данных в ней. В нашем случае F_S является целевым параметром, который требуется вычислить при помощи алгоритмов машинного обучения.

Признаковое описание маршрута X содержит неоднородные показатели. Необходимо применение алгоритмов, которые хорошо работали бы на неоднородных данных. Для оценки качества маршрутов связи в текущей работе

используются алгоритмы градиентного бустинга деревьев принятия решений [91] и обобщённой регрессионной нейронной сети [92]. В дальнейшем (глава 4) производится сравнение их эффективности по отношению к текущей задаче.

Градиентный бустинг (ГБ) над деревьями принятия решений представляет способ построения композиции решающих деревьев. Деревья строятся последовательно, и каждое последующее дерево стремится скомпенсировать ошибку уже построенных. Среди преимуществ градиентного бустинга можно выделить способность осуществлять работу с произвольными функциями потерь в задачах регрессии, относительно низкая подверженность переобучению, достаточная точность восстановления искомым функций, отсутствие необходимости нормирования входных данных. Для решения задачи прогнозирования вещественного значения производится построение композиции из k базовых функций $f_i(X)$ деревьев принятия решений:

$$F_k(X) = f_1(X) + f_2(X) + \dots + f_k(X). \quad (2.2)$$

где X – набор частных маршрутных метрик, $F_k(X)$ – композиция из k -функций деревьев принятия решений.

Каждая из последующих j -ых функций предназначена для компенсации ошибки $j-1$ предыдущих:

$$f_j(X) = \arg \min L(y, F_{j-1}(X)), \quad (2.3)$$

$$L(y, y') = \frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2, \quad (2.4)$$

$$F_{j-1}(X) = f_1(X) + f_2(X) + \dots + f_{j-1}(X). \quad (2.5)$$

где $L(y, y')$ – функция ошибки прогнозирования, y и y' – вектора истинных и спрогнозированных значений целевой переменной, $F_{j-1}(X)$ – композиция из $j-1$ функций. Тогда функция оценки маршрута связи $F_S(X)$ является моделью $F_k(X)$ из k деревьев принятия решений.

Обобщенная регрессионная нейронная сеть (ОРНС) является частным случаем радиально-базисных нейронных сетей [93]. Её структура

предполагает наличие двух промежуточных слоёв. Первый промежуточный слой сети ОРНС состоит из радиальных элементов. Второй промежуточный слой содержит элементы, которые помогают оценить взвешенное среднее. Архитектура сети имеет следующую математическую модель (2.6):

$$Y(X) = \frac{\sum_i^N y_i G(X, X_i')}{\sum_i^N G(X, X_i')}, \quad G(X, X_i') = \exp\left(\frac{(X - X_i')^T (X - X_i')}{2\sigma^2}\right), \quad (2.6)$$

где $G(X, X_i)$ – гауссово ядро, X_i, y_i – входы и выходы обучающей выборки. ОРНС копирует обучающую выборку и использует её для оценки отклика в произвольной точке. Выходное значение представляет взвешенное среднее выходов по обучающейся выборке. Тогда функция оценки маршрута связи $F_S(X)$ является моделью $Y(X)$ ОРНС обладает такими преимуществами, как отсутствие потребности в методе обратного распространения ошибки, обучение за один проход, высокая точность оценки, способность обрабатывать шумы на входных данных.

Рассмотрим процесс обучения моделей градиентного бустинга и ОРНС для функции $F_S(X)$ с коэффициентами $K_{S1} = 0,7, K_{S2} = 0,3$. В ходе работы при помощи имитационной модели на базе разработанной программной платформы был собран набор данных. В наборе отсутствуют категориальные признаки [94], что упрощает решение задачи. Для стабильной работы алгоритма ОРНС производится нормирование признаков по формуле (2.7):

$$X'_j = \frac{X_j - \mu_j}{\sigma_j}. \quad (2.7)$$

Параметры моделирования представлены в табл. 2.1. Каждая запись в наборе представляет массив частных метрик X качества маршрута связи, полученных в процессе его построения и является признаковым описанием маршрута. Каждая запись содержит метрику $F(X)$, которая рассчитана на основе значений PDR и E2ED, вычисленных с применением статистики по передаче данных по маршруту связи.

Для оценки возможности построить качественные модели машинного обучения был произведён корреляционный анализ между признаковым описанием маршрута связи и F_S . Результаты представлены на рис. 2.9. F_S имеет выраженную статистическую взаимосвязь с I_{all} , L_{av} , Q_{av} , D_{av} , R_{ch} , R_n . Результаты корреляционного анализа демонстрируют возможность построить качественную модель для оценки F_S . Подробное исследование и сравнение алгоритмов будет представлено в главе 4.

Таблица 2.1 – Параметры модели ДСПД при сборе данных

Параметр	Значение
Максимальная длина маршрута	5, 6, 7, ..., 19, 20
Скорость передачи данных	8 Кб/с.
Трафик между абонентами	2 Кб/с.
Максимальная скорость движения абонентов	15 м/с.
Модель движения абонентов	Модель случайных перемещений на основе случайных точек
Радиус радиопередачи	400 м.
Размер карты	1500 x 1500 м ²
Ограничение по виртуальному времени моделирования	1200 с.

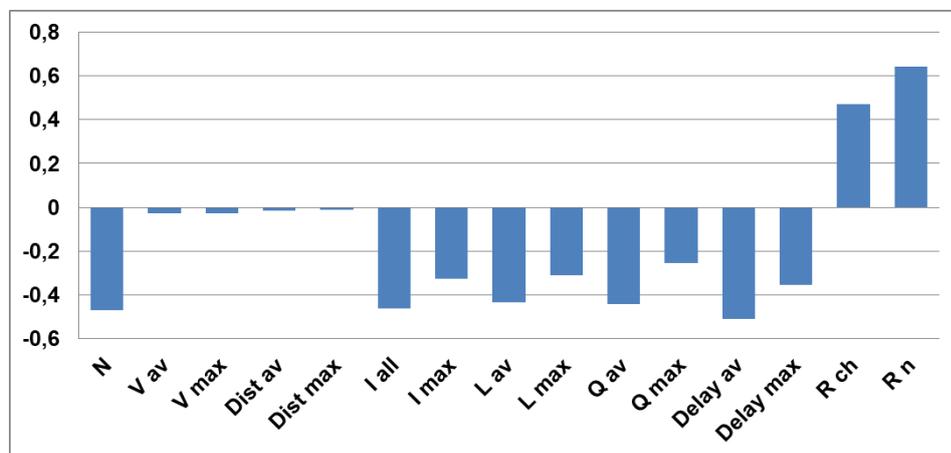


Рисунок 2.9 – Корреляционный анализ между признаками и F_S

3. Прерывание обработки сообщений производится в случае предельного значения обобщённой маршрутной метрики F_S' : $F_S < F_S'$. Значение F_S' определяется как свёртка предельно допустимых значений PDR (PDR') и E2ED (E2ED'):

$$F_S' = \frac{(\text{PDR}')^{K_{S1}}}{\left(\frac{\text{E2ED}'}{\text{E2ED}_{norm}} \right)^{K_{S2}}} . \quad (2.8)$$

Таким образом, метод позволяет осуществлять динамической фильтрацию и упорядочивание информационных сообщений на промежуточных узлах сети, что позволяет повысить эффективность функционирования ДСПД за счет снижения суммарного объема служебного трафика между программным обеспечением сетевой маршрутизации отдельных абонентов.

2.2.2. Алгоритм взаимодействия ПОСМ при запросе маршрута

Алгоритм взаимодействия ПОСМ при запросе маршрута инициализируется, когда отсутствует актуальный исправный маршрут к целевому узлу D . На вход алгоритма поступает двухшаговый путь к соседним абонентам. В процессе функционирования алгоритма производится накопление актуальной информации о функционировании узлов сети и каналов связи между ними, а также формируется децентрализованный граф переходов сетевых пакетов между абонентами S и D . Алгоритм начинает свою работу с коммуникационного запроса `RouteSearch`, который широковещательно рассылается абонентам первого порядка. Сообщение содержит следующие поля:

- RS_{ID} – идентификатор пакета поиска маршрута. Построенный в процессе функционирования текущей процедуры поиска маршрут будет идентифицироваться RS_{ID} и адресом узла-инициатора;
- адрес абонента-инициатора поиска $Source$;
- адрес абонента-получателя $Dest$;
- X – признаковое описание маршрута – частные маршрутные метрики;
- $RList$ – список транзитных абонентов, которые были посещены пакетом поиска маршрута в процессе широковещательной рассылки;
- $HopCount$ – количество переходов, выполненных сообщением.

В процессе работы алгоритма определяются возможные направления передачи трафика между промежуточными абонентами, а также анализируются

ДХКС и ДХСУ входящих в формируемый маршрут узлов и каналов связи между ними. Алгоритм содержит следующие шаги:

1) ПОСМ абонента S создаёт пакет $RouteSearch$ и производит его широковещательную рассылку соседним абонентам первого порядка. Перед этим пакету назначается уникальный идентификатор RS_{ID} , при помощи которого в дальнейшем будут идентифицироваться квазипараллельные пути, сформированные в результате функционирования текущей сессии построения маршрутов.

2) При получении запроса $RouteSearch$ промежуточный узел T :

2.1) Проверяет значение поля $HopCount$. Если его значение стало равно предельно допустимому ($HopCount = HopCountMax$), пакет уничтожается. $HopCountMax$ – максимально допустимое количество переходов при рассылке пакета $RouteSearch$;

2.2) Проверяет наличие собственного адреса в списке транзитных абонентов $RList$. Наличие собственного адреса в $RList$ свидетельствует о том, что в формируемом маршруте образовалась петля. Петля формируется, когда узел повторно принял пакет $RouteSearch$ с конкретным значением RS_{ID} , и в списке $RList$ уже имеется его адрес (рис. 2.10). В случае обнаружения в пакете петли он уничтожается;

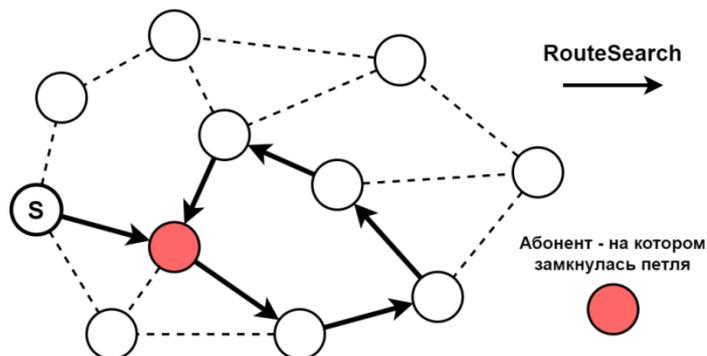


Рисунок 2.10 – Петля в маршруте

2.3) Производит обновление метрики X с учётом динамических характеристик узла и динамических характеристик связывающего его и предыдущего в маршруте абонента канала связи;

2.4) Если маршрут не удовлетворяет условию $F_S(X) \geq F_S'$, то пакет уничтожается. Это позволяет прерывать формирование маршрутов связи с их низкой оценочной пригодностью для передачи данных;

2.5) Запоминает информацию о формируемом маршруте путём добавления копии пакета в список INPUT ассоциативного массива $RouteItem[RS_{ID}, S]$, принадлежащего записи таблицы $AbonentInformation$ s, соответствующей абоненту D . В списке INPUT пакеты отсортированы в порядке убывания пакетов по значению функции $F_S(X)$;

2.6) Если пакет с текущими значениями RS_{ID} и S был получен впервые, сетевое ПО узла T производит его широковещательную ретрансляцию своим соседним абонентам. При этом в передаваемом пакете $RouteSearch$ значение поля $HopCount$ увеличивается на 1, а в поле $RList$ добавляется адрес абонента T ;

3) При получении запроса $RouteSearch$ абонент D :

3.1) Если пакет с содержащимися в нём значениями полей RS_{ID} и S был получен впервые, производится создание коммуникационного ответа в форме пакета $RouteAnswer$. В $RouteAnswer$ устанавливается флаг временного маршрута F_{tmp} . Сформированный в списке $RList$ пакета $RouteSearch$ маршрут записывается в соответствующее поле пакета $RouteAnswer$, по которому он отправляется абоненту S . Сформированный путь представляет временный маршрут, который может быть использован при передаче данных между S и D , до формирования постоянных путей;

3.2) В течение временного периода $TIME_RECV_WAIT$ узел продолжает приём пакетов $RouteSearch$ с аналогичными значениями полей RS_{ID} и S . По окончании периода этап прямого хода считается завершённым и запускается процедура возврата маршрутов связи. Все поступившие после сообщения $RouteSearch$ уничтожаются.

Результатом работы алгоритма запроса маршрута является распределённый ориентированный ациклический граф (рис. 2.7). Входящими дугами обозначены пакеты $RouteSearch$, содержащие частично сформированные маршруты. На рисунке более толстыми входящими дугами обозначены более качественные частично сформированные маршруты по значению функции $F_S(X)$. Схема алгоритма взаимодействия ПОСМ при запросе маршрута представлена на рис. 2.11.

В результате взаимодействия ПОСМ узлов по описанному алгоритму на каждом транзитном абоненте хранятся несколько нисходящих ветвей частично сформированных маршрутов от абонента источника. Маршруты отсортированы в порядке убывания $F_S(X)$. В RouteAnswer с флагом F_{tmp} содержит временный маршрут связи. Он может быть использован для передачи данных между абонентами S и D , пока не будет сформированным хотя бы один постоянный маршрут.

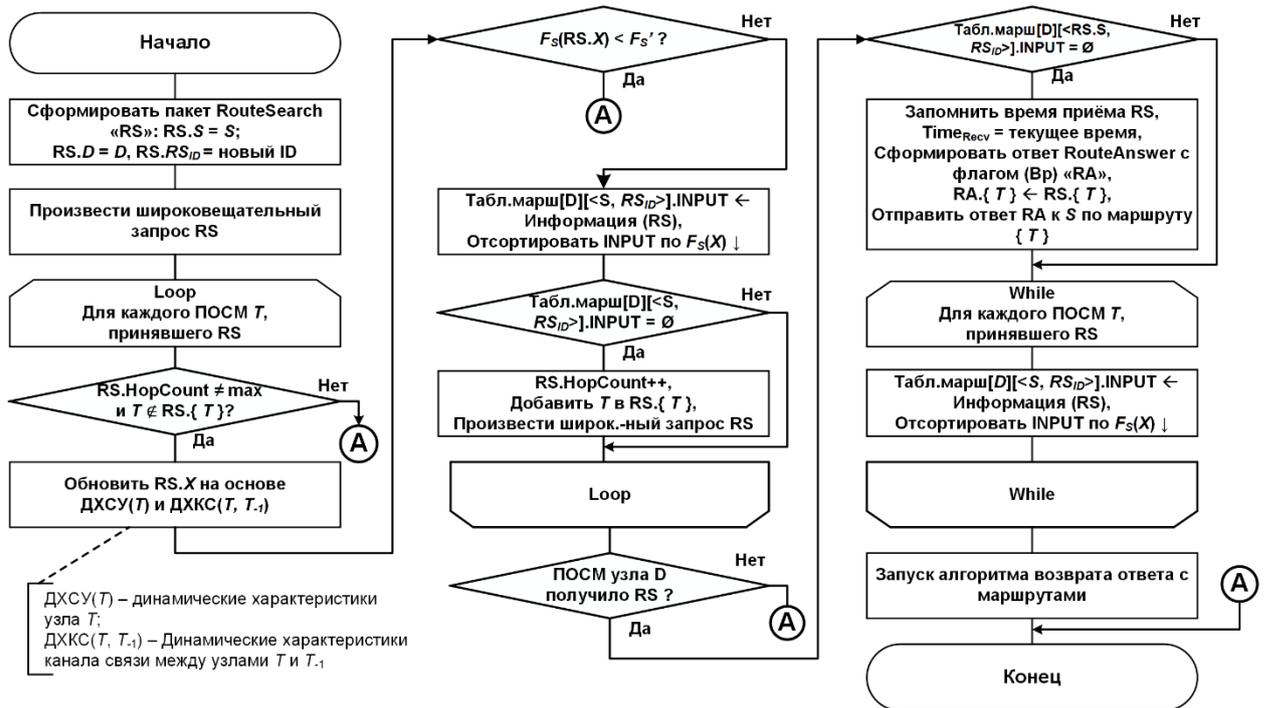


Рисунок 2.11 – Алгоритм взаимодействия ПОСМ при запросе маршрута

2.2.3. Алгоритм взаимодействия ПОСМ при возврате маршрутов связи

Для практической реализации второго этапа метода взаимодействия ПОСМ при построении квазипараллельных путей разработан алгоритм возврата многопутевого маршрута абоненту-источнику. Результатами работы алгоритма являются сформированные основной и альтернативный маршруты связи. Основным является маршрут, который предпочтительнее остальных по значению функции $F_S(X)$. Альтернативные маршруты уступают основному по значению $F_S(X)$, могут частично с ним пересекаться.

При построении путей посредством алгоритма возврата квазипараллельных маршрутов абоненту-источнику используется коммуникационный ответ

в виде сообщения RouteAnswer. Он может быть передан в адресном или широковещательном режимах в зависимости от типа создаваемого маршрута. Адресная передача используется при построении оптимального маршрута между абонентами S и D . Широковещательная рассылка применяется при определении альтернативных маршрутов.

Пакет RouteAnswer включает поля запроса RouteSearch, а также:

- T – абонент, которому в адресном режиме передаётся RouteAnswer;
- F_m – флаг, который устанавливается, если пакет был передан абонентом основного маршрута;
- F_b – флаг, который устанавливается при широковещательной рассылке RouteAnswer;
- c_m – количество абонентов, с которыми имеет пересечение альтернативный маршрут.

Алгоритм содержит следующие шаги:

1) Абонент D формирует пакет RouteAnswer. В поле T записывается адрес абонента, с которого был принят лучший маршрут по показателю $F_S(X)$, а также устанавливается флаг F_m основного маршрута. Производится адресная отправка пакета на узел T ;

2) Через время TIME_SEND_WAIT абонент D производит широковещательную рассылку пакета RouteAnswer с установленным флагом F_b ;

3) Если абонент T принял адресный пакет RouteAnswer:

3.1) Абонент T проверяет наличие петли в $RList$. При её обнаружении производится удаление частично сформированного маршрута, поступившего от абонента T , из списка INPUT таблицы AbonentInformation. Далее пакет ретранслируется по следующему маршруту, взятому из INPUT;

3.2) Абонент T запоминает адрес узла, с которого был принят пакет RouteAnswer. Данный абонент становится следующим транзитным узлом при передаче трафика между S и D ;

3.3) Абонент T добавляется свой адрес в $RList$ пакета RouteAnswer;

3.4) ПОСМ абонента T производит рассылку пакета RouteAnswer аналогично п.1 алгоритма. При пересылке в пакетах устанавливается флаг оптимального маршрута;

3.5) При достижении пакета RouteAnswer с флагом F_m абонента S основной маршрут считается сформированным. Он запоминается абонентом S и может в дальнейшем использоваться для маршрутизации трафика. Временный маршрут удаляется;

4) Если промежуточный абонент принял адресный пакет RouteAnswer от узла основного маршрута, то он запоминает отправителя как абонента, который входит в состав основного маршрута;

5) Если промежуточный абонент принял широковещательный пакет RouteAnswer от узла основного маршрута, то он запускает широковещательную процедуру поиска маршрута обратно к S ;

6) На каждом шаге проверяется, соответствует ли формируемый маршрут минимально допустимому значению функции $F_S(X)$ и не пересекается ли он с основным маршрутом более чем на $(100 * K_{NS}) \%$. Когда одно из условий не выполняется, рассылка сообщения прерывается;

7) Если абонент, не входящий в основной маршрут, принимает повторно широковещательный пакет RouteAnswer, то широковещательная рассылка прерывается;

8) Альтернативный маршрут считается сформированным, если достигает узла S или узла основного маршрута, при этом альтернативный маршрут не должен пересекаться с основным маршрутом более чем на $(100 * K_{NS}) \%$;

9) Если альтернативный маршрут окончательно сформирован на абоненте основного маршрута, то есть альтернативный маршрут имеет пересечение с оптимальным маршрутом, то текущий абонент производится отправку пакета RouteAnswer по оптимальному маршруту.

Схема алгоритма представлен на рис. 2.12. Также в алгоритме используются параметры ACTIVE_ROUTE_TIME и ROUTE_SEARCH_TIME. ACTIVE_ROUTE_TIME – временной промежуток, в течение которого найденные

маршруты считаются актуальными. ROUTE_SEARCH_TIME – временной период, за который абонент-инициатор поиска маршрута должен получить ответный пакет RouteAnswer. В ином случае принимается, что маршрут к целевому абоненту отсутствует.

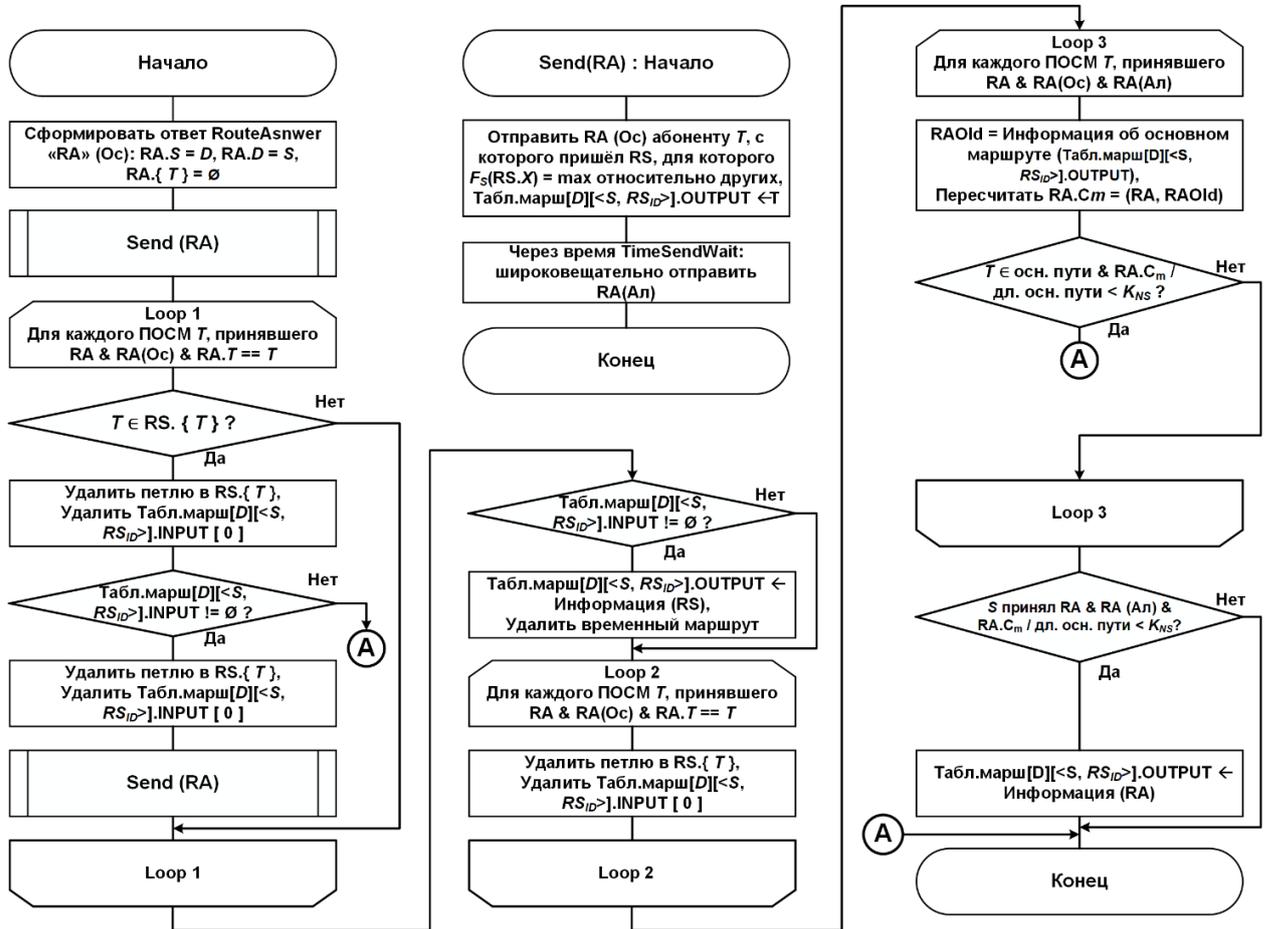


Рисунок 2.12 – Алгоритм взаимодействия ПОСМ

при возврате маршрутов связи

Если альтернативный маршрут окончательно сформирован на абоненте основного маршрута и альтернативный маршрут имеет пересечение с оптимальным маршрутом, то абонент производит отправку пакета RouteAnswer по оптимальному маршруту.

Результатом функционирования алгоритмов является построение оптимального и нескольких альтернативных маршрутов между абонентами S и D . Для передачи уведомлений о доставке сообщений используется оптимальный маршрут. Временная диаграмма построения маршрутов связи представлена на рис. 2.13.

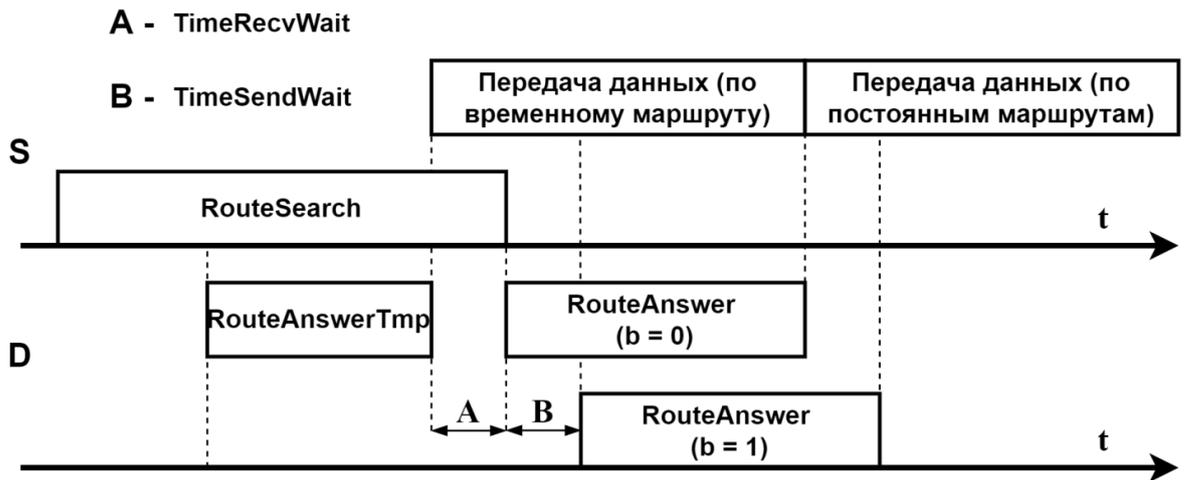


Рисунок 2.13 – Временная диаграмма рассылки сетевых пакетов

Таким образом, предложенный метод взаимодействия сетевого ПО при построении квазипараллельных маршрутов включает алгоритмы запроса маршрута и возврата маршрута абоненту-инициатору, оценку маршрутов связи $F_S(X)$, а также сессию поиска маршрутов, хранимую в таблице `AbonentInformation`. По сформированным квазипараллельным маршрутам может быть передан трафик, для которого может быть осуществлена балансировка.

2.3. Метод взаимодействия ПОСМ при восстановлении маршрутов

Восстановление маршрутов связи представляет корректировку путей передачи трафика при возникновении разрывов на отдельных каналах связи между узлами [95-98]. Каждый узел хранит информацию о динамических характеристиках соседних абонентов первого и второго порядков и каналов связи между ними. За счёт этого, ПОСМ узла, который обнаружил неисправность, может инициировать восстановление маршрута за счёт включения, исключения или замены узлов в пути.

Разрыв соединения между узлами маршрута определяется, когда ПОСМ не принимает сообщений `NeighborMessage` и `NeighborMessageAnswer` от соседнего узла в маршруте в течение заданного интервала. Ухудшение частных критериев эффективности передачи данных по пути выявляется после обновления его маршрутных метрик на основе информации из сообщений поддержания связи от соседних абонентов, принадлежащих маршруту. Это может

наблюдаться при резком увеличении загруженности абонента и канала связи трафиком, что оказывает критическое влияние на динамические характеристики узлов и каналов между ними в маршруте при передаче данных по нему. В этом случае показатель $F_S(X)$ маршрута оказывается ниже критического значения: $F_S(X) < F_S'$.

В диссертации предлагается метод взаимодействия ПОСМ при восстановлении маршрутов, который позволяет осуществить несколько вариантов корректировки пути (рис. 2.14–2.18), включающий сокращение, сохранение и увеличение длины маршрута. В варианте, продемонстрированном на рис. 2.14, маршрут можно сократить, исключив абонента N_{+1} , если установлено соединение с следующим абонентом маршрута N_{+2} . В случае, показанном на рис. 2.15 маршрут сокращается за счёт исключения из маршрута текущего абонента. В третьем варианте (рис. 2.16) связь со следующим абонентом N_{+1} была потеряна, но ее можно восстановить, если N_{+1} является соседним абонентом второго порядка через N_A . В других случаях (рис. 2.17-2.18) производится замена транзитного абонента с сохранением длины маршрута.

Предлагаемый метод содержит следующие шаги:

1. ПОСМ текущего абонента N , которая произвела обнаружение разрыва связи со следующим абонентом N_{+1} в маршруте, проверяет, является ли текущий абонент и N_{+2} соседними. При положительной проверке производится коррекция маршрута путём удаления из него абонента N_{+1} (рис. 2.14). В ином случае производится переход к шагу 2.

2. ПОСМ текущего абонента N , которая произвела обнаружение разрыва связи со следующим абонентом N_{+1} в маршруте, проверяет, является ли текущий абонент и N_{+2} соседними узлами второго порядка. При положительной проверке производится коррекция маршрута путём замены в маршруте абонента N_{+1} на T (рис. 2.15). В ином случае производится переход к пункту 3.

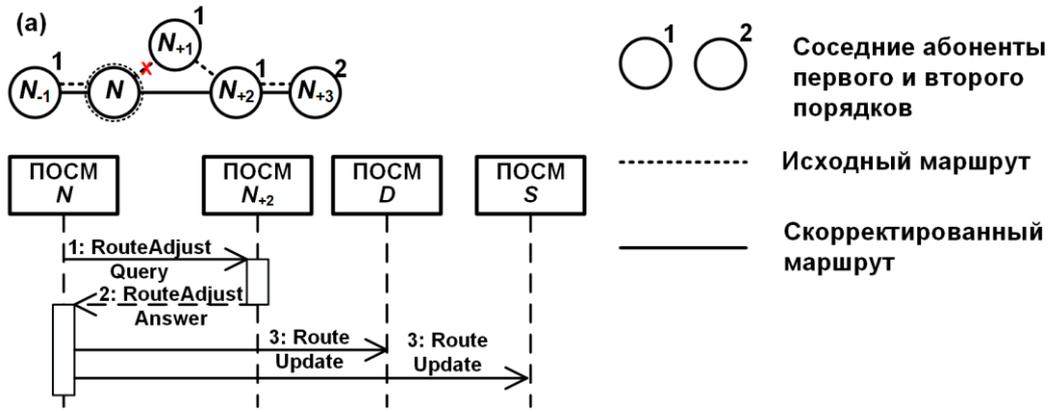


Рисунок 2.14 – Схема метода взаимодействия ПОСМ при восстановлении маршрутов: вариант (а) восстановления маршрутов

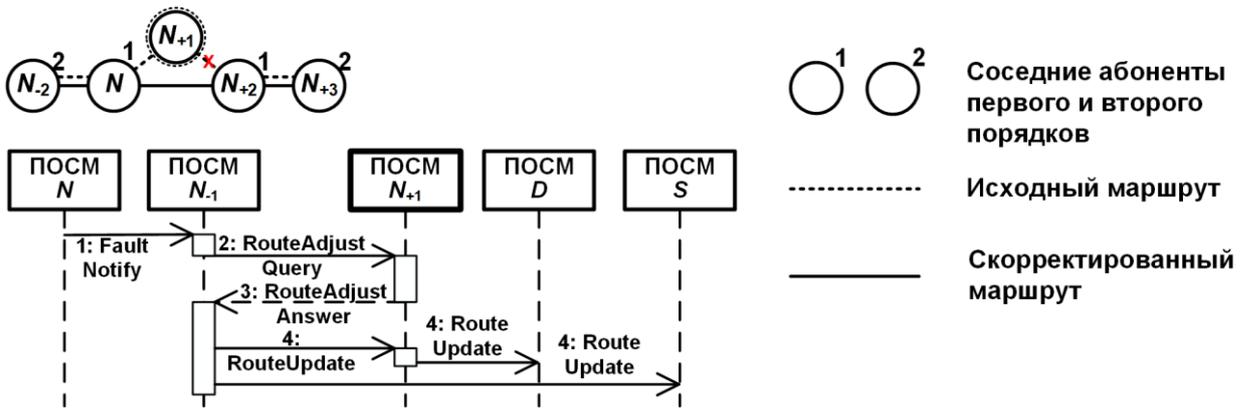


Рисунок 2.15 – Схема метода взаимодействия ПОСМ при восстановлении маршрутов: вариант (б) восстановления маршрутов

3. ПОСМ текущего абонента N , которая произвела обнаружение разрыва связи со следующим в маршруте абонентом N_{+1} , проверяет, является ли N_{+1} для N соседним абонентом второго порядка транзитивно через T . При положительной проверке производится корректировка маршрута путём включения в него абонента T (рис. 2.17). В ином случае производится переход к шагу 4.

4. ПОСМ отправляет запрос восстановления маршрута предыдущему абоненту маршрута N_{-1} . Приняв запрос, абонент N производит попытку восстановления маршрута в соответствии с шагами 1 и 2. Если маршрут не удалось восстановить, производится уведомление абонента-инициатора построения маршрута S о сбое пути.

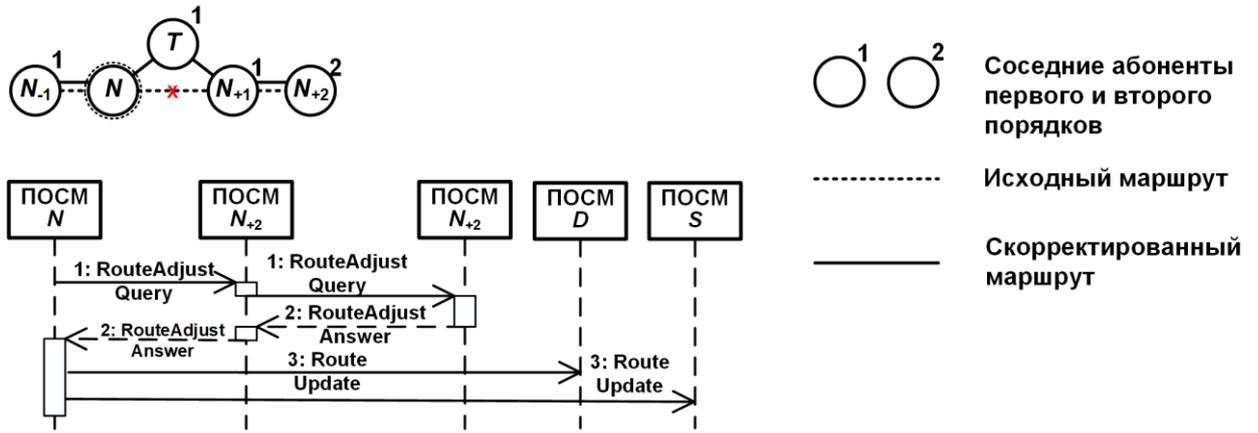


Рисунок 2.16 – Схема метода взаимодействия ПОСМ при восстановлении маршрутов: вариант (в) восстановления маршрутов

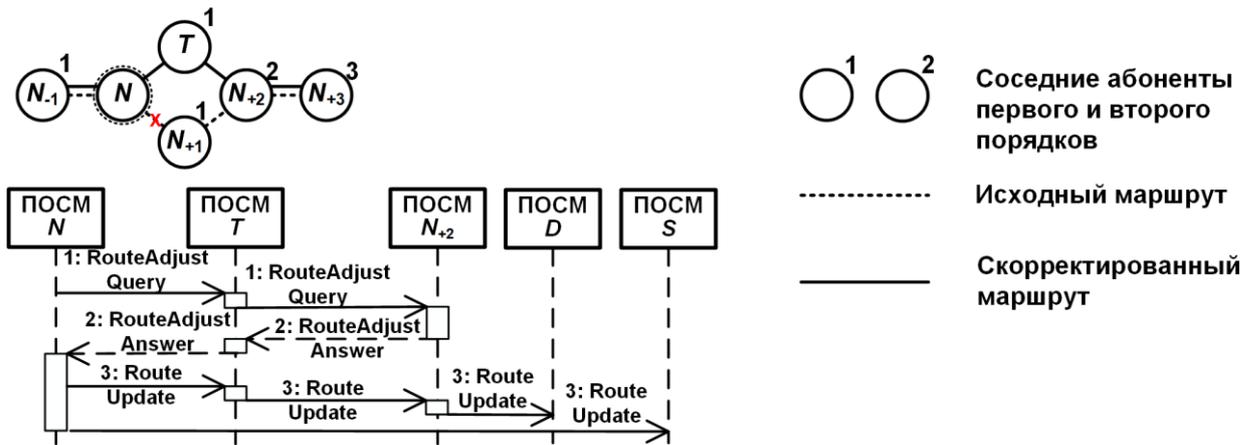


Рисунок 2.17 – Схема метода взаимодействия ПОСМ при восстановлении маршрутов: вариант (г) восстановления маршрутов

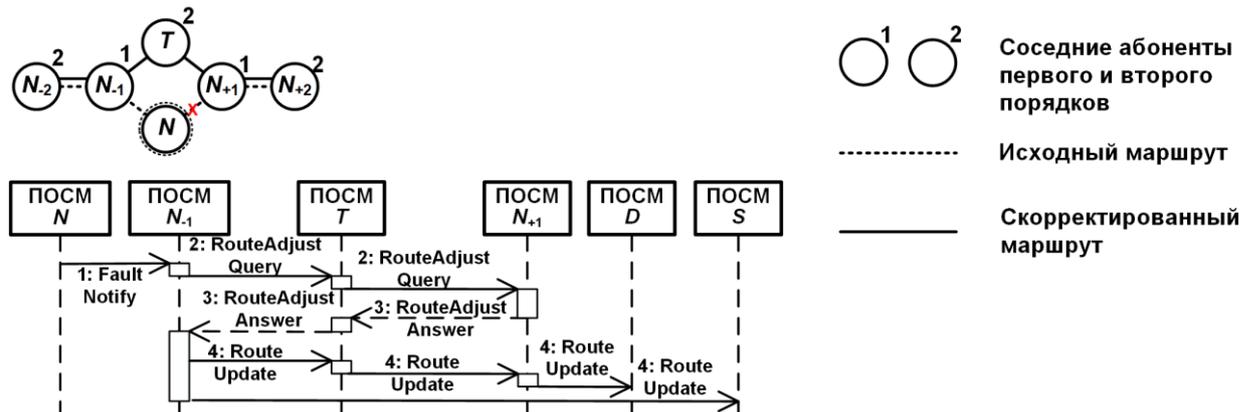


Рисунок 2.18 – Схема метода взаимодействия ПОСМ при восстановлении маршрутов: вариант (д) восстановления маршрутов

5. В случае если маршрут удалось восстановить, производится уведомление абонентов пути по нисходящей ветви о внесённых в него изменениях. В

ином случае производится оповещение о неисправности маршрута входящих в его состав абонентов.

2.3.1. Алгоритм взаимодействия ПОСМ узлов при восстановлении маршрутов

Алгоритм, при помощи которого реализуется предложенный метод, представлен на рис. 8. За счёт того, что сетевое ПО хранит данные о ДХСУ соседних абонентов и ДХКС ассоциированных с ними каналов, которые накапливались в результате проактивного взаимодействия между ПОСМ на этапе поддержания соединения сетью, информационный обмен между узлами осуществляется исключительно в непосредственной окрестности узла-инициатора восстановления маршрута (не более двух промежуточных звеньев). Это позволяет сократить время корректировки маршрута, и как следствие снизить сквозную задержку передачи пакетов данных, вызванную ожиданием восстановления пути. В рамках ПОСМ обмениваются сообщениями *RouteAdjustQuery*, *RouteAdjustQuery2*, *RouteAdjustAnswer*, *RouteAdjustAnswer2*, *RouteUpdate*, *FaultNotify*, *RouteAdjustAnswerNo*, *RouteError*. Пакеты *RouteAdjustQuery* и *RouteAdjustQuery2* предназначены для запроса и уведомления входящего абонента маршрута о внесении в него изменений:

- *Source* – адрес абонента-инициатора построения маршрута;
- *Dest* – абонент, к которому построен маршрута;
- *RS_{ID}* – идентификатор сессии поиска маршрута;
- *N* – абонент-инициатор восстановления маршрута;
- *N₁* – адрес абонента, которому адресован запрос;
- *T* – транзитный абонент для доставки сообщения абоненту *N₁*, если сообщение имеет тип *RouteAdjustQuery*.

Пакеты *RouteAdjustAnswer* и *RouteAdjustAnswer2* предназначены для подтверждения возможности восстановления маршрута предложенным способом. Структура пакетов аналогична *RouteAdjustQuery* и *RouteAdjustQuery2*, при этом добавляются следующие поля:

- *X* – признаковое описание скорректированного маршрута;

– $RList$ – обновлённый список абонентов восходящего маршрута связи.

Уведомление $RouteUpdate$ предназначено для обновления информации о скорректированном маршруте на абонентах нисходящей ветви маршрута. Пакет имеет следующую структуру:

– $Source$ – адрес абонента-инициатора построения маршрута;

– $Dest$ – абонент, к которому построен маршрута;

– $RSID$ – идентификатор сессии поиска маршрута;

– X – признаковое описание скорректированного маршрута;

– $RList$ – обновлённый список абонентов восходящего маршрута связи.

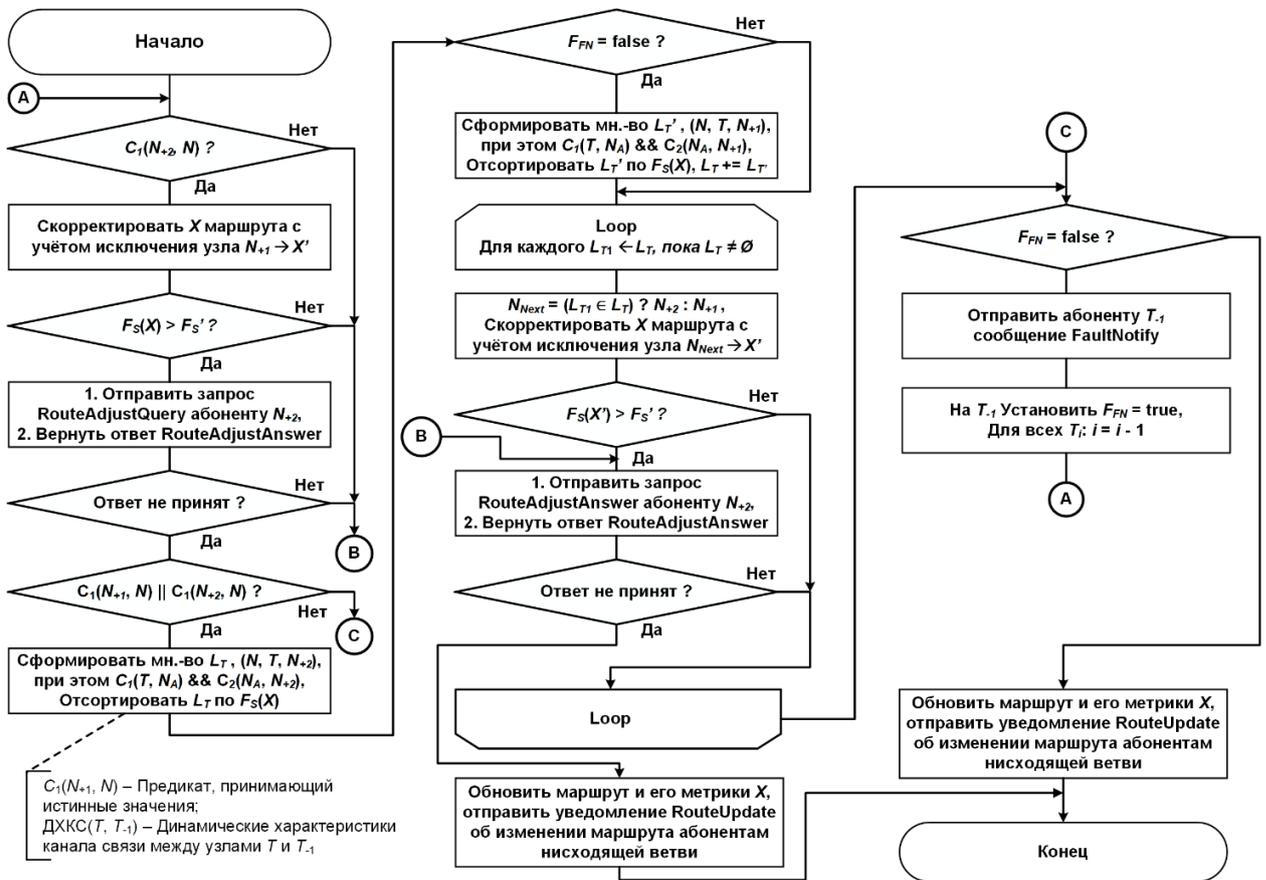


Рисунок 2.19 – Алгоритм взаимодействия ПОСМ

при восстановлении маршрутов связи

Если ПОСМ текущего абонента не способна произвести восстановление маршрута связи, он должен произвести уведомление предыдущего абонента маршрута при помощи пакета $FaultNotify$, который имеет следующую структуру:

– $Source$ – адрес абонента-инициатора построения маршрута;

- *Dest* – абонент, к которому построен маршрута;
- *RS_{ID}* – идентификатор сессии поиска маршрута;
- *N* – абонент-инициатор восстановления маршрута;
- *T* – абонент, которому отправляется уведомление FaultNotify.

Пакет RouteAdjustAnswerNo используется для уведомления абонента-инициатора о невозможности восстановления маршрута выбранным способом. Его структура аналогична сообщениям RouteAdjustQuery и RouteAdjustQuery2. Пакет RouteError предназначен для уведомления нисходящих абонентов маршрута о том, что он стал неисправным. В сообщении используются следующие поля:

- *Source* – адрес абонента-инициатора построения маршрута;
- *Dest* – абонент, к которому построен маршрут;
- *RS_{ID}* – идентификатор сессии поиска маршрута;
- *N* – абонент, который инициировал неудачный процесс восстановления маршрута;
- *X* – признаковое описание неисправного маршрута;
- *RList* – список абонентов, входящих в неисправный маршрут связи.

Пусть на некотором участке маршрут последовательно включает узлы N_{-2} , N_{-1} , N , T_{+1} , N_{+2} , N_{+3} . Тогда взаимодействие ПОСМ при восстановлении маршрутов связи производится по следующему алгоритму:

1) Абонент N обнаруживает неисправность в актуальном маршруте, где следующим абонентом является N_{+1} . Это может быть обусловлено разрывом соединения между узлами N и N_{+1} или снижением производительности маршрута связи до критического значения из-за перегрузки абонентов или канала связи между ними ($F_S(X) < F_S$);

2) Если абонент T_{+2} является соседним для абонента T , производится попытка восстановления маршрута абонента за счёт сокращения его длины путём исключения из него абонента T_{+1} (рис. 2.14);

2.1) ПОСМ узла N на основе информации о производительности канала связи корректирует метрики X для текущего маршрута связи на основе

обновлённой информации о производительности соседних абонентов и каналов связи между ними, извлекаемой из сообщений Hello, а также с учетом удаления из него абонента $N_{+1}: X \rightarrow X'$;

2.2) Если критерий эффективности скорректированного маршрута ниже порогового значения ($F_S(X') < F_S'$), производится переход к пункту 4 текущего метода;

2.3) ПОСМ узла N отправляет абоненту N_{+2} запрос корректировки пути RouteAdjustQuery;

2.4) После приёма пакета ПОСМ узла N_{+2} отправляет подтверждение RouteAdjustAnswer с обновлёнными маршрутными метриками восходящей ветви маршрута;

2.5) ПОСМ узла N ожидает подтверждение в течение интервала ROUTE_ADJUST_INTERVAL. Если пакет был принят в течение указанного периода, ПОСМ узла N формирует пакет RouteUpdate, в котором записывается обновлённый маршрут с его метриками и по нисходящей ветви маршрута отправляет пакет абоненту источнику и абоненту назначения. Узлы маршрута на основе информации из пакета RouteUpdate обновляют информацию о текущем пути;

2.6) В ином случае принимается, что указанным способом невозможно восстановить маршрут. Производится переход к пункту 3;

3) Если абоненты N_{+2} или N_{+1} являются соседями второго порядка для узла T , производится попытка восстановить маршрут путём замены одного абонента (рис. 2.17) или включением в путь дополнительного узла (рис. 2.16);

3.1) Для этого формируется множество двухшаговых маршрутов $L_T = (N, T, N_{+2})$, где N_A - соседний абонент, через которого транзитивно как сосед второго порядка доступен N_{+2} - следующий за N_{+1} абонент в маршруте связи. Множество L_T сортируется по убыванию метрики $F_S(X)$;

3.2) Если ранее не был установлен флаг F_{FN} , то производится формирование множества кортежей – двухшаговых маршрутов $L_T' = (N, N_A, N_{+1})$, где N_A - соседний абонент, через которого транзитивно доступен N_{+1} . Множество L_T'

сортируется по убыванию метрики $F_S(X)$, после чего L_T' добавляется в конец множества L_T ;

3.3) Из множества L_T поочерёдно достаются кортежи L_{T1} . При помощи узлов из кортежа производится корректировка повреждённого участка пути (рис. 2.17 и рис. 2.18). Если оценка критерия эффективности скорректированного маршрута меньше порогового значения $F_S(X_{LN1}) < F_S'$, производится переход к пункту 4. При этом вводится обозначение N_{Next} , где $N_{Next} = N_{+2}$, если $N_{Next} \in L_T$ или $N_{Next} = N_{+1}$, если $N_{Next} \in L_T'$;

3.4) ПОСМ узла N отправляет абоненту N_{Next} запрос корректировки RouteAdjustQuery2, при этом в качестве транзитного абонента указывается N ;

3.5) При получении пакета RouteAdjustQuery2 ПОСМ узла T проверяет, возможно ли включение текущего абонента в маршрут. При невозможности абоненту N отправляется уведомление RouteAdjustAnswerNo, в ином случае сообщение RouteAdjustQuery2 ретранслируется на следующий узел;

3.6) При получении пакета ПОСМ узла T_{Next} отправляет подтверждение RouteAdjustAnswer2 с обновлёнными маршрутными метриками;

3.7) ПОСМ узла T ожидает подтверждение в течение заданного интервала времени. Если в течение этого промежутка был получен пакет RouteAdjustAnswer2, абонент N формирует пакет RouteUpdate, в который записывает обновлённый маршрут с метриками и отправляет его по нисходящей ветви пути абоненту-источнику и целевому абоненту. При прохождении пакета RouteUpdate через узлы маршрута они производят обновление информации о пути в своих таблицах маршрутизации;

3.8) Если время ожидания пакета RouteAdjustAnswer2 превысило максимально допустимое значение, и при этом множество L_T не пусто, производится переход к пункту 3.3. Если множество L_T не содержит элементов, производится переход к пункту 5;

3.9) Если ПОСМ узла N принимает пакет RouteAdjustAnswerNo от абонента T , производится переход к пункту 3.3;

4) Если абонент не получал ранее сообщения FaultNotify (для текущего маршрута не установлен флаг F_{FN}):

4.1) Если N_{-1} или N_{+1} являются соседними абонентами (рис. 2.15 и 2.18), абонент N отправляет уведомление узлу N_{-1} о сбое маршрута путём отправки пакета FaultNotify. ПОСМ узла N_{-1} , приняв FaultNotify, выполняет действия пункта 2 (обозначение абонента смещается на -1 в значении индекса), при этом устанавливает флаг F_{FN} ;

4.2) В ином случае производится переход к пункту 5;

5) Иначе узел N помечает маршрут сбойным. N формирует пакет RouteError, в который записывает множество узлов нисходящей ветви маршрута. При получении пакета RouteError абоненты удаляют из своих таблиц маршрутизации маршрут с заданным в RouteError идентификатором;

В случае невозможности обновления маршрута за счёт соседних узлов, абоненту источнику отправляется пакет RouteError с информацией о разрыве связи. В этом случае трафик распределяется между оставшимися маршрутами. Если отсутствуют исправные альтернативные пути, процесс их построения может быть запущен повторно через некоторый интервал времени;

В работе сети возможны случаи, когда между двумя абонентами невозможно построить маршрут. Например, целевой узел D или подсеть, в которой он расположен, не связана с подсетью текущего абонента. Такая ситуация обнаруживается, когда абонент D не является смежным узлом 1-го или 2-го порядка, или когда истекло время ROUTE_SEARCH_TIME, и маршрут не был получен. Тогда через время REPEAT_SEARCH_TIME инициируется повторный запрос маршрута связи. Если повторный поиск не принёс результатов, считается, что абонент D не доступен для передачи данных.

2.4. Метод балансировки трафика по квазипараллельным маршрутам

Для равномерного распределения нагрузки между сформированными квазипараллельными маршрутами предложен метод балансировки потоков данных в ДСПД, позволяющий снизить сквозную задержку доставки сообщений в случаях, когда стандартные подходы не обеспечивают построение

параллельных путей. Он отличается от известных децентрализованным взаимодействием сетевых узлов при распределении пакетов трафика по квазипараллельным маршрутам с учётом значений их динамических метрик, собираемых при информационном обмене между программным обеспечением сетевой маршрутизации.

Метод (рис. 9) заключается в определении весов маршрутов на основе частных метрик, собранных на этапе взаимодействия ПОСМ при формировании путей, расчёте пропорций, в которых трафик будет распределён по квазипараллельным маршрутам, и в корректировке частных метрик при получении информационных уведомлений о восстановлении маршрута связи. При этом процесс балансировки выполняется программным обеспечением децентрализованно на каждом из узлов сети, у которых имеется две и более исходящие маршрутные ветви.

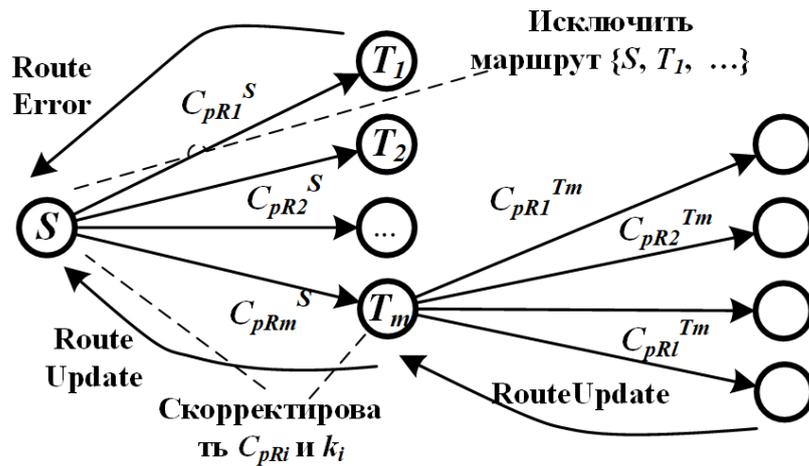


Рисунок 2.20 – Схема метода балансировки трафика

В качестве абсолютных значений весов маршрутов используется обобщённый критерий F_B , который аналогично метрике F_S является взвешенным произведением частных показателей PDR и E2ED, но взятых с иными весовыми коэффициентами K_{B1} и K_{B2} :

$$F_B = \frac{\text{PDR}^{K_{B1}}}{\left(\frac{\text{E2ED}}{\text{E2ED}_{norm}} \right)^{K_{B2}}} \cdot \quad (2.9)$$

Для оценки весов маршрутов F_B используется функция $F_B(X)$, значения которой определяются с помощью процедуры градиентного бустинга деревьев

принятия решений. Количество пакетов, отправляемых по каждой исходящей ветви, равняется:

$$C_{pR1} = C_{all} * k_1, C_{pR2} = C_{all} * k_2, \dots, C_{pRm} = C_{all} * k_m, \quad (2.10)$$

$$k_1 = F_B(X_{R1}) / S, k_2 = F_B(X_{R2}) / S, \dots, k_m = F_B(X_{Rm}) / S, \quad (2.11)$$

$$S = F_B(X_{R1}) + F_B(X_{R2}) + \dots + F_B(X_{Rm}), \quad (2.12)$$

где C_{all} – общее количество пакетов, которые отправляет промежуточный абонент или узел-источник по нескольким исходящим маршрутам, C_{pRi} , k_i – количество и доля пакетов, отправляемых по i -му исходящему маршруту, X_{Ri} – метрики i -го исходящего маршрута.

В качестве сообщения, при помощи которого могут быть переданы данные, выступает пакет DataRoute. Сообщение должно содержать следующие поля:

- *Source* – абонент источник данных;
- *Dest* – абонент, которому передаются данные;
- *T* – транзитный абонент, являющийся следующим в цепочке ретрансляции данных по сети;
- *RS_{ID}* – идентификатор маршрута, по которому производится доставка данных;
- *F_{tmp}* – флаг, который устанавливается, когда передача данных осуществляется по временному маршруту.

2.4.1. Алгоритм коммуникации между ПОСМ при доставке пакетов с данными

Для практической реализации предложенного метода балансировки трафика по квазипараллельным маршрутам передачи данных был разработан алгоритм коммуникации между ПОСМ узлов при доставке пакетов данных, схема которого представлена на рис. 2.21. Балансировка трафика может быть осуществлена следующим образом:

1) Устанавливается максимально допустимая задержка очереди на узлах маршрута T_{OMX} . В случае, если пакет с данными находится в очереди дольше, он уничтожается и считается сбойным;

2) Если целевой абонент D является соседом 1-го уровня, данные передаются напрямую D без дополнительного взаимодействия с целевым узлом. В поле T устанавливается значение 0;

3) Если абонент D является соседом 2-го порядка, в поле T – устанавливается адрес узла, через которого транзитивно доступен абонент T . Пакет с данными передаётся в два перехода;

4) Если ПОСМ узла S имеет временный маршрут к абоненту D , но отсутствуют постоянные маршруты, передача данных производится по временному маршруту;

4.1) Для передачи по временному маршруту в пакете DataRoute устанавливается флаг временного маршрута ($F_{tmp} = 1$).

4.2) В процессе передачи данных адреса последующих абонентов ретрансляторов T определяются по записям $RouteItem[RS_{ID}, S].OUTPUT_TMP$ из $AbonentInformation$;

5) Если ПОСМ узла S содержит актуальные постоянные маршруты:

5.1) Для передачи данных по постоянным маршрутам используется пакет DataRoute с признаком постоянного маршрута ($F_{tmp} = 0$).

5.2) В качестве узлов ретрансляторов трафика используются абоненты T_1, T_2, \dots, T_m из записи $RouteItem[RS_{ID}, S].OUTPUT$ таблицы $AbonentInformation$;

5.3) Трафик между абонентами распределяется в пропорциях $k_1 : k_2 : \dots : k_m$, если количество маршрутов больше одного;

5.4) Аналогичным образом трафик распределяется с последующих транзитных абонентов, если соответствующие записи $RouteItem[RS_{ID}, S].OUTPUT$ содержат несколько узлов;

б) Передача подтверждающих квитанций о доставке данных при необходимости может быть осуществлена либо по временному, либо по основному маршруту связи.

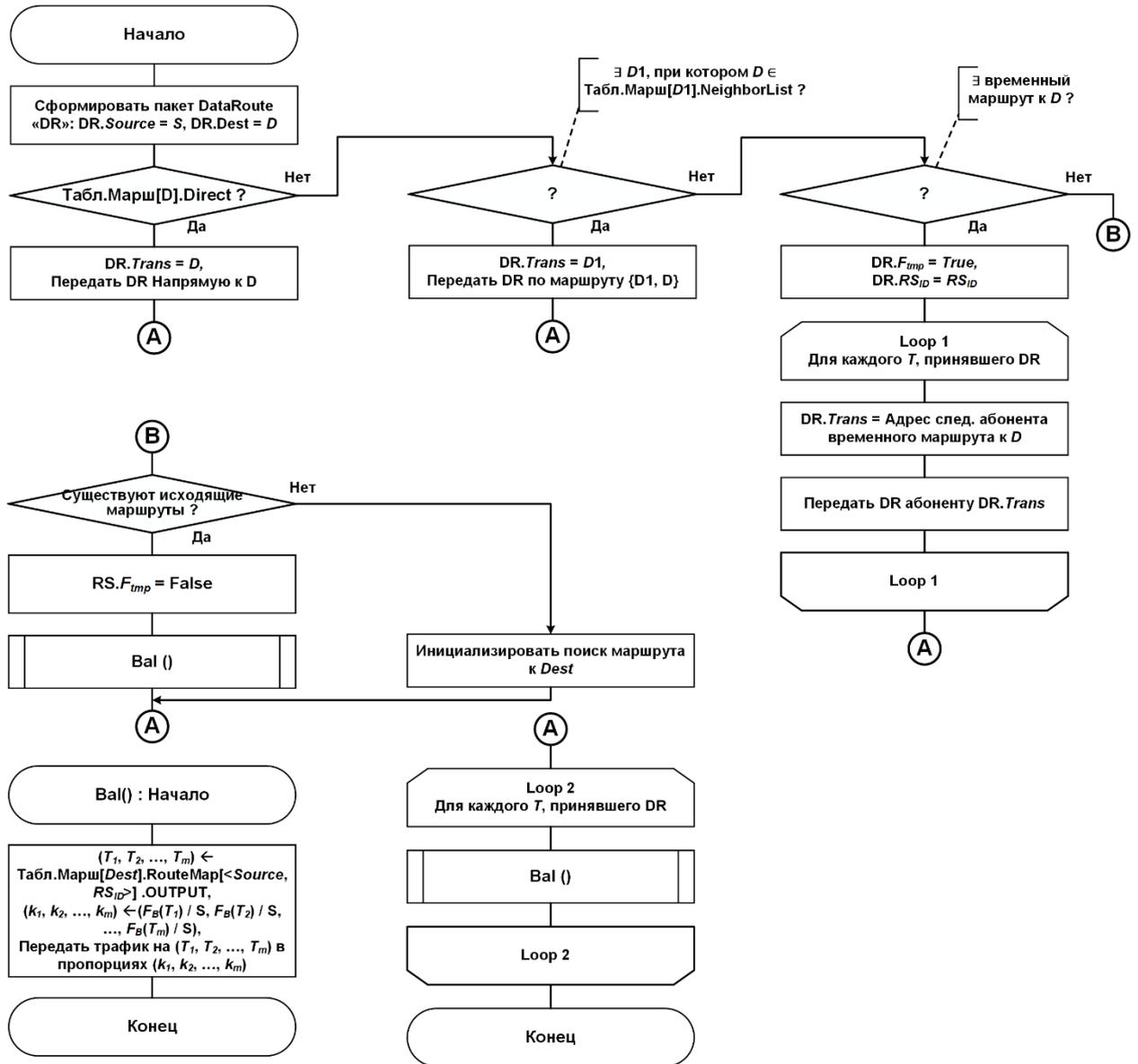


Рисунок 2.21 – Алгоритм коммуникации между ПОСМ узлов при доставке пакетов с данными

7) Если абонент, который производит распределение трафика по нескольким исходящим маршрутам, принял пакет RouteUpdate для пути, который задействован в балансировки, обновляется информация о нём, и выполняется п. 5 текущего алгоритма.

8) Если абонент, который производит распределение трафика по нескольким исходящим маршрутам, принял пакет RouteError для пути, который

задействован в балансировки, производится удаление маршрута и выполняется п. 5 текущего алгоритма.

Если в процессе построения маршрутов был сформирован временный путь, передача данных осуществляется по нему. При наличии нескольких квазипараллельных маршрутов пакеты распределяются в соответствии с весами маршрутами, которые были получены при помощи функции $F_B(X)$. При этом в балансировки трафика задействованы абоненты, которые имеют несколько исходящих маршрутов к целевому абоненту.

2.5. Выводы по главе

В текущем разделе были достигнуты следующие результаты:

1. Предложен метод взаимодействия программного обеспечения сетевой маршрутизации при установлении и дальнейшем поддержании соединения с сетью, включая сбор ДХКС и динамических характеристик сетевого узла (ДХСУ) для дальнейшего расчёта показателей качества обслуживания многопутевых маршрутов при их первичном формировании и дальнейшем поддержании.

2. Предложен метод взаимодействия сетевого ПО при построении квазипараллельных маршрутов, который включает алгоритм запроса маршрута, алгоритм возврата маршрута абоненту-инициатору поиска. Данный метод использует функцию вычисления показателя качества маршрута $F_S(X)$ для исключения ненадёжных частично сформированных маршрутов и их упорядочивания. Также он обеспечивает процедуру поиска маршрутов связи, результаты которой сохраняются в таблице `AbonentInformation`. Предложенные метод и алгоритмы отличаются от известных возможностью поддержки множества частных маршрутных метрик, которые комплексированы в единый показатель F_S качества обслуживания пути. При этом обеспечивается построение квазипараллельных маршрутов, между которыми имеется возможность распределять поток трафика.

3. Предложен метод взаимодействия сетевого ПО при восстановлении маршрутов связи, включающий локальное изменение маршрута передачи данных за счёт включения или исключения из него соседних абонентов. Элементами новизны предложенного решения по сравнению с известными методами

является использование локализованного реактивного алгоритма, дополненного проактивным обменом между смежными абонентами информацией о значении динамических характеристик, ассоциированных с ними узлов и каналов связи.

4. Предложен метод балансировки трафика по квазипараллельным маршрутам связи, которые формируются на основе разработанных в диссертации методов и алгоритмов взаимодействия программных служб сетевых узлов. Данный метод использует функцию $F_B(X)$, введенную в работе для вычисления весов маршрутов, в соответствии с которыми трафик будет распределяться по найденным квазипараллельным путям, на основе собранной в процессе взаимодействия ПОСМ информации о путевых метриках;

5. Разработан метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия программного обеспечения сетевых узлов, отличающийся от стандартных решений возможностью динамической фильтрации и упорядочивания информационных сообщений на промежуточных узлах сети, что позволяет повысить эффективность функционирования ДСПД за счет снижения суммарного объема служебного трафика между программным обеспечением сетевой маршрутизации отдельных абонентов.

6. Для практической проверки предложенных методов и алгоритмов требуется наличие средства имитационного моделирования, позволяющего эмулировать среду передачи данных и проверять различные протоколы и механизмы формирования ДСПД, процессы информационного обмена и взаимодействия между сетевым ПО отдельных узлов и процедуры маршрутизации трафика между ними, а также сетевой протокол взаимодействия сетевого ПО с форматами пакетов, при помощи которых производится взаимодействие.

3. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СЕТЕВЫХ УЗЛОВ ДЛЯ УМЕНЬШЕНИЯ СКВОЗНОЙ ЗАДЕРЖКИ И ПОВЫШЕНИЯ НАДЕЖНОСТИ ПЕРЕДАЧИ ДАННЫХ

3.1. Программное средство для моделирования взаимодействия ПОСМ узлов ДСПД

3.1.1. Анализ симуляторов функционирования сетей связи

Существует множество пакетов моделирования компьютерных сетей [99, 100], которые применимы для исследования, разработки, настройки и оценки качества взаимодействия сетевого программного обеспечения. Среди симуляторов можно выделить как пакеты для моделирования компьютерных сетей в целом (NS-2, NS-3, Omnet++, OPNET, Cisco Packet Tracer), так и специализированное программное обеспечение, предназначенное для моделирования беспроводных ДСПД (Dingo, J-Sim).

При разработке новых алгоритмов взаимодействия ПОСМ требуется производить проверку корректности их функционирования и отладку, что может быть выполнено при помощи сетевых симуляторов. Большинство инструментариев не позволяет добавлять пользовательские реализации программного обеспечения сетевой маршрутизации (Cisco Packet Tracer, OPNET). Имеющие такую функцию прикладные пакеты программного обеспечения позволяют только корректировать уже встроенные решения, ограниченные интегрированными в симулятор сетевыми стеками.

При отладке и исследовании процессов взаимодействия сетевого ПО может возникнуть потребность в наблюдении за определёнными этапами коммуникации между конкретными абонентами, передачи данных в рамках одного маршрута или пакета с последующим визуальным отображением указанных процессов. Системы моделирования NS2 и NS3, распространяемые по лицензии GPL, не имеют встроенный графический интерфейс. Omnet++ производит визуализацию передачи пакетов по сети, однако не позволяет отдельно выделять трафик между конкретными абонентами. Кроме того, высокая степень

детализации существующих моделей может затруднить отладку протоколов передачи данных на начальном этапе разработки. Из-за проблем и ошибок на транспортном, канальном и физическом уровнях сетевого стека будет затруднительно определить, какие ошибки в передаче данных вызваны логикой взаимодействия программного обеспечения сетевой маршрутизации.

Поэтому возникает потребность в собственном программном инструменте для моделирования взаимодействия ПОСМ узлов ДСПД, в котором были бы преодолены указанные выше недостатки. Он должна предоставлять возможность произвести исследование и оценку разработанных алгоритмов взаимодействия сетевого ПО. Кроме того, программное средство должно обеспечивать функционал, который отсутствует или реализован не в полной мере в существующих пакетах:

- осуществлять визуализацию функционирования алгоритмов поиска маршрутов и передачи данных по сети;
- обеспечивать сбор основных показателей качества работы ДСПД и отдельных узлов, таких как время обнаружения маршрута, время доставки данных, коэффициент доставки пакетов данных, загрузка сети и отдельных каналов связи;
- позволять задавать сценарии работы сети, включая установление и разрыв каналов связи между абонентами, отправку прикладного трафика;
- реализовывать различные протоколы и методы взаимодействия сетевого ПО при реализации коммуникаций между абонентами, как стандартные, так и предложенные в диссертационной работе.

3.1.2. Структура программного средства

В связи с предъявленными требованиями к программному инструменту моделирования беспроводных децентрализованных сетей передачи данных предлагается использовать структуру, представленную на рис. 3.1. В её состав входят:

- модуль сетевого узла;
- модуль среды передачи;

- модуль сценариев;
- модуль сбора статистики;
- модуль визуализации.

Модуль сетевого узла предназначен для имитации абонентов ДСПД. Он состоит из подмодулей программного обеспечения сетевой маршрутизации, прикладного программного обеспечения и подмодуля канала. Модуль прикладного программного обеспечения предназначен для генерации прикладного трафика сети по встроенным или задаваемым пользователями сценариям. Такой трафик может быть периодическим, либо генерироваться по равномерному или экспоненциальному законам распределения [101].

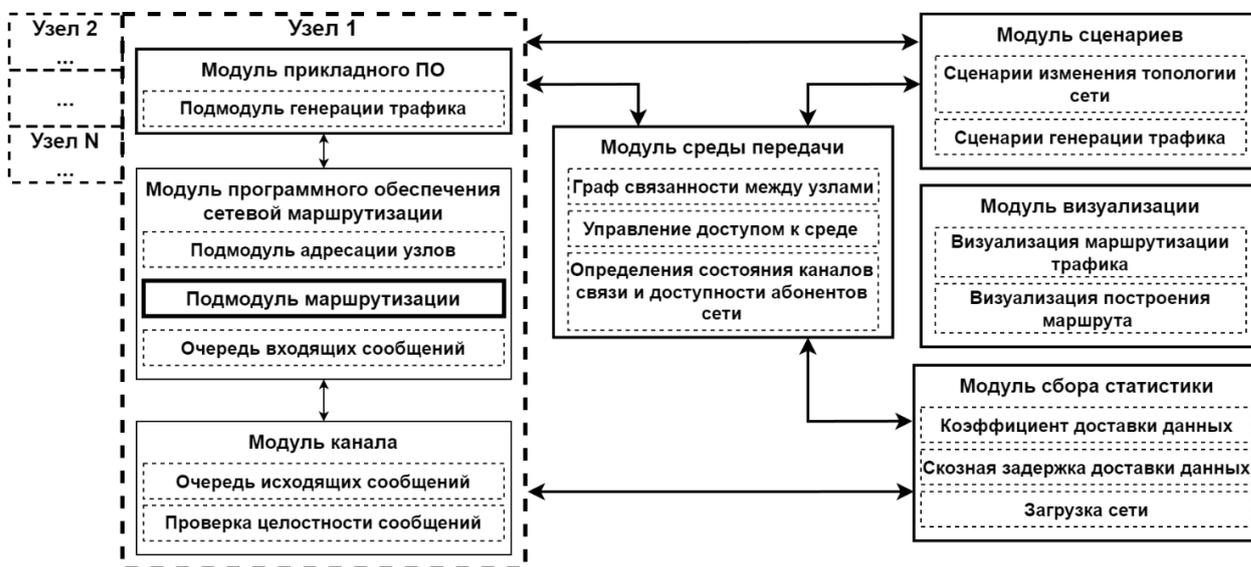


Рисунок 3.1 – Структура программного средства

Модуль программного обеспечения сетевой маршрутизации предназначен для имитации функционирования ПОСМ, которые коммуницируют между собой по алгоритмам взаимодействия. Ключевой структурной единицей в нём является подмодуль маршрутизации, в котором инициируется работа алгоритмов для подключения абонента к ДСПД и дальнейшего поддержания соединения с ней, обнаружения квазипараллельных маршрутов и последующего сохранения их в актуальном состоянии. Принимаемые от модуля канала пакеты сохраняются в очереди входящих сообщений. Подмодуль адресации отвечает за назначение адресов абонентам. Адреса узлов в процессе моделирования

могут быть статически заданы во входном файле или автоматически назначаться новым абонентам при их подключении к сети.

Модуль канала содержит очередь исходящих сообщений, проверяет целостность входящих пакетов и отвечает за непосредственное взаимодействие абонента с модулем среды передачи. В основе модуля среды передачи лежит граф каналов связи между абонентами. Канал характеризуется надёжностью – долей пакетов данных, которые достигают получателя, а также скоростью передачи данных. Модуль производит контроль доступа к среде передачи данных для абонентов ДСПД.

Модуль сценариев предназначен для управления сценариями работы программного средства, а именно, генерации прикладного трафика и изменения топологии ДСПД. Модуль сбора статистики предназначен для определения коэффициентов доставки данных, сквозной задержки передачи пакетов по отдельным маршрутам и между конкретными абонентами, а также загрузки каналов связи и сети в целом. Модуль визуализации должен обеспечивать демонстрацию процессов взаимодействия ПОСМ и доставки данных между заданными узлами.

Также разработан модуль «Задания движения абонента в пространстве» для моделирования перемещения мобильных абонентов ДСПД на местности и отображения изменений состояния каналов связи между ними в зависимости от взаимного положения узлов друг относительно друга. Для симулирования движения мобильных абонентов используются модели случайных перемещений на основе случайных точек с неограниченной областью моделирования, а сценарии перемещения узлов, в которых прописаны маршруты движения абонентов, их стартовые положения и скорость, могут быть заданы пользователем в качестве параметров модели. Для узлов указывается зона покрытия – максимальное расстояние между абонентами, при котором возможна прямая коммуникация между ними. Перемещение абонентов в пространстве сопровождается изменением состояния каналов связи: если узлы расположены за пределами зоны покрытия – связь отсутствует, в ином случае в зависимости от

расстояния определяется её текущий уровень. Модуль позволяет осуществлять визуализацию процессов движения узлов по заданной площадке, формирование и разрыв каналов связи между отдельными абонентами, а также отображать зоны их покрытия.

Указанные модули позволяют реализовать программное средство для моделирования взаимодействия ПОСМ узлов ДСПД, которое позволяет производить отладку режимов взаимодействия сетевого программного обеспечения.

3.1.3. Реализация программной средства моделирования ДСПД

При тестировании, отладке и оптимизации параметров [102; 103] сетевого программного обеспечения требуется производить его запуск на длительное время (от нескольких часов и более). Поэтому в основе проектируемого программного средства заложен принцип дискретно-событийного моделирования [104]. Это обеспечит логическую и физическую непротиворечивость последовательности событий в сети при отработке различных сценариев ее функционирования, а также позволит сократить время функционирования программного средства. На описанные модули были получены свидетельства о государственной регистрации программ для ЭВМ «Программное обеспечение для маршрутизации трафика в беспроводных децентрализованных сетях связи», «Программное обеспечение для моделирования взаимного пространственного перемещения отдельных узлов в беспроводных децентрализованных сетях связи». Для реализации программного средства используются классы, представленные на рис. 3.2.

Для осуществления дискретно-событийного моделирования было использовано API среды Omnet++. Каждый модуль программного средства наследует интерфейс IModule. Среда передачи реализуется при помощи класса RadioMedium. Модули прикладного ПО, ПОСМ и канала реализованы классами AppModule, NetworkModule и RadioModule. Для их взаимодействия используются методы `send()` и `recv()`. Название сообщения указывает на отправителя. Управление сценариями реализовано в классе ScriptModule.

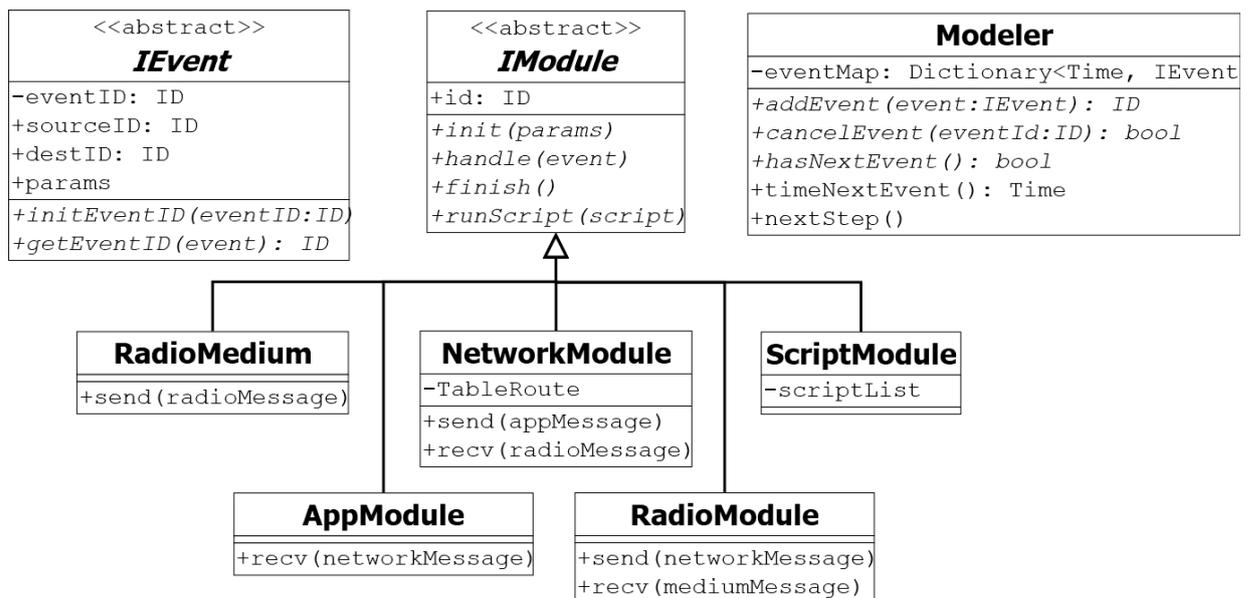


Рисунок 3.2 – Диаграмма классов программного средства

Каждый модуль идентифицируется значением поля `id`. При помощи метода `init()` производится их инициализация соответствующими параметрами. Для обработки событий используется метод `handle()`. При завершении моделирования должен быть вызван метод `finish()`, в котором при взаимодействии с модулем статистики определяются результаты моделирования. Для запуска сценария менеджер сценариев может вызвать `runScript()` на нужном модуле.

Абстрактный класс `IEvent` содержит базовую информацию о событии. Он включает идентификаторы события `eventID`, модуля-инициатора события `sourceID` и модуля-обработчика события `destID`, а также параметры события (время, на которое запланировано событие, количество раз, которое событие будет сгенерировано, пользовательские данные и другие). Для управления процессом моделирования используется класс `Modeler`. Он содержит словарь событий, где в качестве ключа выступает время наступления события. Методы `addEvent()` и `cancelEvent()` используются для создания и отмены событий на модулях в методах `init()` или `handle()`. Методы `hasNextEvent()` и `timeNextEvent()` используются для проверки наличия запланированных событий и возврата времени возникновения

следующего события. Для выполнения следующего шага моделирования может быть использован метод `nextStep()`.

В моделировании задействованы следующие глобальные переменные:

– `Modeler modeler;`

– `ScriptModule scripts;`

– `MediumModule medium;`

– `List<Node> nodeList[N]`, где `N` – количество узлов сети, `Node` – кортеж (`AppModule app`, `NetworkModule netw`, `RadioModule radio`);

– `Time timeLimit` – виртуальное время, в течение которого производится моделирование функционирования ДСПД.

Используя представленные типы данных и переменные, дискретно-событийное моделирование производится по алгоритму, представленному на рис. 3.3.

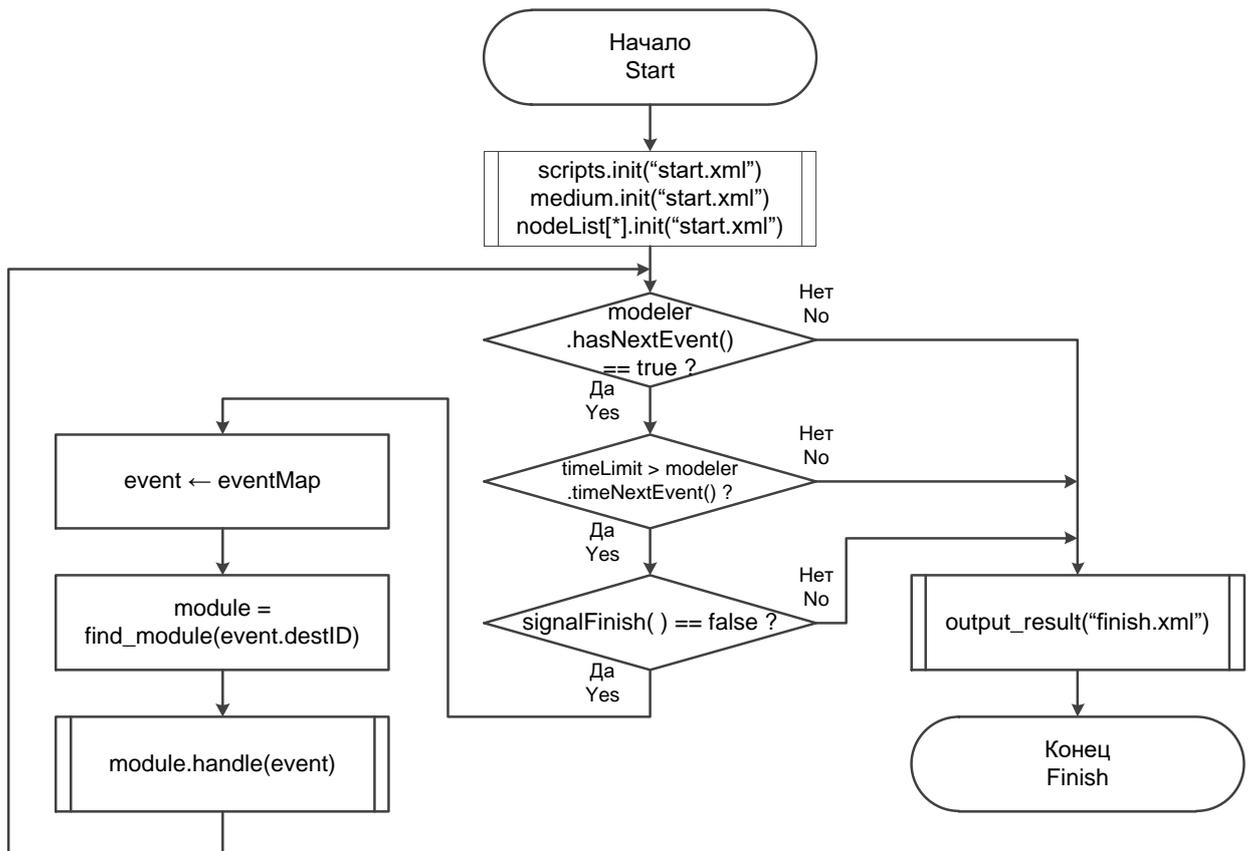


Рисунок 3.3 – Алгоритм дискретно-событийного моделирования

В качестве входных и выходных данных программного инструмента выступают файлы `init.xml` и `finish.xml`. Первый содержит первоначальные

настройки сети, параметры и сценарии моделирования, второй – результаты моделирования. Программное средство было написана на языке программирования C++ с использованием фреймворка Qt5 [105] и API сетевого симулятора Omnet++. Интерфейс программного обеспечения представлена на рис. 3.4.

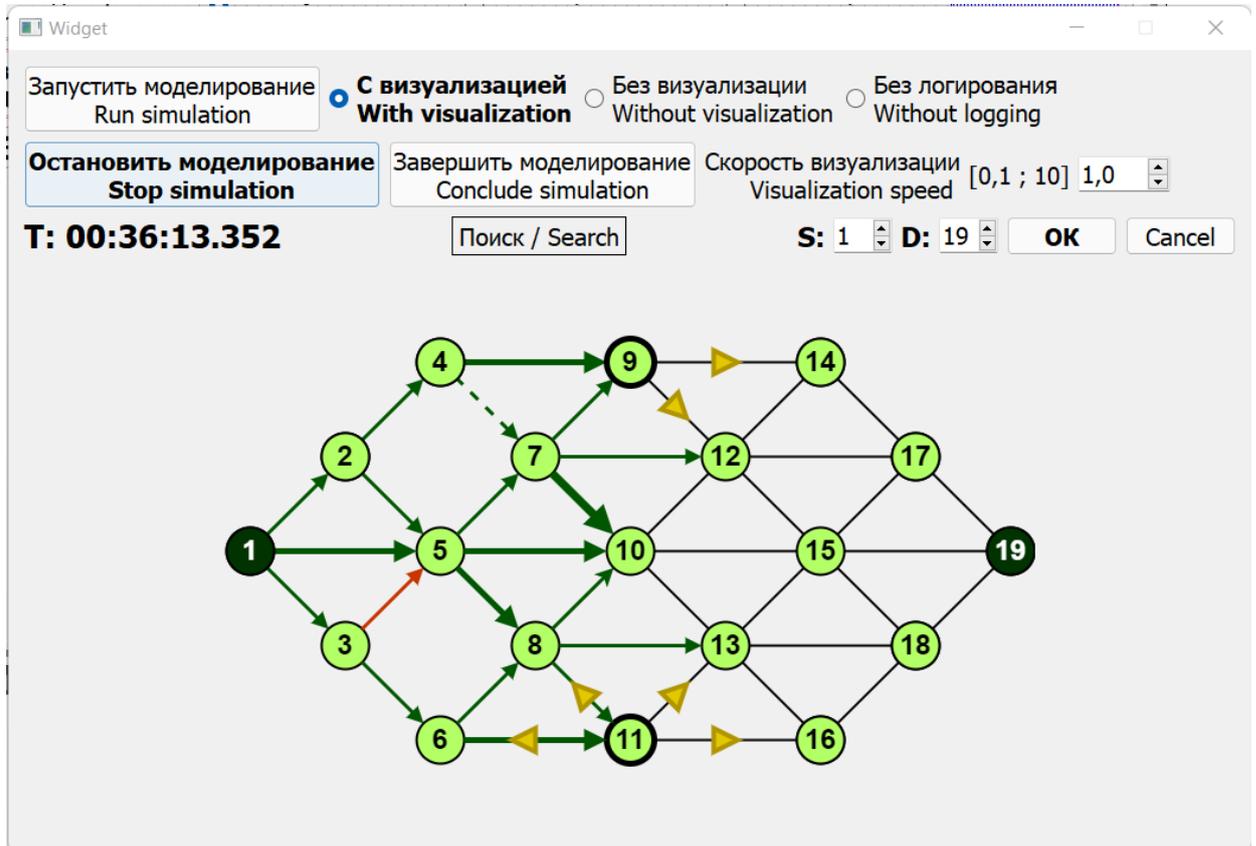


Рисунок 3.4 – Программное обеспечение для моделирования ДСПД в режиме «Поиск»

Представленный программный инструмент позволяет производить моделирование в 3-ёх режимах: с визуализацией, без визуализации, без логирования. В первом режиме имеется возможность визуально отслеживать процессы поиска маршрутов и доставки данных между выбранными узлами. Для этого необходимо выставить соответствующие номера узлов S и D в графах «S:» и «D:». Установленный коэффициент в меню «Скорость визуализации» указывает, с какой скоростью относительно базовой будет воспроизводиться анимация. Визуализация осуществляется либо при поиске маршрута (режим «Поиск / Search»), либо при передаче данных (режим «Передача / Transfer», рис. 3.5).

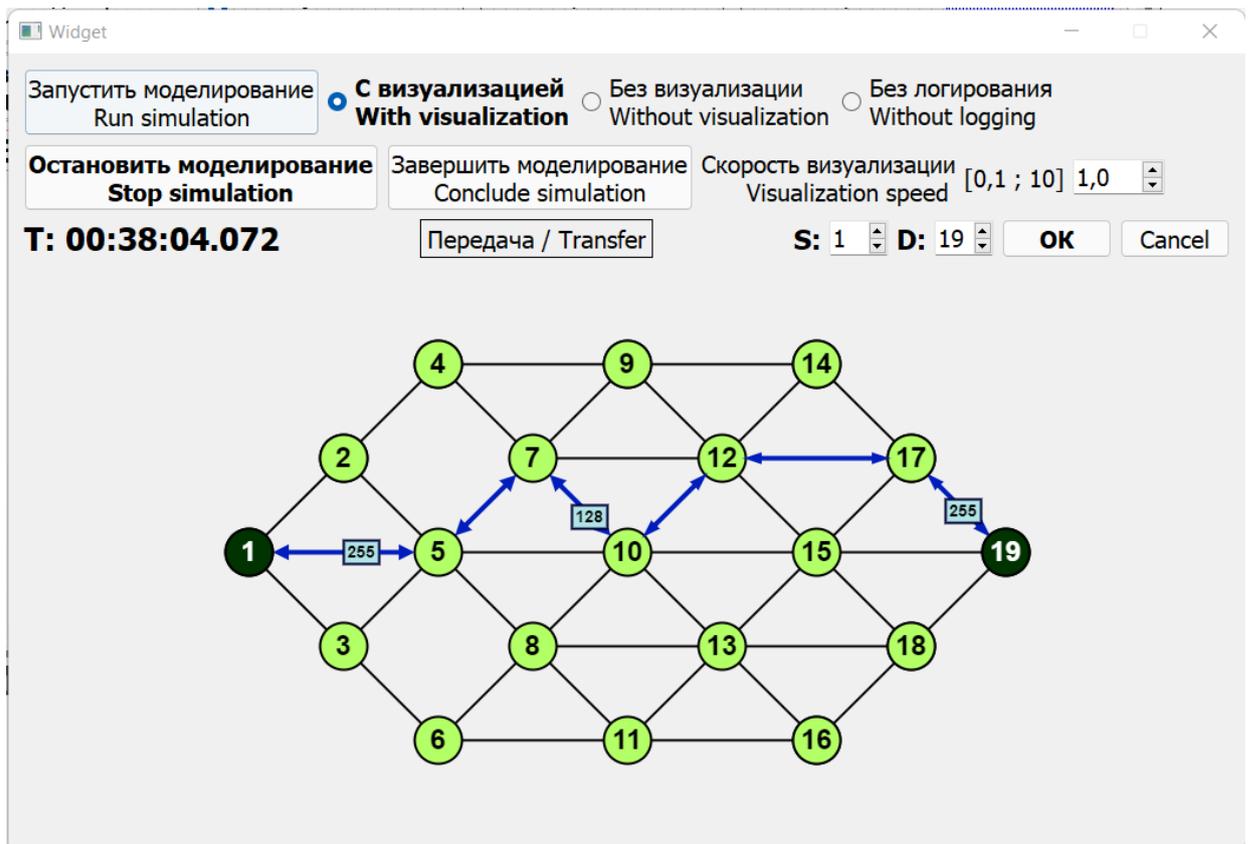


Рисунок 3.5 – Программное обеспечение для моделирования ДСПД сети в режиме «Передача»

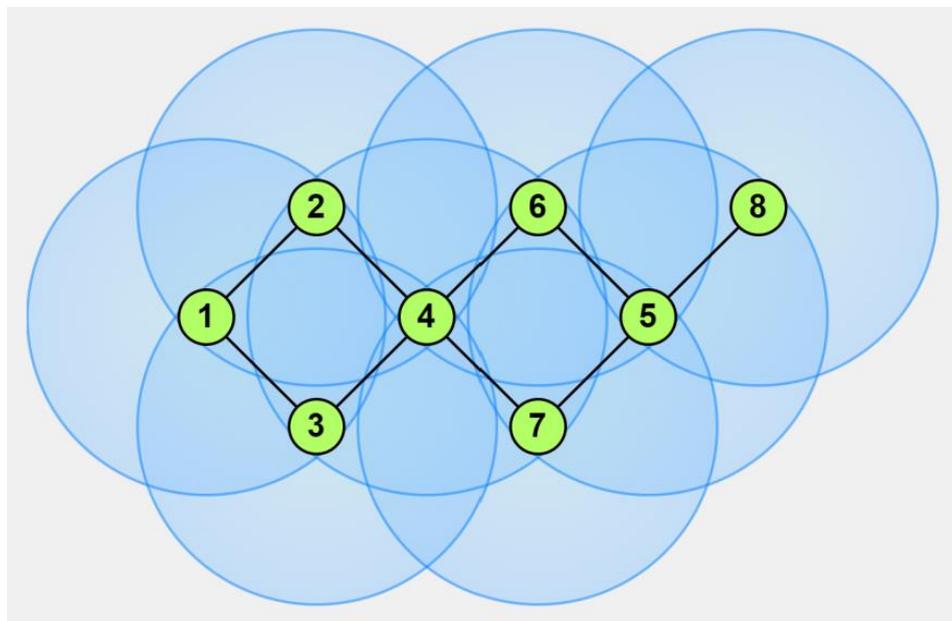


Рисунок 3.6 – Визуализация абонентов в режиме моделирования взаимного пространственного перемещения абонентов

Второй и третий режимы позволяют проводить моделирования без визуализации функционирования сетевых алгоритмов. Третий режим отличается от второго отсутствием логирования, что ускоряет процесс моделирования, но

уменьшает его информативность. На рис. 3.6 представлен вид программного обеспечения при использовании функции взаимного пространственного перемещения отдельных узлов.

Кнопка «Остановить моделирование» предназначена для постановки моделирования на паузу. Кнопка «Завершить моделирование» посылает сигнал `signalFinish()`, который досрочно завершает моделирование, и выводит результаты в файл `finish.xml`. Также в левой части программного обеспечения отображается текущее виртуальное время моделирования.

3.2. Архитектура сетевого программного обеспечения отдельных абонентов ДСПД с поддержкой квазипараллельных маршрутов передачи данных и балансировки трафика в них

На основе предложенных методов и алгоритмов взаимодействия была разработана архитектура программного обеспечения сетевой маршрутизации отдельных узлов для решения задач маршрутизации трафика в ДСПД. Её схема представлена на рис. 3.7. ПОСМ каждого абонента включает следующие модули:

- модуль графа сети и маршрутов (структура `AbonentInformation`);
- модуль оценки маршрутов связи;
- модуль извлечения маршрутных метрик;
- модуль маршрутизации.

Модуль графа сети и маршрутов реализуют структуру `AbonentInformation`. Модуль выполняет следующие функции:

- хранит информацию о соседних абонентах 1-го и 2-го порядков и производительности их узлов и каналов связи между ними, частично сформированные, временный, оптимальный и альтернативные маршруты между абонентами и показатели их производительности;
- предоставляет интерфейс для доступа к структуре `AbonentInformation` другим модулям ПОСМ.

Модуль оценки маршрутов связи предназначен для вычисления показателей качества маршрутов связи во время их построения и балансировки трафика по ним. Он предоставляет интерфейс доступа к функциям $F_S(X)$ и $F_B(X)$, принимающим на вход частные маршрутные метрики и выдающим на выходе критерии качества маршрутов F_S и F_B .

Модуль извлечения маршрутных метрик реализуется для хранения и формирования показателей производительности своего абонента и смежных каналов связи. Хранимая в структуре информация обновляется в процессе функционирования алгоритмов установления и поддержания соединения с соседними абонентами.

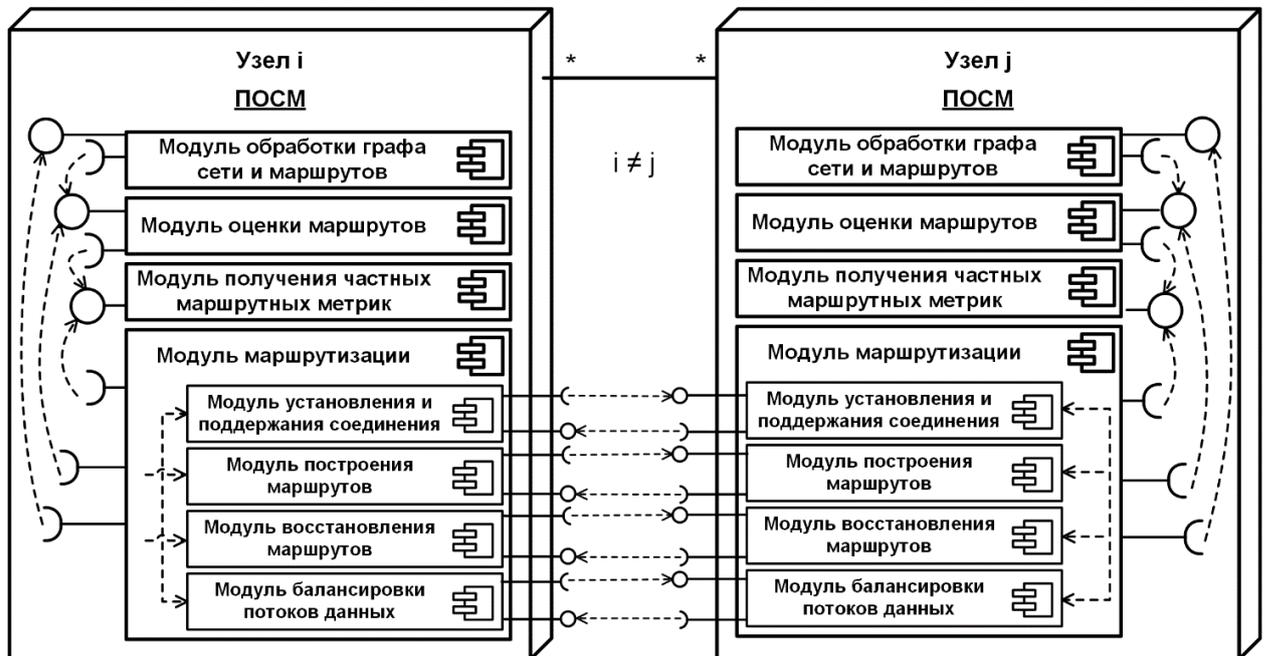


Рисунок 3.7 – Архитектура программного обеспечения взаимодействия абонентов ДСПД при решении задач маршрутизации

Модуль маршрутизации является основным компонентом программного обеспечения. В нём реализуются предложенные в работе методы и алгоритмы взаимодействия сетевого ПО маршрутизации трафика. Каждый из дочерних компонентов модуля предназначен для решения задач маршрутизации, описанных в разделе 1.1.

Модуль установления и поддержания соединения с сетью реализует алгоритм взаимодействия между ПОСМ при установлении соединения с абонентами беспроводной децентрализованной сети, а также дальнейшего

поддержания соединения с ними. Взаимодействие между модулями разных абонентов осуществляется путём обмена сообщениями NeighborMessage и NeighborMessageAnswer.

Модуль построения маршрутов связи предназначен для обнаружения оптимального и альтернативного маршрутов связи от абонента-инициатора к целевому узлу при необходимости передачи трафика между ними. Взаимодействие между модулями данного типа, функционирующими на абонентах сети, осуществляется при помощи сообщений RouteSearch и RouteAnswer.

Модуль восстановления маршрутов связи предназначен для выявления неисправностей (разрывов) в маршрутах связи и исправления найденных неисправностей за счёт включения или исключения из маршрутов соседних абонентов. Модуль балансировки трафика определяет, в каких пропорциях трафик распределяется между абонентами сети. Для их вычисления он использует интерфейс модуля оценки маршрутов связи.

Модульная архитектура программного обеспечения, предлагаемая в диссертационной работе, позволяет заменять реализации отдельных модулей без его перекомпиляции. Модули установления и поддержания соединения с сетью, оценки маршрутов связи, извлечения маршрутных метрик могут иметь несколько реализаций, каждая из которых может быть использована в конкретных условиях функционирования сети.

3.3. Протокол маршрутизации на основе разработанных методов и алгоритмов взаимодействия ПОСМ узлов ДСПД

Для практической проверки разработанных методов и алгоритмов коммуникации программного обеспечения узлов ДСПД при подключении к сети, построении и восстановления маршрутов передачи данных был разработан соответствующий протокол взаимодействия сетевого программного обеспечения при решении задач маршрутизации. Для его реализации было разработано несколько сетевых процедур и конкретные форматы пакетов, при помощи которых производится коммуникационное взаимодействие по предложенным в работе алгоритмам. Протокол состоит из следующих процедур:

- установления узлом соединения с сетью – Hello;
- построения квазипараллельных маршрутов передачи данных – Route Build;
- восстановления квазипараллельных маршрутов передачи данных – Route Service;
- передачи и балансировки трафика по квазипараллельным маршрутам связи – Route Data.

3.3.1. Процедура установления связи с соседними абонентами – Hello

Процедура Hello предназначена для установления соединения с узлами ДСПД при первичном подключении к сети и его дальнейшего поддержания в актуальном состоянии, а также для формирования графа связей с соседними абонентами первого и второго порядков. Для обмена трафиком в рамках процедуры установления и поддержания соединения с сетью были разработаны форматы пакеты NeighborMessage и NeighborMessageAnswer (рис. 3.8). NeighborMessage используется для уведомления других узлов о своём присутствии в сети. NeighborMessageAnswer используется для подтверждения получения NeighborMessage соседними абонентами первого порядка.

Так как необходимость запрашивать маршруты к соседним узлам отсутствует, то пакеты с данными к ним можно отправлять напрямую. В случае необходимости построения более длинных путей передачи данных соседние абоненты используются в качестве промежуточных узлов при построении квазипараллельных маршрутов трафика к удаленным устройствам. Накапливаемая в результате работы процедуры информация хранится в таблице AbonentInformation.

3.3.2. Процедура построения квазипараллельных маршрутов связи – Route Build

Процедура Route Build предназначена для построения квазипараллельных маршрутов между заданными абонентами. Процедура функционирует согласно алгоритмам, представленным в разделе 2.2. Результаты работы шагов алгоритмов процедуры записываются в ассоциативный массив RouteMap на узлах, которых достигли пакеты RouteSearch и RouteAnswer. Начало функционирования

процедуры обнаружения маршрутов до целевого абонента до окончания её работы называется сессией. Каждая сессия имеет идентификатор, который состоит из адреса абонента-инициатора построения маршрутов и уникального идентификатора id , выданного сессии абонентом-инициатором.

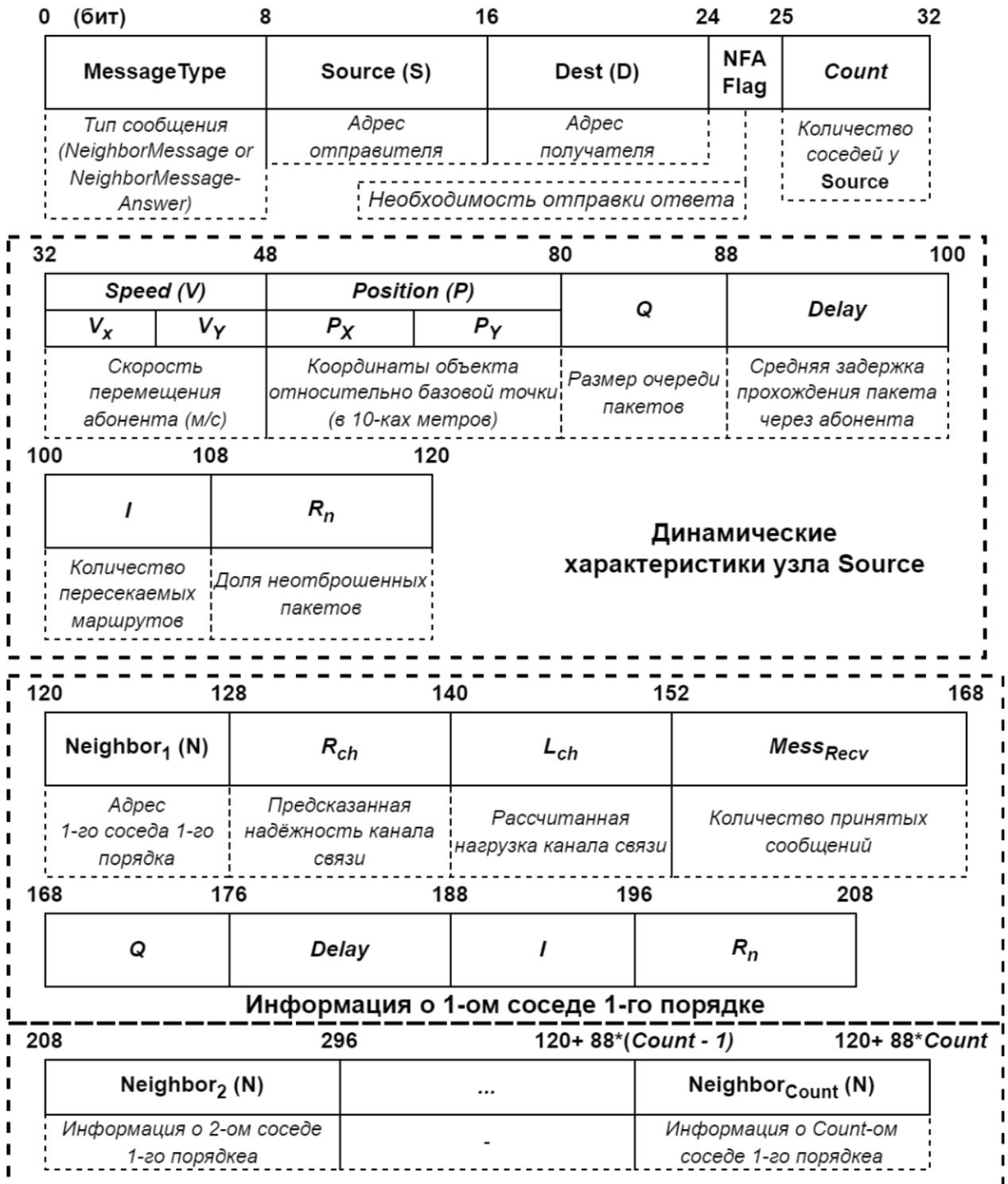


Рисунок 3.8 – Формат пакетов NeighborMessage и NeighborMessageAnswer

На первом этапе функционирования процедуры производится накопление актуальной информации о производительности сети, а также формируется децентрализованный граф пересылок между абонентами S и D . При функционировании алгоритма запроса маршрута используется пакет RouteSearch, представленный на рис. 3.9.

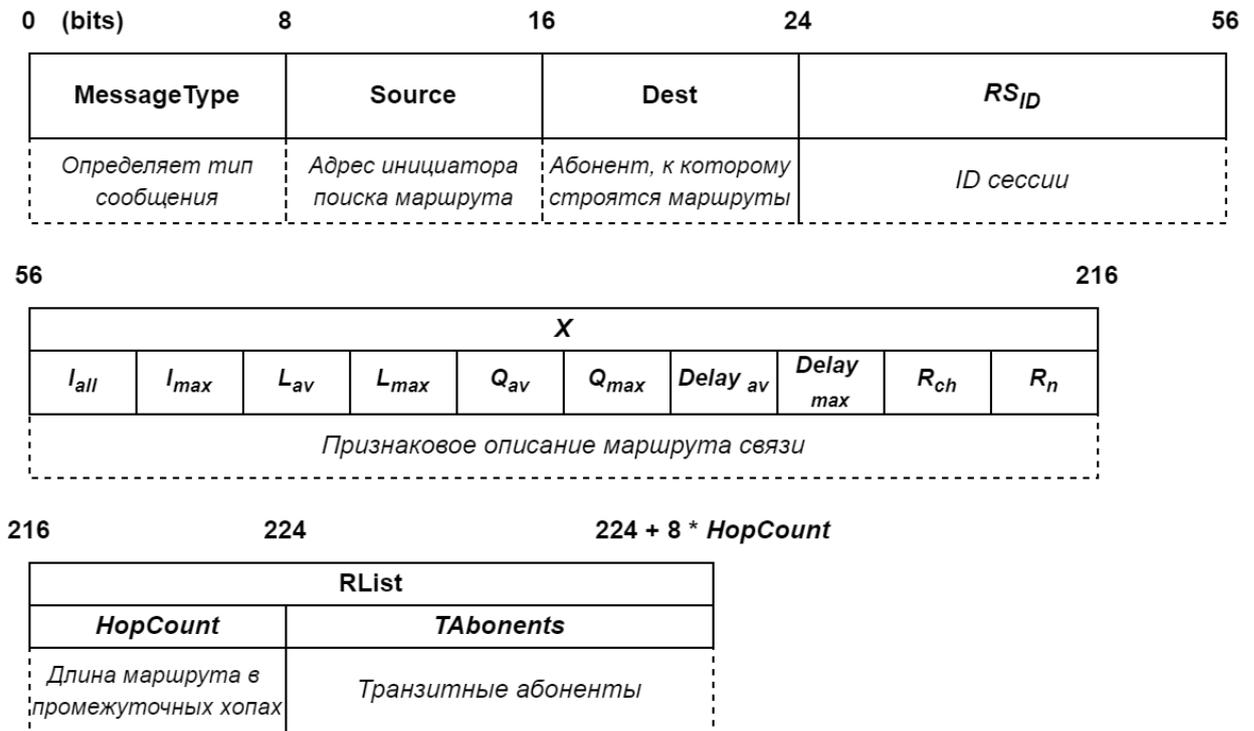


Рисунок 3.9 – Формат пакета RouteSearch

На втором этапе функционирования процедур связи используется ответный коммуникационный ответ в виде пакета RouteAnswer (рис. 3.10). Он может быть передан в адресном или широковещательном режимах. Адресная передача используется при построении основного маршрута между абонентами. Широковещательная рассылка применяется при определении альтернативных маршрутов.

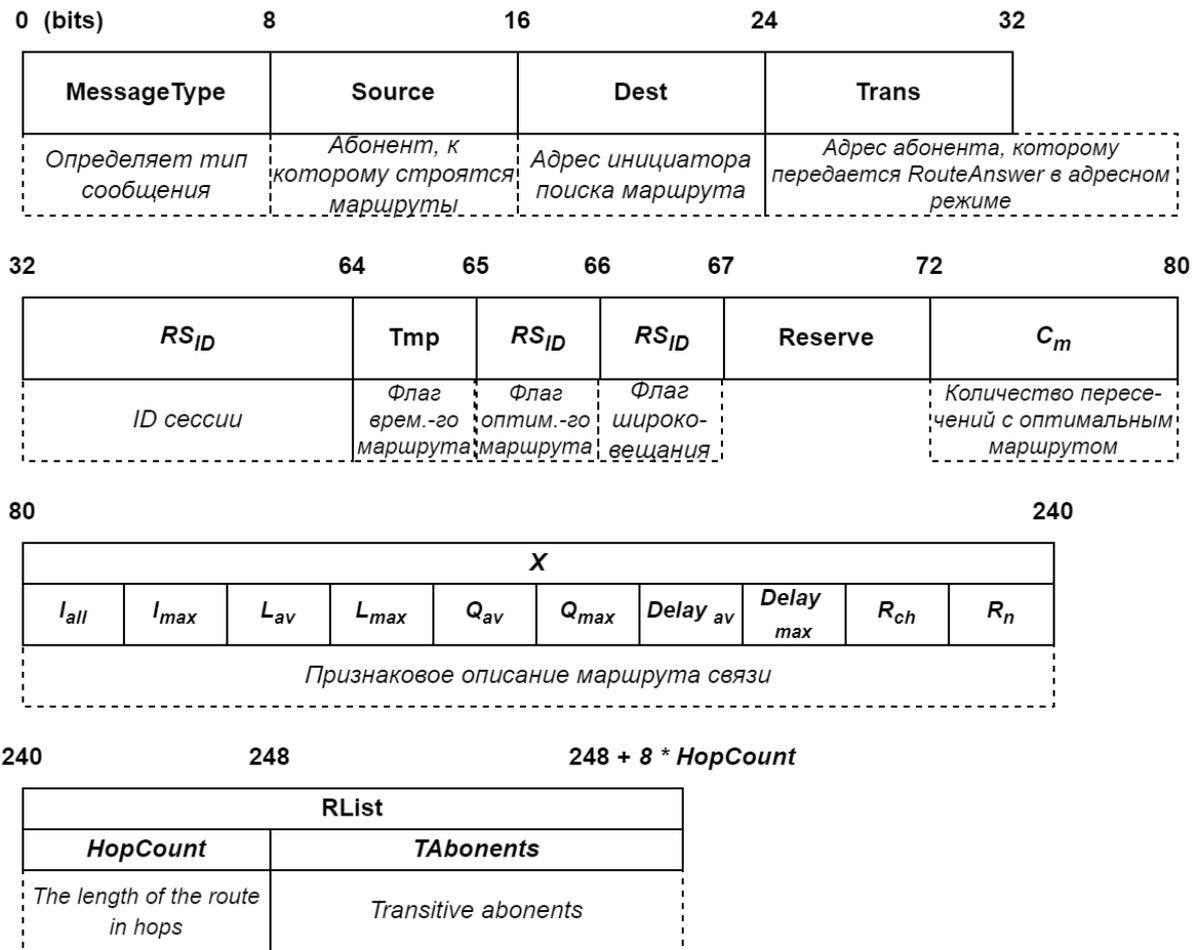


Рисунок 3.10 – Формат пакета RouteAnswer

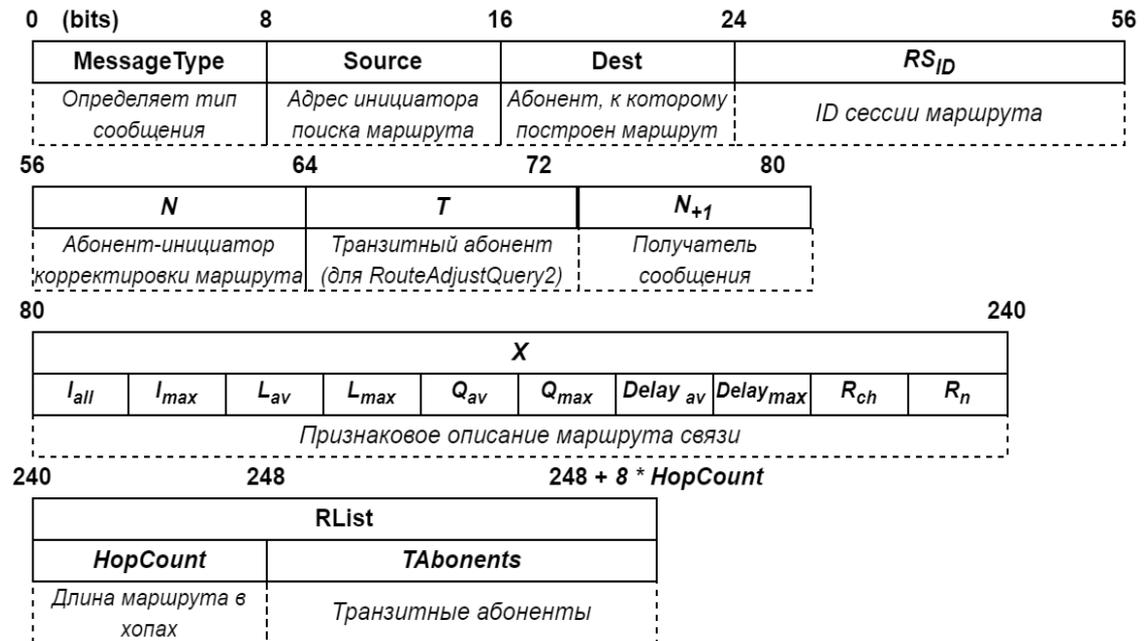
Результатом функционирования процедуры является построение основного и нескольких альтернативных маршрутов между абонентами S и D. Для передачи уведомлений о доставке сообщений используется основной маршрут.

3.3.3. Процедура восстановления маршрутов связи – Service Route

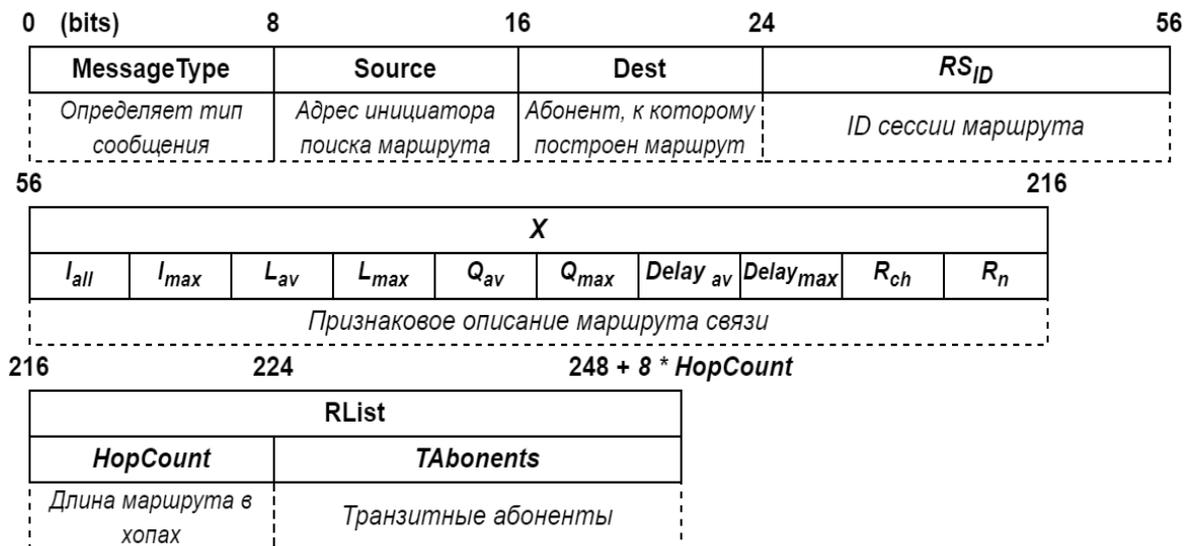
Процедура маршрутизации данных Service Route предназначена для восстановления маршрутов передачи трафика в случаях, когда произошёл разрыв связи между двумя последующими абонентами маршрута. Функционирование процедуры осуществляется в соответствии с алгоритмом, который ранее был описан в разделе 2.3. Формат сообщений, которые используются для взаимодействия ПОСМ в рамках процедуры, представлен на рис. 3.11.



а.



б.



в.

Рисунок 3.11 – Структура пакетов взаимодействия ПОСМ:
 (а) – RouteAdjustQuery и RouteAdjustQuery2, (б) – RouteAdjustAnswer и RouteAdjustAnswer2, (в) – RouteUpdate

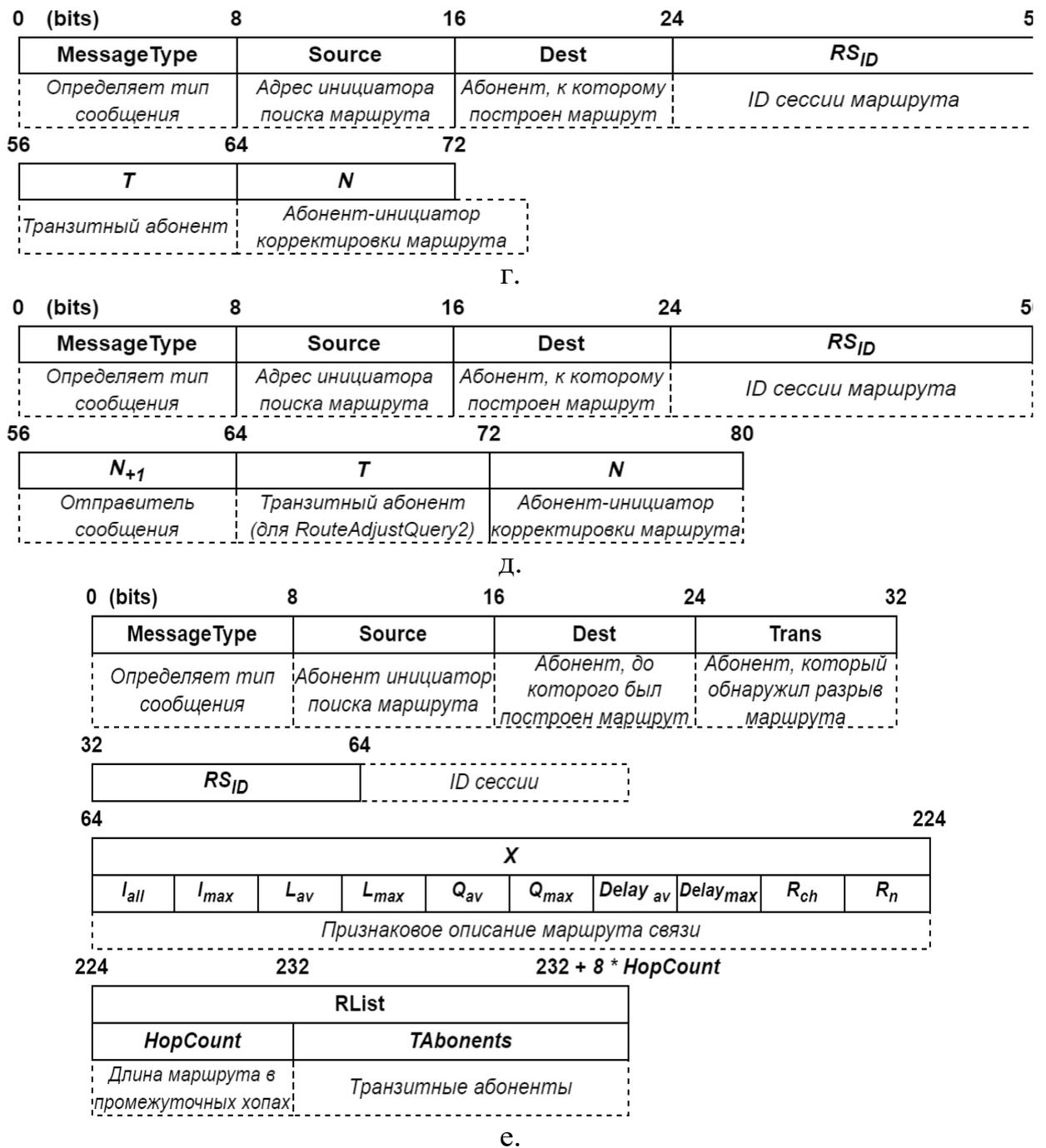


Рисунок 3.11 – Продолжение:

(г) – FaultNotify, (д) – RouteAdjustAnswerNo, (е) – RouteError

Сообщения RouteAdjustQuery и RouteAdjustQuery2 предназначены для запроса восстановления маршрута соседям первого и второго порядков соответственно. Положительный коммуникационный ответ осуществляется отправкой пакетов RouteAdjustAnswer и RouteAdjustAnswer2. В качестве отрицательного коммуникационного ответа используется сообщение RouteAdjustAnswerNo. Пакет RouteUpdate предназначен для уведомления нисходящих

абонентов маршрута об внесённых корректировках. Если текущий абонент не способен скорректировать маршрут, то он отправляет уведомление FaultNotify предыдущему в пути узлу. Когда обновить маршрут за счет соседних узлов не представляется возможным, абоненту-источнику отправляется пакет RouteError с информацией о неисправности маршрута связи. В этом случае трафик распределяется по оставшимся маршрутам. Если исправные альтернативные маршруты связи отсутствуют, производится запуск алгоритма поиска маршрута к целевому абоненту, подобно тому, как это происходит в стандартных реактивных протоколах маршрутизации. Если за максимально допустимое время построения маршрутов их не удалось обнаружить, процедура может быть повторена через заданный период.

3.3.4 Процедура маршрутизации данных –Route Data

Процедура маршрутизации данных Route Data используется для передачи трафика, и его балансировки по квазипараллельным маршрутам (если таковые имеются). Алгоритм передачи трафика был представлен ранее в разделе 2.4. Передача данных осуществляется при помощи пакетов, которые имеют формат, представленный на рис. 3.12.

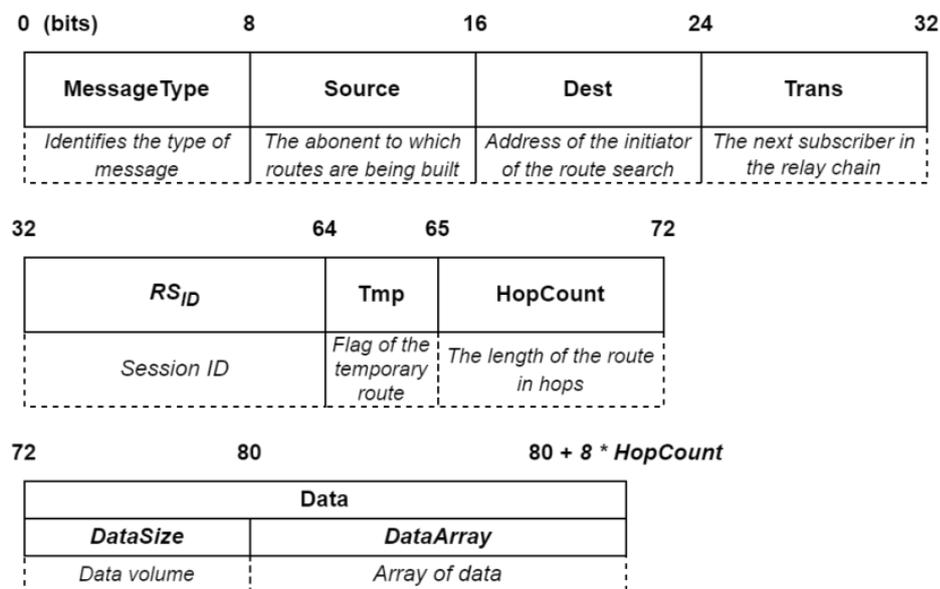


Рисунок 3.12 – Формат пакета DataRoute

Трафик распределяется по квазипараллельному маршруту в соответствии с критерием $F_B(X)$. Он определяет пропорции, в которых трафик будет

распределяться по найденным маршрутам. При этом распределение трафика по нескольким исходящим каналам осуществляют абоненты, через которые пролегают несколько квазипараллельных маршрутов, при этом имеется несколько направлений дальнейшей ретрансляции.

3.3.5 Настраиваемые параметры протокола связи

В данном разделе приведены настраиваемые параметры протокола маршрутизации.

1) HelloInterval (секунд) – период инициализации процедуры HELLO для поддержания (определения) связей с соседними абонентами. Значение по умолчанию: 10 секунд.

2) HelloAnswerInterval (секунд) – период ожидания ответа на пакет NeighborMessage, если был установлен флаг NFAFlag. Значение по умолчанию: 10 секунд.

3) NeighborOkInterval (секунд) – период ожидания ответа на сообщение NeighborOkMessage, половина периода ожидания ответа на сообщение NeighborOk2Message. Значение по умолчанию: 10 секунд.

4) RouteSearchTimeout (секунд) – период времени, в течение которого производится обнаружение маршрутов связи. Значение по умолчанию: 30 секунд.

4) ActiveRouteTimeout (секунд) – период времени, в течение которого найденный маршрут является актуальным для маршрутизации трафика. Значение по умолчанию: 60 секунд.

5) MaxHopCount (шагов) – максимальное количество шагов, которое может совершить пакет поиска маршрутов. Значение по умолчанию: 10 .

6) TimeRecvWait (секунд) – время, в течение которого абонент может принимать пакеты запросов маршрута RouteSearch с момента первого поступления пакета. Значение по умолчанию: 5 секунд.

7) TimeSendWait (секунд) – время, которое ожидает абонент перед отправкой широковещательного пакета RouteSearch после отправки адресного пакета. Значение по умолчанию: 5 секунд.

8) `IncomingRouteCount` – максимальное количество входящих маршрутов, которое может проходить через узел. Значение по умолчанию: 3.

9) `OutgoingRouteCount` – максимальное количество исходящих маршрутов, которое может проходить через узел. Значение по умолчанию: 3.

10) `KNS` – максимально допустимая доля абонентов, на которых могут пересекаться квазипараллельные маршруты связи. Значение по умолчанию: 0,5.

3.4. Программная реализация протокола маршрутизации между абонентами ДСПД

В соответствии с предложенной в п. 3.2 архитектурой было разработано программное обеспечение взаимодействия абонентов беспроводной децентрализованной сети связи для передачи трафика. Каждый модуль реализован через паттерн «одиночка» [106], и для получения их интерфейса необходимо использовать метод `getInstance()` классов `*Factory` соответствующих модулей.

Диаграмма классов, при помощи которых реализуется модуль оценки маршрутов, связи представлена на рис. 3.13. `IFitnes` предоставляет интерфейс функций $F_S(X)$ и $F_B(X)$. В диссертационной работе для оценки маршрутов связи использовалось 3 подхода и в соответствии с этим использовалось три реализации интерфейса *IFitnes*:

- аналитические формулы;
- градиентный бустинг деревьев принятий решений;
- обобщённая регрессионная нейронная сеть.

Для аналитической оценки маршрутов связи используется класс `AnalyticFitnes`. Класс `BoostFitnes` реализует функции, которые используют подход градиентного бустинга решающих деревьев. В классе `GRNNFitnes` описаны функции, позволяющие оценивать маршруты связи при помощи ОРНС.

Для оценки маршрутов на основе градиентного бустинга и ОРНС было разработана утилита на языке `python`. Для решения задачи регрессии при помощи

градиентного бустинга используется CatBoost [107], которая содержит готовые модели ансамбля решающих деревьев. CatBoost – открытая программная библиотека, разработанная компанией Яндекс и реализующая уникальный патентованный алгоритм построения моделей машинного обучения, использующий одну из оригинальных схем градиентного бустинга. Для ОРНС используется реализация регрессионной модели из библиотеки NeuPy [108]. NeuPy – это библиотека python3 для проектирования и построения нейронных сетей, вычисления в которой производятся при помощи Tensorflow.

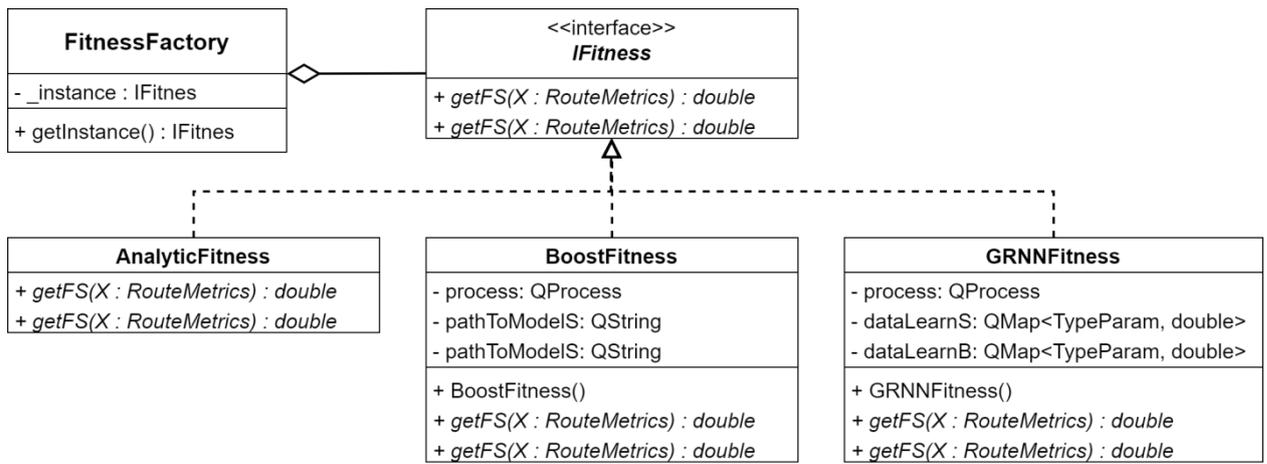


Рисунок 3.13 – Диаграмма классов, реализующих функции оценки маршрутов

Для запуска кода, написанного на языке программирования python3, используется класс QProcess. При этом на ЭВМ должен быть установлен интерпретатор python3, а также библиотеки CatBoost и NeuPy. Используя метод `start (python3)`, включается интерпретатор python3, после чего имеется возможность передавать в него команды при помощи метода `write (command)`. Для чтения данных используется `readAllStandartOutput ()`.

В конструкторе класса BoostFitnes производится подключение необходимых библиотек, а также инициализация регрессионной модели:

```

from catboost import CatBoostRegressor
import pandas as pd
model = CatBoostClassifier()
model.load_model("path to model")
  
```

Метод `load_model()` загружает модель, которая ранее была обучена и сохранена при помощи метода `save_model()`. При этом имеются две отдельные модели для двух фитнес-функций. Для предсказания в методах `getFS(X)` и `getFB(X)` используется следующий код:

```
X = pd.array([[%1, %2, %3, %4, ..., %N]])
Y = model.predict(X)
print(Y[0])
```

Метод `print()` выводит предсказанное значение, которое считывается при помощи метода `readAllStandartOutput()`. Для сети ОРНС используется библиотека `NeuPy`. В конструкторе класса производится подключение библиотеки:

```
from neupy import algorithms
```

Обучение ОРНС не требует использования метода обратного распространения ошибки и производится в один проход, поэтому для инициализации модели в классе используется функция обучения. Для получения корректных результатов, входные переменные должны подаваться на вход ОРНС в нормированном виде:

```
pca = StandardScaler()
X_train = pca.fit(X_base_train)
model = algorithms.GRNN(std=1, verbose=False)
model.train(X_train, Y_train)
```

В отличие от градиентного бустинга подаваемые на вход методу `predict(X)` входные переменные должны быть нормированными для получения корректных результатов:

```
X = pd.array([[%1, %2, %3, %4, ..., %N]])
X_norm = pca.transform(X)
Y = model.predict(X_norm)
print(Y[0])
```

Интерфейс доступа к модулю графа сети и маршрутов связи реализован при помощи класса `AbonentInformation`. Его экземпляр доступен через метод

`getInstance()` класса `AbonentInformationFactory`. Диаграмма классов модуля представлена на рис. 3.14. Каждая из сессий поиска маршрутов связи (класс `Session`) использует интерфейс `IFitness` для сортировки частично-сформированных маршрутов связи (`ListInput`).

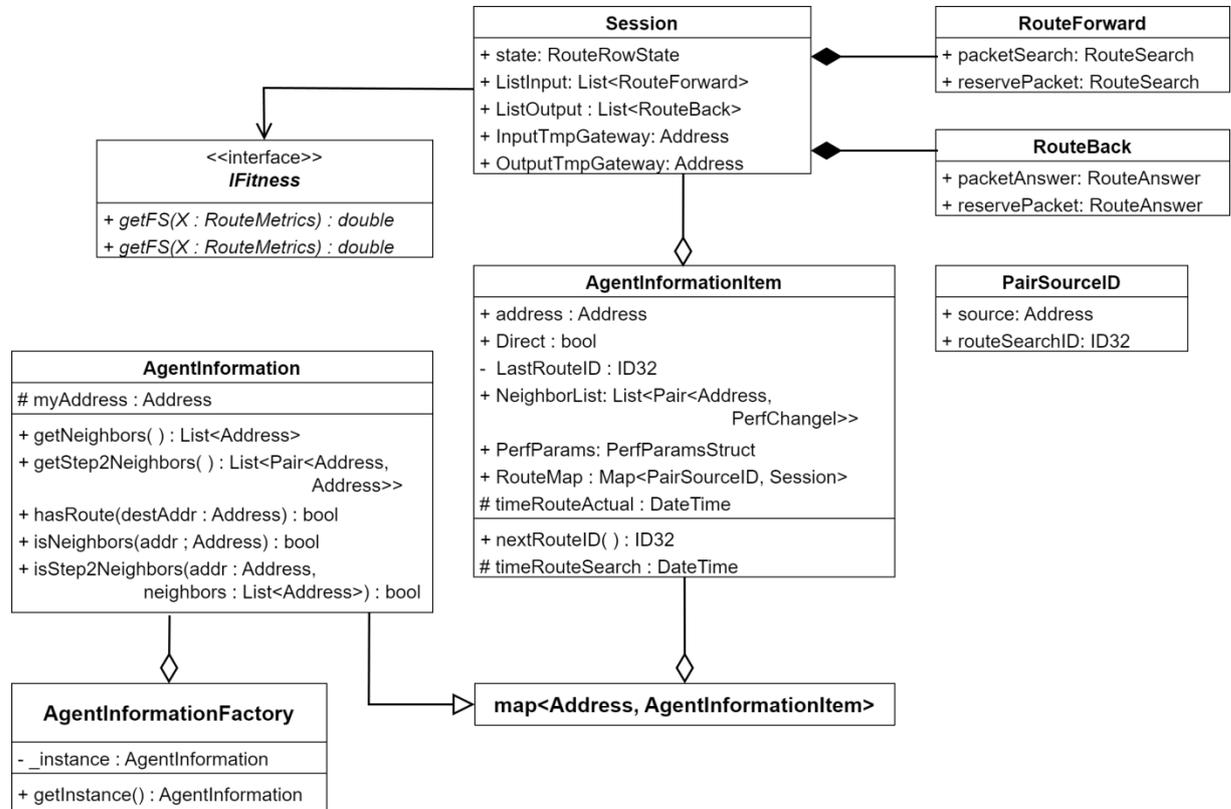


Рисунок 3.14 – Диаграмма классов, реализующих функции оценки маршрутов

`AbonentInformation` представляет собой таблицу, каждая запись которой соответствует абоненту сети ДСПД. Первичными ключами в таблице `AbonentInformation` являются адреса абонентов, с которыми устанавливается связь, а элементы представляют собой тип `AbonentInformationItem`. Каждая запись содержит:

- `Address address` – адрес абонента, соответствующего текущей записи;

- `bool Direct` – флаг установления прямого соединения с текущим абонентом. Если `connected == true`, то абонент является соседним абонентом для текущего;

- `List<Pair<Address, PerfChannel>> NeighborsList` – список соседних узлов для абонента из текущей записи. Если в списке содержится адрес текущего абонента, значит связь – двухсторонняя;

– `DateTime lastRecvHelloTime` – время, когда в последний раз было получено сообщение `NeighborMessage`. Используется для проверки актуальности доступности абонента, соответствующего текущей записи;

– `Map<PairSourceID, Session> RouteMap` – структура, хранящая информацию о многопутевых маршрутах к абоненту, соответствующему текущей записи. `RouteMap` является ассоциативным массивом из пар `RouteSearchID` и `RowSearch`.

Структура `RouteSearchID` содержит пару `Address source` и `ID32 routeSearchID`. `source` содержит адрес абонента, инициализировавшего поиск маршрута к абоненту из записи, `routeSearchID` – идентификатор поиска маршрута. Если поле `source` содержит адрес текущего абонента, то он является инициатором поиска маршрута. В ином случае текущий абонент является ретранслятором при передаче данных от `source` к абоненту, соответствующему текущей записи `AbonentInformationItem`.

Каждый элемент `Session` содержит:

– `RouteRowState State` – состояние поиска маршрута. Используется, если поиск маршрута был инициирован текущим абонентом;

– `List<RouteForward> INPUT` – список входящих пакетов `RouteSearch` и его резервной копии текущей операции поиска маршрута. Если текущий абонент инициатор поиска – список всегда пуст;

– `List<RouteBack> OUTPUT` – список входящих пакетов `RouteAnswer` и его резервной копии. Используется при маршрутизации данных между абонентами сети. Если текущий абонент – адрес назначения поиска маршрута, список всегда пуст;

– `Address INPUT_TMP` – адрес, с которого приходят пакеты по временному маршруту;

– `Address OUTPUT_TMP` – адрес, являющейся следующим в цепочке ретрансляции временного маршрута.

На рис. 3.15 представлены интерфейс и его реализация для модуля извлечения маршрутных метрик, который предназначен для получения показателей производительности собственного узла и каналов связи со смежными абонентами. В текущей работе используется одна реализация интерфейса *IMetricsSource*.

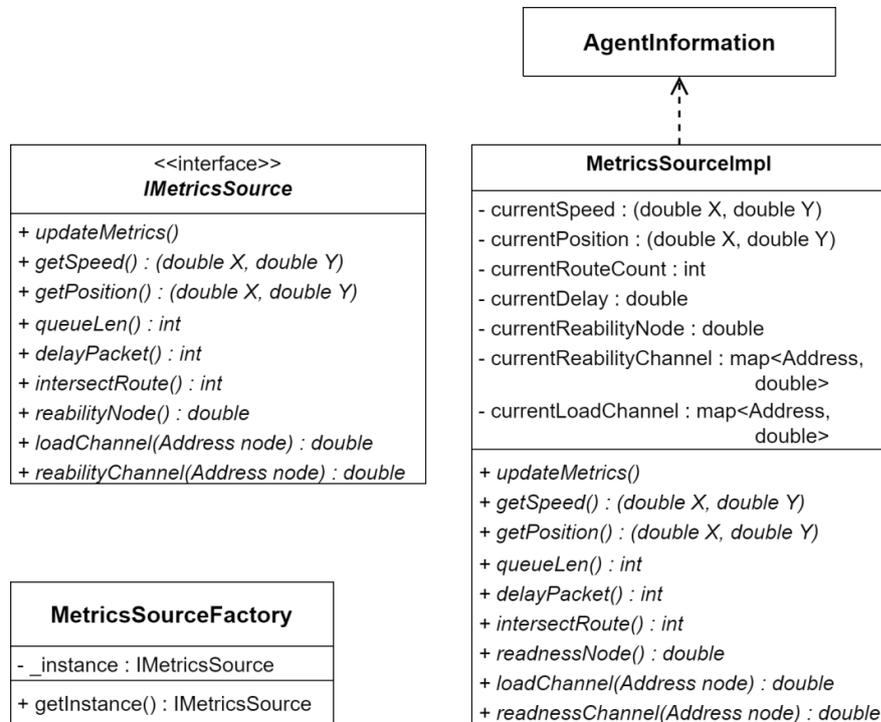


Рисунок 3.15 – Диаграмма классов протокола связи

На рис. 3.16 представлена группа классов, при помощи которых был непосредственно реализован модуль программного обеспечения сетевой маршрутизации и его подмодули. Класс *ProtocolModule* реализует подмодуль передачи и балансировки трафика по сети. Тип данных *NeighborsModule* предназначен для установления и поддержания соединения с сетью, а именно, с соседними абонентами первого и второго порядков. *RoutingModule* реализует алгоритмы взаимодействия абонентов связи для построения оптимального и параллельного маршрутов связи. Алгоритмы взаимодействия ПОСМ при восстановлении маршрута связи реализованы при помощи класса *RepairModule*.

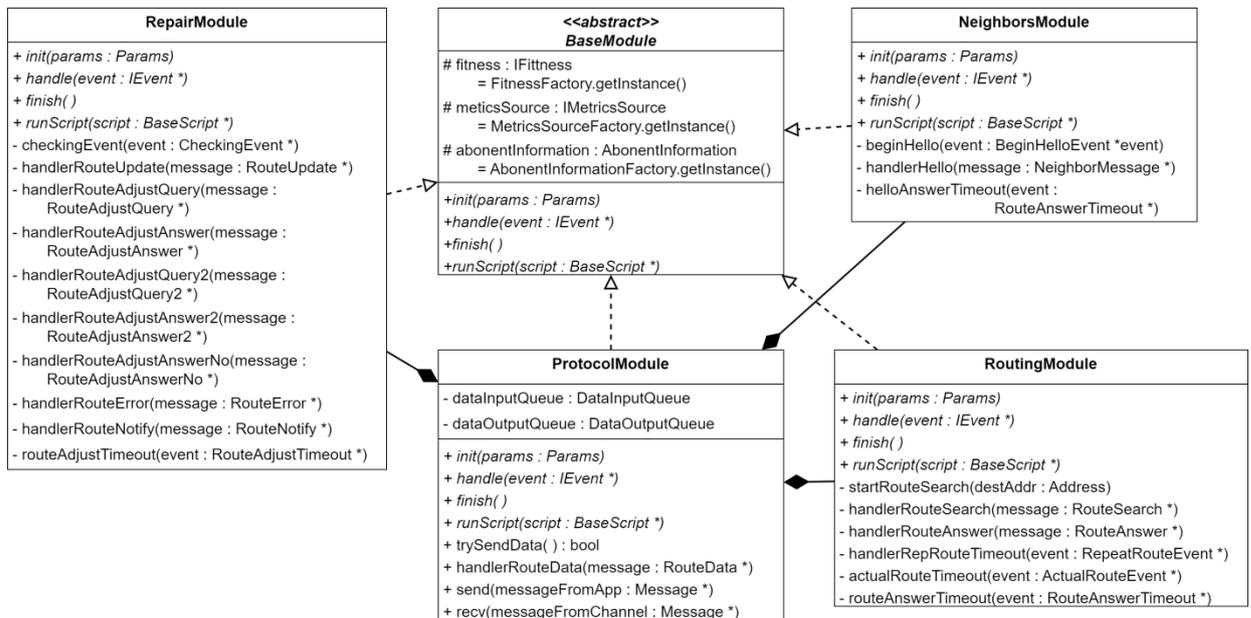


Рисунок 3.16 – Диаграмма классов протокола связи

Базовым классом является `BaseModule`, реализующий общие свойства и предоставляющий интерфейс для наследующих его модулей. Среди свойств можно выделить ссылки на интерфейсы модулей оценки маршрутов связи (`fitness`), извлечения маршрутных метрик (`metricSource`), таблицы с данными о соседних абонентах и маршрутах связи (`abonentInformation`). Были определены следующие интерфейсные методы:

- `init(Params params)` – используется для инициализации модуля параметрами (в частности, параметра текущего протокола маршрутизации);

- `finish()` – используется при завершении процесса функционирования модуля. В данном методе может быть агрегирована статистика по работе модуля с момента его инициализации;

- `handle(IEvent *event)` – используется для обработки событий. В частности, к таким событиям относится запуск таймеров для различных процедур;

- `runScript(BaseScript *script)` – используется для запуска сценариев на текущем модуле (например, сценарий инициализации поиска маршрута).

В методах класса `ProtocolModule` производится обработка входящих и исходящих сообщений абонента сети, на котором он функционирует, и поэтому класс содержит соответствующие свойства:

– `DataInputQueue dataInputQueue` – очередь входящих сообщений от внешних узлов;

– `DataOutputQueue dataOutputQueue` – очередь исходящих сообщений от модуля приложения.

3.5. Выводы по главе

В текущем разделе были достигнуты следующие результаты:

1. Разработана структура программного обеспечения для моделирования взаимодействия абонентов в ДСПД. Выполнена программная реализация специализированного инструмента для исследования взаимодействий программного обеспечения сетевой маршрутизации, которая позволяет осуществлять имитационное моделирование функционирования сети в нескольких режимах, включая режим визуализации взаимодействия сетевого ПО по заданным алгоритмам, и предоставляет возможность исследовать, отлаживать и производить оценку разрабатываемых алгоритмов.

2. Разработана архитектура программного обеспечения сетевой маршрутизации для реализации взаимодействия отдельных абонентов ДСПД при решении соответствующих задач, также созданы интерфейсы для их децентрализованной кооперации с целью обеспечения взаимодействия абонентов при балансировке трафика в беспроводных децентрализованных сетях. Предложенная архитектура отличается модульной структурой, которая позволяет заменять реализации отдельных модулей без необходимости перекомпиляции программного обеспечения для исследования различных методов и протоколов маршрутизации трафика.

3. Разработанные в диссертации методы и алгоритмы реализованы в виде протокола связи прикладного уровня. При его практической реализации дополнительно были предложены форматы пакетов, при помощи которых осуществляется взаимодействие программного обеспечения сетевой маршрутизации узлов ДСПД.

4. Спроектированы классы, при помощи которых может быть реализован протокол маршрутизации трафика на основе предложенных в диссертации методов и алгоритмов. Особенностью разработанной схемы является использование

интерфейсов IFitnes и IMetricsSource. IFitnes предоставляет методы, которые соответствуют ранее определённым функциям для оценки качества маршрутов. IMetricsSource предоставляет методы, которые позволяют получить текущие параметры для ДХСУ и ДХКС. Использование предложенной схемы позволяет гибко настраивать реализации функций оценки качества функционирования маршрутов и функционала по получению показателей ДХСУ и ДХКС, не внося изменений в исходный код модуля сетевого ПО.

5. На разработанное программное обеспечение были получены свидетельства о государственной регистрации программы для ЭВМ «Программа для определения оптимальных значений параметров протоколов верхнего уровня в ad-hoc сетях», «Программное обеспечение для маршрутизации трафика в беспроводных децентрализованных сетях связи», «Программа для определения оптимальных значений параметров протоколов верхнего уровня в ad-hoc сетях», «Программное обеспечение для моделирования взаимного пространственного перемещения отдельных узлов в беспроводных децентрализованных сетях связи».

6. Использование разработанного программного инструмента позволяет проводить моделирование и исследование различных протоколов и механизмов взаимодействия сетевого ПО абонентов для оценки показателей их функционирования, в частности для определения временных характеристик и процента потерянных пакетов трафика. В рамках разработанного программного средства можно производить сравнительную оценку предложенного в диссертации решения со стандартными методами и протоколами маршрутизации для определения более эффективного подхода.

4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ РАЗРАБОТАННЫХ МЕТОДОВ И АЛГОРИТМОВ ВЗАИМОДЕЙСТВИЯ СЕТЕВОГО ПО ПРИ МАРШРУТИЗАЦИИ ТРАФИКА

Настоящий раздел посвящен исследованию корректности предложенных в работе методов и алгоритмов взаимодействия программного обеспечения сетевой маршрутизации и экспериментальному подтверждению, что их практическая реализация позволяет обеспечить улучшение целевых показателей качества обслуживания в ДСПД в различных сценариях их функционирования: при установлении и дальнейшем поддержании соединения с сетью, построении квазипараллельных маршрутов между абонентами, передаче трафика и его балансировке по найденным путям, а также при восстановлении маршрутов во время разрывов каналов связи. В качестве основного инструмента, при помощи которого производились имитационное моделирование и эксперименты по исследованию коммуникации в сетях, использовалось разработанное программное средство моделирования ДСПД. Предложенные методы и алгоритмы взаимодействия программного обеспечения сетевой маршрутизации могут считаться корректными, если успешно решаются задачи:

- установления и дальнейшего поддержания соединения с сетью;
- построения квазипараллельных маршрутов связи между целевыми абонентами;
- восстановления маршрутов связи при возникновении неисправностей в них;
- передача трафика и его балансировка по найденным маршрутам.

Для достижения поставленной цели работы необходимо, чтобы в результате применения предложенных методов и алгоритмов на практике обеспечивалось снижение сквозной задержки и повышение коэффициента доставки пакетов, и как итог, улучшение значения обобщённого критерия эффективности $\text{Delay}_{\text{Reserve}}$. Для экспериментальной проверки разработанных методов и алгоритмов была произведена реализация протокола маршрутизации (п. 3.3), в состав которого они входят. Характеристики его функционирования сравнивались с соответствующими

параметрами, полученными для протокола AOMDV. Выбор данной технологии обусловлен тем, что она является в настоящее время основной при построении и развертывании ДСПД в реальных режимах их работы. Существующие модификации AOMDV существенных отличий от базовой версии протокола не имеют и используются только в частных сценариях функционирования сетей.

Экспериментальное исследование включает четыре этапа:

1. На первом производится оценка корректности алгоритмов взаимодействия программного обеспечения сетевой маршрутизации, обеспечивающих решение задачи установления соединения с сетью, обнаружения маршрутов и балансировки трафика по ним. Для этого использовались сценарии со статичной топологией сети. В качестве функций оценки маршрутов передачи данных и весов путей использовались аналитические выражения.

2. Вторым этапом экспериментального исследования посвящён процессу обучения моделей градиентного бустинга и обобщённой регрессионной нейронной сети для расчёта показателей эффективности маршрутов передачи трафика.

3. На третьем этапе проведена серия экспериментов, в ходе которой исследовались коэффициент доставки пакетов данных, а также сквозная задержка передачи большого фрейма данных до целевого абонента в зависимости от количества абонентов сети, используемых протоколов связи и функций оценки маршрутов.

4. Четвёртый этап включает проверку эффективности алгоритма взаимодействия сетевого ПО при восстановлении маршрутов, сравнение временных задержек, возникающих в процессе восстановления путей с аналогичным процессом протокола AOMDV, а также определение степени улучшения показателей надёжности и сквозной задержки доставки данных.

4.1. Экспериментальная проверка корректности функционирования методов и алгоритмов взаимодействия сетевого ПО

В ходе экспериментального исследования производилась проверка корректности функционирования предложенных в статье методов и алгоритмов взаимодействия ПОСМ при установлении соединения с сетью, построении маршрутов и балансировки трафика по ним. Корректность процесса установления соединения

с сетью может быть подтверждена фактом формирования ДСПД из абонентов, между которыми имеется прямое соединение, и их участие в процедурах поиска маршрута и передачи трафика. Построение квазипараллельных маршрутов и передача трафика по ним является свидетельством корректного функционирования алгоритмов взаимодействия сетевого программного обеспечения при решении указанных задач. При этом использовались сценарии с относительно низкой скоростью передачи трафика. Аналогичные эксперименты были проведены для протокола AOMDV.

В качестве функций оценки маршрутов связи использовались аналитические функции [109] 4.1 и 4.2:

$$F_S(X) = \frac{R_{ch} R_n}{0,1 \log \left(\frac{Delay_{qmax}}{Delay_{av}} \right) + 1}, \quad (4.1)$$

$$F_{BA}(X) = \frac{R_{ch} R_n}{Delay_{max}}, \quad (4.2)$$

где $F_{BA}(X)$ – вариант функции $F_B(X)$, выраженный аналитически, $Delay_{qmax}$ – максимально допустимое время пребывания пакета в очереди. $F_S(X)$ использует мультипликативную свёртку показателей надёжности маршрутов ($R_{ch} * R_n$). Для снижения веса маршрутов с высокими задержками используется знаменатель выражения. Значение $F_{BA}(X)$ увеличивается для маршрутов с наименьшей максимальной задержкой при передаче пакетов по пути.

В качестве критериев качества формируемых маршрутов, которые исследовались в описываемой серии экспериментов, использовались следующие:

- PDR (packetdeliveryratio) – доля доставленных пакетов данных до целевого абонента. Значение критерия показывает надёжность маршрута;
- E2ED (end-to-enddelay) – сквозная задержка передачи данных;
- E2EDG (E2EDgroup) – общее время передачи по квазипараллельным маршрутам группы пакетов в составе потока.

В текущем исследовании скорость передачи данных принималась равной 8 кб/с, что соответствует низкоскоростным средствам УКВ радиосвязи [110] с

низкой мощностью сигнала. $Delay_{qmax} = 10$ секунд. Прикладной трафик между абонентами составил 8 пакетов секунду, каждый из которых имел объём 255 кб (2 кб/с). Пакеты передавались потоком по 255 пакетов. Аналогичный трафик генерировался другими абонентами и рассылался случайным узлам сети. Другие параметры моделирования представлены в табл. 4.1. Уровень надёжности каналов связи и среднее время задержки пакетов при их обработке абонентами в одних случаях задавались статически перед началом функционирования сети в первых двух экспериментах, а в третьем определялись взаимным пространственным расположением абонентов и генерируемым трафиком. В текущем сценарии абоненты неподвижны, наличие прямых связей между абонентами задавалось сценарием.

Таблица 4.1 – Параметры моделирования

Параметр	Значение
Максимальная длина маршрута	5, 6, 7, ..., 29, 30
Скорость передачи данных	8 Кб/с.
Трафик между абонентами	2 Кб/с.
Максимальная скорость движения абонентов	15 м/с.
Модель движения абонентов	Модель перемещений на основе случайных точек
Радиус радиопередачи	400 м.
Размер карты	1500 x 1500 м ²
Ограничение по виртуальному времени моделирования	1200 с.

В первом эксперименте использовалась топология сети А (рис. 4.1), в которой между абонентом узлом-источником S и целевым абонентом D имеется только три доступных маршрута. В результате функционирования алгоритма взаимодействия ПОСМ при поиске маршрутов нижний участок был определён как основной маршрут, другие как альтернативные. Результаты экспериментального исследования представлены в табл. 4.2.

Во втором эксперименте использовалась топология В с множеством связей между абонентами (рис. 4.2). Визуально сложно наглядно отобразить, как могут быть построены маршруты между абонентами S и D, но в результате работы предложенных алгоритмов были получены маршруты, обеспечивающие повышение надёжности доставки данных. Результаты эксперимента представлены в табл. 4.3. При этом вероятность построения альтернативных параллельных путей

составляла от 3% до 9%, ввиду чего отсутствовала возможность произвести балансировку трафика стандартными методами. В обоих случаях демонстрируется организация абонентов в единую ДСПД. Было получено подтверждение корректности функционирования процедур, отвечающих за построение маршрутов связи и передачи трафика по ним, и обеспечивающих их работу алгоритмов взаимодействия программного обеспечения сетевой маршрутизации. Кроме того, существенно были улучшены показатели PDRи E2EDG по сравнению со стандартным протоколом AOMDV.

Таблица 4.2 – Результаты экспериментального исследования для топологии сети А

Стандартный подход AOMDV		Предложенный подход						
E2ED (с)	PDR	Маршрут	E2ED (с)	E2EDG (с)	PDR	FS(X)	FB(X)	Доля трафика
18,12	0,361	Осн.	25,49		0,634	0,586	4,374	0,399
E2EDG (с)		Альт. 1	28,15		0,392	0,384	3,326	0,304
		Альт. 2	19,53		0,363	0,343	3,254	0,297
2276		В целом	25,23	760	0,467			1

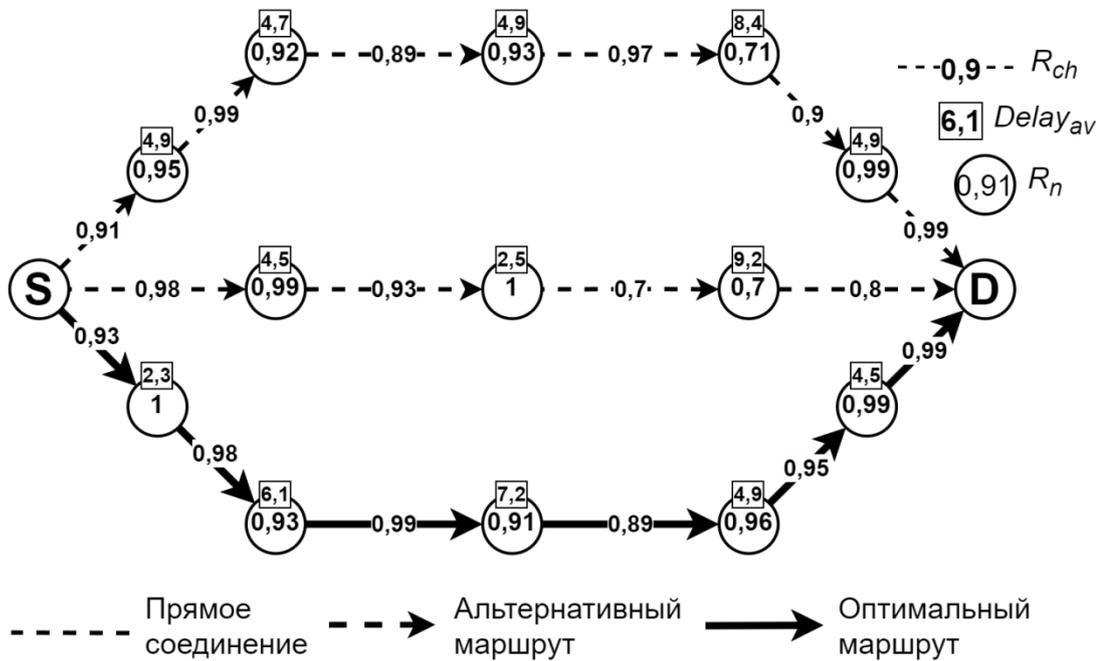


Рисунок 4.1 – Топология сети А и значения ДХКС и ДХСУ каналов связи и абонентов

Таблица 4.3 – Результаты экспериментального исследования
для топологии сети В

Стандартный подход AOMDV		Предложенный подход						
E2ED (с)	PDR	Маршрут	E2ED (с)	E2EDG (с)	PDR	$F_S(X)$	$F_B(X)$	Доля трафика
18,12	0,361	Осн.	24,53		0,87	0,815	5,186	0,400
E2EDG (с)		Альт. 1			0,55	0,431	3,411	0,263
		Альт. 2	27,57		0,61	0,586	4,374	0,337
2312		В целом	26,81	659	0,69			1

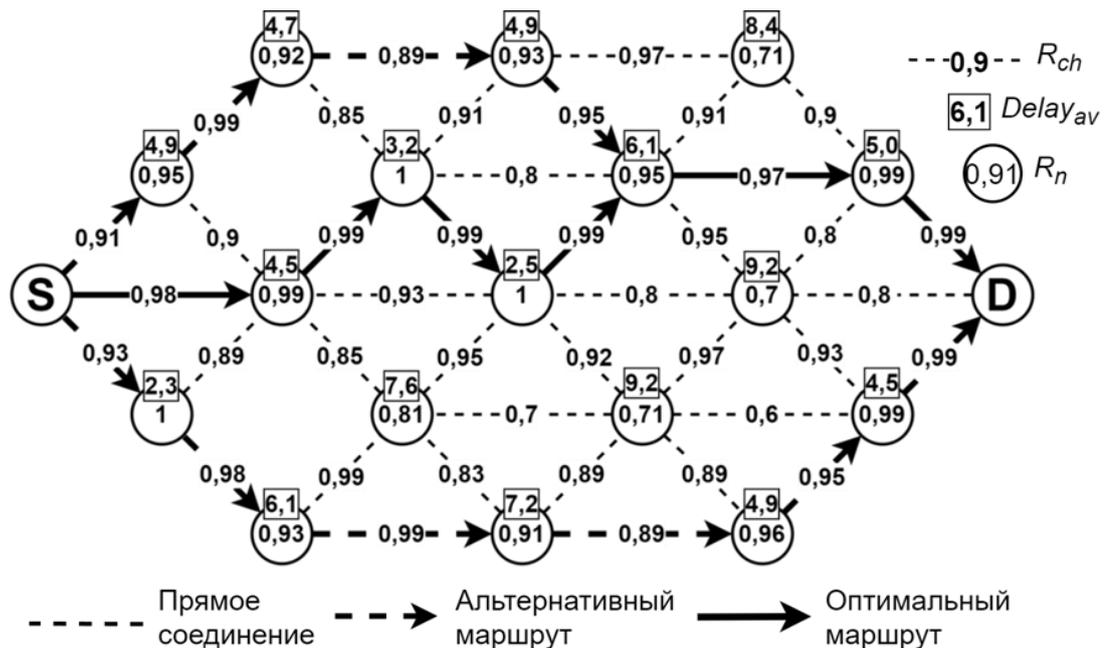


Рисунок 4.2 – Топология сети В и значения ДХКС и ДХСУ
каналов связи и абонентов

В третьем эксперименте, согласно сценарию, использовалась ячеистая топология С (рис. 4.3), выстроенная прямоугольными ячейками. Перед началом функционирования сети случайным образом задавались значения надёжности каналов связи R_{ch} и прохождения трафика через абонента R_n . Определялась зависимость показателей PDR и E2EDG от количества узлов сети. Результаты экспериментальных исследований представлены на рис. 4.4 и рис. 4.5. Они демонстрируют, что предложенные алгоритмы позволили повысить надёжность доставки данных в среднем на 5%, а сквозную задержку снизить в среднем в 3 раза в зависимости от количества образовавшихся квазипараллельных маршрутов. При этом формировалось 3-4 квазипараллельных маршрута, между которыми распределялся трафик,

что позволило снизить количество потерь пакетов и их сквозную задержку их доставки.

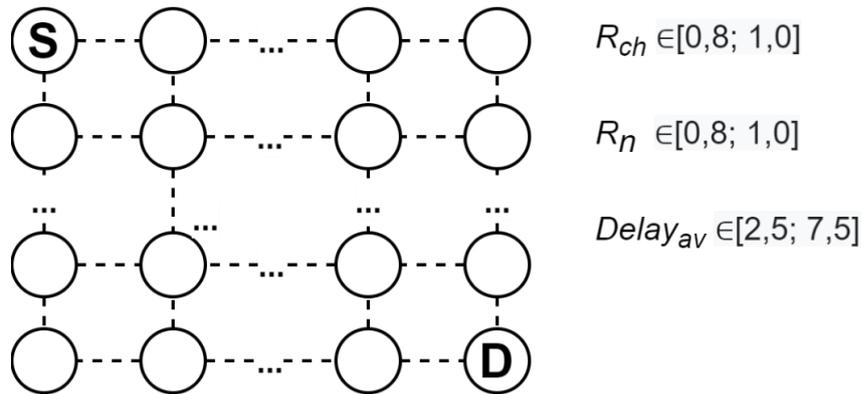


Рисунок 4.3 – Топология сети С и значения ДХКС и ДХСУ каналов связи и абонентов

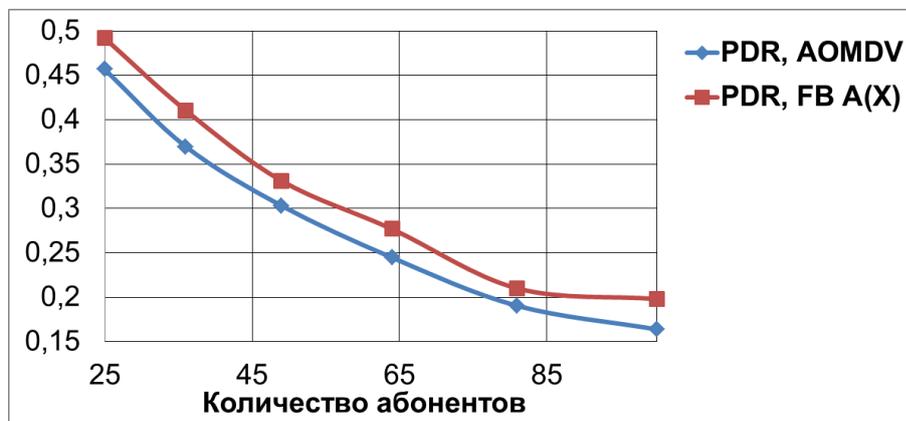


Рисунок 4.4 – Зависимость показателя PDR от количества абонентов в сценарии с топологией С

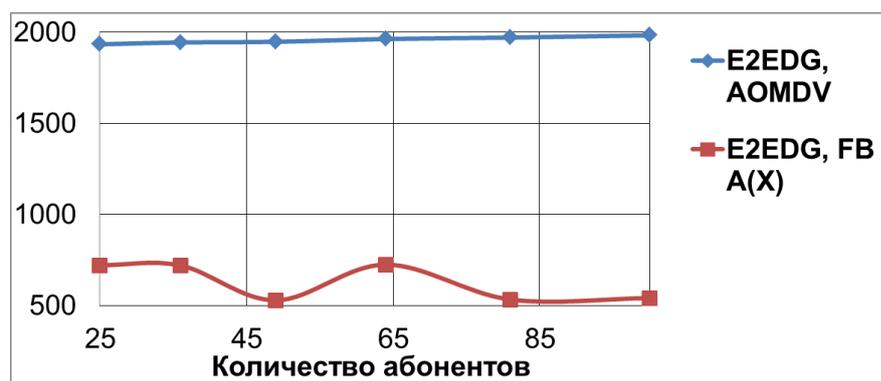


Рисунок 4.5 – Зависимость показателя E2EDG от количества абонентов в сценарии с топологией С

Проведённые эксперименты показали корректность функционирования процедур установления соединения с ДСПД и дальнейшего его поддержания, обнаружения маршрутов и передачи трафика по ним. За счет новых методов и

алгоритмов взаимодействия абонентов сети удалось повысить надёжность доставки данных (показатель PDR) по сравнению с применяемыми в настоящее время на практике подходами. В первую очередь это обусловлено тем, что в процессе обмена пакетов построения маршрутов ПОСМ исключали каналы связи с высокими потерями информационных пакетов и перегруженных абонентов ДСПД. При этом общая длина маршрутов возросла, что повлекло увеличение времени доставки отдельных пакетов (показатель E2ED). Однако за счёт передачи трафика по квазипараллельным путям удалось уменьшить время доставки потока данных, фрагментированного на отдельные пакеты.

4.2. Применение и настройка алгоритмов машинного обучения для оценки качества маршрутов связи

В качестве обобщённого критерия эффективности маршрута F_S , вычисляемого при помощи алгоритмов регрессии, используется формула 4.3, основанная на функции (2.1) представляющее свёртку критериев PDR и E2ED:

$$F_S = \frac{PDR^{0,7}}{E2ED^{0,3} / E2ED_{norm}}, \quad (4.3)$$

где в качестве весов K_{S1} , K_{S2} используются значения 0,7 и 0,3 соответственно. Выбор весовых коэффициентов произведён в соответствии с методикой [111]. Приоритет отдаётся маршрутам с более высоким значением доли доставляемых пакетов до целевого абонента.

В ходе работы при помощи созданной программной платформы на основе симулятора Omnet++ был собран набор данных. Каждая запись в наборе представляет массив частных метрик X пути, полученных в процессе его построения и является признаковым описанием маршрута. Параметры моделирования аналогичны тем, что были представлены в табл. 4.1. Кроме того, каждая запись содержит значения частных метрик PDR и E2ED (end-to-end delay), определенных по результатам функционирования маршрута. PDR и E2ED – целевые переменные, которые требуется вычислять на основе массива частных метрик.

В ходе построения модели машинного обучения при решении задачи вычисления маршрутных метрик был выполнен линейный корреляционный анализ [112] между признаковым описанием маршрута и метриками F_S , PDR и E2ED. Результаты анализа представлены в табл. 4.4.

Таблица 4.4 – Результаты корреляционного анализа между частными метриками и критериями качества функционирования маршрута

Корреляция признаков и PDR							
N	V_{av}	V_{max}	$Dist_{av}$	$Dist_{max}$	I_{all}	I_{max}	
-0,53	-0,094	-0,09	-0,041	-0,028	-0,493	-0,36	
L_{av}	L_{max}	Q_{av}	Q_{max}	$Delay_{av}$	$Delay_{max}$	R_{ch}	R_n
-0,382	-0,35	-0,267	-0,188	-0,424	-0,364	0,569	0,681
Корреляция признаков и E2ED							
N	V_{av}	V_{max}	$Dist_{av}$	$Dist_{max}$	I_{all}	I_{max}	
0,297	-0,006	-0,008	-0,004	-0,002	0,499	0,34	
L_{av}	L_{max}	Q_{av}	Q_{max}	$Delay_{av}$	$Delay_{max}$	R_{ch}	R_n
0,461	0,32	0,449	0,235	0,765	0,399	-0,318	-0,65
Корреляция признаков и F_S							
N	V_{av}	V_{max}	$Dist_{av}$	$Dist_{max}$	I_{all}	I_{max}	
-0,47	-0,026	-0,028	-0,014	-0,013	-0,46	-0,32	
L_{av}	L_{max}	Q_{av}	Q_{max}	$Delay_{av}$	$Delay_{max}$	R_{ch}	R_n
-0,43	-0,31	-0,44	-0,25	-0,51	-0,35	-0,47	-0,64

Значения PDR имеют корреляцию среднего уровня с признаками N , R_{ch} , R_n , а также несколько более слабую статистическую зависимость с I_{all} , I_{max} , L_{av} , L_{max} , D_{av} , D_{max} . Показатель E2ED сильно коррелирует с признаками D_{av} и R_n , и слабее с D_{max} , Q_{av} , L_{av} , I_{all} , I_{max} . F_S имеет выраженную статистическую взаимосвязь с I_{all} , L_{av} , Q_{av} , D_{av} , R_{ch} , R_n . Результаты проведенного корреляционного анализа подтвердили возможность построения качественных моделей машинного обучения, которые будут рассчитывать PDR, E2ED, F_S с высокой степенью достоверности.

Для проверки эффективности алгоритмов градиентного бустинга и нейронной сети набор данных был разделён на обучающую (Xtrain, PDRtrain, E2EDtrain) и тестовую (Xtest, PDRtest, E2EDtest) выборки в пропорции 4:1. Отсутствие в наборе категориальных признаков упрощает решение задачи. Для стабильной работы модели ОРНС произведём нормирование признаков по формуле (4.4):

$$X'_j = \frac{X_j - \mu_j}{\sigma_j}. \quad (4.4)$$

где μ_j – среднее значение X_j на обучающем наборе данных, σ_j – стандартное отклонение. Такой процедуре должны быть подвержены все признаки, поступающие на вход обобщённой регрессионной нейронной сети.

В ходе экспериментального исследования производилось обучение алгоритмов градиентного бустинга и ОРНС на одинаковой обучающей выборке, после чего эффективность их работы сравнивалась на тестовом наборе данных. Для исследования были использованы язык программирования python3, а также библиотеки catboost, содержащая алгоритм градиентного бустинга, и NeuPy, содержащая алгоритм ОРНС.

Для сравнения результатов работы алгоритмов использовались такие метрики обучения, как квадратный корень среднеквадратичной ошибки $RMSE$ и коэффициент детерминации R^2 [113]. $RMSE$ вычисляется по формуле 4.5:

$$RMSE(y, Y) = \sqrt{\frac{1}{l} \sum_{i=1}^l (y_i - Y_i)^2}. \quad (4.5)$$

где y – массив истинных значений, Y – массив оценок целевых значений. R^2 вычисляется по следующей формуле 4.6:

$$R^2(Y, y_i) = 1 - \frac{\sum_{i=1}^l (y_i - Y_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2}. \quad (4.6)$$

где \bar{y} – среднее истинное значение целевой переменной.

В ходе экспериментального исследования алгоритмы были обучены отдельно для вычисления PDR, E2ED и F_S . Результаты экспериментального исследования представлены в табл. 4.5. Для обучения алгоритма градиентного бустинга использовались стандартные настройки регрессора CatBoostRegressor (количество деревьев – 100, глубина дерева – 5). Для регрессора ОРНС были установлены значения σ , равные 0,7 для оценки метрики PDR, 1,3 для расчёта E2ED, 1,0 для оценки F_S . Кроме того, в таблице представлены $RMSE$ и R^2 для F_S' – значения показателя F_S , аналитически рассчитанное на основе спрогнозированных PDR и E2ED по формуле (4.3).

Таблица 4.5 – Результаты обучения алгоритмов
градиентного бустинга и ОРНС

Вычисляемый показатель	RMSE		R2	
	ГБ	ОРНС	ГБ	ОРНС
PDR	0,07	0,07	0,68	0,67
E2ED	11,57	15,1	0,95	0,92
F_S'	5,67	12,34	0,94	0,73
F_S	4,63	10,49	0,96	0,81

Результаты экспериментального исследования демонстрируют, что алгоритм градиентного бустинга лучше, чем ОРНС, вычисляет метрики PDR, E2ED, F_S . При этом оба алгоритма относительно хорошо произвели вычисление метрики E2ED и значительно хуже осуществили оценку PDR. Коэффициент детерминации больше 0,8 свидетельствует о высоком качестве работы алгоритмов, в тоже время значения R^2 0,67 и 0,68 для PDR можно считать приемлемыми.

При этом градиентный бустинг значительно лучше рассчитывает метрику F_0 , что показывают значительно меньшее значение $RMSE$ (в 2,27 раз меньше, чем для ОРНС) и высокие значения R^2 , которое близко к 1,0. Аналитически рассчитанное значение F_S' несколько уступает в качестве вычисленному напрямую значению, поэтому для оценки маршрута связи целесообразно использовать показатель F_S , полученный при помощи алгоритма градиентного бустинга.

Аналогичным образом вычисляется метрика F_B . Всего было подготовлено по 4 модели ГБ и ОРНС. Первый вариант представляет собой оценочное значение надёжности доставки данных по маршруту (4.7). Вторая функция совпадает с $F_S(X)$ (4.8). В третьей критерии PDR и E2ED имеют равный приоритет (4.9). В четвёртой приоритет отдаётся критерию E2ED (4.10).

$$F_{B1}(X) = PDR(X). \quad (4.7)$$

$$F_{B2}(X) = \frac{PDR(X)^{0,3}}{\left(\frac{E2ED(X)}{E2ED_{norm}}\right)^{0,7}}. \quad (4.8)$$

$$F_{B3}(X) = \frac{PDR(X)^{0,5}}{\left(\frac{E2ED(X)}{E2ED_{norm}}\right)^{0,5}}. \quad (4.9)$$

$$F_{B4}(X) = \frac{PDR(X)^{0,7}}{\left(\frac{E2ED(X)}{E2ED_{norm}} \right)^{0,3}}. \quad (4.10)$$

4.3. Экспериментальная проверка разработанных методов и алгоритмов взаимодействия ПОСМ при формировании и восстановлении квазипараллельных маршрутов

Целью текущего этапа экспериментального исследования является изучение того, насколько определяемые в результате применения предложенных алгоритмов взаимодействия ПОСМ маршруты передачи трафика являются эффективными по сравнению со стандартной технологией AOMDV. Для этого была проведена серия экспериментов, в которых выявлялись зависимости показателей PDR и E2EDG для формируемых путей до целевого абонента от количества узлов сети, используемых протоколов связи и функций оценки маршрутов $F_B(X)$.

Первый эксперимент аналогичен таковому из раздела 4.1 для топологии А, при этом для оценки маршрутов и их весов используются функции $F_S(X)$ и $F_B(X)$, основанные на методах градиентного бустинга и ОРНС. Результаты экспериментов для моделей на основе градиентного бустинга представлены в табл. 4.6, для моделей на основе ОРНС – в табл. 4.7. Взаимное сравнение показателей приведено на рис. 4.6.

Результаты эксперимента демонстрируют, что в результате работы алгоритмов градиентного бустинга и ОРНС оптимальным был определён тот же маршрут, что и в первом эксперименте. В 5 случаях удалось улучшить оптимизируемые параметры по сравнению со стандартным подходом. Компромиссными решениями в эксперименте являлись использование функций $F_{B2}(X)$ градиентного бустинга и $F_{B3}(X)$ ОРНС. Несмотря на то, что в результате эксперимента удалось найти функции, которые улучшают оба показателя работы сети, из-за отсутствия альтернатив в построении маршрутов улучшения оказались незначительными.

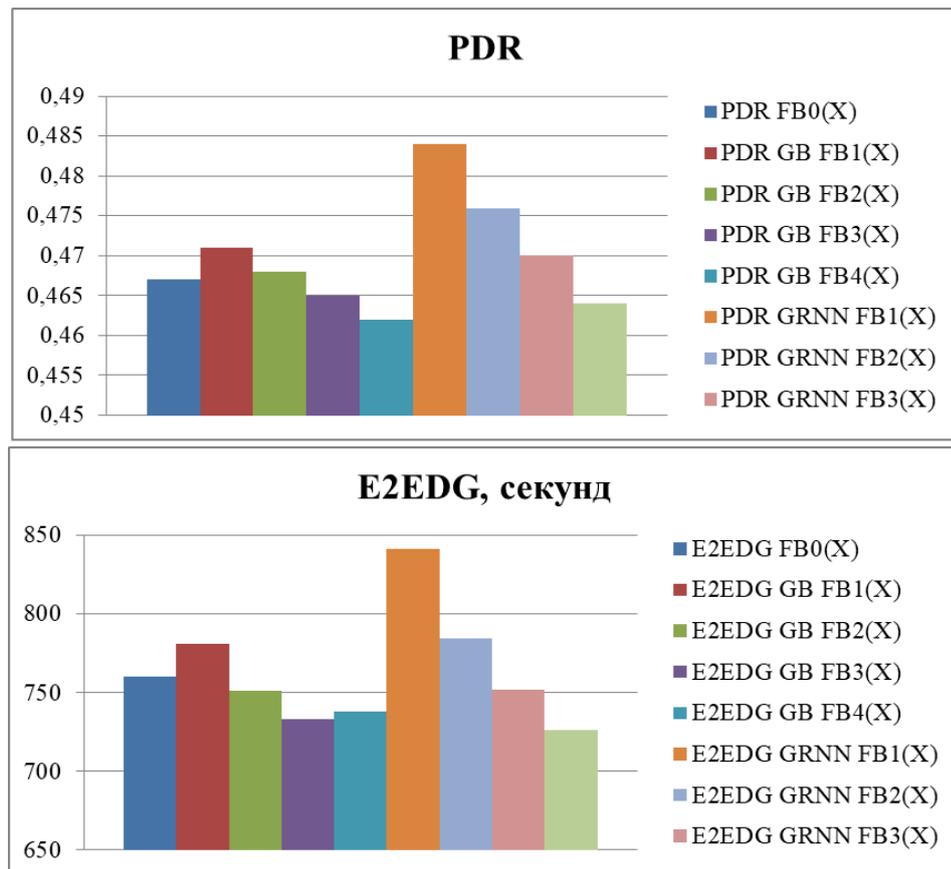


Рисунок 4.6 – Показатели функционирования сети для топологии А при использовании моделей машинного обучения

Таблица 4.6 – Результаты экспериментального исследования использования модели машинного обучения на основе градиентного бустинга для топологии А

Маршрут	E2ED (с)	E2EDG (с)	PDR	$F_S(X)$	$F_B(X)$	Доля графика
$F_B(X)=F_{B1}(X)$						
Осн.	26,37		0,612	30,12	0,539	0,41
Альт. 1	28,64		0,397	25,92	0,422	0,321
Альт. 2	19,77		0,351	21,27	0,352	0,268
В целом	24,74	781	0,471			1
$F_B(X)=F_{B2}(X)$						
Осн.	24,78		0,602	30,69	30,69	0,394
Альт. 1	28,53		0,386	25,34	25,34	0,326
Альт. 2	20,16		0,362	21,77	21,77	0,279
В целом	24,69	751	0,468			1
$F_B(X)=F_{B3}(X)$						
Осн.	26,32		0,623	30,79	4,54	0,384
Альт. 1	26,97		0,408	24,98	3,89	0,328
Альт. 2	18,81		0,371	21,16	3,4	0,288
В целом	24,65	733	0,465			1
$F_B(X)=F_{B4}(X)$						
Осн.	24,53		0,641	30,97	0,672	0,373
Альт. 1	28,01		0,393	25,84	0,596	0,33
Альт. 2	20,5		0,352	21,14	0,532	0,296
В целом	24,6	738	0,462			1

Таблица 4.7 – Результаты экспериментального исследования использования модели машинного обучения на основе ОРНС для топологии А

Маршрут	E2ED (с)	E2EDG (с)	PDR	$F_S(X)$	$F_B(X)$	Доля трафика
$F_B(X)=F_{B1}(X)$						
Осн.	25,79		0,591	35,75	0,672	0,466
Альт. 1	27,42		0,379	25,02	0,431	0,299
Альт. 2	19,15		0,351	21,21	0,338	0,235
В целом	24,89	841	0,484			1
$F_B(X)=F_{B2}(X)$						
Осн.	25,59		0,581	35,74	35,74	0,429
Альт. 1	28,46		0,382	25,72	25,72	0,309
Альт. 2	19,59		0,35	21,84	21,84	0,262
В целом	24,75	784	0,476			1
$F_B(X)=F_{B3}(X)$						
Осн.	25,16		0,631	36,12	5,064	0,405
Альт. 1	28,35		0,423	26,32	3,928	0,314
Альт. 2	19,42		0,358	22,32	3,515	0,281
В целом	24,65	752	0,47			1
$F_B(X)=F_{B4}(X)$						
Осн.	25,45		0,628	35,46	0,718	0,381
Альт. 1	27,02		0,382	25,22	0,599	0,318
Альт. 2	19,93		0,339	21,01	0,566	0,301
В целом	24,55	726	0,464			1

В четвёртом эксперименте производилось построение маршрутов и балансировка трафика при помощи моделей машинного обучения на сценарии с топологией С эксперимента из раздела 4.1. При этом учитывались только те случаи, когда между абонентами S и D формировались 3 альтернативных маршрута. Результаты исследования представлены на рис. 4.7 и 4.8.

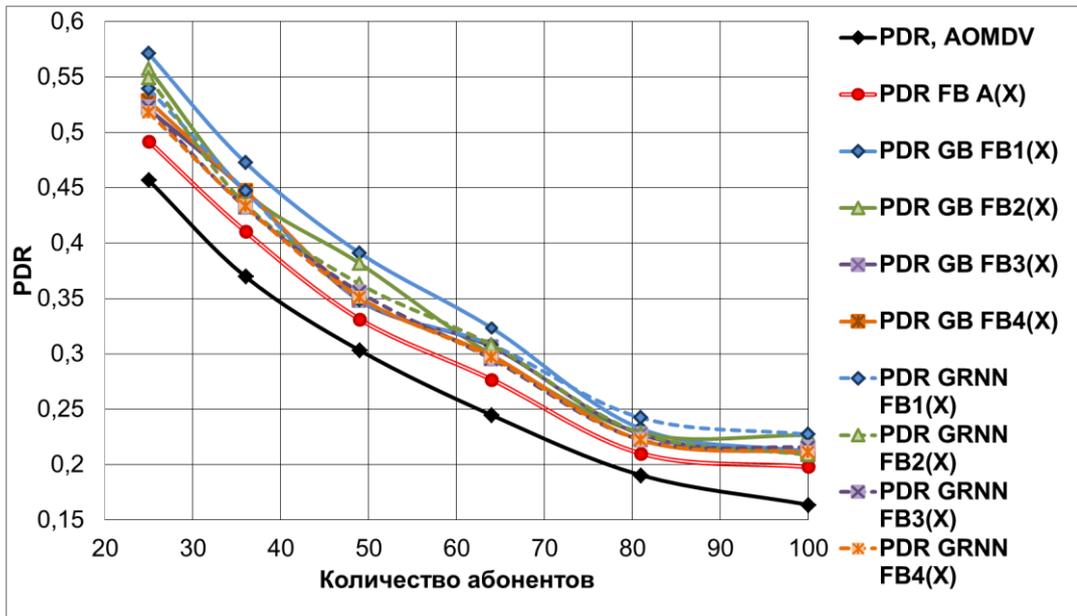


Рисунок 4.7 – Зависимость показателя PDR от количества абонентов в эксперименте на топологии С

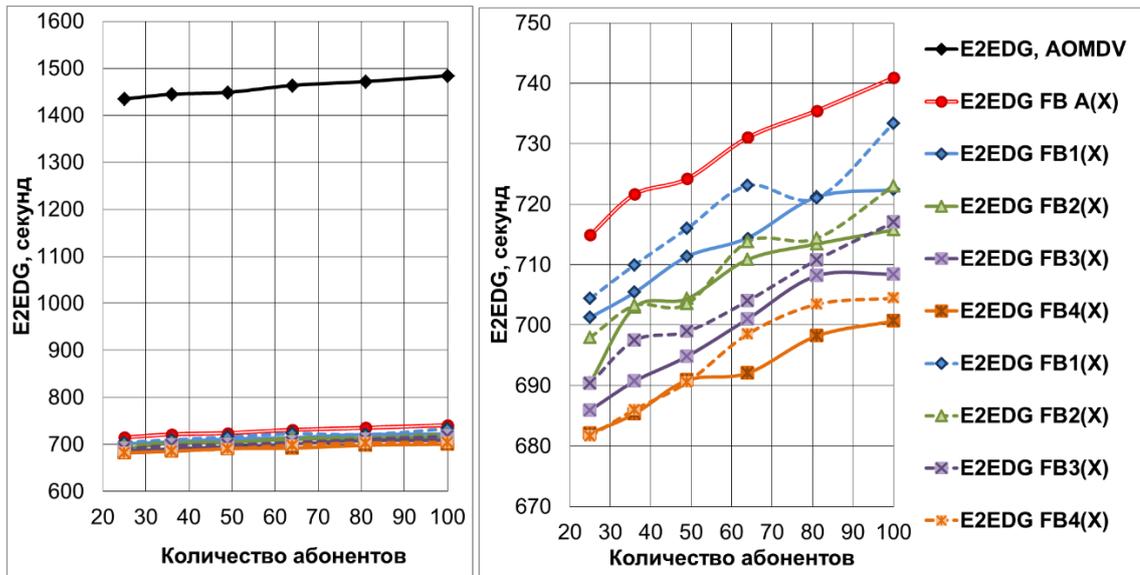


Рисунок 4.8 – Зависимость показателя E2EDG от количества абонентов в эксперименте на топологии С

В рассматриваемом эксперименте для всех функций $F_B(X)$ удалось улучшить как PDR, так E2EDG. При этом при использовании моделей, основанных на градиентном бустинге, качество работы сети несколько выше, чем при использовании моделей на основе ОРНС. Это соотносится с результатами проверки алгоритмов на тестовой выборке из раздела 4.2. При этом достигнуты существенные улучшения показателей по сравнению со стандартным протоколом AOMDV. Показатель PDR был улучшен в среднем в 1,08-1,39 раз, E2EDG в среднем в 2,0-2,12

раз. Проведённые эксперименты продемонстрировали, что предложенные в диссертации методы и алгоритмы позволяют значительно улучшить целевые параметры функционирования ДСПД со статичной топологией.

Следующая серия экспериментов была посвящена проверке алгоритмов на сценариях с высокой динамикой изменения топологии сети (рис. 4.9). Доля потерянных пакетов при передаче и наличие прямых связей между абонентами определялось расстоянием между ними. Движение сетевых узлов описывается моделью перемещений на основе случайных точек в заданной области. Абоненты отправитель и получатель задаются перед началом моделирования, им присваиваются координаты за пределами зоны пространственного перемещения мобильных узлов. Параметры моделирования представлены в табл. 4.8. Передача данных, кроме выделенных узлов, ведётся только между соседними абонентами.

Таблица 4.8 – Параметры моделирования в сценариях с моделью перемещения абонентов

Параметр	Значение
Максимальная длина маршрута	10, 12, 14, 16, 18, 20
Количество мобильных абонентов	8, 10, 11, 13, 14, 16
Количество абонентов с высокой мобильностью	4, 5, 5, 6, 7, 8
Скорость передачи данных	8 Кб/с.
Трафик между абонентами	2 Кб/с.
Максимальная скорость движения абонентов	6 м/с 15 м/с
Модель движения абонентов	Модель перемещений на основе случайных точек
Радиус радиопередачи	400 м.
Размер карты	1000 x 1000 м ² , 1200 x 1000 м ² , 1400 x 1000 м ² , 1600 x 1000 м ² , 1800 x 1000 м ² , 2000 x 1000 м ²
Ограничение по виртуальному времени моделирования	1200 с.

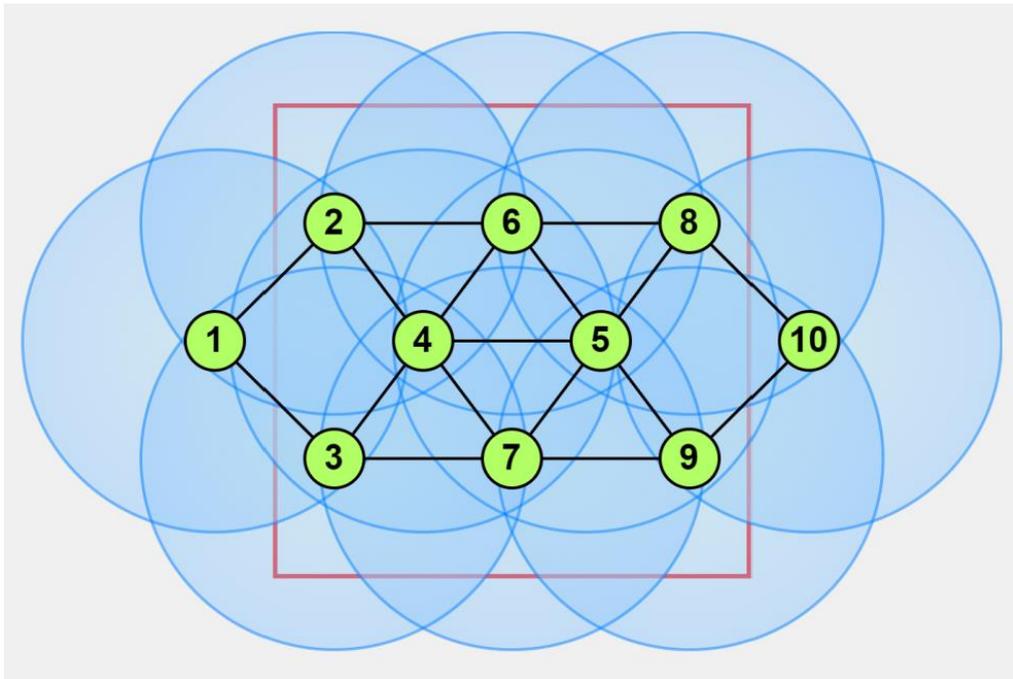


Рисунок 4.9 – Топология сети D с моделью перемещения абонентов

Первый сценарий с динамическим изменением топологии сети предполагал относительно медленное взаимное перемещение узлов— его максимальная скорость не превышала 6 м/с. Результаты моделирования представлена на рис. 4.10, рис. 4.11 и рис. 4.12. В этой серии экспериментов PDR был улучшен в среднем в 1,04–1,18 раз, E2EDG – в 1,94–2,25 раз, а общая эффективность передачи данных по сети возросла в 1,22-1,58 раз.

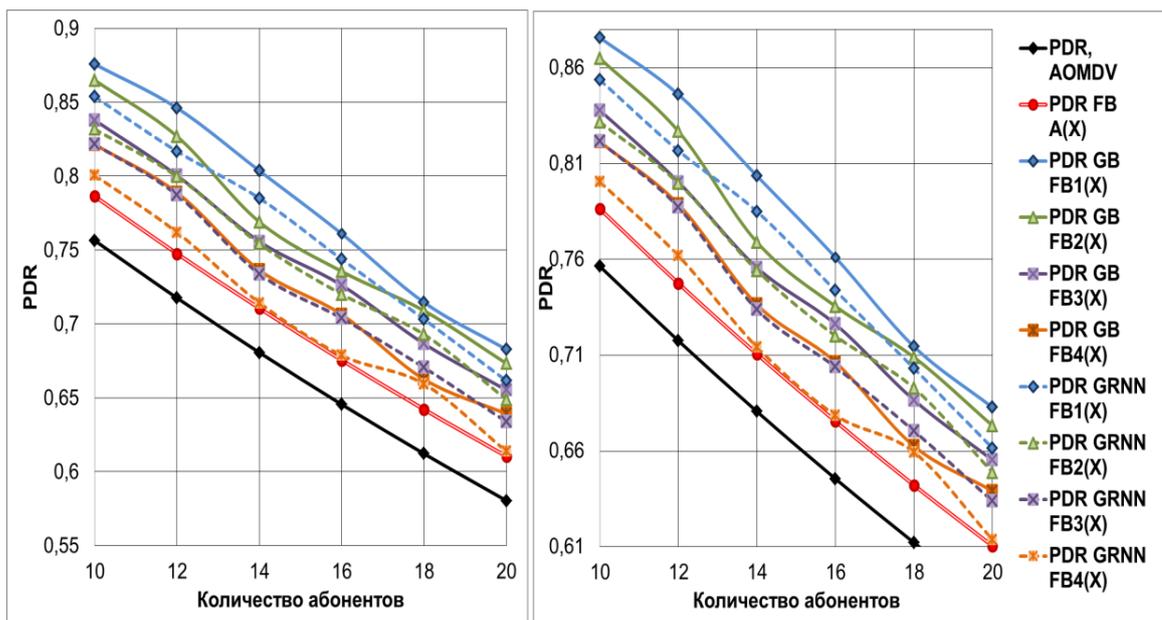


Рисунок 4.10 – Зависимость показателя PDR от количества абонентов в эксперименте на топологии D с низкой мобильностью абонентов

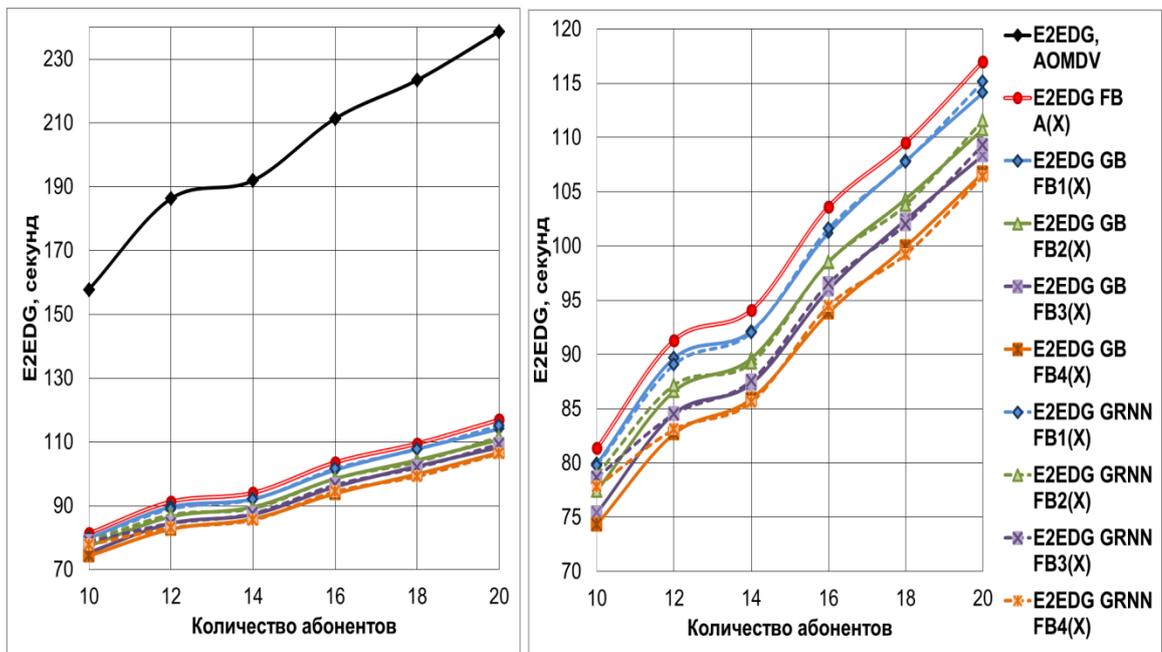


Рисунок 4.11 – Зависимость показателя E2EDG от количества абонентов в эксперименте на топологии D с низкой мобильностью абонентов

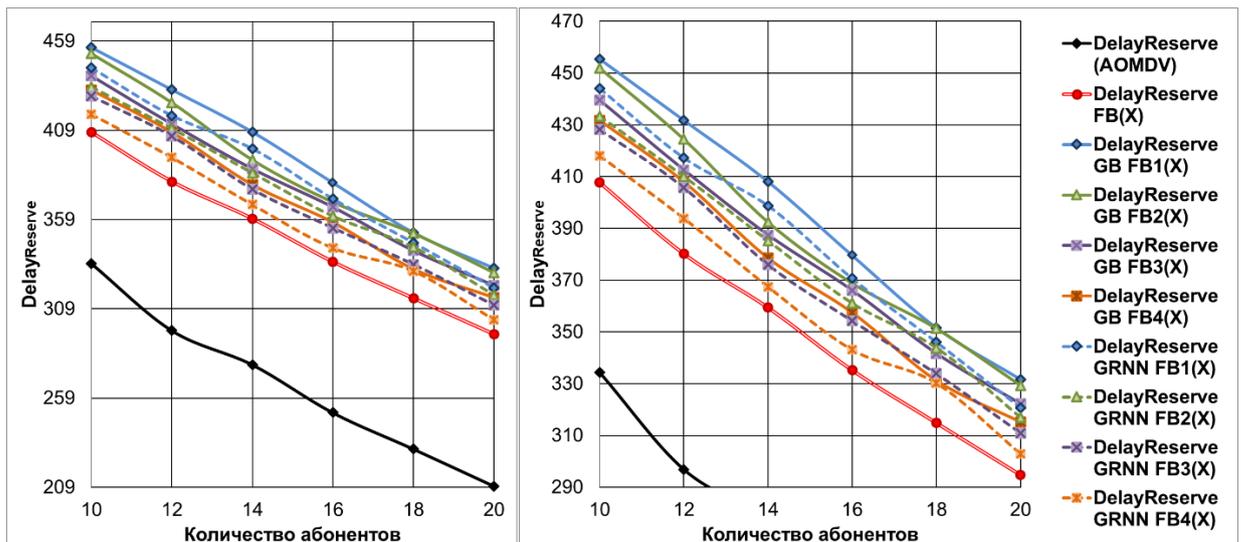


Рисунок 4.12 – Зависимость показателя $\text{Delay}_{\text{Reserve}}$ от количества абонентов в эксперименте на топологии D с низкой мобильностью абонентов

Во втором сценарии беспроводная сеть имеет высокую динамику изменения топологии вследствие увеличенной скорости перемещения абонентов – до 15 м/с. Результаты моделирования представлены на рис. 4.13, 4.14 и 4.15. В рассматриваемой серии экспериментов улучшение показателей качества обслуживания сети более выражено: PDR был улучшен в среднем в 1,12–1,36 раз, E2EDG – в 2,05–2,27 раз, а общая эффективность передачи данных по сети возросла в 1,4–1,92 раза.

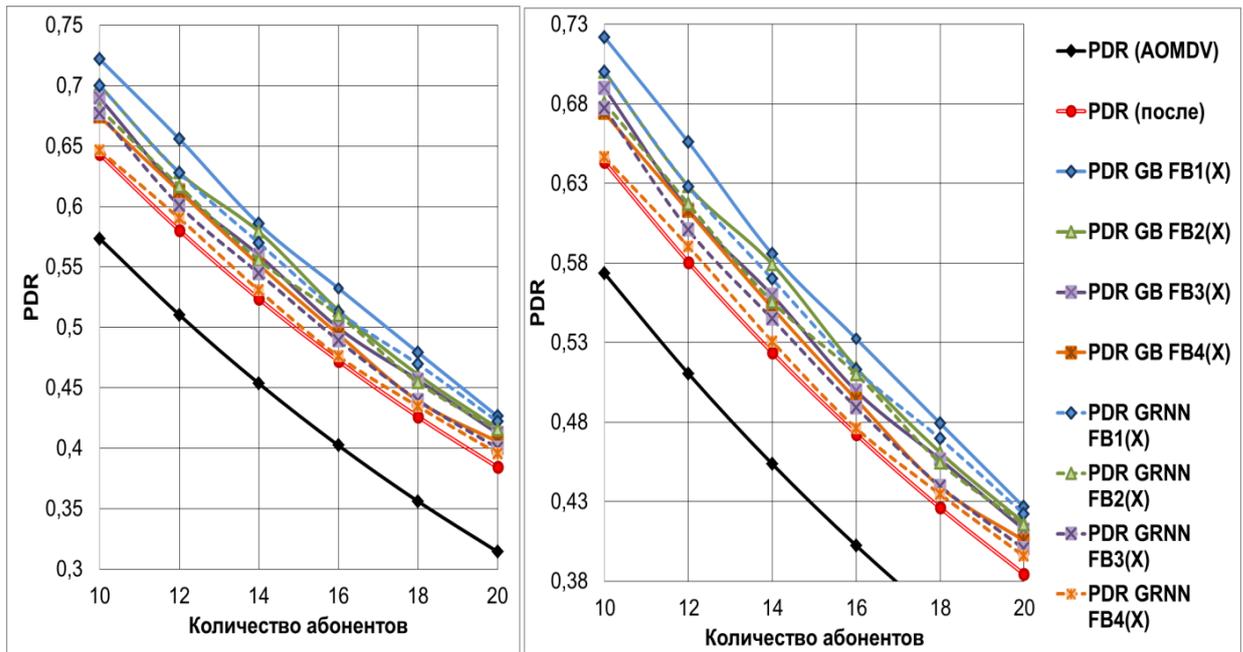


Рисунок 4.13 – Зависимость показателя PDR от количества абонентов в эксперименте на топологии D с высокой мобильностью абонентов

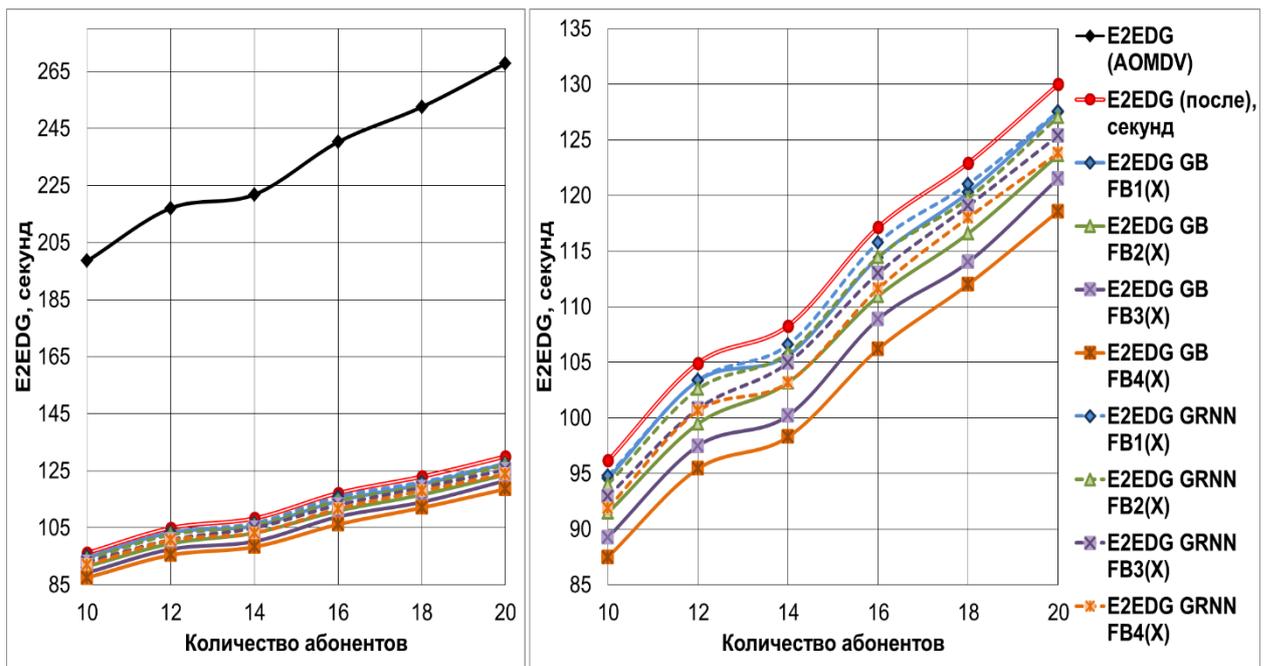


Рисунок 4.14 – Зависимость показателя E2EDG от количества абонентов в эксперименте на топологии D с высокой мобильностью абонентов

Таким образом, установлено, что предложенные алгоритмы взаимодействия сетевого ПО при построении, восстановлении маршрутов и балансировки трафика по ним эффективно работают как на сценариях с низкой динамикой изменения топологии, так и с высокой. Причем наиболее значимый эффект достигается, именно, для высокомобильных абонентов. В работе показано, что существенного

роста параметров, характеризующих функционирование сети в целом, можно добиться при использовании моделей на основе градиентного бустинга деревьев принятия решений для определения значений функций $F_S(X)$ и $F_B(X)$. При этом, во всех случаях одновременно удалось повысить как значение PDR, так и E2EDG. Было показано, что компромиссным решением, которое позволит в равной степени улучшить оба показателя, является использование функции $F_{B_2}(X)$ на основе градиентного бустинга. Следовательно, проведённая серия экспериментов подтверждает, что предложенные методы и алгоритмы взаимодействия сетевого ПО позволяют существенно повысить надёжность и снизить сквозную задержку передачи данных.

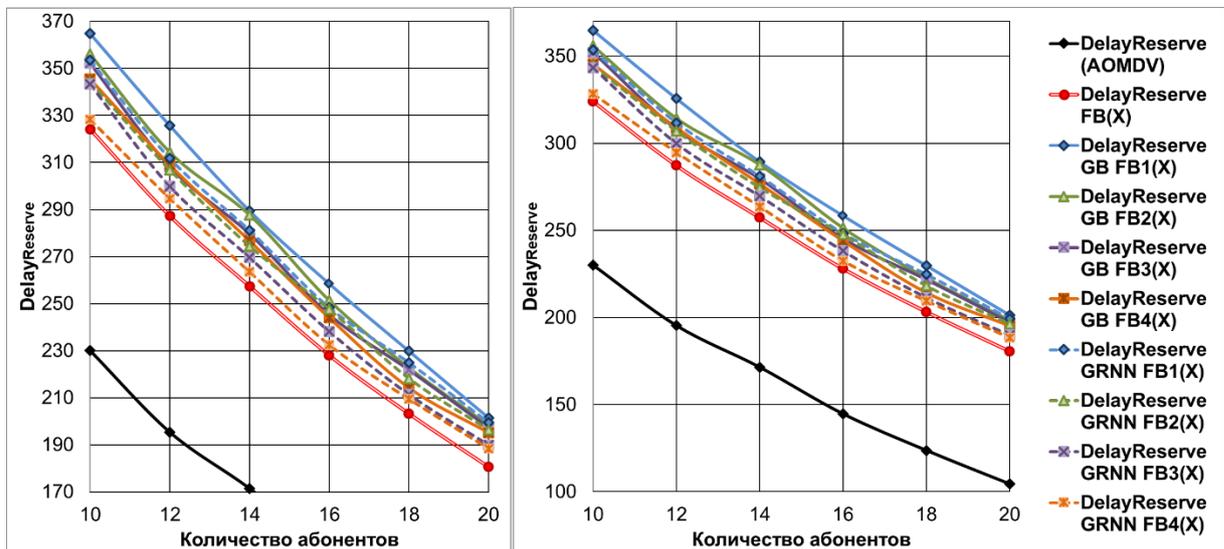


Рисунок 4.15 – Зависимость показателя $\text{Delay}_{\text{Reserve}}$ от количества абонентов в эксперименте на топологии D с высокой мобильностью абонентов

4.4. Экспериментальная проверка алгоритма взаимодействия ПОСМ при восстановлении маршрутов

В экспериментах, посвящённых исследованию алгоритма взаимодействия программного обеспечения абонентов при восстановлении маршрутов связи, осуществлялось определение временных затрат на процесс восстановления, а также его влияние на показатели надёжности, оперативности доставки данных и на общую эффективность передачи трафика по сети. Параметры моделирования аналогичны предыдущим экспериментам над сетью с высокой динамикой изменения топологии.

В первом эксперименте изучалась зависимость времени восстановления маршрута связи от номера узла T , между которым и последующим абонентом маршрута произошел разрыв канала связи. Узел T является инициатором восстановления маршрута. Время фиксировалось от момента отправки запроса на восстановление маршрута связи до момента готовности обновленного маршрута участвовать в процессе передачи трафика по сети. Результаты моделирования представлены на рис. 4.16.

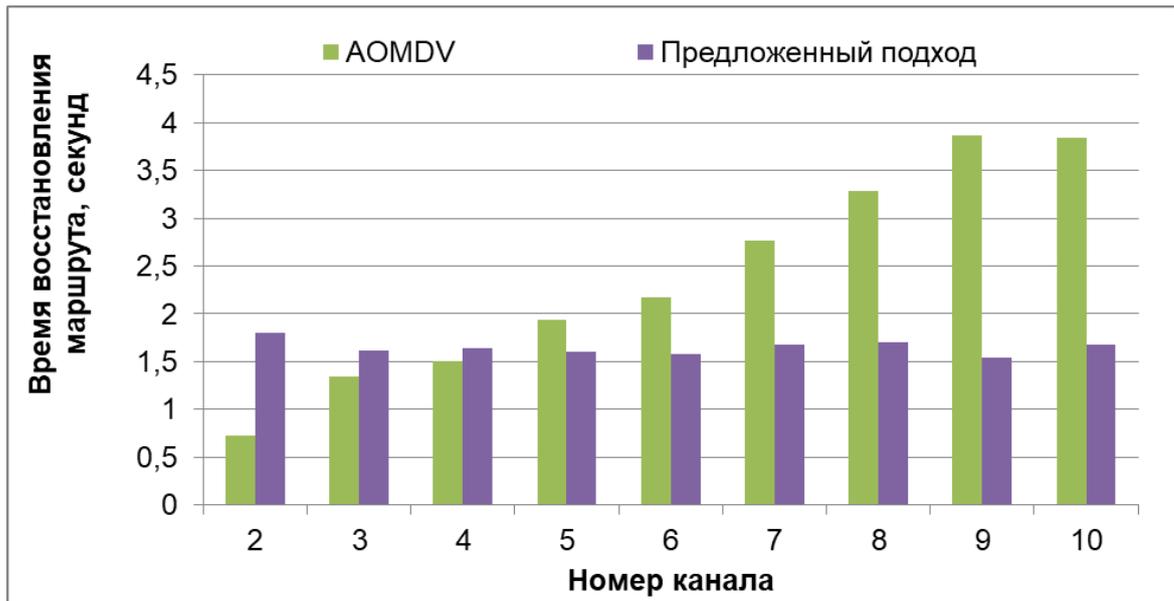


Рисунок 4.16 – Зависимость времени восстановления маршрута связи от номера абонента, который инициировал процесс восстановления

Результаты демонстрируют, что восстановление маршрута при использовании AOMDV более затратное по времени, чем в разработанном в диссертации протоколе, и оно увеличивается по мере того, насколько далеко от первого абонента маршрута произошел разрыв связи. В предлагаемом решении время восстановления близко к константе вне зависимости от месторасположения разрыва.

Первый сценарий с динамическим изменением топологии сети предполагал относительно низкую скорость её изменения. Результаты моделирования представлены на рис. 4.17, табл. 4.9, 4.10. Было получено улучшение значений как частных, так и интегрального критериев эффективности передачи данных по сети на основе авторского решения относительно AOMDV. Степень

улучшения показателей при применении разработанных в работе методов и алгоритмов представлена в табл. 4.9. Таким образом, применение методов взаимодействия ПОСМ при построении и восстановлении квазипараллельных маршрутов позволяет повысить в среднем интегральный показатель эффективности $\text{Delay}_{\text{Reserve}}$ в 1,23 раз, PDR в 1,17 раз, E2EDG в 1,17 раз. В сценарии, где исключалась возможность построения параллельным путей, но обеспечивалось построение квазипараллельных маршрутов – $\text{Delay}_{\text{Reserve}}$ улучшен в 1,51 раз, PDR в 1,16 раз, E2EDG – в 2,36 раз.

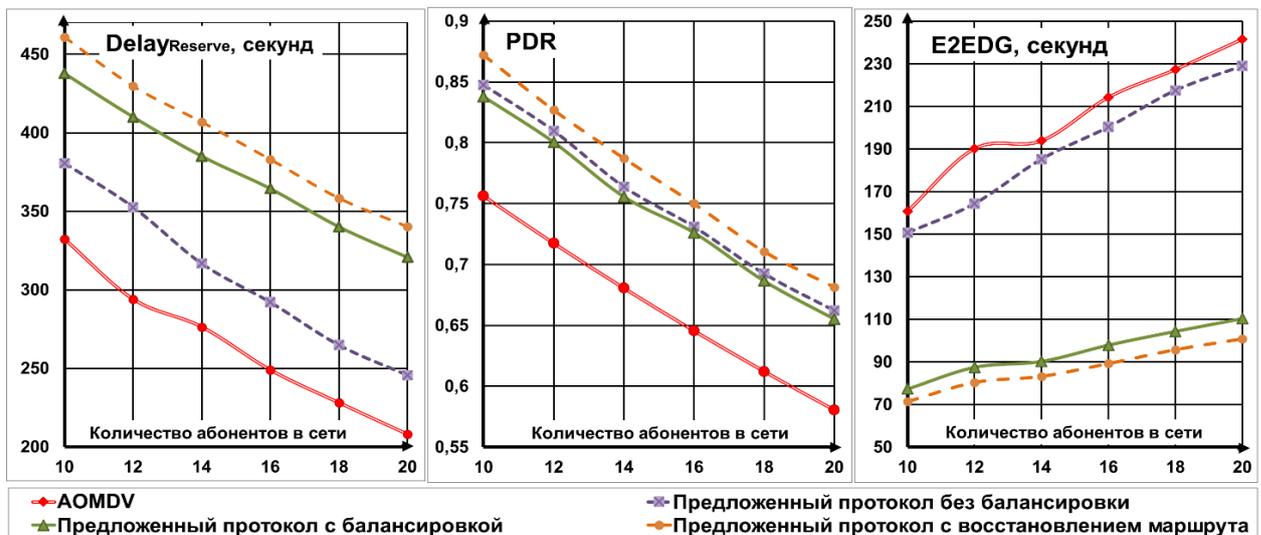


Рисунок 4.17 – Зависимость показателей $\text{Delay}_{\text{Reserve}}$, PDR, E2EDG от количества абонентов в эксперименте на топологии D с низкой мобильностью абонентов

Таблица 4.9 – Сравнение значений критериев эффективности передачи данных в ДСПД при использовании авторского решения с протоколом AOMDV в сценариях с низкой динамикой изменения топологии

Применяемый метод	$\text{Delay}_{\text{Reserve}}$	PDR	E2EDG
Взаимодействия ПОСМ при построении квазипараллельных маршрутов	1,17	1,13	1,07
Взаимодействия ПОСМ при восстановлении маршрутов	1,05	1,04	1,09
Одновременное применение двух методов	1,23	1,17	1,17
Метод балансировка потоков данных	1,23	0,99	2,01
Одновременное применение трёх методов	1,51	1,16	2,36

* $\text{Delay}_{\text{Reserve}}$ – запас времени на передачу потока данных, PDR (packet delivery ratio – коэффициент доставки пакетов), E2EDG (end-to-end delay group – сквозная задержка)

Таблица 4.10 – Результаты экспериментального исследования качества функционирования маршрутов связи с активированным алгоритмом восстановления маршрутов

Протокол	Количество абонентов					
	10	12	14	16	18	20
Delay_{Reserve} (секунд)						
АОМДV	332,16	293,97	276,34	248,90	228,05	207,96
Предложенный протокол без балансировки	380,68	352,68	316,73	292,11	264,82	245,60
Предложенный протокол с балансировкой	437,83	410,16	385,18	364,55	340,24	320,83
Предложенный протокол с восстановлением маршрутов	460,99	429,68	406,86	383,03	358,34	340,13
PDR						
АОМДV	0,76	0,72	0,68	0,65	0,61	0,58
Предложенный протокол без балансировки	0,85	0,81	0,76	0,73	0,69	0,66
Предложенный протокол с балансировкой	0,84	0,80	0,76	0,73	0,69	0,66
Предложенный протокол с восстановлением маршрутов	0,87	0,83	0,79	0,75	0,71	0,68
E2EDG (секунд)						
АОМДV	160,84	190,29	193,96	214,39	227,43	241,68
Предложенный протокол без балансировки	150,83	164,50	185,36	200,42	217,59	229,07
Предложенный протокол с балансировкой	77,4	87,55	90,30	97,96	104,41	110,35
Предложенный протокол с восстановлением маршрутов	70,55	80,69	82,59	89,52	96,64	102,01

Во следующем сценарии ДСПД имеет высокую динамику изменения топологии вследствие задания более высокой скорости взаимного перемещения абонентов друг относительно друга. Результаты имитационного моделирования представлены на рис. 4.18, в табл. 4.11 и 4.12. В данном исследовании было достигнуто более выраженное улучшение значений маршрутных метрик: эффективность была повышена в 1,39 раз, PDR в 1,32 раз, а E2EDG снижена в 1,16 раз, причем дополнительное использование метода балансировки трафика

позволило повысить эффективность в 1,8 раз, PDR в 1,31 раз, а E2EDG снизить – в 2,43 раз.

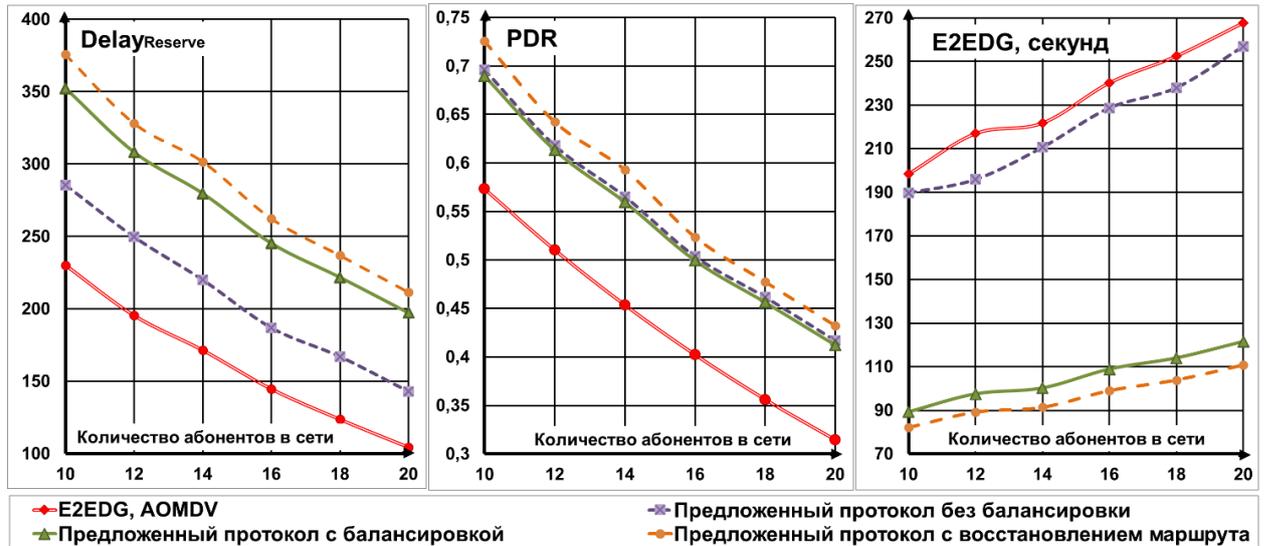


Рисунок 4.18 – Зависимость показателя E2EDG от количества абонентов в эксперименте на топологии *D* с высокой мобильностью абонентов и активированным алгоритмом восстановления маршрутов

Таблица 4.11 – Сравнение значений критериев эффективности передачи данных в ДСПД при использовании авторского решения с протоколом AOMDV в сценариях с высокой динамикой изменения топологии

Применяемый метод	DelayReserve	PDR	E2EDG
Взаимодействия ПОСМ при построении квазипараллельных маршрутов	1,30	1,26	1,06
Взаимодействия ПОСМ при восстановлении маршрутов	1,07	1,05	1,10
Одновременное применение двух методов	1,39	1,32	1,16
Метод балансировка потоков данных	1,29	0,99	2,09
Одновременное применение трёх методов	1,80	1,31	2,43

* DelayReserve – запас времени на передачу потока данных, PDR (packet delivery ratio – коэффициент доставки пакетов, E2EDG (end-to-end delay group – сквозная задержка)

Таблица 4.12 – Результаты экспериментального исследования качества маршрутов связи с активированным алгоритмом восстановления путей

Протокол	Количество абонентов					
	10	12	14	16	18	20
Delay_{Reserve} (секунд)						
АОМДV	230,18	195,45	171,52	144,72	123,72	104,44
Предложенный протокол без балансировки	285,64	249,77	220,11	187,01	167,10	143,00
Предложенный протокол с балансировкой	352,28	308,18	279,73	245,34	221,82	197,29
Предложенный протокол с восстановлением маршрутов	375,78	328,19	301,64	262,33	236,84	211,43
PDR						
АОМДV	0,57	0,51	0,45	0,40	0,36	0,31
Предложенный протокол без балансировки	0,70	0,62	0,57	0,50	0,46	0,42
Предложенный протокол с балансировкой	0,69	0,61	0,56	0,5	0,46	0,41
Предложенный протокол с восстановлением маршрутов	0,73	0,64	0,59	0,52	0,48	0,43
E2EDG (секунд)						
АОМДV	198,55	217,05	221,84	240,28	252,55	267,75
Предложенный протокол без балансировки	189,81	196,00	210,74	228,72	238,00	256,86
Предложенный протокол с балансировкой	89,27	97,5	100,2	108,88	114,00	121,54
Предложенный протокол с восстановлением маршрутов	81,32	89,43	91,33	99,34	103,82	111,50

4.5 Экспериментальная проверка влияния метода оценки маршрутов передачи данных на объём служебного трафика

Одним из преимуществ предложенного метода оценки маршрутов передачи данных является возможность фильтрации информационных запросов маршрутов ($F_S < F_S'$, формула 2.8). Это означает, что запросы, в которых начали формироваться маршруты с низкой оценкой, не будут ретранслированы далее в сеть. Кроме того, ПОСМ во время процедуры восстановления маршрута, используя рассматриваемый фильтр, производит проверку возможности внесения изменений в путь. Если проверка не пройдена, процесс

восстановления не инициируется, и как следствие, в сеть не отправляются запросы восстановления маршрутов.

В экспериментальной проверке, рассматриваемой в текущей главе, использовался сценарий на топологии D с высококомобильными абонентами. При этом в каждом случае производился запуск сценария как с активированной фильтрацией запросов, так и без неё. Результаты представлены в табл. 4.13.

Таблица 4.13 – Результаты экспериментальной проверки влияния метода оценки маршрутов передачи данных на объём служебного трафика

Значение	Количество абонентов					
	10	12	14	16	18	20
Доля служебного трафика без применения фильтра	0,089	0,099	0,104	0,113	0,122	0,127
Доля служебного трафика с применением фильтра	0,085	0,094	0,099	0,106	0,114	0,119
Степень снижения объёма служебного трафика, %	4,67	5,01	5,26	5,95	6,32	6,57
Среднее значение снижения объёма служебного трафика, %	5,63					

Эксперимент продемонстрировал, что применение фильтрации позволило сократить в среднем на 5,63% объём служебных сообщений при взаимодействии ПОСМ во время построения и восстановления маршрутов передачи данных. Это сокращает общую загрузку сети, что повышает пропускную способность маршрутов связи.

4.6. Выводы по главе

В ходе проведения экспериментального исследования были получены следующие результаты:

1. Проведена серия экспериментов, которая подтвердила корректность функционирования предложенных в диссертации методов и алгоритмов обеспечения взаимодействия сетевого программного обеспечения абонентов ДСПД. Экспериментально показано, что разработанные процедуры обнаружения соседних абонентов, построения маршрутов и передачи трафика по ним позволяют улучшить значения эффективности передачи данных по сети различных топологий как с низкокомобильными, так и с высококомобильными абонентами.

Установлено, что использование методов и алгоритмов позволяет улучшить надежность доставки данных (PDR) по сравнению со стандартными подходами маршрутизации в одноранговых сетях на заданных сценариях. Критерий PDR также был увеличен для каждого найденного маршрута в отдельности. В то же время удалось снизить показатель E2EDG в 3-4 раза по сравнению со стандартным подходом при доставке фреймов данных объемом 255 пакетов ввиду передачи трафика по параллельным маршрутам. Нагрузка на узлы была распределена более равномерно, что позволило избежать перегрузок отдельных маршрутов.

2. Проведено обучение моделей градиентного бустинга деревьев принятия решений и обобщённой регрессионной нейронной сети для оценки качества маршрутов связи в ДСПД. Для решения этой задачи была собрана обучающая выборка данных, которые содержат информацию о маршрутах, по которым передавались потоки трафика, их метрики, а также показатели эффективности передачи данных по маршрутам – коэффициент доставки данных и среднее время сквозной задержки. Была выявлена высокая пригодность полученных моделей для оценки качества маршрутов связи, при этом модель на основе градиентного бустинга продемонстрировала более точные результаты.

3. Обученные модели градиентного бустинга деревьев принятия решений и ОРНС были применены для оценки маршрутов на этапе их обнаружения и для вычисления их весов при балансировке трафика. Было выявлено, что использование указанных моделей позволяет повысить качество работы сети как в сетях низкой динамикой изменения топологии, так и с высокой. Более эффективно сети функционировали при использовании моделей на основе градиентного бустинга решающих деревьев. Компромиссным решением, которое позволяет в равной степени улучшить оба показателя, является использование функции $F_{B2}(X)$ для вычисления весов на основе градиентного бустинга. Проведённое исследование демонстрирует, что предложенные методы и алгоритмы взаимодействия ПОСМ при выполнении процедур маршрутизации и балансировки трафика позволяют существенно улучшить показатели качества маршрутов для передачи трафика в ДСПД.

4. Установлено, что использование разработанного подхода к восстановлению маршрутов связи позволяет дополнительно повысить PDR в среднем в 1,03 - 1,04 раза относительно базового протокола и в 1,15-1,17 раз относительно AOMDV, показатель E2EDG, соответственно, в 1,08-1,09 раз относительно базового протокола и в 2,25-2,39 раз относительно AOMDV, и таким образом, повысить эффективность функционирования сети по критерию запаса времени на передачу пакетов трафика в составе потока в 2,36 раз относительно AOMDV. В серии экспериментов для сетей с высококомобильными абонентами были получены еще более существенные улучшения параметров, характеризующих эффективность передачи данных по сети. Показатели PDR были увеличены в 1,05-1,06 раза относительно базового протокола и в 1,26-1,37 раз относительно протокола AOMDV, в то же время E2EDG удалось снизить в 1,1-1,2 раз относительно базового протокола и в 2,4-2,44 раз относительно AOMDV, а эффективность передачи данных по обобщённому критерию в 2,43 раза.

5. Таким образом, предложенные методы и алгоритмы взаимодействия программного обеспечения сетевой маршрутизации позволяют повысить эффективность процесса обмена данными между абонентами в ДСПД, снизить сквозную задержку и повысить коэффициент доставки сетевых пакетов. Применяемая балансировка трафика способствует разгрузке каналов связи, что позволяет увеличить пропускную способность сети. Имитационное моделирование подтвердило, что разработанные методы и алгоритмы позволяют сократить сквозную задержку передачи данных. При маршрутизации не требуется перераспределять нагрузку между оставшимися доступными маршрутами, поскольку производится корректировка текущего пути. В итоге можно сделать вывод, что предложенные в работе методы и алгоритмы взаимодействия сетевого ПО позволяют обеспечить повышение эффективности передачи данных в децентрализованных сетях связи за счёт разработки новых методов организации взаимодействия программного обеспечения сетевой маршрутизации узлов при построении многопутевых маршрутов передачи данных.

ЗАКЛЮЧЕНИЕ

В ходе выполнения диссертационной работы удалось повысить эффективность передачи данных в беспроводных децентрализованных сетях за счет создания новых методов и алгоритмов организации взаимодействия сетевого программного обеспечения отдельных узлов при решении задач маршрутизации, и таким образом достигнуть цели исследования. В диссертации были получены следующие результаты:

1. Предложены методы и алгоритмы взаимодействия программного обеспечения сетевой маршрутизации при построении и восстановлении маршрутов связи в ДСПД, которые отличаются от существующих решений совмещением реактивной и проактивной схем обмена сообщениями, совместным использованием адресных и широковещательных информационных ответов, что позволяет повысить значение коэффициента доставки сетевых пакетов и уменьшить сквозную задержку при их передаче по найденным путям, а также обеспечивает возможность построения и поддержки квазипараллельных маршрутов передачи трафика.

2. Предложен новый метод оценки маршрутов передачи данных, формируемых в процессе взаимодействия программного обеспечения сетевых узлов и отличающийся от стандартных решений возможностью динамической фильтрации и упорядочивания информационных сообщений на промежуточных узлах сети, что позволяет повысить эффективность функционирования ДСПД за счет снижения суммарного объема служебного трафика между программным обеспечением сетевой маршрутизации отдельных абонентов.

3. Предложен метод балансировки потоков данных в ДСПД, позволяющий снизить сквозную задержку доставки сообщений и отличающийся децентрализованным взаимодействием сетевых узлов при распределении пакетов трафика по квазипараллельным маршрутам с учётом их динамических метрик, собираемых при информационном обмене между программным обеспечением сетевой маршрутизации.

4. Реализовано программное средство, которое позволяет проводить моделирование процессов функционирования сети в различных сценариях работы и предоставляет возможность исследовать, отлаживать и выполнять оценку предложенных методов и алгоритмов взаимодействия сетевого программного обеспечения узлов ДСПД.

5. Предложена архитектура программного обеспечения сетевой маршрутизации отдельных узлов ДСПД, обеспечивающая интеграцию разработанных методов и алгоритмов взаимодействия ПО в рамках единого решения и их применение на практике.

6. Предложенные в диссертации методы и алгоритмы реализованы в виде протокола связи, на основе которого созданы программные модули для выполнения процедур сетевой маршрутизации, функционирующие в рамках разработанного программного средства.

7. Проведено имитационное моделирование, которое показало, что применение на практике предложенных в диссертации методов и алгоритмов обеспечения взаимодействия программного обеспечения сетевой маршрутизации позволяет повысить коэффициент доставки пакетов в среднем в 1,24 раза, снизить сквозную задержку передачи данных в среднем в 1,17 раз относительно стандартных технологий организации ДСПД, и таким образом, повысить эффективность функционирования сети по критерию запаса времени на передачу пакетов трафика в составе потока в 1,31 раза. В тоже время, применение метода балансировки трафика по квазипараллельным маршрутам позволяет повысить коэффициент доставки пакетов в 1,37 раз, снизить сквозную задержку передачи контрольного потока данных в 2,4 раза и, таким образом, дополнительно повысить эффективность передачи потока данных в 1,7. При этом применения предложенных методов организации взаимодействия программного обеспечения узлов в ДСПД позволило сократить объём служебного трафика в среднем на 5,63% во время работы сети.

8. Результаты диссертационной работы были внедрены в деятельность ООО «Дипвью Групп», ООО «АДВЕРТ-ПРО», а также в учебный процесс

кафедры «Вычислительная техника» ТулГУ при изучении дисциплин «Сетевое программирование», «Проектирование и тестирование программного обеспечения», «Сетевое программирование и протоколы передачи данных». Разработанные методы и алгоритмы взаимодействия ПОСМ, а также программный прототип предложенного в работе протокола маршрутизации были апробированы в научно-исследовательской работе «Разработка протокола верхнего уровня для организации связи в беспроводных децентрализованных сетях передачи данных», выполненной в рамках гранта правительства Тульской области в сфере науки и техники (договор ДС/286 от 25.10.2021).

Полученные в работе результаты могут быть использованы для повышения эффективности сетевого программного обеспечения, функционирующего в условиях низкой пропускной способности и нестабильности каналов связи, вызванных высокой динамикой изменения топологии сети и необходимостью передачи большого объема информации. Такие решения могут использоваться для обмена сетевым трафиком между распределенными абонентами и организации распределенных вычислений, передачи потоковых данных в ДСПД.

Направлением дальнейших исследований являются изучение возможности включения протоколов на основе предложенных в работе методов и алгоритмов взаимодействия ПОСМ в локальные и глобальные сети на основе ТСР/IP, а также проработка вариантов применения предложенных методов и алгоритмов в сетях с многоадресной маршрутизации трафика.

СПИСОК ЛИТЕРАТУРЫ

1. Брокмайер Д., Лебланк Д.-А., Маккарти Р. Маршрутизация в Linux = Linux Routing. – М.: «Вильямс», 2002. –С. 240.
2. Зарипова Э. Р., Альдемар А. П. Метод оценки времени установления соединения по радиоканалу случайного доступа //Discrete and Continuous Models and Applied Computational Science. 2017. Т. 25. №. 1. С. 9-18.
3. Бородин В. В., Петраков А. М., Шевцов В. А. Анализ алгоритмов маршрутизации в сети связи группировки беспилотных летательных аппаратов // Труды МАИ. 2016. №. 87. С. 16.
4. Catalan-Cid M. и др. DEMON: preemptive route recovery for AODV in multi-hop wireless networks based on performance degradation monitoring // EURASIP Journal on Wireless Communications and Networking. 2013. Т. 2013. С. 1-21.
5. Афонин С. И. и др. Методы решения задач распределения информационных потоков в сети передачи данных предприятия на основе резервирования ресурсов // Информационные системы и технологии. 2012. №. 1. С. 78-84.
6. Kirubasri G., Sankar S., Pandey D., Pandey B.K., Nassa V.K., Dadheech P. Software-Defined Networking-Based Ad hoc Networks Routing Protocols // EAI/Springer Innovations in Communication and Computing. Springer, Cham. 2022. Pp. 1-38. DOI: 10.1007/978-3-030-91149-2_5.
7. Bu C., Wang X., Cheng H., Huang M., Li K. Routing as a service (RaaS): An open framework for customizing routing services // Journal of Network and Computer Applications. 2019. Volume 125. Pp. 130-145. DOI: <https://doi.org/10.1016/j.jnca.2018.10.010>.
8. Tsao M., Yang K., Gopalakrishnan K., Pavone M. Private Location Sharing for Decentralized Routing Services // 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), Macau, China. 2022. Pp. 2479-2486. DOI: 10.1109/ITSC55140.2022.9922387.

9. Cisco Systems. Руководство Cisco по междоменной многоадресатной маршрутизации = Interdomain Multicast Solutions Guide. — М.: «Вильямс», 2004. — С. 320.
10. Abdelsalam A. и др. Performance of IPv6 Segment Routing in Linux Kernel // 2018 14th International Conference on Network and Service Management (CNSM), Rome, Italy. 2018. Pp. 414-419.
11. Bankovic L., Samardži D. One solution of 802.11 MESH communication protocol, implemented in LINUX kernel // 2013 21st Telecommunications Forum Telfor (TELFOR), Belgrade, Serbia. 2013. Pp. 46-49. DOI: 10.1109/TELFOR.2013.6716168.
12. Routing Software [Электронный ресурс] // Tech-FAQ. URL: <https://www.tech-faq.com/routing-software.html> (дата обращения 01.06.2024).
13. Давыдов А.В., Куклин В.В., Хайдаров К.А. Информационные сети и телекоммуникационные каналы [Электронный ресурс] // Bourabai Research. URL: <http://bourabai.ru/telecom/index.htm> (дата обращения: 01.06.2024).
14. Вишнеvский В. М. Теоретические основы проектирования компьютерных сетей / В. М. Вишнеvский. — М.: Техносфера, 2003. — 512 с.
15. Ластовченко, М. М. Системный анализ эффективности функционирования широкополосной транспортной платформы интеллектуальных сетей / М. М. Ластовченко, В. Е. Русецкий, В. Н. Ярошенко // Математичні машини і системи. — 2006. — № 1. — С. 28–39
16. Abdul Latiff N. M., Ibrahim I., Syed Yusof S. K., Abdul Malik N. N. N., Arsat R., Abdullah A. S. Load distributed routing protocol for wireless mesh networks // 2016 IEEE 3rd International Symposium on Telecommunication Technologies (ISTT), Kuala Lumpur, Malaysia. 2016. Pp. 41-46. DOI: 10.1109/ISTT.2016.7918082.
17. Jung S., Kim B. -S., Kim K. -I., Roh B., Ham J. -H. Implementation of AODV-UU on Linux 4.15 Kernel // 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW), Monterey, CA, USA. 2019. Pp. 160-161. DOI: 10.1109/MASSW.2019.00039.

18. Manjunath M., Manjaiah D. H. Spatial DSDV (S-DSDV) routing algorithm for mobile ad hoc network // 2014 International Conference on Contemporary Computing and Informatics (IC3I), Mysore, India. 2014. Pp. 625-629. DOI: 10.1109/IC3I.2014.7019587.
19. Shaha S. A. N., Pai V., Shenoy U. K. Comparison of wireless routing protocols over FTP traffic in mobile and non mobile nodes // 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kerala, India. 2017. Pp. 349-353. DOI: 10.1109/ICICICT1.2017.8342588.
20. Chroboczek J., Schinazi D. (January 2021). Протокол маршрутизации Babel. doi:10.17487/RFC8966. RFC 8966.
21. Murthy S., Garcia-Luna-Aceves J. J. An efficient routing protocol for wireless networks // Mob. Netw. Appl. 1996. Volume 1, Part 2. Pp. 183–197. DOI: doi.org/10.1007/BF01193336.
22. Ouacha A., Lakki N., El Abbadi J., Habbani A., El Koutbi M. OLSR protocol enhancement through mobility integration // 2013 10th IEEE INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC), Evry, France. 2013. Pp. 17-22. DOI: 10.1109/ICNSC.2013.6548704.
23. Aliyu U., Takruri H., Hope M., Halilu A. G. DS-OLSR – Disaster Scenario Optimized Link State Routing Protocol // 2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), Porto, Portugal. 2020. Pp. 1-6. DOI: 10.1109/CSNDSP49049.2020.9249639.
24. Oyakhire O., Gyoda K. Improved OLSR considering node density and residual energy of nodes in dense networks // 2020 35th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Nagoya, Japan. 2020. Pp. 161-165.
25. Manickavelu D., Vaidyanathan R.U. Particle swarm optimization (PSO)-based node and link lifetime prediction algorithm for route recovery in MANET // J Wireless Com Network. 2014. Volume 107. DOI: doi.org/10.1186/1687-1499-2014-107.

26. Deepak S., Anandakumar H. AODV Route Discovery and Route Maintenance in MANETs // 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India. 2019. Pp. 1187-1191. DOI: 10.1109/ICACCS.2019.8728456.
27. Chakeres I. D., Perkins C. E. DYnamic MANET On demand (DYMO) routing protocol // Internet-Draft Version 21, IETF, January 2011.
28. Zhong H., Zhou T. -t. Research and Implementation of AOMDV Multipath Routing Protocol // 2018 Chinese Automation Congress (CAC), Xi'an, China. 2018. Pp. 611-616. DOI: 10.1109/CAC.2018.8623785.
29. Alghamdi S.A. Load balancing ad hoc on-demand multipath distance vector (LBAOMDV) routing protocol // J Wireless Com Network 2015. 2015. Volume 242. 11p. DOI: <https://doi.org/10.1186/s13638-015-0453-8>.
30. Hamrioui S., Lorenz P. EQ-AODV: Energy and QoS supported AODV for better performance in WMSNs // Proc. 2016 IEEE International Conference on Communications (ICC). 2016. Pp. 1–6. DOI: 10.1109/ICC.2016.7510730.
31. Er-rouidi, M., Moudni, H., Mouncif, H., Merbouha, A. A balanced energy consumption in mobile ad-hoc network // Procedia Comput. Sci. 2019. 2019 Volume 151. Pp. 1182–1187. DOI:10.1016/j.procs.2019.04.169.
32. Malek A. -G., Li C., Layuan L., Bo W. New energy model: Prolonging the lifetime of Ad-hoc On-Demand Distance Vector Routing Protocols (AODV) // 2010 2nd International Conference on Future Computer and Communication, Wuhan, China. 2010. Volume 2. Pp. 426-429. DOI: 10.1109/ICFCC.2010.5497468.
33. Singh J., Singh P., Rani S. Enhanced Local Repair AODV (ELRAODV) // 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies, Bangalore, India. 2009. Pp. 787-791. DOI: 10.1109/ACT.2009.199.
34. Srinivasan P., Kamalakkannan P. Enhancing route maintenance in RSEA-AODV for mobile ad hoc networks // 2013 7th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India. 2013. Pp. 464-469. DOI: 10.1109/ISCO.2013.6481199.

35. Basha S.R., Sharma C., Sayeed F., Arularasan A. N., Pramila P. V., Shinde S.K., Pant B., Rajaram A., Yeshitla A. Implementation of Reliability Antecedent Forwarding Technique Using Straddling Path Recovery in Manet // *Wireless Communications and Mobile Computing*. 2022. Pp. 1-9. DOI: 10.1155/2022/6489185.

36. Jeon J., Lee K., Kim C. Fast route recovery scheme for Mobile Ad Hoc Networks // *The International Conference on Information Networking 2011 (ICOIN2011)*, Kuala Lumpur, Malaysia. 2011. Pp. 419-423. DOI: 10.1109/ICOIN.2011.5723121.

37. Istikmal I. Analysis and evaluation optimization dynamic source routing (DSR) protocol in Mobile Adhoc network based on ant algorithm // *2013 International Conference of Information and Communication Technology (ICoICT)*, Bandung, Indonesia. 2013. Pp. 400-404. DOI: 10.1109/ICoICT.2013.6574609.

38. Jin J., Ahn S., Oh H. A multipath routing protocol based on bloom filter for multi-hop wireless networks // *2015 International Conference on Information Networking (ICOIN)*, Cambodia. 2015. Pp. 521-522. DOI: 10.1109/ICOIN.2015.7057960.

39. Priyanshu, Maurya A. K. Impact of node density on the performance of GSR & TORA routing protocols // *2014 International Conference on Computer and Communication Technology (ICCCT)*. 2014. Pp. 251–255. DOI: 10.1109/ICCCT.2014.7001500.

40. Sanchez-Iborra R., Cano M.D. JOKER: A Novel Opportunistic Routing Protocol // *IEEE Journal on Selected Areas in Communications*. 2016. Vol. 34, Is. 5. Pp. 1690–1703. DOI:10.1109/JSAC.2016.2545439.

41. Разработка адаптивного алгоритма маршрутизации на основе роевого интеллекта пчелиной колонии для самоорганизующихся сетей беспилотных летательных аппаратов : автореферат дис. ... кандидата технических наук : 05.12.13 / Леонов Алексей Викторович; [Место защиты: Сибирский государственный университет телекоммуникаций и информатики]. - Омск, 2020. - 24 с.

42. Lee H., Jeon D., A Mobile Ad-hoc Network multi-path routing protocol based on biological attractor selection for disaster recovery communication // *ICT Express*. 2015. Volume 1. Issue 2. Pp. 86-89. DOI: <http://dx.doi.org/10.1016/j.icte.2015.10.001>.

43. Tabatabaei S., Nosrati Nahook H. A new routing protocol in MANET using cuckoo optimization algorithm // *J. Electr. Comput. Eng. Innovations*. 2021. Volume 9. Issue 1. Pp. 75-82. DOI: 10.22061/JECEI.2020.7511.397.

44. Subramaniam K., Tamilselvan L. Predictive energy efficient and reliable multicast routing in MANET // *Res. J. Appl. Sci. Eng. Technol.* 2015. Volume 9. Pp. 706–714. DOI:10.19026/rjaset.9.2615.

45. Papanna N., Reddy A.R.M., Seetha, M. EELAM: Energy efficient lifetime aware multicast route selection for mobile ad hoc networks // *Appl. Comput. Inform.* 2019. Volume 15. Pp. 120–128. DOI:10.1016/j.aci.2017.12.003.

46. Kacem I., Sait B., Mekhilef S., Sabeur N. A New Routing Approach for Mobile Ad Hoc Systems Based on Fuzzy Petri Nets and Ant System // *In IEEE Access*. 2018. Volume 6. Pp. 65705-65720. DOI: 10.1109/ACCESS.2018.2878145.

47. Ben Haj Frej, M., Mandalapa Bhoopathy, V., Ebenezer Amalorpavaraj S.R., Bhoopathy, A. Zone Routing Protocol (ZRP)—A Novel Routing Protocol for Vehicular Ad-hoc Networks // *In Proceedings of the ASEE-NE 2016, Kingston, RI, USA*. 2016. Pp. 28–30.

48. Young-Bae K, Nitin H. Vaidya. Location-Aided Routing (LAR) in Mobile Ad Hoc Networks // *J. Wireless Networks*, Kluwer Academic Publishers. 2000. Vol. 6, No. 4. Pp. 307–321. DOI:10.1145/288235.288252.

49. Пат. 110972209 КНР, МПК Н04W 28/10. Dynamic control method for wireless ad hoc network routing protocol propagation range // Чжан Ян (CN), Цянган Лю (CN), Гексяо Ланьмин (CN); заявитель и патентообладатель Shanghai Zhige Network Communication Technology (CN). – №: 201911261996.0; заявл. 10.12.19; опубл. 07.04.20.

50. Пат. 104202724 КНР, МПК Н04W 4/02. A kind of AANET joint route algorithms based on geographical location information // Тан Сяохэн и др. (CN);

заявитель и патентообладатель Chung Kyung University (CN). – №: 201410460463.6; заявл. 11.09.14; опубл. 10.11.17.

51. Пат. 104202724 КНР, МПК H04W 4/029. Self-adaptive method and system for intermittently connected unmanned aerial vehicle self-organizing network // Li Xianfeng и др. (CN); заявитель и патентообладатель Shenzhou University City North (CN). – №: 201711238204.9; заявл. 30.11.17; опубл. 18.08.20.

52. Пат. 105848247 КНР, МПК H04W 40/24. Vehicular Ad Hoc network self-adaption routing protocol method // Peng Shi (CN), Zhou Jieying (CN), Liu Yinglin (CN), Xu Yang Peng (CN); заявитель и патентообладатель Nakayama University (CN). – №: 201610329566.8; заявл. 17.05.16; опубл. 10.08.16.

53. Gerla M., Xu K. Multimedia streaming in large-scale sensor networks with mobile swarms // ACM SIGMOD Record. 2003. Volume 32. Issue 4. Pp. 72–74. DOI:10.1145/959060.959073

54. Weining Q., Xia Zhang X., Yu H. An improved CEDAR routing protocol // The Fourth International Conference on Computer and Information Technology 2004. CIT '04., Wuhan, China. 2004. Pp. 621-626. DOI: 10.1109/CIT.2004.1357264.

55. Пат. 2697833 Российская федерация, МПК G01S 13/02. Способ связи в ad-hoc сети // А. Шиодини (FR); заявитель и патентообладатель САФРАН ЭЛЕКТРОНИК Э ДЕФАНС САС (FR). – № 2017138925; заявл. 06.04.16; опубл. 13.05.19, Бюл. № 14.

56. Пат. 2681692 Российская федерация, МПК H04W 40/00. Способ ретрансляции, используемый в сети радиосвязи, и терминал для использования упомянутого способа // А. Шиодини (FR), А. Рове (FR), К. Геттье (FR), Ж. Аллар (FR), Ж. Йелло (FR), П.-А. Анри (FR); заявитель и патентообладатель САФРАН ЭЛЕКТРОНИК Э ДЕФАНС (FR). – № 2016135043; заявл. 20.01.15; опубл. 28.08.18, Бюл. № 25.

57. Романов, Сергей Владимирович. Метод иерархической маршрутизации мобильной самоорганизующейся сети доступа : автореферат дис. на соискание учёной степени кандидата технических наук : 05.12.13 / Романов Сергей

Владимирович; [Место защиты: Моск. гос. техн. ун-т радиотехники, электроники и автоматики]. - Киров, 2014. –19 с.

58. Махров, Станислав Станиславович. Использование нейронных механизмов искусственного интеллекта для кластеризации узлов и маршрутизации данных в беспроводных сенсорных сетях : автореферат дис. на соискание учёной степени кандидата технических наук : 05.12.13 / Махров Станислав Станиславович; [Место защиты: Моск. техн. ун-т связи и информатики]. - Москва, 2015. - 23 с.

59. Tang H., Xue F., Huang P. MP-MAODV: A MAODV-Based Multipath Routing Algorithm // 2008 IFIP International Conference on Network and Parallel Computing, Shanghai, China. 2008. Pp. 296-301. DOI: 10.1109/NPC.2008.23.

60. Ktari S., Labiod H., Frikha M. Load Balanced Multipath Routing in Mobile Ad hoc Network // 2006 10th IEEE Singapore International Conference on Communication Systems, Singapore. 2006. Pp. 1-5. DOI: 10.1109/ICCS.2006.301522.

61. Ramesh V., Subbaiah P., Chaitanya N.S., Supriya K.S. Performance comparison of congestion aware multi-path routing (with load balancing) and ordinary DSR // 2010 IEEE 4th International Conference on Internet Multimedia Services Architecture and Application, Bangalore, India. 2010. Pp. 1-5. DOI: 10.1109/IMSAA.2010.5729390.

62. Chen Y.-S., Chiang C.-Y. PER: a power-life extension routing protocol using a round robin scheme for mobile ad hoc networks // Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication (IEEE Cat. No.04EX710), Shanghai, China. 2004. Volume 1. Pp. 5-8. DOI: 10.1109/CASSET.2004.1322903.

63. Naseem M., Kumar C. Congestion-Aware Fibonacci Sequence Based Multipath Load Balancing Routing Protocol for MANETs // Wireless Personal Communications. 2015. Volume 84. Pp. 2955–2974. DOI: 10.1007/s11277-015-2775-6.

64. Bhattacharya, A., Sinha, K. An efficient protocol for load-balanced multipath routing in mobile ad hoc networks // *Ad Hoc Networks*. 2017. Volume 63. Pp. 104–114. DOI: 10.1016/j.adhoc.2017.05.008.

65. Pourbemany J., Mirjalily G., Abouei J., Fahim Raouf A.H. Load Balanced Ad-Hoc On-Demand Routing Based on Weighted Mean Queue Length Metric // *Electrical Engineering (ICEE), Iranian Conference on, Mashhad, Iran*. 2018. Pp. 470-475. DOI: 10.1109/ICEE.2018.8472705.

66. Pal A., Dutta P., Chakrabarti A., Singh J. A Multipath Load Balancing Routing Protocol in Mobile Ad Hoc Network Using Recurrent Neural Network // *Communications in Computer and Information Science*. 2019. Volume. 1030. Pp. 458–464, 10.1007/978-981-13-8578-0_36.

67. Пат. 2640349 Российская федерация, МПК H04W 40/02. Способ связи в ad-нос сети // Б.Я. Лихтциндер (RU), Е.Ю. Голубничая Екатерина Юрьевна (RU); заявитель и патентообладатель Федеральное государственное бюджетное образовательное учреждение высшего образования "Поволжский государственный университет телекоммуникаций и информатики" (ФГБОУ ВО ПГУТИ) (RU). – № 2016130621; заявл. 25.07.16; опубл. 28.12.17, Бюл. № 1.

68. Sahu P. K., Acharya B. and Panda N. QoS Based Performance Analysis of AODV and DSR Routing Protocols in MANET // *2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), Changsha, China*. 2018. Pp. 221-225. DOI: 10.1109/ICDSBA.2018.00046.

69. Анализ эффективности механизмов доставки потоковых данных с заданными требованиями к качеству обслуживания в самоорганизующихся беспроводных сетях : диссертация ... кандидата технических наук : 05.12.13 / Хоров Евгений Михайлович; [Место защиты: Ин-т проблем передачи информации РАН]. - Москва, 2012. - 142 с.

70. Mohapatra S., Tripathy T. MM-OLSR: Multi metric based optimized link state routing protocol for wireless ad-hoc network // *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPEs), Paralakhemundi, India*, 2016. Pp. 153-158. DOI: 10.1109/SCOPEs.2016.7955709.

71. Xiaoxia Q., Wang Q., Jiang F. Multi-path Routing Improved Protocol in AODV Based on Nodes Energy // International Journal of Future Generation Communication and Networking. 2015. Volume 8. No. 1. Pp. 207-214. DOI: 10.14257/ijfgcn.2015.8.1.21.

72. Smail O., Cousin B., Mekkakia Z., Mekki R. Energy aware and stable Multipath Routing protocol in clustered wireless ad hoc networks // 2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), Doha, Qatar. 2014. Pp. 222-229. DOI: 10.1109/AICCSA.2014.7073202.

73. Tilwari V., Bani-Bakr A., Qamar F., Hindia M. N., Jayakody D. N. K., Hassan R. Mobility and Queue Length Aware Routing Approach for Network Stability and Load Balancing in MANET // 2021 International Conference on Electrical Engineering and Informatics (ICEEI), Kuala Terengganu, Malaysia. 2021. Pp. 1-5. DOI: 10.1109/ICEEI52609.2021.9611119.

74. Romanik J., Kraśniewski A., Golan E. RESA-OLSR: RESources-aware OLSR-based routing mechanism for mobile ad-hoc networks // 2016 International Conference on Military Communications and Information Systems (ICMCIS), Brussels, Belgium. 2016. Pp. 1-6. DOI: 10.1109/ICMCIS.2016.7496549.

75. Yazdinejad A., Kavei S., Karizno S. R. Increasing the performance of reactive routing protocol using the load balancing and congestion control mechanism in MANET // Journal of Computer and Knowledge Engineering. 2019. Volume. 2. No. 1. DOI: 10.22067/CKE.V2I1.74711.

76. Taha A., Alsaqour R., Uddin M., Abdelhaq M., Saba T. Energy Efficient Multipath Routing Protocol for Mobile Ad-Hoc Network Using the Fitness Function // in IEEE Access. 2017. Volume 5. Pp. 10369-10381. DOI: 10.1109/ACCESS.2017.2707537.

77. Разработка алгоритмов передачи потоковых данных на прикладном уровне в сетях беспилотных летательных аппаратов : диссертация ... кандидата технических наук : 05.12.13 / Васильев Данил Сергеевич; [Место защиты: Поволж. гос. акад. телекоммуникаций и информатики]. - Ижевск, 2015. - 147 с.

78. Разработка адаптивного алгоритма маршрутизации для беспроводных многоузловых сетей передачи данных : диссертация ... кандидата технических наук : 05.12.13 / Дугаев Дмитрий Александрович; [Место защиты: Сиб. гос. ун-т телекоммуникаций и информатики]. - Новосибирск, 2017. - 137 с.

79. Youssef M., Ibrahim M., Abdelatif M., Chen L., Vasilakos A. V. Routing Metrics of Cognitive Radio Networks: A Survey // in IEEE Communications Surveys & Tutorials. 2014. Volume. 16. No. 1. Pp. 92-109. DOI: 10.1109/SURV.2013.082713.00184.

80. Selezinski N., Duan X., Scharff K., Kaller D., Harrer H. Routing Length Impact on Differential Via Crosstalk Cancellation // 2023 IEEE 27th Workshop on Signal and Power Integrity (SPI), Aveiro, Portugal. 2023. Pp. 1-4. DOI: 10.1109/SPI57109.2023.10145577.

81. Прозоров Д. Е. и др. Протоколы геомаршрутизации самоорганизующихся мобильных сетей // Т-Comm-Телекоммуникации и Транспорт. 2012. №5. С. 16-19.

82. Шорин А. О. Влияние скорости движения абонентов на интенсивность потерь соединений в сетях с сигналами OFDM // Экономика и качество систем связи. 2017. №1 (3). С. 69-78.

83. Гаврилин Е. А. Анализ объема трафика протоколов динамической маршрутизации RIP, OSPF и оценка загрузки каналов // REDS: Телекоммуникационные устройства и системы. 2017. Т. 7. №. 2. С. 192-193.

84. Воробьев Л. В., Обердерфер В. Н. Анализ алгоритмов организации и обслуживания очередей потоков пакетов в мультисервисной сети связи специального назначения // Проблемы технического обеспечения войск в современных условиях: сборник трудов конференции. 2019. С. 237-240.

85. Дедушкин О. В., Кокорев О. А. Анализ причин сетевой задержки в сетях с коммутацией пакетов // Символ науки. 2015. №. 9 (1). С. 62-65.

86. Лемешко А. В. и др. Модель безопасной маршрутизации на основе определения максимального количества непересекающихся путей для

минимизации вероятности компрометации конфиденциальных сообщений //Радиотехника. 2019. №. 197. С. 31-37.

87. Амбаткар С.В., Кулкарни П., Рентюк В. Neighbor Node Discovery для беспроводной самоорганизующейся одноранговой сети типа Ad-hoc // Беспроводные технологии. 2017. № 1 (46). С. 19-21.

88. Tzeng M.-F. Routing table partitioning for speedy packet lookups in scalable routers // in IEEE Transactions on Parallel and Distributed Systems. 2006. Volume 17. No. 5. Pp. 481-494. DOI: 10.1109/TPDS.2006.65.

89. Moy J. RFC 2328. OSPF Version 2. Chapter 13. The Flooding Procedure April 1998. Pp. 143-156.

90. Yuan Y.H., Chen H.M., Jia M. An Optimized Ad-hoc On-demand Multipath Distance Vector(AOMDV) Routing Protocol // Proceeding of Asia-Pacific Conference on Communications. 2005. Pp. 569-573. DOI: 10.1109/APCC.2005.1554125.

91. Piryonesi S.M., El-Diraby T. Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index // Journal of Infra-structure Systems. 2020. No. 26. Vol. 1. 25p. DOI:10.1061/(ASCE)IS.1943-555X.0000512.

92. Nader F., Hong G., Kazem M., Seyyed Ali S., Keramat H., Mohammad Reza E. «Nonlinear Poisson regression using neural networks: A simulation study», Neural Computing and Applications, No. 18, vol. 8, pp 939–943. DOI:10.1007/s00521-009-0277-8. S2CID 18980875.

93. Nader F., Hong G., Kazem M., Seyyed Ali S., Keramat H. Mohammad Reza E. Nonlinear Poisson regression using neural networks: A simulation study // Neural Computing and Applications. 2009. No. 18. Pp 939–943. DOI:10.1007/s00521-009-0277-8. S2CID 18980875.

94. Дьяконов А. Г. Методы решения задач классификации с категориальными признаками //Прикладная математика и информатика. Труды факультета Вычислительной математики и кибернетики МГУ имени МВ Ломоносова. 2014. №. 46. С. 103.

95. Макаренко С. И. Метод обеспечения устойчивости телекоммуникационной сети за счет использования ее топологической избыточности // Системы управления, связи и безопасности. 2018. № 3. С. 14–30. DOI: 10.24411/2410–9916–2018–10302.

96. Цветков К. Ю., Макаренко С. И., Михайлов Р. Л. Формирование резервных путей на основе алгоритма Дейкстры в целях повышения устойчивости информационно-телекоммуникационных сетей // Информационно-управляющие системы. 2014. № 2 (69). С. 71–78.

97. Перепелкин Д. А. Алгоритм парных перестановок маршрутов на базе протокола OSPF при динамическом отказе узлов и линий связи корпоративной сети // Вестник Рязанского государственного радиотехнического университета. 2014. № 1 (47). С. 84–91.

98. Одоевский С. М., Бусыгин А. В. Аналитическая модель обслуживания мультимедийного трафика с распределением Парето на основе аппроксимации результатов имитационного моделирования // Системы управления, связи и безопасности. 2020. № 1. С. 74–108. DOI: 10.24411/2410-9916-2020-10104.

99. Жолобов А. Н., Прозоров Д. Е., Романов С. В. Симуляторы беспроводных MANET-сетей // Инфокоммуникационные технологии. Самара. 2012. Т. 10. № 3. С. 28–33.

100. Павлов А.А., Датъев И.О. Проблемы использования средств тестирования многошаговых беспроводных сетей // Труды Кольского научного центра РАН. 2017. Т. 8. № 3–8. С. 116–123.

101. Flood J.E. Telecommunications Switching, Traffic and Networks, Chapter 4: Telecommunications Traffic, New York: Prentice-Hall, 1998. – 310 p.

102. Lobiyala D.K., Kattia C.P., Giria, A.K. Parameter Value Optimization of Ad-hoc On Demand Multipath Distance Vector Routing using Particle Swarm Optimization // Procedia Computer Science. 2015. Volume. 46. Pp. 151–158.

103. Шипицын С.Н. Алгоритм адаптивной настройки параметров протокола маршрутизации AODV. // Инженерный вестник Дона, Ростов-на-Дону. 2018.

9 с. [Электронный ресурс]. URL: <http://ivdon.ru/ru/magazine/archive/N2y2018/5029> (Дата обращения 01.06.2024).

104. Robinson S. Simulation: The Practice of Model Development and Use // New York, NY: Palgrave Macmillan, 2014. 336 p.

105. Шлеё М. Qt 5.10. Профессиональное программирование на C++. СПб.: БХВ-Петербург, 2018. – 1072 с.

106. Майоров А. Варианты реализации паттерна singleton // Системный администратор. – 2013. – №. 1-2. – С. 92-95.

107. Yegulalp S. «Yandex open sources CatBoost machine learning library», 2020: [Электронный ресурс]. URL: <https://www.infoworld.com/article/3209124/yandex-open-sources-catboost-machine-learning-library.html>. (Дата обращения: 01.06.2024).

108. NeuPy. Neural Networks in Python. – Официальный сайт: [Электронный ресурс]. URL: <http://neupy.com/pages/home.html>. (Дата обращения: 01.06.2024).

109. Безрук В. М., Варич В. В. Многокритериальный подход к маршрутизации в сетях связи // ВЕЖПТ. 2010. №9 (46). URL: <https://cyberleninka.ru/article/n/mnogokriterialnyu-podhod-k-marshrutizatsii-v-setyah-svyazi> (дата обращения: 01.06.2024).

110. Казбаев А. В., Китаева Е. Е., Герасимова Е. В. Анализ обеспечения информационного обмена по коротковолновым и ультракоротковолновым каналам связи // Вопросы радиоэлектроники. – 2018. – №. 12. – С. 14-19.

111. Постников В.М., Спиридонов С.Б. Методы выбора весовых коэффициентов локальных критериев // Наука и Образование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2015. № 06. С. 267–287. DOI: 10.7463/0615.0780334.

112. Гмурман В. Е. Теория вероятностей и математическая статистика: Учебное пособие для вузов. – 10-е издание, стереотипное. – Москва: Высшая школа, 2004. – 479 с.

113. Ивченко Г. И., Медведев Ю. И. Введение в математическую статистику. – Москва: Издательство ЛКИ, 2010. – 600 с.

114. Novikov A.S., Ivutin A.N., Voloshko A.G., Pestin M.S. Method for optimizing ad-hoc networks communication protocol parameter values // 2020 9th Mediterranean Conference on Embedded Computing (MECO), Proceedings. – 8-11 June 2020, Budva, Montenegro – IEEE, 2020. DOI: 10.1109/MECO49872.2020.9134154.

115. Pestin M.S., Novikov A.S., Ivutin A.N., Voloshko A.G. Multicriteria problem of optimization of data flows in heterogeneous information systems // 2021 10th Mediterranean Conference on Embedded Computing (MECO), – IEEE, 2021, с. 234-237. DOI:10.1109/MECO52532.2021.9460269.

116. Pestin M.S., Novikov A.S. Protocol for Multipath Routing of Traffic in Wireless Ad-Hoc Networks Based on the Status of Channels and Network Nodes // 2022 International Russian Automation Conference (RusAutoCon), Proceeding. 2022. С. 553-558. DOI: 10.1109/RusAutoCon54946.2022.9896315.

117. Хохлов Н.И., Ларин Д.В., Ларин А.В., Ивутин А.Н., Новиков А.С., Пестин М.С. Исследование протокола маршрутизации беспроводной децентрализованной самоорганизующейся сети // Известия Тульского государственного университета. Технические науки. Вупуск 6. Изд-во: ТулГУ, Тула, 2019, с. 131-144.

118. Новиков А. С., Пестин М. С. Программное обеспечение для настройки протоколов верхнего уровня ad-hoc сетей // Прикладная информатика. 2020. Т. 15. № 3. С. 60–74. DOI: 10.37791/2687-0649-2020-15-3-60-74.

119. Хохлов Н.И., Ивутин А.Н., Новиков А.С., Подчуфаров Ю.Б., Ларин Д.В., Пестин М.С. Протокол организации децентрализованной беспроводной сети связи для обеспечения взаимодействия группы интеллектуальных мобильных роботов // Изв. вузов. Приборостроение. 2020. Т. 63, № 12. С. 1081—1093. DOI: 10.17586/0021-3454-2020-63-12-1081-1093.

120. Новиков А.С., Пестин М.С. Распределённая маршрутизация трафика в беспроводных децентрализованных самоорганизующихся сетях связи // Известия Тульского государственного университета. Технические науки. 2021. № 5. С. 149-154.

121. Пестин М.С., Новиков А.С. Алгоритм многопутевой маршрутизации трафика в MANET сетях на основе анализа пропускной способности каналов связи // Вестник Рязанского государственного радиотехнического университета. 2022. № 80. С. 84-93. DOI: 10.21667/1995-4565-2022-80-84-93.

122. Пестин М. С., Новиков А. С. Имитационная модель беспроводной ad-hoc сети для исследования алгоритмов маршрутизации трафика // Прикладная информатика. 2022. Т. 17. № 4. С. 75–86. DOI: 10.37791/2687-0649-2022-17-4-75-86.

123. Пестин М.С. Методы машинного обучения в задачах построения маршрутов связи в беспроводных ad-hoc- сетях передачи данных // Известия Тульского государственного университета. Технические науки. 2022. № 8. С. 121-127. DOI: 10.24412/2071-6168-2022-8-121-128.

124. Пестин М.С., Новиков А.С. Взаимодействие абонентов MANET при установлении и поддержании соединения абонентов с сетью // Известия Тульского государственного университета. Технические науки. Принято к публикации.

125. Ивутин А.Н., Новиков А.С., Пестин М.С., Волошко А.Г. Децентрализованные протокол организации связи в подразделениях аварийно-спасательных служб // Информатика и автоматизация (К1), Санкт-Петербург. 2024. Том. 23. №3. С. 727-765. DOI: 10.15622/ia.23.3.4.

126. Новиков А. С., Ивутин А. Н., Пестин М. С. Алгоритм маршрутизации для обеспечения быстрого восстановления связи при обрывах маршрутов в сетях MANET // Системы управления, связи и безопасности (К1). 2024. № 2. С. 14-42. DOI: 10.24412/2410-9916-2024-2-014-042.

127. Пестин М.С. Метод оптимизации параметров протоколов связи в ad-hoc сетях. // Интеллектуальные и информационные системы. Интеллект – 2019. Труды Всероссийской научно-технической конференции. Тула, 19-20 ноября 2019. С. 311 – 317.

128. Пестин М.С. Подход к скаляризации векторных критериев качества функционирования беспроводных сетей в задаче поиска оптимальных

параметров протоколов связи методом fish school search // В сборнике: Мой шаг в науку. Материалы III Всероссийской научно-практической конференции. Отв. редактор Э.Р. Сайдимова. 2020. С. 156-159.

129. Пестин М.С. Сравнение методов свёртки вектора критериев производительности работы ad-hoc сети в задаче оптимизации параметров протокола связи // В сборнике: ПРОМЫШЛЕННАЯ РЕВОЛЮЦИЯ 4.0: ВЗГЛЯД МОЛОДЕЖИ. Тезисы докладов II Межрегиональной научной конференции. Тула, 2020. С. 39-40. URL: <https://www.elibrary.ru/item.asp?id=44488764>.

130. Пестин М.С. Метод передачи данных в беспроводных сетях связи в децентрализованных системах обработки информации // Оптико-электронные приборы и устройства в системах распознавания образов и обработки изображений (Распознавание – 2021), – Сборник материалов XVI Международной научно-технической конференции, 14-17 сентября 2021 г. С. 268-270.

131. Пестин М.С. Оптимизация параметров протоколов связи в ad-hoc сетях при помощи приведённого времени приёма-передачи. // Интеллектуальные и информационные системы. Интеллект – 2021. Труды Всероссийской научно-технической конференции. Тула, 16-17 ноября 2021. С. 161-166.

132. Пестин М.С., Новиков А.С., Ивутин А.Н.. Подход к разработке протоколов децентрализованной связи для повышения степени автономности группы интеллектуальных мобильных роботов при решении кооперативных задач. // Интеллектуальные и информационные системы. Интеллект – 2021. Труды Всероссийской научно-технической конференции. Тула, 16-17 ноября 2021. С. 237-239.

133. Пестин М.С. Взаимодействие программных агентов сетевых узлов для маршрутизации потоков данных в сетях MANET // Сборник трудов XVII международной научно-технической конференция «Оптико-электронные приборы и устройства в системах, распознавания образов и Обработки изображений» (Распознавание — 2023), Курск, Россия, 12-15 сентября 2023, с. 183-184.

134. Пестин М.С., Новиков А.С. Интеллектуальные агенты маршрутизации трафика беспроводных ad-hoc сетей // Труды конгресса

«Интеллектуальные системы и информационные технологии - 2023» (IS&IT'23), Таганрог, Россия, 1-7 сентября 2023, с. 354-351.

135. Новиков А.С., Пестин М.С. Программа для определения оптимальных значений параметров протоколов верхнего уровня в ad-hoc сетях // Свидетельство о регистрации программы для ЭВМ №2020663199 от 23.10.2020. Заявка № 2020662098 от 12.10.2020.

136. Новиков А.С., Пестин М.С. Программа для исследования характеристик протоколов маршрутизации в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ №2022612832 от 28.02.2022. Заявка № 2022611714 от 09.02.2022.

137. Пестин М. С., Новиков А. С. Программное обеспечение для моделирования взаимного пространственного перемещения отдельных узлов в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ №2022663501 от 15.07.2022. Заявка №2022662776 от 06.07.2022.

138. Пестин М. С., Новиков А. С. Программное обеспечение для маршрутизации трафика в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ 2022663502 от 15.07.2022. Заявка № 2022662775 от 06.07.2022.

139. Разработка протокола верхнего уровня для организации связи в беспроводных децентрализованных сетях передачи данных: отчёт о НИР (заключительный) / ФГБОУ ВО «Тульский государственный университет»; руководитель Пестин М.С.; Исполнители: Пестин М.С., Новиков А.С. – 2022. – 163 л. – № 121112200057-2.

ПРИЛОЖЕНИЯ

Приложение А. Свидетельства о государственной регистрации программ для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО
о государственной регистрации программы для ЭВМ
№ 2020663199

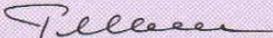
«Программа для определения оптимальных значений параметров протоколов верхнего уровня в ad-hoc сетях»

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

Авторы: *Новиков Александр Сергеевич (RU),
Пестин Максим Сергеевич (RU)*

Заявка № **2020662098**
Дата поступления **12 октября 2020 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ **23 октября 2020 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

 **Г.П. Ивлиев**



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022612832

**«Программа для исследования характеристик
протоколов маршрутизации в беспроводных
децентрализованных сетях связи»**

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

Авторы: *Новиков Александр Сергеевич (RU), Пестин
Максим Сергеевич (RU)*

Заявка № 2022611714

Дата поступления 09 февраля 2022 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 28 февраля 2022 г.



*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 68b80077e14e40f0a94eabd24145d5c7
Владелец **Зубов Юрий Сергеевич**
Действителен с 2.03.2022 по 26.05.2023

Ю.С. Зубов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022663501

**«Программное обеспечение для моделирования
взаимного пространственного перемещения отдельных
узлов в беспроводных децентрализованных сетях связи»**

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

Авторы: *Пестин Максим Сергеевич (RU), Новиков
Александр Сергеевич (RU)*

Заявка № 2022662776

Дата поступления 06 июля 2022 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 15 июля 2022 г.



*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 68b80077e14e40f0a94e6bd24145d5c7
Владелец **Зубов Юрий Сергеевич**
Действителен с 20.05.2022 по 26.05.2023

Ю.С. Зубов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022663502

**«Программное обеспечение для маршрутизации
трафика в беспроводных децентрализованных сетях
связи»**

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

Авторы: *Пестин Максим Сергеевич (RU), Новиков
Александр Сергеевич (RU)*



Заявка № 2022662775

Дата поступления 06 июля 2022 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 15 июля 2022 г.

*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 6Bb80077e14e40f0a94e8bd24145d5c7
Владелец **Зубов Юрий Сергеевич**
Действителен с 20.05.2022 по 26.05.2023

Ю.С. Зубов

Приложение Б. Акты об использовании результатов диссертационной работы

ООО «Дипвью Групп»

Тел.: +7 (920) 759 28-04, e-mail: admin@deepviewer.ru
ОГРН 1217100013239 / ИНН 7100013359 / КПП 710001001

АКТ

о внедрении результатов диссертационной работы

Настоящий акт о внедрении свидетельствует о том, что программные продукты, разработанные Пестиным Максимом Сергеевичем в рамках диссертационной работы, имеющие следующие свидетельства о регистрации программы для ЭВМ:

1. Программа для исследования характеристик протоколов маршрутизации в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ №2022612832 от 28.02.2022. Заявка № 2022611714 от 09.02.2022.
2. Программное обеспечение для моделирования взаимного пространственного перемещения отдельных узлов в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ №2022663501 от 15.07.2022. Заявка №2022662776 от 06.07.2022.
3. Программное обеспечение для маршрутизации трафика в беспроводных децентрализованных сетях связи // Свидетельство о регистрации программы для ЭВМ 2022663502 от 15.07.2022. Заявка № 2022662775 от 06.07.2022.
4. Программа для определения оптимальных значений параметров протоколов верхнего уровня в ad-hoc сетях // Свидетельство о регистрации программы для ЭВМ №2020663199 от 23.10.2020. Заявка № 2020662098 от 12.10.2020.

внедрены в эксплуатацию в ООО «Дипвью Групп».

В ходе эксплуатации программных продуктов было подтверждено, что они обладают всеми заявленными возможностями.

Генеральный директор



П.А. Савенков



ООО "АДВЕРТ-ПРО"

Юридический адрес: 300024, Тульская обл, Тула г., Жуковского ул., дом №
38Б, помещение 3

Телефон +74959692780, e-mail info@advertpro.ru

ИНН 7107106789/КПП 710701001

АКТ

внедрения результатов диссертационной работы

Акт составлен о том, что результаты диссертационной работы Пестина Максима Сергеевича:

– протокол взаимодействия сетевых служб при решении задач маршрутизации;

– программное обеспечение для маршрутизации трафика в беспроводных децентрализованных сетях связи (свидетельство о регистрации программы для ЭВМ №2022663502 от 15.07.2022. Заявка №2022662775 от 06.07.2022);

– программа для исследования характеристик протоколов маршрутизации в беспроводных децентрализованных сетях связи (свидетельство о регистрации программы для ЭВМ №2022612832 от 28.02.2022. Заявка №2022611714 от 09.02.2022)

внедрены в эксплуатацию «ООО АДВЕРТ-ПРО» и использованы для организации связи между мобильными устройствами поддержки дополненной реальности.

Технический уровень предложенных решений отвечает современным требованиям.

Генеральный директор



Ильин А.А.

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»



Утверждаю
Проректор по учебной работе
ФГБОУ ВО ТулГУ
д.т.н., доцент
В.В. Котов
2024 г.

«02» 09

АКТ

**о внедрении результатов научно-исследовательской работы
в учебный процесс**

Настоящий акт подтверждает, что научные положения и результаты диссертационного исследования Пестина Максима Сергеевича; выполненного по теме «Методы и алгоритмы взаимодействия программного обеспечения узлов беспроводных децентрализованных сетей передачи данных при решении задач маршрутизации» успешно внедрены в учебный процесс и используются во время чтения лекций, а также в ходе проведения лабораторных и практических занятий по дисциплинам «Сетевое программирование», «Проектирование и тестирование программного обеспечения», «Сетевое программирование и протоколы передачи данных» для студентов, обучающихся на кафедре «Вычислительная техника» ТулГУ по специальности 09.06.01 Информатика и вычислительная техника, профилям: «Программное обеспечение средств вычислительной техники и автоматизированных систем», «Электронно-вычислительные машины, комплексы, системы и сети».

Директор института прикладной
математики и компьютерных наук
д.т.н., доц.

А.А. Сычугов

Заведующий кафедрой
вычислительной техники
д.т.н., проф.

А.Н. Ивутин