

Санкт-Петербургский государственный университет

Денисов Лев Евгениевич

Выпускная квалификационная работа

Балансировка нагрузки на сотах с целью
максимизации пропускной способности
базовой станции LTE

Уровень образования: бакалавриат

Направление *02.03.03 «Математическое обеспечение и администрирование
информационных систем»*

Основная образовательная программа *СВ.5162.2021 «Технологии программирования»*

Научный руководитель:
доцент кафедры системного программирования, к.ф.-м.н. Д. В. Луцев

Рецензент:
главный инженер-программист ООО «НТР» В. П. Рыжов

Санкт-Петербург
2025

Saint Petersburg State University

Denisov Lev

Bachelor's Thesis

Load balancing on cells in order to maximize the bandwidth of the LTE base station

Education level: bachelor

Speciality *02.03.03 "Software and Administration of Information Systems"*

Programme *CB.5162.2021 "Programming Technologies"*

Scientific supervisor:

C.Sc., System Programming chair docent D. V. Luciv

Reviewer:

chief software engineer OOO «HTP» V. P. Ryzhov

Saint Petersburg
2025

Оглавление

Введение	4
1. Постановка задачи	5
2. Обзор предметной области	6
2.1. Архитектура сети LTE	6
2.2. Процедура измерения сигнала соседних сот	8
3. Обзор	9
3.1. Обзор существующих решений	9
3.2. Оценка загрузки сотового пользователя	12
3.3. Категории пользователей	13
3.4. Процедура установки первоначального соединения с пользователем . . .	14
3.5. Процедура передачи обслуживания	14
4. Разработка подходов балансировки нагрузки	15
4.1. Формулировка задачи балансировки нагрузки	15
4.2. Процедура перераспределения нагрузки	15
4.3. Контроль перегрузки	20
5. Реализация алгоритма	21
5.1. Проектирование алгоритма	21
5.2. Реализация алгоритма балансировки нагрузки	23
5.3. Конфигурация алгоритма	27
6. Апробация	30
6.1. Цель и исследовательские вопросы	30
6.2. Корректность работы	30
6.3. Демонстрация работы алгоритма	30
6.4. Вывод	34
Заключение	35
Список литературы	36

Введение

Мобильные сети представляют собой разновидность подвижной радиосвязи. Рост количества пользователей и увеличение потребности в передаче данных способствовали разработке и внедрению стандарта *Long Term Evolution* (LTE) [7], разработанного консорциумом *3rd Generation Partnership Project* (3GPP) в 2009 году. Введение данного стандарта позволило повысить эффективность работы мобильных сетей за счет перехода к новой сетевой архитектуре и отказа от канальной коммутации.

Изначально ключевым принципом функционирования мобильной сети было разделение зоны покрытия на отдельные ячейки (соты), каждая из которых обслуживалась отдельной базовой станцией, имеющей монолитную архитектуру. Для решения данной проблемы была разработана концепция Open Radio Access Network (O-RAN) [19], которая предусматривает открытую и модульную архитектуру базовых станций, позволившую разделить RAN на отдельные функции (рис. 1).

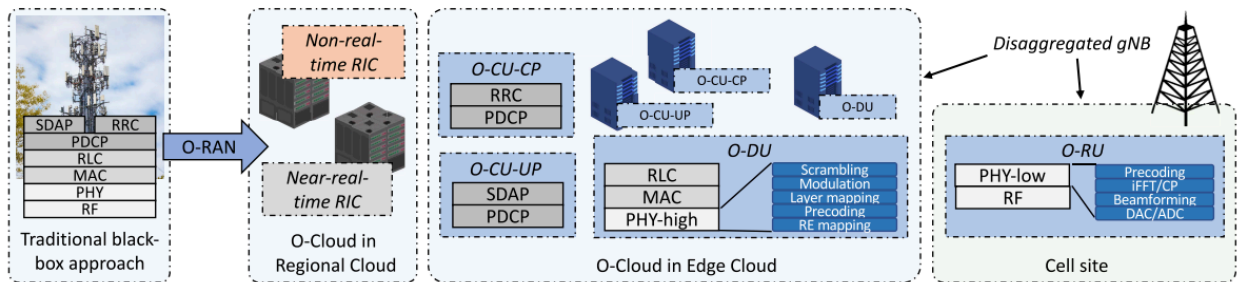


Рис. 1: Эволюция RAN (изображения были взяты из работы [21])

Одной из актуальных проблем функционирования мобильных сетей является неравномерное распределение нагрузки между сотовыми ячейками, что приводит к снижению пропускной способности сети и возникновению отказов радиосоединений. Для решения данной проблемы применяются механизмы балансировки нагрузки для перераспределения трафика, где высоко нагруженные соты передают часть пользователей на менее загруженные соседние ячейки.

В настоящее время популярность набирают сервисы, для которых не гарантируется минимальная скорость передачи данных. К таким сервисам можно отнести веб-серфинг, потоковое видео и обмен файлами. Они не предъявляют жёстких требований к пропускной способности и задержкам передачи данных. Однако, их массовое использование приводит к существенной нагрузке на сеть. Компания ООО «НТР» занимается разработкой отечественного программного обеспечения для базовых станций с использованием архитектуры O-RAN.

Целью данной работы является реализация алгоритма балансировки нагрузки между сотовыми ячейками. Алгоритм нацелен на перераспределение нагрузки между сотами в группе, которая, в рамках данной работы, называется сектором.

1. Постановка задачи

Целью данной работы является разработка алгоритма межчастотной балансировки нагрузки для распределенной базовой станции, ориентированного на пользователей, использующих сервисы, для которых не гарантируется минимальная скорость передачи данных. Алгоритм должен обеспечивать перераспределение нагрузки между сотовыми ячейками, работающими в разных частотных диапазонах, в пределах одного сектора. Для достижения цели были поставлены следующие задачи:

1. Выполнить обзор и описать существующие алгоритмы балансировки нагрузки.
2. Описать подходы, которые позволят балансировать нагрузку между соседними сотами, а также способы передачи обслуживания абонентских устройств между сотами во время балансировки.
3. Спроектировать и реализовать алгоритм балансировки нагрузки.
4. Протестировать работоспособность алгоритма и его совместимость с существующими процедурами.

2. Обзор предметной области

В данном разделе представлен краткий обзор архитектуры сети LTE.

2.1. Архитектура сети LTE

Архитектура сети LTE включает два компонента: Evolved Universal Terrestrial Radio Access Network (E-UTRAN) и Evolved Packet Core (EPC). Эти компоненты выполняют передачу данных и управление подключениями в мобильных сетях. Для взаимодействия компонентов сети LTE используются два основных интерфейса:

- S1 интерфейс [13] — используется для обмена сообщениями между eNodeB и EPC;
- X2 интерфейс [14] — используется для обмена сообщениями между соседними eNodeB.

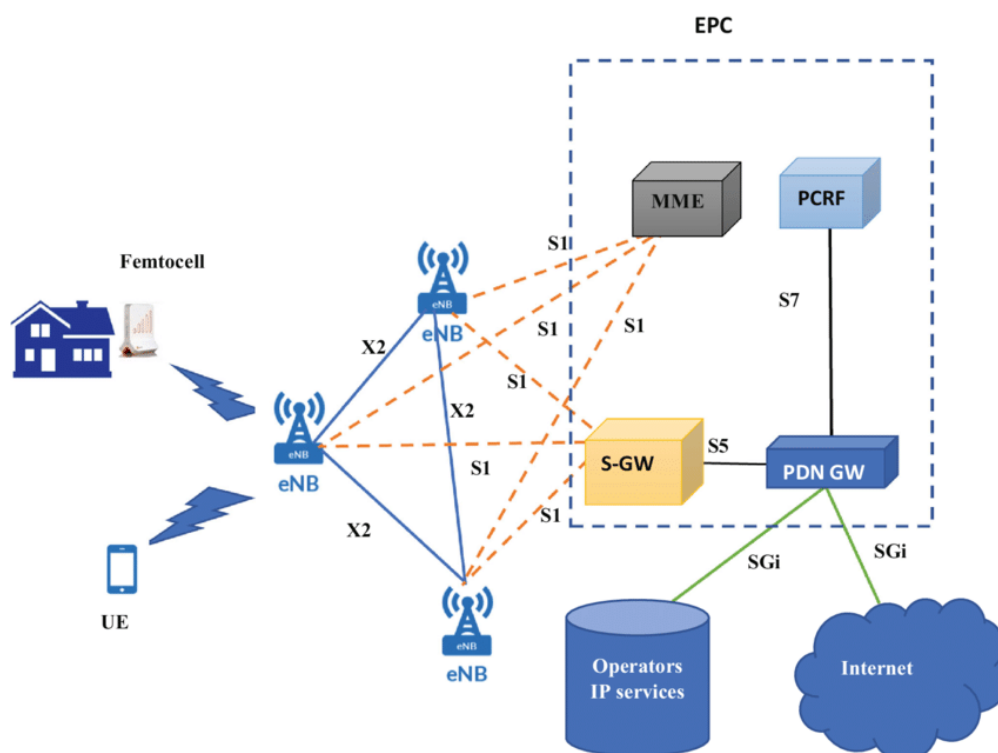


Рис. 2: Архитектура сети LTE (изображения были взяты из работы [23])

2.1.1. Базовая станция

Базовая станция (evolved Node B — eNodeB) является основным компонентом радиодоступа в сети LTE. Она выполняет функции управления радиоресурсами и обеспечивает передачу данных между пользовательским оборудованием (UE — User

Equipment) и EPC. Взаимодействие между UE и eNodeB осуществляется через протокольный стек (рис. 3). В архитектуре eNodeB предусмотрено разделение протоколов на две плоскости:

- Control Plane (C-Plane) — используется для передачи контрольной информации, необходимой для управления соединениями;
- User Plane (U-Plane) — используется для передачи пользовательского трафика.

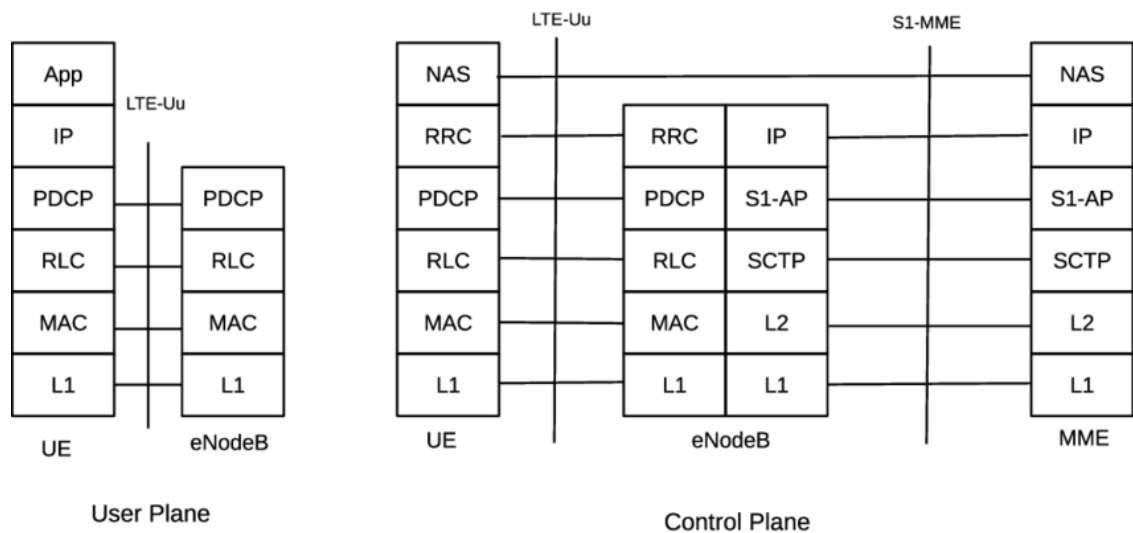


Рис. 3: Стек протоколов C-Plane и U-Plane

Для управления радиоресурсами в базовой станции используется механизм Admission Control (ADM). Модуль ADM занимается принятием решений о выделении радиоресурсов, предоставлении сервисов пользователям в зависимости от загрузки сети и обработкой запросов на передачу обслуживания пользователей между сотами.

2.1.2. Сектор

Множество сот, образованное за счёт направленного излучения антенн базовой станции. Каждая сота работает в уникальном частотном диапазоне в рамках сектора (рис. 4), что позволяет избежать интерференции между ними.

2.1.3. Сота

Базовая структурная единица мобильной сети, представляющая собой область радиопокрытия, обеспечиваемую одной базовой станцией или её сектором. Каждая сота функционирует в определённом частотном диапазоне в пределах выделенной полосы частот. Она использует радиочастотные ресурсы для передачи данных между базовой станцией и пользовательским оборудованием.

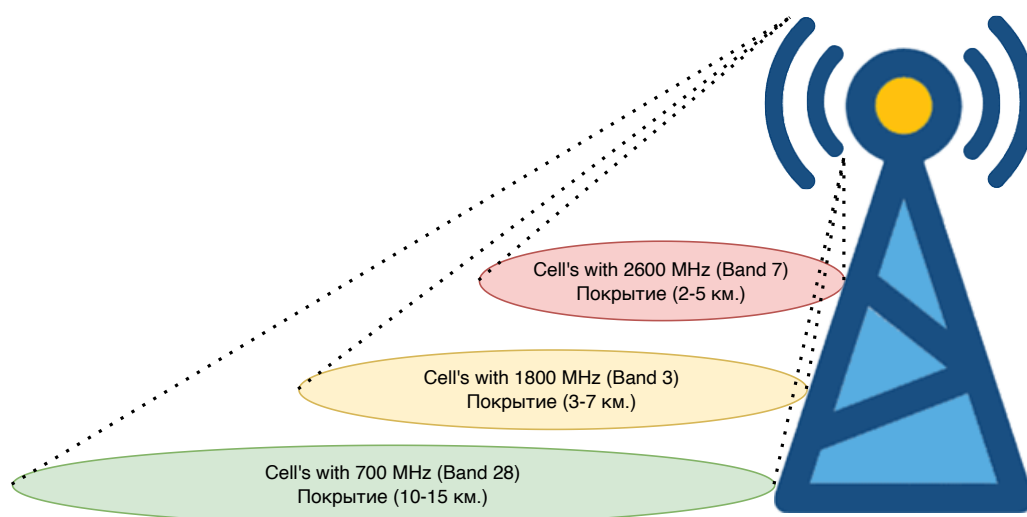


Рис. 4: Пример конфигурации радиомодулей в секторе

2.2. Процедура измерения сигнала соседних сот

Для уведомления базовой станции о качестве сигнала сот используется процедура отправки отчёта об измерениях (*Measurement Report*) [12]. В рамках этой процедуры пользователь отправляет информацию о качестве сигнала сот, которые были измерены за определённый промежуток времени, при наступлении определённого события. Конфигурация параметров измерений сот задаётся с помощью процедуры *RRCCONNECTIONReconfiguration* [11] в параметре *Measurement config*.

Для отправки отчётов об измерениях в сети LTE используются события измерений, которые инициируют передачу соответствующего отчёта пользовательским оборудованием при наступлении определённых условий. В таблице 1 представлены основные события измерений, применяемые в сетях LTE.

Таблица 1: Описание событий A1–A5 в LTE

Событие	Описание
A1	Сигнал обслуживающей соты становится лучше заданного порогового значения.
A2	Сигнал обслуживающей соты становится хуже заданного порогового значения.
A3	Сигнал соседней соты становится лучше, чем сигнал текущей соты на заданную величину.
A4	Сигнал соседней соты превышает заданное абсолютное пороговое значение.
A5	Сигнал обслуживающей соты ухудшается ниже первого порога и одновременно сигнал соседней соты превышает второй порог.

3. Обзор

В данном разделе рассматриваются существующие решения балансировки нагрузки в сети LTE, а также характеристики, позволяющие оценить загруженность соты пользователями, использующими сервисы, для которых не гарантируется минимальная скорость передачи данных.

3.1. Обзор существующих решений

3.1.1. Научные исследования

В исследовании [18, 16] представлен алгоритм *Mobility Load Balancing* (MLB) для сети LTE. Основной принцип его работы заключается в перераспределении нагрузки путём передачи обслуживания пользователей, находящихся на границе перегруженных сот, в менее загруженные соседние соты. Преимуществом такого подхода заключается в эффективном использовании радиоресурсов, поскольку пользователи, находящиеся на границе сот, обладают плохим качеством соединения с сотой, что требует выделения дополнительных ресурсных блоков для компенсации ухудшенных условий связи. Для принятия решений алгоритм учитывает измерения, полученные от соседних сот по событию A3 и текущей, а также загруженность на основе метрики *Resource Block Utilization Ratio* (RBUR).

В исследовании [2] рассматривается координация между алгоритмами MLB и механизмом Handover Parameter Optimization (HPO) [5]. Основная идея HPO заключается в динамическом регулировании параметров мобильности в конфигурации измерений, таких как гистерезис и время до срабатывания на основе эффективности процедуры передачи обслуживания (Handover Performance Indicators (HPI)). Данный подход позволяет уменьшить количество неудачных процедур передачи обслуживания [6]. Для взаимодействия алгоритмов MLB и HPO был предложен механизм координации (рис. 5).

В исследовании [24] был рассмотрен параметр Handover Margin (НОМ), который определяет пороговые значения для передачи обслуживания, учитывая загрузку соты на основе RBUR и характеристики передаваемого трафика.

В исследовании [22] предложен алгоритм MLB на основе механизма Cell Reselection (CR). Данный подход предполагает изменение параметров CR в режиме ожидания у пользовательского оборудования, чтобы перенаправлять пользователей в слабонагруженные соты. После перехода пользовательского оборудования в активное состояние, оно будет иметь активное соединение с менее загруженной сотой.

Поскольку балансировка достигается путём корректировки параметров мобильности, в работе [17] проведён анализ влияния мобильности пользователей на алгоритмы MLB. Исследование включает компьютерное моделирование различных сце-

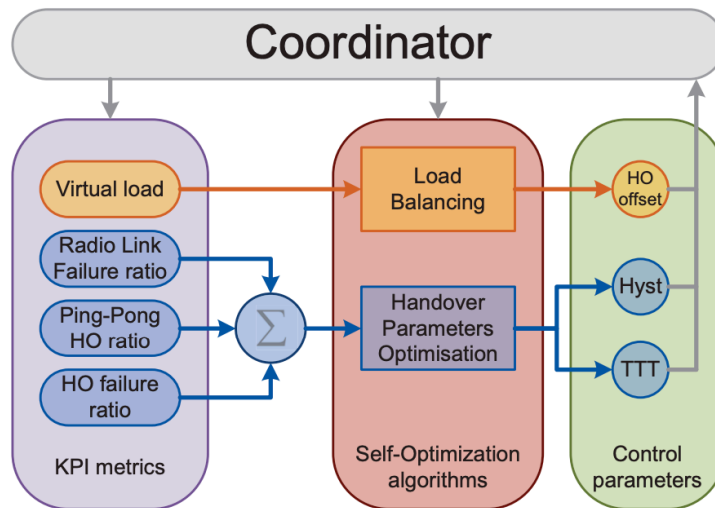


Рис. 5: Схема координатора для алгоритмов НРО и LB (изображения были взяты из работы [2])

нариев движения абонентов. Результаты показали, что наибольшая эффективность алгоритмов MLB достигается для пешеходов с круговым характером движения и для транспортных средств, движущихся по маршрутам прямоугольной формы.

В исследовании [3] представлен алгоритм *Utility-based Mobility Load Balancing* (UMLB), который использует A4-событие для сбора информации о пользователях, находящихся на границе соты, а также об их соседних ячейках. В отличие от предыдущих алгоритмов, учитывающих только параметры, полезные с точки зрения оператора (например, загруженность сот), UMLB анализирует показатели, значимые для пользователей. В частности, алгоритм учитывает такие метрики, как задержка и скорость передачи данных.

3.1.2. Решения промышленных вендоров

В дополнение к научным исследованиям в рамках данной работы были рассмотрены подходы к реализации механизмов балансировки нагрузки, используемые двумя ведущими вендорами в области разработки оборудования для базовых станций — Nokia и Huawei.

В работе [20] исследователи компании Nokia предлагают модифицированный подход к алгоритму MLB. В отличие от классических решений, ориентированных исключительно на перераспределение радионагрузки, предлагаемый подход дополнительно учитывает состояние транспортной сети.

В техническом документе компании Huawei [8] описывается детальная реализация алгоритма MLB. Однако в качестве метрики загруженности соты используется не только RBUR, но и счётчики активных пользователей.

3.1.3. Вывод

Проведенный анализ показал, что большинство существующих алгоритмов используют показатель RBUR в качестве основной метрики загрузки. Однако, такой подход эффективен для сервисов, для которых гарантируется минимальная скорость передачи данных. Для них планировщик на уровне MAC выделяет фиксированное количество ресурсных блоков для передачи данных. Для сервисов, для которых не гарантируется минимальная скорость передачи данных, планировщик не гарантирует фиксированное количество ресурсных блоков для передачи данных. В дополнении к этому, существующие алгоритмы, как правило, не учитывают качество радиоканала и текущую активность пользователя, полагаясь исключительно на значения RSRP при выборе пользователей для передачи обслуживания во время разгрузки.

Кроме того, в рассмотренных алгоритмах, принятие решений о балансировке нагрузки осуществляется локально на уровне отдельных базовых станций. Это требует постоянного обмена сигнальными сообщениями между базовыми станциями через интерфейсы S1 или X2, которое выполняется с определенной задержкой. В данной работе рассматривается распределённая архитектура базовой станции (рис. 6), что позволяет хранить актуальную информацию о загрузке большого количества сот в централизованном месте.

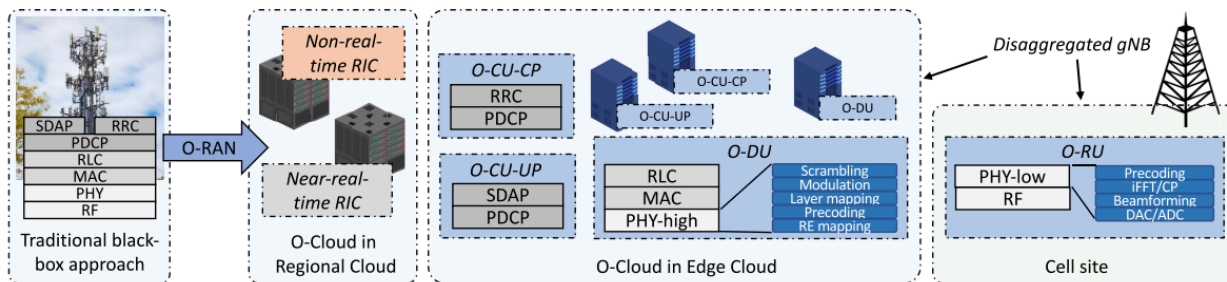


Рис. 6: Эволюция RAN (изображения были взяты из работы [21])

Стоит также отметить, в рамках текущей работы рассматривается задача межчастотной балансировки нагрузки, при которой перераспределение пользователей осуществляется между сотами, функционирующими на разных частотных диапазонах. Такой подход накладывает ряд дополнительных ограничений, ключевым из которых является необходимость использования механизма *Measurement Gap* [12]. Во время этого процесса пользовательское оборудование приостанавливает передачу и приём данных, чтобы измерить сигналы соседних сот. Это приводит к временным потерям данных и задержкам.

В рамках дальнейшей работы планируется адаптация и расширение представленных в литературе подходов для реализации межчастотной балансировки нагрузки.

3.2. Оценка загрузки сотовой сети пользователем

Для передачи информации по восходящему и нисходящему каналам протокольный уровень MAC выделяет пользователям радиоресурсные блоки. Передача данных в LTE может быть представлена в виде ресурсной сетки, которая описывает структуру распределения частотно-временных ресурсов.

Количество выделяемых ресурсных блоков зависит от пропускной способности канала. Например, если пропускная способность канала составляет 1,4 МГц, то в каждом сабфрейме выделяется 6 ресурсных блоков.

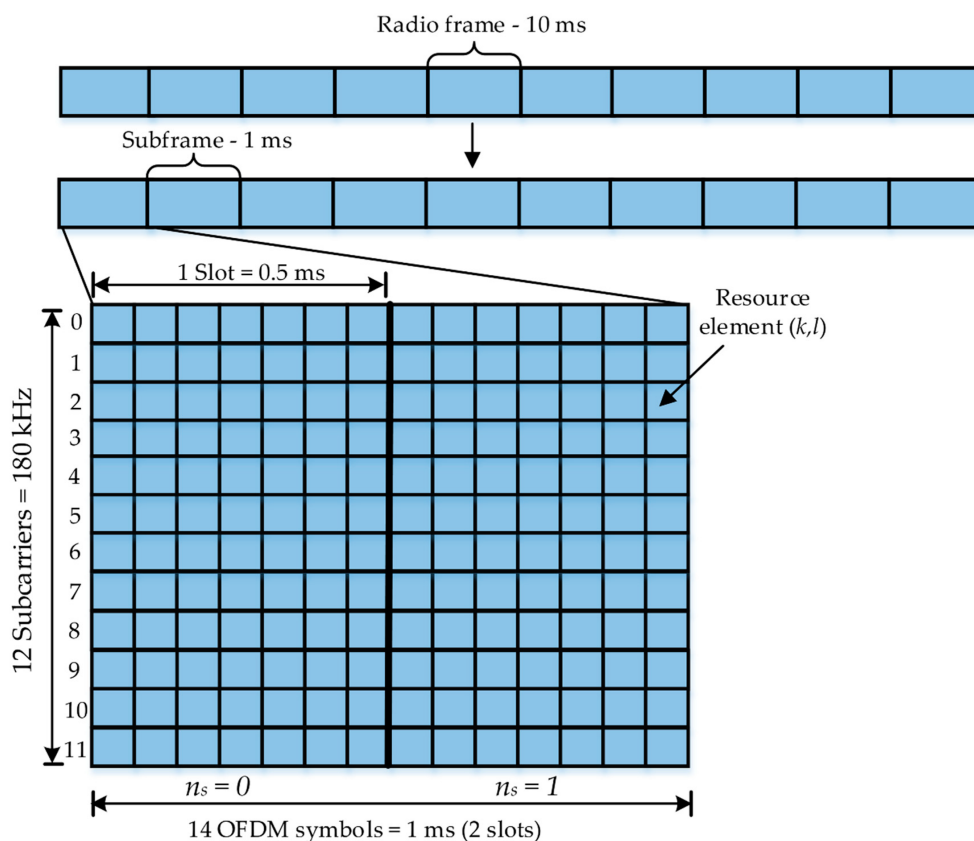


Рис. 7: Ресурсная сетка в MAC (изображения были взяты из работы[10])

Каждый ресурсный блок содержит:

- 12 поднесущих в частотной области;
- 7 или 6 OFDM-символов на каждый слот во временной области, в зависимости от используемого циклического префикса.

Каждый OFDM-символ может кодировать определённое количество бит информации, что зависит от используемой схемы модуляции:

- QPSK (Quadrature Phase Shift Keying) — 2 бита на символ;

- 16QAM (16 Quadrature Amplitude Modulation) — 4 бита на символ;
- 64QAM — 6 бит на символ;
- 1024QAM — 10 бит на символ.

Выбор схемы модуляции зависит от радиоусловий, в которых находится пользователь. Для определения оптимальной схемы модуляции используется Signal-to-Interference-plus-Noise Ratio (SINR) — отношение мощности полезного сигнала к сумме мощности помех и шума. Чем выше значение SINR, тем более сложную схему модуляции можно применить, что позволяет передавать больше данных за единицу времени.

В данной работе значение SINR пользовательского устройства будет учтен при разработке алгоритма межчастотной балансировки нагрузки, поскольку его значение напрямую влияет на количество дополнительных ресурсных блоков, выделяемых планировщиком MAC, для передачи данных.

3.3. Категории пользователей

3.3.1. Активные пользователи

Для перераспределения нагрузки в рамках балансировки будут учитываться активные пользователи, передающие большой объём трафика по восходящему каналу в текущий момент. Для идентификации активных пользователей используется передача сообщений с протокольного уровня RRC на MAC, в котором содержится идентификатор пользователя и статус его текущей активности.

На данном этапе реализации используется жадный алгоритм, который не учитывает степень активности пользователей, а лишь определяет их состояние. В будущем планируется интеграция более сложных механизмов оценки активности, например, на основе анализа интенсивности и типа трафика.

3.3.2. Пользователи в режиме ожидания

Если пользовательское оборудование имеет активное RRC-соединение (*Radio Resource Control Connection*) с сотой, но при этом оно не передаёт и не принимает пользовательский трафик, протокольный уровень MAC отправляет на протокольный уровень RRC сообщение о не активности пользователя. RRC в свою очередь запускает таймер, по истечению которого пользовательское устройство будет считаться не активным. В этом случае протокольный уровень RRC инициирует для устройства процедуру *RRCConnectionRelease*. Данная процедура направлена на освобождение радиоресурсов, занятых пользовательским оборудованием, и его переводом в режим ожидания (IDLE mode).

3.4. Процедура установки первоначального соединения с пользователем

Процесс первоначального подключения пользователя к базовой станции начинается с отправки запроса *RRCConnectionRequest* [11] пользовательским оборудованием. После получения данного запроса базовая станция передаёт его в модуль ADM, который принимает решение об установлении RRC-соединения с данным пользователем. В зависимости от принятого решения возможны два сценария:

1. В случае положительного решения, базовая станция отправляет пользователю сообщение *RRCConnectionSetup* и ожидает *RRCConnectionSetupComplete* в ответ.
2. В случае отказа в установлении соединения, базовая станция отправляет пользователю сообщение *RRCConnectionRelease* с указанием причины отклонения запроса.

3.5. Процедура передачи обслуживания

Запуск процедуры передачи обслуживания (*Handover*) пользователя во время его перемещения между сотами осуществляется базовой станцией. Для уведомления пользователем о качестве сигналов соседних сот, работающих в том же частотном диапазоне, в котором находится пользовательское устройство, используется событие A3. При его наступлении пользовательское оборудование отправляет отчёт об измерениях, если уровень сигнала соседней соты превышает уровень сигнала текущей соты на заданное значение. Для уведомления о качестве сигналов соседних сот, работающих в другом частотном диапазоне, используется событие A5. Однако, межчастотные измерения активируются только в случае ухудшения качества сигнала текущей соты. Оценка качества сигнала с текущей сотой для включения или выключения A5 событий выполняется на основе A1 и A2 событий.

4. Разработка подходов балансировки нагрузки

В данном разделе будут описаны подходы балансировки нагрузки между сотами, работающими в разных частотных диапазонах в одном секторе, процедура разгрузки соты при неравномерной загрузке сот в секторе и процедура передачи обслуживания пользователя во время разгрузки соты.

4.1. Формулировка задачи балансировки нагрузки

Основная идея балансировки нагрузки состоит в равномерном распределении активных пользователей между сотами в каждом секторе.

Обозначим множество секторов как: $S = \{s_1, s_2, \dots, s_n\}$. Для каждого сектора s_i определим множество сот, принадлежащим данному сектору, как: $C_{s_i} = \{c_{i,1}, c_{i,2}, \dots, c_{i,m}\}$. Для оценки загрузки сот в рамках предложенной работы введём функцию загрузки соты $L(c)$, которая определяет количество активных пользователей в соте c в конкретный момент времени.

Для оценки равномерности загрузки сот в секторе s введём стандартное отклонение σ , которое характеризует отклонение значения загрузки сот от среднего значения загрузки всех сот в секторе. Данное значение вычисляется по формуле:

$$\sigma(s) = \sqrt{\frac{\sum_{c \in C_s} (L(c) - L)^2}{|C_s| - 1}}$$

где L — среднее количество активных пользователей в сотах C_s сектора s

Значение $\sigma(s)$ лежит в диапазоне $[0, +\infty)$, где меньшее значение соответствует более равномерному распределению нагрузки между сотами в секторе. Таким образом, целью данной работы является минимизация значения $\sigma(s)$ для каждого сектора s .

Будем считать, что нагрузка распределена равномерно, если выполняется следующее неравенство:

$$\sigma(s_i) \leq th_\sigma$$

где th_σ — пороговое значение, определяющее равномерность распределения нагрузки между сотами в секторе. Если значение $\sigma(s_i)$ превышает th_σ , это указывает на неравномерное распределение нагрузки.

4.2. Процедура перераспределения нагрузки

Процедура перераспределения нагрузки в секторе s выполняется, если загрузка по всем сотам в секторе s распределена неравномерно. Для этого анализируется

текущая загруженность сот в секторе s , после чего выбирается сота с максимальной загруженностью для её разгрузки.

Основная цель данной процедуры заключается в переводе одного из пользователей на соседнюю соту в текущем секторе для минимизации значения $\sigma(s)$. Однако, немедленная передача пользователя на наименее загруженную соту невозможна в связи со следующими ограничениями:

1. **Поддержка частотных диапазонов** — пользовательское оборудование может не поддерживать определённые частотные диапазоны, что делает невозможным передачу его обслуживания на конкретную соту.
2. **Необходимость в конфигурации измерений** — передача пользователя на соседнюю соту требует актуальной информации об уровнях сигналов соседних сот сектора от пользователя. Поскольку межчастотные измерения не выполняются постоянно, необходимо предварительно отправить конфигурацию на включение измерений на пользовательское устройство и дождаться отчёта об измерениях.
3. **Задержка между этапами передачи конфигурации и получением отчёта об измерениях** — после отправки конфигурации на включение измерений пользователю присутствует временная задержка до получения отчёта. В связи с этим, перед началом процедуры передачи обслуживания после получения отчёта об измерениях от пользователя требуется повторная оценка загруженности сот в секторе.

Выбор пользователя, соты и процедура передачи обслуживания с включением межчастотных измерений будут детально рассмотрены ниже.

4.2.1. Выбор пользователя и целевой соты

Основной целью данной процедуры является выбор пользователя на перегруженной соте и отправка запроса на процедуру передачи обслуживания на менее загруженную соту с включением межчастотных измерений.

Критерием для выбора пользователя, подлежащего передаче, является низкое значение его параметра SINR, которое указывает на плохое качество его радиоканала. Дополнительным фактором при выборе пользователя является совместимость частот сотовой связи, поддерживаемым им. Информация о поддерживаемых частотах пользовательским устройством хранится в параметре `RadioCapability` [1], доступном в контексте пользователя на базовой станции. Если в перегруженной соте не удаётся определить пользователя, удовлетворяющего критериям передачи, рассматривается другая менее загруженная сота, загруженность которой не ниже средней по сектору.

Целевая сота для передачи обслуживания пользователя выбирается из полученного отчёта об измерениях на основе минимальной загруженности. Диаграмма работы процедуры представлена ниже (рис. 9).

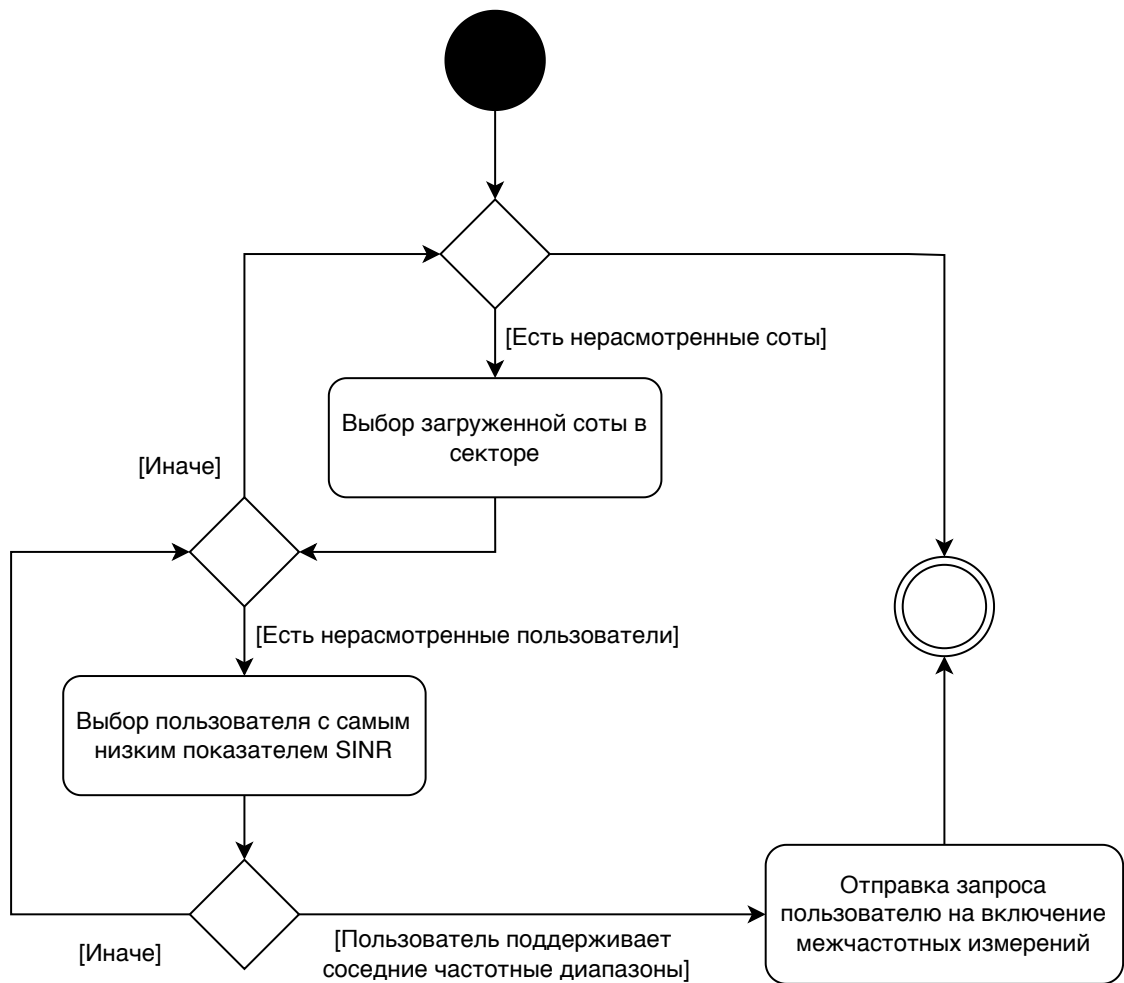


Рис. 8: Выбор соты и пользователя для перераспределения нагрузки между сотами в секторе

4.2.2. Передача обслуживания с включением межчастотных измерений

Процедура передачи обслуживания с включением межчастотных измерений представляет собой отдельную процедуру, которую необходимо реализовать для балансировки нагрузки. Она состоит из следующих этапов:

1. Отправка пользователю сообщения *RRCConnectionReconfiguration*, содержащий *Measurement config* с включением A4-событием для межчастотных измерений, и ожидание от него сообщения *RRCConnectionReconfigurationComplete*, подтверждающего применение конфигурации.

2. Ожидание отчёта об измерениях соседних сот по событию A4 от пользовательского устройства.
3. Запуск процедуры передачи обслуживания с помощью отправки пользователю сообщения *RRConnectionReconfiguration* после получения отчёта об измерениях и выбора из него целевой соты с учётом текущей загрузки сот в секторе.

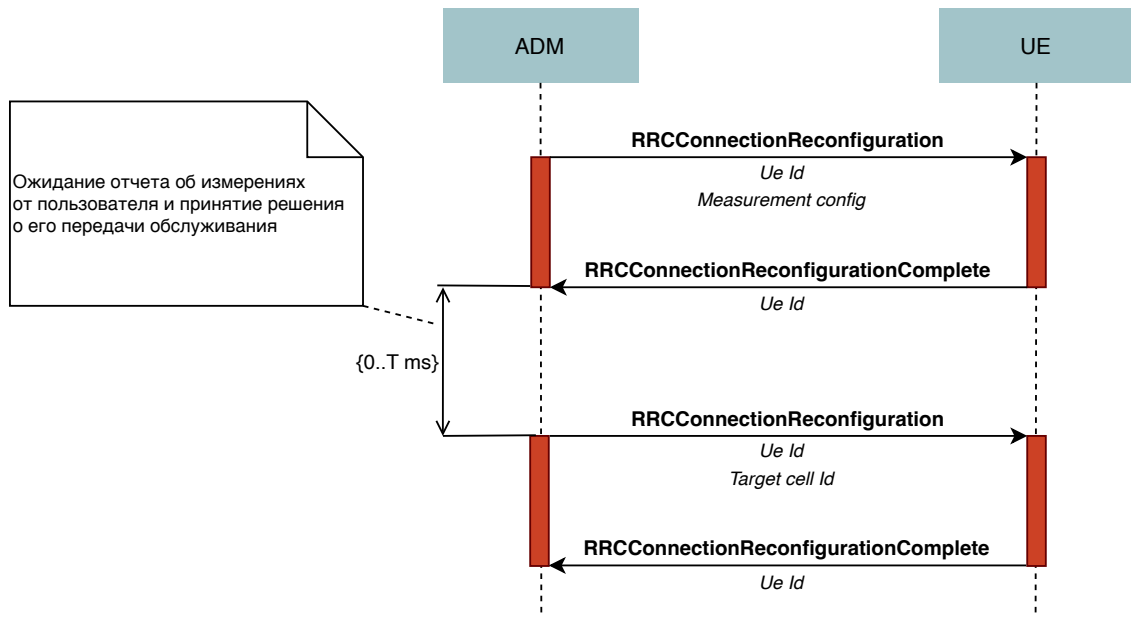


Рис. 9: Включение межчастотных измерений и процедура передачи обслуживания пользовательского оборудования

Основным недостатком данной процедуры является задержка, возникающая на этапе ожидания отчёта об измерениях (шаг 2). В течение данного ожидания возможна инициация других процедур, таких как *Circuit-Switched Fallback* [4] или *E-RAB Modify Request* [13], которые могут потребовать отмены межчастотных измерений.

В дополнении к этому, для предотвращения бесконечного ожидания отчёта об измерениях, необходим таймер ожидания, по истечении которого пользователю отправится сообщение *RRConnectionReconfiguration*, отключающее межчастотные измерения, если отчёт не был получен.

4.2.3. Условия для проверки загрузки

Оценка загрузки сот в секторе будет выполняться на основе событий, происходящих в модуле ADM. К ним относятся:

- получение сообщения от PCDP в ADM, информирующего об активности конкретного пользователя;

- передача пользователя из другого сектора в текущий в рамках процедуры передачи обслуживания (handover);
- подключение нового пользователя к соте;
- отключение пользователя от соты;
- изменение типа обслуживаемого сервиса у пользователя.

4.2.4. Способ измерения соседних сот

Условием для передачи пользователем отчётов об измерениях сигналов соседних сот в рамках процедуры разгрузки соты будет наступление события A4 во время межчастотных измерений. Выбор данного события обусловлен следующими факторами:

1. События A1, A2, A3 и A5 зарезервированы для другой функциональности.
2. Условия срабатывания события A4 соответствуют требованиям текущей функциональности, так как оно фиксирует момент, когда сигнал соседней соты превышает определённый порог.

Событие A4 инициируется в момент, когда уровень сигнала от соседней базовой станции превышает установленный порог. Формально это условие можно представить следующим выражением:

$$Meas_{neigh} + O_{neigh,freq} + O_{neigh,cell} - Hyst > Threshold$$

где:

- $Meas_{neigh}$ — текущий уровень сигнала от соседней соты;
- $O_{neigh,freq}$ и $O_{neigh,cell}$ — приоритеты соседней соты, задаваемые в зависимости от частоты. Данные параметры могут принимать значения в диапазоне от -24 до $+25$ дБ;
- $Hyst$ — гистерезис, используемый для предотвращения частых переключений состояний (активное или неактивное событие). Может принимать значения от 0 до 30 дБ;
- $Threshold$ — пороговое значение мощности сигнала, которое должно быть превышено для активации события A4.

4.2.5. Вывод

Таким образом, для реализации процедур для разгрузки соты необходимо тщательное проектирование, направленное на оценку влияния разрабатываемой функциональности на работу модуля ADM. Кроме того, важно учитывать взаимодействие предложенных алгоритмов с другими процедурами в сети.

4.3. Контроль перегрузки

В данном разделе рассматриваются подходы для оптимизации существующих процедур, направленные на равномерное распределение активных пользователей на сотах в секторе, а также на снижение вероятности возникновения перегрузки.

4.3.1. Выбор соты во время мобильности пользователя

Во время мобильности пользователя, в случае ухудшения качества соединения с текущей сотой, пользовательское оборудование может передавать отчёты об измерениях по наступлению события A5. После получения отчёта об измерениях ADM модулем принимается решение о передаче обслуживания пользователя на одну из сот, полученных в отчёте. Поскольку данный отчёт содержит информацию о соседних сотах, которые работают в отличных от текущей частотных диапазонах, то при выборе соты, помимо качества сигнала RSRQ и RSRQ будет также учитываться их загруженность.

4.3.2. Балансировка неактивных пользователей

В случае, если пользовательское устройство в течение определённого промежутка времени, заданного на уровне MAC, не передаёт и не принимает данные, базовой станцией инициируется процедура *RRCConnectionRelease*. Для равномерного распределения загруженности между сотами модуль ADM может включить в сообщение *RRCConnectionRelease* параметр *IdleModeMobilityControlInfo*. Этот параметр предоставляет приоритеты повторного выбора ячеек (*Cell Reselection Priorities*), которые определяют предпочтительные соты для подключения пользователя при его повторном выходе из режима ожидания.

4.3.3. Выбор соты в секторе во время процедуры подключения

Во время процедуры установления первоначального соединения пользователя с базовой станцией модуль ADM может выбрать предпочтительную соту в секторе для установления RRC-соединения с пользователем. Для этого модуль ADM может включить в сообщение *RRCConnectionRelease* параметр *RedirectedCarrierInfo*, указывающий целевую соту, на которую следует перенаправить обслуживание пользователя.

5. Реализация алгоритма

Описанные подходы в разделе 4 должны быть реализованы в соответствии с требованиями стандартов и рекомендаций, обеспечивающих эффективное управление мобильностью в сетях LTE. С учётом сложности предметной области и взаимодействия алгоритмов с другими компонентами сети, необходимо придерживаться структурированного подхода к разработке, начиная с формализации требований и заканчивая реализацией алгоритма в рамках архитектуры модуля ADM.

5.1. Проектирование алгоритма

Для реализации алгоритма необходимо принимать во внимание множество факторов: от поддержания возможности удобной модификации разрабатываемого алгоритма до возможности совместимости с существующими алгоритмами в ADM модуле. В данной секции сформулированы требования к разрабатываемым алгоритмам.

5.1.1. Разработка требований

Разрабатываемый алгоритм должен балансировать активных пользователей, использующих сервисы, для которых не гарантируется минимальная скорость передачи данных, в пределах одного сектора базовой станции. Проверка загрузки сектора должна осуществляться при выполнении определенных событий в ADM модуле.

Во время проверки перегруженности сектора, а также при выполнении процедуры разгрузки соты, алгоритм должен учитывать пользователей, находящихся в процессе передачи обслуживания. Это позволяет снизить вероятность повторной перегрузки сектора за счёт предотвращения избыточного перемещения пользователей.

С учётом предполагаемой сложности алгоритма, архитектура должна быть спроектирована таким образом, чтобы обеспечивалась её модульность. Это позволит вносить изменения в отдельные компоненты алгоритма без необходимости модификации других частей, тем самым обеспечивая гибкость и расширяемость при последующей доработке и развитии решения.

5.1.2. Архитектура алгоритма

Диаграмма компонентов алгоритма представлена на рисунке 10. Описание каждого компонента, представленного на диаграмме, находится ниже. Для верификации корректности работы каждого компонента разработаны отдельные модульные тесты. Компонент `UnitTests` представляет набор модульных тестов, предназначенных для проверки функциональной целостности алгоритма, а также его совместимости с другими алгоритмами модуля ADM. Модульные тесты реализованы с использованием библиотеки `GoogleTest` [9]. Преимуществом предложенного архитектурного решения

является разделение логики алгоритма на независимые компоненты, что обеспечивает простую модифицируемость, повышенную тестируемость и гибкость интеграции.

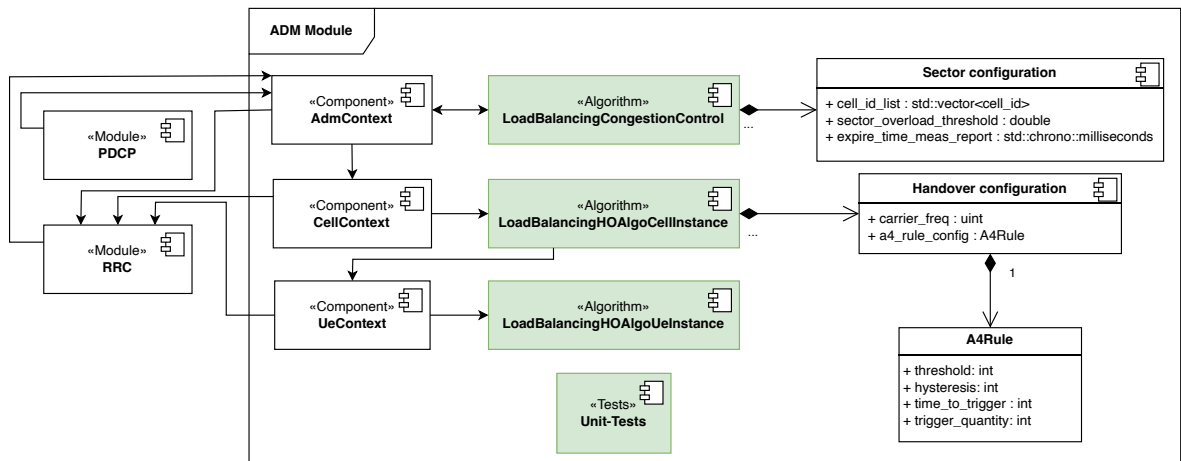


Рис. 10: Диаграмма компонентов алгоритма

Класс `AdmContext` отвечает за прием и обработку сообщений, поступающих других модулей. При получении сообщения, относящегося к конкретному пользователю или соте, оно передаётся для дальнейшей обработки в соответствующий контекст соты `CellContext`, который управляется отдельным потоком. Компонент `LoadBalancingCongestionControl` находится в классе `AdmContext`. Он реализует механизм контроля загруженности сот и содержит актуальную информацию о текущей загруженности всех сот БС. При изменении загруженности одной из сот, модуль проверяет равномерность распределения загруженности в секторе, к которому относится соответствующая сота. В случае обнаружения дисбаланса нагрузки инициируется процедура разгрузки: выбирается сота с максимальной загруженностью в секторе, и ей отправляется запрос на запуск процедуры перераспределения пользователей.

Компонент `LoadBalancingHOAlgoCellInstance` является компонентом класса `CellContext`. Он хранит информацию о текущей загруженности соты, выполняет процедуру разгрузки соты, а также контролирует процедуру передачи обслуживания во время разгрузки, включая управление межчастотными измерениями. В случае изменения загруженности соты или запуска процедуры разгрузки, данный модуль уведомляет `LoadBalancingCongestionControl`. В процессе разгрузки данный модуль взаимодействует с классом `UeContext`, отправляя команды для реконфигурации и передачи обслуживания.

Для хранения и управления информацией, связанной с отдельным пользователем, используется класс `UeContext`. Он предоставляет доступ к информации пользовательского устройства и для выполнения процедур. Компонент `LoadBalancingHOAlgoUeInstance` находится в `UeContext` и он отвечает за конфигурацию `Measurement config`, необходимого для включения или отключения межчастот-

ных измерений в рамках процедуры разгрузки соты.

5.2. Реализация алгоритма балансировки нагрузки

В данной секции представлено описание реализации алгоритма балансировки нагрузки. На диаграмме 10 представлена общая схема архитектуры алгоритма, отображающая основные её компоненты и их взаимодействие. Диаграмма классов, отражающая основные структуры и зависимости между элементами алгоритма, приведена на диаграмме 11.

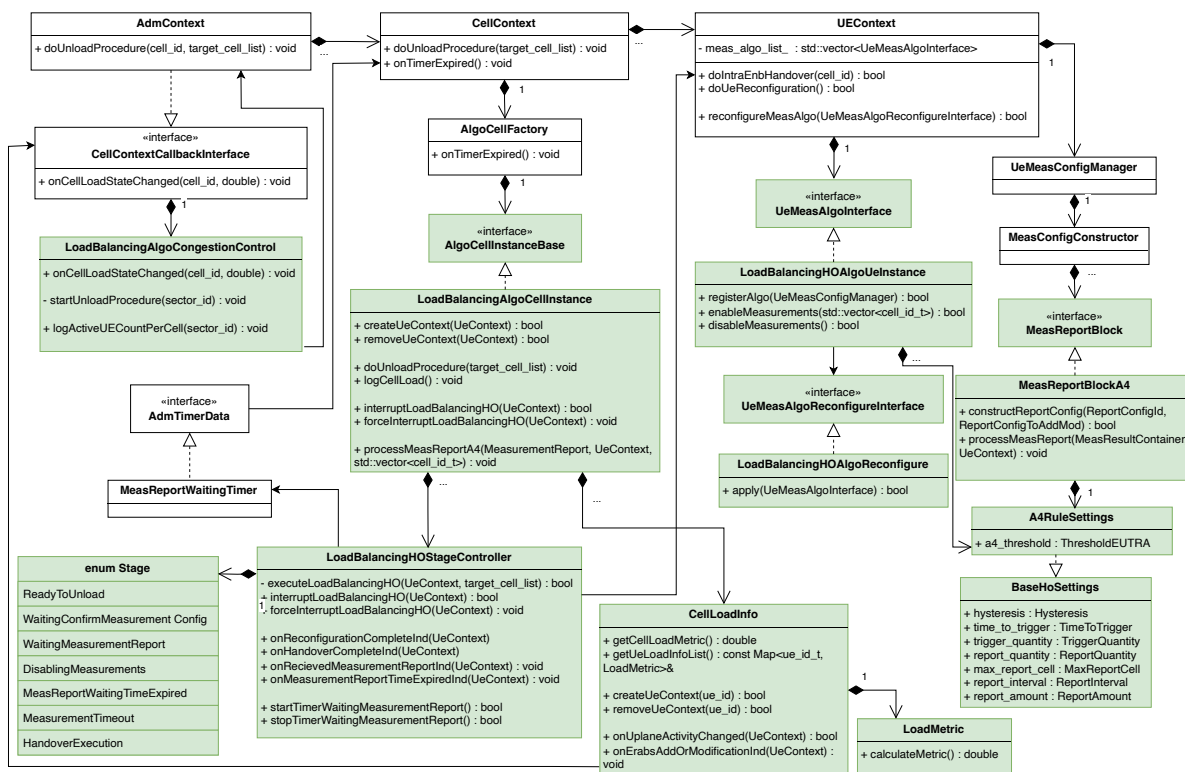


Рис. 11: Диаграмма классов алгоритма

5.2.1. LoadBalancingAlgoCongestionControl

Класс `LoadBalancingAlgoCongestionControl`, находящийся в `AdmContext`, отвечает за контроль и мониторинг загруженности всех сот в БС. Основные методы данного класса представлены ниже.

- `onCellLoadStateChanged(...)` — вызывается при поступлении уведомления от класса `CellContext` об изменении загруженности соты. При вызове метода обновляется внутренняя информация о текущей загруженности соты, после чего выполняется оценка распределения нагрузки в секторе. В случае выявления перекоса нагрузки инициируется процедура разгрузки;

- **startUnloadProcedure (...)** — вызывается при обнаружении перегрузки в секторе. В рамках этого метода определяется наиболее загруженная сота в перегруженном секторе, для которой инициируется процедура разгрузки.

Процедура разгрузки запускается с помощью метода **doUnloadProcedure (...)**, где:

- **cell_id** — идентификатор соты, подлежащей разгрузке;
- **target_cell_list** — список соседних сот, отсортированных по убыванию загруженности, среди которых будет выбираться целевая сота для перераспределения трафика.

5.2.2. LoadBalancingAlgoCellInstance

Класс **LoadBalancingAlgoCellInstance** функционирует на уровне **CellContext** и является компонентом, реализующим логику алгоритма балансировки нагрузки на уровне соты. Основные задачи данного класса включают: хранение и обновление информации о загруженности соты в контексте текущего алгоритма и управление процедурой передачи обслуживания с включением межчастотных измерений в процессе разгрузки соты. Основные методы данного класса представлены ниже.

- **createUeContext (...)** и **removeUeContext (...)** — вызываются при создании, удалении и модификации пользовательских контекстов в модуле ADM;
- **doUnloadProcedure (...)** — вызывается при получении команды на разгрузку соты от класса **AdmContext**. В рамках этого метода осуществляется поиск подходящего пользователя, которого можно передать на менее загруженную соту, входящую в переданный список **target_cell_list**.

5.2.3. LoadBalancingHOStageController

Класс **LoadBalancingHOStageController** представляет реализацию машины состояний, управляющей процессом передачи обслуживания с включением межчастотных измерений для конкретного пользователя. Данный компонент координирует действия, связанные с конфигурацией измерений, ожиданием отчётов об измерениях и принятием решений о передаче обслуживания. Диаграмма машины состояний представлена на рисунке 12. Описание всех состояний представлено ниже.

- **ReadyToUnload** — пользователь находится в состоянии готовности к участию в процедуре разгрузки соты;
- **WaitingConfirmMeasurementConfig** — присваивается пользователю после запуска процедуры передачи обслуживания с включением межчастотных измерений. В этом состоянии пользователю отправляется сообщение

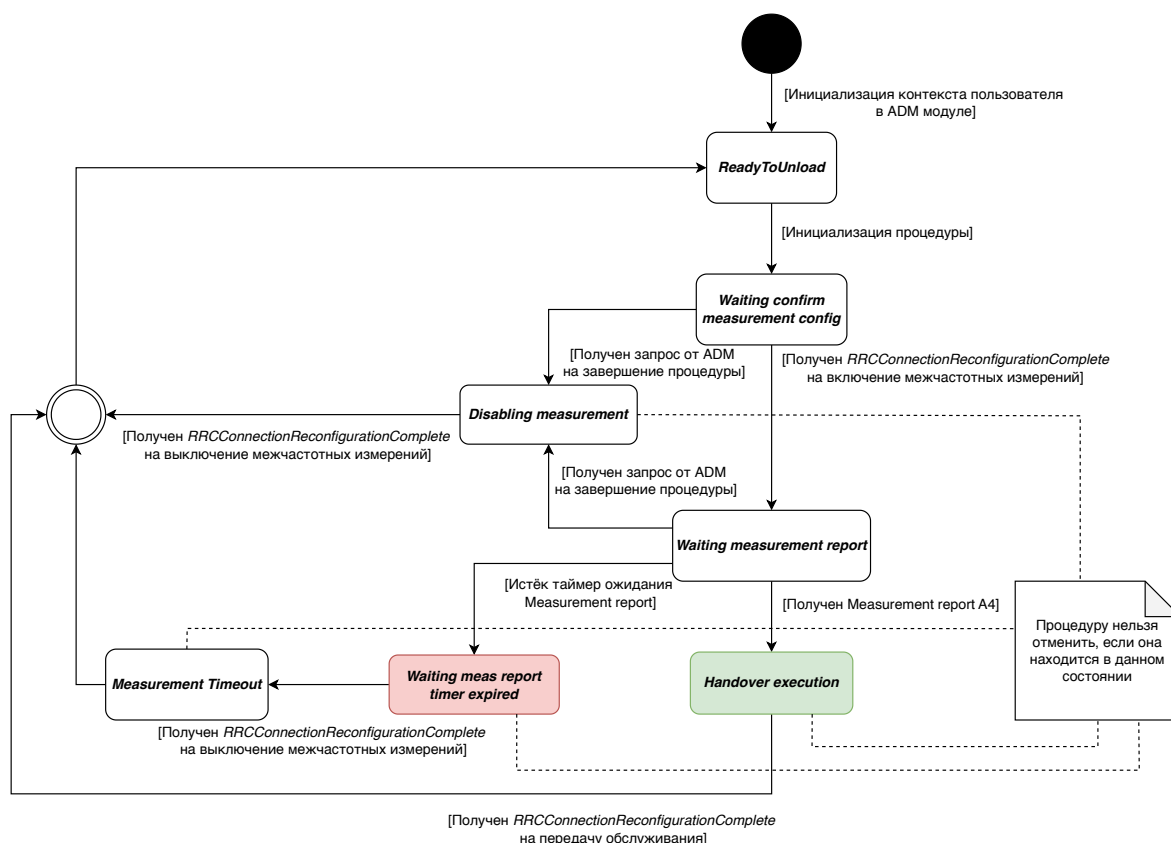


Рис. 12: Диаграмма состояний процедуры передачи обслуживания пользовательского оборудования с включением межчастотных измерений

RRCConnectionReconfiguration, содержащее *Measurement Config* с активацией измерений по событию A4;

- **Waiting measurement report** — устанавливается после получения от пользователя сообщения *RRCConnectionReconfigurationComplete*. С этого момента начинается ожидание отчёта об измерениях. На данном этапе пользователь может участвовать в других процедурах и продолжать передачу трафика. Также в этом состоянии запускается таймер, реализованный в классе *AdmTimerData*, по истечение которого ожидание отчёта будет прекращено;
- **Handover execution** — состояние, которое устанавливается для пользователя после получения ADM модулем отчёта об измерениях по событию A4 и выбора соты для передачи обслуживания;
- **Disabling measurement** — устанавливается, если инициирована отмена процедуры передачи обслуживания. Пользователю отправляется сообщение *RRCConnectionReconfiguration*, содержащее *Measurement Config* с деактивацией межчастотных измерений;

- **Waiting meas report timer expired** — состояние, соответствующее истечению таймера в фазе *WaitingMeasurementReport*. Поведение аналогично состоянию *DisablingMeasurement* — измерения отключаются;
- **Measurement Timeout** — присваивается после получения сообщения *RRCConnectionReconfigurationComplete* от пользователя, подтверждающего прекращение межчастотных измерений. Пользователь с этим состоянием исключается из дальнейших процедур разгрузки соты.

Методы `onReconfigurationCompleteInd(...)`, `onHandoverCompleteInd(...)`, `onReceivedMeasurementReportInd(...)` и `onMeasurementReportTimeExpiredInd(...)` обрабатывают соответствующие события, происходящие в процессе выполнения процедуры, и обеспечивают переход между этапами в соответствии с логикой алгоритма. Метод `executeLoadBalancingHO(...)` запускает процедуру передачи обслуживания с межчастотными измерениями для конкретного пользователя. Метод `interruptLoadBalancingHO(...)` используется для прерывания текущей процедуры передачи обслуживания в соответствии с логикой, описанной на рис. 12. Этот метод обеспечивает корректное завершение процедуры и переход в завершающее состояние без нарушения согласованности. Метод `forceInterruptLoadBalancingHO(...)` выполняет принудительное прерывание процедуры, не дожидаясь её завершения. Данный метод применяется в случае, если требуется приоритетное выполнение других системных процедур, связанных с данным пользователем (например, передача по CSFB или изменение типа сервиса).

5.2.4. CellLoadInfo

Класс `CellLoadInfo` предназначен для хранения информации о загруженности текущей соты. В составе `CellLoadInfo` реализован подкласс `LoadMetric`, который инициализируется для каждого пользователя, обслуживаемого данной сотой. Данный подкласс содержит информацию о загруженности соты конкретным пользователем. На текущем этапе реализации данная информация включает индикатор активности пользователя и тип используемого им сервиса.

5.2.5. LoadBalancingHOAlgoUeInstance

Класс `LoadBalancingHOAlgoUeInstance` реализован на уровне `UeContext` и отвечает за управление конфигурацией измерений (*Measurement config*) в рамках процедур включения и выключения межчастотных измерений. Метод `registerAlgo(...)` вызывается во время инициализации контекста пользователя в модуле ADM и осуществляет резервирование идентификаторов конфигураций измерений, что позволяет в дальнейшем модифицировать параметры измерений без необходимости полной

переинициализации конфигурации.

Класс `LoadBalancingH0AlgoReconfigure` реализует интерфейс, необходимый для координации с другими алгоритмами, использующие конфигурацию измерений.

Класс `BaseH0Settings` и его наследник `A4RuleSettings` содержат параметры конфигурации для межчастотных измерений, основанных на событии A4. Эти классы позволяют настраивать правила срабатывания события и учитывать особенности конкретной несущей частоты.

5.3. Конфигурация алгоритма

Для описания конфигурации алгоритма балансировки нагрузки используется язык ASN.1 (Abstract Syntax Notation One) [15]. Он позволяет создавать структуры данных, в которых будет описываться конфигурация алгоритма. На этапе компиляции исходные ASN.1-файлы преобразуются в эквивалентные структуры на языке C++, в которые впоследствии осуществляется десериализация конфигурационных данных.

Конфигурационные параметры задаются в формате XER (XML Encoding Rules) [15]. Общая схема процесса генерации и десериализации представлена на рисунке 13.

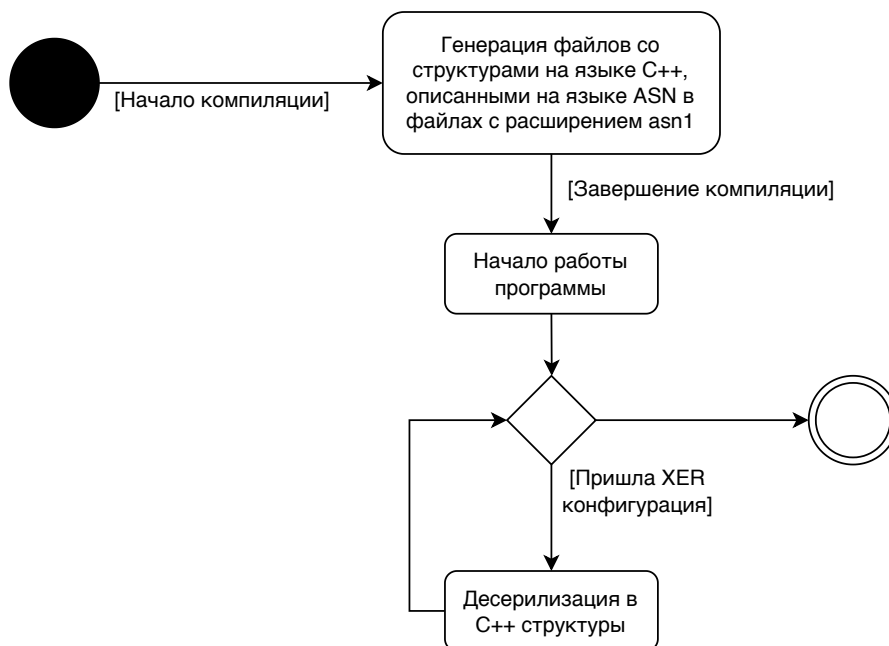


Рис. 13: Процесс генерации структур и десериализации конфигурации

Конфигурация каждого сектора для алгоритма должна включать следующее:

- пороговое значение стандартного отклонения по загруженности сот;
- список сот, участвующих в алгоритме балансировки;

- время ожидания отчёта об измерениях в миллисекундах.

Пример структуры конфигурации Congestion Control на ASN.1 приведён ниже. Структура LoadBalancingCongestionControl задаёт общую конфигурацию алгоритма, а LoadBalancingCongestionControlSectorConfig содержит настройки для каждого сектора.

Листинг 1: Конфигурация для Congestion control.

```
LoadBalancingCongestionControlSectorConfig ::= SEQUENCE
{
    cellIdList                SEQUENCE(SIZE(2..)) OF INTEGER(0..),
    sectorOverloadThreshold    REAL,
    expireTimeMeastReportInMs  INTEGER(0..)
}

LoadBalancingCongestionControl ::= SEQUENCE
{
    loadBalancingCongestionControlSectorConfig
SEQUENCE (SIZE (0..)) OF LoadBalancingCongestionControlSectorConfig
}
```

Для каждой соты (несущей частоты), участвующей в алгоритме, также задаются параметры для процедуры передачи обслуживания с включением межчастотных измерений, которые активируются во время процедуры разгрузки соты. Параметры подбираются с учётом радиоусловий и характеристик несущей. Структура конфигурации формируется на основе требований стандарта 3GPP TS 36.331 [11].

Листинг 2: Конфигурация для процедуры передачи обслуживания с включением межчастотных измерений.

```
A4Rule ::= SEQUENCE
{
    thresh                ThresholdEUTRA,
    hyst                  Hysteresis,
    timeToTrigger         TimeToTrigger,
    triggerQuan           TriggerQuantity,
    ...
}

LoadBalancingHORule ::= SEQUENCE
{
    carrierFreq           ARFCN-ValueEUTRA,
```

```

    a4Rule      A4Rule,
    ...
}

LoadBalancingH0 ::= SEQUENCE
{
    loadBalancingHORuleList SEQUENCE (SIZE (2..3)) OF LoadBalancingHORule,
    ...
}

```

6. Апробация

Реализованный алгоритм балансировки нагрузки требует тщательного тестирования для проверки его работоспособности, а также совместимости с существующими процедурами в ADM модуле.

6.1. Цель и исследовательские вопросы

Целью данной апробации является оценка производительности реализованного алгоритма балансировки нагрузки, а также анализ корректности его функционирования и влияния на существующие процедуры в ADM модуле. Для достижения поставленной цели были сформулированы следующие исследовательские вопросы:

RQ1 : Обеспечивается ли корректная работа внедрённого алгоритма?

RQ2 : Как реализованные подходы влияют на существующую функциональность в модуле ADM?

RQ3 : Как реализованные подходы повлияют на пропускную способность базовой станции?

6.2. Корректность работы

Для проверки корректности работы реализованного алгоритма балансировки были разработаны модульные тесты для каждого из компонентов, представленных на диаграмме 10. Тесты охватывают как базовые сценарии, проверяющие корректность выполнения ключевых функций алгоритма, так и нагрузочные сценарии, предназначенные для верификации устойчивости алгоритма при высокой пользовательской активности. Модульные тесты реализованы с использованием библиотеки GoogleTest [9].

6.3. Демонстрация работы алгоритма

Целью данного эксперимента является проверка работоспособности алгоритма на базовой станции в лабораторных условиях.

6.3.1. Постановка эксперимента

Для проведения экспериментального исследования использовалась следующая конфигурация аппаратно-программного комплекса базовой станции и пользовательского оборудования:

1. Два блока обработки сигнала основной полосы (Baseband Unit (BBU)), разработанные компанией ООО «НТР».

2. По одному радиоблоку (Remote Radio Unit (RRU)) разработанные компанией ООО «НТР» на каждую BBU.
3. Сервер ItTelo HP DL360 G9, на котором развернута централизованная часть протокольного стека (CU).
4. Три устройства Google Pixel 4.

6.3.2. Конфигурация алгоритма

Для компонента Congestion Control задается следующая конфигурация.

Листинг 3: Конфигурация компонента Congestion control

```
<loadBalancingCongestionControl>
  <loadBalancingCongestionControlSectorConfig>
    <LoadBalancingCongestionControlSectorConfig>
      <cellIdList>
        <Cell-ID>10</Cell-ID>
        <Cell-ID>20</Cell-ID>
      </cellIdList>
      <sectorOverloadThreshold>1</sectorOverloadThreshold>
      <expireTimeMeasReportInMs>150</expireTimeMeasReportInMs>
    </LoadBalancingCongestionControlSectorConfig>
  </loadBalancingCongestionControlSectorConfig>
</loadBalancingCongestionControl>
```

Для включения A4 событий в Measurement Config при передаче *RRCConnectionReconfiguration* пользователю во время разгрузки соты, для каждой соты задается конфигурация, описанная ниже.

Листинг 4: Конфигурация A4 измерений для сот

```
<loadBalancingHO>
  <loadBalancingHORuleList>
    <LoadBalancingHORule>
      <carrierFreq>1301</carrierFreq>
      <a4Rule>
        <thresh>
          <threshold-RSRP>30</threshold-RSRP>
        </thresh>
        <hyst>10</hyst>
        <timeToTrigger>ms320</timeToTrigger>
        <triggerQuan>rsrp</triggerQuan>
```

```

        </a4Rule>
    </LoadBalancingHORule>
</loadBalancingHORuleList>
</loadBalancingHO>

```

6.3.3. Работа алгоритма

Пользователь с идентификатором "01" (UE[01]) подключается к соте с идентификатором "10" (Cell[010]). После подключения к соте он начинает передавать трафик и модуль ADM получает сообщение об активности пользователя с модуля PDCP. Перегрузки на данном этапе нет.

```

log #3286 [13:13:41.986] cell_id[010]:
PDCP_ADM_LTE_UE_UPLANE_STATUS_IND_EVENT {UE[01] drb_num=1
DRBs={DRB[drbId[01] drbActiv=Active]}}
log #3287 [13:13:41.986] cell_id[010]: Processing cell load changed
log #3288 [13:13:41.986] [Sector 0]: Active Non-gbr UE load variance 0.707
(Threshold: 1.000) -> Not overloaded

```

Далее пользователь с идентификатором "02" (UE[02]) подключается к соте с идентификатором "10" (Cell[010]). После подключения к соте он начинает передавать трафик и модуль ADM получает сообщение об активности пользователя с модуля PDCP. Поскольку на Cell[010] находится 2 активных пользователя, а на Cell[020] 0 активных пользователей, то стандартное отклонение по активным пользователям превышает пороговое значение и сектор считается перегруженным.

```

log #3897 [13:13:41.986] cell_id[010]:
PDCP_ADM_LTE_UE_UPLANE_STATUS_IND_EVENT {UE[02] drb_num=1
DRBs={DRB[drbId[01] drbActiv=Active]}}
log #3898 [13:13:41.986] cell_id[010]: Processing cell load changed
log #3899 [13:13:41.986] [Sector 0]: Active Non-gbr UE load variance 1.414
(Threshold: 1.000) -> Overloaded

```

Далее Congestion control выбирает самую загруженную соту в секторе, в котором произошла перегрузка, и отправляет ей запрос на процедуру разгрузки.

```

log #3900 [13:14:46.438] [Sector 0] Initialize unload procedure request
log #3901 [13:14:46.438] [Sector 0] Total Active Non-gbr UE: 2
log #3902 [13:14:46.438] [Sector 0] Cell [10] -> Active Non-gbr UE: 2
log #3903 [13:14:46.438] [Sector 0] Cell [20] -> Active Non-gbr UE: 0
log #3904 [13:14:46.438] Processing unload procedure
Target unload cell: cell_id[010]

```

Cell[10] после получения запроса на процедуру разгрузки, выбирает пользователя для передачи обслуживания на менее загруженную соту, и отправляет ей сообщение *RRCConnectionReconfiguration* для включения межчастотных измерений.

```
log #3905 [13:14:46.438] cell_id[010]: [Sector 0]: Unloading procedure |
cell_id[010]: Current load: 2.000 active non-gbr UE
log #3908 [13:14:46.438] cell_id[010], UE[02]:
ADM_RRC_LTE_UE_RECONFIGURATION_REQ{UE[02], measConfig<Configured>}
log #3909 [13:14:46.438] cell_id[010]: UE[02]:
Sending Measurement Config with enabling A4 events
log #3910 [13:14:46.438] cell_id[010]: [Sector 0]: Unloading procedure |
cell_id[010] was chosen to unload. Candidate for unloading: UE[02]
```

Если пользователь получил *RRCConnectionReconfiguration* и успешно его обработал, то в ответ БС он отправляет *RRCConnectionReconfigurationComplete*. После получения данного сообщения АМД модулем, запускается таймер ожидания отчёта об измерениях по наступлению события А4 от пользователя.

```
log #3964 [13:14:46.502] cell_id[010]: UE[02]:
Received RRCConnectionReconfigurationComplete for the
Stage::WaitingConfirmMeasurementConfig
log #3965 [13:14:46.502] cell_id[010]:
Started a timer with an interval of [2000] milliseconds
for Load Balancing H0 Stage Controller.
```

В случае отправки отчёта об измерениях пользователем, АМД модуль, после его получения, выбирает наименее загруженную соту из списка и запускает процедуру передачи обслуживания на нее. В данном случае это Cell[020].

```
log #3981 [13:14:46.910] cell_id[010]: RRC_ADM_LTE_MEAS_REPORT {UE[02]}
log #3982 [13:14:46.910] cell_id[010], UE[02]:
Processing A4 meas report
log #3989 [13:14:46.910] cell_id[010], UE[02]:
ADM_RRC_LTE_UE_BLIND_INTRA_ENB_HO_REQ{UE[02], target_cell_id[020],
measConfig<Configured>, drxConfig<None>}
log #3990 [13:14:46.910] cell_id[010], UE[02]:
Received measurement report for the Stage::WaitingMeasurementReport
```

Поскольку передача обслуживания выполняется также через процедуру *RRCConnectionReconfiguration*, то пользователь в случае удачного ее выполнения отправляет *RRCConnectionReconfigurationComplete*.

```
log #4106 [13:14:46.983] cell_id[010]:
```

```
RRC_ADM_LTE_HO_COMPLETE {UE[02]}  
log #4107 [13:14:46.983] cell_id[010]: UE[02]:  
Received HandoverComplete for the Stage::HandoverExecution  
log #4108 [13:14:46.983] cell_id[010], UE[02]:  
Load Balancing handover procedure was successful. Time completed: [545 ms.]
```

Далее пользователь с идентификатором "03" (UE[03]) подключается к соте с идентификатором "10" (Cell[010]). После подключения к соте он начинает передавать трафик и модуль ADM получает сообщение об активности пользователя с модуля PDCP. Поскольку на Cell[010] находится 2 активных пользователя, а на Cell[020] 1, то перегрузки не будет.

```
log #5247 [13:15:04.819] cell_id[010]:  
PDCP_ADM_LTE_UE_UPLANE_STATUS_IND_EVENT {UE[03] drb_num=1  
DRBs={DRB[drbId[01] drbActiv=Active]}}  
log #5248 [13:15:04.819] cell_id[010]: Processing cell load changed  
log #5249 [13:15:04.819] [Sector 0]:  
Active Non-gbr UE load variance 0.707 (Threshold: 1.000) -> Not overloaded
```

6.4. Вывод

На основании проведённой апробации получены следующие ответы на поставленные исследовательские вопросы:

RQ1. Разработанные модульные тесты проверяют работу компонентов алгоритма по отдельности, а также проверяют корректность функционирования всего алгоритма, включая его поведение при различных входных условиях. В дополнение к этому было проведено тестирование алгоритма балансировки на базовой станции в лабораторных условиях.

RQ2. Для оценки влияния внедрённого решения на существующую функциональность использовались результаты автоматических (CI) тестов, а также дополнительно разработанные сценарии, имитирующие запуск конкурирующих процедур. Тестирование подтвердило корректную остановку алгоритма в случае приоритетного выполнения других процессов в модуле ADM.

RQ3. Оценка влияния алгоритма на пропускную способность планируется в ходе тестирования в летний период.

Заключение

Цель работы была достигнута, были выполнены следующие задачи.

1. Описаны существующие алгоритмы балансировки нагрузки, представленные как в научной литературе, так и в решениях ведущих производителей телекоммуникационного оборудования.
2. Описан алгоритм балансировки нагрузки, процедура передачи обслуживания абонентских устройств между сотами во время перераспределения нагрузки и оптимизации существующих процедур для контроля перегрузки.
3. Спроектированы и реализованы подходы для балансировки нагрузки среди пользователей с активным трафиком и использующих сервисы, для которых не гарантируется минимальная скорость передачи данных.
4. Выполнено модульное тестирование алгоритма, проверяющее его работоспособность и совместимость с существующими процедурами в ADM модуле, а также функциональное тестирование на базовой станции.

В дальнейшем планируется оптимизация работы алгоритма за счет классификации пользователя по типу используемого трафика, и информации о расположении пользователя на соте.

Список литературы

- [1] 3GPP TS 36.306 V17.5.0 (2023-03): Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio access capabilities : Rep. : TS 36.306 / 3rd Generation Partnership Project (3GPP) ; Executor: 3rd Generation Partnership Project (3GPP) : 2024. — January. — URL: https://www.3gpp.org/ftp/Specs/archive/36_series/36.306/. — Available online: https://www.3gpp.org/ftp/Specs/archive/36_series/36.306/.
- [2] [Coordinating Handover Parameter Optimization and Load Balancing in LTE Self-Optimizing Networks](#) / Andreas Lobinger, Szymon Stefanski, Thomas Jansen, Irina Balan. — 2011. — 05. — P. 1–5.
- [3] Dynamic Mobility Load Balancing for 5G Small Cell Networks Based on Utility Functions / Khaled Addali, Suhil Bani Melhem, Yaser Khamayseh et al. // [IEEE Access](#). — 2019. — 09. — Vol. PP. — P. 1–1.
- [4] Elnashar Ayman, El-Saidny Mohamed, Mahmoud Mohamed. Practical Performance Analyses of Circuit Switched Fallback (CSFB) and Voice over LTE (VoLTE) // [IEEE Transactions on Vehicular Technology](#). — 2016. — 01. — Vol. 66. — P. 1–1.
- [5] [Handover Parameter Optimization in LTE Self-Organizing Networks](#) / Thomas Jansen, Irina Balan, John Turk et al. — 2010. — 09. — P. 1–5.
- [6] [Handover parameter optimization for self-organizing LTE networks](#) / Ili Isa, Mohd Baba, Azita Yusof, Ruhani ab rahman. — 2015. — 04. — P. 1–6.
- [7] Hicham Magri, Abghour Noredine, Ouzzif Mohammed. 4G System: Network Architecture and Performance. — 2015. — 04.
- [8] Huawei Technologies Co., Ltd. — Intra-RAT Mobility Load Balancing (eRAN20.0_02). — Huawei Technologies Co., Ltd., Shenzhen, China, 2024. — May. — Issue 02, 2024-05-31. URL: <https://ru.scribd.com/document/838587925/Intra-RAT-Mobility-Load-Balancing-eRAN20-0-02>.
- [9] Inc. Google. GoogleTest - Google Testing and Mocking Framework. — <https://github.com/google/googletest>. — Accessed: 2022-11-15.
- [10] Jewel Md, Zakariyya Rabiul, Lin Fujiang. On Channel Estimation in LTE-Based Downlink Narrowband Internet of Things Systems // [Electronics](#). — 2021. — 05. — Vol. 10. — P. 1246.
- [11] LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification : Rep. : TS 36.331 version 9.16.0 Release 9 / 3GPP : 2013. — URL: <https://www.3gpp.org/>. — 3GPP TS 36.331 version 9.16.0 Release 9.

- [12] LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Requirements for support of radio resource management : Rep. : TS 36.133 version 9.22.0 Release 9 / 3GPP : 2015. — URL: <https://www.3gpp.org/>. — 3GPP TS 36.133 version 9.22.0 Release 9.
- [13] LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP) : Rep. : TS 36.413 version 15.3.0 Release 15 / 3GPP : 2018. — URL: <https://www.3gpp.org/>. — 3GPP TS 36.413 version 15.3.0 Release 15.
- [14] LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 Application Protocol (X2AP) : Rep. : TS 36.423 version 16.2.0 Release 16 / 3GPP : 2021. — URL: <https://www.3gpp.org/>. — 3GPP TS 36.423 version 16.2.0 Release 16.
- [15] Legg Dr. Steven. Abstract Syntax Notation X (ASN.X). — RFC 4912. — 2007. — . — URL: <https://www.rfc-editor.org/info/rfc4912>.
- [16] [Load Balancing in Downlink LTE Self-Optimizing Networks](#) / Andreas Lobinger, Szymon Stefanski, Thomas Jansen, Irina Balan. — 2010. — 05. — P. 1–5.
- [17] Oh Sangchul, Kim Hong-Soog, Kim Yeongjin. [User mobility impacts to mobility load balancing for self-organizing network over LTE system](#). — 2018. — 02. — P. 1082–1086.
- [18] [On Mobility Load Balancing for LTE Systems](#) / Raymond Kwan, Rob Arnott, Robert Paterson et al. — Vol. 1. — 2010. — 10. — P. 1 – 5.
- [19] Liyanage Madhusanka, Braeken An, Shahabuddin Shahriar, Ranaweera Pasika. Open RAN Security: Challenges and Opportunities. — 2022. — [2212.01510](#).
- [20] Szilagyi Peter, Vincze Zoltan, Vulkán Csaba. [Integrated Mobility Load Balancing and Traffic Steering mechanism in LTE](#). — 2013. — 09. — P. 2148–2153.
- [21] Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges / Michele Polese, Leonardo Bonati, Salvatore d'oro et al. // [IEEE Communications Surveys Tutorials](#). — 2023. — 01. — Vol. PP. — P. 1–1.
- [22] Yamamoto Toshiaki, Komine Toshihiko, Konishi Satoshi. Mobility Load Balancing Scheme based on Cell Reselection // International Conference on Wireless and Mobile Communications. — 2012. — URL: <https://api.semanticscholar.org/CorpusID:11984065>.
- [23] A survey on heterogeneous mobile networks planning in indoor dense areas / Lamia Fattouh, Hesham Salman, Zaki Taha et al. // [Personal and Ubiquitous Computing](#). — 2020. — 08. — Vol. 24.

- [24] A threshold-based multi-traffic load balance mechanism in LTE-A networks / Zhang-peng Huang, Jing Liu, Qiang Shen et al. // [2015 IEEE Wireless Communications and Networking Conference, WCNC 2015](#). — 2015. — 06. — P. 1273–1278.