Cloud-RAN моделирование, основанное на параллельной обработке

**Аннотация**

Аннотация — В данной статье мы рассматриваем реализацию облачной сети радиодоступа (C-RAN) на централизованной многоядерной системе, поддерживающей обработку базового диапазона нескольких распределенных антенн. Мы представляем модель параллельной обработки, основанную как на функциональной декомпозиции, так и на декомпозиции данных функций virtualized Base Band Unit (BBU), чтобы сократить время их выполнения. Мы изучаем две стратегии планирования параллельно выполняемых заданий BBU, где вычислительные ресурсы могут быть распределены либо по пользовательскому оборудованию (UE), либо по блокам кода (CB). Используя данные, полученные при запуске RAN-кода с открытым исходным кодом (а именно, OAI), мы вводим модель пакетной очереди (мультисервисная система M [X] / M /C) для оценки необходимой вычислительной мощности в центре обработки данных при соблюдении жестких требований к задержке в нисходящем и восходящем каналах указания. Предложенная модель подтверждена имитационным моделированием при обработке ста LTE-ячеек в многоядерной системе. Результаты дают ценные рекомендации по определению размера и развертыванию облачных систем.

Термины индекса — Cloud-RAN, виртуализация, NFV, VNF, теория массового обслуживания, параллельная обработка, пакетная модель, система M [X]/M/C, планирование, объединение ресурсов, виртуализированный BBU, многоядерные системы,

**I. ВВЕДЕНИЕ**

Облачность мобильных сетей обещает большую экономию средств, а также лучшее управление радиоресурсами. Кроме того, эта новая концепция позволит сетевым операторам развертывать виртуализированные сети по требованию и предлагать инновационные услуги. Эта цель соответствует общей концепции виртуализации сетевых функций (NFV) [1], которая как раз и состоит в замене сетевых функций, работающих на выделенном и проприетарном оборудовании, открытыми программными приложениями, работающими на общих коммерческих готовых серверах (COTS) на облачных платформах. Таким образом, сетевые операторы могут создавать экземпляры виртуализированных сетевых функций (VNFs) "на лету" в различных сетевых местоположениях в соответствии с требованиями заказчика [2].

Основополагающий принцип виртуализации заключается в размещении сетевых функций на одной или нескольких виртуальных машинах (VMs) или контейнерах. VNFs развертываются поверх виртуализированной инфраструктуры, которая может охватывать более одного физического местоположения и даже инфраструктуру облачных вычислений. В идеале VNFs должны располагаться там, где они наиболее эффективны с точки зрения производительности и стоимости. VNFs могут размещаться в центрах обработки данных, сетевых узлах или даже на устройствах конечных пользователей в зависимости от требуемой производительности (в частности, задержки) и ресурсов (пропускная способность, хранение и вычисления).

Однако облачность и коммерциализация сетевых функций порождают новые проблемы, особенно при виртуализации сетей беспроводного доступа. Это, в частности, относится к Cloud-RAN (C-RAN), целью которого является реализация всей обработки радиосигналов в основной полосе частот в программном обеспечении. В настоящее время изучается несколько функциональных разделений; наиболее амбициозным из них, безусловно, является полная централизация функций базового диапазона. Эта архитектура основана на распределенных антеннах и блоках базовой полосы (BBU), сгруппированных в центральном офисе (CO). BBU включает в себя критически важные функции нижнего уровня, такие как кодирование и декодирование каналов, модуляция и демодуляция, а также планирование радиосвязи, управление радиоканалом и конвергенция данных.

Полностью централизованная архитектура позволяет обеспечить радиосвязь между несколькими узлами и управление помехами, что обеспечивает как лучшую эффективность использования спектра частот, так и удобство для пользователей. [3], [4]. Однако полная централизация должна соответствовать строгим требованиям к задержке, определенным стандартами Long Term Evolution (LTE) [5]. Фактически, обработка основной полосы частот должна быть завершена в течение 1 миллисекунды в направлении нисходящей линии связи и 2 миллисекунд в направлении восходящей линии связи. Как следствие, высокопроизводительные процессоры и их эффективное использование являются обязательными

Многие усилия уже были направлены на внедрение традиционной сети радиодоступа (RAN) с помощью виртуализированных функций. Некоторые исследования показывают, что функции, принадлежащие физическому уровню, особенно функция кодирования канала, потребляют наибольшее количество времени обработки и вычислительных ресурсов; см., например, [6]. Таким образом, помимо требования к высокопроизводительным процессорам, можно рассматривать методы параллельного программирования для обеспечения гибкости, обещанной NFV [7], [8]. Более того, одна из основных проблем производительности C-RAN связана с недетерминированным поведением функции кодирования канала, т.е. изменчивостью времени выполнения, которое требуется процессами кодирования и декодирования. Большая часть изменчивости обусловлена условиями радиоканала каждого пользовательского оборудования (UE), подключенного к eNodeB (eNB), требуемой скоростью передачи данных на UE, а также объемом трафика в ячейке.

Приведенные выше наблюдения поднимают фундаментальные вопросы, касающиеся определения требуемой вычислительной мощности при недетерминированных условиях. Чтобы решить эту проблему, мы применяем в данной статье подход вероятностного моделирования. Мы рассматриваем ряд BBU, размещенных в многоядерной системе. Мы моделируем, как вызываются различные задачи обработки BBU, и используем данные, полученные при запуске RAN-кода с открытым исходным кодом Open Air Interface (OAI), для настройки параметров нашей модели. Оказывается, что система массового обслуживания M[X]/M/C [2] может быть использована для определения требуемой вычислительной мощности (т.е. количества процессорных блоков) при распараллеливании функций BBU и для установления правил определения размеров для реализации C-RAN.

Этот документ организован следующим образом: Соответствующая работа и наиболее популярные сценарии C-RAN представлены в разделе II. Подробный анализ параллельно выполняемых виртуальных функций RAN представлен в разделе III. В разделе IV мы вводим допущения моделирования и формулировку системы массового обслуживания. Теоретический анализ модели C-RAN представлен в разделе V. Некоторые численные эксперименты приведены в разделе VI, где мы также проверяем модель, эмулируя реальную систему C-RAN. Наконец, в разделе VII мы представляем основные выводы этого исследования.

**II. СОПУТСТВУЮЩАЯ РАБОТА**

C-RAN, также называемый централизованным RAN, был впервые представлен China Mobile Research Institute в 2010 году [9]. С тех пор в литературе появились различные исследования и платформы для проведения испытаний. Безусловно, главной проблемой этого многообещающего подхода, основанного на программном обеспечении, является требуемое поведение виртуальных функций RAN в реальном времени. Эта проблема была в значительной степени изучена и проанализирована промышленностью [4], [10] и сетевыми операторами [3], [7], [11], а также академические исследователи, в частности, благодаря разработке нескольких решений с открытым исходным кодом или даже проприетарных решений, таких как OAI [12] и Amarisoft [13].

Анализ производительности виртуализированных функций RAN при использовании OAI и нескольких сред виртуализации, таких как виртуальная машина на основе ядра (KVM), Docker и контейнеры Linux (LXC), был представлен в [6]. Результаты показывают, что направление восходящей связи является доминирующей нагрузкой на обработку и требует разделения и/или ускорения. Функции кодирования и декодирования являются наиболее трудоемкими и отличаются высокой вариабельностью. Также обнаружено, что подходы, основанные на контейнерах, обеспечивают несколько лучшую производительность, чем подходы, основанные на гипервизоре.

Первые попытки сократить время выполнения функций RAN были представлены в [14]. Авторы предлагают структуру, которая разбивает набор BBU на группы, которые одновременно обрабатываются на общей однородной вычислительной платформе. Эта работа показывает, что централизованная архитектура потенциально может привести к экономии вычислительных ресурсов по меньшей мере на 22% за счет использования различий в нагрузке на обработку между ENB (BBU). Прирост производительности при выполнении принципов совместного использования ресурсов и статистического мультиплексирования также показан в [15].

Обратите внимание, что системы реального времени и параллельной обработки широко изучались в литературе в течение многих лет. Большая часть усилий была направлена на улучшение стратегий планирования как в однопроцессорных, так и в многопроцессорных моделях, например, в моделях "Самый ранний крайний срок" (EDF) и "Совместное использование процессоров" (PS). Тем не менее, единственный способ повысить производительность C-RAN с точки зрения задержки, избегая при этом накладных расходов, - это разложить тяжелые длинные задачи на параллельно выполняемые небольшие задания [8], [11].

Наиболее популярные варианты использования C-RAN основаны на областях с огромным спросом, таких как городские районы с высокой плотностью населения с макро- и мелкими ячейками, общественные места и т.д. Следующие из них в настоящее время рассматриваются академическими кругами и промышленностью:

• Разделение сети предлагает разумный способ сегментации сети и поддержки индивидуальных сервисов, например, частной мобильной сети. Срезы могут быть развернуты с определенными характеристиками с точки зрения качества обслуживания (QoS), задержки, пропускной способности, безопасности, доступности и т.д. Этот сценарий и строгие требования к производительности C-RAN изучены в [16].

• Сети 5G с несколькими арендаторами работают с различными операторами виртуальных сетей мобильной связи (VNO), различными соглашениями об уровне обслуживания (SLA) и технологиями радиодоступа (RATs). Глобальный сценарий 5G C-RAN приведен в [17]. Авторы предлагают централизованное управление виртуальными радиоресурсами (VRRMs), так называемый “V-RAN enabler” для организации глобальной среды. Объект управления оценивает доступные радиоресурсы на основе скорости передачи данных различных технологий доступа и распределяет их между различными службами в сети с помощью планировщика OAI.

• Сосуществование разнородных функциональных подразделений для поддержки полной или частичной централизации функций сети RAN. Углубленный анализ прироста производительности при выполнении различных функциональных разделений представлен в [10]. Результаты показывают, что производительность снижается, когда функции RAN нижнего уровня сохраняются рядом с антеннами. В этой работе рекомендуется полная централизация, чтобы воспользоваться преимуществами “физической” межсотовой связи для развертывания передовых технологий многоточечного взаимодействия для повышения качества обслуживания (QoE).

• Интеллектуальные сети, которые позволяют автоматизировать развертывание ENBS для предоставления дополнительной пропускной способности в режиме реального времени [17]. Эти интеллектуальные процедуры приносят огромную экономическую выгоду сетевым операторам, которые сегодня массово инвестируют в расширение пропускной способности своей сети.

Другие растущие облачные приложения были кратко описаны в [10]. Она включает в себя массовые приложения Интернета вещей (IoT), широкополосную связь, чувствительную к задержкам (например, виртуальная реальность, видеоповтор на стадионах и т.д.), приложения с низкой задержкой и высокой надежностью, такие как ассистированное вождение и покрытие железных дорог, поскольку C-RAN обеспечивает быструю передачу для ЕЭС, движущихся с высокой скоростью.

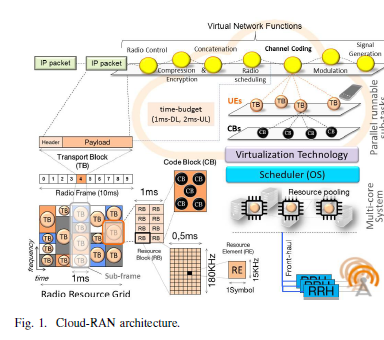
III. C-RAN система

C-RAN нацелен на централизацию обработки радиосигналов в основной полосе частот, поступающих от различных антенн в центральный офис или, в более общем плане, в облаке. Другими словами, C-RAN разъединяет антенны (головки дистанционного управления радио (RRHs)) и блоки обработки сигналов (BBU). C-RAN можно рассматривать как пул BBU, который обрабатывает десятки или даже сотни cотовых станций (eNBs). Станция обычно состоит из 3 секторов, каждый из которых оснащен RRH. RRH имеет два радиочастотных тракта для нисходящих и восходящих радиосигналов, которые передаются по оптоволоконным линиям связи в пул BBU.

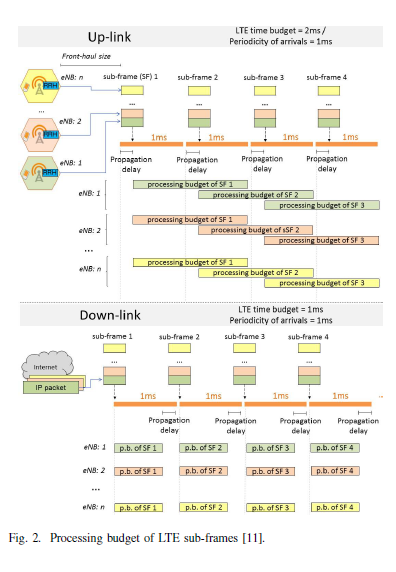
A. Виртуальные функции

VNF от C-RAN - это не что иное, как виртуализированный BBU (vBBU), который программно реализует все сетевые функции, принадлежащие трем нижним уровням стека протоколов LTE. Эти функции в основном касаются таких функций PHY, как генерация сигнала, IFFT /FFT, модуляция и демодуляция, кодирование и декодирование; планирование радиосвязи; объединение /сегментация протокола управления радиоканалом (RLC); и процедуры шифрования/дешифрования протокола конвергенции пакетных данных (PDCP) для нисходящего и восходящего потоков [18]. Задача состоит в том, чтобы выполнять виртуализированные функции BBU достаточно быстро, чтобы увеличить расстояние между RRHs и функциями BBU (а именно, пулом BBU) и, таким образом, повысить уровень концентрации BBU в головном офисе для экономии капитальных и операционных затрат.

Как показано на рисунке 1, vBBU затем может быть создан в верхней части облачной инфраструктуры и смоделирован с помощью графика пересылки подфункций, которые, в свою очередь, могут быть разделены на параллельно выполняемые задачи или задания. Задания BBU могут выполняться на многоядерной платформе в соответствии со стратегией планирования, которая находится в ядре операционной системы (OS) хоста. В предлагаемой архитектуре C-RAN мы используем преимущества производительности, обеспечиваемой контейнерами, которые, в отличие от виртуальных машин, содержат единую ОС [19], т.е. все ядра управляются глобальным планировщиком.



Когда UE требует либо передачи, либо приема данных, вызывается экземпляр vBBU. Как следствие, различные экземпляры виртуализированного BBU выполняются одновременно на вычислительной платформе. В сотовых системах на основе LTE блок данных передачи (а именно, подкадр) может состоять из данных различных UE. Вся обработка подкадра в основной полосе частот должна выполняться в течение 2 миллисекунд и 1 миллисекунды в направлении восходящей линии связи и нисходящей линии связи соответственно. Поскольку субкадр генерируется каждую миллисекунду в обоих направлениях (см. иллюстрацию на рис. 2), обработка в базовом диапазоне всех ячеек, принадлежащих облачной системе, скорее всего, требует высокопроизводительных параллельных вычислений.



При выполнении функций BBU в многоядерной системе могут быть предусмотрены различные стратегии планирования, такие как планирование для каждого подкадра LTE [14] или с более высокой степенью детализации, например, для UE или для блока кода (CB) [7], [8]. Стратегия планирования должна повысить производительность пула BBU с точки зрения задержки. Как отмечено в [7], [8], прирост производительности более важен, когда планировщик имеет дело с короткими заданиями (например, обработка CBS) вместо того, чтобы выделять вычислительные ресурсы для тяжелых задач (например, обработка подкадров) [11].

B. Параллельная обработка

Общая философия параллельных вычислений состоит в разделении больших задач на более мелкие подзадачи с возможностью параллельного выполнения, которые могут выполняться в многоядерных системах. Параллельное выполнение подзадач позволяет сократить время выполнения всей задачи. В облачной системе обработка функции кодирования канала является наиболее ресурсоемкой и, кроме того, имеет недетерминированное поведение [6]. Далее мы сосредоточимся на параллельной обработке функции кодирования канала, которая может выполняться либо для каждого UEs (т.е. для транспортных блоков (TBs)), либо для CBs параллельно.

Чтобы быть более конкретным, мы используем тот факт, что в LTE, когда размер ТB слишком велик, он сегментируется на более короткие блоки данных, называемые CBs, перед обработкой подфункцией кодирования / декодирования. CB представляет собой наименьший блок обработки, который может выполняться параллельно.

Стоит отметить, что передача радиосигнала между UE и eNB генерирует TB каждую миллисекунду. Время обработки TB зависит от условий радиоканала, загрузки данных на UE и объема трафика в ячейке. Планировщик радиосвязи выделяет количество блоков физических ресурсов (RBS) для каждого UE в зависимости от объема трафика в соте и определяет схему модуляции и кодирования (MCS) на основе качества радиоканала. Как количество блоков ресурсов (NRB), так и MCS определяют размер транспортного блока (TBS), т.е. полезные данные [11].

На рисунке 1 показаны облачные блоки данных (т.е. TBs, CBs, RBs и элементы ресурсов (REs)) и предлагаемое параллельное выполнение [7], [8]. В настоящей работе мы рассматриваем RBS с обычным циклическим префиксом (CP) (наиболее распространенным в сетях LTE) [20]. Таким образом, RB состоит из 12 поднесущих в частотной области и 7 символов во временной области. Следовательно, наименьшей определенной единицей является RE, которая состоит из одной поднесущей (15 кГц) и одного символа.

Мы используем параллельную обработку в строгом смысле этого слова, чтобы задания выполнялись одновременно на отдельных ядрах, таким образом избегая процессоров с разделением времени. При параллельных вычислениях каждое задание выполняется на одном ядре и только на одном в любой момент [21]. И наоборот, параллельные вычисления позволяют одновременное выполнение заданий на одном ядре за счет перекрытия периодов времени; это приводит к моделям PS с общим использованием процессоров [22]. Однако недостаток совместного использования процессора заключается в том, что многозадачность на одном ядре требует переключения контекста и разделения памяти, что может заметно увеличить задержку.

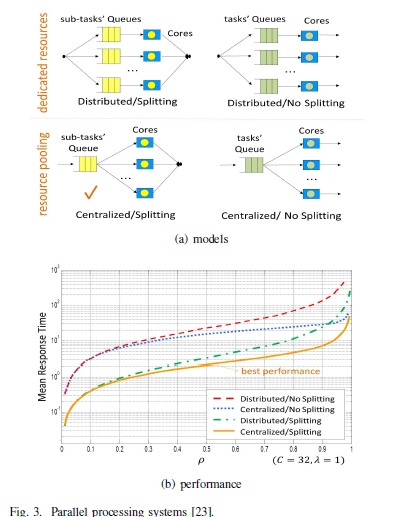
При выполнении параллелизма для каждого UE количество заданий параллельного кодирования определяется количеством UE, запланированных в подкадре LTE. При применении параллелизма для каждого CBs количество заданий параллельного кодирования определяется произведением количества запланированных UE и количества CBs для каждого из них. Количество CBs на UE задается большим целым числом TBS, деленным на размер используемого блока кода (CBS), т.е. Ncb = dT BS / (6144 − 24)e [7], где мы используем тот факт, что LTE определяет минимальный и максимальный размер блока кода, равный 40 и 6144 бита соответственно. Последние 24 бита каждого CB соответствуют циклической проверке избыточности (CRC) [5],[11].

**IV. ПРИНЦИПЫ МОДЕЛИРОВАНИЯ**

A. Моделирование обработки данных

С точки зрения моделирования, каждая антенна (RRH) представляет собой источник заданий в направлении восходящей линии связи, в то время как для направления нисходящей линии связи задания поступают из базовой сети, которая обеспечивает подключение к внешним сетям (например, Интернету или другим сервисным платформам). Для каждого сектора сотовой связи существуют две очереди заданий, по одной в каждом направлении. Поскольку временные затраты на обработку подкадров нисходящей линии связи составляют половину от затрат на обработку подкадров восходящей линии связи, они могут выполняться отдельно на выделенных процессорах. Однако выделение процессоров для каждой очереди не является эффективным способом использования ограниченных ресурсов..

Нельсон и др. в [23] оценивается производительность различных моделей параллельной обработки при рассмотрении “централизованных” (а именно, доступ с одной очередью в многоядерных системах) и “распределенных” архитектур (а именно, доступ с несколькими очередями в многоядерных системах). Также рассматриваются модели параллелизма (так называемое “расщепление”) и непараллелизма (так называемое “отсутствие расщепления”). Результаты показывают, что при любой загрузке системы (а именно ρ) наименьшее (наивысшее) среднее время отклика задания достигается системой “централизованный /разделяющий” (“распределенный / без разделения”), т.е. наилучшая производительность с точки зрения задержки (время отклика) достигается при обработке параллельно выполняемых задач в едином общем пуле ресурсов. Смотрите иллюстрацию на рисунке 3.



Учитывая вышеизложенные наблюдения, мы предлагаем использовать единую систему массового обслуживания с общим пулом процессоров, а именно многоядерную систему с C-ядрами. Глобальный планировщик выделяет вычислительные ресурсы для каждого выполняемого задания кодирования (по нисходящей ссылке) или декодирования (по восходящей ссылке).

Мы предполагаем, что vBBUs (в частности, функции виртуального кодирования / декодирования) вызываются в соответствии с процессом Пуассона, т.е. время между приходами выполняемых функций BBU распределено экспоненциально. Это разумно отражает тот факт, что существует достаточно большое количество антенн, которые не синхронизированы. Возникновение заданий является результатом суперпозиции независимых точечных процессов. Это оправдывает предположение Пуассона. На практике кадры происходят с фиксированными относительными фазами. Таким образом, предположение Пуассона в некотором смысле является предположением наихудшего случая. Поступления на работу не синхронизированы, поскольку RRHs находятся на разных расстояниях от пула BBU. Кроме того, при отсутствии выделенных каналов задержка на входе (время между прибытиями) может сильно варьироваться из-за сетевого трафика.

Параллельное выполнение задач кодирования и декодирования в многоядерной системе с ядрами C может быть смоделировано системами массового прибытия, а именно системой массового обслуживания M[X]/G/C. Далее мы рассматриваем каждое поступление задачи как в действительности поступление B параллельно выполняемых подзадач или заданий, при этом B является случайной величиной. Каждая подзадача требует отдельного этапа обслуживания с общим распределением времени. Время выполнения каждой подзадачи зависит от рабочей нагрузки, а также от сетевой подфункции, которую она реализует. Количество параллельно выполняемых подзадач, принадлежащих сетевой подфункции, является переменным. Таким образом, мы рассматриваем объем нефиксированного размера, который поступает в момент поступления каждого запроса. Время между прибытиями экспоненциально зависит от скорости λ. Размер пакета B не зависит от состояния системы.

В случае Cloud-RAN полный функциональный параллелизм невозможен, поскольку некоторые процедуры базовой полосы (например, IFFT, модуляция и т.д.) требуют последовательного выполнения. Однако параллелизм данных функций BBU (в частности, декодирования и кодирования) обещает значительное повышение производительности. Эти утверждения тщательно изучены в [7], [8]. Результаты показывают, что время выполнения функций BBU может быть значительно сокращено при выполнении параллельной обработки в подкадре, т.е. за счет параллельного выполнения либо UEs, либо даже меньших блоков данных, так называемых CBs. Ниже мы представляем стохастическую модель обслуживания для каждой из схем распараллеливания, чтобы оценить производительность облачной системы. Смотрите иллюстрацию на рисунке 1.

B. Параллелизм со стороны UEs

В LTE несколько UEs пользовательских интерфейсов могут обслуживаться в субкадре продолжительностью 1 миллисекунда. Максимальное и минимальное количество UEs пользовательских интерфейсов, запланированных для каждого подкадра, определяется пропускной способностью eNB. LTE поддерживает масштабируемую полосу пропускания 1:25, 2:5, 5, 10 и 20 МГц. В подкадре каждый запланированный UE получает TB (а именно, группу радиоресурсов в форме RB) либо для передачи, либо для приема. Например, при рассмотрении eNB с частотой 20 МГц доступно 100 RBs. Согласно LTE [5], минимальное количество RBS, выделенных на UE, равно 6. Следовательно, максимальное количество подключенных UE на подкадр задается bmax = 100/6. TBS определяется планировщиком радиосвязи в зависимости от условий индивидуального радиоканала UES, а также от объема трафика в ячейке.

Из предыдущего раздела следует, что параллельная обработка базовой полосы (в частности, канальное кодирование) субкадров LTE может быть смоделирована как система массового обслуживания M[X] /G/C. При рассмотрении распараллеливания для каждого UE количество заданий в пакете соответствует количеству UE, запланированных в подкадре LTE, например, количеству заданий декодирования в миллисекунду в диапазоне eNB 20 МГц от 1 до 16. Затем подкадр содержит переменное количество UE, которое представлено случайной величиной B.

Далее мы предполагаем, что время обработки задания (а именно TB) экспоненциально. Это предположение предназначено для учета случайности во времени обработки UES из-за недетерминированного поведения функции кодирования канала. Например, время выполнения декодирования одного UE может варьироваться от нескольких десятков микросекунд до почти всего временного бюджета, т.е. 2000 микросекунд [12]. На практике это время обслуживания охватывает время отклика каждого компонента системы облачных вычислений, т.е. процессорных блоков, оперативной памяти, внутренних шин, механизма виртуализации, каналов передачи данных и т.д. В дальнейшем мы точно предполагаем, что время обслуживания TB (т.е. задания) распределяется экспоненциально со средним значением 1/mu . Если мы далее предположим, что количество B UEs на подкадр геометрически распределено со средним значением 1 / (1-q) (то есть P (B = k) = (1-q) q ^ (k-1) для k > = 1), полное время обслуживания кадра затем экспоненциально распределяется со средним значением 1/((1-q) mu).

Геометрическое распределение как дискретный аналог экспоненциального распределения отражает изменчивость запланированных UEs в подкадре. Размер B зависит как от количества UEs, требующих обслуживания в соте, так и от условий радиоканала каждого из них. Кроме того, B тесно связан со стратегией планирования радиопередач (например, круговой отбор, пропорциональная ярмарка и т.д.). Количество UEs всегда варьируется от 1 до bmax, где это последнее количество зависит от пропускной способности eNBs. В LTE b\_max достигается, когда пользователи испытывают плохие условия радиосвязи, т.е. при использовании надежной модуляции в виде QPSK и высокой степени избыточности. Для средних условий радиосвязи и ненасыщенных eNBs более вероятно наличие небольших партий UEs. Геометрическое распределение предназначено для отражения сочетания условий радиосвязи в UEs и их потребностей в передаче.

Что касается глобальной облачной архитектуры, общее количество времени t, которое требуется для обработки функций BBU, определяется как t = s + w, где s - время выполнения задания, а w - время ожидания задания при отсутствии свободных процессорных блоков. Задержка перед отправкой между RRHS и BBU-пулом затем фиксируется распределением прибытия. Смотрите иллюстрацию на рисунке 4.

Если предположить, что вычислительная платформа имеет неограниченный буфер, стабильность системы требует:



Далее нас интересует время пребывания подкадров (пакетов) в системе, имея в виду, что если время пребывания превышает некоторый порог (т.е. 1 миллисекунду для кодирования и 2 миллисекунды для декодирования), то подкадр теряется. Если мы измерим систему таким образом, чтобы вероятность того, что время пребывания превысит пороговое значение, была небольшой, мы можем затем аппроксимировать скорость потери субкадра по этой вероятности. Стоит отметить, что в LTE подтверждения передачи и приема обрабатываются для каждого подкадра гибридным процессом автоматического повторного запроса (HARQ). Когда TB теряется, весь подкадр повторно отправляется.

C. Параллелизм со стороны CBs

В LTE, когда ТБ слишком велик, он разбивается на более мелкие блоки данных, называемые CBs. Если мы предположим, что время обработки CB экспоненциально со средним значением 1 /mu, мы снова получим модель M [X] / M / C, где размер партии равен количеству CB в TB. Если это число распределено геометрически, то время обслуживания ТБ экспоненциально, как предполагалось выше. Ключевое отличие теперь заключается в том, что отдельные CBS обрабатываются параллельно ядрами C. Планировщик способен выделить ядро для каждого CB благодаря более атомарной декомпозиции подкадров и TBS.

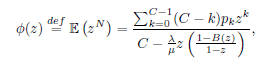
D. Отсутствие параллелизма

Если обработка TBs или CBs не является параллельной, планирование основывается на подкадрах, как представлено в [14]. Все еще предполагая многоядерную систему, где подкадры поступают в соответствии с процессом Пуассона, мы вынуждены рассмотреть систему массового обслуживания M / G / C. Делая экспоненциальные допущения для времени обслуживания CBs и TBS, а также предполагая геометрическое число CBS на TB, мы получаем очередь M / M = C, которая хорошо известна в литературе по организации массового обслуживания [22].

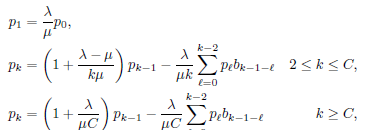
**V. ПАКЕТНАЯ МОДЕЛЬ**

Из анализа, проведенного в предыдущем разделе, модель M[X] /M/C может быть разумно использована для оценки времени обработки подкадра в облачной архитектуре, основанной на многоядерной платформе. В то время как время пребывания произвольного задания в партии было проанализировано в [24], время пребывания всей партии, по-видимому, получило меньше внимания в технической литературе. В этом разделе мы выводим преобразование Лапласа для этой последней величины; это в конечном итоге позволяет нам получить асимптотическую оценку вероятности превышения большого порога.

Давайте рассмотрим очередь M [X] / M / C с пакетами размера B, поступающими в соответствии с пуассоновским процессом со скоростью lam. Время обслуживания задания в пакете экспоненциально со средним значением 1/mu . Мы предполагаем, что условие устойчивости (1) выполняется так, что существует стационарный режим. Количество N заданий в системе в стационарном режиме таково, что [24]



где p\_k = P(N = k) и B(z) - производящая функция размера партии B, т.е. B(z) = sum\_(k=0)^inf( P(B = k))z^k. Как объяснено в [24], вероятности p\_k для k>= 1 удовлетворяют уравнениям баланса:



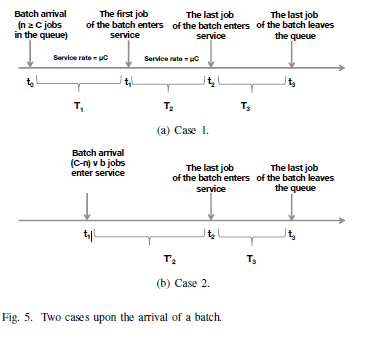
где b\_l - вероятность того, что размер партии равен l. Мы видим, в частности, что вероятности pk для k = 2; ...; C линейно зависят от p\_0, которые в конечном итоге могут быть вычислены с использованием условия нормализации

sum\_(k=0)^(C-1)(C-k)p\_k = C(1-p).

Мы рассматриваем пакет размера b, поступающий в момент времени t\_0 и обнаруживающий n заданий в очереди. Мы рассмотрим два случая (см. рис. 5):

Случай n >=C: В этом случае первое задание из помеченного пакета должно подождать, прежде чем оно поступит в работу.

Случай n < C: В этом случае b /\(C - n)= min(b;C - n) задания помеченного пакета немедленно поступают в эксплуатацию; 0 \/ (b+n - C) = max(0; b+n- C) задания должны подождать, прежде чем приступить к обслуживанию



*A. Анализ первого случая*

В случае n >= C помеченный пакет должен будет подождать определенное время, прежде чем первое задание поступит на обслуживание. Пусть t\_1 обозначает время, в которое первое задание из помеченного пакета начинает свое обслуживание. Очевидно, что мы имеем, что T\_1 = t\_1 – t\_0 равно сумме n-C +1 независимых случайных величин, экспоненциально распределённых со средним значением 1/(мю\* C). Преобразование Лапласа T\_1 определяется для R(s)>=0 с помощью



где E\_b - условное ожидание размера партии b.

Пусть t\_2 обозначает время, в которое последнее задание из пакета поступает на обслуживание. Разница T\_2 = t\_2 – t\_1, очевидно, является суммой b - 1 независимых экспоненциальных случайных величин со средним значением 1 / (мю\*C) (величина мю\*C является скоростью обслуживания системы); преобразование Лапласа этой разницы равно



Чтобы полностью определить время пребывания помеченного пакета, необходимо знать количество y\_b заданий, которые принадлежат этому пакету и которые находятся в очереди, когда пакет загружен. последнее задание пакета начинает свое обслуживание. Пусть t1 = T1 < T2 < … T\_b = t\_2 обозначает время завершения обслуживания заданий (не обязательно относящихся к помеченному пакету) в интервале [t1; t2]. (Обратите внимание, что точка t1, соответствующая времени поступления на обслуживание первого задания из помеченной партии, сама по себе является временем завершения обслуживания одного клиента, присутствующего в очереди при поступлении помеченной партии.) По определению таф\_n – это время, в которое n-е задание помеченного пакета поступает в эксплуатацию.

Обозначим через y\_n количество заданий, принадлежащих помеченному пакету в момент времени таф\_n^+. Тогда последовательность (y\_n) представляет собой цепочку Маркова, для которой условные вероятности перехода могут быть выражены в терминах чисел Стирлинга второго рода S(n; k) [25] определяется для 0<= k<= n с помощью



Числа Стирлинга таковы, что S (n; n) = 1 для n >= 0, S (n; 1) = 1 и S (n; 0) = 0 для n >=1 и удовлетворяют рекурсии для n>= 0 и k >=1



Чтобы сформулировать результаты, мы альтернативно используем многочлены A\_(n,k)(x) определяется с помощью чисел Стирлинга следующим образом:

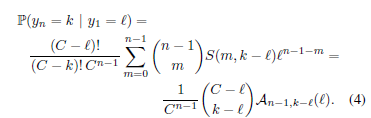


Многочлены A(n,p)(x) удовлетворяют рекурсии для n,p>= 0

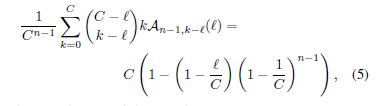


Мы точно имеем следующий результат.

Лемма 1: Условные вероятности перехода цепи Маркова (y\_n) задаются для k >= эль(l) с помощью



Из приведенной выше леммы мы выводим тождество



где мы использовали тот факт, что



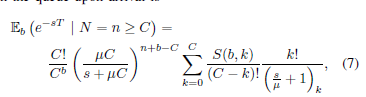
С учетом приведенных выше результатов, когда b-е задание помеченного пакета поступает на обслуживание, в очереди есть y\_b заданий этого пакета. Время T\_3 для выполнения этих заданий равно



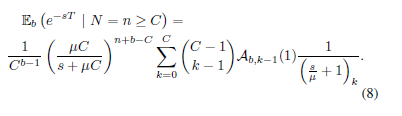
где E(k \*мю) для k = 1,…,y\_b - независимые случайные величины со средним значением 1/(k\*мю ). Преобразование Лапласа для T\_3, зная, что y\_b равно



где (x)k - символ Почхаммера (он же возрастающий факториал) определяется через (x)\_k = x(x+1) : : : (x+k-1). Используя лемму 1,следует, что преобразование Лапласа времени пребывания T партии размером b в системе, когда в очереди по прибытии находится n>=C клиентов, равно



который может быть переписан с помощью многочленов A\_(n,p)(x), определенных уравнением (3) как

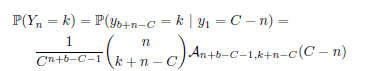


*B. Анализ второго случая*

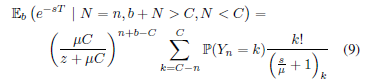
Когда количество n заданий в очереди становится меньше C при поступлении помеченной партии размером b, тогда клиенты b /\(C-n) немедленно начинают свое обслуживание. Давайте сначала предположим, что b +n > C. Принимая поступление помеченной партии в качестве источника времени, последнее задание помеченной партии поступает на обслуживание в случайное время T’\_2 c преобразованием Лапласа



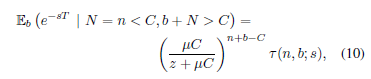
Количество заданий помеченного пакета, присутствующих в системе на момент поступления последнего задания на обслуживание, равно Yn таким образом, что



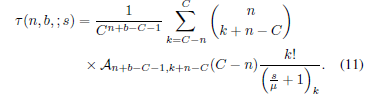
используя уравнение (4). Для заданного значения Yn = k время T\_3, необходимое для обслуживания всех заданий помеченного пакета, имеет преобразование Лапласа, заданное уравнением (6). Используя лемму 1, мы приходим к выводу, что в предположении n < C и b + n > C время пребывания T помеченной партии имеет преобразование Лапласа



и, следовательно,



где



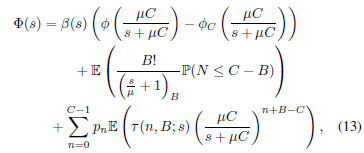
Когда b + n<= C, все задания помеченного пакета поступают на обслуживание сразу после прибытия и преобразования Лапласа времени пребывания является



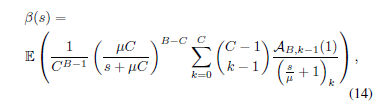
*C. Основной результат*

Используя результаты предыдущих разделов, мы определяем преобразование Лапласа Ф(s) = E(e^(-sT) ) времени пребывания пакета в очереди M^[X]/M/C.

Теорема 1: Преобразование Ф(s) Лапласа задается формулой



Где

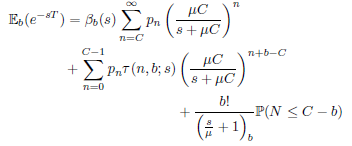


функция Ф\_C (z) задается формулой



и таф(n; b; s), определяемые уравнением (11).

Доказательство: Определяя размер партии b, мы получаем из двух предыдущих разделов



c



и таф(n; b; s) определяется уравнением (11). Обратите внимание, что мы используем тот факт, что таф(n; b; s) = 0, если b < C-n в приведенном выше уравнении. Путем декондиционирования в зависимости от размера партии получается уравнение (13).

Следуя [24], определим z1 как корень с наименьшим модулем в уравнении



корень z1 действителен и больше 1. Отрицательное действительное число



является особенностью с наименьшим модулем преобразования Ф(s) Лапласа, если s1>-мю ( а именно, z\_1 < C/(C-1)).

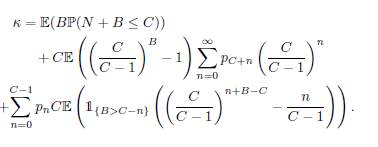
Следствие 1: Если s1 > -мю , то когда t стремится к бесконечности



Где  Если s\_1 < -мю тогда хвост распределения T таков, что когда t стремится к бесконечности

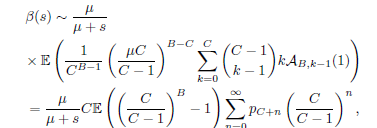


Где



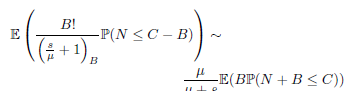
Доказательство: Когда s1 > -мю , корень с наименьшим модулем преобразования Лапласа Ф(s) равен s1 , а оценка (15) непосредственно за этим следует использование стандартных результатов для преобразований Лапласа [26].

Когда s1 < -мю , корень с наименьшим модулем равен -мю . У нас есть для s - это окрестности -мю

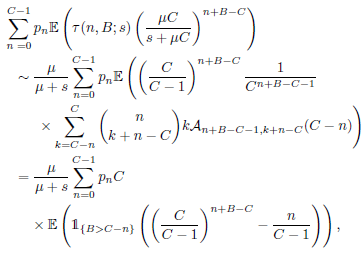


где мы использовали уравнение (5) для эль(l) = 1.

Кроме того, при тех же условиях,



и



где мы использовали уравнение (5) для эль(l) = C - n. Объединение приведенных выше вычислений остатков приводит к уравнению (16).

Следствие 1 гласит, что когда пропускная способность системы достаточно велика, в конце времени пребывания пакета преобладает время обслуживания одного задания. Также стоит отметить, что вопреки тому, что указано в [24], тот же результат справедлив для скорости затухания времени пребывания задания в системе. Наконец, при большом C и при умеренных значениях загрузки и среднего размера партии, k ~E(BP(N + B<=C))~E(B). Это означает, что существует примерно мультипликативный коэффициент E(B) между конечным временем пребывания пакета и временем выполнения задания.

Когда размер партии геометрически распределен со средним значением

1/(1 - q), мы имеем s1 = -(1 - q)\*мю\* C(1 - ро ) и



У нас есть  тогда и только тогда, когда 

**VI. ЧИСЛЕННЫЕ ЭКСПЕРИМЕНТЫ**

В этом разделе мы оцениваем с помощью моделирования поведение облачной системы, в которой размещена обработка базового диапазона ста базовых станций. Цель состоит в том, чтобы проверить актуальность модели M^ [X] / M/C для целей определения размеров и вывести правила определения размеров. Определение размера C-RAN относится к определению минимального количества серверов (ядер), которые требуются для обеспечения обработки подкадров LTE в установленные сроки для заданного количества базовых станций (eENBs), а также максимального расстояния между антеннами и пулом BBU.

В этом случае крайние сроки применяются ко всему подфрейму.

Например, когда время выполнения обработки подкадра в базовой полосе в направлении восходящей линии связи превышает 2 миллисекунды, весь подкадр теряется и, следовательно, передается повторно. Чтобы по-новому взглянуть на эффективность радиоканала, мы также оцениваем потери отдельных пользователей, чтобы управляемые системы могли содержать меньше избыточных данных. Потеря подкадров, а также UEs фиксируются в модели M^[X]/M/C из-за нетерпеливости партий и клиентов, соответственно.

*A. Настройки моделирования*

Мы оцениваем систему C-RAN, в которой размещено 100 eNBs, где каждый из них имеет полосу пропускания 20 МГц. Все eNBs имеют одну антенну (т.е. работают с одним входом и одним выходом (Конфигурация SISO)) и использовать дуплекс с частотным разделением Режим передачи (FDD). Антенны (eNBs) распределены вокруг вычислительного центра в радиусе 100 км. Далее мы сосредоточим наш анализ на функциях декодирования и кодирования, выполняемых во время обработки по восходящей и нисходящей линиям связи, соответственно, из-за их недетерминированного поведение, а также потому, что они являются наибольшим потребителем вычислительных ресурсов из всех функций BBU [8], [12]. Чтобы оценить время выполнения функций декодирования и кодирования, мы используем Код OAIs, который реализует функции RAN в программном обеспечении с открытым исходным кодом [12].

*B. Анализ модели*

Чтобы представить поведение облачной системы с использованием модели M^[X]/M/C, мы вводим в систему массового обслуживания статистические параметры, полученные из эмуляции C-RAN в течение часа загруженности; см. рис. 6. Мы фиксируем поведение функции декодирования в многоядерной системе, выполняющей параллелизм с помощью UEs. Полученные параметры являются следующими:

* Среднее время обслуживания заданий декодирования, E[S], равно 281 микросекунде. Каждое задание декодирования соответствует данным одного UE.
* Среднее количество заданий декодирования, требующих обслуживания одновременно, т.е. средний размер пакета, задается через E[B] = 5. Как поясняется в разделе III, количество UEs, запланированные для каждого подкадра, могут варьироваться от 1 до 16 для eNB 20 МГц. Это может быть аппроксимировано геометрическим распределением с параметром q = 0:8 (q = 1- 1/(E[B])). Размеры пакетов находятся в интервале [1; 16] с вероятностью, равной 0:97. На рисунке 6 приведен процентный показатель каждого из типов подкадров в системе.
* Среднее время между поступлением пакетов составляет 10 микросекунд. Каждый eNB генерирует основную часть заданий декодирования (подкадр) каждую миллисекунду. Среднее время между поступлениями вычисляется путем деления периодичности подкадров на количество eNBs.
* Временной бюджет (крайний срок) для обработки восходящей линии связи задается равным дельта = 2000 микросекундам.

Затем мы можем оценить модель M^[X]/M/C со следующими параметрами: мю= 1/281 и ламда = 1/10. Согласно уравнению (1) для C = 150 нагрузка равна ро = 0,9367. Файлы CDFs, отражающие время пребывания заданий и пакетов, показаны на рисунке 7(а). С помощью Следствие 1, мы проверяем, что если D - это время пребывания задания в очереди M^[X]/M/C, то P(T > t)/P(D > t) стремится к постоянному значению при . Также можно проверить, что наклоны кривых -log(P(D > t))/t и -log(P(T > t))/t для больших t равны мю.

На практике прерывание выполнения подкадров, которые превышают установленные сроки, крайне желательно для экономии вычислительных ресурсов. Затем нас интересует поведение M^[X]/M/C при отказе как от клиентов, так и от партий. Задание (клиент) покидает систему (даже во время обслуживания) когда время его пребывания достигнет заданного крайнего срока. В случае повторного использования партий время пребывания партии рассчитывается с момента поступления до момента подачи последнего задания, составляющего партию. Результаты работы с нетерпеливыми клиентами и партиями представлены на рисунке 7(а).

При нетерпении частота потерь заданий и пакетов составляет соответственно 0,0013 и 0,0065. Мы наблюдаем, что разрыв между двумя показателями (т.е. 0,0065/0,0013) близок к среднему размеру партии E[B]. Это верно, когда уровень потерь составляет, по крайней мере, порядка 10^(-3).

Из-за сложности теоретического анализа моделей, основанных на нетерпении, мы решили использовать производительность

системы M^[X] /M/C, не изменяя размер C-Ran инфраструктуры. Поскольку эта модель стохастически доминирует в системе с отказом, мы получаем консервативные оценки. Как показано на рисунке 7(b), мы проверяем как для заданий, так и для пакетов, что вероятность превышения крайнего срока всегда выше в системе без пересмотра, и, более того, эти две вероятности близки друг к другу при увеличении C.

*C. Определение размеров C-RAN*

Конечная цель определения размера C-Ran- определить объем вычислительных ресурсов, необходимых в облаке (или центре обработки данных), чтобы гарантировать обработку в базовой полосе заданного количества eNBs в установленные сроки. Для этой цели мы оцениваем модель M^[X]/ M / C (без изменения), увеличивая C, пока не будет достигнута приемлемая вероятность превышения крайнего срока (скажем, ипсилон). Требуемое количество ядер является тогда первым значением, которое достигает P(T >дельта ) < ипсилон.

Мы подтверждаем путем моделирования эффективность модели M^[X] /M/C поведением реальной системы C-RAN во время процесса приема (восходящей линии связи) подкадров LTE. Иллюстрацию смотрите на рисунке 8. Результаты показывают, что для заданного ипсилон = 0,00615 требуемое количество ядер равно C\_r = 151, что соответствует реальным показателям C-RAN, где вероятность превышения крайнего срока составляет всего 0,00018.

Когда C принимает значения ниже определенного порогового значения C\_s, система, работающая на C-RAN, перегружена, т.е. количество ядер недостаточно для обработки рабочей нагрузки vBBUs; в этом случае система нестабильна. Пороговое значение C\_s может быть легко получено из

Уравнение (1); для ро= 1, C\_s =[lamda \* E[B]/mu]= 141 ядро.

*D. Анализ эффективности*

Сейчас нас интересует эффективность работы всей C-RAN системы, работающай в центре обработки данных, оснащенном 151 ядром. Система обрабатывает как восходящие, так и нисходящие подкадры, принадлежащие 100 eNBs. Результаты показывают значительный выигрыш при выполнении параллелизма на CBs как при приеме (см. рис. 9(а)), так и при передаче (см. рис. 9(б)). обработка.

Наблюдается, что более 99% подкадров обрабатываются в течение 472 микросекунд и 1490 микросекунд при выполнении параллелизма CBs и UEs соответственно. Это представляет собой выигрыш в 1130 микросекунд (CB) и 100 микросекунд (UE) по отношению к исходной системе (непараллельность). Такое увеличение времени пребывания позволяет оператору увеличить максимальное расстояние между антеннами и центральным офисом. Следовательно, принимая во внимание скорость света в оптоволокне, т.е. 2,25\* 10^8 м/с, расстояние может быть увеличено до 250 км при параллельном подключении CBs. На рисунке 10 показан CDF времени пребывания подкадров LTE при выполнении параллельного программирования.

*VII. ЗАКЛЮЧЕНИЕ*

В этой статье мы изучили производительность виртуализированных функций базовой полосы частот при использовании параллельной обработки. Мы конкретно оценили время обработки подкадров LTE в системе C-RAN. Чтобы уменьшить задержку, мы исследовали функциональную декомпозицию функций BBU и декомпозицию данных, которая приводит к пакетному поступлению параллельно выполняемых заданий с недетерминированным временем выполнения. Для оценки требуемой производительности обработки для поддержки системы C-RAN мы внедрили модель массового обслуживания поступлений, а именно Система массового обслуживания M^[X]/M/C, в которой размер пакета соответствует геометрическому распределению. Изменчивость задержки на границе и времени выполнения заданий учитывается распределением прибытия и обслуживания соответственно. Поскольку время выполнения подкадра LTE становится временем пребывания пакета, мы вывели преобразование Лапласа для этой последней величины, а также вероятность превышения определенного порога для соблюдения крайних сроков.

Мы проверили модель с помощью моделирования при выполнении система C-RAN с сотней ENB на частоте 20 МГц в рабочее время. Мы дополнительно проиллюстрировали, что критерий нетерпеливости, отражающий временные бюджеты LTE, не применяется, когда вероятность превышения крайнего срока достаточно мала. Наконец, после определения размеров системы C-RAN мы оценили ее производительность при обработке как восходящих, так и нисходящих подкадров. Результаты показывают значительный выигрыш с точки зрения задержки при выполнении параллельной обработки подкадров LTE и подходе, основанном на пакетной модели к определению размеров систем C-RAN.