Научный журнал "T-Comm: Телекоммуникации и транспорт"

Журнал включен в перечень периодических научных изданий, рекомендуемый ВАК Минобразования России для публикации научных работ, отражающих основное научное содержание кандидатских и докторских диссертаций

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия

Свидетельство о регистрации СМИ:
ПИ № ФС77-55956.
Дата выдачи: 07 ноября 2013 г.

Язык публикации: русский, английский.

Территория распространения:
Российская Федерация, зарубежные страны

Тираж: 1000 экз.
Периодичность выхода: 12 номеров в год
Стоимость одного экземпляра: 1000 руб.

Плата с аспирантов за публикацию рукописи не взимается

Предпечатная подготовка:
ООО "ИД Медиа Паблишер"

Мнения авторов не всегда совпадают с точкой зрения редакции.
За содержание рекламных материалов редакция ответственности не несет

Адрес редакции и издателя
111024, Россия, Москва, ул. Авиамоторная,
д. 8, стр. 1, офис 323
e-mail: t-comm@media-publisher.ru
Тел.: +7 (495) 957-77-43

Адрес типографии
Москва, ул. Складочная, д. 3, корп. 6

Индексация журнала:
Ulrich's Periodicals Directory; RSCI; EBSCO; elibrary.ru (РИНЦ)
Google Scholar; CyberLeninka (Open Schience);
Bielefeld Academic Search Engine (BASE); OCLC WorldCat;
Registry of Open Access Repositories (ROAR)

Journal is registered by Federal Service for monitoring compliance with cultural heritage protection law

ISSN 2072-8743 (Online) ISSN 2072-8735 (Print)

Media Registration Certificate
PI No. FS77-55956. Date of issue: November 7, 2013

Publication language: Russian, English.

Distribution Territory: Russian Federation, foreign countries

Editorial and Publisher Address
111024, Russia, Moscow, Aviamotornaya str. 8, bloc 1, office 323
e-mail: t-comm@media-publisher.ru.
Tel.: +7 (495) 957-77-43

T-Comm
Телекоммуникации и транспорт
ТОМ 20. №6-2026
Дата выхода: 20.06.2026 г.

Журнал включен в перечень периодических научных изданий, рекомендуемый ВАК Минобразования России для публикации научных работ, отражающих основное научное содержание кандидатских и докторских диссертаций

Учредитель
ООО "Издательский дом Медиа Паблишер"

Главный редактор
Тихвинский Валерий Олегович

Издатель
Дымкова Светлана Сергеевна
ds@media-publisher.ru

www.media-publisher.ru

СОДЕРЖАНИЕ

CONTENT

T-Comm

Telecommunications and transport

Volume 20. No. 6-2026

Release date: 20.06.2026

The journal is included in the list of scientific publications, recommended Higher Attestation Commission Russian Ministry of Education for the publication of scientific works, which reflect the basic scientific content of candidate and doctoral theses.

Founder: "Media Publisher", Ltd.

Publisher: Svetlana S. Dymkova
ds@media-publisher.ru

Editor in Chief: Dr. Valery O. Tikhvinskiy

ЖАДНЫЕ АЛГОРИТМЫ ПОЗИЦИОНИРОВАНИЯ БПЛА-РЕТРАНСЛЯТОРОВ В СИСТЕМАХ ТРАНКИНГОВОЙ СВЯЗИ

DOI: 10.36724/2072-8735-2026-20-6-4-11

Иванов Вячеслав Сергеевич,
МИРЭА – Российский технологический университет, Москва, Россия,
ivanov_vs@mirea.ru

Увайсов Сайгид Увайсович,
МИРЭА – Российский технологический университет, Москва, Россия,
uvajsov@mirea.ru

Manuscript received 12 March 2026;
Accepted 15 May 2026

Ключевые слова: *БПЛА, ретранслятор связи, транкинговые системы связи, жадный алгоритм, эвристические алгоритмы*

В настоящей работе исследуется задача оперативного развертывания систем транкинговой связи с использованием беспилотных летательных аппаратов, выполняющих функции воздушных ретрансляторов. В качестве критерия наличия устойчивой связи между ретранслятором и наземными абонентами рассматривается соотношение между допустимым уровнем потерь сигнала и реальными потерями на трассе распространения, рассчитываемыми по эмпирической модели Окумура-Хата, адаптированной для учета рельефа местности и разности высот. Для определения позиций беспилотных летательных аппаратов предлагается использовать жадные эвристические алгоритмы, обеспечивающие приемлемую скорость расчетов в условиях ограниченного времени на принятие решений. Проводится сравнительная оценка трех вариантов жадного алгоритма, различающихся правилами выбора точки размещения на каждом шаге. При помощи этой оценки делается заключение о пригодности каждого подхода с точки зрения баланса между количеством используемых ретрансляторов и площадью нежелательного покрытия вне зоны обслуживания, и приводятся результаты вычислительных экспериментов на сетке со случайным рельефом. На текущем этапе предполагается, что полученные жадными алгоритмами решения будут использованы в качестве начальной популяции для генетического алгоритма, выполняющего более глубокий поиск глобального оптимума. На языке Python разработано программное обеспечение, позволяющее наглядно визуализировать зоны покрытия и анализировать результаты.

Информация об авторах:
Иванов Вячеслав Сергеевич, *МИРЭА – Российский технологический университет, Москва, Россия. ORCID: 0000-0001-9827-1690*
Увайсов Сайгид Увайсович, *д.т.н., профессор, МИРЭА – Российский технологический университет, Москва, Россия.ORCID: 0000-0003-1943-6819*

Для цитирования:
Иванов В.С., Увайсов С.У. Жадные алгоритмы позиционирования БПЛА-ретрансляторов в системах транкинговой связи // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 4-11.
For citation:
V.S. Ivanov, S.U. Uvajsov, "Greedy Algorithms for Positioning UAV Repeater Nodes in Trunking Communication Systems," *T-Comm*, 2026, vol. 20, no. 6, pp. 4-11. (*in Russian*)

Введение

Современный мир невозможно представить без надежной и оперативной связи, которая стала не просто инструментом коммуникации, но и критическим элементом инфраструктуры. Сегодня существует несколько основных видов подвижной связи, каждый из которых занимает свою нишу [1, 2]. Спутниковая связь обеспечивает глобальное покрытие, позволяя работать в самых удаленных уголках планеты, однако её главными недостатками остаются высокая стоимость терминалов и трафика, а также ощутимые задержки сигнала [3, 4]. Привычная обществу мобильная сотовая связь, напротив, предлагает абонентам доступные тарифы и высокую скорость передачи данных, но её работа полностью зависит от наличия наземной инфраструктуры - вышек с базовыми станциями, что делает её уязвимой и бесполезной в зонах отчуждения или при чрезвычайных ситуациях [5, 6].

Особое место занимает транкинговая связь. Изначально разработанная для профессиональных пользователей (силовых структур, служб такси, коммунальных предприятий), она сочетает в себе эффективность и надежность. Её ключевое преимущество - оперативное распределение ограниченного количества радиочастот между большим числом абонентов и мгновенное установление соединения, что критически важно в оперативной обстановке. В отличие от сотовой, транкинговая связь менее требовательна к плотности покрытия и способна работать в условиях высоких нагрузок, а по сравнению со спутниковой – она значительно дешевле в эксплуатации и не имеет ощутимых задержек. Именно транкинговая связь благодаря своему балансу цены, скорости и надежности часто становится оптимальным выбором для обеспечения координации в нестандартных условиях [7, 8].

Однако проектирование и развертывание любых наземных систем связи, будь то сотовая или транкинговая, традиционно упирается в проблему инфраструктуры. Для обеспечения устойчивого радиопокрытия необходимо возведение вышек и антенно-мачтовых сооружений. Этот процесс требует длительных согласований, проведения геодезических изысканий, строительных работ и прокладки коммуникаций, что занимает недели и даже месяцы, не говоря о значительных финансовых затратах [9]. Но существует широкий класс сценариев, когда времени на строительство нет. Природные катаклизмы, техногенные аварии, масштабные поисково-спасательные операции или временные массовые мероприятия требуют оперативной организации связи. В таких ситуациях на помощь приходят мобильные ретрансляторы, поднятые высоко над землей. И наиболее перспективным и гибким решением становится использование беспилотных летательных аппаратов (БПЛА). Размещение транкингового ретранслятора на борту БПЛА позволяет в кратчайшие сроки создать зону уверенного приема на обширной территории, обеспечив жизненно важную связь там, где это невозможно сделать наземными средствами [10]. В данной статье будет рассмотрен алгоритм оптимального размещения такого воздушного ретранслятора для максимизации зоны покрытия и качества сигнала.

1 Эвристические алгоритмы

Задача размещения беспилотного летательного аппарата с ретрансляционным оборудованием на борту является классической оптимизационной проблемой. Необходимо найти такую точку в пространстве, которая обеспечит максимальное покрытие абонентов при соблюдении ряда ограничений: дальность прямой видимости, время полета и энергопотребление БПЛА, а также рельеф местности. Поскольку зачастую речь идет о работе в динамичной или непредсказуемой обстановке, поиск решения требует применения эффективных вычислительных методов.

В теории оптимизации известен целый ряд подходов к решению подобных задач. Наиболее интуитивно понятным является жадный алгоритм, суть которого заключается в принятии локально оптимальных решений на каждом шаге [11, 12]. Однако существуют и более сложные метаэвристические методы, вдохновленные природными процессами [13]. Генетический алгоритм имитирует механизмы естественного отбора: он оперирует множеством потенциальных решений (популяцией), которые скрещиваются и мутируют, оставляя в итоге только наиболее эффективные варианты размещения [14]. Другие методы, такие как пчелиный или муравьиный алгоритмы, также заимствуют поведение живых организмов. Пчелиный алгоритм моделирует поиск нектара пчелами-разведчицами, которые находят перспективные участки и мобилизуют в эти районы основную группу для более детального изучения [15, 16]. Муравьиный алгоритм основан на способности насекомых находить кратчайший путь к источнику пищи, помечая свой маршрут феромонами, что позволяет постепенно усиливать наилучшие траектории [17-19].

Каждый из перечисленных методов обладает своими достоинствами и нюансами, однако для решения задачи оперативного развертывания ретранслятора важна не только точность конечного результата, но и скорость вычислений. В связи с этим в данной работе предлагается комбинированный подход. На первом этапе, когда требуется получить хотя бы какое-то работоспособное решение в кратчайшие сроки, целесообразно применение жадного алгоритма. Более того, учитывая специфику исходных данных, могут быть использованы различные вариации жадной стратегии, отличающиеся правилами выбора стартовой точки или критериями локальной оптимизации. Это позволит сформировать несколько базовых вариантов размещения БПЛА для различных сценариев. Однако жадный подход, при всей своей скорости, не гарантирует глобального оптимума. Поэтому на втором этапе полученные результаты будут использованы в качестве отправной точки для генетического алгоритма, основная задача которого - исследовать область решений более глубоко и найти наилучшие из возможных позиций, обеспечивающих максимальную эффективность ретрансляции. В этой работе будут рассмотрены различные вариации жадного алгоритма, которые в дальнейшем станут отправной точкой.

2 Разработанные жадные алгоритмы

Выбор оптимальной позиции БПЛА невозможен без адекватной модели распространения радиоволн, позволяющей оценить, будет ли связь между ретранслятором и наземными абонентами устойчивой. Реальная зона покрытия зависит от сложного переплетения факторов. Определяющее влияние оказывают характеристики окружающей местности: рельеф, наличие высотных зданий, плотность лесного массива – все это создает препятствия на пути сигнала, вызывая отражения, дифракцию и затухание.

Не менее важны и технические параметры самого оборудования: мощность передатчика, чувствительность приемника,

а также тип и диаграмма направленности антенн, установленных как на борту БПЛА, так и у абонентов. Учесть все эти переменные в полном объеме - чрезвычайно сложная задача, требующая детальных геоданных и громоздких вычислений, что зачастую противоречит требованию оперативности.

В рамках данного исследования предлагается использовать подход, основанный на энергетическом расчете радиолинии. Ключевым критерием наличия связи выступает соотношение между допустимым уровнем потерь при распространении сигнала и реальными потерями, возникающими на трассе.

Зона уверенного приема будет очерчена множеством точек, для которых рассчитанные потери на трассе не превышают это допустимое значение. Для определения потерь в зависимости от расстояния между ретранслятором и абонентом будет использована модель Окумура-Хата, дополнительно учитывающая разность высот над уровнем моря в местах нахождения передатчика и приемника. Таким образом, задача сводится к сравнению вычисленного для каждого возможного радиуса покрытия значения потерь с пороговым уровнем.

Предлагается три варианта жадного алгоритма, с помощью которых на следующем этапе будет выявлено лучшее решение путем скрещивания решений генетическим алгоритмом. Задача заключается в том, чтобы покрыть все клетки прямоугольной сетки N×N (рабочая зона) зонами покрытия БПЛА, минимизируя при этом общее количество БПЛА. Радиус покрытия каждого БПЛА зависит от высоты местности в точке размещения и высот окружающих клеток, потенциальных мест нахождений абонентов с портативной станцией, согласно эмпирической формуле. Входными данными для алгоритма служат матрица высот местности H размером N×N, где каждый элемент H[i][j] соответствует высоте клетки с координатами (i,j), рабочая частота передатчиков f (в МГц) и допустимый уровень потерь сигнала $L_{доп}$ (в дБ), превышение которого делает связь неустойчивой. Алгоритм реализует жадную эвристику: на каждом шаге выбирается наилучшая непокрытая клетка для установки нового БПЛА, после чего все клетки, попавшие в его зону, исключаются из дальнейшего рассмотрения.

2.1 Предварительный расчет максимальных зон покрытия

На первом этапе для каждой клетки (x,y) рабочей зоны, где координаты x и y принимают целые значения от 0 до N−1, необходимо определить максимальный радиус $R_{max}(x,y)$, при котором беспилотный летательный аппарат, размещённый в данной точке, сможет обеспечить устойчивую связь со всеми абонентами, находящимися внутри круга этого радиуса. Решение этой задачи требует последовательного перебора пробных целочисленных значений радиуса d=0,1,2,… вплоть до максимально возможного расстояния между любыми двумя клетками сетки.

$$d_{\max} = \sqrt{(N-1)^2 + (N-1)^2} \tag{1}$$

Для текущего пробного радиуса d рассматриваются все клетки (i,j), расстояние от центра (x,y) до которых не превышает *d*, то есть удовлетворяющие условию:

$$d \geq \sqrt{(i-x)^2 + (j-y)^2} \tag{2}$$

Для каждой такой клетки по эмпирической модели Окумура-Хата, дополнительно учитывающей разницу высот местности и адаптированной для условий прямой видимости с учётом высот передатчика и приёмника, вычисляется требуемый радиус связи Rt [9].

$$\begin{aligned} L_U &= 69.55 + 26.16 \times \log_{10} f - 13.82 \times \log_{10}(h_{bs} \pm \Delta h) - \\ &-(0.8 + (1.1 \times \log_{10} f - 0.7) \times (h_{ps} \pm \Delta h)) + (44.9 - \\ &-6.55 \times \log_{10}(h_{bs} \pm \Delta h)) \times \log_{10} R_t; \end{aligned} \tag{3}$$

где L_U -уровень потерь на трассе распространения радиосигнала (дБ); f- частота передачи сигнала, (МГц); h_{bs}-высота подвеса антенны БС, (м); h_{ps}-высота подвеса антенны ПС, (м); Δh – разница в высотах над уровнем моря в местах нахождения БС и ПС, (м); R_t- расстояние между объектами, (км).

Если для какой-либо из проверяемых клеток окажется, что требуемый радиус меньше фактического расстояния до неё:

$$R_t < \sqrt{(i-x)^2 + (j-y)^2} \tag{4}$$

это означает невозможность установления устойчивой радиосвязи с данным абонентом при текущем пробном радиусе d. В таком случае дальнейшее увеличение радиуса не имеет смысла, и максимально допустимый радиус для позиции (x,y) фиксируется как предыдущее успешное значение:

$$R_{\max}(x,y) = d - 1 \tag{5}$$

Перебор для данной точки прекращается. Если же все клетки внутри круга успешно проходят проверку, радиус *d* увеличивается на единицу, и процедура повторяется для нового, большего круга. В случае, когда все проверки оказываются успешными вплоть до максимального радиуса $d=d_{max}$, это значение и становится максимальным радиусом для данной позиции:

$$R_{\max}(x,y) = d_{\max} \tag{6}$$

Выполнив описанные действия для всех клеток рабочей зоны, мы получаем матрицу radius_map, где для каждой точки (x,y) хранится значение максимального радиуса, гарантирующего связь со всеми абонентами внутри соответствующего круга.

Прежде чем переходить непосредственно к размещению, необходимо подготовить структуры данных, которые будут использоваться в дальнейшем. Вводится битовая матрица покрытия C того же размера N×N. Значение C[i][j]=0 означает, что клетка (i,j) ещё не обслуживается ни одним из установленных БПЛА, а C[i][j]=1 – что она уже покрыта. Изначально все элементы матрицы C устанавливаются в 0 или C=False[N][N]. Также создаётся пустой список D, в который будут последовательно добавляться записи о каждом размещённом БПЛА в формате тройки чисел (x,y, R_{max}) – координаты центра и радиус покрытия. Для количественной оценки нежелательного излучения за пределами рабочей зоны вводится переменная P_{sum}, накапливающая общее количество так называемых «внешних точек» - клеток за границами сетки, попадающих в зоны покрытия установленных ретрансляторов. Начальное значение этой переменной равно нулю.

2.2 Основной цикл размещения

Второй этап для каждого из алгоритмов отличается. Запускается основной цикл, который повторяется до тех пор, пока не будут покрыты все клетки рабочей зоны.

$$\exists(i;j):C[i][j]==0 \quad (7)$$

На каждом шаге сначала ведётся поиск наилучшего кандидата среди ещё не покрытых клеток. Если клетка уже отмечена как покрытая, она пропускается.

Для каждой итерации проводится инициализация переменных. Первая переменная - K, количество непокрытых клеток, которые окажутся внутри круга радиуса R с центром в (x,y). Начальное значение -1 гарантирует, что любой кандидат с K >= 0 будет лучше. Вторая переменная – P, количество целочисленных точек, лежащих за пределами сетки (то есть с координатами меньше 0 или больше N−1), но попадающих в тот же круг. Изначально бесконечность, чтобы первый же кандидат с любым конечным P стал лучшим (при сравнении). Best_x и Best_y – координаты лучшего кандидата, пока не найдены, равны -1.

Далее кандидат сравнивается с текущим лучшим по следующему правилу: если K больше, чем у лучшего ($K[x][y]>K_{max}$), или если K равно, но P меньше ($K[x][y]= K_{max}$ && $P[x][y]<P_{min}$, то этот кандидат запоминается как новый лучший.

$$\begin{aligned} K[x][y] &= K_{\max} \\ P[x][y] &= P_{\min} \\ [x][y] &= best \end{aligned} \quad (8)$$

Таким образом, приоритет отдаётся максимизации пользы внутри зоны, а при равной пользе – минимизации нежелательного покрытия снаружи. Процесс завершается, когда все клетки становятся покрытыми.

$$\forall(i;j):C[i][j]==1 \quad (9)$$

В конце каждой итерации выбирается точка, с наибольшим радиусом. В список D добавляются координаты и радиус БПЛА. Все клетки, находящиеся внутри радиуса, помечаются как покрытые с помощью функции marc_c. К значению переменной P_{sum} добавляется значение P данной точки.

В случае, если на каком-то шаге не найдётся ни одного кандидата с положительным значением K (что может свидетельствовать о недостижимости оставшихся клеток из-за особенностей рельефа), алгоритм также завершает работу, но в условиях задачи предполагается, что полное покрытие достижимо.

Во втором варианте алгоритма подсчитываются те же две величины, однако кандидат сравнивается с текущим лучшим по следующему правилу: если P меньше, чем у лучшего ($P[x][y]<P_{min}$), или если P равно, но K больше ($P[x][y]=P_{min}$ &&$K[x][y]>K_{max}$), то этот кандидат запоминается как новый лучший. Таким образом, приоритет отдаётся минимизации нежелательного покрытия снаружи, а при равных значениях - максимизации внутри зоны.

Третий вариант алгоритма полностью игнорирует точки за пределами рабочей зоны. Здесь главным показателем выступает максимально возможный радиус БПЛА R. Среди непокрытых клеток ищется та, у которой значение R наибольшее. При равенстве радиусов предпочтение отдаётся кандидату с большим K. Поскольку P никак не влияет на выбор, внешняя площадь может оказаться значительной.

2.3 Визуализация расчетов

По окончании работы алгоритмов формируется список координат БПЛА и их радиусов. Для наглядного представления результатов строится визуализация: карта высот отображается цветовой заливкой, на неё наносятся красные точки в местах установки БПЛА и красные пунктирные круги, соответствующие зонам покрытия каждого БПЛА. Такое графическое представление позволяет оценить, насколько равномерно распределены БПЛА и есть ли выход зон покрытия за пределы рабочей зоны. Таким образом, алгоритмы полностью автоматически, без вмешательства пользователя, определяет позиции БПЛА, обеспечивая полное покрытие при минимизации лишней площади и общего количества устройств. Детерминированность вычислений гарантирует, что для одних и тех же входных данных результат будет всегда одинаковым. На рис. 1 представлен первый вариант жадного алгоритма.

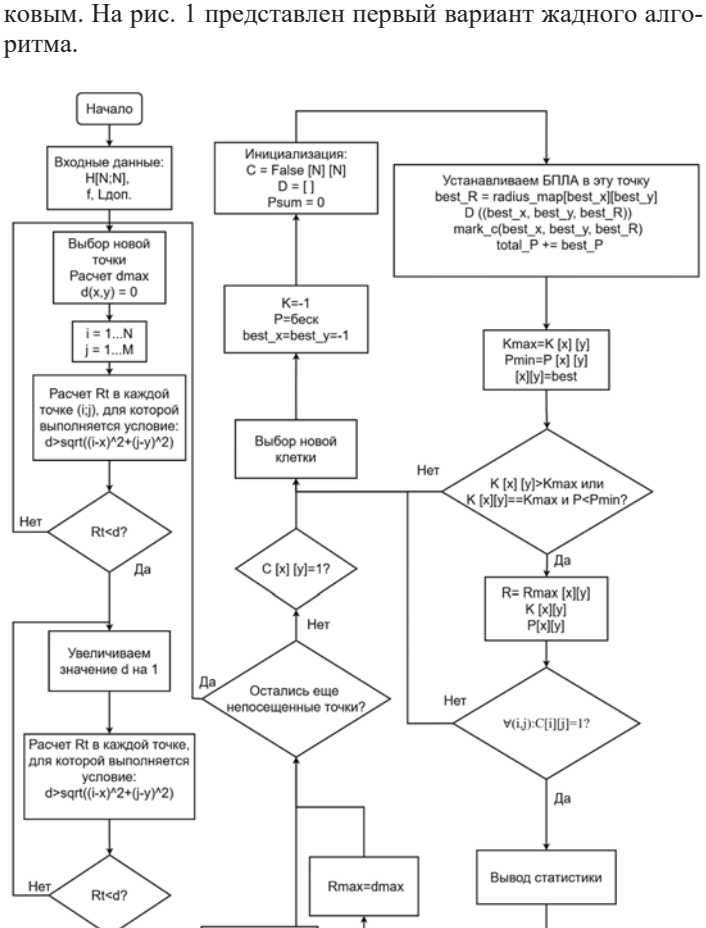


Рис. 1. Жадный алгоритм №1

Для проведения вычислительных экспериментов и визуализации было разработано специализированное программное обеспечение на языке Python. Расчёт радиусов покрытия БПЛА, жадные алгоритмы размещения, а также построение карт высот и зон покрытия реализованы с использованием библиотек NumPy и Matplotlib. Для сравнения эффективности трёх разработанных жадных алгоритмов в программном обеспечении была реализована возможность последовательного запуска каждого из них на одних и тех же входных данных. В качестве входных данных использовалась матрица высот размером 20×20 клеток, сгенерированная случайным образом (начальное значение генератора случайных чисел фиксировано для воспроизводимости результатов). Допустимый уровень потерь на трассе распространения сигнала был принят равным 120 дБ.

В ходе выполнения каждого алгоритма в консоль выводилась пошаговая информация о выбранной позиции БПЛА, его радиусе, количестве вновь покрытых клеток внутри рабочей зоны и числе точек покрытия за её пределами (рис. 2). По окончании работы алгоритма рассчитывались и выводились итоговые метрики:

- общее количество размещённых БПЛА;
- суммарное количество внешних точек покрытия (с учётом возможных пересечений зон);
- уникальная внешняя площадь – количество различных клеток вне рабочей зоны, попавших в зону покрытия хотя бы одного БПЛА;
- доля уникального внешнего покрытия относительно общей покрытой площади.

```
АЛГОРИТМ 1: максимизация inside, затем минимизация outside
Предвычисление радиусов...
Радиусы вычислены.
Шаг 1: дрон в (3, 6) с радиусом 3, покрыто 29 новых клеток, внешних точек 0
Шаг 2: дрон в (2, 16) с радиусом 3, покрыто 28 новых клеток, внешних точек 1
Шаг 3: дрон в (15, 18) с радиусом 3, покрыто 23 новых клеток, внешних точек 6
Шаг 4: дрон в (18, 6) с радиусом 3, покрыто 23 новых клеток, внешних точек 6
Шаг 5: дрон в (6, 4) с радиусом 3, покрыто 21 новых клеток, внешних точек 0
Шаг 6: дрон в (2, 1) с радиусом 3, покрыто 18 новых клеток, внешних точек 7
Шаг 7: дрон в (19, 12) с радиусом 3, покрыто 18 новых клеток, внешних точек 11
Шаг 8: дрон в (18, 1) с радиусом 3, покрыто 16 новых клеток, внешних точек 11
Шаг 9: дрон в (4, 11) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 10: дрон в (7, 9) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 11: дрон в (7, 14) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 12: дрон в (9, 17) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 13: дрон в (10, 7) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 14: дрон в (10, 12) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 15: дрон в (11, 2) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 16: дрон в (13, 5) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 17: дрон в (13, 10) с радиусом 2, покрыто 13 новых клеток, внешних точек 0
Шаг 18: дрон в (12, 15) с радиусом 2, покрыто 12 новых клеток, внешних точек 0
Шаг 19: дрон в (15, 13) с радиусом 2, покрыто 10 новых клеток, внешних точек 0
Шаг 20: дрон в (1, 11) с радиусом 2, покрыто 10 новых клеток, внешних точек 1
Шаг 21: дрон в (6, 18) с радиусом 2, покрыто 9 новых клеток, внешних точек 1
Шаг 22: дрон в (14, 1) с радиусом 2, покрыто 8 новых клеток, внешних точек 1
Шаг 23: дрон в (8, 0) с радиусом 2, покрыто 8 новых клеток, внешних точек 4
Шаг 24: дрон в (15, 9) с радиусом 2, покрыто 7 новых клеток, внешних точек 0
Шаг 25: дрон в (18, 17) с радиусом 2, покрыто 7 новых клеток, внешних точек 1
Шаг 26: дрон в (10, 19) с радиусом 2, покрыто 5 новых клеток, внешних точек 4
Шаг 27: дрон в (5, 13) с радиусом 2, покрыто 4 новых клеток, внешних точек 0
Шаг 28: дрон в (0, 7) с радиусом 2, покрыто 4 новых клеток, внешних точек 4
Шаг 29: дрон в (9, 3) с радиусом 2, покрыто 3 новых клеток, внешних точек 0
Шаг 30: дрон в (3, 19) с радиусом 2, покрыто 3 новых клеток, внешних точек 4
Шаг 31: дрон в (8, 11) с радиусом 2, покрыто 2 новых клеток, внешних точек 0
Шаг 32: дрон в (11, 9) с радиусом 2, покрыто 2 новых клеток, внешних точек 0
Шаг 33: дрон в (15, 3) с радиусом 2, покрыто 2 новых клеток, внешних точек 0
Шаг 34: дрон в (16, 15) с радиусом 1, покрыто 2 новых клеток, внешних точек 0
Шаг 35: дрон в (2, 9) с радиусом 1, покрыто 1 новых клеток, внешних точек 0
Шаг 36: дрон в (14, 7) с радиусом 2, покрыто 1 новых клеток, внешних точек 0
Шаг 37: дрон в (16, 11) с радиусом 2, покрыто 1 новых клеток, внешних точек 0
Шаг 38: дрон в (0, 4) с радиусом 2, покрыто 1 новых клеток, внешних точек 4
Шаг 39: дрон в (0, 13) с радиусом 2, покрыто 1 новых клеток, внешних точек 4
Шаг 40: дрон в (5, 0) с радиусом 2, покрыто 1 новых клеток, внешних точек 4
Шаг 41: дрон в (12, 0) с радиусом 2, покрыто 1 новых клеток, внешних точек 4
Шаг 42: дрон в (0, 19) с радиусом 2, покрыто 1 новых клеток, внешних точек 7
Шаг 43: дрон в (19, 19) с радиусом 2, покрыто 1 новых клеток, внешних точек 7

=== Итоговая статистика (алгоритм 1) ===
Количество дронов: 43
Сумма внешних точек (с повторами): 92
Уникальная внешняя площадь: 91
Уникальная доля внешнего покрытия: 18.53%
```

Рис. 2. Пример выводных данных

Для наглядного сопоставления результатов работы каждого алгоритма производилась визуализация карты высот с наложенными зонами покрытия БПЛА, что позволяло качественно оценить характер размещения. Благодаря детерминированности алгоритмов, повторные запуски на тех же данных давали идентичные результаты, что обеспечивало корректность сравнения (рис. 3).

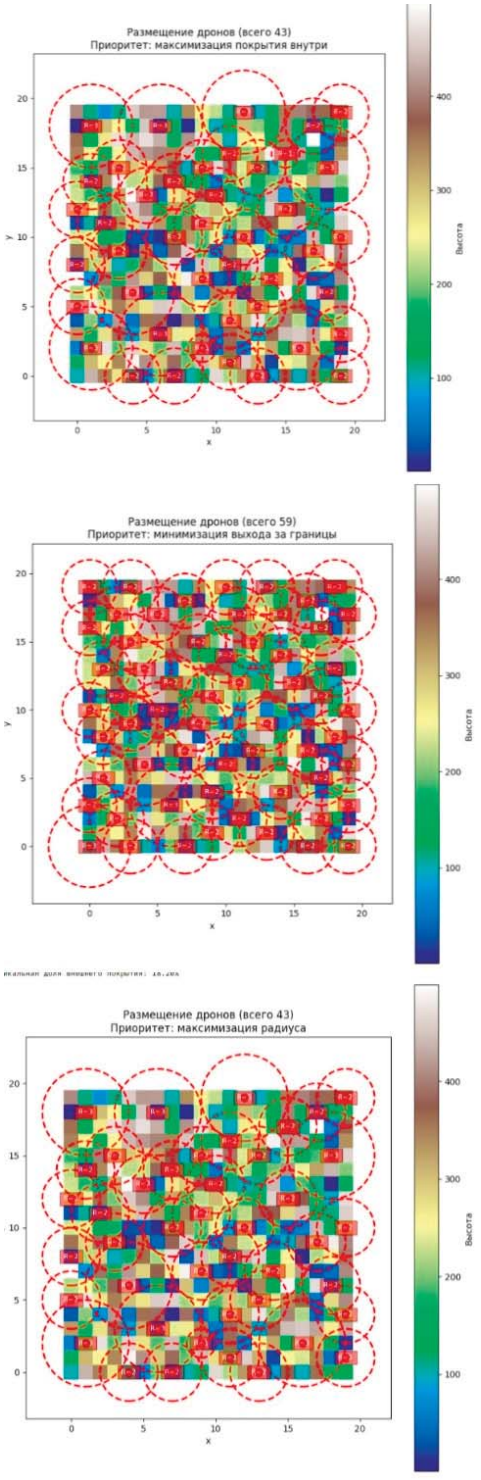


Рис. 3. Визуализация расчетов

Проведённое сравнение позволило выявить особенности каждого подхода:

• алгоритм, максимизирующий покрытие внутри зоны, показал количество БПЛА равное 43;

• алгоритм с приоритетом минимизации выхода за границы привёл к увеличению числа БПЛА, но не всегда к уменьшению уникальной внешней площади;

• алгоритм, ориентированный на максимальный радиус, дал результаты, близкие к первому, но с иным распределением БПЛА.

2.4 Сравнительный анализ

Дополнительно был проведен эксперимент, включавший десять тестов на различных случайных картах высот (размер сетки 20×20), были получены количественные характеристики трёх жадных алгоритмов размещения БПЛА. Для каждого теста фиксировалось количество размещённых БПЛА и уникальная площадь покрытия вне рабочей зоны. Результаты позволяют выявить устойчивые закономерности поведения алгоритмов и сделать обоснованные выводы об их эффективности (рис. 4).

```
--- Тест 1 (seed=0) ---
Алг.1: дронов=43, уникальная внешняя площадь=88
Алг.2: дронов=56, уникальная внешняя площадь=75
Алг.3: дронов=44, уникальная внешняя площадь=100

--- Тест 2 (seed=1) ---
Алг.1: дронов=39, уникальная внешняя площадь=68
Алг.2: дронов=55, уникальная внешняя площадь=76
Алг.3: дронов=37, уникальная внешняя площадь=86

--- Тест 3 (seed=2) ---
Алг.1: дронов=40, уникальная внешняя площадь=96
Алг.2: дронов=56, уникальная внешняя площадь=72
Алг.3: дронов=37, уникальная внешняя площадь=99

--- Тест 4 (seed=3) ---
Алг.1: дронов=43, уникальная внешняя площадь=87
Алг.2: дронов=54, уникальная внешняя площадь=70
Алг.3: дронов=42, уникальная внешняя площадь=93

--- Тест 5 (seed=4) ---
Алг.1: дронов=43, уникальная внешняя площадь=74
Алг.2: дронов=59, уникальная внешняя площадь=64
Алг.3: дронов=43, уникальная внешняя площадь=87

--- Тест 6 (seed=5) ---
Алг.1: дронов=40, уникальная внешняя площадь=100
Алг.2: дронов=57, уникальная внешняя площадь=83
Алг.3: дронов=40, уникальная внешняя площадь=112

--- Тест 7 (seed=6) ---
Алг.1: дронов=44, уникальная внешняя площадь=81
Алг.2: дронов=55, уникальная внешняя площадь=67
Алг.3: дронов=45, уникальная внешняя площадь=89

--- Тест 8 (seed=7) ---
Алг.1: дронов=41, уникальная внешняя площадь=114
Алг.2: дронов=56, уникальная внешняя площадь=75
Алг.3: дронов=40, уникальная внешняя площадь=108

--- Тест 9 (seed=8) ---
Алг.1: дронов=43, уникальная внешняя площадь=72
Алг.2: дронов=57, уникальная внешняя площадь=62
Алг.3: дронов=44, уникальная внешняя площадь=71

--- Тест 10 (seed=9) ---
Алг.1: дронов=39, уникальная внешняя площадь=72
Алг.2: дронов=54, уникальная внешняя площадь=61
Алг.3: дронов=40, уникальная внешняя площадь=99
```

Рис. 4. Сравнение результатов расчетов алгоритмов

Первый алгоритм, ориентированный на максимизацию покрытия внутри зоны с последующей минимизацией выхода за границы, демонстрирует сбалансированные результаты. Он использует умеренное количество БПЛА (в среднем около 42) и обеспечивает сравнительно невысокую внешнюю площадь (85 клеток). При этом разброс значений как по числу БПЛА (от 39 до 44), так и по внешней площади (от 68 до 114) указывает на чувствительность к рельефу, но в целом алгоритм сохраняет предсказуемость.

Второй алгоритм, ставящий во главу угла минимизацию выхода за границы на каждом шаге, закономерно приводит к значительному увеличению числа БПЛА – в среднем 56, что больше, чем у алгоритмов 1 и 3. Однако это позволяет достичь наименьшей уникальной внешней площади (в среднем 70 клеток). Таким образом, алгоритм 2 эффективен, если приоритетом является жёсткое ограничение нежелательного покрытия, даже ценой роста количества устройств. Стабильность внешней площади (диапазон 61-83) подтверждает, что стратегия последовательной минимизации выхода работает предсказуемо.

Третий алгоритм, выбирающий позиции с максимальным радиусом, даёт примерно то же количество БПЛА, что и первый алгоритм, но при этом характеризуется наибольшей внешней площадью (94 клетки) и самым широким её разбросом (от 71 до 112). Это объясняется тем, что крупные БПЛА, размещаемые в первую очередь, могут значительно выходить за границы, и этот эффект не компенсируется последующими мелкими дронами.

3 Заключение

В представленной работе была рассмотрена актуальная задача оперативного развертывания систем связи с использованием беспилотных летательных аппаратов, выступающих в роли воздушных ретрансляторов. В ходе исследования были проанализированы существующие виды подвижной связи, обоснованы преимущества транкинговой связи как наиболее подходящей для решения задач оперативной координации, а также показана необходимость поиска эффективных алгоритмов размещения БПЛА в условиях, когда развертывание стационарной инфраструктуры невозможно или нецелесообразно.

В рамках работы были разработаны и программно реализованы три варианта жадного алгоритма размещения ретрансляторов, каждый из которых руководствуется собственной стратегией выбора позиций: максимизация покрытия внутри рабочей зоны, минимизация выхода сигнала за её границы и приоритет максимального радиуса действия. Проведённое имитационное моделирование на сетке 20×20 со случайным рельефом позволило выявить сильные и слабые стороны каждого подхода.

Полученные в ходе вычислительных экспериментов результаты имеют не только самостоятельное значение, но и служат основой для следующего этапа исследований. Сформированные жадными алгоритмами наборы позиций БПЛА представляют собой качественные начальные приближения, которые могут быть использованы в качестве стартовой популяции для генетического алгоритма.

Дальнейшее развитие исследований может быть направлено в несколько сторон. Во-первых, представляет интерес адаптация предложенного подхода к динамическим сценариям, когда положение абонентов или рельеф местности могут изменяться в реальном времени.

Во-вторых, перспективным направлением является учёт не только количества покрытых абонентов, но и качества канала связи, включая соотношение сигнал/шум и помеховую обстановку. Наконец, важным шагом станет переход от двумерной модели к трёхмерной, учитывающей не только высоты рельефа, но и препятствия в виде зданий и растительности, что позволит приблизить модель к реальным условиям городской или лесной местности.

Литература

1. *Буснюк Н.Н., Мельянец Г.И.* Системы мобильной связи: учебное пособие для вузов, 2-е изд., стер. Санкт-Петербург: Лань, 2023. 128 с.

2. *Хазан В.Л.* Системы специальной радиосвязи: учебное пособие. Москва; Вологда: Инфра-Инженерия, 2025. 236 с.

3. *Kodheli O., Lagunas E., Maturo N., Sharma S.K., Shankar B., Montoya J.F.M., Duncan J.C.M., Spano D., Chatzinotas S., Kisseleff S., Querol J., Lei L., Vu T.X., Goussetis G.* Satellite Communications in the New Space Era: A Survey and Future Challenges // IEEE Communications Surveys & Tutorials. 2021. Vol. 23, No. 1, pp. 70-109.

4. *Роенков Д.Н., Плеханов П.А.* Системы мобильной связи. Коротковолновая и спутниковая связь. Санкт-Петербург: Петербургский государственный университет путей сообщения Императора Александра I, 2023. 31 с. ISBN 978-5-7641-1916-8. EDN TGBRVE.

5. *Addad R.A., Bagaa M., Taleb T., Dutta D., Flinck H.* Optimization Model for Placement of 5G Network Functions for Localization Services // IEEE Transactions on Network and Service Management. 2022. Vol. 19, No. 2, pp. 1809–1825.

6. *Росляков А.В.* Поколения сетей фиксированной связи f1g-F5G часть 2 // Первая миля. 2023. № 1(109). С. 36-47. DOI 10.22184/2070-8963.2023.109.1.36.46. EDN ODRGNG.

7. *Baldini G., Karanasios S., Allen D., Vergari F.* Survey of Wireless Communication Technologies for Public Safety // IEEE Communications Surveys & Tutorials. 2022.

8. *Шорин О.А., Каспари Р.Ю.* Российский рынок профессиональной радиосвязи и критически важных коммуникаций: перспективы // Электросвязь.2021, №6. С. 14-24.

9. *Реформат А.Г., Сосунов В.Г., Плыгунов О.В.* Обзор методик расчета зон покрытия базовых станций сетей подвижной радиосвязи // Символ науки: Международный научный журнал. 2015, №5. С.51-55.

10. *Mozaffari M., Saad W., Bennis M., Debbah M.* A tutorial on UAVs for wireless networks: Applications, challenges, and open problems // IEEE Communications Surveys & Tutorials. 2019. Vol. 21, No. 3, pp. 2334-2360.

11. *Крижановский М.Н., Тихонова О.В.* Методика расстановки базовых станций системы локального позиционирования в рабочей зоне с препятствиями // Моделирование, оптимизация и информационные технологии. 2024. Т. 12, № 2(45). DOI 10.26102/2310-6018/2024.45.2.037. EDN AMTTND.

12. *Степанов О.А., Рожков В.Н.* Применение эвристических алгоритмов для формирования начальной популяции при оптимизации целераспределения с использованием генетических алгоритмов // Научный вестник. Развитие систем управления. 2024. № 3. С. 35-40. EDN KTMECY.

13. *Leonov A.V.* Applying Bio-Inspired Algorithms to Routing Problem Solution in FANET // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». 2017. Т. 17, № 2. С. 5-23. DOI: 10.14529/ctcr170201

14. *Адонин Л.С., Владыко А.Г.* Алгоритмы роевого интеллекта для решения задач оптимизации в системах телекоммуникаций // Труды учебных заведений связи. 2025. Т. 11, № 3. С. 7-24. DOI 10.31854/1813-324X-2025-11-3-7-24. EDN JUAAMB.

15. *Перепелкин Д.А., Нгуен В.Т.* Интеллектуальная многопутевая маршрутизация в программно-конфигурируемых сетях на основе алгоритма искусственной пчелиной колонии // Информационные технологии. 2022. Т. 28, № 8. С. 395-404. DOI 10.17587/it.28.395-404. EDN BONWWZ.

16. *Егорова К.В.* Имитационная модель управления полетом группы беспилотных летательных аппаратов на основе алгоритма пчелиной колонии // Вестник Воронежского государственного технического университета. 2023. Т. 19, № 2. С. 68-71. DOI 10.36622/VSTU.2023.19.2.010. EDN INIXQJ.

17. *Chen Y., Li N., Zhong X., Xie W.* Joint Trajectory and Scheduling Optimization for The Mobile UAV Aerial Base Station: A Fairness Version // Applied Sciences. 2019. Vol. 9, No. 15. 3101.

18. *Wen X., Ruan Y., Li Y., Xia H., Zhang R., Wang C., Liu W., Jiang X.* Improved Genetic Algorithm based 3-D deployment of UAVs // Journal of Communications and Networks. 2022. Vol. 24, No. 2, pp. 223-231. DOI: 10.23919/JCN.2022.000014

19. *Костин А.С., Кучко Д.В.* Исследование алгоритма маршрутизации для решения задачи коммивояжера на примере алгоритма муравьиной колонии // Системный анализ и логистика. 2023. № 4(38). С. 47-53. DOI 10.31799/2077-5687-2023-4-47-53. EDN VTXDDL.

20. *Иванов В.С., Увайсов С.У., Иванов И.А.* Алгоритм автоматического размещения базовых станций транкинговых систем связи // Труды учебных заведений связи. 2023. Т. 9, № 5. С. 25-34. DOI 10.31854/1813-324X-2023-9-5-25-34. EDN ENPOPN.

GREEDY ALGORITHMS FOR POSITIONING UAV-RETRANSLATORS IN TRUNKING COMMUNICATION SYSTEMS

Vyacheslav S. Ivanov, *MIREA – Russian Technological University, Moscow, Russia,* ***ivanov_vs@mirea.ru***
Saygid U. Uvajsov, *MIREA – Russian Technological University, Moscow, Russia,* ***uvajsov@mirea.ru***

Abstract

In this paper, the task of operational deployment of trunking communication systems using unmanned aerial vehicles performing the functions of aerial repeaters is investigated. As a criterion for the availability of a stable connection between the repeater and ground subscribers, the ratio between the permissible level of signal loss and actual losses along the propagation path is considered, calculated using the Okumura-Hata empirical model adapted to account for terrain and elevation differences. To determine the positions of unmanned aerial vehicles, it is proposed to use greedy heuristic algorithms that ensure an acceptable calculation speed in conditions of limited decision-making time. A comparative evaluation of three variants of the greedy algorithm is carried out, which differ in the rules for choosing the placement point at each step. Using this assessment, a conclusion is made about the suitability of each approach in terms of the balance between the number of repeaters used and the area of undesirable coverage outside the service area, and the results of computational experiments on a grid with random relief are presented. At the current stage, it is assumed that the solutions obtained by greedy algorithms will be used as the initial population for a genetic algorithm that performs a deeper search for the global optimum. Software has been developed in Python to visually visualize coverage areas and analyze the results.

Keywords: *UAVs, communication repeater, trunking communication systems, greedy algorithm, heuristic algorithms..*

References

[1] N.N. Busnyuk, G.I. Melianets , "Mobile Communication Systems: Textbook for Universities," 2nd ed., ster. Saint Petersburg: Lan, 2023. 128 p.

[2] V.L. Khazan, "Special Radio Communication Systems: Textbook," Moscow; Vologda: Infra-Engineering, 2025. 236 p.

[3] O. Kodheli, E. Lagunas, N. Maturo, S.K. Sharma, B. Shankar, J.F.M. Montoya, J.C.M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T.X. Vu, "Goussetis G. Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys & Tutorials.* 2021. Vol. 23, No. 1, pp. 70-109.

[4] D.N. Roenkov, P.A. Plekhanov, "Mobile Communication Systems. Shortwave and Satellite Communication," St. Petersburg: St. Petersburg State Transport University of Emperor Alexander I, 2023. 31 p. ISBN 978-5-7641-1916-8.

[5] R.A., Addad, M., Bagaa, T. Taleb, D. Dutta, H. Flinck, "Optimization Model for Placement of 5G Network Functions for Localization Services," *IEEE Transactions on Network and Service Management*. 2022. Vol. 19, No. 2, pp. 1809-1825.

[6] A. V. Roslyako, "Generations of Fixed Communication Networks f1g-F5G Part 2 / v," *First Mile*. 2023. No. 1(109)., pp. 36-47. DOI 10.22184/2070-8963.2023.109.1.36.46. - EDN ODRGNG.

[7] G. Baldini, S. Karanasios, D. Allen, F. Vergari, "Survey of Wireless Communication Technologies for Public Safety," *IEEE Communications Surveys & Tutorials*. 2022.

[8] O.A. Shorin, R.Yu. Kaspari, "Russian Market of Professional Radio Communication and Critical Communications: Prospects," *Elektrosvyaz*. 2021, No.6, pp.14-24.

[9] A.G. Reformat, V.G. Sosunov, O.V. Plygunov, "Overview of Methods for Calculating Coverage Zones of Base Stations in Mobile Radio Communication Networks," *Symbol of Science: International Scientific Journal.* 2015, No. 5, pp. 51-55.

[10] M. Mozaffari, W. Saad, M. Bennis, M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Communications Surveys & Tutorials.* 2019. Vol. 21, No. 3, pp. 2334-2360.

[11] M. N. Krizhanovsky, O. V. Tikhonova, "Methodology for Placing Base Stations of a Local Positioning System in an Obstructed Working Area," *Modeling, Optimization, and Information Technologies*. 2024. Vol. 12, No. 2(45). DOI 10.26102/2310-6018/2024.45.2.037.

[12] O. A. Stepanov, V. N. Rozhkov, "Application of heuristic algorithms for the formation of the initial population in the optimization of target allocation using genetic algorithms," *Scientific Bulletin. Development of Control Systems*. 2024. No. 3, pp. 35-40.

[13] A.V. Leonov, "Applying Bio-Inspired Algorithms to Routing Problem Solution in FANET," *Bulletin of the South Ural State University. Series "Computer Technologies, Control, and Radioelectronics*". 2017. Vol. 17, No. 2, pp. 5-23. DOI: 10.14529/ctcr170201.

[14] L. S. Adonin, A. G. Vladyko, "Swarm Intelligence Algorithms for Solving Optimization Problems in Telecommunication Systems," *Proceedings of Communications Educational Institutions.* 2025. Vol. 11, No. 3, pp. 7-24. DOI 10.31854/1813-324X-2025-11-3-7-24.

[15] D. A. Perepelkin, V. T. Nguyen, "ntelligent Multi-Path Routing in Software-Defined Networks Based on the Artificial Bee Colony Algorithm," *Information Technologies*. 2022. Vol. 28, No. 8, pp. 395-404. DOI 10.17587/it.28.395-404.

[16] K. V. Egorova, "Simulation model of flight control of a group of unmanned aerial vehicles based on the bee colony algorithm," *Bulletin of the Voronezh State Technical University*. 2023. Vol. 19, No. 2, pp. 68-71. DOI 10.36622/VSTU.2023.19.2.010.

[17] Y. Chen, N. Li, X. Zhong, W. Xie, "Joint Trajectory and Scheduling Optimization for The Mobile UAV Aerial Base Station: A Fairness Version," *Applied Sciences.* 2019. Vol. 9, No. 15. 3101.

[18] X. Wen, Y. Ruan, Y. Li, H. Xia, R. Zhang, C. Wang, W. Liu, X. Jiang, "Improved Genetic Algorithm based 3-D deployment of UAV," *Journal of Communications and Networks.* 2022. Vol. 24, No. 2, pp 223-231. DOI: 10.23919/JCN.2022.000014

[19] A. S. Kostin, D. V. Kuchko, "Research of the Routing Algorithm for Solving the Traveling Salesman Problem Using the Ant Colony Algorithm," *System Analysis and Logistics.* 2023. No. 4(38), pp. 47-53. DOI 10.31799/2077-5687-2023-4-47-53.

[20] V. S. Ivanov, S. U. Uvaisov, and I. A. Ivanov, "Algorithm for Automatic Placement of Base Stations in Trunking Communication Systems," *Proceedings of Communications Educational Institutions*. 2023. Vol. 9, No. 5, pp. 25-34. DOI 10.31854/1813-324X-2023-9-5-25-34.

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ СМЕЩЕНИЯ ОЦЕНОК ВРЕМЕНИ КРУГОВОГО ОБОРОТА В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ ПРИ НЕСТАЦИОНАРНЫХ НАГРУЗКАХ

DOI: 10.36724/2072-8735-2026-20-6-12-19

Manuscript received 27 February 2026;
Accepted 07 April 2026

Ключевые слова: *математическое моделирование, время кругового оборота, система M/M/1, нестационарный режим, телекоммуникационные сети, QUIC, оценка задержки*

Батенков Кирилл Александрович,
Московский технический университет связи и информатики, Москва, Россия, ***pustur@yandex.ru***

Исследуется проблема точности оценки времени кругового оборота (RTT) в современных протоколах транспортного уровня, таких как TCP и QUIC, функционирующих в условиях высокодинамичного нестационарного трафика. Традиционные алгоритмы экспоненциального сглаживания (EWMA), регламентированные стандартами вроде RFC 9002, базируются на упрощающем предположении о стационарности процессов обслуживания очередей. Однако в реальных телекоммуникационных системах внезапные всплески нагрузки существенно нарушают это условие, вызывая систематические погрешности измерений задержек. В работе представлена математическая модель на базе классической системы массового обслуживания M/M/1, рассматриваемой в переходном режиме. С использованием системы дифференциальных уравнений Колмогорова и численного метода матричной экспоненты получены точные аналитические выражения для эволюции плотностей вероятностей, представляющих собой смеси распределений Эрланга, для "истинного" RTT, соответствующего произвольному наблюдателю, и "измеряемого" RTT, фиксируемого уходящим потоком пакетов. Теоретически обосновано и численно подтверждено различие между распределением состояний системы в произвольный момент времени и распределением, наблюдаемым уходящими заявками в нестационарной фазе. Это различие обуславливает существенное смещение математического ожидания измеряемой задержки относительно действительной величины преимущественно в периоды транзиторных процессов роста очереди. Численное моделирование сценария скачка коэффициента использования канала с начального значения до конечного продемонстрировало возникновение пика относительной погрешности оценки RTT величиной до 9,6% в фазе активного накопления заявок. Полученные результаты имеют критическое практическое значение для разработки новых адаптивных алгоритмов настройки таймаутов повторной передачи (PTO/RTO), позволяющих минимизировать количество ложных ретрансмиссий и предотвратить явление перегрузки буферов. Работа объединяет теоретический анализ и прикладные задачи связи. Исследование вносит вклад в развитие методов математического моделирования стохастических систем и повышения эффективности управления потоками данных в телекоммуникационных сетях следующего поколения.

Информация об авторе:
Батенков Кирилл Александрович, *д.т.н., профессор кафедры Сети связи и системы коммутации, Московский технический университет связи и информатики, Москва, Россия, orcid.org/0000-0001-6083-1242*

Для цитирования:
Батенков К.А. Математическое моделирование смещения оценок времени кругового оборота в телекоммуникационных системах при нестационарных нагрузках // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 12-19.
For citation:
K.A. Batenkov "Mathematical Modeling of Round-Trip Time Estimation Bias in Telecommunication Systems under Non-Stationary Loads," *T-Comm*, 2026, vol. 20, no.6, pp. 12-19. (*in Russian*)

Введение

Современные телекоммуникационные системы характеризуются высокой динамикой трафика и жесткими требованиями к задержкам [1]. Протоколы транспортного уровня, такие как Transmission Control Protocol (TCP) и Quick UDP Internet Connections (QUIC), для управления механизмами контроля перегрузки, регулирования потока и восстановления после потери пакетов в значительной степени полагаются на точные оценки времени прохождения сигнала (Round-Trip Time, RTT) [2, 3]. Суть этих механизмов заключается в способности различать потерю пакетов из-за перегрузки и потерю из-за случайных ошибок в канале, что в первую очередь определяется значениями тайм-аута, рассчитанными на основе измерений RTT [4].

В спецификации QUIC (RFC 9002) и современных версиях TCP (например, TCP CUBIC или BBR) используется экспоненциальное скользящее среднее (EWMA) для сглаживания измеренных значений RTT [5]. Данный подход эффективен в стационарных режимах работы сети, когда статистические характеристики потока заявок и времени обслуживания остаются постоянными во времени [6, 7]. Однако реальные телекоммуникационные сети часто функционируют в нестационарных режимах: всплески трафика, масштабирование облачных ресурсов, восстановление после сбоев маршрутизаторов [8, 9] – все эти сценарии приводят к переходным процессам в очередях сетевых устройств [10].

Актуальность исследования обусловлена тем, что стандартные модели оценки RTT не учитывают различие между распределением состояния системы, наблюдаемым произвольным участником, и распределением, наблюдаемым уходящим пакетом (измерителем) [11]. В стационарном режиме для системы M/M/1 это различие нивелируется благодаря теореме Берка, однако в переходных периодах возникает систематическое смещение [12, 13]. Это смещение может привести к некорректной установке таймаутов повторной передачи (RTO), вызывая ложные ретрансмиссии или, наоборот, излишне долгую реакцию на потерю пакетов, что напрямую влияет на пропускную способность и задержки в сетях связи [14].

Целью данной работы является построение строгой математической модели эволюции распределения RTT в системе M/M/1 при скачкообразном изменении нагрузки, количественная оценка возникающего смещения между измеряемым и действительным RTT и формулировка рекомендаций для разработчиков телекоммуникационных протоколов [15].

1 Математическая постановка задачи

1.1 Описание системы массового обслуживания

Рассмотрим одноканальную систему массового обслуживания (СМО) типа M/M/1, которая является классической моделью для описания буфера маршрутизатора или сетевого интерфейса. Пусть $\lambda(t)$ – интенсивность входящего потока заявок (пакетов), а μ – интенсивность обслуживания (пропускная способность канала). В общем случае процесс является неоднородным пуассоновским, однако для анализа переходного процесса при скачке нагрузки удобно рассматривать кусочно-стационарную модель.

Состояние системы в момент времени t описывается случайной величиной $N(t)$ — числом заявок в системе (на обслуживании и в очереди). Пространство состояний дискретно: $S = \{0,1,2,\dots\}$. Эволюция вероятностей состояний $P_i(t) = \mathbb{P}\{N(t) = i\}$ описывается системой дифференциальных уравнений Колмогорова (прямые уравнения) [16]:

$$\frac{dP_i(t)}{dt} = -(\lambda+\mu)P_i(t) + \lambda P_{i-1}(t) + \mu P_{i+1}(t), \quad i \geq 1,$$

$$\frac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t).$$

Здесь предполагается, что в рассматриваемый интервал времени интенсивности λ и μ постоянны. В матричной форме эту систему можно записать как [17]:

$$\frac{d\boldsymbol{P}(t)}{dt} = \boldsymbol{P}(t)\boldsymbol{Q},$$

где $\boldsymbol{P}(t) = [P_0(t), P_1(t), \dots, P_N(t)]$ – вектор-строка вероятностей состояний (для численной реализации пространство состояний ограничивается достаточно большим числом N), а $\boldsymbol{Q}$ – инфинитезимальная матрица переходных интенсивностей (генератор процесса) [18]:

$$\boldsymbol{Q} = \begin{pmatrix} -\lambda & \lambda & 0 & \cdots \\ \mu & -(\lambda+\mu) & \lambda & \cdots \\ 0 & \mu & -(\lambda+\mu) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Решение этой системы при начальном условии $\boldsymbol{P}(0)$ имеет вид:

$$\boldsymbol{P}(t) = \boldsymbol{P}(0)e^{\boldsymbol{Q}t},$$

где $e^{\boldsymbol{Q}t}$ – матричная экспонента. Данный метод является основой численного анализа, позволяя получать точные значения вероятностей состояний в любой момент времени переходного процесса [19].

1.2 Моделирование времени обслуживания

В контексте телекоммуникаций время обслуживания заявки соответствует времени передачи пакета фиксированного размера через канал связи. В модели M/M/1 время обслуживания одной заявки τ является экспоненциально распределенной случайной величиной с параметром μ:

$$f_\tau(x) = \mu e^{-\mu x}, \quad x \geq 0.$$

Математическое ожидание времени обслуживания равно $\mathbb{E}[\tau] = 1/\mu$, а дисперсия $D[\tau] = 1/\mu^2$.

Если пакет застал в системе i заявок перед собой (одна на обслуживании и $i-1$ в очереди, либо i в очереди, если считать состояние системы в момент прихода), то полное время ожидания начала обслуживания складывается из времен обслуживания этих i заявок. Сумма k независимых экспоненциально распределенных величин с одинаковым параметром μ имеет распределение Эрланга порядка k.

Пусть пакет застал в системе i заявок. Тогда время ожидания в очереди W_q имеет распределение Эрланга порядка i с параметром μ. Полное время пребывания в системе (время обслуживания + ожидание) для этого пакета будет суммой $i+1$ экспоненциальных величин (ожидание i предыдущих + собственное обслуживание). Таким образом, условное время пребывания $T_{sys} \mid i$ распределено по закону Эрланга порядка $k = i+1$:

$$f_{T_{sys}|i}(x) = \frac{\mu^{i+1} x^i e^{-\mu x}}{i!}, \quad x \geq 0.$$

1.3 Модель задержки RTT

В телекоммуникационных сетях время кругового оборота (RTT) складывается из времени передачи в прямом направлении, времени обработки в узле назначения, времени передачи в обратном направлении и времени ожидания в очередях. Для упрощения модели будем считать, что основная вариативность вносится очередью в исследуемом узле, а времена распространения сигнала (d_{prop}) и обработки детерминированы или пренебрежимо малы по сравнению с временем ожидания в очереди при высоких нагрузках.

Тогда RTT можно представить как:

$$\mathrm{RTT} = 2d_{prop} + T_{sys},$$

где T_{sys} — полное время пребывания пакета в системе (очередь + обслуживание). Условное распределение RTT при условии, что пакет застал i заявок в системе, является сдвинутым распределением Эрланга:

$$f_{\mathrm{RTT}|i}(x) = f_{T_{sys}|i}(x - 2d_{prop}), \quad x \geq 2d_{prop}.$$

Математическое ожидание условного RTT равно:

$$\mathbb{E}[\mathrm{RTT} \mid i] = 2d_{prop} + \frac{i+1}{\mu}.$$

Данная формула показывает линейную зависимость ожидаемой задержки от длины очереди, что является свойством, используемым в алгоритмах управления перегрузками (например, в алгоритме BBR для оценки RTprop).

2 Различие между истинным и измеряемым RTT

Ключевой проблемой математического моделирования в телекоммуникациях является корректный учет того, как производится измерение. Существует принципиальное различие между характеристиками системы, наблюдаемыми внешним наблюдателем в произвольный момент времени, и характеристиками, наблюдаемыми уходящим потоком пакетов.

2.1. Распределение состояний для произвольного наблюдателя

Распределение $P_i(t)$, полученное из уравнений Колмогорова, описывает вероятность того, что в произвольный момент времени t в системе находится i заявок. Это распределение характеризует «истинное» состояние системы. Если бы мы могли мгновенно сканировать очередь в случайные моменты времени, мы бы получили гистограмму, соответствующую $P_i(t)$.

«Истинное» математическое ожидание длины очереди в момент t вычисляется как:

$$\mathbb{E}_{true}[N(t)] = \sum_{i=0}^{\infty} i\, P_i(t).$$

Соответственно, «действительное» среднее время пребывания в системе (и, следовательно, базовая компонента RTT) для гипотетического пакета, пришедшего в момент t, оценивается через усреднение условных ожиданий по распределению $P_i(t)$:

$$\mathbb{E}_{true}[\mathrm{RTT}(t)] = 2d_{prop} + \frac{1}{\mu}\left(\sum_{i=0}^{\infty}(i+1)P_i(t)\right) = 2d_{prop} + \frac{\mathbb{E}_{true}[N(t)] + 1}{\mu}.$$

2.2. Распределение состояний для уходящего пакета

Протоколы TCP и QUIC измеряют RTT только для пакетов, которые были успешно переданы и подтверждены. То есть измерение производится по факту ухода пакета из системы. В стационарном режиме для системы M/M/1 справедлива теорема Берка, которая утверждает, что выходной поток также является пуассоновским, и распределение числа заявок в системе в момент ухода пакета совпадает с распределением в произвольный момент времени ($D_i = P_i$).

Однако в нестационарном режиме это равенство нарушается. Интенсивность ухода пакетов пропорциональна μ, если система не пуста. Вероятность того, что уходящий пакет оставит в системе i заявок, пропорциональна вероятности нахождения в системе $i+1$ заявок непосредственно перед уходом.

Обозначим через $D_i(t)$ вероятность того, что уходящий в момент t пакет застал в системе i других заявок (или, эквивалентно, оставил после себя i заявок). Для процесса рождения и гибели связь между распределением состояний в произвольный момент $P_i(t)$ и распределением по уходящим заявкам $D_i(t)$ в нестационарном режиме определяется соотношением потоков.

В общем случае, плотность вероятности ухода из состояния $i+1$ в состояние i равна $\mu P_{i+1}(t)$. Нормируя на полную интенсивность ухода из системы, получаем:

$$D_i(t) = \frac{\mu P_{i+1}(t)}{\sum_{j=1}^{\infty} \mu\, P_j(t)} = \frac{P_{i+1}(t)}{1 - P_0(t)}, \quad i \geq 0.$$

Это соотношение справедливо при условии, что система не пуста ($1 - P_0(t) > 0$). Если система пуста, уход невозможен, и измерение RTT не производится.

Таким образом, «измеряемое» распределение длины очереди сдвинуто относительно «истинного». Уходящий пакет «видит» систему более загруженной, чем она есть в среднем в произвольный момент, особенно в фазе роста очереди. Это явление известно как «парадокс инспектора» или эффект выборки по длине интервала, адаптированный для потоков событий.

«Измеряемое» математическое ожидание длины очереди, которое фактически влияет на оценку взвешенного среднего измеренного времени задержки в протоколах, равно:

$$\mathbb{E}_{meas}[N(t)] = \sum_{i=0}^{\infty} i\, D_i(t) = \sum_{i=0}^{\infty} i \frac{P_{i+1}(t)}{1 - P_0(t)}.$$

Преобразуем сумму:

$$\sum_{i=0}^{\infty} i\, P_{i+1}(t) = \sum_{k=1}^{\infty}(k-1)P_k(t) = \sum_{k=1}^{\infty} k\, P_k(t) - \sum_{k=1}^{\infty} P_k(t) = \mathbb{E}_{true}[N(t)] - (1 - P_0(t)).$$

Подставляя это в выражение для $\mathbb{E}_{meas}$, получаем формулу смещения:

$$\mathbb{E}_{meas}[N(t)] = \frac{\mathbb{E}_{true}[N(t)] - (1 - P_0(t))}{1 - P_0(t)} = \frac{\mathbb{E}_{true}[N(t)]}{1 - P_0(t)} - 1.$$

2.3. Формула смещения RTT

На основе полученных выражений для длин очередей можно вывести формулу для смещения математического ожидания RTT.

Действительное среднее RTT (для произвольного пакета, который мог бы прийти):

$$\mathbb{E}_{true}[\mathrm{RTT}] = 2d_{prop} + \frac{\mathbb{E}_{true}[N] + 1}{\mu}.$$

Измеряемое среднее RTT (для пакета, который успешно прошел систему):

$$\mathbb{E}_{meas}[\mathrm{RTT}] = 2d_{prop} + \frac{\mathbb{E}_{meas}[N] + 1}{\mu}.$$

Подставим выражение для $\mathbb{E}_{meas}[N]$:

$$\mathbb{E}_{meas}[\mathrm{RTT}] = 2d_{prop} + \frac{1}{\mu}\left(\frac{\mathbb{E}_{true}[N]}{1 - P_0} - 1 + 1\right) = 2d_{prop} + \frac{\mathbb{E}_{true}[N]}{\mu(1 - P_0)}.$$

Разность между измеряемым и действительным средним RTT составляет:

$$\Delta\mathrm{RTT} = \mathbb{E}_{meas}[\mathrm{RTT}] - \mathbb{E}_{true}[\mathrm{RTT}] = \frac{\mathbb{E}_{true}[N]}{\mu(1 - P_0)} - \frac{\mathbb{E}_{true}[N] + 1}{\mu}.$$

Приводя к общему знаменателю:

$$\Delta\mathrm{RTT} = \frac{1}{\mu}\left(\frac{\mathbb{E}_{true}[N] - (\mathbb{E}_{true}[N] + 1)(1 - P_0)}{1 - P_0}\right) = \frac{1}{\mu}\left(\frac{\mathbb{E}_{true}[N]P_0 - (1 - P_0)}{1 - P_0}\right).$$

Относительная погрешность (смещение в процентах) определяется как:

$$\delta(t) = \frac{|\mathbb{E}_{meas}[\mathrm{RTT}] - \mathbb{E}_{true}[\mathrm{RTT}]|}{\mathbb{E}_{true}[\mathrm{RTT}]} \times 100\%.$$

В стационарном режиме для M/M/1 известно, что $\mathbb{E}_{true}[N] = \frac{\rho}{1-\rho}$ и $P_0 = 1 - \rho$, где $\rho = \lambda/\mu$. Подставив эти значения, легко убедиться, что $\Delta\mathrm{RTT} = 0$. Это подтверждает корректность модели: в стационаре смещение отсутствует. Однако в переходном режиме $P_0(t)$ и $\mathbb{E}_{true}[N(t)]$ меняются несогласованно, что приводит к ненулевому ΔRTT.

3 Численное моделирование и анализ результатов

Для верификации теоретических выкладок и оценки масштаба эффекта было проведено численное моделирование эволюции распределений вероятностей. Использовался метод матричной экспоненты для решения системы уравнений Колмогорова, реализованный в среде Python с использованием библиотеки SciPy.

3.1. Параметры эксперимента

Моделировался сценарий скачка нагрузки на сетевом интерфейсе.

Интенсивность обслуживания: $\mu = 1.0$ (нормированная единица, например, 1000 пакетов/сек).

Начальный коэффициент использования: $\rho_0 = 0.35$. Соответствует режиму малой загрузки.

Конечный коэффициент использования: $\rho_1 = 0.82$. Соответствует режиму высокой загрузки, близкому к насыщению.

Время распространения сигнала: $d_{prop} = 0.02$ (в нормированных единицах времени обслуживания).

Начальное распределение: стационарное распределение для ρ_0.

Интервал моделирования: $t \in [0, 16]$ единиц времени.

Такой сценарий типичен для телекоммуникационных приложений: например, начало видеотрансляции, открытие новой вкладки браузера с тяжелым контентом или переключение маршрута трафика на резервный канал.

3.2. Эволюция распределения длины очереди

На рисунке 1 представлена эволюция распределения вероятностей длины очереди $P_i(t)$ в различные моменты времени.

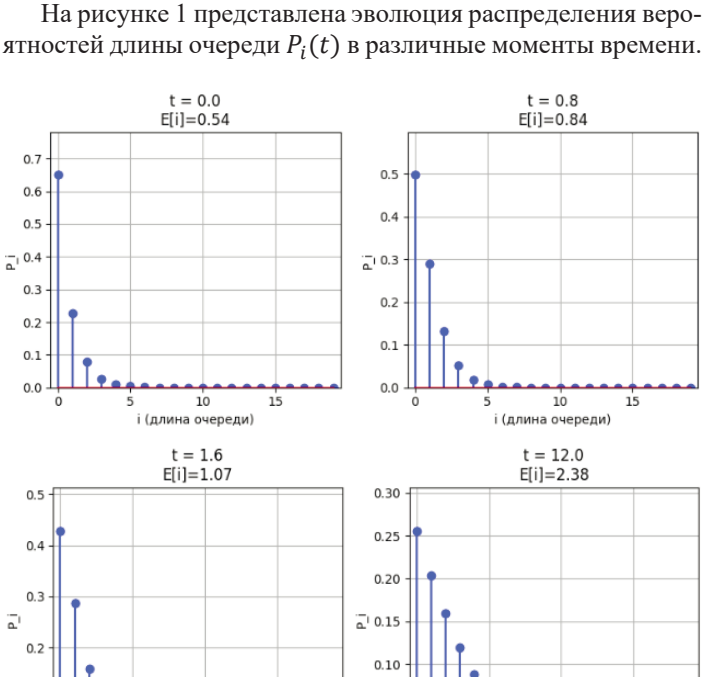


Рис. 1. Эволюция распределения вероятностей длины очереди $P_i(t)$ в различные моменты времени

$t = 0$: Система находится в стационарном состоянии для $\rho = 0{,}35$. Математическое ожидание длины очереди $E[i] \approx 0{,}54$. Распределение быстро убывает, вероятность пустой системы высока ($P_0 \approx 0{,}65$).

$t = 0{,}8$: Произошел скачок интенсивности входной нагрузки. Очередь начинает расти. $E[i]$ увеличивается до 0,84. Хвост распределения утяжеляется.

$t = 1{,}6$: Система приближается к пику переходного процесса. $E[i] \approx 1{,}07$. Наблюдается существенное расхождение между текущим состоянием и будущим стационарным.

$t = 12$: Система вышла на новый стационарный режим для $\rho = 0{,}82$. $E[i] \approx 2{,}38$. Распределение стало более пологим, вероятность простоя упала до $P_0 \approx 0{,}25$.

Важно отметить, что рост среднего значения очереди нелинеен. В начальный момент скорость роста определяется разницей $\lambda - \mu$, но по мере накопления очереди влияние границы $i = 0$ ослабевает, и система ведет себя подобно случайному блужданию с дрейфом.

3.3. Эволюция плотностей вероятности RTT

На рисунках 2–5 представлены графики плотностей вероятности $f(\text{RTT})$ для истинного (RTT_{true}) и измеряемого (RTT_{meas}) распределений в те же моменты времени. Плотности рассчитывались как смесь распределений Эрланга:

$$f_{\text{RTT}}(x) = \sum_{i=0}^{N} w_i \cdot \text{Erlang}(x - 2d_{prop}; k = i + 1, \mu),$$

где веса w_i равны $P_i(t)$ для истинного распределения и $D_i(t)$ для измеряемого.

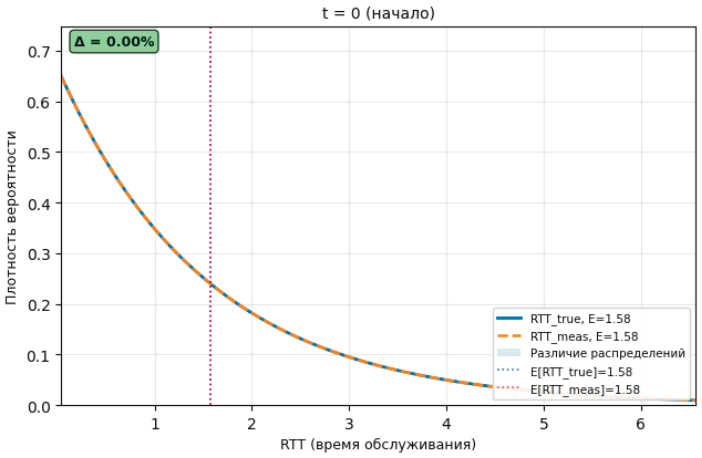


Рис. 2. Графики плотностей вероятности для истинного и измеряемого распределений RTT в $t = 0$

Анализ графиков:

1. Начальный момент (рис. 2, $t = 0$): Кривые практически совпадают. Смещение $\Delta = 0\%$. Это подтверждает теоретический вывод о том, что в стационарном режиме (даже при низкой нагрузке) оценки несмещенные.

2. Середина переходного процесса (рис. 3, $t = 0{,}8$): Наблюдается расхождение хвостов распределений. Измеряемое распределение смещено вправо. Пик погрешности средних значений составляет 8,76%. Это означает, что протокол, измеряющий RTT только по успешным пакетам, видит задержку почти на 9% выше, чем средняя задержка для всех потенциальных пакетов в этот момент.

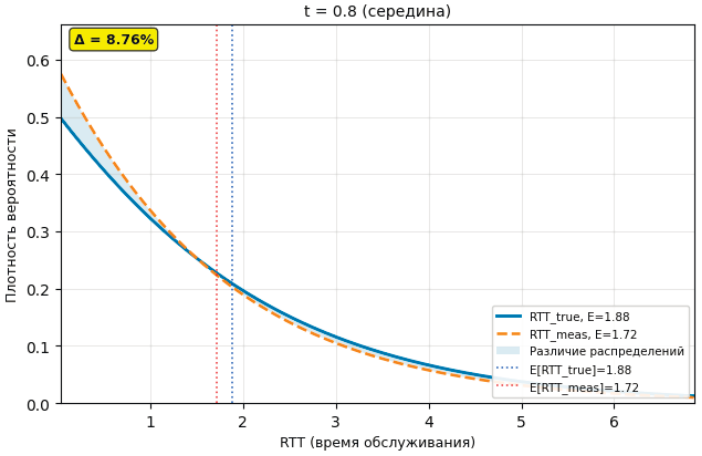


Рис. 3. Графики плотностей вероятности для истинного и измеряемого распределений RTT в $t = 0{,}8$

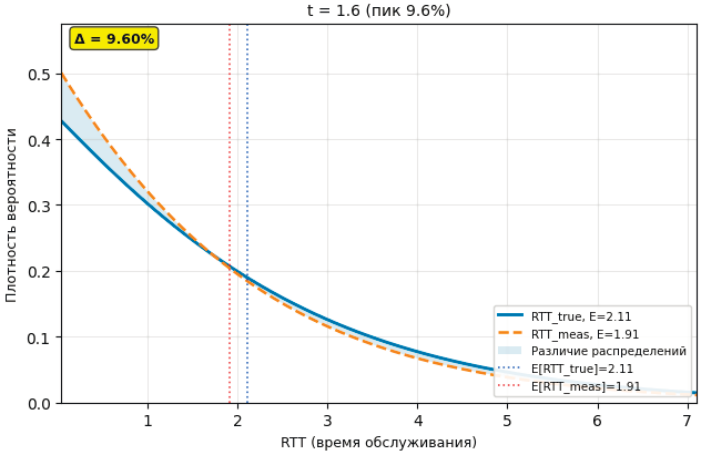


Рис. 4. Графики плотностей вероятности для истинного и измеряемого распределений RTT в $t = 1{,}6$

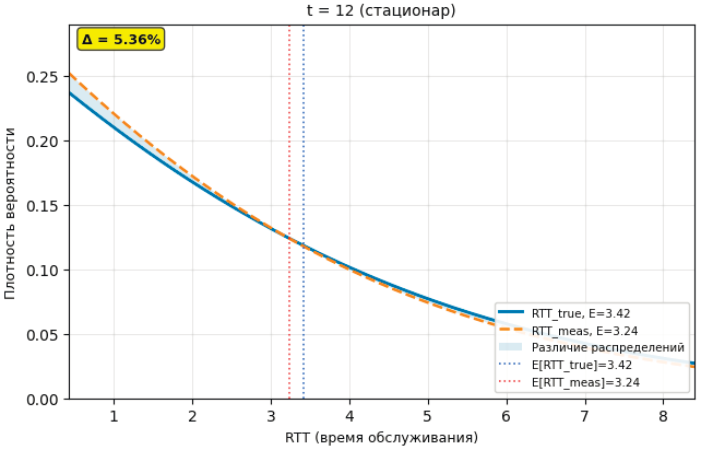


Рис. 5. Графики плотностей вероятности для истинного и измеряемого распределений RTT в $t = 12$

3. Пик погрешности (рис. 4, $t = 1{,}6$): Достигается максимальное расхождение $\Delta = 9{,}60\%$. В этот момент система наиболее нестабильна. Вероятность $P_0(t)$ еще не успела упасть до стационарного уровня нового режима, а средняя длина очереди уже значительно выросла.

Согласно выведенной формуле $\mathbb{E}_{meas} \propto \frac{\mathbb{E}_{true}}{1-P_0}$, знаменатель уменьшается медленнее, чем растет числитель в определенные фазы, усиливая эффект.

4. Близкое к стационарному (рис. 5, $t = 12$): Кривые снова сходятся. Погрешность падает до 5,36% (в данном конкретном срезе из-за дискретизации и ограничений хвоста, в пределе стремится к 0). В стационарном режиме $\rho = 0{,}82$ смещение должно отсутствовать, однако на графике $t = 12$ видно остаточное расхождение, связанное с тем, что система еще не полностью достигла асимптотики, либо с особенностями усечения ряда в численном методе. Тем не менее, тренд на сходимость очевиден.

3.4. Интерпретация с точки зрения телекоммуникаций

Полученный результат имеет прямое практическое значение. Алгоритмы оценки RTT в QUIC (RFC 9002) используют фильтр первого порядка:

$$\text{smoothed\_rtt} = (1-\alpha)\cdot \text{smoothed\_rtt} + \alpha \cdot \text{sample\_rtt},$$

где sample_rtt – это измеряемое значение. Как показано в работе, sample_rtt в переходном режиме систематически завышено относительно среднего времени пребывания в системе. Это завышение приводит к двум возможным сценариям:

1. Завышение PTO (Probe Timeout): Таймаут становится консервативным. Это снижает риск ложных повторных передач, но увеличивает задержку восстановления при реальной потере пакета. Для приложений реального времени (VoIP, онлайн-игры) это зачастую недопустимо.

2. Искажение оценки доступной пропускной способности: В алгоритмах типа BBR, которые используют RTT для быстрой оценки объема данных (BDP – Bandwidth-Delay Product), завышенная оценка RTT приведет к завышенной оценке BDP и, как следствие, к переполнению буферов и росту очередей.

Пик погрешности в 9,6% является значительным. Например, при базовом RTT 100 мс ошибка составляет почти 10 мс. Для высокочастотного трейдинга или тактильного интернета это может стать критичным.

4. Обсуждение и рекомендации

Результаты моделирования демонстрируют, что детерминированная аппроксимация или использование только стационарных формул Литтла недопустимы при анализе динамики сетей. Необходимо учитывать вероятностную природу процесса измерения.

4.1. Влияние на проектирование протоколов

Разработчикам стека TCP/IP рекомендуется учитывать возможность временного смещения оценок при резких изменениях трафика.

Инициализация PTO: При установлении соединения или после периода простоя не следует полагаться на первое измерение RTT. Рекомендуется использовать увеличенный начальный запас, так как первое измерение может попасть в фазу роста очереди.

Адаптивный коэффициент сглаживания: В периоды высокой дисперсии измерений RTT (джиттера) целесообразно уменьшать коэффициент α в EWMA-фильтре, чтобы снизить влияние выбросов, вызванных переходными процессами в очередях.

Явная сигнализация перегрузки (ECN): Использование механизмов ECN (Explicit Congestion Notification) позволяет получать информацию о заполнении очереди без ожидания потери пакетов или искажения RTT, что частично нивелирует описанный эффект.

4.2. Применение в задачах QoS

Для операторов связи полученные формулы позволяют строить более точные модели SLA (Service Level Agreement). Гарантии задержки обычно даются для перцентилей распределения (например, 95-й перцентиль). Как показано в работе, форма распределения измеряемого RTT отличается от истинного (рис. 3–5, область различия распределений). В переходном режиме хвост измеряемого распределения может быть тяжелее. Это значит, что мониторинг сети активными зондами (ping, twamp), которые имитируют уходящие пакеты, может показывать худшую картину, чем реально испытывает средний пользовательский трафик в тот же момент, или наоборот, в зависимости от синхронизации зондов с фазой переходного процесса.

4.3. Ограничения модели

Предложенная модель базируется на предположении об экспоненциальном распределении времени обслуживания (M/M/1). В реальных сетях размеры пакетов могут быть детерминированными (M/D/1) или иметь иное распределение. Однако качественный эффект смещения, вызванный различием распределений $P_i(t)$ и $D_i(t)$ в нестационарном режиме, сохранится и для других дисциплин обслуживания, так как он обусловлен свойствами потоков событий, а не конкретным видом функции обслуживания. Количественная оценка погрешности может измениться, но порядок величин останется сопоставимым.

Также в модели не учтены механизмы повторной передачи. В реальности потеря пакетов и их ретрансмиссия создают дополнительную корреляцию во входном потоке, превращая его в немарковский. Однако для первого приближения и оценки порядка эффекта модель M/M/1 является достаточной и общепринятой в академической среде.

Заключение

В работе проведено комплексное исследование особенностей оценки времени кругового оборота в телекоммуникационных системах в нестационарных режимах.

1. Построена полная вероятностная модель RTT на основе смеси распределений Эрланга с весами, определяемыми решением системы уравнений Колмогорова.

2. Аналитически доказано и численно подтверждено наличие систематического смещения между математическим ожиданием истинного и измеряемого RTT в переходных процессах.

3. Показано, что для сценария скачка нагрузки с $\rho = 0{,}35$ до $\rho = 0{,}82$ пиковая погрешность достигает 9,6%, что существенно влияет на работу алгоритмов управления перегрузками.

4. Обоснована необходимость учета данного эффекта при настройке таймаутов в протоколах QUIC и TCP для минимизации ложных ретрансмиссий и оптимизации использования пропускной способности канала.

Полученные результаты развивают теоретические основы математического моделирования сетей связи и могут быть использованы при создании новых версий транспортных протоколов, а также в системах активного мониторинга качества услуг.

Литература

1. *Choi Y.-H., Kim D., Ko M.J.* ML-Based 5G Traffic Generation for Practical Simulations Using Open Datasets // IEEE Communications Magazine. 2023. Vol. 61, No. 9, pp. 12-20. DOI: 10.1109/MCOM.001.2200679

2. *Ferreira G.O., Ravazzi C., Dabbene F., Calafiore G.* Forecasting network traffic: a survey and tutorial with open-source comparative evaluation // IEEE Access. 2023. DOI: 10.1109/ACCESS.2023.3236261.

3. *Pi Y., Jamin S., Wei Y.* Measuring congestion-induced performance imbalance in Internet load balancing at scale // Computer Networks. 2024. Vol. 240, Art. 110189. DOI: 10.1016/j.comnet.2024.110189.

4. *Sun J., Wo T., Liu X., Ma T.*, … Scalable inter-domain network virtualization // Journal of Network and Computer Applications. 2023. Vol. 218. Art. 103701. DOI: 10.1016/j.jnca.2023.103701

5. *Kaj I., Morozova T.* Retransmission performance in a stochastic geometric cellular network model // Performance Evaluation. 2024. Vol. 165. Art. 102428. DOI: 10.1016/j.peva.2024.102428

6. *Liu Y., Liu L., Jiang T., Chai X.* Transient and steady-state analysis of an M/PH_2/1 queue with catastrophes // Axioms. 2024. Vol. 13, № 10, Art. 716. DOI: 10.3390/axioms13100716.

7. *Hillas L.A., Caldentey R., Gupta V.* Heavy traffic analysis of multi-class bipartite queueing systems under FCFS. Queueing Systems. 2024. Vol. 106, pp. 239-284. DOI: 10.1007/s11134-024-09903-4

8. *Батенков К. А.* Анализ надежности двухполюсных сетей связи на основе метода приведения (редукции) // Известия высших учебных заведений России. Радиоэлектроника. 2025. Т. 28, № 6. С. 56-70. DOI 10.32603/1993-8985-2025-28-6-56-70. EDN HXNUDF.

9. *Батенков К.А., Фокин А.Б.* Анализ надежности телекоммуникационных сетей, поддерживающих механизмы защитного переключения и восстановления для одного основного маршрута // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2023. № 65. С. 58-68. DOI 10.17223/19988605/65/6. EDN XFWQSO.

10. *Thompson Y. L. E., Levine G. M., Chen W., Sahiner B., Li Q., Petrick N., Delfino J. G., Lago M. A., Cao Q., Samuelson F. W.* Applying queueing theory to evaluate wait-time-savings of triage algorithms // Queueing Systems. 2024. Vol. 108, № 3-4, pp. 579-610. DOI: 10.1007/s11134-024-09927-w

11. *Kanavetas O., Logothetis D.* Unreliable service system with strategic customers: effect of abandonments and repair time // Annals of Operations Research. 2024, pp. 1-38. DOI: 10.1007/s10479-024-06297-7

12. *Boelema H.M., Dams D.J.J., O'Reilly M.M., Scheinhardt W.R.W., Taylor P.G.* A stochastic fluid model approach to the stationary distribution of the maximal priority process // Stochastic Models. 2024. DOI: 10.1080/15326349.2024.2375195

13. *Wang D., Wang R.* Transmission Control Protocol (TCP)-Based Delay Tolerant Networking for Space-Vehicle Communications in Cislunar Domain: An Experimental Approach // Sensors. 2025. Vol. 25, № 4, Art. 1136. DOI: 10.3390/s25041136.

14. *Kong C., Song L., Li Y.* Toward enhanced reliability: an efficient method for link-local retransmission in a programmable data plane // Electronics. 2025. Vol. 14, No. 1: 131. DOI: 10.3390/electronics14010131

15. *Vishnevsky V., Vytovtov K., Barabanova E., Semenova O.* Transient Behavior of the MAP/M/1/N Queuing System // Mathematics. 2021. Vol. 9, Art. 2559. DOI: 10.3390/math9202559

16. *Caicedo-Solano N.E., Peña-González D., Vergara D., Machado I.F., Ariza-Echeverri E.A.* Advanced stochastic modeling for series production processes: a Markov chains and queuing theory approach to optimizing manufacturing efficiency // Processes. 2025. Vol. 13, № 11: 3468. DOI: 10.3390/pr13113468

17. *Abramov V.M.* A new criterion for recurrence of Markov chains with an infinitely countable set of states // Theory of Probability and Mathematical Statistics. 2025. Vol. 112, pp. 1-15. DOI: 10.1090/tpms/1224

18. *Kalmulina E.Yu., Zverkina G.A.* Generalised Lorden's inequality and convergence rate estimates for non-regenerative stochastic models // arXiv:2501.18329. 2025. DOI: 10.48550/arXiv.2501.18329

19. *Masanet T., Moyal P.* Perfect sampling of stochastic matching models with reneging // Advances in Applied Probability. 2024. Vol. 56, № 4, pp. 1307-1339. DOI: 10.1017/apr.2023.62

MATHEMATICAL MODELING OF ROUND-TRIP TIME ESTIMATION BIAS IN TELECOMMUNICATION SYSTEMS UNDER NON-STATIONARY LOADS

Kirill A. Batenkov, *Moscow Technical University of Communications and Informatics, Moscow, Russia,* ***pustur@yandex.ru***

Abstract

This paper examines the problem of accurately estimating the round-trip time (RTT) in modern transport layer protocols such as TCP and QUIC, operating under highly dynamic non-stationary traffic conditions. Traditional exponential smoothing algorithms (EWMA), regulated by standards such as RFC 9002, are based on the simplifying assumption of stationarity of queue servicing processes. However, in real telecommunications systems, sudden load surges significantly violate this condition, causing systematic errors in delay measurements. This paper presents a mathematical model based on a classical M/M/1 queueing system, considered in a transient mode. Using a system of Kolmogorov differential equations and the numerical method of matrix exponentials, we obtain precise analytical expressions for the evolution of probability densities representing mixtures of Erlang distributions for the "true" RTT, corresponding to an arbitrary observer, and the "measured" RTT, recorded by the outgoing packet flow. The difference between the distribution of system states at an arbitrary point in time and the distribution observed by outgoing requests in the non-stationary phase is theoretically substantiated and numerically confirmed. This difference causes a significant shift in the mathematical expectation of the measured delay relative to the actual value, primarily during periods of transient queue growth. Numerical modeling of a scenario in which the channel utilization rate jumps from the initial to the final value demonstrated the emergence of a peak in the relative error of the RTT estimate of up to 9.6% during the phase of active request accumulation. The obtained results are of critical practical importance for the development of new adaptive algorithms for configuring retransmission timeouts (PTO/RTO), which minimize the number of false retransmissions and prevent buffer overload. This work combines theoretical analysis and applied communications problems. The study contributes to the development of methods for mathematical modeling of stochastic systems and improving the efficiency of data flow management in next-generation telecommunication networks.

Keywords: *mathematical modeling, round-trip time, M/M/1 system, non-stationary mode, telecommunication networks, QUIC, delay estimation*

References

[1] Y. H. Choi, D. Kim, and M. J. Ko, "ML based 5G traffic generation for practical simulations using open datasets," *IEEE Communications Magazine*, vol. 61, no. 9, pp. 12-20, 2023, doi: 10.1109/MCOM.001.2200679.

[2] G. O. Ferreira, C. Ravazzi, F. Dabbene, and G. Calafiore, "Forecasting network traffic: A survey and tutorial with open source comparative evaluation," *IEEE Access*, 2023, doi: 10.1109/ACCESS.2023.3236261.

[3] Y. Pi, S. Jamin, and Y. Wei, "Measuring congestion induced performance imbalance in Internet load balancing at scale," *Computer Networks,* vol. 240, Art. no. 110189, 2024, doi: 10.1016/j.comnet.2024.110189.

[4] J. Sun, T. Wo, X. Liu, T. Ma, et al., "Scalable inter-domain network virtualization," *Journal of Network and Computer Applications,* vol. 218, Art. no. 103701, 2023, doi: 10.1016/j.jnca.2023.103701.

[5] I. Kaj and T. Morozova, "Retransmission performance in a stochastic geometric cellular network model," *Performance Evaluation,* vol. 165, Art. no. 102428, 2024, doi: 10.1016/j.peva.2024.102428.

[6] Y. Liu, L. Liu, T. Jiang, and X. Chai, "Transient and steady-state analysis of an M/PH?/1 queue with catastrophes," *Axioms*, vol. 13, no. 10, Art. no. 716, 2024, doi: 10.3390/axioms13100716.

[7] L. A. Hillas, R. Caldentey, and V. Gupta, "Heavy traffic analysis of multi-class bipartite queueing systems under FCFS," *Queueing Systems,* vol. 106, pp. 239-284, 2024, doi: 10.1007/s11134-024-09903-4.

[8] K. A. Batenkov, "Two-terminal reliability analysis of telecommunication networks based on reduction method," *Journal of the Russian Universities. Radioelectronics*, vol. 28, no. 6, pp. 56-70, 2025, doi: 10.32603/1993-8985-2025-28-6-56-70. (in Russian).

[9] K. A. Batenkov and A. B. Fokin, "Reliability analysis of telecommunication networks supporting protective switching and recovery mechanisms for one main route," *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitelnaja tehnika i informatika* (Tomsk State University Journal of Control and Computer Science), no. 65, pp. 58-68, 2023, doi: 10.17223/19988605/65/6. (in Russian).

[10] Y. L. E. Thompson, G. M. Levine, W. Chen, B. Sahiner, Q. Li, N. Petrick, J. G. Delfino, M. A. Lago, Q. Cao, and F. W. Samuelson, "Applying queueing theory to evaluate wait time savings of triage algorithms," *Queueing Systems*, vol. 108, no. 3-4, pp. 579-610, 2024, doi: 10.1007/s11134-024-09927-w.

[11] O. Kanavetas and D. Logothetis, "Unreliable service system with strategic customers: Effect of abandonments and repair time," *Annals of Operations Research*, pp. 1-38, 2024, doi: 10.1007/s10479-024-06297-7.

[12] H. M. Boelema, D. J. J. Dams, M. M. O'Reilly, W. R. W. Scheinhardt, and P. G. Taylor, "A stochastic fluid model approach to the stationary distribution of the maximal priority process," *Stochastic Models*, 2024, doi: 10.1080/15326349.2024.2375195.

[13] D. Wang and R. Wang, "Transmission control protocol (TCP) based delay tolerant networking for space vehicle communications in cislunar domain: An experimental approach," *Sensors,* vol. 25, no. 4, Art. no. 1136, 2025, doi: 10.3390/s25041136.

[14] C. Kong, L. Song, and Y. Li, "Toward enhanced reliability: An efficient method for link-local retransmission in a programmable data plane," Electronics, vol. 14, no. 1, Art. no. 131, 2025, doi: 10.3390/electronics14010131.

[15] V. Vishnevsky, K. Vytovtov, E. Barabanova, and O. Semenova, "Transient behavior of the MAP/M/1/N queuing system," *Mathematics*, vol. 9, Art. no. 2559, 2021, doi: 10.3390/math9202559.

[16] N. E. Caicedo-Solano, D. Pe?a-Gonz?lez, D. Vergara, I. F. Machado, and E. A. Ariza-Echeverri, "Advanced stochastic modeling for series production processes: A Markov chains and queuing theory approach to optimizing manufacturing efficiency," *Processes*, vol. 13, no. 11, Art. no. 3468, 2025, doi: 10.3390/pr13113468.

[17] V. M. Abramov, "A new criterion for recurrence of Markov chains with an infinitely countable set of states," *Theory of Probability and Mathematical Statistics*, vol. 112, pp. 1-15, 2025, doi: 10.1090/tpms/1224.

[18] E. Yu. Kalmulina and G. A. Zverkina, "Generalised Lorden's inequality and convergence rate estimates for non-regenerative stochastic models," arXiv preprint arXiv:2501.18329, 2025, doi: 10.48550/arXiv.2501.18329.

[19] T. Masanet and P. Moyal, "Perfect sampling of stochastic matching models with reneging," *Advances in Applied Probability*, vol. 56, no. 4, pp. 1307-1339, 2024, doi: 10.1017/apr.2023.62.

Information about author:

Kirill A. Batenkov, *full professor, Dr. Sc. (Tech.), Professor of the Department of Communication Networks and Switching Systems, Moscow Technical University of Communications and Informatics, Moscow, Russia, orcid.org: 0000-0001-6083-1242*

АНАЛИЗ СТОХАСТИЧЕСКИХ ТАНДЕМНЫХ СИСТЕМ С ПОВТОРНЫМИ ВЫЗОВАМИ

DOI: 10.36724/2072-8735-2026-20-6-20-29

Manuscript received 21 March 2026;
Accepted 25 May 2026

Вишневский Владимир Миронович,
Институт проблем управления им. В. А. Трапезникова РАН, Москва, Россия, ***vishn@inbox.ru***

Семёнова Ольга Валерьевна,
Институт проблем управления им. В. А. Трапезникова РАН, Москва, Россия, ***olga.semenova@ipu.ru***

Ключевые слова: *сети массового обслуживания, тандемная система, повторные вызовы (орбита), MAP-поток, распределение фазового типа, имитационное моделирование, машинное обучение*

В настоящее время одним из важных направлений проектирования и оценки производительности современных телекоммуникационных систем и сетей является анализ математических моделей тандемных систем. Тандемные системы являются математическими моделями различных технических и социальных систем, в частности, являются адекватными моделями широкополосных беспроводных сетей вдоль протяженных транспортных магистралей (автомобильные и железные дороги, нефтяные и газовые трубопроводы и т.д.). Рассматриваемая в настоящей работе тандемная система включает произвольное, но конечное число обслуживающих устройств (фаз) с ограниченным буфером на каждой фазе. Поток заявок (пакетов) на первую фазу является коррелированным MAP-потоком, а случайное время обслуживания описывается фазовым распределением. Если при поступлении заявки на любую фазу буферная память этой фазы заполнена, то заявка отправляется на орбиту (общую память для всех фаз), из которой через случайное время повторно поступает на первую фазу. Оцениваются основные характеристики производительности системы, включая межконцевую задержку (время с момента поступления на первую фазу до момента успешного завершения обслуживания на последней фазе), средние длины очередей на фазах системы и на орбите. Для систем с двумя очередями получены точные формулы для расчета требуемых характеристик. Для систем с произвольным числом фаз использован метод машинного обучения в сочетании с методом имитационного моделирования. Проведены многочисленные численные эксперименты.

Информация об авторах:
Вишневский Владимир Миронович, *доктор технических наук, профессор, главный научный сотрудник, лаборатория "Телекоммуникационных систем", Институт проблем управления им. В. А. Трапезникова РАН, Москва, Россия. ORCID 0000-0001-7373-4847*
Семёнова Ольга Валерьевна, *доктор физико-математических наук, ведущий научный сотрудник, лаборатория "Телекоммуникационных систем", Институт проблем управления им. В. А. Трапезникова РАН, Москва, Россия. ORCID 0000-0001-7846-6212*

Для цитирования:
Вишневский В.М., Семёнова О.В. Анализ стохастических тандемных систем с повторными вызовами // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 20-29.

For citation:
V.M. Vishnevsky, O.V. Semenova, "Analysis of stochastic tandem systems with retrials", *T-Comm*, 2026, vol. 20, no.6, pp. 20-29. (*in Russian*)

Введение

Для моделирования современных систем связи транспортных и других систем широко используются стохастические тандемные системы (многофазные системы массового обслуживания) см., например [1-3]. Такие модели представляют особый интерес для оценки характеристик производительности широкополосных беспроводных сетей вдоль протяженных транспортных магистралей (автомобильные и железные дороги, нефтяные и газовые трубопроводы и т.д.) [4-5]. В связи с этим, исследование тандемных систем, начатое в середине прошлого века, интенсивно продолжается до настоящего времени [6-10]. Следует отметить, что большинство работ посвящено исследованию лишь двухфазных тандемных систем, хотя наибольший практический интерес представляют тандемные системы большой размерности (с числом узлов больше двух). Точное аналитическое решение для тандемных систем большой размерности, особенно в случае коррелированного входного потока и произвольной функции распределения времени обслуживания на фазах не было получено. Разработаны лишь многочисленные приближенные методы оценки производительности таких систем [9-10].

Заметим также, что в мировой литературе, посвященной исследованию стохастических тандемных систем, слабо освещены модели тандемных систем с повторными вызовами. В то же время, механизм повторных заявок является неотъемлемой характеристикой большинства современных телекоммуникационных сетей и систем (например, механизм повторной передачи протокола TCP – Transmission Control Protocol). Данная особенность значительно усложняет математический анализ соответствующих моделей, в отличие от моделей без повторных заявок, поскольку марковские цепи, описывающие поведение систем с повторными заявками, являются пространственно неоднородными. В то же время их изучение важно не только для практических приложений, но и представляет значительный теоретический интерес.

Обзоры ранних (до 2016 года) работ по математическому анализу систем массового обслуживания с повторными заявками можно найти в [11]. Тандемные системы с повторными вызовами исследовались в [1, 12-14]. В большинстве данных работ исследовались двухфазные системы, и предполагалась возможность присоединения к орбите только заявок, заблокированных на первой фазе. Хотя в различных реальных протоколах в вычислительных сетях с гарантированной доставкой пакетов разработан механизм повторной передачи пакетов из общей памяти (орбиты). Поэтому рассматриваемая в данной статье задача анализа характеристик тандемной системы большой размерности с общей орбитой является актуальной. Подобные тандемные системы с общей орбитой рассматривались в работах [13, 14].

В работе [13] для тандемной системы с произвольным конечным числом фаз (станций) предложен метод приближенного расчета среднего времени пребывания заявки в системе и среднего числа посещений орбиты. Для рассмотренной модели не накладывается ограничений на распределение между моментами поступления заявок, времени обслуживания и интервалов между повторными попытками заявки обслужиться. Предполагается, что известна лишь вероятность блокировки на каждой фазе. Исходя из результатов, полученных для двухфазной системы, авторы делают вывод о том, что приближенный метод дает удовлетворительные результаты при относительно низкой интенсивности входного потока.

В статье [14] исследована тандемная система с двумя фазами без буферов и общей орбитой для повторных вызовов с произвольным распределением времени обслуживания на первой фазе и экспоненциальным – на второй. Время между повторными вызовами также распределено по экспоненциальному закону, при этом используется классическая стратегия повторных попыток. Предполагая, что интенсивность повторных вызовов довольно мала, авторы показывают, что масштабированное число заявок на орбите асимптотически близко к диффузионному процессу, что далее используется для приближенной оценки числа заявок на орбите в стационарном режиме. В статье [1] исследуется двухфазная система с орбитой на первой фазе и отдыхами обоих обслуживающих устройств по завершению обслуживания всех заявок в системе.

При проектировании и оценке характеристик современных телекоммуникационных сетей особенно важно учитывать нестационарный и коррелированный характер циркулирующих в них потоков данных. При моделировании учет этих особенностей возможен с использованием математических моделей потоков типа BMAP (Batch Markovian Arrival Process) и MAP (Markovian Arrival Process) [15], которые адекватно описывают потоки в современных вычислительных сетях.

В настоящей статье представлен анализ тандемной системы большой размерности (с числом фаз $N \geq 2$) с общей орбитой, имеющей, в отличие от моделей, исследованных в перечисленных выше работах, более общее предположение о характере входного коррелированного потока и обобщенное фазовое распределение времени обслуживания на фазах системы. Для прогнозирования характеристик производительности тандемной системы большой размерности впервые использована комбинация методов имитационного моделирования и машинного обучения [16].

Статья организована следующим образом. Раздел 1 содержит постановку задачи и описание математической модели. В разделе 2 приведен анализ двухфазной тандемной системы и представлен алгоритм точного расчёта характеристик производительности системы. В разделе 3 для системы с произвольным числом фаз впервые применена комбинация методов машинного обучения и имитационного моделирования для оценки межконцевой задержки и других характеристик. Также проведен сравнительный анализ применения различных методов машинного обучения. Раздел 4 содержит заключение к статье.

1 Постановка задачи и описание математической модели

Рассматривается стохастическая тандемная система, состоящая из $N \geq 2$ фаз, представленных системами массового обслуживания типа MAP/PH/1/K, и общей орбитой (рис. 1).

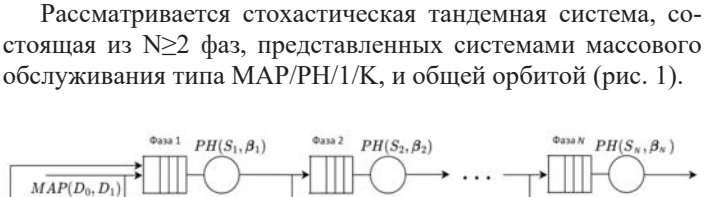


Рис. 1. Стохастическая тандемная система с общей орбитой

Число мест для ожидания в буфере на фазе i ограничено числом K_i, $i=\overline{1,N}$. Заявки поступают извне на первую фазу согласно марковскому входному потоку (MAP-потоку), который описывается матрицами D_0 и D_1 и средней интенсивностью поступления заявок $\lambda=\theta D_1\mathbf{e}$. В последнем равенстве вектор-строка θ находится как единственное решение системы уравнений $\boldsymbol{\theta}(D_0+D_1)=\mathbf{0},\boldsymbol{\theta}\mathbf{e}=1$, где $\mathbf{e}=(1,1,...,1)^T$, $\mathbf{0}=(0,...,0)$.

Время обслуживания заявки на k-й фазе имеет распределение фазового типа (PH, Phase-type), которое задается парой $(\boldsymbol{\beta}_i,S_i),i=\overline{1,N}$, где $\boldsymbol{\beta}_k=(\beta_k^{(1)},\ldots,\beta_k^{(M_k)})$ – стохастический вектор-строка, а S_i – субгенератор. Интенсивность обслуживания определяется как $\mu_k=-[\boldsymbol{\beta}_k S_k^{-1}\mathbf{e}]^{-1}$. Среднее время обслуживания – $b_k=\mu_k^{-1},k=\overline{1,N}$. Более подробное описание MAP-потока как частного случая группового потока BMAP и фазового распределения и их свойства можно найти в [15].

Если при поступлении заявки на i-ю фазу свободные места для ожидания в буфере отсутствуют, заявка переходит на орбиту для повторного обслуживания на первой фазе. Время пребывания повторной заявки на орбите экспоненциально распределено с параметром α_i, где i – число заявок на орбите. Предполагается, что $\alpha_0=0,\alpha_i\to\infty$ при $i\to\infty$. Такая зависимость включает, в частности, линейную стратегию повторных вызовов $\alpha_i=i\alpha,\ \alpha>0$.

Задача состоит в отыскании основных характеристик производительности, включая среднее время пребывания заявок в системе (межконцевая задержка) и среднее число заявок на каждой фазе и орбите.

2 Анализ двухфазной тандемной системы с повторными вызовами и общей орбитой

В данном разделе рассмотрим частный случай тандемной системы, описанной в предыдущем разделе, с количеством фаз N=2. Ее функционирование может быть описано случайным процессом, принадлежащим классу асипмтотически-квазитеплицевых цепей Маркова [15].

Пусть в момент времени t:

– i_t – число заявок на орбите, $i\ge 0$,

– j_t – число заявок на первой фазе, $j_t=\overline{0,K_1}$,

– n_t – число заявок на второй фазе, $n_t=\overline{0,K_2}$,

– ν_t – состояние управляющего процесса MAP $\nu_t=\overline{0,W}$,

– $m_t^{(k)}$ – состояние управляющего процесса PH на k-й фазе, $m_t^{(k)}=\overline{1,M^{(k)}}$, $k=\overline{1,2}$.

Случайный процесс $\xi_t=\{i_t,j_t,n_t,\nu_t,m_t^{(1)},m_t^{(2)}\},t\ge 0$ является регулярной неприводимой цепью Маркова с пространством состояний

$$\Omega=\{(i,j,n,\nu),i\ge 0,j=0,n=0,\nu=\overline{0,W}\}\cup$$
$$\{(i,j,n,\nu,m^{(1)}),i\ge 0,j=\overline{0,K_1},n=0,\nu=\overline{0,W},m^{(1)}=\overline{1,M^{(1)}}\}\cup$$
$$\{(i,j,n,\nu,m^{(2)}),i\ge 0,j=0,n=\overline{0,K_2},\nu=\overline{0,W},m^{(2)}=\overline{1,M^{(2)}}\}\cup$$
$$\{(i,j,n,\nu,m^{(1)},m^{(2)}),i\ge 0,j=\overline{0,K_1},n=\overline{0,K_2},\nu=\overline{0,W},$$
$$m^{(1)}=\overline{1,M^{(1)}},m^{(2)}=\overline{1,M^{(2)}}\}.$$

Упорядочим состояния из множества $\boldsymbol{\Omega}$ в лексикографическом порядке согласно возрастанию значений счетной компоненты i_t. Интенсивности переходов между состояниями формируют инфинитезимальный генератор Q, структура которого будет описана далее.

Пусть I – единичная матрица, O – нулевая матрица, $\delta(i,j)$ – символ Кронекера, $\oplus$ и $\otimes$ – символы Кронекеровой суммы и произведения матриц, соответственно. Обозначим через $\mathbf{a}=(a_1,...,a_n)$ и будем использовать следующие вспомогательные матрицы: $diag\{\mathbf{a}\}$ – диагональная матрица размера $n\times n$ с диагональными элементами, задаваемыми вектором $\mathbf{a}$, $diag^+(\mathbf{a})$ ($diag^-(\mathbf{a})$) – матрица размера $n\times n$, имеющая ненулевые элементы с индексами $(i,i+1)$, равные a_i для $i=\overline{1,n-1}$ (соответственно, с индексами $(i-1,i)$, равные a_i для $i=\overline{2,n}$).

Лемма 1. Инфинитезимальный генератор Q цепи Маркова $\xi_t,t\ge 0$ имеет блочную трехдиагональную структуру

$$Q=\begin{pmatrix} Q_{0,0} & Q_{0,1} & O & O & O & \ldots \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & O & O & \ldots \\ O & Q_{2,1} & Q_{2,2} & Q_{2,3} & O & \ldots \\ O & O & Q_{3,2} & Q_{3,3} & Q_{3,4} & \ldots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \qquad (1)$$

где матрицы $Q_{i,j}$ также имеют блочную структуру вида $Q_{i,j}=(Q_{i,i'}(j,j'))_{i'\ge 0,j'=\overline{0,K_1}}$ с матрицами $Q_{i,i'}(j,j')$, определяемыми равенствами

$$Q_{i,i-1}(j,j-1)=O,i\ge 1,j=\overline{1,K_1},\ Q_{i,i-1}(j,j)=O,i\ge 1,j=\overline{0,K_1},$$
$$Q_{i,i-1}(0,1)=\alpha_i diag\{I_W\otimes\boldsymbol{\beta}_1,\underbrace{I_W\otimes\boldsymbol{\beta}_1\otimes I_{M_2}}_{K_2}\},i\ge 1,j=\overline{0,K_1-1},$$
$$Q_{i,i-1}(j,j+1)=\alpha_i I_{WM_1(1+K_2M_2)},\quad i\ge 1,j=\overline{1,K_1-1},$$
$$Q_{i,i}(1,0)=diag^+\{I_W\otimes\mathbf{S}_0^{(1)}\otimes\boldsymbol{\beta}_2,\underbrace{I_W\otimes\mathbf{S}_0^{(1)}\otimes I_{M_2}}_{K_2-1}\},\ i\ge 0,$$
$$Q_{i,i}(j,j-1)=diag^+\{I_W\otimes\mathbf{S}_0^{(1)}\boldsymbol{\beta}_1\otimes\boldsymbol{\beta}_2,\underbrace{I_W\otimes\mathbf{S}_0^{(1)}\boldsymbol{\beta}_1\otimes I_{M_2}}_{K_2-1}\},$$
$$i\ge 0,j=\overline{2,K_1},$$
$$Q_{i,i}(0,0)=diag\{D_0,\underbrace{D_0\oplus S_2}_{K_2}\}+diag^-\{I_W\otimes\mathbf{S}_0^{(2)},\underbrace{I_W\otimes\mathbf{S}_0^{(2)}\boldsymbol{\beta}_2}_{K_2-1}\}-$$
$$-\alpha_i I,i\ge 0,$$
$$Q_{i,i}(j,j)=diag\{D_0\oplus S_1,\underbrace{D_0\oplus S_1\oplus S_2}_{K_2}\}-\left[1-\delta(j,K_1)\right]\alpha_i I+$$
$$+diag^-\{I_{WM_1}\otimes\mathbf{S}_0^{(2)},\underbrace{I_{WM_1}\otimes\mathbf{S}_0^{(2)}\boldsymbol{\beta}_2}_{K_2-1}\},i\ge 0,j=\overline{1,K_1},$$
$$Q_{i,i}(0,1)=diag\{D_1\otimes\boldsymbol{\beta}_1,\underbrace{D_1\otimes\boldsymbol{\beta}_1\otimes I_{M_2}}_{K_2}\},\quad i\ge 0,$$

$$Q_{i,i}(j,j+1)=diag\{D_1\otimes I_{M_1},\underbrace{D_1\otimes I_{M_1}\otimes I_{M_2}}_{K_2}\},\ i\ge 0,\ j=\overline{1,K_1-1},$$

$$Q_{i,i+1}(1,0)=diag\{O_W\otimes\mathbf{S}_0^{(1)},\underbrace{O_W\otimes\mathbf{S}_0^{(1)}\otimes I_{M_2}}_{K_2},I_W\otimes\mathbf{S}_0^{(1)}\otimes I_{M_2}\},i\ge 0,$$

$$Q_{i,i+1}(j,j-1)=diag\{O_{WM_1(1+(K_2-1)M_2)},I_W\otimes\mathbf{S}_0^{(1)}\boldsymbol{\beta}_1\otimes I_{M_2}\},$$

$$i\ge 0,\ j=\overline{2,K_1},$$

$$Q_{i,i+1}(j,j)=O,\ i\ge 0,\ j=\overline{1,K_1-1},$$

$$Q_{i,i+1}(K_1,K_1)=diag\{D_1\otimes I_{M_1},\underbrace{D_1\otimes I_{M_1M_2}}_{K_2}\},\ i\ge 0,$$

$$Q_{i,i+1}(j,j+1)=O,\ i\ge 0,\ j=\overline{1,K_1-1}.$$

Остальные матрицы $Q_{i,i'}(j,j'),i'\ge 0,j'=\overline{0,K_1}$, не перечисленные выше, являются нулевыми, $\mathbf{S}_0^{(k)}=-S_k\mathbf{e}$.

Доказательство. Блочная структура вида (1) следует из анализа возможных переходов цепи Маркова $\xi_t,t\ge 0$ между состояниями. С целью сокращения изложения опишем лишь характеристику этих блоков.

Блоки $Q_{i,i-1}(j,j+1)$ описывают интенсивности переходов цепи Маркова $\xi_t,t\ge 0$, которые сопровождаются успешным переходом заявки с орбиты на первую фазу. Если первая фаза пуста ($j=0$), эти переходы сопровождаются выбором начальной фазы процесса обслуживания на первой фазе в соответствии с вектором вероятностей $\boldsymbol{\beta}_1$.

Блоки $Q_{i,i}(j,j')$ описывают интенсивности переходов цепи $\xi_t,t\ge 0$, которые не приводят к изменению числа заявок на орбите. Пусть $j'=j-1$. Если $j=1$, такие переходы сопровождаются завершением обслуживания заявки на первой фазе (интенсивности переходов его задаются матрицей $\mathbf{S}_0^{(1)}$) и переходом этой заявки на вторую фазу. В случае $j>1$ очередная заявка поступает на обслуживание на первой фазе из буфера. В случае $j'=j$ вид блочных матриц $Q_{i,i}(j,j)$ зависит от значения j и устанавливается путем анализа возможных переходов процесса $\xi_t,t\ge 0$, которые не сопровождаются событиями, изменяющими значение j.

Матрицы $Q_{i,i+1}(j,j')$ описывают интенсивности переходов цепи Маркова $\xi_t,t\ge 0$, которые приводят к увеличению числа заявок на орбите на единицу в случае поступления заявки извне, когда нет мест для ожиданий на первой фазе.

Матрицы $Q_{i,i+1}(j,j-1),j=\overline{1,K_1}$, описывают интенсивности переходов процесса $\xi_t,t\ge 0$ в случае, когда нет мест для ожидания на второй фазе в момент выхода заявки с первой фазы (в момент завершения ее обслуживания).

Следствие 1. Цепь Маркова $\xi_t,t\ge 0$ является асимптотически квазитеплицевой цепью Маркова [15].

Доказательство. Пусть

$$T_i=diag\left\{\left|\left(Q_{i,i}\right)_{1,1}\right|,\left|\left(Q_{i,i}\right)_{2,2}\right|,...,\left|\left(Q_{i,i}\right)_{m,m}\right|\right\},$$

где m – размерность матрицы $Q_{i,i},i\ge 0$. Покажем, что пределы

$$Y_0=lim_{i\to\infty}T_i^{-1}Q_{i,i-1},Y_1=lim_{i\to\infty}T_i^{-1}Q_{i,i}+I,Y_2=lim_{i\to\infty}T_i^{-1}Q_{i,i+1}$$

существуют, а матрица $Y_0+Y_1+Y_2$ стохастическая. Заметим, что матрица T_i, а также $Q_{i,i}(K_1,K_1-1),Q_{i,i}(K_1,K_1)$, $Q_{i,i+1}(K_1,K_1)$, $Q_{i,i+1}(K_1,K_1-1)$ не зависят от i и K_1, поэтому можно ввести обозначения:

$$A=Q_{i,i}(K_1,K_1-1),B=Q_{i,i}(K_1,K_1)+Q_{i,i+1}(K_1,K_1),$$
$$C=Q_{i,i+1}(K_1,K_1-1). \qquad (2)$$

С помощью алгебраических преобразований можно показать, что

$$Y_0=diag^+\{\mathbf{e}^T\otimes I_{WM_1(1+K_2M_2)}\},$$

$$Y_1=\left(\begin{array}{c|c}O & O\\ \hline O & \begin{array}{cc}T^{-1}A & R^{-1}C+I\end{array}\end{array}\right),\ Y_2=\left(\begin{array}{c|c}O & O\\ \hline O & T^{-1}B\end{array}\right),$$

где T – диагональная матрица, сформированная модулями диагональных элементов матрицы C. Нетрудно проверить, что матрица $Y_0+Y_1+Y_2$ стохастическая. В силу этого свойства согласно [15] справедливо утверждение следствия.

Далее условие эргодичности цепи Маркова $\xi_t,t\ge 0$ можно сформулировать в терминах матриц Y_0,Y_1,Y_2. Следуя [15], вначале получим выражение для производящей функции $Y(z)$ этих матриц.

Лемма 2. Матричная производящая функция $Y(z)=Y_0+Y_1z+Y_2z^2$ имеет вид

$$Y(z)=diag^+\{\mathbf{e}^T\otimes I_{WM_1(1+K_2M_2)}\}+$$
$$+\left(\begin{array}{c|c}O & O\\ \hline O & \begin{array}{cc}T^{-1}Az & z\left[T^{-1}(C+Bz)\right]+zI\end{array}\end{array}\right),$$

где матрицы A,B,C определены в (2).

Теорема 1. 1) Цепь Маркова $\xi_t,t\ge 0$ эргодична, если справедливо следующее неравенство:

$$\lambda<\frac{1}{2}\left\{\left[y_0+\sum_{n=1}^{K_2-1}y_n\left(I_{M_1}\otimes\mathbf{e}_{M_2}\right)\right]\mathbf{S}_0^{(1)}+y_{K_2}\left(\mathbf{e}_{M_1}\otimes I_{M_2}\right)\mathbf{S}_0^{(2)}\right\}, \qquad (3)$$

где вектор $y=\left(y_0,y_1,\ldots,y_{K_2}\right)$ – единственное решение системы уравнений

$$yV=0,ye=1, \qquad (4)$$

а матрица V имеет вид

$$V = diag^{+}\{\mathbf{S}_0^{(1)}\boldsymbol{\beta}_1 \otimes \boldsymbol{\beta}_2, \underbrace{\mathbf{S}_0^{(1)}\boldsymbol{\beta}_1 \otimes I_{M_2}}_{K_2-1}\} +$$
$$+ diag^{-}\{I_{M_1} \otimes \mathbf{S}_0^{(2)}, \underbrace{I_{M_1} \otimes \mathbf{S}_0^{(2)}\boldsymbol{\beta}_2}_{K_2-1}\} +$$
$$+ diag\{S_1, \underbrace{S_1 \otimes S_2}_{K_2-1}, S_1 \otimes S_2 + \mathbf{S}_0^{(1)}\boldsymbol{\beta}_1 \otimes I_{M_2}\}.$$

2) цепь Маркова $\xi_t, t \geq 0$ не эргодична, если справедливо неравенство (3), взятое с противоположным знаком.

Доказательство. Матрица $Y(1)$ приводима. Обозначим через $Y^{K_2}(1)$ ее нормальную форму (см., например, [15]):

$$Y^{K_2}(z) = diag^{-}\{\mathbf{e}^T \otimes I_{WM_1(1+K_2M_2)}\} +$$
$$+ \begin{pmatrix} z\left[T^{-1}(C+Bz)\right]+zI & T^{-1}Az & O \\ \hline O & & O \end{pmatrix}.$$

Заметим, что единственным неприводимым стохастическим блоком на диагонали данной матрицы является блочная структура вида

$$\tilde{Y}(z) = \begin{pmatrix} z\left[T^{-1}(C+Bz)\right]+zI & T^{-1}A \\ I_{WM_1(1+K_2M_2)} & O \end{pmatrix}.$$

Из [15] (Теорема 3.14) следует, что цепь Маркова $\xi_t, t \geq 0$ эргодична, если

$$\frac{d}{dz}[\det(zI - \tilde{Y}(z))]\Big|_{z=1} > 0. \qquad (5)$$

На основании блочной структуры матрицы $\tilde{Y}(z)$, определитель в последнем неравенстве можно свести к следующему виду:

$$\det(zI - \tilde{Y}(z)) = \det(zT^{-1})\det[-z(C+Bz) - A]. \qquad (6)$$

При $z=1$ второй определитель в правой части (6) обращается в ноль в силу свойства инфинитезимального генератора. Заметим, что $\det(zT^{-1}) > 0$ при $z=1$, поэтому выражение (5) принимает вид

$$\frac{d}{dz}[\det(-z(C+Bz) - A)]\Big|_{z=1} > 0,$$

который, в свою очередь, можно свести к неравенству

$$\mathbf{x}(C+2B)\mathbf{e} < 0, \qquad (7)$$

где $\mathbf{x}$ - единственное решение системы уравнений

$$\mathbf{x}(C+B+A) = 0, \mathbf{x}\mathbf{e} = 1. \qquad (8)$$

Рассмотрим далее вектор $\mathbf{x}$ в виде

$$\mathbf{x} = (\boldsymbol{\theta} \otimes \mathbf{y}_0, \boldsymbol{\theta} \otimes \mathbf{y}_1, \ldots, \boldsymbol{\theta} \otimes \mathbf{y}_{K_2}), \qquad (9)$$

где вектор $\mathbf{y}_0$ имеет размерность M_1, а векторы $\mathbf{y}_1, \ldots, \mathbf{y}_{K_2}$ – размерность M_1M_2. Далее подставим вектор $\mathbf{x}$ вида (9) и выражения (2) в (8). С учетом того, что $\boldsymbol{\theta}(D_0 + D_1) = 0$, $\boldsymbol{\theta}\mathbf{e} = 1$ после преобразований получим (4) для вектора $\mathbf{y} = (\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{K_2})$. Таким образом, система (8) для вектора $\mathbf{x}$ сводится к (4) для вектора $\mathbf{y}$.

Далее рассмотрим неравенство (7). Подставляя вектор $\mathbf{x}$ в (9), матрицы B, C, определенные соотношением (2), в (7) с учетом $\boldsymbol{\theta}D_1\mathbf{e} = \lambda$ после алгебраических преобразований получим неравенство (3), эквивалентное (7). Таким образом, утверждение 1 теоремы доказано. Утверждение 2 доказывается аналогично с помощью [15] (Теорема 3.14).

Из утверждения теоремы 1 следует, что в случае двухфазной тандемной системы без буферной памяти (с параметрами $K_1 = K_2 = 1, M_k = 1, S_k = -\mu_k, \boldsymbol{\beta}_k = 1, k = 1,2$) неравенство (3) принимает вид $\lambda < \frac{\mu_1\mu_2}{\mu_1 + \mu_2}$.

Получим далее стационарное распределение вероятностей состояний двухфазной тандемной системы. Обозначим через $\mathbf{p}_i, i \geq 0$, векторы стационарных вероятностей цепи Маркова $\xi_t, t \geq 0$. Здесь индекс i обозначает число заявок на орбите. Зная структуру инфинитезимального генератора цепи Маркова $\xi_t, t \geq 0$, можно применить матрично-аналитический подход для расчета векторов $\mathbf{p}_i, i \geq 0$ [15]:

1. Решаем матричное уравнение $G = Y(G)$ с помощью итерационной схемы $G^{(n+1)} = Y(G^{(n)}), n \geq 1$ с начальным условием $G^{(0)} = O$.

2. Вычисляем матрицы $G_{i_0-1}, G_{i_0-2}, \ldots, G_0$ с помощью обратной рекурсии

$$G_i = (-Q_{i+1,i+1} - Q_{i+1,i+2}G_{i+1})^{-1}Q_{i+1,i} \text{ для } i = i_0 - 1, i_0 - 2, \ldots, 0$$

с граничным условием $G_i = G, i \geq i_0$, где i_0 - неотрицательное целое число, такое, что для некоторого положительного ε (точности вычисления) выполняется неравенство $\| G_{i_0} - G \| < \varepsilon$.

3. Вычисляем матрицы $\bar{Q}_{i,i} = Q_{i,i} + Q_{i,i+1}G_i$, $\bar{Q}_{i,i+1} = Q_{i,i+1}$, где $G_i = G, i \geq i_0$.

4. Вычисляем матрицы $F_i, i \geq 1$ с помощью рекурсии

$$F_0 = I, F_i = F_{i-1}\bar{Q}_{i-1,i}(-\bar{Q}_{i,i})^{-1}, i \geq 1.$$

5. Вычисляем вектор $\mathbf{p}_0$ как единственное решение системы

$$\mathbf{p}_0(-\bar{Q}_{0,0}) = 0, \mathbf{p}_0\sum_{i=0}^{\infty}F_i\mathbf{e} = 1.$$

6. Вычисляем векторы $\mathbf{p}_i = \mathbf{p}_0F_i, i \geq 1$.

Вычислив стационарное распределение $\mathbf{p}_i, i \geq 0$, можно получить ряд характеристик производительности рассматриваемой системы и применить закон Литтла для расчета среднего времени пребывания во всей системе:

1. Стационарные вероятности числа заявок на орбите $p_i = \mathbf{p}_i\mathbf{e}, i \geq 0$.

2. Среднее число мест, занятых повторными заявками

$$L = \sum_{i=1}^{\infty} i\mathbf{p}_i\mathbf{e}.$$

3. Совместное распределение

$$q_i(j,n), i \geq 0, j = \overline{0,K_1}, n = \overline{0,K_2}$$

числа заявок на орбите, на первой и на второй фазах. Здесь $q_i(j,n)$ - вероятность того, что в системе имеется i повторных заявок, на первой фазе ожидают j заявок, а на второй фазе – n заявок. Обозначим через $R = M_1(1+M_2K_2)$. Вероятности $q_i(0,0)$, $q_i(j,0)$, $q_i(0,n)$, $q_i(j,n)$ вычисляются по формулам:

$$q_i(0,0) = \mathbf{p}_i \begin{pmatrix} \mathbf{e}_W \\ 0^T_{W(M_2K_2+RK_1)} \end{pmatrix}, i \geq 0,$$

$$q_i(0,n) = \mathbf{p}_i \begin{pmatrix} 0^T_{W[1+M_2(n-1)]} \\ \mathbf{e}_{WM_2} \\ 0^T_{W[M_2(K_2-n)+RK_1]} \end{pmatrix}, i \geq 0, n = \overline{1,K_2},$$

$$q_i(j,0) = \mathbf{p}_i \begin{pmatrix} 0^T_{W[1+M_2K_2+R(j-1)]} \\ \mathbf{e}_{WM_1} \\ 0^T_{W[M_1M_2K_2+R(K_1-j)]} \end{pmatrix},$$

$$q_i(j,n) = \mathbf{p}_i \begin{pmatrix} 0^T_{W[1+M_2K_2+R(j-1)+M_1(1+M_2(n-1))]} \\ \mathbf{e}_{WM_1M_2} \\ 0^T_{W[M_1M_2(K_2-n)+R(K_1-j)]} \end{pmatrix},$$

$$i \geq 0, j = \overline{1,K_1}, n = \overline{1,K_2}.$$

4. Стационарные вероятности состояний первой фазы:

$$q_j^{(1)} = \sum_{i=0}^{\infty} p_i \sum_{n=0}^{K_2} q_i(j,n), \quad j = \overline{0,K_1}.$$

5. Среднее число заявок, ожидающих на первой фазе:

$$L_1 = \sum_{j=1}^{K_1} j q_j^{(1)}.$$

6. Стационарные вероятности состояний второй фазы:

$$q_n^{(2)} = \sum_{i=0}^{\infty} p_i \sum_{j=0}^{K_1} q_i(j,n), \quad n = \overline{0,K_2}.$$

7. Среднее число заявок, ожидающих на второй фазе:

$$L_2 = \sum_{n=1}^{K_2} n q_n^{(2)}.$$

8. Межконцевая задержка: $W = (L + L_1 + L_2)\lambda^{-1}$.

Для систем, имеющих практическое применение в компьютерных сетях, необходимо, как правило, исследовать модели с более, чем двумя узлами (фазами). Отметим, что результаты, полученные в данном разделе, имеют не только теоретическое значение, но и будут использованы в следующем разделе для верификации имитационной модели, описывающей функционирование стохастической тандемной системы с произвольным числом фаз ($N > 2$).

3 Анализ тандемных систем с произвольным числом фаз методами машинного обучения

В данном разделе для исследования тандемной системы с повторными вызовами и произвольным числом фаз впервые предложено использование комбинации методов имитационного моделирования и машинного обучения. Имитационная модель позволяет получить набор данных (датасет) о характеристиках тандемной системы, который далее используется для прогнозирования характеристик производительности тандемной системы с использованием методов машинного обучения.

3.1 Генерация синтетического набора данных

Для применения методов машинного обучения [16], на первом этапе необходимо сгенерировать обучающий набор данных, включающий в себя входные параметры тандемной системы с произвольным числом фаз:

– матрицы D_0, D_1, описывающие MAP-поток;

– ч сло фаз N;

– матрицы (S_i, $\boldsymbol{\beta}_i$), характеризующие фазовое распределение, и размер буфера K_i на i-й фазе, $i = \overline{1,N}$;

– максимальное количество мест на орбите K_{orbit} и скорость α поступления заявок,

а также характеристики производительности тандемной системы (межконцевая задержка, среднее количество заявок в системе и т.д.), вычисленных с помощью других методов (аналитический расчет, имитационное моделирование). В данном примере будем рассматривать лишь одну характеристику – межконцевую задержку.

Дискретно-событийная модель реализована на языке программирования C++. Модуль машинного обучения имеет две основные структурные части: предобработка данных и обучение моделей. Каждая из них реализована на языке программирования Python. Задача скрипта обработки данных – считать и преобразовать эти данные в числовой формат, пригодный для обработки библиотеками машинного обучения. Из-за различий в параметрах обучения и используемых библиотеках у каждого метода машинного обучения есть свой собственный скрипт обучения.

Для обучения машинных моделей необходим набор данных, полученных с помощью имитационной модели тандемной системы с большим количеством наборов данных $(\mathbf{X}, \mathrm{y})$, где $\mathbf{X}$ соответствует входным параметрам программы имитационного моделирования, а y - прогнозируемый искомый параметр (межконцевая задержка). Набор данных сгенерирован на основе возможных комбинаций входных параметров, включая: $N = \overline{2,15}$; размер матриц, описывающих MAP и PH - от 1 до 16; $K_i = \overline{0,32}$ для $i = \overline{1,N}$; размер орбиты $K_{orbit} = \overline{0,512}$. Заметим, что для имитационной модели размер

орбиты должен быть ограничен.

С целью представления более разнообразного набора данных использованы два метода выбора недиагональных значений элементов матриц D_0, D_1, S_i, $i = \overline{1, N}$ и интенсивности α поступления заявок с орбиты на первую фазу:

Способ 1. Недиагональные элементы матрицы D_0 и элементы матрицы D_1 генерируются как непрерывные случайные величины, равномерно распределенные на отрезке [1;1024]. Недиагональные элементы матриц S_i, интенсивности переходов $\mathbf{S}_i^{(0)}$ в поглощающее состояние и интенсивность повторных попыток на орбите α генерируются как непрерывные случайные величины, равномерно распределенные на отрезке [2;2048].

Способ 2. Недиагональные элементы матриц D_0, D_1 генерируются как e^p; недиагональные элементы матриц S_i, интенсивности переходов $\mathbf{S}_i^{(0)}$ в поглощающее состояние и интенсивность повторных заявок α генерируются как $2e^p$, где p – непрерывная случайная величина, равномерно распределенная на отрезке [-2,5; 2,5].

Диагональные же элементы матриц D_0, S_i задаются элементами векторов $-(D_0 + D_1)\mathbf{e}$ и $-(S_i\mathbf{e} + \mathbf{S}_i^{(0)})$, соответственно. Далее входные параметры масштабируются посредством деления элементов матриц D_0, D_1, S_i и интенсивности повторных попыток на орбите α на коэффициент, равный λ (среднее число заявок, поступающих в единицу времени) с целью, чтобы все входные наборы данных имели одинаковую интенсивность входного потока, равную 1.

Данные датасета получены с использованием имитационной модели тандемной системы: *Dataset300k* - набор данных для обучения, содержащих 200 тыс. элементов $(\mathbf{X}, \mathrm{y})$ (способ 1), дополненных 100 тыс. элементов, сгенерированных методом 2, и *Test60k* - набор данных для тестирования, содержащий из 60 тыс. элементов $(\mathbf{X}, \mathrm{y})$ (способ 2).

С целью сокращения вычислительной сложности на конечном этапе подготовки данных предусмотрена процедура уменьшения размерности MAP-потоков с использованием аппроксимации, описанной в [18].

В следующем разделе проведем сравнительный анализ различных методов машинного обучения: градиентного бустинга, случайного леса и искусственных нейронных сетей [16]. Большая часть обучения методами градиентного бустинга и случайного леса выполнено с помощью библиотеки scikit-learn – готовой Python-библиотеки для машинного обучения. Для метода нейронных сетей использована Python-библиотека PyTorch.

3.2 Градиентный бустинг

Градиентный бустинг – это ансамблевый метод машинного обучения, который последовательно создаёт набор слабых с точки зрения прогноза требуемых значений моделей, комбинируя их в единую более сильную прогностическую модель.

Ансамбли – это алгоритмы, сочетающие сразу несколько моделей для исправления ошибок, допущенных предыдущими моделями.

Для решения задачи прогнозирования межконцевой задержки стохастической тандемной системы будем обучать модели градиентного бустинга с параметрами, заданными по умолчанию, скоростью обучения 0,01, методом наименьших квадратов для оценки функции потерь и 10000 итераций обучения. Рассмотрим также два варианта: с логарифмическим преобразованием искомого параметра (межконцевой задержки) и без него, что предусматривается возможностями метода и позволяет выбрать способ обучения, дающий более точный результат. Из результатов обучения, представлены в таблице 1, следует, что лучший результат дает опция логарифмического преобразования искомого параметра $\ln(\mathbf{y})$.

Таблица 1

Результаты оценки моделей методом градиентного бустинга

Набор данных	Обучение без логарифмического преобразования		Обучение с логарифмическим преобразованием	
	Score	σ	Score	σ
Dataset300k	0,5988	1,8435	0,0790	0,1351
Test60k	0,5659	1,8155	0,0900	0,2021

3.3 Случайный лес

Случайный лес – это ещё один популярный алгоритм машинного обучения, который так же, как и градиентный бустинг, относится к классу ансамблевых методов. Метод предполагает создание множества отдельных независимых «решающих деревьев», что позволяет получать более точные результаты и, в то же время, снижать тенденцию к переобучению модели.

Результаты прогнозирования межконцевой задержки методом случайного леса представлены в таблице 2. Как и в случае применения градиентного бустинга, результаты получены с использованием логарифмического преобразования искомого параметра (межконцевой задержки), а также без него.

Сравнивая результаты таблиц 1-2, можно отметить более высокую тенденцию к переобучению при использовании метода случайного леса, поскольку на обучающей выборке точность модели случайного леса (4,6%) выше, чем у градиентного бустинга (7,9%), но хуже на тестовой выборке данных Test60k.

Существенным недостатком метода случайного леса является значительно больший объем требуемой памяти (таблица 3). Например, даже при количестве деревьев 100, что является значением по умолчанию, модели требуется 2,6 ГБ памяти. Это объясняется тем, что по умолчанию библиотека Scikit-learn не предполагает ограничения на глубину деревьев решений (параметр $\mathrm{maxDepth} = 0$). Следовательно, для больших обучающих наборов данных в нашем случае деревья решений будут разрастаться до максимально возможного размера, что приводит к большому размеру результирующей модели. Чтобы уменьшить размер модели, следует обучать ее с параметром $\mathrm{maxDepth} > 0$ за счет снижения точности.

Таблица 2

Результаты оценки моделей методом случайного леса

Набор данных	Обучение без логарифмического преобразования Число деревьев			
	100		300	
	Score	σ	Score	σ
Dataset300k	0,0551	0,1603	0,0551	0,1587
Test60k	0,1176	0,2693	0,1181	0,2703
	500		800	
	Score	σ	Score	σ
Dataset300k	0,0551	0,1580	0,0551	0,1579
Test60k	0,1180	0,2694	0,1177	0,2686
	Обучение с логарифмическим преобразованием Число деревьев			
	100		300	
	Score	σ	Score	σ
Dataset300k	0,0467	0,0914	0,0458	0,0897
Test60k	0,1097	0,2357	0,1089	0,2344
	500	800		
	Score	σ	Score	σ
Dataset300k	0,0456	0,0893	0,0455	0,0891
Test60k	0,1085	0,2343	0,1085	0,2343

Таблица 3

Результаты оценки моделей случайного леса, обученных с различными значениями maxDepth

Макс. глубина maxDepth	Набор данных				Объём памяти
	Dataset300k		Test60k		
	Score	σ	Score	σ	
4	0,5205	0,5744	0,4781	0,5450	243 KB
8	0,2601	0,3397	0,2419	0,3417	3,49 MB
16	0,1153	0,1429	0,1399	0,2500	209MB
32	0,0476	0,0905	0,1098	0,2358	2,34 GB
64	0,0467	0,0915	0,1094	0,2350	2,60 GB
0	0,0467	0,0914	0,1097	0,2357	2,6 GB

3.4 Нейронные сети

В современной практике машинного обучения искусственные нейронные сети занимают доминирующее положение, демонстрируя высокую эффективность в широком спектре прикладных задач. Несмотря на то, что данный метод известен довольно давно, его масштабное внедрение стало возможным лишь в последние десятилетия благодаря существенному росту вычислительной мощности оборудования, оптимизированного для выполнения высокопроизводительных операций. Это позволило эффективно обучать глубокие нейронные сети с большим числом параметров на масштабных наборах данных, что, в свою очередь, обеспечило превосходство методов, базирующихся на нейронных сетях, над альтернативными методами в задачах прогнозирования по критерию точности.

Структура нейронной сети, используемой для решения задачи прогнозирования межконцевой задержки в тандемной системе с повторными вызовами, представлена на рис. 2.

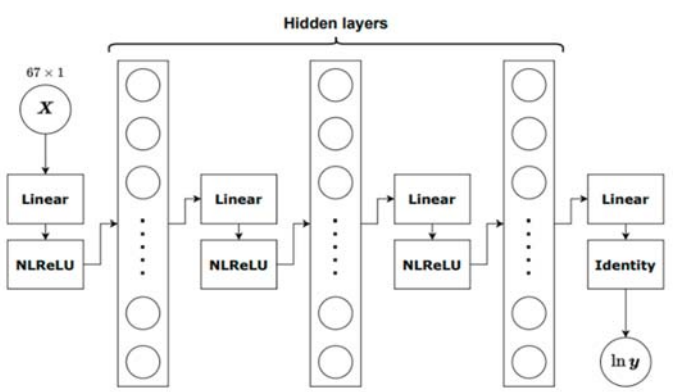


Рис. 2. Структура модели нейронной сети

Сеть содержит три скрытых слоя, каждый из которых состоит из 1024 нейронов. Изначально используется стандартная функция активации ReLU между каждым скрытым слоем, однако во время обучения возникает проблема: значение потерь становится очень высоким (взрывной градиент), что делает модель необучаемой. Простое решение этой проблемы состоит в замене функции активации на ее логарифмический вариант ReLU:

$$NLReLU(x) = ln(\mathrm{Re}\,LU(x)+1) = ln(\max(0,x)+1).$$

В таблице 4 представлены значения межконцевой задержки для некоторых этапов обучения, что демонстрирует более высокую обучающую способность нейронной сети, чем методы градиентного бустинга и случайного леса.

Таблица 4

Результаты оценки модели нейронной сети на некоторых этапах обучения

Набор данных	Момент обучения			
	5670		6912	
	Score	σ	Score	σ
Dataset300k	0,0733	0,2068	0,07070	0,1950
Test60k	0,0646	0,0936	0,0576	0,1005
	8448		9344	
	Score	σ	Score	σ
Dataset300k	0,0761	0,2048	0,0741	0,1986
Test60k	0,0659	0,1317	0,046	0,0716

На основании приведенных выше результатов использования различных методов машинного обучения можно провести их сравнительный анализ. Модель случайного леса прогнозирует больше значений межконцевой задержки с относительно небольшой ошибкой ($\leq 10\%$), чем градиентный бустинг, но при этом результатов, получаемых с неприемлемой ошибкой ($\geq 90\%$) несоизмеримо больше, чем у других методов. Метод градиентного бустинга для рассматриваемой тандемной системы дает наименьшее число результатов, близких к точному, но при этом число результатов с недопустимой ошибкой также меньше, чем у других рассмотренных методов (табл. 5).

Таблица 5

Число экспериментов при различных значениях ошибки ε

ε	Нейросеть	Градиент. Бустинг	Случайный лес
≤10%	532	347	474
≥50%	66	90	89
≥90%	35	25	43

3.5 Сравнительный анализ времени вычислений

Основная задача применения методов машинного обучения к задачам теории массового обслуживания – сокращение времени вычисления требуемых характеристик модели, возможно, с некоторым снижением точности вычислений. Анализ временных затрат показывает, что генерация одной выборки данных посредством имитационного моделирования занимает в среднем 4 секунды. Для масштабных наборов данных (сотни тысяч или миллионы выборок) это приводит к существенным временным затратам, ограничивающим применимость метода в условиях жёстких временных ограничений. В таблице 6 приведены сравнительные данные о производительности и объеме требуемой памяти для методов машинного обучения, что демонстрирует превосходство данных методов по вычислительной скорости на порядки величин по сравнению с методом имитационного моделирования для оценки производительности модели тандемной системы с повторными вызовами и общей орбитой.

Таблица 6

Сравнение объема памяти и среднего времени обучения

Метод	Объем памяти	Число итераций обучения	Общее время обучения
Градиентный бустинг	34,5 Мб	10000	143 с
Случайный лес	2,6 Гб	100	306 с
Нейронная сеть	24,8 Мб	10000	13 ч

4 Заключение

Для двухфазной тандемной системы разработан алгоритм расчета точных характеристик производительности (межконцевая задержка, средняя число заявок в узлах и на орбите и др.). Такое исследование имеет не только теоретическое значение, но используется для верификации имитационной модели, описывающей функционирование стохастической тандемной системы с произвольным числом фаз. Для исследования системы с произвольным числом фаз впервые предложен подход с использованием методов машинного обучения, обеспечивающий высокую вычислительную эффективность по сравнению с имитационным моделированием.

Проведена оценка достоинств и недостатков различных методов машинного обучения (градиентный бустинг, случайный лес и нейронные сети) с целью их применения для прогнозирования основных характеристик производительности стохастических тандемных систем с повторными вызовами и общей орбитой.

Литература

1. *Anitha K., Poongothai V., Godhandaraman P.* Performance analysis of a queueing system with tandem nodes, retrial and server vacations // Results in Control and Optimization. 2025. Vol. 18. Article 100520. DOI: 10.1016/j.rico.2025.100520.

2. *Vishnevsky V. M., Larionov A. A., Mukhtarov A. A., Sokolov A. M.* Investigation of tandem queuing systems using machine learning methods // Control Sciences. 2024. Vol. 4. P. 10-21. DOI: 10.25728/cs.2024.4.2.

3. *Rovetto C., Cruz E., Nuñez I., Santana K., Smolarz A., Rangel J., Cano E. E.* Minimizing intersection waiting time: proposal of a queue network model using Kendall's notation in Panama City // Applied Sciences. 2023. Vol. 13. No. 18. Article 10030. DOI: 10.3390/app131810030.

4. *Вишневский В. М., Ларионов А. А., Целикин Ю. В.* Анализ и исследование методов проектирования автоматизированных систем безопасности на автодорогах с использованием новых широкополосных беспроводных средств и RFID-технологий // T-Comm: Телекоммуникации и транспорт. 2012. № 7. С. 48-54.

5. *Вишневский В. М., Минниханов Р. Н., Барский И. В., Ларионов А. А.* Гибридная система идентификации транспортных средств // Вестник НЦБЖД. 2022. № 4 (54). С. 33-42.

6. *Dudin S. A., Dudina O. S., Dudin A. N.* Analysis of tandem queue with multi-server stages and group service at the second stage // Axioms. 2024. Vol. 13. No. 4. Article 214. DOI: 10.3390/axioms13040214.

7. *Dieleman N. A., Berkhout J., Heidergott B.* A neural network approach to performance analysis of tandem lines: the value of analytical knowledge // Computers and Operations Research. 2023. Vol. 152. No. 3. Article 106124. DOI: 10.1016/j.cor.2022.106124.

8. *Dudin S. A., Dudin A. N., Dudina O. S., Chakravarthy S. R.* Analysis of a tandem queuing system with blocking and group service in the second node // International Journal of Systems Science: Operations & Logistics. 2023. Vol. 10. No. 1. Article 2235270. DOI: 10.1080/23302674.2023.2235270.

9. *Yeger Y., Boxma O. J., Resing J., Vlasiou M.* ASIP tandem queues with consumption // Performance Evaluation Review. 2024., Vol. 51. No. 4, pp. 14-15. DOI: 10.1145/3649477.3649486

10. *Вишневский В. М., Ларионов А. А., Мухтаров А. А., Соколов А. М.* Исследование многофазных систем массового обслуживания с помощью методов машинного обучения // Проблемы управления. 2024. № 4. С. 13-25.

11. *Kim J., Kim B.* A survey of retrial queueing systems // Annals of Operations Research. 2016. Vol. 247, pp. 3-36. DOI: 10.1007/s10479-015-2038-7.

12. *Saravanan V., Poongothai V., Godhandaraman P.* Performance analysis of a multi server retrial queueing system with unreliable server, discouragement and vacation model // Mathematics and Computers in Simulation. 2023. Vol. 214. No. C, pp. 204-226. DOI: 10.1016/j.matcom.2023.07.008

13. *Avrachenkov K., Yechiali U.* On tandem blocking queues with a common retrial queue // Computers & Operations Research. 2010. Vol. 37. P. 1174-1180. DOI: 10.1016/j.cor.2009.10.004.

14. *Nazarov A. A., Paul S. V., Phung-Duc T., Morozova M.* Analysis of tandem retrial queue with common orbit and MMPP incoming flow // Lecture Notes in Computer Science. 2023. Vol. 13766, pp. 270-283. DOI: 10.1007/978-3-031-23207-7_21

15. *Вишневский В. М., Дудин А. Н., Клименок В. И.* Стохастические системы с коррелированными потоками. Теория и применение в телекоммуникационных сетях. М.: ТЕХНОСФЕРА, 2018. 564 с.

16. *Вишневский В. М., Ефросинин Д. В.* Теория очередей и машинное обучение. М.: ИНФРА-М, 2024. 370 с.

17. *Efrosinin D., Vishnevsky V., Stepanova N.* Optimal scheduling in general multi queue system by combining simulation and neural network techniques // Sensors. 2023. Vol. 23. No. 12. Article No. 5479.

18. *Vishnevsky V., Larionov A., Ivanov R., Semenova O.* Estimation of IEEE 802.11 DCF access performance in wireless networks with linear topology using PH service time approximations and MAP input // Proceedings of the IEEE 11th International Conference on Application of Information and Communication Technologies (AICT). 2017, pp. 1-5. DOI: 10.1109/ICAICT.2017.8687247.

ANALYSIS OF STOCHASTIC TANDEM SYSTEMS WITH RETRIALS

Vladimir V. Vishnevsky, *V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia,* ***vishn@inbox.ru***

Olga V. Semenova, *V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia,* ***olga.semenova@ipu.ru***

Abstract

At present, one of the key areas in the design and performance evaluation of modern telecommunication systems and networks is the analysis of mathematical models of tandem systems. Tandem systems are mathematical models of various technical and social systems, in particular, they are adequate models for broadband wireless networks deployed along extended transportation routes such as highways, railways, oil and gas pipelines, etc. The tandem system considered in this study consists of an arbitrary finite number of service units (stages) with a limited waiting space. The arrival process (packet flow) at the first stage is a correlated Markovian Arrival Process (MAP) and the service times have the phase?type distribution. If the buffer memory of any stage is full upon the customer arrival the customer is placed to an orbit (a shared memory for all stages). From there, after a random period of time, it is tries to get served at the first stage. We evaluate the key performance metrics: end?to?end delay (the time from the moment a customer enters the first stage from outside the system until it is successfully served at the final stage); average queue lengths at the system stages and in the orbit. For systems with two queues, exact formulas are derived to calculate the required performance characteristics. For systems with an arbitrary number of stages, we employ a machine learning approach in combination with a simulation modeling. The study involved conducting numerous numerical experiments across a wide range of parameters.

Keywords: *Queueing systems, tandem system, retrials (orbit), MAP-input, phase-type distribution, simulation, machine learning*

References

[1] K. Anitha, V. Poongothai, and P. Godhandaraman, "Performance analysis of a queueing system with tandem nodes, retrial and server vacations," *Results in Control and Optimization*, vol. 18, article 100520, 2025. doi: 10.1016/j.rico.2025.100520.

[2] V. M. Vishnevsky, A. A. Larionov, A. A. Mukhtarov, and A. M. Sokolov, "Investigation of tandem queuing systems using machine learning methods," *Control Sciences*, vol. 4, pp. 10-21, 2024. doi: 10.25728/cs.2024.4.2.

[3] C. Rovetto, E. Cruz, I. Nu?ez, K. Santana, A. Smolarz, J. Rangel, and E. E. Cano, "Minimizing intersection waiting time: proposal of a queue network model using Kendall's notation in Panama City," *Applied Sciences*, vol. 13, no. 18, article 10030, 2023. doi: 10.3390/app131810030.

[4] V. M. Vishnevskii, A. A. Larionov, and Yu. V. Tselikin, "Analysis and research of design methods for automated road safety systems using new broadband wireless tools and RFID technologies," *T-Comm*, no. 7, pp. 48-54, 2012.

[5] V. M. Vishnevskii, R. N. Minnikhanov, I. V. Barskii, and A. A. Larionov, "Hybrid vehicle identification system," *Vestnik NTsBZhD*, no. 4 (54), pp. 33-42, 2022.

[6] S. A. Dudin, O. S. Dudina, and A. N. Dudin, "Analysis of tandem queue with multi server stages and group service at the second stage," *Axioms*, vol. 13, no. 4, article 214, 2024. doi: 10.3390/axioms13040214.

[7] N. A. Dieleman, J. Berkhout, and B. Heidergott, "A neural network approach to performance analysis of tandem lines: the value of analytical knowledge," *Computers and Operations Research*, vol. 152, no. 3, article 106124, 2023. doi: 10.1016/j.cor.2022.106124.

[8] S. A. Dudin, A. N. Dudin, O. S. Dudina, and S. R. Chakravarthy, "Analysis of a tandem queuing system with blocking and group service in the second node," *International Journal of Systems Science: Operations & Logistics*, vol. 10, no. 1, article 2235270, 2023. doi: 10.1080/23302674.2023.2235270

[9] Y. Yeger, O. J. Boxma, J. Resing, and M. Vlasiou, "ASIP tandem queues with consumption," *ACM SIGMETRICS Performance Evaluation Review*, vol. 51, no. 4, pp. 14-15, Feb. 2024. doi: 10.1145/3649477.3649486.

[10] V. M. Vishnevskii, A. A. Larionov, A. A. Mukhtarov, and A. M. Sokolov, "Research of multi phase queuing systems using machine learning methods," *Problemy Upravleniya*, no. 4, pp. 13-25, 2024.

[11] J. Kim and B. Kim, "A survey of retrial queueing systems," *Annals of Operations Research*, vol. 247, pp. 3-36, 2016. doi: 10.1007/s10479-015-2038-7.

[12] V. Saravanan, V. Poongothai, and P. Godhandaraman, "Performance analysis of a multi server retrial queueing system with unreliable server, discouragement and vacation model," *Mathematics and Computers in Simulation*, vol. 214, pp. 204-226, 2023. doi: 10.1016/j.matcom.2023.07.008.

[13] K. Avrachenkov and U. Yechiali, "On tandem blocking queues with a common retrial queue," *Computers & Operations Research*, vol. 37, pp. 1174-1180, 2010. doi: 10.1016/j.cor.2009.10.004.

[14] A. A. Nazarov, S. V. Paul, T. Phung Duc, and M. Morozova, "Analysis of tandem retrial queue with common orbit and MMPP incoming flow," in *Lecture Notes in Computer Science*, vol. 13766, pp. 270-283, 2023. doi: 10.1007/978-3-031-23207-7_21.

[15] V. M. Vishnevskii, A. N. Dudin, and V. I. Klimenok, Stochastic Systems with Correlated Flows: Theory and Application in Telecommunication Networks. Moscow: Tekhnosfera, 2018, 564 pp.

[16] V. M. Vishnevskii and D. V. Efrosinin, Queueing Theory and Machine Learning. Moscow: INFRA-M, 2024, 370 pp.

[17] D. Efrosinin, V. Vishnevsky, and N. Stepanova, "Optimal scheduling in general multi queue system by combining simulation and neural network techniques," *Sensors*, vol. 23, no. 12, article 5479, 2023.

[18] V. Vishnevsky, A. Larionov, R. Ivanov, and O. Semenova, "Estimation of IEEE 802.11 DCF access performance in wireless networks with linear topology using PH service time approximations and MAP input," in *Proceedings of the IEEE 11th International Conference on Application of Information and Communication Technologies (AICT),* 2017, pp. 1-5. doi: 10.1109/ICAICT.2017.8687247.

Information about authors:

Vladimir V. Vishnevsky, *Dr. Sci. (Techn.), Professor, Chief Researcher, Laboratory of Telecommunication Systems, V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. ORCID 0000-0001-7373-4847*

Olga V. Semenova, *Dr. Sci. (Phys.-Math.), Leading Researcher, Laboratory of Telecommunication Systems, V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences. ORCID 0000-0001-7846-6212*

ГРУППОВЫЕ ПОТОКИ В СИСТЕМАХ МАССОВОГО ОБСЛУЖИВАНИЯ: МОДЕЛИРОВАНИЕ ПАКЕТНОЙ РЕАЛЬНОСТИ В ТЕЛЕКОММУНИКАЦИЯХ

DOI: 10.36724/2072-8735-2026-20-6-30-43

Лихтциндер Борис Яковлевич,
Поволжский государственный университет телекоммуникаций и информатики, Самара, Россия,
lixt@psuti.ru

Тетюев Михаил Петрович,
Поволжский государственный университет телекоммуникаций и информатики, Самара, Россия,
yhyr@mail.ru

Васильев Александр Протальонович,
Московский технический университет связи и информатики, Москва, Россия

Manuscript received 17 March 2026;
Accepted 28 May 2026

Ключевые слова: *системы массового обслуживания, групповые потоки, групповой пуассоновский поток, групповой детерминированный поток, смешанный групповой поток, квазипуассоновский поток, парный групповой поток, гиперэкспоненциальный поток, интервальный метод анализа, формула Полячека-Хинчина, средняя длина очереди, видеотрафик*

В данной статье рассматриваются системы массового обслуживания (СМО) с групповыми (пачечными) входными потоками, в которых заявки поступают не по отдельности, а пачками. Целью работы является систематизация моделей групповых потоков, получение аналитических выражений для средней длины очереди и дисперсии в одноканальной СМО с постоянным временем обслуживания (G/D/1). Основу методологии составляет интервальный метод анализа, позволяющий описывать неординарные потоки через распределение числа заявок, поступающих за время обслуживания одной заявки. В работе рассмотрены следующие модели: групповой пуассоновский поток; смесь группового пуассоновского потока с фоновым ординарным пуассоновским потоком; групповой детерминированный поток с постоянными интервалами между пачками; смешанный групповой поток, объединяющий детерминированную и пуассоновскую компоненты; квазипуассоновский поток с одинаковыми паузами, где каждый меж пачечный интервал содержит постоянную составляющую; парное обобщение такого потока, в котором за один цикл формируются две пачки; гиперэкспоненциальный групповой поток второго порядка. Различные модели позволяют аппроксимировать различные виды поведения реального трафика. Для каждой модели получены аналитические выражения, связывающие среднюю длину очереди и коэффициент загрузки. Достоверность предложенных моделей подтверждена результатами имитационного моделирования, показавшими высокую степень соответствия аналитических и экспериментальных данных во всём диапазоне рабочих нагрузок. Также рассмотрена аппроксимация реального видеотрафика стандарта H.264 с помощью группового пуассоновского потока. Показано, что данная модель существенно точнее описывает среднюю длину очереди и её дисперсию по сравнению с ординарным пуассоновским потоком. Предложенные модели представляют собой аналитически простой, но достаточно точный инструмент для расчёта очередей в системах с групповым поступлением заявок, пригодный для аппроксимации реальных потоков.

Информация об авторах:

Лихтциндер Борис Яковлевич, *д.т.н., профессор, главный научный сотрудник научно-исследовательской лаборатории "Инфокоммуникационные технологии" Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ), г. Самара, Россия*

Тетюев Михаил Петрович, *аспирант ПГУТИ, г. Самара, Россия*

Васильев Александр Протальонович, *Московский технический университет связи и информатики, Москва, Россия*

Для цитирования:
Лихтциндер Б.Я., Тетюев М.П., Васильев А.П. Групповые потоки в системах массового обслуживания: моделирование пакетной реальности в телекоммуникациях // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 30-43.

For citation:
B.Ya. Likhttsinder, M.P. Tetiuev, A.P. Vasiliev, "Group Flows In Queuing Systems: Modelling Packet Reality In Telecommunications," *T-Comm*, 2026, vol. 20, no.6, pp. 30-43. (*in Russian*)

Введение

Теория систем массового обслуживания (СМО) представляет собой математический аппарат для анализа процессов, где возникают очереди заявок. Основой любой модели СМО является математическое описание входящего потока заявок на обслуживание. Эволюция моделей потоков отражает стремление исследователей адекватно описать реальные процессы, которые зачастую существенно отклоняются от идеализированных предположений. Исторически первой и наиболее глубоко изученной моделью стал пуассоновский (простейший) поток, который лег в основу классической теории СМО, разработанной А.Я. Хинчиным, А.К. Эрлангом и другими учеными в начале XX века [1, 2]. Основные свойства пуассоновского потока это: стационарность (вероятностные характеристики не зависят от времени), ординарность (вероятность появления двух и более событий в малый промежуток времени пренебрежимо мала), отсутствие последействия (интервалы между событиями независимы и распределены экспоненциально). Пуассоновский поток стал основополагающим благодаря своей математической «доступности» – марковскому свойству, которое позволяет строить аналитические решения для широкого класса СМО.

Модели M/M/1, M/M/n и их производные до сих пор остаются важным инструментом предварительного анализа систем. По мере применения теории СМО к реальным телекоммуникационным системам обнаружились существенные ограничения пуассоновской модели. Реальные потоки пакетов часто демонстрировали высокую пачечность, зависимость интенсивности от времени, корреляцию между интервалами. Это привело к развитию немарковских моделей потоков: рекуррентные потоки, в которых интервалы между событиями независимы и одинаково распределены, но не обязательно по экспоненциальному закону [3].

Эти потоки стали важным обобщением, сохраняющим свойство отсутствия последействия при произвольном распределении интервалов. Были разработаны специализированные классы потоков, такие как модулированные марковские потоки (MMPP), где интенсивность событий управляется внешним марковским процессом с конечным числом состояний. MMPP позволил моделировать коррелированные потоки с переменной интенсивностью, находящие применение в телекоммуникациях для описания трафика с "пачечной" структурой [4].

К марковским моделям также относятся многоканальные СМО с групповым обслуживанием, зависящим от размера группы, и нетерпеливостью заявок [5, 6] С развитием интернет-трафика в 1990-х годах были обнаружены потоки с долгосрочной зависимостью и самоподобием [7]. Модели на основе фрактального броуновского движения и устойчивых распределений Леви позволили адекватно описывать сетевой трафик, демонстрирующий масштабно-инвариантные свойства. Однако указанные модели оказались достаточно сложными и в настоящее время не получили значительного развития. Аналитические методы исследования потоков развивались от преобразований Лапласа-Стилтьеса к матрично-аналитическим методам, разработанным М.Ф. Нейтсом [8].

1 Классическая пуассоновская модель

В литературе сложились противоположные взгляды на применимость пуассоновской модели к суммарному трафику. Как показано в [9], неизменная пуассоновская модель сохраняет объяснительную силу при анализе суммарного трафика на определённых временных масштабах. Анализ магистральных каналов связи стандарта OC48 и WIDE демонстрирует, что на интервалах меньше секунды распределение промежутков времени между пакетами является близким к экспоненциальному, а корреляция между соседними интервалами и размерами пакетов являются статистически незначимыми (функция автокорреляции не выходит за пределы 95%-го доверительного интервала). Однако в [10] приводится, что даже на магистральном трафике (измерения на оптическом канале Gigabit Ethernet с наносекундной точностью) гипотеза о пуассоновском процессе отвергается формальными статистическими тестами. Авторы отмечают, что огрубление временных меток до микросекунд может скрывать зависимость

В трассах данной работы обнаружено самоподобие и нетривиальное мультифрактальное масштабирование, что указывает на наличие долговременной зависимости, которую классическая пуассоновская модель по определению не способна уловить. Сравнение моделей показало, что классическая пуассоновская модель систематически недооценивает вероятность блокировки в многоканальных системах и вероятность переполнения буфера по сравнению с исходными трассами, хотя и имеет степень приближения, способную описывать определённые закономерности на масштабах меньше секунды.

Таким образом, несмотря на противоположность исходных посылок, результаты рассмотренных исследований сходятся в том, что пуассоновские модели способны к частичному описанию реального трафика. В настоящей работе предлагается модифицировать пуассоновскую модель таким образом, чтобы она была способна учитывать определенные зависимости и закономерности. Такая модификация позволяет модели оставаться достаточно простой, но при этом описывать конкретные сценарии трафика с учётом зависимостей.

2 Интервальный анализ СМО

В данной работе используются интервальные методы анализа, так как они оперируют неординарными потоками заявок. Поэтому в дальнейшем рассматривается анализ интервальными методами СМО с групповыми входными потоками заявок (пакетов, кадров) [11].

Классическая теория СМО предполагает рассмотрение двух распределений вероятностей: распределения вероятностей длин временных интервалов между соседними заявками и распределения вероятностей времен обслуживания заявок. При использовании интервального метода предполагается анализ одной характеристики – вероятностного распределения числа заявок, приходящих за период обслуживания одной заявки.

Заявки, поступающие в течение интервала обслуживания одной заявки, образуют неординарный поток, хорошо согласующийся с групповыми потоками, рассматриваемыми в данной работе. С помощью интервального метода была получена зависимость среднего размера очереди от выбранного коэффициента загрузки [11]:

$$\overline{q(\rho)} = \frac{D_m(\rho) + 2Cov_{q_{i-1}m_i}(\rho)}{2(1-\rho)} - \frac{\rho}{2}, \quad (1)$$

Здесь ρ – коэффициент загрузки. Отношение интенсивности входящего потока заявок к пропускной способности системы. При $\rho \geq 1$ интенсивность поступления превышает возможности обслуживания, что приводит к неограниченному росту очереди и превышению пропускной способности системы. $m_i(\rho)$ – зависимость числа заявок приходящих на i-й интерал. $Cov_{q_{i-1}m_i}$ – ковариация случайных величин – количества заявок, поступивших за i-й интервал, и длины очереди на предшествующем интервале. Определяется выражением:

$$Cov_{q_{i-1}m_i}(\rho) = \overline{[q_{i-1}(\rho)] \cdot [m_i(\rho) - \rho]}.$$

Соотношение (1) – это обобщение формулы Полячека-Хинчина и является справедливым для любых стационарных потоков заявок при условии фиксированного значения коэффициента загрузки. Если в потоке отсутствуют корреляционные свойства $Cov_{q_{i-1}m_i}(\rho) = 0$, то получим:

$$\overline{q(\rho)} = \frac{D_m(\rho)}{2(1-\rho)} - \frac{\rho}{2}, \quad (2)$$

При анализе очередей в СМО с групповыми потоками ключевой является формула (1). Так как для пуассоновского потока заявок $D_m(\rho) = \rho$, то формула (2) принимает вид известного соотношения:

$$\overline{q(\rho)} = \frac{\rho^2}{2(1-\rho)}. \quad (3)$$

3 Групповые потоки

Одним из видов потоков, наиболее полно отражающих свойства пачечного трафика телекоммуникационных систем, следует считать рассматриваемые в данной работе групповые потоки.

В телекоммуникационных сетях с пакетной коммутацией (сети Ethernet, IP) пакеты данных передаются группами, образуя групповые (пачечные) потоки заявок. В СМО различают групповые входные потоки (Arrival Batch), когда заявки прибывают группами случайного или фиксированного размера, и групповое обслуживание (Service Batch), когда канал обслуживания за одну операцию может обслужить несколько заявок одновременно. Комбинация этих двух типов создает богатый класс моделей, более адекватно отражающих сложные реальные системы.

Для описания таких СМО используется расширенная нотация Кендалла. Классическая запись A/B/m дополняется двумя параметрами: G^b: групповой вход, где b – распределение размера группы входного потока (математическое ожидание G(B)). Например, означает групповой пуассоновский поток, с распределением числа заявок в группах – b и экспоненциальным распределением времен обслуживания в одноканальной СМО.

Пусть имеется пуассоновский поток независимых событий с интенсивностью λ. Каждое i-е событие (i = 1, 2, …) означает одновременное поступление пачки пакетов, подлежащих передаче. Здесь размеры пачек B_i – независимые одинаково распределённые случайные величины с распределением $\Pr\{B_i = k\} = f_k, k = 1, 2....$ Время передачи каждого пакета задаётся дискретным распределением $\Pr\{\tau_i = t_k\} = p_k, k = 1, 2....K$, причём величины независимы и одинаково распределены.

Для дальнейшего расчета необходимо вывести моменты $m(\tau_i)$ с помощью производящих функций. Определим производящую функцию размера пачки как $f(z) = \sum_{k=1}^{\infty} f_k z^k$.

В таком случае для $m(\tau_i)$ имеем:

$$G_{m(\tau_i)}(z) = \sum_{k=1}^{K} \Pr\{\tau_i = t_k\} = \sum_{n=1}^{\infty} \frac{(\lambda t_k)^n}{n!} e^{-\lambda t_k} (f(z))^n =$$

$$= . \sum_{k=1}^{K} \mathrm{P}_k e^{\lambda t_k (f(z)-1)}$$

Здесь использовались свойства пачек пуассоновского потока:

$$\left. \frac{\partial G_{m(\tau_i)}(z)}{\partial Z} \right|_{z=1} = \left. \sum_{k=1}^{K} \mathrm{P}_k \lambda t_k f'(z) e^{\lambda t_k (f(z)-1)} \right|_{z=1} \quad (4)$$

$$= \lambda \overline{B} \sum_{k=1}^{K} p_k t_k = \lambda \overline{\tau} \overline{B} = \overline{m(\tau_i)}$$

$$m(\tau_i) = \lambda \overline{\tau} \overline{B}$$

В свою очередь $\overline{B}$ и $\overline{\tau}$ это математические ожидания соответствующих случайных величин, аналогично:

$$\left. \frac{\partial^2 G_{m(\tau_i)}(z)}{(\partial z)^2} \right|_{z=1} =$$

$$\left. \sum_{k=1}^{K} p_k \left(\lambda t_k f''(z) + (\lambda t_k f'(z))^2 \right) e^{\lambda t_k (f(z)-1)} \right|_{z=1} =$$

$$\lambda \overline{\tau} (\overline{B^2} - \overline{B}) + \overline{\tau^2} (\lambda \overline{B})^2 =$$

$$\overline{m^2(\tau_i)} - \overline{m(\tau_i)}, \overline{m^2(\tau_i)} = \lambda \overline{\tau} \overline{B^2} + \overline{\tau^2} (\lambda \overline{B})^2$$

В результате получаем:

$$D\left(m(\tau_i)\right) = \lambda \overline{\tau} \overline{B^2} + (\lambda \overline{B})^2 D(\tau)$$

Если принять время обслуживания заявок постоянным, то учитывая, что $B\lambda\tau = \rho$, получим:

$$D_m(\rho) = B(1-\nu_B^2)\rho, \quad (5)$$

где ν_B^2 - квадрат коэффициента вариации размеров пачек. На рисунках 1 и 2 прямой линией обозначена зависимость для размера пачки B=10, точечной линией для B=20, точкой-тире B=40, тире B=80. При этом, на рис. 1 показаны графики зависимости дисперсии от коэффициента загрузки при различных размерах пачек B. Зависимости, как и следовало ожидать, линейные.

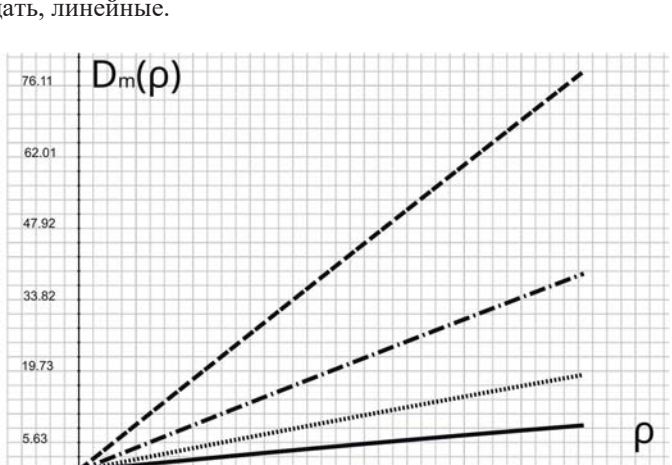


Рис. 1. Зависимости дисперсии $D_m(\rho)$ от коэффициента загрузки ρ для указанных выше различных значений параметров B.

Если подставить выражения (4) и (5) в (2), результирующим выражением будет:

$$\overline{q(\rho)} = \frac{B(1-\nu_B^2)\rho}{2(1-\rho)} - \frac{\rho}{2} \quad (6)$$

Формула (6) даёт возможность вычислять средние размеры очередей для СМО, на вход которых поступают групповые пуассоновские потоки [12]. Зависимость средней очереди $\overline{q(\rho)}$ от коэффициента загрузки приведена на рис. 2.

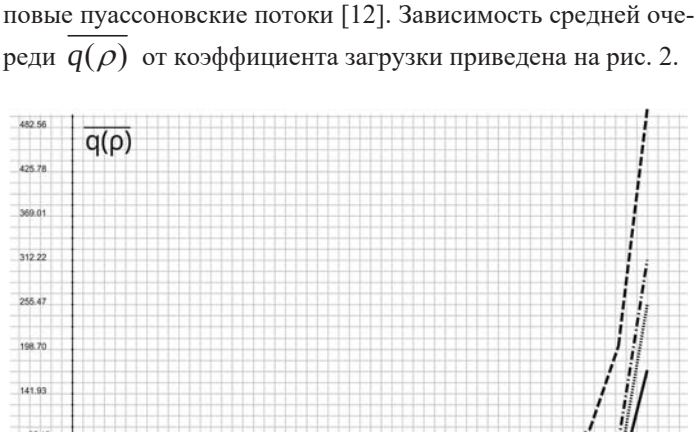


Рис. 2. Кривые зависимостей средней длины очереди $\overline{q(\rho)}$ от коэффициента загрузки ρ для группового пуассоновского потока для указанных выше значений параметров B.

4 Второй начальный момент очередей в СМО с групповыми пуассоновскими потоками

Второй начальный момент очереди можно определить используя уравнение баланса [13]:

$$\begin{aligned} &q_i(\rho) = q_{i-1}(\rho) + m_i(\rho) - \delta_i(\rho); \\ &\delta_i(\rho) = 0, \textit{если}\ \ q_{i-1}(\rho) = m_i(\rho) = 0; \\ &\delta_i(\rho) = 1, \quad \textit{в противном случае.} \end{aligned} \quad (7)$$

Здесь, как и прежде, $m_i(\rho)$ – число поступивших заявок, $q_i(\rho)$ – значение очереди, а $\delta_i(\rho)$ – количество заявок, обработанных за временной интервал – τ_i .

При имеющихся ограничениях:

$$\begin{aligned} &\delta_i^k(\rho) = \delta_i(\rho), \quad m_i(\rho)\delta_i(\rho) = m_i(\rho), \\ &q_{i-1}(\rho)\delta_i(\rho) = q_{i-1}(\rho), \quad \overline{\delta_i(\rho)} = \overline{m_i(\rho)}. \end{aligned} \quad (8)$$

В процессе возведения уравнения (7) в третью степень, для сокращения записи коэффициент загрузки ρ опускается:

$$q^3{}_i = q^3{}_{i-1} + 3q^2{}_{i-1}(m_i - \delta_i) + 3q\ {}_{i-1}(m_i - \delta_i)^2 + (m_i - \delta_i)^3.$$

В результате соответствующих преобразований, с учетом ограничений (8), получается следующее:

$$\begin{aligned} q^3{}_i = q^3{}_{i-1} + 3q^2{}_{i-1}m_i - 3q^2{}_{i-1} \\ +3q_{i-1}(m_i{}^2 - 2m_i + 1) + \\ +(m_i{}^3 - 3m_i{}^2 + 3m_i - \delta_i) \end{aligned} .$$

Произведем усреднение обеих частей:

$$\begin{aligned} 3\overline{q^2{}_{i-1}}\left(1-\overline{m_i}\right) = 3\overline{q_{i-1}}(\overline{m_i{}^2} - 2\overline{m_i} + 1) + \\ +(\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}) \end{aligned} .$$

$$\begin{aligned} 3\overline{q^2{}_{i-1}}\left(1-\overline{m_i}\right) = 3\overline{q_{i-1}}(\overline{D_m} + \overline{m_i}^2 - 2\overline{m_i} + 1) + \\ +(\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}) \end{aligned} .$$

$$\begin{aligned} 3\overline{q^2{}_{i-1}}\left(1-\overline{m_i}\right) = 3\overline{q_{i-1}}\,D_m + 3\overline{q_{i-1}}(\overline{m_i} - 1)^2 + \\ +(\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}) \end{aligned} .$$

Далее, необходимо определить $\overline{q^2{}_{i-1}}$.

$$\overline{q^2{}_{i-1}} = \frac{\overline{q_{i-1}}\,D_m}{\left(1-\overline{m_i}\right)} + \overline{q_{i-1}}\cdot\left(1-\overline{m_i}\right) + \frac{\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)}.$$

$$\overline{q^2{}_{i-1}} = \overline{q_{i-1}}\left[\frac{D_m}{\left(1-\overline{m_i}\right)} + \left(1-\overline{m_i}\right) + \frac{\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)}\right].$$

$$\overline{q^2{}_{i-1}} = \overline{q_{i-1}}\,\frac{D_m + \left(1-\overline{m_i}\right)^2}{\left(1-\overline{m_i}\right)} + \frac{\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)},$$

$$\overline{q^2{}_{i-1}} = \frac{\overline{q_{i-1}}}{\left(1-\overline{m_i}\right)}\left[D_m + \left(1-\overline{m_i}\right)^2\right] + \frac{\overline{m_i{}^3} - 3\overline{m_i{}^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)}.$$

Третий начальный момент $\overline{m_i^3}$ выражается через третий центральный момент:

$$\mu_3 = \overline{(m_i - m_i)^3},$$
$$\overline{m_i^3} = \overline{(m_i - m_i)^3} + 3\overline{m_i}D_m + \overline{m_i^3} = \mu_3 + 3\overline{m_i}D_m + \overline{m_i^3}.$$

В результате подстановки получается следующее:

$$\overline{q^2_{i-1}} = \overline{q_{i-1}}\left[\frac{D_m + \left(1-\overline{m_i}\right)^2}{\left(1-\overline{m_i}\right)}\right]$$
$$+\frac{3\overline{m_i}D_m + \overline{m_i^3} - 3D_m - 3\overline{m_i^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)} + \frac{\mu_3}{3\left(1-\overline{m_i}\right)}$$
$$\overline{q^2_{i-1}} = \frac{\overline{q_{i-1}}}{\left(1-\overline{m_i}\right)}\left[D_m + \left(1-\overline{m_i}\right)^2\right] +$$
$$+\frac{\overline{m_i^3} + 3\overline{m_i}D_m - 3\overline{m_i^2} + 2\overline{m_i}}{3\left(1-\overline{m_i}\right)} + \frac{\mu_3}{3\left(1-\overline{m_i}\right)}.$$

Ввиду того, что в групповом пуассоновском потоке пачки независимы, то в (1) ковариация равна нулю, в таком случае при подстановке $\overline{q_{i-1}} = \overline{q(\rho)}$ из (1), при учёте, что $\overline{m(\rho)} = \lambda\tau = \rho$, получим:

$$\overline{q^2(\rho)} = \frac{D_m(\rho) - \rho(1-\rho)}{2\left(1-\rho\right)^2}\left[D_m(\rho) + \left(1-\rho\right)^2\right] +$$
$$+\frac{\rho^3 + 3\rho D_m(\rho) - 3D_m(\rho) - 3\rho^2 + 2\rho}{3\left(1-\overline{\rho}\right)} + \frac{\mu_3(\rho)}{3\left(1-\rho\right)}.$$

Второй начальный момент размера очереди в СМО с групповыми пуассоновскими потоками определяется приведённым ниже соотношением:

$$\overline{q^2(\rho)} = \frac{B\rho - \rho + \rho^2}{2\left(1-\rho\right)^2}\left[B\rho + 1 - 2\rho + \rho^2\right] +$$
$$+\frac{\rho^3 + 3B\rho^2 - 3B\rho - 3\rho^2 + 2\rho}{3\left(1-\overline{\rho}\right)} + \frac{\mu_3(\rho)}{3\left(1-\rho\right)}.$$

Для простейшего потока $B_m = 1,\ \mu_3(\rho) = \rho$, выражение сводится к известному соотношению:

$$\overline{q^2(\rho)} = \frac{\rho^2}{2\left(1-\rho\right)^2}\left(1 - \frac{1}{3}\rho + \frac{1}{3}\rho^2\right)$$

Таким образом, в последнем случае второй момент зависит исключительно от коэффициента загрузки.

5 Дисперсия очередей в СМО с групповыми пуассоновскими потоками

Определим дисперсию размеров очередей используя как основу следующее соотношение [14]:

$$\overline{D_q(\rho)} = \overline{q^2(\rho)} - \overline{q(\rho)}^2$$

В результате преобразований и подстановок:

$$\overline{D_q(\rho)} = \frac{D_m(\rho) - \rho(1-\rho)}{4\left(1-\rho\right)^2}\left[D_m(\rho) + 2 - 3\rho + \rho^2\right]$$
$$+\frac{\rho^3 + 3\rho D_m(\rho) - 3D_m(\rho) - 3\rho^2 + 2\rho}{3\left(1-\overline{\rho}\right)} + \frac{\mu_3(\rho)}{3\left(1-\rho\right)} \qquad (9)$$

Дисперсия в данном случае имеет зависимость от третьего центрального момента.

Применительно к простейшему пуассоновскому потоку выражение (9) становится существенно проще и сводится к известному соотношению:

$$B = 1,\ D_m(\rho) = \mu_3(\rho) = \rho,$$
$$\overline{D_q(\rho)} = \frac{\rho^2}{2\left(1-\rho\right)^2}\left(1 - \frac{1}{3}\rho - \frac{1}{6}\rho^2\right).$$

Характер изменения дисперсии очереди различен для двух типов потоков. Для пуассоновского потока: дисперсия очереди это функция исключительно коэффициента загрузки. Для группового пуассоновского потока: дисперсия выражается через третий центральный момент распределения размеров пачек. Последний, в свою очередь, определяется степенью симметричности этого распределения.

6 Суммирование независимых групповых пуассоновских потоков

Групповые пуассоновские потоки обладают нулевой корреляцией и линейной зависимостью дисперсии от коэффициента загрузки. Кроме того, при мультиплексировании нескольких таких потоков результирующий поток остаётся групповым пуассоновским. В таком случае дисперсия результата равна сумме дисперсий исходных потоков, а коэффициент загрузки – сумме их коэффициентов:

$$D_m(\rho) = \sum_i^M D_m(\rho_i),$$
$$\rho = \sum_i^M \rho_i,$$

где M это количество исходных потоков. В соответствии с выражением (2) для одноканальной системы массового обслуживания с групповым пуассоновским потоком средний размер очереди записывается следующим образом:

$$\overline{q(\rho)} = \frac{\rho E}{2(1-\rho)} - \frac{\rho}{2}.$$

Тогда формула для среднего значения очереди суммарного потока принимает вид [15]:

$$\overline{q(\rho)} = \frac{\sum_{i=1}^M D_m(\rho_i)}{2(1-\sum_i^M \rho_i)} - \frac{\sum_{i=1}^M \rho_i}{2} = \frac{\rho\sum_{i=1}^M P_i\frac{\overline{B_i^2}}{\overline{B_i}}}{2(1-\rho)} - \frac{\rho}{2}.$$

здесь $P_i = \frac{\rho_i}{\rho}$ это вероятность загрузки i-м потоком.

7 Групповой пуассоновский поток в условиях существования фонового ординарного пуассоновского потока

Как было сказано ранее, групповому пуассоновскому потоку свойственны стационарность и отсутствие последействия, но не свойственна ординарность: допускается одновременное прибытие нескольких заявок («пачки»). Событием является прибытие пачки со случайным числом заявок B_i. В частном случае группового пуассоновского потока пачка, состоит ровно из одной заявки и соответствует обычному пуассоновскому потоку. В общем же случае событие в групповом потоке означает одновременное прибытие пачки, содержащей случайное число заявок (распределённых согласно некоторому закону). При этом сохраняются стационарность и отсутствие последействия, но теряется ординарность (разрешены множественные поступления в один момент). В данной работе за случайное число заявок считается достаточным принять значение B как постоянное означающее среднее количество поступающих пакетов и будет использовано подобным образом в дальнейших формулах, до тех пор, пока не указано иначе.

Как было отмечено ранее, зависимость дисперсии $D_m(\rho)$ от коэффициента загрузки ρ носит линейный характер, что справедливо как для группового, так и для обычного пуассоновского потока. Коэффициенты загрузки одноканальной СМО от группового пуассоновского и от обычного пуассоновского потоков обозначим как ρ_1 и ρ_2 соответственно. Для независимых потоков дисперсия и коэффициент загрузки результирующего потока равны суммам соответствующих величин отдельных потоков

Следовательно, суммарный коэффициент загрузки принимает значение:

$$\rho = \rho_1 + \rho_2 = \rho P + \rho(1-P),$$

где P – вероятность того, что загрузка обусловлена групповым пуассоновским потоком.

Средняя длина очереди в СМО, на которую поступает смесь указанных потоков, выражается следующим соотношением:

$$\overline{q(\rho)} = \frac{[P(B-1)+1]\rho}{2(1-\rho)} - \frac{\rho}{2}, \quad B \geq 1.$$

Числитель в данном выражении линейно связан с коэффициентом загрузки ρ. При значении $P = 1$ фоновый пуассоновский поток отсутствует, и при фиксированном параметре B исходное выражение приводится к виду (1) с нулевой ковариацией. В ситуации, когда $P = 0$, групповой пуассоновский поток отсутствует, и формула переходит в соотношение (3). После соответствующих преобразований исходной зависимости получается выражение, которое учитывает оба рассмотренных случая.

$$\overline{q(\rho)} = \frac{\rho P(B-1) + \rho^2}{2(1-\rho)}, \quad B \geq 1.$$

Изменяя значение коэффициента P, можно регулировать начальный угол наклона кривой, отражающей зависимость средней длины очереди от загрузки системы. Это позволяет осуществлять настройку параметров модели для аппроксимации реального видеопотока. В данной модели числитель содержит две компоненты: линейную, которая определяется долей группового пуассоновского потока, и квадратичную, описывающую вклад фонового группового потока.

При размере пачки B=1, поток становится обычным пуассоновским потоком, и данная формула становится классическим выражением (3).

7.1 Групповые детерминированные и смешанные потоки

В данном разделе вводятся и рассматриваются групповые потоки, характеризующиеся тем, что интервалы между последовательными пачками являются постоянными, то есть носят детерминированный характер. В рассматриваемом частном случае, когда размеры пачек постоянны, формула для средних размеров пачек получается достаточно простой [14]:

$$\overline{q_Д(\rho)} = \frac{B\rho}{2} - \frac{\rho}{2}.$$

Указанные зависимости очередей носят линейный характер. Если групповой пуассоновский поток совместить с групповым детерминированным, то такой поток будем называть групповым смешанным. Такой поток может быть использован в качестве моделей пачечного трафика.

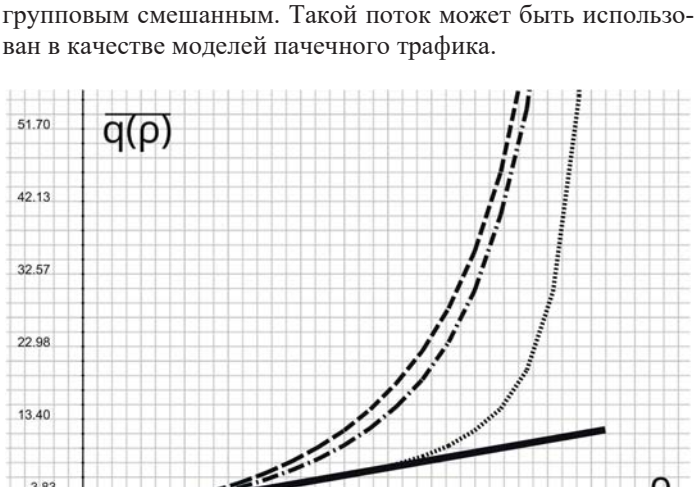


Рис. 3. Кривые зависимостей средних размеров очередей $\overline{q(\rho)}$ от коэффициента загрузки ρ в случае смешанных групповых потоков

На рис. 3 показано, как меняются средние длины очередей в зависимости от коэффициента загрузки ρ для смешанных групповых потоков. Рассматриваются различные соотношения интенсивностей детерминированной и групповой составляющих, число заявок в пачках при этом одинаковое. Верхняя кривая отвечает групповому пуассоновскому потоку, нижняя – групповому детерминированному. Средние кривые отражают зависимости средних размеров очередей $\overline{q(\rho)}$ от коэффициента загрузки ρ для смешанных групповых потоков. Чем ниже кривая, тем больше доля детерминированной части среди смешанного потока.

Если сложить средние значения очередей, получаемые при независимой генерации групповых детерминированных и групповых пуассоновских потоков, приходим к следующему выражению:

$$\overline{q(\rho)} = \overline{q_Д(\rho)}P + \overline{q_П(\rho)}(1-P);$$
$$\overline{q(\rho)} = \frac{B\rho(1-\rho P)}{2(1-\rho)} - \frac{\rho}{2}. \qquad (10)$$

$\overline{q_Д(p)}$ и $\overline{q_П(p)}$ – соответствующие средние очереди для групповых детерминированного и пуассоновского случаев. Размеры пачек для обоих потоков принимаются одинаковыми, постоянными и равными B. Вероятность появления интервала детерминированного потока равна *P*, пуассоновского (1 – *P*). Соотношение (10) даёт достаточно точную оценку средних очередей для смешанных групповых потоков, что подтверждается рисунком 4.

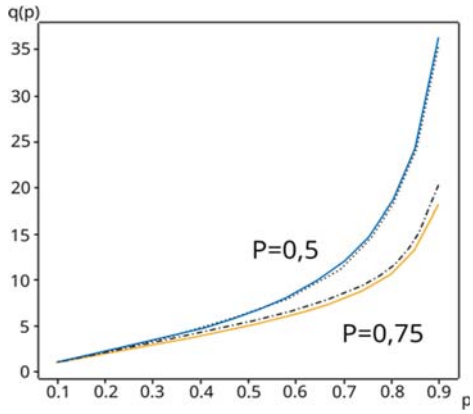


Рис. 4. Сопоставление зависимости средней длины очереди от загрузки прибора, построенной по данным имитационного моделирования, с расчётами по формуле (10). Размер пачки B = 20.

8 Групповые квазипуассоновские потоки с одинаковыми паузами

На базе модели группового пуассоновского потока предлагается ее расширение – групповой квазипуассоновский поток, характеризующийся одинаковыми паузами.

Характерной особенностью рассматриваемого группового потока является то, что к каждому экспоненциально распределенному интервалу добавляется одинаковый и постоянный интервал T. Поступление заявок в данном потоке происходит пачками, где числа заявок в каждой пачке - независимые, одинаково распределённые дискретные случайные величины B_k.

Интервал времени между моментами прибытия последовательных пачек представляет собой сумму экспоненциально распределённой случайной величины с параметром λ и постоянной величины времени *T*. Поскольку ключевой интерес представляет связь между средней длиной очереди и загрузкой прибора, и при этом временной масштаб не важен, рассматриваемый квазипуассоновский поток может быть описан с помощью двух параметров: B – постоянный размер пачки, $\alpha = \lambda T$ – величина, равная отношению детерминированной паузы к среднему значению времени экспоненциально распределённого интервала

Для определения параметров модели использовалось обобщение формулы Полячека-Хинчина для системы G/D/1, полученное интервальным методом [16]. В результате получена формула, позволяющая оценить среднюю очередь в СМО G/D/1 в зависимости от коэффициента загрузки прибора. Полученная формула демонстрирует хорошее соответствие с результатами имитационного моделирования.

$$\overline{q(\rho)} = \frac{\rho B - \rho(1-\rho)}{2}, \quad если\ \rho \le \rho_0$$
$$\overline{q(\rho)} = \frac{\rho_0}{\rho}(2 - \frac{\rho_0}{\rho})(\frac{\rho B - \rho(1-\rho)}{2}) + \frac{\rho^* B^* - \rho^*(1-\rho)^*}{2(1-\rho^*)}, \ если\ \rho > \rho_0 \qquad (11)$$

На рис. 5а и 5б линиями изображены выходные данные имитационной модели, маркерами - расчётные точки по формуле (11). При этом, на рисунке 5а звездой обозначены результаты при параметрах α =0.5, треугольником α =1, кругом α =2. Результаты моделирования, обозначенные тире α =0.5, точка-тире =1 и сплошной α =2 соответственно. На рисунке 5б звездой обозначено оценочное значение с параметром *B*=10, треугольником *B*=50, кругом *B*=100. Результаты моделирования обозначенные тире имеют значение *B*=10, точкой-тире *B*=50 и сплошной *B*=100 соответственно.

Полученное приближение показывает высокую степень соответствия.

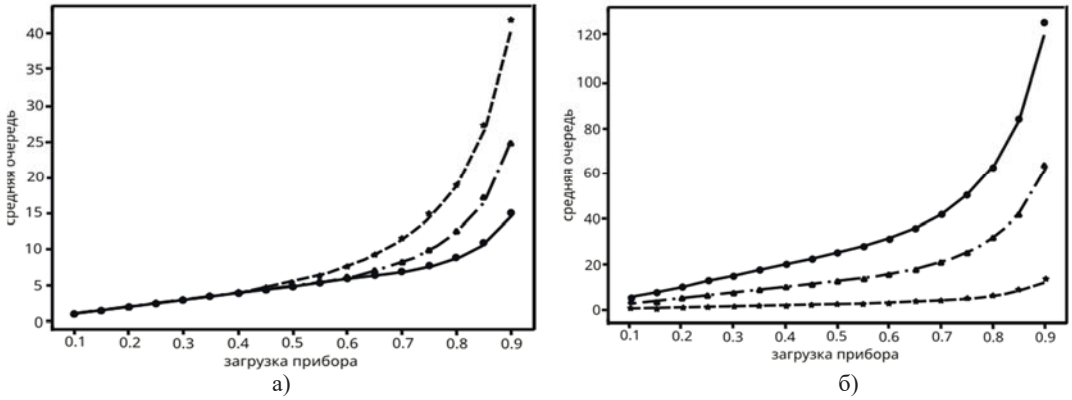


Рис. 5 Сопоставление результатов имитационного моделирования и расчётов по формуле (11) для зависимости средней длины очереди от коэффициента загрузки системы: а) все потоки имеют параметр *B* = 20; б) все потоки имеют параметр α = 1.

8.1 Парные групповые квазипуассоновские модели пачечного трафика

Парный групповой квазипуассоновский поток представляет собой модификацию группового пуассоновского потока, в которой каждый межпачечный интервал содержит дополнительную постоянную временную составляющую T. После окончания постоянной составляющей происходит генерация дополнительной пачки заявок. Следовательно, в рамках одного интервала имеется два момента прибытия пачек с размерами B_1 и B_2.

$$B_1 = \beta B,\ B_2 = (1-\beta)B,\ \ B_1 + B_2 = B,\ \beta \in [0,1].$$

Схема одного цикла парного группового квазипуассоновского потока представлена на рис. 6.

Рассматриваемая модель является модификацией группового квазипуассоновского потока с одинаковыми паузами. При B_2=0 эта модификация становится аналогичной исходному потоку. В данной модели отражено свойство реального трафика: большие пачки пакетов разделяются на набор нескольких меньших подпачек.

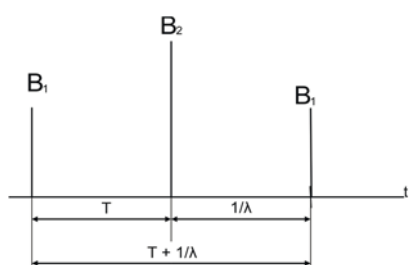


Рис. 6 Схематическое представление временного интервала парного группового квазипуассоновского потока в пределах одного цикла.

Такое разделение снижает пиковую нагрузку и даёт возможность более точно описывать временную протяжённость передачи пакетов в сети.

Вводится ограничение: постоянный интервал T не превышает средней длительности экспоненциальной составляющей для того, чтобы групповой поток не становился излишне детерминированным. Для выражения соотношения постоянного интервала к экспоненциальному используется выражение: $\alpha = \lambda T$. Выражая данный факт через неравенство, получаем: $\alpha \le 1$.

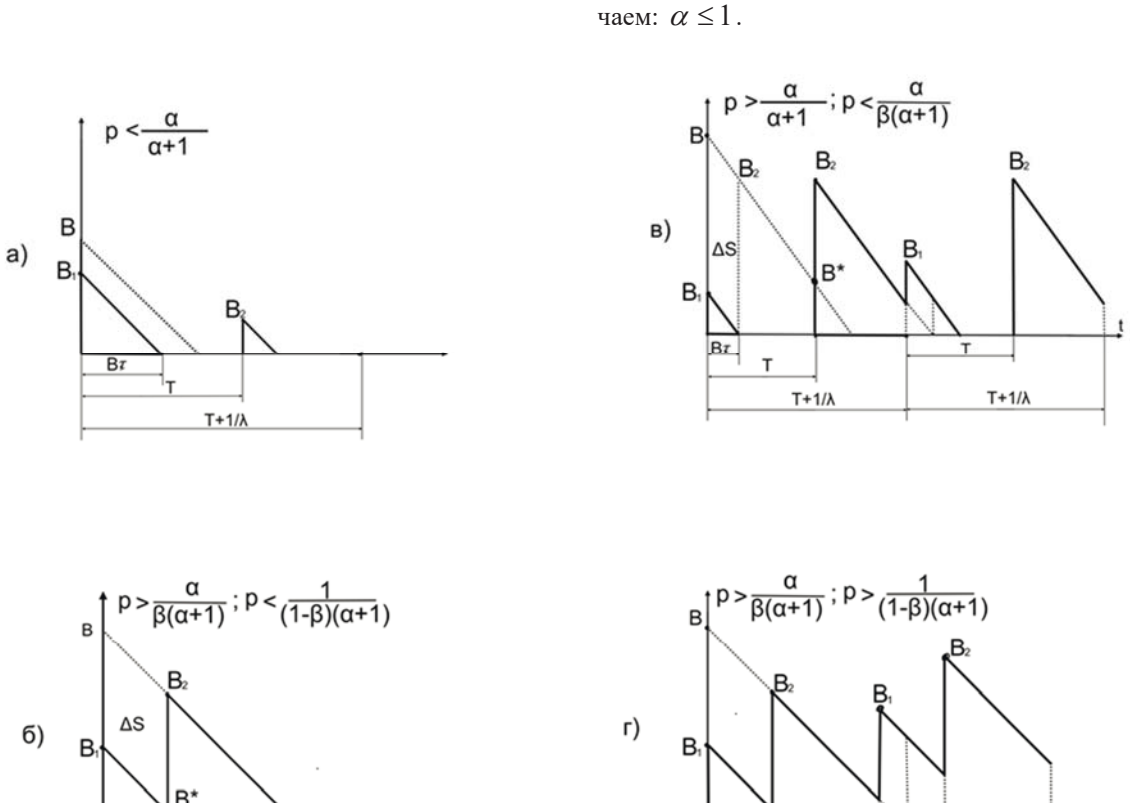


Рис. 7. Изменение числа заявок в системе для различных случаев обработки пачек B_1 и B_2:

а) $B = B_1 + B_2$ успевает обработаться за промежуток T; б) $B = B_1 + B_2$ не успевает обработаться за промежуток T, при этом пачка B_1 не успевает обрабатываться за промежуток T; в) Пачка $B = B_1 + B_2$ не успевает обработаться за промежуток T, при этом пачка B_1 успевает обрабатываться за промежуток T; г) $B = B_1 + B_2$ не успевает обработаться за промежуток T, при этом пачки B_1 и B_2 не успевают обработаться за соответствующие промежутки T и $\frac{1}{\lambda}$.

Для разработки аналитической модели использовались предположения, схожие с результатами модифицируемой модели, описанной в предыдущем разделе. Рисунок 7 иллюстрирует динамику изменения очереди во времени для анализируемой модели. Величина площади ΔS соответствует преобразованию треугольной области, используемой при вычислении поправочного коэффициента исходной модели.

В зависимости от уровня нагрузки возможно несколько случаев которые необходимы отразить в аналитической модели. В частности, рассматриваются следующие условия: успевает ли быть обслуженной пачка B до окончания промежутка T; успевают ли быть обслуженными B_1 и B_2 до окончания постоянного промежутка T и до окончания экспоненциального промежутка со средним значением $\frac{1}{\lambda}$ соответственно. В случае г), когда обе пачки B_1 и B_2 не успевают обслужиться в пределах своих интервалов, то очередь растет бесконечно, и определение её среднего значения аналитически теряет практическую ценность. Это связано с тем, что заявки поступают быстрее, чем обрабатываются, поэтому очередь неограниченно растет, а канал оказывается полностью загруженным. Усредненный размер очереди перестает отображать реальное состояние системы.

Таким образом, данные условия можно выразить аналитически с помощью следующих соотношений:

$$B\tau < T,$$
$$B_1\tau < T,$$
$$B_2\tau < \frac{1}{\lambda}.$$

С практической точки зрения данные условия целесообразно определять через коэффициент загрузки. Для этого можно использовать уже имеющиеся аналитические соотношения модифицируемой модели; они в дальнейшем также будут применяться при расчётах и для преобразования итоговых выражений:

$$\tau = \rho\frac{T+\frac{1}{\lambda}}{B}, \quad (12)$$

$$\rho^* = \frac{p-p_0}{1-p_0},$$

$$p_0 = \frac{\alpha}{\alpha+1} = \frac{T}{T+\frac{1}{\lambda}}, \quad (13)$$

$$B^* = \left\lceil B\left(1-\frac{\rho_0}{\rho}\right)\right\rceil, \quad (14)$$

Тогда ограничения для разных вариантов обработки пачек можно представить в виде следующих аналитических неравенств:

$$\rho < \frac{\alpha}{(\alpha+1)}, \quad (15)$$

$$\rho < \frac{\alpha}{\beta(\alpha+1)}, \quad (16)$$

$$\rho < \frac{1}{(1-\beta)(\alpha+1)}. \quad (17)$$

При этом, если выполняется условие (15), то выполнится и условие (16), так как $\beta \le 1$

В случае (а) возможен прямой расчёт средней очереди, исходя из поведения во времени. Цикл включает в себя несколько последовательных интервалов. Средняя длина очереди за цикл представляет собой усреднённую величину очереди за весь цикл работы системы.

Определим среднюю продолжительность цикла и коэффициент нагрузки как:

$$T+\frac{1}{\lambda} = \frac{1}{\lambda^*}.$$
$$p = \lambda^* B\tau. \quad (18)$$

На основании этого, средняя длина очереди для данного случая определяется следующим образом:

$$\bar{q}(\rho) = (S_{B1}+S_{B2})\lambda^* = \frac{(B_1^2\tau + B_2^2\tau)}{2}\lambda^* =$$
$$= \frac{(\beta^2B^2\tau + (1-\beta)^2B^2\tau)}{2}\lambda^*.$$

Если учесть (12) или (18), результирующим выражением средней очереди для случая, когда B успевает обработаться за промежуток T, является:

$$\bar{q}(\rho) = \frac{\rho B\left(1-2\beta+2\beta^2\right)}{2}.$$

Случаи (б) и (в) можно рассчитать, скорректировав формулу модифицируемой модели. Поправочный коэффициент определяется как:

$$K = \frac{\rho_0}{\rho}\left(2-\frac{\rho_0}{\rho}\right) = \frac{S^*}{S},$$

где S^* – площадь трапеции, включающая в себя площадь ΔS, а S – площадь треугольника, равная $S = \frac{TB^2}{2\left(B-B^*\right)}$.

Из рисунка 7 видно, что после разделения пачки на две составляющие, результирующее соотношение площадей для поправочного коэффициента принимает вид:

$$K = \frac{\Delta S}{S}.$$

При этом в зависимости от условий (16), (17) и (11) величина ΔS будет иметь различное значение.

$$\Delta S = \begin{cases} TB_2, \begin{cases} \rho > \dfrac{\alpha}{\beta(\alpha+1)}; \\ \rho < \dfrac{1}{(1-\beta)(\alpha+1)}. \end{cases} \\ B_2B_1\tau, \begin{cases} \rho < \dfrac{\alpha}{\beta(\alpha+1)}; \\ \rho > \dfrac{\alpha}{(\alpha+1)}. \end{cases} \end{cases}$$

Таким образом, коэффициент для случая, когда условия (15) и (16) не выполняются считается следующим образом:

$$K_{1new} = \frac{S^* - \Delta S}{S} = \frac{\dfrac{\left(B+B^*\right)T}{2} + TB_2}{\dfrac{TB^2}{2\left(B-B^*\right)}} = K - \frac{2TB_2\left(B-B^*\right)}{TB^2}.$$

Подставим выражение (14) в формулу выше, представив его в виде:

$$\frac{(B-B^*)}{B} = \frac{\rho_0}{\rho}.$$

В результате получаем:

$$K - 2(1-\beta)\left(\frac{\rho_0}{\rho}\right).$$

Если условия (15) и (17) выполняются, но условие (16) не выполняется:

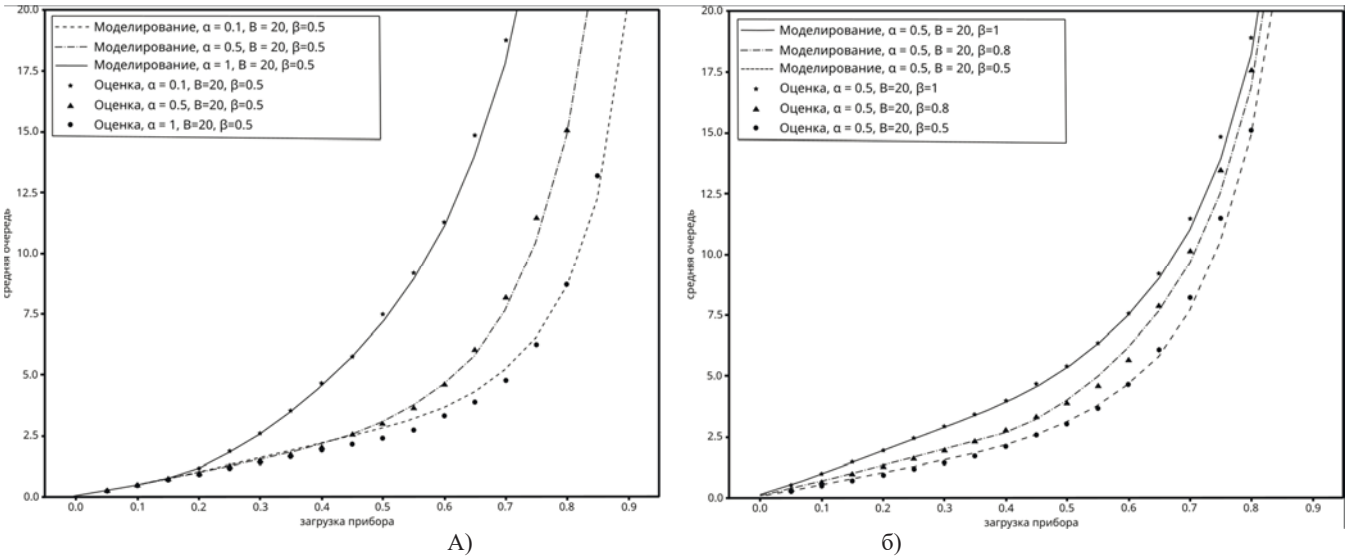
$$K_{2new} = \frac{S^* - \Delta S}{S} = \frac{\dfrac{\left(B+B^*\right)T}{2} + B_2B_1\tau}{\dfrac{TB^2}{2\left(B-B^*\right)}} = K - \frac{2B_1B_2\tau\left(B-B^*\right)}{TB^2} = K - \frac{2\beta(1-\beta)\tau\left(B-B^*\right)}{T}.$$

При учете выражений (12), (13) и (14) результирующей формулой для данного случая является: $K - 2\beta(1-\beta)$.

Тогда, если собрать все результирующие вычисления в одно аналитическое выражение, отражающее разрабатываемую модель, получим окончательно (19).

На рисунке 8 приведено сравнение результатов расчета средней длины очереди по формуле (19) с результатами имитационного эксперимента прохождения парного группового квазипуассоновского потока через СМО типа G/D/1. На рисунке 8а приведены примеры для фиксированного суммарного значения B и фиксированного значения α при различных значениях коэффициента β. Верхняя сплошная кривая отражает зависимость для значения α=1, средняя α=0.5, нижняя α=0.1 Коэффициент β определяет соотношение B_1 и B_2. В предельных случаях, при $\beta = 0$, $\beta = 1$ парный групповой квазипуассоновский поток сводится к модифицируемому групповому квазипуассоновскому потоку с одинаковыми паузами, а соответствующее аналитическое выражение становится аналогичным выражению (11). На рисунке 8б приведены примеры для фиксированного суммарного значения B и фиксированного значения β при различных значениях коэффициента α. Верхняя сплошная кривая отражает зависимость для значения β=1, средняя β=0.5, нижняя β=0.1 Полученное приближение можно признать вполне удовлетворительным.

$$\overline{q(\rho)} = \begin{cases} \dfrac{\rho B\left(1-2\beta+2\beta^2\right)}{2}, \begin{cases} \rho < \dfrac{\alpha}{(\alpha+1)} \end{cases} \\ \left(\dfrac{\rho_0}{\rho}\left(2-\dfrac{\rho_0}{\rho}\right) - 2(1-\beta)\left(\dfrac{\rho_0}{\rho}\right)\right)\dfrac{\rho B - \rho(1-\rho)}{2} + \dfrac{\rho^* B^* - \rho^*\left(1-\rho^*\right)}{2\left(1-\rho^*\right)}, \begin{cases} \rho > \dfrac{\alpha}{\beta(\alpha+1)} \\ \rho < \dfrac{1}{(1-\beta)(\alpha+1)} \end{cases} \\ \left(\dfrac{\rho_0}{\rho}\left(2-\dfrac{\rho_0}{\rho}\right) - 2\beta(1-\beta)\right)\dfrac{\rho B - \rho(1-\rho)}{2} + \dfrac{\rho^* B^* - \rho^*\left(1-\rho^*\right)}{2\left(1-\rho^*\right)}, \begin{cases} \rho > \dfrac{\alpha}{(\alpha+1)} \\ \rho < \dfrac{\alpha}{\beta(\alpha+1)} \end{cases} \end{cases} \quad (19)$$



А) б)

Рис. 8. Зависимость средней очереди от коэффициента загрузки прибора. Сопоставление результатов имитационного эксперимента (сплошные кривые) с вычислениями по формуле (13), указанные с помощью маркеров: а) Для потоков β = 0,5; б) Для потоков α = 0,5.

9 Гиперэкспоненциальные групповые модели трафика

Последовательностью независимых событий, в которой промежутки времени между событиями распределены по гиперэкспоненциальному закону, называется групповой гиперэкспоненциальный поток. При этом каждый промежуток трактуется как отдельная фаза, а порядок следования фаз случаен и не зависит от предыдущих фаз.

Как и в описанных выше групповых потоках событием выступает одиночная заявка. В групповом варианте в каждый момент времени t_k возникает пачка B_k случайных независимых заявок. В данном случае рассматривается поток второго порядка, в котором предполагается всего две фазы, следующие друг за другом с вероятностями p и противоположной $1-p$, соответственно. Законы распределения числа заявок в пачках на разных фазах могут различаться и описываться с помощью двух разных законов обозначаемых f_{1k} и f_{2k}. Функция распределения интервалов между соседними заявками для такого потока имеет вид:

$$F(\theta) = 1 - pe^{(-\lambda_1\theta)} - (1-p)e^{(-\lambda_2\theta)} \qquad (20)$$

Здесь, λ_1 и λ_2 это параметры интенсивности для каждой из фаз, а θ это случайный интервал времени между событиями.

При разработке данной модели предполагалась СМО H2/H2/1, в которой заявки поступают и обслуживаются по гиперэкспоненциальным законам второго порядка. С учетом (20) поток обслуживания данной системы задан распределением вида:

$$\Phi(\tau) = 1 - b_1 e^{(-B_1\tau)} - b_2 e^{(-B_2\tau)}$$

Данная модель исходит из предположения об отсутствии корреляционных зависимостей в потоках, а также о полной независимости поступающих заявок и длительностей их обслуживания. Это сужает сферу применимости гиперэкспоненциальных моделей при анализе реальных потоков в мультисервисных сетях связи.

Для иллюстрации свойств ординарного и группового гиперэкспоненциальных потоков. представлены результаты имитационного моделирования для СМО с детерминированным (постоянным) временем обслуживания на рисунке 9. Сопоставляются два потока:

Групповой гиперэкспоненциальный поток с параметрами λ1=3, λ2=0.6, ρ = 0.5, размер пачки B=10 и постоянен.

Ординарный гиперэкспоненциальный поток с параметрами λ1=30, λ2=6, ρ = 0.5

Из рисунка видно, что для обоих типов потоков угол наклона кривой, отражающей зависимость средней длины очереди от коэффициента загрузки ρ в малых диапазонах коэффициент ρ оказывается близким к нулю.

Для гиперэкспоненциальных моделей (включая групповые разновидности с пачками небольшого размера) характерна лишь ограниченная пригодность к аппроксимации характеристик очередей в мультисервисных системах, что подтверждается описанным выше обстоятельством.

На рисунке 9 для обоих типов потоков в области малых значений коэффициента загрузки наклон кривой (зависимость средней длины очереди от коэффициента загрузки) близок к нулю.

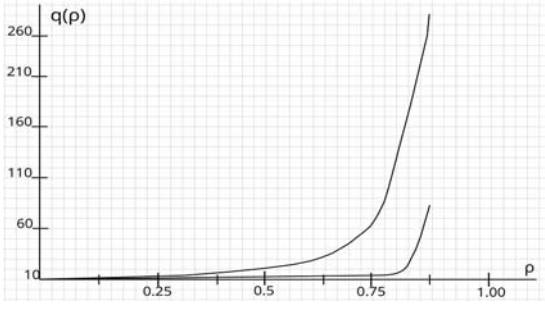


Рис. 9. Графики зависимости средней очереди от коэффициента загрузки ρ: верхняя кривая – ординарный гиперэкспоненциальный поток (p = 0,5, λ_1 = 30, λ_2 = 6), нижняя кривая – групповой гиперэкспоненциальный поток (p = 0,5, λ_1 = 3, λ_2 = 0,6, фиксированный размер пачки 10)

10 Аппроксимация очередей реального видеотрафика

В ходе имитационного моделирования одноканальной СМО в качестве входного потока использовалась реальная трасса видеоданных, полученная с помощью кодека H.264-1000. Трасса содержала времена поступления пакетов и их размеры (в битах) и вводилась в виде текстового файла.

Путём варьирования скорости обслуживания задавались различные значения коэффициента загрузки ρ в диапазоне от 0,05 до 0,9. Длина очереди измерялась в пакетах, где за размер одного пакета принималось среднее по всей трассе значение. Средний размер очереди вычислялся экспериментально для каждого ρ, и на основе имитационного эксперимента для данных значений эмпирически находились величины:

$$\overline{q^*(\rho_i)}, i = 1...N_M,$$
$$D(q^*(\rho_j)), j = 1...N_D.$$

В то же время, аналитически рассчитывались величины $\overline{q(\rho_i)}, i = 1...N_M$ и $D(q(\rho_j)), j = 1...N_D$ с помощью следующих уравнений:

$$\overline{q(\rho)} = \frac{D(A(\tau))}{(2(1-\rho)} - \frac{\rho}{2} \qquad (21)$$

$$D(q) = \frac{\left(D\left(A(\tau)\right) - \rho(1-\rho)\right)\left(D\left(A(\tau)\right) + 2 - 3\rho + \rho^2\right)}{\left(4(1-\rho)^2\right)} + \frac{\left(\rho^3 + 3D\left(A(\tau)\right)\rho - 3D\left(A(\tau)\right) - 3\rho^2 + 2\rho\right)}{\left(3(1-\rho)\right)} + \frac{\mu_3\left(A(\tau)\right)}{\left(3(1-\rho)\right)} \qquad (22)$$

В данном контексте $A(\tau)$ это случайная величина, равная числу поступивших заявок на интервале τ. В [13] указан вывод моментов случайной величины. Ниже приводятся результаты:

$$\overline{A(\tau)} = \lambda\tau B = \rho$$
$$D(A(\tau)) = \lambda\tau B^2 = \rho B^2$$
$$\mu_3(A(\tau)) = \lambda\tau B^3 = \rho B^3$$

Для метода наименьших квадратов использовалась целевая функция от параметра B.

В случае, когда B=1 то формула становится выражением для ординарного пуассоновского потока:

$$\overline{q(\rho)} = \frac{\rho^2}{2(1-\rho)},$$
$$D(\rho) = \left(1 - \frac{\rho}{3} - \frac{\rho^2}{6}\right)\frac{\rho^2}{\left(2(1-\rho)^2\right)}. \qquad (23)$$

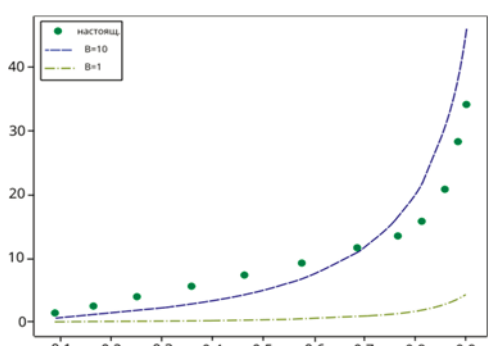


Рис. 10. Изменение средней длины очереди в функции коэффициента загрузки обслуживающего прибора.

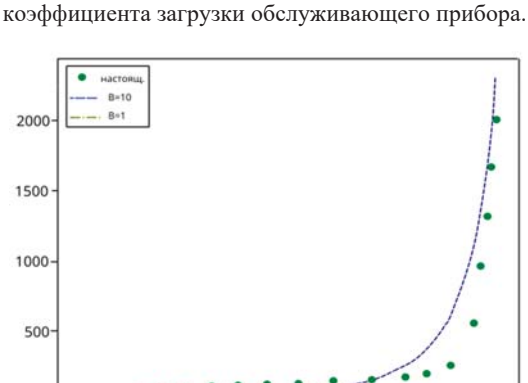


Рис. 11. Изменение дисперсии длины очереди в функции коэффициента загрузки обслуживающего прибора.

В рамках метода наименьших квадратов введена целевая функция, зависящая от параметра *B*. Указанная функция минимизировалась численно по параметру *B*, при условии сохранения заданного значения ρ, означающее соответствующее изменение интенсивности λ при изменении *B*.

$$J(B) = \sum_i^{N_M}\left(\overline{q^*(\rho_i)} - \overline{q(\rho_i)}\right)^2 + \sum_j^{N_D}\left(D\left(Q^*\left(\rho_j\right)\right) - D\left(Q\left(\rho_j\right)\right)\right)^2 \qquad (24)$$

Результаты вычислений отражены на рис. 10 и 11. На них представлены кривые, построенные согласно соотношениям (21) и (22) для того целого значения параметра *B*, которое находится ближе всего к точке минимума целевой функции (24). Для сравнения на тех же графиках приведены характеристики, соответствующие ординарному пуассоновскому потоку (случай $B = 1$ рассчитанные по формулам (23), а именно зависимости средней длины очереди и её дисперсии от коэффициента загрузки ρ) Сопоставление демонстрирует, что использование группового пуассоновского потока обеспечивает заметно более высокую точность аппроксимации для математического ожидания и дисперсии чем ординарный пуассоновский поток. Однако в результате всегда присутствовали отклонения и получить полное совпадение не удалось. Аналогичные закономерности обнаружены и для трасс, полученных с помощью других видеокодеков из семейства H.264.

Помимо указанного, были реализованы вычислительные эксперименты, в которых размер пачки в групповом пуассоновском потоке с заранее заданными вероятностями выбирался одним из двух возможных значений. В роли оптимизируемых параметров выступали сами значения размеров пачек и соответствующие им вероятности. Однако подобная модификация не обеспечила сколь-либо значимого увеличения точности аппроксимации.

11 Выводы

В работе предложены, разобраны и рассмотрены модели для пачечного телекоммуникационного трафика среди них: групповые пуассоновские потоки, групповые детерминированные и смешанные потоки, квазипуассоновские потоки с одинаковыми паузами, парные групповые квазипуассоновские потоки, а также групповые гиперэкспоненциальные потоки.

Для всех рассмотренных моделей приведены аналитические выражения для оценки средней длины очереди в системе массового обслуживания типа G/D/1, подтвержденные результатами имитационного моделирования. Данные потоки могут рассматриваться в качестве моделей, аппроксимирующих характеристики очередей реальных потоков, и предоставляют весьма простые соотношения для аналитических расчетов. Данные модели являются эффективным компромиссом между аналитической простотой и адекватностью описания реального пачечного трафика.

Литература

1. *Вишневский В.М., Дудин А.Н.* Системы массового обслуживания с коррелированными входными потоками и их применение для моделирования телекоммуникационных сетей // Автоматика и телемеханика. 2017. № 8. С. 3-59.

2. *Xiong Y.* Research and Application Analysis of the Basic Theory of Queuing Theory // Proceedings of the 2023 International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2023). Atlantis Press, 2023, pp. 454-463. DOI: 10.2991/978-94-6463-300-9_46.

3. *Назаров А.А., Лопухова С.В.* Полумарковские процессы и специальные потоки однородных событий: учебное пособие. Том. гос. ун-т, Ин-т дистанционного образования. Томск: ИДО ТГУ, 2010. URL: http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000405029.

4. *Ramaswami V.* The N/G/1 queue and its detailed analysis // Advances in Applied Probability, 1980. Vol. 12, no. 1, pp. 222-261. DOI: 10.2307/1426503

5. *Lakatos L., Szeidl L., Telek M.* Introduction to Queueing Systems with Telecommunication Applications // Springer Science+Business Media, 2013. 388 p.https://doi.org/10.1007/978-1-4614-5317-8

6. *Dudin A., Dudina O.* Analysis of a Multi-Server Queue with Group Service and Service Time Dependent on the Size of a Group as a Model of a Delivery System // Mathematics. 2023. Vol. 11, no. 22. P. 4587. DOI: 10.3390/math11224587.

7. *Агаларов Я.М.* Об однопороговом управлении очередью в системе массового обслуживания с нетерпеливыми заявками, Информ. и её примен., 18:2. 2024, pp. 40-46.

8. *Marcel F.* Neuts Matrix-analytic methods in queuing theory // European Journal of Operational Research. 1984. Vol. 15, Issue 1, pp. 2-12.

9. *Karagiannis T., Molle Mart, Faloutsos Michalis, Broido A.* A nonstationary Poisson view of Internet traffic. Proceedings – IEEE INFOCOM. 2004. 3, pp. 1558-1569 vol.3. 10.1109/INFCOM.2004.1354569.

10. *de Vega Rodrigo M., Spadaro S., Remiche M.-A., Careglio D., Barrantes J., Götz J.* On the Statistical Nature of highly-aggregated Internet Traffic. In: 4th International Workshop on Internet Performance, Simulation, Monitoring and Measurement, Salzburg, Austria. 2006.

11. *Лихтциндер Б.Я.* Трафик мультисервисных сетей доступа (интервальный анализ и проектирование). М.: Горячая Линия, 2018. 290 с.

12. *Лихтциндер Б.Я., Привалов А.Ю.* Обобщение формул для моментов очереди при неординарном пуассоновском потоке для очередей пакетов в системах телекоммуникаций // Пробл. передачи информ., 59:4. 2023. С. 32-37; Problems Inform. Transmission, 59:4 (2023), 243–248

13. *Лихтциндер Б.Я., Моисеев В.И., Привалов А.Ю.* Вторые моменты очередей в системах массового обслуживания с групповыми пуассоновскими потоками // Информационные технологии и нанотехнологии (ИТНТ-2023): сборник трудов по материалам IX Международной конференции и молодежной школы: в 6 т., Самара, 17-23 апреля 2023 га. Том 5. Самара: Самарский национальный исследовательский университет имени академика С.П. Королева, 2023. С. 50192. EDN TZARVO.

14. *Лихтциндер Б.Я., Моисеев В.И.* Групповые пуассоновские и гиперпуассоновские модели пакетного трафика // I-methods. 2022. Т. 14, № 3. EDN HMTGQI.

15. *Лихтциндер Б.Я., Привалов А.Ю.* Об использовании группового пуассоновского потока в имитационном моделировании современного видеотрафика // Инфокоммуникационные технологии. 2023. Т. 21, № 3 (83). С. 11-16.

16. *Likhttsinder B.Y., Bakai Yu.O.* Models of group poisson flows in telecommunication traffic control // Вестник Самарского государственного технического университета. Серия: Технические науки. 2020. Vol. 28, No. 3(67), pp. 75-89. DOI 10.14498/tech.2020.3.5. EDN RYAIHW.

17. *Шелухин О.И., Раковский Д.И.* Бинарная классификация многоатрибутных размеченных аномальных событий компьютерных систем с помощью алгоритма SVDD // Наукоемкие технологии в космических исследованиях Земли. 2021. Т. 13, № 2. С. 74-84. DOI: 10.36724/2409-5419-2021-13-2-74-84.

18. *Maslov A. A., Sebekin G. V., Stepanov S. N.* et al. Model of processes for joint maintenance of real-time multiservice traffic and elastic data traffic in a network of low-power mobile subscriber terminals based on high-throughput satellites // T-Comm: Телекоммуникации и транспорт. 2024. Vol. 18, No. 3, pp. 41-49. DOI 10.36724/2072-8735-2024-18-3-41-49. EDN UFIBHG.

GROUP FLOWS IN QUEUING SYSTEMS: MODELLING PACKET REALITY IN TELECOMMUNICATIONS

Boris Ya. Likhttsinder, *State University of Telecommunications and Informatics, Samara, Russia,* ***lixt@psuti.ru***
Mikhail P. Tetiuev, *State University of Telecommunications and Informatics, Samara, Russia,* ***yhyr@mail.ru***
Alexander P. Vasiliev, *Moscow Technical University of Communications and Informatics, Moscow, Russia*

Abstract
This paper considers queueing systems with batch (burst) arrival flows, in which requests arrive not individually but in batches. The goat of the work is to systematize models of batch flows and to obtain analytical expressions for the mean queue length and its variance in a single-server queue with constant service time (G/D/1). The methodology is based on the interval method of analysis, which describes non-ordinary flows via the distribution of the number of requests arriving during the service time of a single request. The following models are considered: a batch Poisson flow; a mixture of a batch Poisson flow with a background ordinary Poisson flow; a batch deterministic flow with constant inter-batch intervals; a mixed batch flow combining deterministic and Poisson components; a quasi-Poisson flow with constant pauses, where each inter-batch interval contains a constant component; a pairwise generalization of such a flow, in which two batches are generated per cycle; and a second-order hyperexponential batch flow. Different models allow approximating different types of real traffic behavior. For each model, analytical expressions relating the mean queue length to the load factor are obtained. The validity of the proposed models is confirmed by simulation results, which show a high degree of agreement between analytical and experimental data over the entire range of workloads. An approximation of real H.264 video traffic using a batch Poisson flow is also considered. It is shown that this model describes the mean queue length and its variance much more accurately than an ordinary Poisson flow. The proposed models represent an analytically simple yet sufficiently accurate tool for calculating queues in systems with batch arrivals, suitable for approximating real traffic.

Keywords:*. queuing systems, batch flows, batch Poisson flow, batch deterministic flow, mixed batch flow, quasi-Poisson flow, paired batch flow, hyperexponential flow, interval analysis method, Pollaczek-Khinchin formula, mean queue length, video traffic*

References

[1] V. M. Vishnevskii, A. N. Dudin, "Queueing systems with correlated arrival flows and their applications to modeling telecommunication networks", *Avtomat. i Telemekh*., 2017, no. 8, 3-59; Autom. Remote Control, 78:8 (2017), 1361-1403.
[2] Y. Xiong, "Research and application analysis of the basic theory of queuing theory," in *Advances in Computer Science Research, Dordrecht: Atlantis Press International BV*, 2023, pp. 454-463, doi: 10.2991/978-94-6463-300-9_46
[3] A. A. Nazarov and S. V. Lopukhova, "Semi-Markov processes and special flows of homogeneous events: a textbook," *Tomsk State University, Institute of Distance Education*, Tomsk, Russia, 2010. [Online]. Available: http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000405029
[4] V. Ramaswami, "The N/G/1 queue and its detailed analysis," *Adv. Appl. Probab*., vol. 12, no. 1, pp. 222-261, 1980, doi: 10.2307/1426503
[5] L. Lakatos, L. Szeidl, and M. Telek, Introduction to queueing systems with telecommunication applications, 2013th ed. New York, NY: Springer, 2012, doi: 10.1007/978-1-4614-5317-8
[6] S. Dudin and O. Dudina, "Analysis of a multi-server queue with group service and service time dependent on the size of a group as a model of a delivery system," *Mathematics*, vol. 11, no. 22, p. 4587, 2023, doi: 10.3390/math11224587
[7] "On single-threshold queue management in a queuing system with impatient customers," *Inform. Appl*., 2024, doi: 10.14357/19922264240206
[8] Neuts, Marcel F., 1984. "Matrix-analytic methods in queuing theory," *European Journal of Operational Research*, Elsevier, vol. 15(1), pp. 2-12, January.
[9] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A nonstationary poisson view of internet traffic," in *IEEE INFOCOM 2004*, 2004.
[10] M. de Vega, S. Spadaro, M. A. Remiche, D. Careglio, J. Barrantes, J. Goetz, "On the Statistical Nature of Highly-Aggregated Internet Traffic", *4th International Workshop on Internet Performance, Simulation, Monitoring and Measurement*, Salzburg, Austria, February 27-28, 2006.
[11] B.Ya. Lihtcinder, "Traffic of multiservice access networks (interval analysis and design)," Moscow: Goryachaya liniya – Telekom, 2018. 290 p. (In Russian)
[12] B. Y. Lichtzinder and A. Y. Privalov, "Generalization of formulas for queue length moments under nonordinary poissonian arrivals for batch queues in telecommunication systems," *Probl. Inf. Transm*., vol. 59, no. 4, pp. 243-248, 2023, doi: 10.1134/s003294602304004
[13] B. Ya. Likhttsinder, V. I. Moiseev, and A. Yu. Privalov, "Second moments of queues in queuing systems with group Poisson flows," in *ITNT-2023*, Samara, Russia, Apr. 17-23, 2023, vol. 5, p. 50192. [Online]. Available: https://www.elibrary.ru/item.asp?id=54700454 (Accessed: May 5, 2026).
[14] B. Ya. Likhttsinder and V. I. Moiseev, "Group Poisson and hyperPoisson arrival process teletraffic models," *I-methods,* vol. 14, no. 3, 2022.
[15] "Use of group Poisson flow in simulation modeling of modern video traffic," *Infokommunikacionnye Tehnol*., pp. 11-16, 2024., doi: 10.18469/ikt.2023.21.3.02
[16] B. Y. Likhttsinder and Y. O. Bakai, "Models of group poisson flows in telecommunica-tion traffic control," *Vestnik of Samara State Technical University. Technical Sciences Series*, vol. 28, no. 3, pp. 75-89, 2020., doi: 10.14498/tech.2020.3.5
[17] O. I. Sheluhin, D.I. Rakovskiy, "Binary classification of multi-attribute tagged data about anomalous events in computer systems using the SVDD algorithm," *H&ES Research*, vol. 13, no. 2, pp. 74-84, 2021. Doi: 10.36724/2409-5419-2021-13-2-74-84.
[18] A.A. Maslov, G.V. Sebekin, S.N. Stepanov, A.O. Shchurkov, A.P. Vasilyev, "Model of processes for joint maintenance of real-time multiservice traffic and elastic data traffic in a network of low-power mobile subscriber terminals based on high-throughput satellites," *T-Comm*. 2024. vol. 18, no.3, pp. 41-49. DOI: 10.36724/2072-8735-2024-18-3-41-49.

Information about authors:
Boris Ya. Likhttsinder, *Chief Researcher of the Infocommunication Technologies Research Laboratory, Professor of Networks and Communication Systems Department, Doctor of Technical Science., Povolzhskiy State University of Telecommunications and Informatics, Samara, Russia*
Mikhail P. Tetiuev, *Povolzhskiy State University of Telecommunications and Informatics, Samara, Russia*
Alexander P. Vasiliev, *Moscow Technical University of Communications and Informatics, Moscow, Russia*

ОТБОР ИНФОРМАТИВНЫХ РЕГРЕССОРОВ ДЛЯ ПОСТРОЕННЫХ ПО КРИТЕРИЮ МАКСИМАЛЬНОЙ СОГЛАСОВАННОСТИ ПОВЕДЕНИЯ МОДЕЛЕЙ

DOI: 10.36724/2072-8735-2026-20-6-44-51

Manuscript received 12 March 2026;
Accepted 24 May 2026

Носков Сергей Иванович,
Иркутский государственный университет путей сообщения, Иркутск, Россия,
sergey.noskov.57@mail.ru

Ключевые слова: *линейная регрессия, отбор регрессоров, метод максимальной согласованности, спецификация модели, линейно-булево программирование*

В работе рассматривается актуальная задача отбора наиболее значимых независимых переменных (факторов) в линейной регрессионной модели, параметры которой оцениваются на основе метода максимальной согласованности поведения (ММС). Известно, что в классическом регрессионном анализе качество модели традиционно оценивается величиной среднеквадратичной ошибки, суммы квадратов невязок или каких-то других функций потерь. ММС же ориентирован на максимизацию числа пар наблюдений, для которых знаки приращений фактических и расчётных значений зависимой переменной совпадают. Такой критерий, называемый критерием согласованности поведения (КСП), слабо реагирует на выбросы в данных, не требует предположений о нормальности распределения ошибок и позволяет отражать структурные закономерности в исходной информации. Они могут состоять, в частности, в направлениях изменения тенденций в характере анализируемого процесса. Однако до настоящего времени не были разработаны процедуры отбора регрессоров, основанных на специфике данного дискретного критерия. Это сильно ограничивало применение ММС в задачах с большим числом потенциальных независимых факторов. В статье впервые предлагается оригинальный алгоритм отбора регрессоров, основанный на последовательном включении переменных, максимизирующих вклад в прирост значения критерия согласованности поведения. Описывается пошаговая процедура прямого отбора. Для каждого кандидата решается задача частично-булева линейного программирования, позволяющая точно вычислить максимальное значение КСП при данном наборе регрессоров. Приводятся формальные постановки задач. Доказываются условия разрешимости (существование решения) и свойство монотонности критерия по включению регрессоров. На иллюстративном примере, включающем по десять наблюдений и регрессоров, демонстрируется работоспособность предложенного подхода. При этом алгоритм правильно выделяет все значимые переменные и устойчив к добавлению неинформативных (малозначимых) факторов. Результаты могут быть использованы при построении регрессионных моделей сложных технических и транспортных систем, в задачах прогнозирования нагрузки на телекоммуникационные сети, анализа пассажиропотоков и других прикладных областях. При этом крайне важно правильное воспроизведение направлений в изменении значений зависимой переменной.

Информация об авторе:
Носков Сергей Иванович, *доктор технических наук, профессор, Иркутский государственный университет путей сообщения, Иркутск, Россия. https://orcid.org/0000-0003-4097-2720*

Для цитирования:
Носков С. И. Отбор информативных регрессоров для построенных по критерию максимальной согласованности поведения моделей // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 44-51.

For citation:
S.I. Noskov, "Selection of informative regressors for models constructed using the maximum consistency criterion," *T-Comm*, 2026, vol. 20, no. 6, pp. 44-51. (*in Russian*)

Введение

Построение регрессионных моделей является одним из основных инструментов анализа данных в таких областях, как телекоммуникации, транспорт, экономика и техника, социальные науки [1, 2]. В практических задачах исследователь часто сталкивается с ситуацией, когда число потенциальных факторов (предикторов) весьма велико. Оно может существенно превышать количество доступных наблюдений. Многие из этих факторов могут также не оказывать заметного влияния на зависимую переменную. Включение в модель неинформативных регрессоров приводит к снижению точности оценок, ухудшению прогнозных свойств, увеличению дисперсии предсказаний и размытости в интерпретируемости результатов. Поэтому проблема отбора наиболее значимых независимых переменных – формирования спецификации модели – является одной из важнейших в регрессионном анализе [3].

Классические методы отбора условно регрессоров условно можно разделить на три основные группы.

Методы перебора подмножеств предполагают построение всех 2^m (где m –число потенциальных регрессоров) возможных моделей и выбор наилучшей по некоторому критерию (например, скорректированный критерий детерминации R^2, AIC, BIC). При больших m этот подход вычислительно трудно нереализуем.

Пошаговые методы включают в себя процедуры прямого включения, обратного исключения и комбинированные [1]. Они не гарантируют глобальной оптимальности, но, в основном благодаря приемлемой вычислительной сложности, широко используются на практике.

Методы регуляризации осуществляют отбор регрессоров путём добавления штрафного члена к функции потерь [4-6]. Наибольшую популярность здесь приобрёл метод Lasso, в котором минимизируется сумма квадратов остатков с L_1-штрафом, что позволяет обнулять коэффициенты при неинформативных переменных.

Перечисленные методы в основном ориентированы на квадратичную функцию потерь (либо, в случае регуляризации, на минимизацию среднеквадратичной ошибки с дополнительными ограничениями). Однако во многих прикладных задачах, особенно связанных с прогнозированием состояния сложных динамических систем (например, загрузка телекоммуникационных каналов, пассажиропотоки на транспорте, показатели надёжности технических устройств), не менее важным является воспроизведение самой структуры изменения выходной переменной – направления её роста или убывания, моментов смены тенденций, знаков приращений между наблюдениями. Именно на эту особенность ориентирован метод максимальной согласованности поведения (ММС), разработанный автором [7] и его непрерывная форма, не предполагающая решения задач с булевыми переменными [8].

В отличие от традиционных подходов, ММС оценивает неизвестные параметры регрессии из условия максимизации числа пар наблюдений, для которых знаки приращений расчётных и фактических значений зависимой переменной совпадают. ММС основан на критерии согласованности поведения (КСП) и обладает рядом привлекательных свойств: он устойчив к выбросам (робастность), не требует предположений о распределении ошибок, позволяет строить модели, хорошо воспроизводящие тренды в данных [9].

Однако до настоящего времени задача отбора информативных регрессоров для моделей, построенных по критерию согласованности поведения, прямо не рассматривалась. Классические процедуры отбора, основанные на сравнении остаточных сумм квадратов, F-статистик или информационных критериев, по-видимому, не могут быть непосредственно перенесены на случай дискретного критерия КСП. Более того, пошаговые методы, использующие квадратичные критерии, часто ошибаются при наличии выбросов или коррелированных шумовых переменных, тогда как ММС потенциально к ним более устойчив. Данная работа направлена на восполнение этого пробела.

Целью исследования является разработка метода отбора наиболее значимых регрессоров в линейной регрессионной модели, параметры которой оцениваются путём максимизации критерия согласованности поведения. Для достижения цели в работе:

– формулируется задача построения спецификации модели в терминах КСП;

– предлагается пошаговый алгоритм прямого включения на основе прироста критерия;

– приводится явная вычислительная схема в виде задачи линейно-булева программирования (ЛБП);

– описываются результаты вычислительного эксперимента, демонстрирующие работоспособность подхода.

1 Постановка задачи

Рассмотрим линейное регрессионное уравнение (модель):

$$y_k = \alpha_0 + \sum_{i=1}^{m} \alpha_i x_{ki} + \varepsilon_k,\ k = \overline{1,n},$$

где k – номер наблюдения, y – зависимая переменная; x_i – i-ая независимая переменная (предиктор, регрессор); α_i, $i = \overline{1,m}$ – неизвестные параметры модели; ε_k, $k = \overline{1,n}$ – ошибки аппроксимации; n – объём выборки; m – количество потенциальных регрессоров. Все переменные в (1) детерминированы. В векторно-матричной форме (1) представимо в форме:

$$\mathrm{y} = X\alpha + \varepsilon, \qquad (1)$$

где $\mathrm{y} = (y_1, \mathrm{j}\ \ , y_n)^T$, $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_m)^T$ X– матрица размерности $n \times (m+1)$ с элементами $x_{k0} = 1$, $k = \overline{1,n}$.

Предполагается, что исследователь располагает выборкой $(X,\ \mathrm{y})$ с точечными значениями всех переменных. Задача построения спецификации модели заключается в выборе подмножества индексов $S \subseteq \{1, 2, \ldots, m\}$ такого, что регрессионная модель, построенная только по регрессорам из S, обладает наилучшими свойствами в смысле некоторого критерия качества.

В данной работе в качестве основного критерия качества используется критерий согласованности поведения. Введём расчётные значения зависимой переменной:

$$y_k^* = \alpha_0 + \sum_{i=1}^{m} \alpha_i x_{ki}, k = \overline{1,n}.$$

Тогда КСП имеет вид:

$$\Phi(\alpha)=\sum_{k=1}^{n-1}\sum_{s=k+1}^{n}\sigma_{ks}, \qquad (2)$$

где

$$\sigma_{ks}=\begin{cases}1, & (y_k-y_s)(y_k^*-y_s^*)\geq 0,\\ 0, & \text{в противном случае.}\end{cases}$$

Величина $\Phi(\alpha)$ принимает целые значения от 0 до $N=n(n-1)/2$ и равна числу пар наблюдений, для которых знаки приращений фактических и расчётных значений совпадают. Иногда удобно также использовать относительный КСП:

$$\tilde{\Phi}(\alpha)=\frac{2\Phi(\alpha)}{n(n-1)}\in[0,1].$$

Метод максимальной согласованности заключается в нахождении вектора оценок $\hat{a}$, максимизирующего значение $\Phi(\alpha)$:

$$\hat{\alpha}=\arg\max_{\alpha\in\mathbb{R}^{m+1}}\Phi(\alpha). \qquad (3)$$

В работах [7, 8] показано, что задача (3) может быть сведена к задаче ЛБП.

Задача отбора регрессоров в контексте ММС формулируется следующим образом: найти такое подмножество $S^*\subseteq\{1,\ldots,m\}$, которое максимизирует значение КСП Φ_S, вычисленное по модели, построенной только по регрессорам из S:

$$S^*=\arg\max_{S\subseteq\{1,\ldots,m\}}\Phi_S. \qquad (4)$$

2 Отбор регрессоров на основе прироста КСП

Основная идея предлагаемого подхода состоит в следующем. Пусть имеется текущее множество отобранных регрессоров S (возможно, пустое). Соответствующая модель строится путём решения задачи максимизации КСП с использованием только регрессоров из S. Обозначим через $\Phi(S)$ максимальное значение КСП, достижимое для этого набора. Предлагаемый алгоритм последовательного включения состоит в следующем.

На каждом шаге для произвольного регрессора $i\Pi S$ вычисляется прирост критерия:

$$\Delta\Phi_i=\Phi(S\cup\{i\})-\Phi(S).$$

Регрессор с максимальным положительным приростом $\Delta\Phi_i>0$ включается в модель. Процедура останавливается, когда либо для всех оставшихся регрессоров этот прирост окажется неположительным ($\Delta\Phi_i\leq 0$), либо будет достигнуто заданное максимальное число регрессоров.

Для произвольного фиксированного множества регрессоров S (допустим, при этом $|S|=p$) задача максимизации КСП сводится к задаче линейно-булева программирования следующим образом [7, 8]. Введём неотрицательные вещественные переменные u_k, v_k, представляющие собой соответственно положительную и отрицательную части ошибок аппроксимации ε_k в (1):

$$u_k=\begin{cases}y_k-\left(\alpha_0+\sum_{i\in S}\alpha_i x_{ki}\right), & y_k>\alpha_0+\sum_{i\in S}\alpha_i x_{ki},\\ 0, & \text{в противном случае,}\end{cases}$$

$$v_k=\begin{cases}-y_k+\left(\alpha_0+\sum_{i\in S}\alpha_i x_{ki}\right), & y_k<\alpha_0+\sum_{i\in S}\alpha_i x_{ki},\\ 0, & \text{в противном случае.}\end{cases}$$

При этом $\varepsilon_k=u_k-v_k$, $u_k v_k=0$, $|\varepsilon_k|=u_k+v_k$.

Введём также булевы переменные $\sigma_{ks}\in\{0,1\}$, $k=\overline{1,n-1}$, $s=\overline{k+1,n}$. Тогда задача максимизации КСП для набора S принимает вид:

$$\max\sum_{k=1}^{n-1}\sum_{s=k+1}^{n}\sigma_{ks} \qquad (5)$$

при ограничениях

$$\sum_{i\in S\cup\{0\}}\alpha_i x_{ki}+u_k-v_k=y_k,\ k=\overline{1,n}, \qquad (6)$$

$$(y_k-y_s)\sum_{i\in S}\alpha_i(x_{ki}-x_{si})+M\sigma_{ks}\geq M,$$

$$k=\overline{1,n-1}, s=\overline{k+1,n}, \qquad (7)$$

$$\sigma_{ks}\in\{0,1\},\ k=\overline{1,n-1}, s=\overline{k+1,n}, \qquad (8)$$

$$u_k\geq 0, v_k\geq 0,\ k=\overline{1,n}. \qquad (9)$$

Здесь M – заранее выбранное большое отрицательное число. Неизвестными являются переменные a_i, u_k, v_k, s_{ks}. Целевая функция (5) максимизирует число совпадений знаков соответствующих приращений, а ее оптимальное значение равно $\Phi(S)$.

Заметим, что задача (5) – (9), как и любая задача ЛБП, является NP-трудной. Это явно указывает на отсутствие алгоритмов с полиномиальной временной сложностью для нахождения точного решения в общем виде. На практике это приводит к тому, что время вычислений может расти экспоненциально – в зависимости от размерности входных данных, в частности, от количества наблюдений и числа потенциальных регрессоров. Учитывая комбинаторную природу задачи, количество возможных вариантов сочетаний булевых переменных s_{ks} значительно увеличивает пространство поиска.

Тем не менее, для задач умеренной размерности применение современных решателей позволяет получать оптимальные значения критерия за приемлемое время. В случаях работы с массивами данных очень большого объема целесообразным становится использование специализированных эвристических процедур или методов декомпозиции.

Это могло бы позволить находить т.н. субоптимальные решения при существенном сокращении требуемых вычисли-

тельных мощностей и временных затрат на реализацию предложенного ниже алгоритма отбора.

Для пустого множества $S = \varnothing$ модель содержит только свободный член a_0. Для удобства при отборе регрессоров можно полагать $\Phi(\varnothing) = 0$.

Приведем теперь описание алгоритма, реализующего данный подход. Пусть заданы входные данные:

– в борка (X, у);

– максимальное допустимое число регрессоров $m_{\max}$ (если не задано, то $m_{\max} = m$).

Необходимо сформировать множество отобранных регрессоров S^*.

Шаг 1 (инициализация). $S := \varnothing$, $\Phi_{тек} := \Phi(\varnothing) = 0$.

Шаг 2 (основной цикл).

Повторять шаги 2.1–2.5.

Шаг 2.1 (поиск лучшего кандидата-регрессора)

Установить: $\Delta\Phi_{\max} := 0$, $i_{луч} := -1$ (ни один регрессор ещё не выбран).

Шаг 2.2 (перебор всех неиспользованных регрессоров).

Для каждого $i \in \{1,\ldots,m\} \setminus S$ реализуются действия:

2.2.1. Вычислить $\Phi(S \cup \{i\})$.

Для этого решается задача ЛБП (5) – (9). В результате получаем $\Phi_{нов} = \Phi(S \cup \{i\})$.

2.2.2. Вычислить прирост $\Delta\Phi := \Phi_{нов} - \Phi_{тек}$.

2.2.3. Если $\Delta\Phi > \Delta\Phi_{\max}$, то $\Delta\Phi_{\max} := \Delta\Phi$, $i_{луч} := i$.

Шаг 2.3.

Если $\Delta\Phi_{\max} = 0$, или $|S| \geq m_{\max}$, то перейти к шагу 3 (выход из алгоритма).

Шаг 2.4 (включение регрессора). $S := S \cup \{i_{луч}\}$, $\Phi_{тек} := \Phi(S)$.

Шаг 2.5 (продолжение цикла).

Перейти к началу шага 2 (следующая итерация).

Шаг 3 (завершение).

Объявить процесс формирования множества S^* завершенным.

Замечание. Вычислительная сложность одной итерации алгоритма – $O(m \cdot T)$, где T – сложность решения задачи ММС для фиксированного набора регрессоров. Общая сложность не превышает $O(m^2 \cdot T)$.

Действительно, на итерации с p отобранными регрессорами решаются $m - p$ задач, каждая сложностью $T(p+1)$. Суммируя по всем итерациям, получаем $\sum_{p=0}^{m-1}(m-p)T(p+1)$, что не равно m^2T при переменном $T(p)$. Таким образом, верхняя оценка – $O(m^2 \cdot T_{\max})$, где $T_{\max} = \max_p T(p)$.

Следовательно, общая вычислительная сложность алгоритма не превышает $O(m^2 \cdot T)$. Данная оценка является пессимистичной, так как на практике за счёт отсева части регрессоров и возможности использования оптимального решения с предыдущей итерации в качестве начального приближения для решателя реальное время вычислений может быть существенно меньше. Основной вклад в сложность вносит необходимость многократного решения задач ЛБП, что накладывает ограничения на применимость метода при больших m и n.

3 Теоретическое обоснование алгоритма

Утверждение 1. Для любого множества регрессоров S значение $\Phi(S)$ определено корректно, и задача его вычисления всегда имеет решение.

Доказательство. Задача максимизации $\Phi(S)$ для фиксированного множества регрессоров S эквивалентна задаче максимизации числа совпадений знаков соответствующих приращений, которая сводится к задаче ЛБП с замкнутым допустимым множеством. Ограничения (6) – (9) задают непустое множество, например, ее тривиальное решение со значениями: $u_k = y_k, v_k = 0,\ k = \overline{1,n},\ a_i = 0,\ i = \overline{0,m}$, при $s_{ks} = 1$, $k = \overline{1,n-1}, s = \overline{k+1,n}$. Целевая функция ограничена сверху числом N, поэтому максимум достигается. Доказано.

Утверждение 2. Функция $\Phi(S)$ монотонна по включению: добавление любого регрессора не может уменьшить максимальное значение КСП, т.е. $\Phi(S \cup \{i\}) \geq \Phi(S)$ для любого $i \notin S$.

Доказательство. Множество моделей, построенных по регрессорам из S, является подмножеством моделей, построенных по расширенному набору $S \cup \{i\}$ (достаточно положить коэффициент при добавленном регрессоре равным нулю). Следовательно, максимум по большему множеству не меньше. Доказано

Из утверждения 2 следует, что $\Delta\Phi_i \geq 0$ всегда, и процедура отбора может быть остановлена, когда прирост становится нулевым или достигнуто заданное максимальное число регрессоров.

4 Вычислительный эксперимент

Пусть исходная выборка при $n = m = 10$ сформирована случайным образом и представлена в табл. 1.

Таблица 1

Исходные данные

y	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
95	73	84	37	94	9	2	11	63	26	47
29	6	16	18	96	95	29	70	7	62	77
65	78	12	76	64	25	21	13	8	93	8
49	42	82	75	7	55	53	55	55	58	72
97	94	75	67	14	7	63	92	83	22	3
13	91	26	55	49	20	72	54	42	33	93
37	76	8	92	92	76	65	49	68	80	81
92	16	22	9	28	60	91	70	8	9	17
90	33	50	11	42	72	31	89	15	95	92
42	41	67	98	89	64	64	70	34	70	86

Приведённый вычислительный эксперимент носит сугубо иллюстративный характер. Исходные данные (табл. 1) и значения КСП в табл. 2-4 сформированы в качестве модельного

сценария, предназначенного для наглядной визуализации пошаговой логики алгоритма отбора регрессоров. В данном примере задачи ЛБП (5) – (9) с конкретными данными реально численно не решались. Числовые значения в таблицах 2-4 отражают качественную динамику изменения КСП и служат для пошагового разбора вычислительной схемы.

Назначим $m_{\max} = 3$.

Вначале в соответствии с описанным выше алгоритмом для каждого отдельного регрессора x_i, $i = \overline{1,m}$ вычислим максимальное значение КСП $\Phi(\{x_i\})$ и его прирост $\Delta\Phi_i$:

Таблица 2

Регрессор	$\Phi(\{x_i\})$	$\Delta\Phi_i$
x_1	24	24
x_2	30	30
x_3	27	27
x_4	30	30
x_5	33	33
x_6	28	28
x_7	27	27
x_8	27	27
x_9	26	26
x_{10}	34	34

Максимальный прирост даёт x_{10} ($\Delta\Phi = 34$). Включаем x_{10} в множество S: $S\{10\}$.

Затем для каждого i№10 вычислим $\Phi(\{10,i\})$ и соответствующий прирост:

Таблица 3

Регрессор	$\Phi(\{x_i\})$	$\Delta\Phi_i$
x_1	40	6
x_2	42	8
x_3	38	4
x_4	41	7
x_5	44	10
x_6	39	5
x_7	40	6
x_8	38	4
x_9	37	3

Наибольший прирост у x_5 ($\Delta\Phi = 10$). Таким образом, $S = \{10,5\}$, $\Phi(\{10,5\}) = 44$. Для каждого $i \notin \{10,5\}$ вычислим $\Phi(\{10,5,i\})$ и прирост:

Таблица 4

Регрессор	$\Phi(\{10,5,i\})$	$\Delta\Phi_i$
x_1	44	0
x_2	45	1
x_3	44	0
x_4	44	0
x_5	44	0
x_6	44	0
x_7	44	0
x_8	44	0

Положительный прирост даёт только x_2 ($\Delta\Phi = 1$). Получим $S = \{10,5,2\}$.

Достигнуто $m_{\max} = 3$, процедура останавливается.

Финальное множество отобранных регрессоров имеет вид:

$$S^* = \{x_2, x_5, x_{10}\}.$$

Значение критерия согласованности поведения для итоговой модели составляет $\Phi(S^*) = 45$ (максимально возможное, так как все 45 пар наблюдений имеют совпадающие знаки приращений).

Таким образом, рассмотренный модельный сценарий наглядно иллюстрирует, как предложенный алгоритм позволяет выделять информативные регрессоры из исходной совокупности, обеспечивая рост критерия согласованности поведения.

5 Обсуждение результатов

Основные преимущества описанного метода состоят в следующем.

1. Робастность. КСП зависит только от знаков приращений, модель относительно устойчива к выбросам [7-9].

2. Содержательная интерпретируемость. КСП напрямую оценивает способность модели воспроизводить направление изменений значений зависимой переменной.

3. Естественная регуляризация. Монотонность $\Phi(S)$ позволяет использовать максимальное число регрессоров как параметр остановки.

Укажем также ограничения по практическому применению предложенного подхода:

– вычислительная сложность при больших n и m (необходимость решения задачи ЛБП для каждого кандидата в регрессоры);

– пошаговая процедура не гарантирует глобальной оптимальности решения [1, 3].

Дальнейшие исследования могут быть направлены на разработку вычислительных схем комбинирования метода с информационными критериями [10, 11], а также его адаптацию для интервальных данных.

6 Прикладные аспекты применения метода в телекоммуникационных системах и на транспорте

Предложенный метод отбора регрессоров может быть эффективно применен в задачах прогнозирования загрузки телекоммуникационных каналов, анализа пассажиропотоков на основе данных интеллектуальных транспортных систем, моделирования надежности сетевого оборудования и восстановления его пропускной способности при аварийных ситуациях [12, 13]. Все эти задачи, как правило, характеризуются зашумленностью данных, наличием выбросов и пропусков.

В таких условиях классические методы отбора регрессоров теряют в эффективности. Пошаговая регрессия на основе F-статистик [14] предполагает нормальность распределения ошибок, что на реальных транспортных и телекоммуникационных данных часто не выполняется. Методы робастной регрессии [15, 16] частично решают проблему выбросов, но не предназначены для оптимизации непосредственно знаков приращений зависимой переменной.

Теоретическое обоснование предлагаемого метода опирается на доказанные в статье свойства: корректность вычисления максимального значения КСП для любого фиксированного набора регрессоров и монотонность критерия по включению переменных. В отличие от классических методов, где добавление неинформативного регрессора может снизить скорректированный коэффициент детерминации или увеличить значение AIC [17], предложенная процедура гарантирует неухудшение значения КСП при расширении набора регрессоров.

Вычислительная сложность метода определяется необходимостью многократного решения задач линейно-булева программирования. Для размерностей, характерных для упомянутых задач (десятки наблюдений, 20-30 потенциальных регрессоров), современные средства (Gurobi, CPLEX, SCIP) позволяют получить решение за приемлемое время. Для более крупных задач требуется привлечение эффективных подходов, например, алгоритма LARS [18] для предварительного снижения размерности пространства признаков, а также параллельные вычисления.

Критерий согласованности поведения, используемый в предлагаемом методе, учитывает только знаки приращений для зависимой переменной. Вследствие этого единичные аномальные наблюдения не оказывают существенного влияния ни на оценку параметров, ни на процесс отбора регрессоров. Как показано в работах по робастной статистике [15, 16], знаковые критерии обладают высоким порогом развала и сохраняют состоятельность при значительной доле засорения данных.

Методы регуляризации, включая Lasso (предложенный в [19]), чувствительны к аномальным наблюдениям, а информационные критерии (в частности, AIC [20]) опираются на предположение о нормальности распределения остатков и могут давать ненадёжные результаты.

В ряде практических задач, включая краткосрочное прогнозирование загрузки каналов связи и оперативное управление транспортными потоками, более важным является правильное предсказание направления изменения зависимой переменной (рост или снижение), чем абсолютное численное значение. Оперативные решения иногда принимаются именно на основе анализа знака прогнозируемого изменения.

Перспективным направлением дальнейших исследований видится разработка модификаций информационных критериев (AIC, BIC), адаптированных для моделей, построенных по критерию согласованности поведения. В отличие от классического случая, где AIC/BIC опираются на функцию правдоподобия и сумму квадратов остатков, для ММС требуется создание информационных критериев, основанных на знаковых статистиках. Это позволило бы использовать их для выбора сложности модели (остановки пошаговой процедуры) без привлечения предположений о распределении ошибок.

Перспективным направлением дальнейших исследований является сравнительный анализ предложенного метода с классическими подходами отбора регрессоров (Lasso, пошаговая регрессия) на реальных данных для транспортных и телекоммуникационных систем. Теоретически метод максимальной согласованности, ориентированный на знаки приращений, обладает достаточно высокой робастностью к выбросам. Однако количественное сравнение на открытых тестовых наборах данных требует отдельного исследования [21]. В данной статье работоспособность метода продемонстрирована на иллюстративном примере.

Заключение

В работе впервые предложен метод отбора регрессоров для линейных моделей, построенных по критерию согласованности поведения. В отличие от классических подходов, процедура учитывает знаки приращений зависимой переменной – это особенно важно для зашумлённых данных с выбросами. Доказаны два свойства: корректность вычисления максимума критерия для любого набора регрессоров и монотонность по включению переменных. Эти свойства обосновывают возможность пошагового прямого отбора без риска ухудшения модели.

На иллюстративном примере алгоритм выделил значимые переменные. Работоспособность подхода показана на модельных данных. Как метод ведёт себя на реальных зашумлённых выборках с выбросами – отдельный вопрос, требующий дальнейших экспериментов и сравнения с традиционными методами. Области возможного применения: прогноз загрузки сетей, анализ пассажиропотоков, моделирование надёжности – там, где важна правильность именно выявления направления возможных изменений.

Дальнейшие исследования могут быть связаны со снижением трудоёмкости за счёт эвристик, параллельных вычислений или отсева неинформативных регрессоров. Интересным представляется комбинирование с AIC/BIC для автоматического выбора порога остановки. Возможны также адаптация к интервальным данным и разработка обратного и комбинированного отбора.

Решение этих задач расширит практическое применение метода. Наиболее востребованным такое расширение может оказаться в задачах анализа данных для экономических и технических систем, в частности, на транспорте и в телекоммуникациях.

Литература

1. *Montgomery D. C., Peck E. A., Vining G. G.* Introduction to Linear Regression Analysis. 5th ed. Hoboken: Wiley, 2012. 672 p.
2. *Schoon E. W., Melamed D., Breiger R. L.* Regression Inside Out. Cambridge: Cambridge University Press, 2024. 282 p.
3. *Hastie T., Tibshirani R., Wainwright M.* Statistical Learning with Sparsity: The Lasso and Generalizations. Boca Raton: CRC Press LLC, 2015. 368 p.
4. *Friedman J., Hastie T., Tibshirani R.* Regularization Paths for Generalized Linear Models via Coordinate Descent // Journal of Statistical Software. 2010. Vol. 33, No. 1, pp. 1-22.
5. *Breheny P., Huang J.* High-Dimensional Regression Modeling: Methodology, Applications, and Software. Boca Raton, FL: CRC Press, 2024. 450 p.
6. *Tibshirani R. J., Taylor J.* The solution path of the generalized lasso // The Annals of Statistics. 2011. Vol. 39, No. 3, pp. 1335-1371.
7. *Носков С. И.* Метод максимальной согласованности в регрессионном анализе // Известия Тульского государственного университета. Технические науки. 2021. № 10. С. 380-385.
8. *Носков С. И., Бычков Ю. А.* Вычислительные эксперименты с непрерывной формой метода максимальной согласованности в регрессионном анализе // Вестник Воронежского государственного технического университета. 2022. Т. 18. № 2. С. 7-12.
9. *Носков С. И.* Критерий «согласованность поведения» в регрессионном анализе // Современные технологии. Системный

анализ. Моделирование. 2013. № 1 (37). С. 107-110.

10. *Burnham K. P., Anderson D. R.* Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach. 2nd ed. New York: Springer, 2002. 488 p.

11. *Neath A. A., Cavanaugh J. E.* The Bayesian information criterion: background, derivation, and applications // Wiley Interdisciplinary Reviews: Computational Statistics. 2012. Vol. 4, No. 2, pp. 199-203.

12. *Бугаев А.С., Таташев А.Г., Яшина М.В., Лавров О.С., Носов Е.А.* Восстановление динамики транспортного потока на основе детерминированно-стохастической модели и данных с интеллектуально транспортных систем // T-Comm: Телекоммуникации и транспорт. 2019. Т. 13. № 10. С. 35-44.

13. *Буслаев А.П., Кучелев Д.А., Яшина М.В.* Динамические системы и математические модели трафика информации // T-Comm: Телекоммуникации и транспорт. 2018. Т. 12. № 3. С. 22-38.

14. *Miller A.J.* Subset Selection in Regression. 2nd ed. Boca Raton: Chapman & Hall/CRC, 2002. 256 p.

15. *Huber P.J., Ronchetti E.M.* Robust Statistics. 2nd ed. Hoboken: Wiley, 2011. 384 p.

16. *Rousseeuw P.J., Leroy A.M.* Robust Regression and Outlier Detection. New York: Wiley, 1987. 352 p.

17. *James G., Witten D., Hastie T., Tibshirani R.* An Introduction to Statistical Learning. 2nd ed. New York: Springer, 2021. 607 p.

18. *Efron B., Hastie T., Johnstone I., Tibshirani R.* Least Angle Regression // The Annals of Statistics. 2004. Vol. 32. No. 2, pp. 407-499.

19. *Tibshirani R.* Regression Shrinkage and Selection via the Lasso // Journal of the Royal Statistical Society: Series B. 1996. Vol. 58. No. 1, pp. 267-288.

20. *Akaike H.* A new look at the statistical model identification // IEEE Transactions on Automatic Control. 1974. Vol. 19. No. 6, pp. 716-723.

21. *Kuhn M., Johnson K.* Applied Predictive Modeling. New York: Springer, 2013. 600 p.

SELECTION OF INFORMATIVE REGRESSORS FOR MODELS CONSTRUCTED USING THE MAXIMUM CONSISTENCY CRITERION

Sergey I. Noskov, *Irkutsk state transport university, Irkutsk, Russia,* ***sergey.noskov.57@mail.ru***

Abstract

This paper addresses the timely problem of selecting the most significant independent variables (regressors) in a linear regression model, whose parameters are estimated using the maximum consistency method (MCM). Unlike classical regression analysis, where model quality is traditionally assessed by minimizing the mean squared error, the sum of squared residuals, or other loss functions, MMC focuses on maximizing the number of observation pairs for which the signs of the increments of the observed and estimated values of the dependent variable coincide. This criterion, known as the consistency of behavior criterion (CBC), is highly robust to outliers and anomalous values, does not require assumptions about the normality of the error distribution, and allows for the reproduction of structural patterns in the data, including trend directions and inflection points. However, until now, there have been no regressor selection procedures adapted to the specifics of this discrete criterion, which has limited the application of MMS in problems with a large number of potential factors. This paper proposes, for the first time, an original algorithm for selecting regressors based on the sequential inclusion of variables that maximize the contribution to the increase in the value of the consistency criterion. This criterion, known as the consistency of behavior criterion (CBC), is highly robust to outliers and anomalous values, does not require assumptions about the normality of the error distribution, and allows for the reproduction of structural patterns in the data, including the direction of trends and points of trend reversal. However, until now, there have been no regressor selection procedures adapted to the specifics of this discrete criterion, which has limited the application of MMS in problems with a large number of potential factors. This paper proposes, for the first time, an original algorithm for selecting regressors based on the sequential inclusion of variables that maximize the contribution to the increase in the value of the consistency criterion. A step-by-step procedure for direct selection is described; for each candidate, a semi-integer linear programming problem is solved, allowing the maximum value of the consistency criterion to be precisely computed for a given set of regressors. Formal formulations of the problems are provided, and the solvability conditions (existence of a solution) and the monotonicity property of the criterion with respect to the inclusion of regressors are proven. An illustrative example using synthetic data, comprising 10 observations and 10 predictors, demonstrates the effectiveness of the proposed approach: the algorithm correctly identifies all significant variables and is robust to the addition of non-informative factors. The results can be used in building regression models for complex technical and transportation systems, in tasks involving the forecasting of telecommunications network load, the analysis of passenger flows, and other applied areas characterized by noisy data and the importance of correctly reproducing the direction of changes in the dependent variable.

Keywords: *linear regression, feature selection, the method of maximum consistency, model specification, linear-Boolean programming*

References

[1] D. C. Montgomery, E. A. Peck, and G. G. Vining, Introduction to Linear Regression Analysis, 5th ed. Hoboken: Wiley, 2012.
[2] Schoon E. W., Melamed D., Breiger R. L. Regression Inside Out. Cambridge: Cambridge University Press, 2024. 282 p.
[3] T. Hastie, R. Tibshirani, and M. Wainwright, Statistical Learning with Sparsity: The Lasso and Generalizations. Boca Raton: CRC Press LLC, 2015.
[4] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization Paths for Generalized Linear Models via Coordinate Descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1-22, 2010.
[5] Breheny P., Huang J. High-Dimensional Regression Modeling: Methodology, Applications, and Software. Boca Raton, FL: CRC Press, 2024. 450 p.
[6] Tibshirani R. J., Taylor J. "The solution path of the generalized lasso," *The Annals of Statistics*. Vol. 39, No. 3, pp. 1335-1371, 2011
[7] S. I. Noskov, "Metod maksimal'noi soglasovannosti v regressionnom analize," *Izvestiya Tul'skogo gosudarstvennogo universiteta. Tekhnicheskie nauki,* no. 10, pp. 380-385, 2021.
[8] S. I. Noskov and Yu. A. Bychkov, "Vychislitel'nye eksperimenty s nepreryvnoi formoi metoda maksimal'noi soglasovannosti v regressionnom analize," *Vestnik Voronezhskogo gosudarstvennogo tekhnicheskogo universiteta*, vol. 18, no. 2, pp. 7-12, 2022.
[9] S. I. Noskov, "Kriterii 'soglasovannost povedeniya' v regressionnom analize," *Sovremennye tekhnologii. Sistemnyi analiz. Modelirovanie*, no. 1 (37), pp. 107-110, 2013.
[10] K. P. Burnham and D. R. Anderson, Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach, 2nd ed. New York: Springer, 2002.
[11] A. A. Neath and J. E. Cavanaugh, "The Bayesian information criterion: Background, derivation, and applications," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 4, no. 2, pp. 199-203, 2012.
[12] A.S. Bugaev, A.G. Tatashev, M.V. Yashina, O.S., Lavrov, E.A. Nosov, "Vosstanovlenie dinamiki transportnogo potoka na osnove determinirovanno-stoxastich-eskoj modeli i danny`x s intellektual`no transportny`x system," *T-Comm*, vol. 13., no. 10, pp. 35-44, 2019.
[13] A.S. Bugaev, D.A. Kuchelev, M.V. Yashina, "Dinamicheskie sistemy` i matematicheskie modeli trafika informacii," *T-Comm*, vol. 12, no. 3, pp. 22-38, 2018.
[14] Miller A.J. Subset Selection in Regression. 2nd ed. Boca Raton: Chapman & Hall/CRC, 2002.
[15] Huber P.J., Ronchetti E.M. Robust Statistics. 2nd ed. Hoboken: Wiley, 2009.
[16] Rousseeuw P.J., Leroy A.M. Robust Regression and Outlier Detection. New York: Wiley, 1987.
[17] James G., Witten D., Hastie T., Tibshirani R. An Introduction to Statistical Learning. 2nd ed. New York: Springer, 2021.
[18] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, "Least Angle Regression," *The Annals of Statistics*, Vol. 32, No. 2. pp. 407-499.2004.
[19] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society: Series B.* Vol. 58, No. 1, pp. 267-288,1996.
[20] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, Vol. 19, No. 6, pp. 716-723, 1974.
[21] M. Kuhn, K. Johnson, "Applied Predictive Modeling," New York: Springer, 2013.

Information about author:
Sergey I. Noskov, *PhD, Professor, Irkutsk state transport university, Irkutsk, Russia. https://orcid.org/0000-0003-4097-2720*

A WEBASSEMBLY-ENABLED FEDERATED LEARNING DEPLOYMENT FRAMEWORK FOR FOG COMPUTING ENVIRONMENTS

DOI: 10.36724/2072-8735-2026-20-6-52-62

Manuscript received 21 March 2026;
Accepted 26 May 2026

Dang Van Thang,
The Bonch-Bruevich Saint-Petersburg State University of Telecommunications (SPbSUT), St. Petersburg, Russia,
dangvanthang_sut@mail.ru

Andrey E. Koucheryavy,
The Bonch-Bruevich Saint-Petersburg State University of Telecommunications (SPbSUT), St. Petersburg, Russia,
akouch@mail.ru

Keywords: *WebAssembly, federated learning, fog computing, 6G networks, heterogeneous devices*

Federated learning (FL) in fog computing (FC) environments for 6G networks offers significant benefits across a wide range of domains by leveraging data locality to preserve the privacy of sensitive information, while also reducing communication costs and relieving server load. However, deploying FL tasks on fog nodes with heterogeneous hardware configurations, distinct operating systems, and diverse instruction set architectures remains a major challenge, especially given 6G's stringent latency requirements and tight integration with real-time applications. WebAssembly (Wasm) is a portable bytecode format that is secure through sandboxing and capability-based security, delivers near-native performance, has a compact footprint, and runs across platforms without recompilation, providing a potential means to overcome these limitations in 6G-enabled fog computing. In this study, we present a Wasm-integrated federated learning deployment framework for fog computing in 6G networks. To evaluate practical feasibility and performance, we conducted experimental evaluations on a heterogeneous set of devices. The analysis indicates that the Wasm-based framework reduces memory and CPU utilization while maintaining high accuracy, aligning with the performance requirements of 6G.

Information about authors:

Dang Van Thang, *The Bonch-Bruevich Saint-Petersburg State University of Telecommunications (SPbSUT), St. Petersburg, Russia. ORCID: https://orcid.org/0009-0009-2219-3767*

Andrey Koucheryavy, *The Bonch-Bruevich Saint-Petersburg State University of Telecommunications (SPbSUT), St. Petersburg, Russia. ORCID: https://orcid.org/0000-0003-4479-2479*

Для цитирования:
Данг В.Т., Кучерявый А.Е. Структура для развёртывания федеративного обучения на основе WebAssembly в гетерогенных туманных вычислениях // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 52-62.

For citation:
V.T. Dang, A.Y. Koucheryavy, "A WebAssembly-Enabled Federated Learning Deployment Framework for Fog Computing Environments," *T-Comm*, 2026, vol. 20, no. 6, pp. 52-62.

1. Introduction

Over the past decade, the traditional cloud computing model has been a principal driver of the artificial intelligence (AI) revolution. However, as AI applications increasingly penetrate domains requiring real-time interaction and the handling of sensitive data, particularly in 6G networks characterized by very high throughput and ultra-low latency, the inherent limitations of cloud-centric processing, including network latency and privacy risks, have become more apparent than ever. This has spurred a shift from centralized processing toward distributed paradigms such as edge computing and fog computing [1]. Functioning as an intermediate layer between edge devices and the cloud in 6G networks, fog computing extends computation and storage closer to data sources. By processing data locally, this architecture substantially reduces latency, conserves bandwidth, and supports applications such as telepresence, holography-type communications, autonomous vehicles, industrial IoT, and remote medical monitoring, thereby enabling immediate responses to real-world events in 6G environments [2]. Consequently, a tightly coupled Cloud–Fog–Client (IoT) infrastructure architecture has emerged [3], as illustrated in Fig. 1. Recent surveys confirm that fog computing is becoming a key enabler for latency-sensitive services such as telepresence and immersive communications in next-generation networks [4].

In parallel, advanced machine learning approaches have been intensively studied, developed, and deployed in practice. Among these, FL is particularly well suited to distributed computing models such as fog computing in 6G networks. FL enables multiple devices to collaboratively train a shared AI model without transmitting raw data to a central server. Instead, each device (fog node or client) performs local training and communicates only model parameter updates for aggregation. This approach addresses data-privacy concerns, reduces server-side computational load and communication costs, and aligns naturally with fog environments in which data are generated and stored across millions of distributed devices [5].

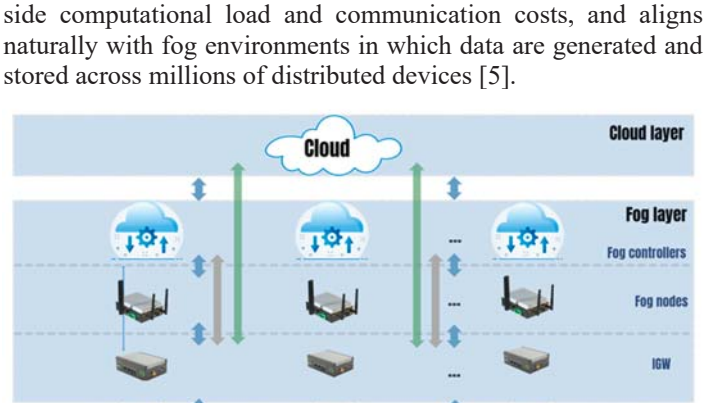


Fig. 1. Cloud–Fog–Edge devices infrastructure architecture

Despite the promise of combining FL with fog computing for 6G, practical deployment faces substantial challenges. Chief among these is heterogeneity across fog nodes: differing instruction set architectures (ISAs), diverse operating systems, and disparities in computational resources, CPU speeds, and RAM capacities—complications that are amplified by 6G's demands for high integration and multi-platform interoperability. Moreover, in fog computing environments, edge devices often operate under severe compute and network constraints. Fog nodes such as IoT devices, mobile devices, or edge servers may have weak CPUs, limited RAM (from a few megabytes to a few gigabytes), and unstable connectivity with low bandwidth or high latency. These constraints complicate FL deployment, as local training is resource-intensive for data processing and model updates, while communication of updates can congest the network [6].

Security is also a critical factor in user-facing service deployments. Although FL in fog computing keeps raw data on the device rather than sending it to a central location, inference attacks, poisoning attacks, and denial-of-service attacks remain significant risks.

To help address this problem, we propose applying Wasm to the deployment of federated learning in fog computing environments. Originally designed as a high-performance compilation target for web applications, Wasm has rapidly evolved into a versatile technology for out-of-browser settings, including fog computing, edge computing, and the Internet of Things (IoT). A principal advantage of deploying federated learning with Wasm is its high performance coupled with a compact bytecode footprint, which reduces the load on constrained computing resources by enabling fast execution without per-device recompilation. This is particularly beneficial for resource-limited fog nodes, where AI models can run efficiently without exhausting CPU or RAM. Moreover, the flexibility of Wasm, particularly its integration with lightweight runtimes such as Wasmtime or WasmEdge, makes it suitable for energy-constrained environments. This allows edge devices to operate for extended periods while ensuring safe and effective on-device processing of sensitive data. From a security perspective, adopting Wasm for federated learning in fog computing provides a substantial additional layer of protection. Wasm is built around a sandboxed security model, meaning that bytecode executes within a strictly isolated environment that prevents malicious code from accessing the host system, external memory, or a device's sensitive data. This design helps mitigate risks arising from vulnerability exploits or injection attacks, particularly on heterogeneous and attack-prone fog nodes.

This paper evaluates the feasibility and effectiveness of using Wasm as a platform for deploying FL in FC for 6G networks. Building on an analysis of key challenges - heterogeneous fog nodes, resource constraints, and security risk - we propose an integrated Wasm-based approach. The main contributions are as follows:

- Wasm-based FL deployment framework. We present a detailed framework that integrates FL with WebAssembly in fog computing, leveraging advantages such as high performance with compact bytecode, cross-platform deployment without recompilation, and integration with lightweight runtimes including Wasmtime and WasmEdge.
- Implementation and experiments on real devices. To validate the approach, we implement an experimental prototype emulating a fog environment on three devices: two Raspberry Pi 4 units and a Xiaomi 13 (Android), representing diverse architectures, operating systems, and resource profiles. FL training is performed on-device; parameter updates are transmitted over the network and aggregated centrally. The results show that Wasm achieves shorter processing times than native code while delivering near-equivalent performance, with reduced resource and memory

consumption. This makes it suitable for weaker fog nodes and supports stable operation under unstable network conditions.

• Performance analysis and future directions. We collect and analyze metrics from fog nodes, evaluating accuracy, loss, training time, and CPU/RAM utilization. Based on the results and related work, we outline extensions such as integrating advanced encryption and optimization for IoT in 6G networks.

The paper is organized into five parts: Section 1 introduces the background, the challenges of fog computing and FL in 6G, the proposed integration of WebAssembly, the objectives, and the contributions. Section 2 summarizes related work. Section 3 describes the architecture for deploying FL with WebAssembly in fog environments. Section 4 presents the experiments and evaluation on Raspberry Pi 4 and Xiaomi 13 devices, including performance analysis and trade-offs. Section 5 discusses security considerations and future research directions.

2 Overview of the Research Landscape

This section reviews the existing literature to delineate the context and the research gap that this work targets within the development of 6G networks. A central challenge is the need for distributed processing and for the deployment capabilities and scalability of AI models on heterogeneous, resource-constrained devices. The analysis is structured around four themes: (i) system-level challenges for federated learning (FL) at the edge; (ii) current approaches to managing heterogeneity; (iii) the role of Wasm as a universal runtime; and (iv) the positioning of the proposed framework's novelty.

2.1 System Challenges in Federated Learning within Fog Environments

Deploying FL in distributed settings such as fog computing faces a range of inherent system-level challenges that have been widely documented in survey literature.

• System and resource heterogeneity. This is among the most significant barriers. Fog and edge nodes differ substantially in hardware (CPU, memory), software (operating systems, ISAs), and network capabilities [7]. Such heterogeneity complicates deployment and leads to performance issues, for example, "stragglers" (slow nodes) that degrade the convergence speed of the overall FL system [8]. Moreover, these devices often operate under severe resource constraints, requiring algorithms and models to be highly efficient in computation and memory.

• Security and privacy: While FL is designed to protect privacy by keeping data local, other threats persist. Model updates can be attacked to infer sensitive training information (inference attacks) or maliciously altered to corrupt the global model (poisoning attacks). Consequently, a secure execution environment at each node is crucial.

2.2 Current Approaches to Managing Heterogeneity

Several approaches have been adopted to cope with heterogeneity, each with trade-offs in the context of fog computing.

• Cross-compilation and native deployment: This traditional approach compiles source code separately for each target platform. Although it can deliver optimal performance, it lacks portability and scalability. Maintaining distinct toolchains and build pipelines for every {ISA, OS} pair is costly and error-prone when managing a diverse device fleet [9].

• Virtualization and containerization: Container technologies such as Docker provide a consistent execution environment by packaging applications with their dependencies, effectively mitigating "dependency hell." However, this comes with notable resource costs. Containers often exhibit slower startup times and larger memory and disk footprints, making them ill-suited to resource-constrained IoT and edge devices [10]. In addition, containers remain architecture-dependent, typically requiring separate images for different architectures (e.g., one for ARM and another for x86).

• Existing FL frameworks: Popular frameworks such as TensorFlow Federated (TFF), PySyft, and Flower offer powerful tools for building the algorithmic logic of FL. Nevertheless, they primarily target the algorithmic layer and provide limited support for system-level deployment challenges. These frameworks often assume that execution environments are pre-provisioned on client devices, and support for less common architectures such as ARM may be limited or not prioritized.

2.3 WebAssembly: A Universal Runtime for Fog Computing

Wasm is emerging as a foundational technology capable of addressing the foregoing issues. Originally designed for the browser, Wasm has rapidly been recognized as a versatile, portable, and secure compilation target for out-of-browser environments [11].

• Standalone runtimes: The maturation of standalone runtimes such as Wasmtime (from the Bytecode Alliance) and WasmEdge (a CNCF project) is pivotal. These runtimes are optimized for server-side and edge applications, deliver high performance via JIT/AOT compilation, and support multiple platforms across both x86 and ARM architectures [12].

• WebAssembly System Interface (WASI): WASI is a core component that makes Wasm useful beyond the browser. It specifies a standardized, platform-independent system interface that enables Wasm modules to interact safely and in a controlled manner with host operating system resources (e.g., file systems, networking, clocks). Notably, WASI adopts capability-based security [13]. By default, a Wasm module has no permissions; the host must explicitly grant "capabilities" (e.g., access to a specific directory). This enforces the Principle of Least Privilege in a natural and robust way.

2.4 Positioning of This Work and the Research Gap

Building on the above analysis, this work is positioned to fill a critical research gap. Its distinctiveness and novelty are best understood in comparison with recent studies and the evolution of the Wasm ecosystem.

A clear bifurcation is emerging in the application of Wasm to machine learning: a web-centric approach and an infrastructure-centric approach. The web-centric direction is exemplified by the recent FLAT framework [14], which aims to allow any device with a browser to join an FL cluster without installation or configuration. FLAT addresses user accessibility and ease of participation; the browser acts as both runtime and sandbox, abstracting away OS/ISA complexity.

By contrast, this work pursues an infrastructure-centric

direction. The objective is to address the challenge of deploying a single codebase across headless fog nodes (e.g., Raspberry Pi or edge servers) that lack user interfaces. The problem tackled here concerns operational and infrastructural complexity for system administrators. In the proposed framework, the Wasm runtime (Wasmtime) is an explicit abstraction layer that must be deployed and managed on the host OS. Consequently, the focus of this study is on system-level performance relative to native code. Table 1 summarizes and contrasts the key characteristics of the four approaches discussed above, highlighting the distinct positioning of the proposed infrastructure-centric Wasm-FL framework.

In addition, the current Wasm ecosystem focuses heavily on accelerating inference through the WASI-NN specification [15], while offering limited support for training. This creates an important and urgent research gap. In a related line of work, the authors previously proposed a framework combining FL and fog computing using client sampling and dynamic thresholding techniques [16], which further motivates the present investigation into Wasm as a portable execution substrate for such systems. The present work contributes a potential solution by providing a performance baseline and a cross-platform operational model, addressing whether FL training with Wasm support in fog environments is feasible from a performance standpoint.

Table 1

Comparative Analysis of Techniques for Managing Heterogeneity in Federated Learning

Criterion	Native (Cross-Compilation)	Web-centric Wasm (FLAT)	Infrastructure-centric Wasm-FL (Proposed)
Portability	Very low. Requires recompilation for each {ISA, OS} pair	Very high. Runs in any modern browser	Very high. Single .wasm, any Wasm runtime
Safety & Isolation	Low. Executes directly on the host OS	Very high. Browser sandbox	Very high. Wasm sandbox, WASI capability-based security
Startup Time	Very fast	Fast	Very fast
Memory Footprint	Very small	Medium	Very small
Performance	Optimal (baseline)	Lower than native due to browser environment	Near-native
Target Environment	Servers, embedded devices	Web browsers, end-user devices	Headless fog/edge infrastructure nodes

3 Proposed Framework

This section presents a Wasm-based federated learning framework deployed in fog computing environments. It is designed to address device heterogeneity and execution safety, and to alleviate bottlenecks in IoT and telecommunications networks. The framework adopts the standard FedAvg paradigm, in which fog nodes act as collection points for data from edge devices and use Wasm as a portable, secure execution layer for local training.

3.1 System Model

We propose a three-tier physical FL architecture comprising a Sensor layer, a Fog layer, and a Cloud layer, as depicted in Figure 2, where:

- *Sensor Layer:* This layer consists of data-acquisition devices (e.g., cameras and sensors). We assume these sensors are locally connected to fog nodes (via LAN, USB, or LoRa) within communication range and continuously stream raw data to their associated fog node.
- *Fog Layer:* This layer corresponds to the clients in FedAvg and includes a set of fog nodes Fog $\mathcal{F}=\{1,\ldots,\mathrm{F}\}$, where $F=|\mathcal{F}|$. Each fog node $f\in\mathcal{F}$ (Raspberry Pi 4, NVIDIA Jetson,…) is responsible for collecting and locally storing the entire dataset D_f from the sensors within its domain. The dataset D_f is considered private and never leaves node *f*. Each fog node runs a Wasm runtime to execute the training module dispatched from the Cloud.
- *Cloud Layer:* A central server (orchestrator) coordinates the FedAvg process: selecting clients, distributing the model, aggregating updates, and redistributing the global model.

The central problem in federated learning is to find the global parameter vector *w** that optimizes the global loss *F(w)* over data distributed across fog nodes, without sharing raw data among nodes or with the Cloud. Formally,

$$\min_{w} F(w)=\sum_{f=1}^{F}\frac{N_f}{N}F_f(w), N=\sum_{f=1}^{F}N_f \qquad (1)$$

Where, $F_f(w)=\frac{1}{N_f}\sum_{(x,y)\in D_f}\ell(w;x,y)$ is the local loss on the private dataset D_f (of size $N_f=|D_f|$) at fog node *f*, $\ell(w;x,y)$ denotes the loss for a single sample, *F* is the number of fog nodes.

This constitutes a distributed optimization problem under data-privacy and heterogeneity constraints. Data at each fog node need not follow the global distribution and may vary in size; furthermore, fog nodes differ in instruction set architectures (ISAs), run different operating systems, and exhibit substantial disparities in computational resources, CPU speeds, and RAM capacities.

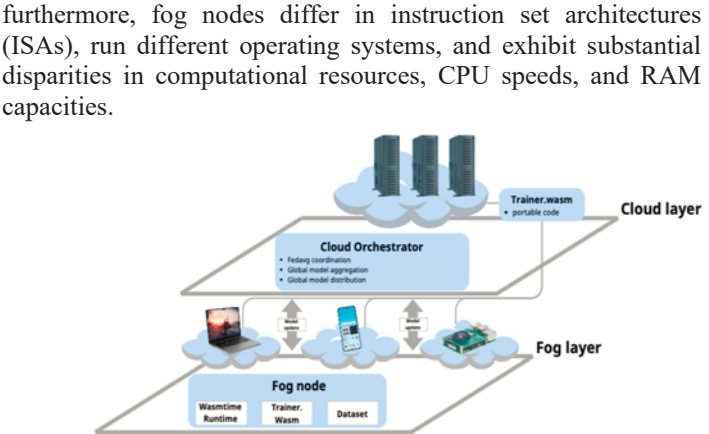


Fig. 2. WebAssembly-integrated system architecture

3.2 Wasm: a new, core component of the system

Wasm is, by design, a low-level machine-code format with standardized semantics, engineered to be independent of any specific embedded platform. This property enables the same module to execute across diverse architectures with deterministic and consistent behavior. In this study, the system is built to run on Wasmtime – a high-performance, standalone runtime developed by the Bytecode Alliance, a nonprofit whose participants include Mozilla, Fastly, Intel, and Red Hat. Wasmtime is designed around four core principles that make it well suited for production deployments [17]:

• *Fast:* Wasmtime integrates an optimizing compiler to produce high-quality machine code and supports both just-in-time (JIT) and ahead-of-time (AOT) compilation. The runtime is specifically optimized for rapid instance initialization and for minimizing overhead in host-guest function calls.

• *Secure:* Implementation in Rust provides a strong foundation for memory safety. The project undergoes continuous fuzz testing funded by Google OSS-Fuzz and follows a rigorous security-response process to patch vulnerabilities promptly.

• *Configurable:* Wasmtime exposes fine-grained configuration options for resource control, allowing limits on CPU and memory consumption per Wasm instance. This makes the runtime suitable for a wide range of deployment environments – from resource-constrained embedded devices to large servers handling many concurrent instances [18].

• *Standards Compliant:* Wasmtime passes the official WebAssembly test suite, and its developers actively participate in standardization efforts. A key commitment is to avoid proprietary extensions, thereby ensuring portability and compatibility of Wasm modules.

To support production readiness, Wasmtime classifies platform support into three tiers, enabling developers to assess deployment risk (Table 2).

Table 2

Wasmtime platform-support tiers

Tier	Description	Example platforms
Tier 1	Production-ready, fully tested, and comprehensively supported	x86_64-unknown-linux-gnu, x86_64-apple-darwin, x86_64-pc-windows-msvc
Tier 2	Near production-ready, well maintained but may lack some Tier-1 criteria	aarch64-unknown-linux-gnu, aarch64-apple-darwin
Tier 3	Supported but not widely tested; not recommended for production	riscv64gc-unknown-linux-gnu, x86_64-unknown-freebsd, aarch64-linux-android

By design, Wasm is a fully isolated computing environment. A Wasm module, by default, cannot perform I/O operations such as reading files, opening network connections, or accessing the system clock [19]. This isolation underpins Wasm's security model but also poses obstacles for practical applications. WASI was developed to address this issue by defining a standardized, platform-independent system interface that provides APIs for safe, controlled interaction between a Wasm module and the outside world. The goal of WASI is to extend Wasm's "write once, run anywhere" promise from the CPU level to the system level, enabling a single module to interact with different operating systems through the same API set.

The core design principle of WASI is capability-based security. Unlike traditional access models (e.g., POSIX), in which a program runs with the user's privileges and can attempt to access any resource, WASI adopts an inverted model: a Wasm module starts with zero authority and no ambient privileges. To access a resource, the host runtime must explicitly grant the module a capability – an unforgeable handle to that resource. The module can act only on resources for which it holds such a handle. This mechanism forces developers to specify required permissions up front and naturally enforces the Principle of Least Privilege [20], while eliminating the "confused deputy" problem common in traditional security systems.

Although Wasm supports multiple source languages (e.g., C/C++ and AssemblyScript), Rust remains the most popular choice due to its balance of performance, memory safety, and usability. The Rust-Wasm pairing is often described as ideal, not only for technical reasons but also for alignment in design philosophy.

• *Memory safely without a Garbage Collector:* Rust guarantees memory safety at compile time via ownership and the borrow checker, eliminating the need for a runtime garbage collector (GC). This aligns with Wasm, which is likewise designed without a built-in GC. Compiling Rust to Wasm yields compact modules that do not carry a bulky GC runtime, reducing binary size, speeding load times, and avoiding unpredictable pauses – thereby delivering stable performance [21].

• *High performance and Zero-Cost Abstractions:* Rust is engineered for performance on par with C/C++. Its hallmark "zero-cost abstractions" (e.g., iterators, closures, Option, Result) are fully optimized by the compiler into efficient machine code with no runtime overhead – matching Wasm's objective of near-native performance.

• *Mature tooling ecosystem:* The Rust community provides comprehensive Wasm tooling. Utilities such as wasm-pack automate build and packaging workflows (including npm-compatible bundles), while wasm-bindgen generates the glue code for seamless Rust-JavaScript interoperation. The cargo package manager simplifies dependency management.

• *A strategic fit across the stack:* Rust is well suited both for writing Wasm modules (guest code) and for building secure, high-performance Wasm runtimes (host code), with Wasmtime being a canonical example [22]. This enables a unified ecosystem in which developers can use the same language and tooling across the entire application stack.

The pipeline from Rust source to an executable WASI binary is illustrated in Figure 3.

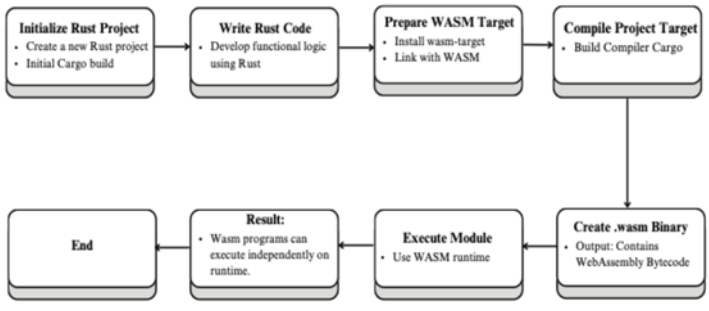


Fig. 3. Compilation pipeline from Rust source to a WASI binary

This pipeline highlights that the Rust-to-WASM toolchain is in a phase of transition and continued maturation. While tools such as wasm-pack are optimized for the web context and IoT edge-computing environments [23], compiling for standalone runtimes like Wasmtime is simpler and typically requires only standard cargo commands. This reflects the growing maturity of WASM as a first-class compilation target within the Rust ecosystem.

Building on the system components described above, the overall execution flow of the proposed framework is organized into the following steps.

• *Step 1 – Initialization.* The Cloud server initializes the global model parameters and compiles the training logic into a portable Wasm module (trainer.wasm) along with a configuration file (config.json) specifying the learning rate, batch size, and number of local epochs.

• *Step 2 – Client selection.* At the beginning of each communication round, the server selects a subset of available fog nodes based on a predefined participation policy.

• *Step 3 – Model and module distribution.* The server broadcasts the current global model parameters and the Wasm training module to all selected fog nodes.

• *Step 4 – Local training within Wasm sandbox.* Each fog node loads the received model and executes local training entirely within a Wasm sandbox. The sandbox enforces capability-based access control, granting the module read access to the local dataset directory and write access to the model output directory only, while blocking all other system resources. The private dataset never leaves the node.

• *Step 5 – Parameter transmission.* Upon completion of local training, each fog node transmits only the updated model parameters and the local dataset size back to the server. No raw data is shared.

• *Step 6 – Aggregation.* The server aggregates the received updates through weighted averaging proportional to each node's dataset size, producing an updated global model.

• *Step 7 – Iteration.* Steps 2 through 6 are repeated for a predefined number of communication rounds until the model converges or the round limit is reached.

4. Implementation and Performance Evaluation

This section details the experimental setup and analyzes results to quantitatively assess the effectiveness of the Wasm–based FL deployment framework relative to a native execution baseline. The objective is to answer the research questions posed: (i) the ability to preserve the convergence trajectory, (ii) system costs (time and resources), and (iii) factors influencing performance.

4.1. Experimental environment design

The experiments were conducted in the MegaNetlab research laboratory at The Bonch-Bruevich Saint Petersburg State University of Telecommunications [24]. The setup emulates federated learning in a fog-computing environment under two conditions: training on the Wasm platform and training with standard native code. The system follows a three-tier Cloud–Fog–Client architecture, in which:

• Cloud (Server): Acts as the central orchestrator running the FedAvg aggregation algorithm, distributing the global model and aggregating updates from fog nodes. The server is deployed on a MacBook Pro M1 with an 8-core CPU, 14-core GPU, 32 GB RAM, and a 1 TB SSD, ensuring fast processing of model aggregation tasks.

• Fog: Comprises three heterogeneous fog nodes in both hardware and operating systems: two Raspberry Pi 4 units (ARM64, Linux) and a Xiaomi 13 smartphone (ARM, Android). This diversity is intentionally chosen to represent the typical heterogeneity of real fog environments. Fog nodes perform local training on their private data and send gradient updates to the server.

• Sensor/Edge: Emulates raw data sources for the fog nodes. In this study, data are pre-partitioned for each fog node to simulate local collection from sensors.

Figure 4 illustrates the actual deployment configuration, where the three fog devices are connected to the server over a LAN, forming a complete FL system.

The experiment is designed with reasonable assumptions to focus on the core performance evaluation of Wasm. Communication between Cloud and Fog uses gRPC/TCP; no raw data are transmitted to the Cloud – only model updates (31.4 KB per direction per round). Fog nodes are assumed to have stable network connections to the server to eliminate noise from uncertain network latency, enabling precise measurement of pure computational overhead.

The data model is MNIST (10 classes), a standard dataset in FL research. Data are partitioned in a non-IID manner using a Dirichlet distribution with α=0.5 to emulate realistic imbalance: each fog node holds a subset with label skew. Specifically, nodes receive different numbers of samples and are concentrated on certain labels, reflecting characteristics of real-world distributed data. The learning model is multiclass logistic regression (Softmax) with 7,850 parameters (784 × 10 weights + 10 biases). This simple model is chosen to isolate and evaluate the overhead of the WebAssembly layer without confounding effects from optimizer complexity. The compact model size (31.4 KB) further minimizes the impact of network bandwidth on measurements.

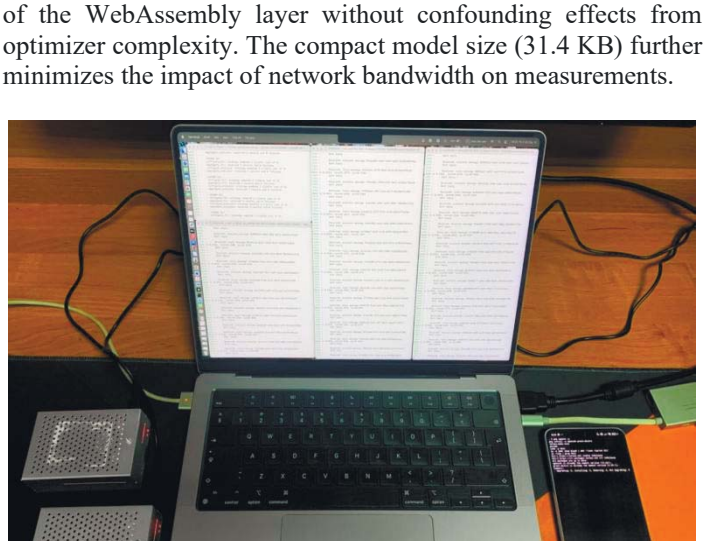


Fig. 4. Experimental deployment on physical devices

The core difference in the experiment lies in the execution layer. Local training code is written in Rust and compiled along two paths for comparison:

• Baseline (Native): Compiled directly to machine code for each specific architecture (aarch64 for Raspberry Pi/Xiaomi; x86_64 for other platforms).

• WebAssembly: Compiled to the wasm32-wasi target to produce the portable module trainer.wasm, which is then executed via the Wasmtime runtime on all fog nodes.

The Wasm environment is configured with strict capability-based security. The module is granted only the permissions needed to read data from a designated directory and write the model to an output directory; it has no access to the network or other file-system areas. This sandboxing ensures safety even if the training code fails or contains malicious components.

The evaluation metrics collected directly include: accuracy and loss for convergence; run time, throughput (samples/sec), maximum resident set size (max RSS), and initialization latency (elapsed sec) for system performance; as well as estimated CPU active time and CPU utilization (%CPU) for resource assessment.

4.2. Evaluation Results and Analysis

Accuracy: Experimental results show that both deployment methods achieve comparable convergence trajectories in terms of accuracy. After 10 training rounds, the native approach attains an average accuracy of 90.15% (±0.12%), while the Wasm approach reaches 89.86% (±0.15%), a difference of only 0.29%. Extending to 50 rounds, this gap narrows to 0.04% (Native: 91.25% ±0.08%, Wasm: 91.29% ±0.10%). At 100 rounds, Wasm achieves 91.49% (±0.09%) versus 91.49% (±0.07%) for Native, indicating high stability over long-horizon convergence (Fig. 5, 6). The small discrepancy may stem from numerical precision in the Wasm environment, but it does not materially affect overall performance.

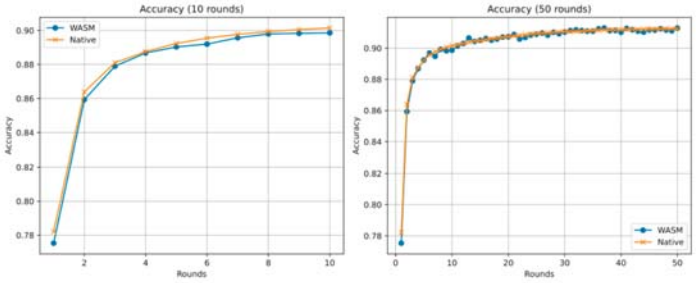


Fig. 5. Accuracy across training rounds (10 and 50 rounds)

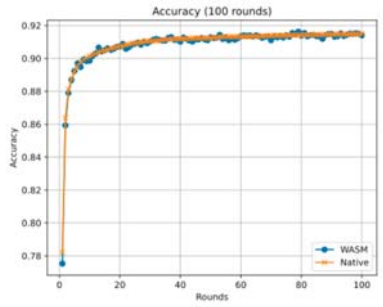


Fig. 6. Accuracy across training 100 rounds

Loss function: The loss-reduction trends of the two methods are nearly identical over training rounds (Fig. 7, 8). At round 10, Native records a loss of 0.339 (±0.005) versus 0.346 (±0.006) for Wasm. By round 50, both converge to approximately 0.296 (Native: 0.296 ±0.004, Wasm: 0.296 ±0.005). After 100 rounds, Native's loss is 0.286 (±0.003) and Wasm's is 0.289 (±0.004), a negligible difference. These results demonstrate that using Wasm does not affect the correctness of the FL algorithm or the model's ability to converge.

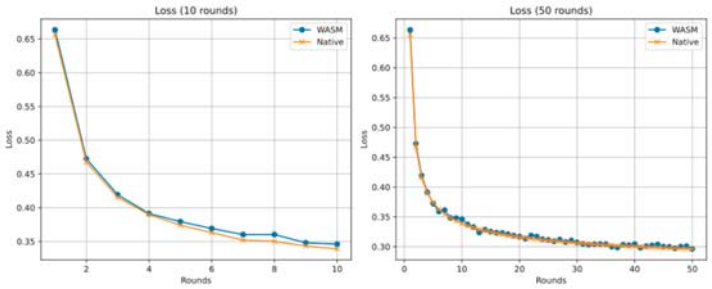


Fig. 7. Loss across training rounds (10 and 50 rounds)

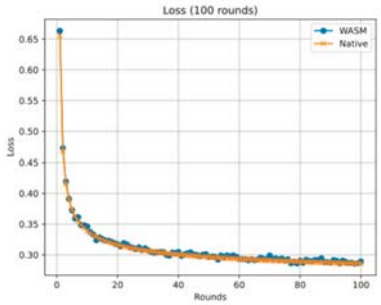


Fig 8. Loss across training 100 rounds

Training time results (Fig. 9) show that Wasm incurs a wall-clock overhead of 64.6% at 10 rounds (132.60 s vs. 80.51 s for Native), which decreases to 23.1% at 50 rounds (310.84 s vs. 252.45 s) and further to 10.2% at 100 rounds (514.00 s vs. 466.63 s). This trend reflects the amortization effect: as the number of rounds increases, the fixed per-round Wasm initialization cost occupies a progressively smaller share of the total training time, making Wasm increasingly time-competitive for longer workloads.

Throughput analysis (Fig. 10) shows that Native processes 120,990 samples/s (±5,120) at 10 rounds versus 51,460 samples/s (±1,230) for Wasm, a ratio of 2.35×. This ratio remains stable at 50 rounds (2.43×) and 100 rounds (2.46×), corresponding to a consistent computational throughput overhead of 135–146%. The stability of this ratio across all scenarios confirms that the throughput gap is a structural characteristic of the Wasm execution layer.

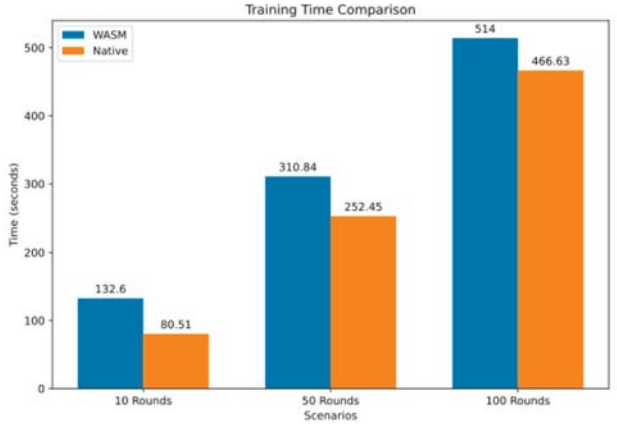


Fig. 9. Traning time comparison

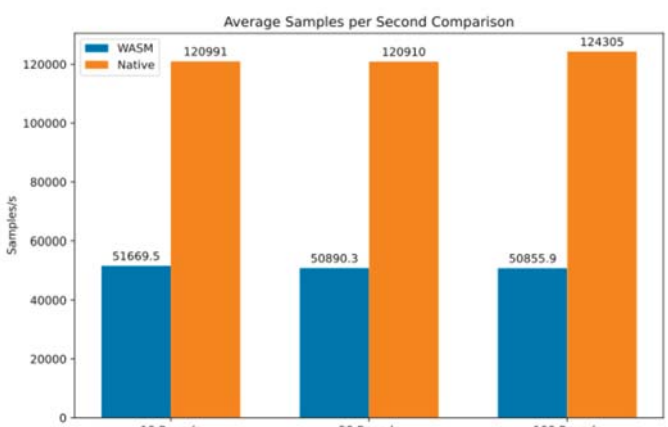


Fig. 10. Averege samples per second comparision

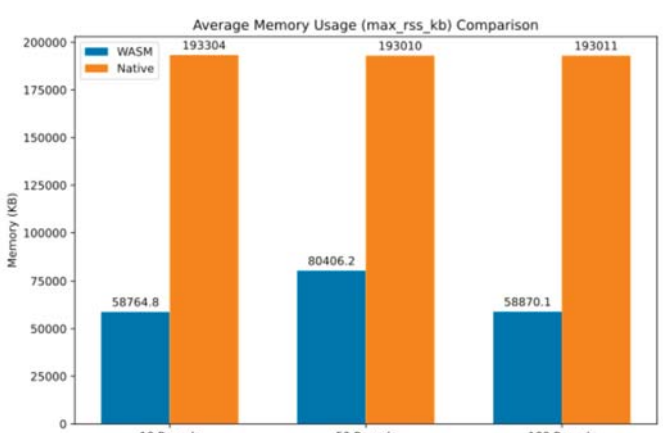


Fig. 12. Average memory usage comparison

The average per-round initialization time for Wasm (2.02 s ±0.04 s at 10 rounds) exceeds that of Native (1.57 s ±0.03 s), but the gap narrows at larger scales (Fig. 11): Wasm 2.03 s (±0.05 s) vs Native 1.58 s (±0.04 s) at 50 rounds; Wasm 2.02 s (±0.05 s) vs Native 1.59 s (±0.04 s) at 100 rounds. This may reflect Wasm's ahead-of-time compilation path, which reduces startup time compared with loading and compiling traditional native code. The average initialization latency across all scenarios is presented in Fig. 12.

RSS memory (Resident Set Size) analysis shows a pronounced advantage for Wasm in memory efficiency. Native consumes 193.0 MB on average (±2.5 MB) and remains stable across scenarios. In contrast, Wasm uses only 58.8 MB on average (±1.2 MB), saving 69.5% of memory relative to Native. As illustrated in Fig. 12, this gap is consistent across all training scenarios (10, 50, and 100 rounds), confirming the stability of Wasm's memory efficiency advantage. Specifically:

- 10 rounds: Wasm 58.8 MB vs Native 193.3 MB (69.6% savings)
- 50 rounds: Wasm 58.8 MB vs Native 193.0 MB (69.5% savings)
- 100 rounds: Wasm 58.9 MB vs Native 193.0 MB (69.5% savings)

This benefit is particularly important in fog-computing settings, where edge devices often have limited memory resources (e.g., Raspberry Pi, Gateway…).

Estimated CPU active time (Fig. 13) shows Wasm exhibiting about 25% higher active time than Native (due to runtime overhead), yet with a lower max total %CPU: Wasm ~22-25% vs Native 82–90% (Fig. 13, 14). This indicates that Wasm is better suited to resource-constrained environments such as Fog and Edge computing.

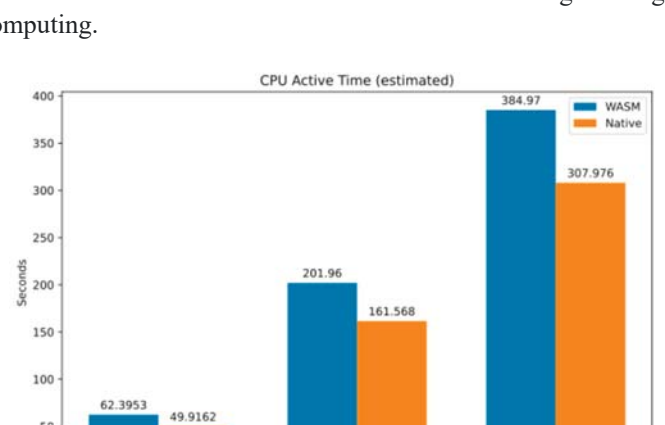


Fig. 13. CPU active time

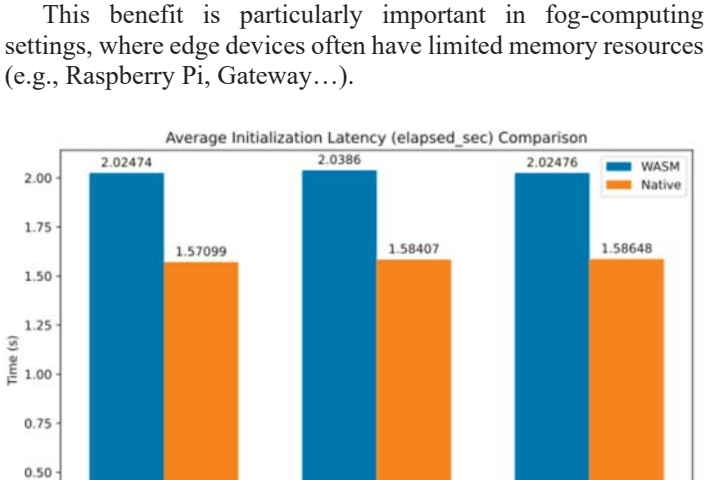


Fig. 11. Average initialization latency comparison

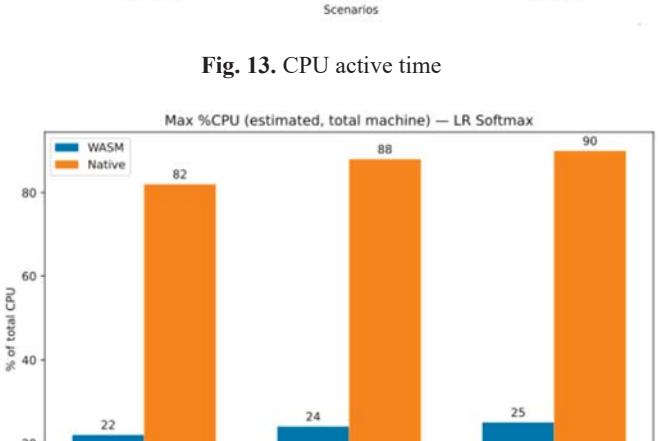


Fig. 14. Max %CPU during training

5. Conclusion

This study presented a Wasm-based framework for deploying federated learning in fog-computing environments, addressing device heterogeneity, resource constraints, and system security. Through theoretical analysis and empirical evaluation on physical devices (Raspberry Pi 4 and Xiaomi 13), we showed that the proposed framework preserves a convergence trajectory equivalent to native execution, with nearly identical accuracy and loss (difference <0.3% after 100 training rounds). At the same time, Wasm offers notable resource-efficiency benefits: a 69.5% reduction in memory usage (58.8 MB versus 193 MB for native) and lower CPU utilization (22-25% versus 82-90%), despite a wall-clock training overhead that decreases from 64.6% at 10 rounds to 10.2% at 100 rounds as initialization costs are amortized, and a stable computational throughput overhead of 135-146% reflecting the intrinsic cost of the Wasm execution layer. These results affirm Wasm as a portable and secure execution layer and clarify the performance–resource trade-offs useful to developers and architects when deploying privacy-preserving distributed AI at the edge/fog in 6G contexts.

Nevertheless, several limitations remain. The evaluation focused on a simple model (multiclass logistic regression on MNIST) and assumed a stable network; it did not measure or analyze energy consumption, an especially important factor for edge devices, which may limit representativeness for complex real-world scenarios. To address these limitations and enhance practicality, several directions are identified for future work.

First, leveraging modern Wasm features represents a natural next step. Wasm support for SIMD (Single Instruction, Multiple Data) enables vectorization of matrix operations and may substantially narrow the performance gap with native code. Additionally, ahead-of-time (AOT) compilation can eliminate JIT-related overhead and improve throughput in production settings.

Second, reducing communication cost remains a priority as the framework scales. Applying gradient compression techniques such as quantization and sparsification, together with communication-efficient FL methods including FedProx and SCAFFOLD, would lower bandwidth requirements when working with larger models. Delta encoding, in which only weight changes rather than full model parameters are transmitted, is another promising direction.

Third, extending support to more complex models will broaden the framework's applicability. Deep architectures such as CNNs (MobileNet, EfficientNet) for computer vision tasks and RNN/LSTM networks for time-series forecasting are natural candidates, requiring optimization of numerical libraries in Rust and improved multi-dimensional tensor handling within Wasm.

Fourth, deploying the framework in real-world domains will validate its practical value. Concrete application areas include healthcare (training diagnostic models on decentralized patient data), smart cities (traffic analytics and energy management), and Industry 4.0 (predictive maintenance), each of which presents distinct requirements in terms of latency, security, and scale.

Fifth, strengthening attack resistance will be essential for production deployment. Integrating Differential Privacy to perturb gradients and mitigate inference attacks, combined with robust aggregation mechanisms such as Krum and coordinate-wise median to detect Byzantine nodes, would complement the existing capability-based security provided by WASI and establish a layered defense architecture.

Overall, this research supports Wasm as a viable foundational technology for federated learning on heterogeneous fog nodes, striking a practical balance among performance, portability, and security. Wasm not only offers a new approach to building distributed AI systems but also advances the realization of fog computing combined with FL in the era of distributed AI. The proposed future directions lay the groundwork for deeper integration with telecommunications and IoT infrastructure and broader 6G platforms.

References

[1] S. Iftikhar, S. S. Gill, C. Song, M. Xu, M. S. Aslanpour, A. N. Toosi, *et al*., "AI-Based Fog and Edge Computing: A Systematic Review, Taxonomy and Future Directions," *Internet of Things*, vol. 21, p. 100674, 2023.

[2] A. A. Ateya, A. A. A. El-Latif, A. Muthanna, A. Volkov, and A. Koucheryavy, *Enabling Metaverse and Telepresence Services in 6G Networks*. New York: River Publishers, 2025, 254 p., doi: 10.1201/9788770046749.

[3] I. Stojmenovic, S. Wen, X. Huang, and H. Luan, "An Overview of Fog Computing and Its Security Issues," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 10, pp. 2991–3005, 2016.

[4] D. V. Thang, A. Volkov, A. Muthanna, A. Koucheryavy, A. A. Ateya, and D. N. K. Jayakody, "Future of telepresence services in the evolving fog computing environment: A survey on research and use cases," *Sensors*, vol. 25, no. 11, p. 3488, 2025, doi: 10.3390/s25113488.

[5] L. Li, Y. Fan, M. Tse, and K. Y. Lin, "A Review of Applications in Federated Learning," *Computers & Industrial Engineering*, vol. 149, p. 106854, 2020.

[6] B. T. Hasan and A. K. Idrees, "Federated Learning for IoT/Edge/Fog Computing Systems," in *Federated Learning*. Palm Bay, FL, USA: Apple Academic Press, 2024, pp. 47–75.

[7] Q. Xia, W. Ye, Z. Tao, J. Wu, and Q. Li, "A Survey of Federated Learning for Edge Computing: Research Problems and Solutions," *High-Confidence Computing*, vol. 1, no. 1, p. 100008, 2021.

[8] L. I. Barona López and T. Borja Saltos, "Heterogeneity Challenges of Federated Learning for Future Wireless Communication Networks," *Journal of Sensor and Actuator Networks*, vol. 14, no. 2, p. 37, 2025.

[9] H. G. Abreha, M. Hayajneh, and M. A. Serhani, "Federated Learning in Edge Computing: A Systematic Survey," *Sensors*, vol. 22, no. 2, p. 450, 2022.

[10] "Unleashing Edge AI with WebAssembly: Performance, Portability, and a Hands-On Guide," *DEV Community*. [Online]. Available: https://dev.to/vaib/unleashing-edge-ai-with-webassembly-performance-portability-and-a-hands-on-guide-p7o (accessed Oct. 28, 2025).

[11] Y. Zhang, M. Liu, H. Wang, Y. Ma, G. Huang, and X. Liu, "Research on WebAssembly Runtimes: A Survey," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 8, pp. 1–47, 2025.

[12] S. E. Khelifa, M. Bagaa, A. O. Messaoud, and A. Ksentini, "Case Study of WebAssembly Runtimes for AI Applications on the Edge," in *Proc. 2024 Global Information Infrastructure and Networking Symposium (GIIS)*, 2024, pp. 1–6.

[13] T. Heinrich, N. C. Will, R. R. Obelheiro, and C. A. Maziero, "Uso de Chamadas WASI para a Identificação de Ameaças em Aplicações WebAssembly," in *Proc. Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)*, 2023, pp. 237–250.

[14] M. Garofalo, M. Colosi, A. Catalfamo, and M. Villari, "Web-Centric Federated Learning over the Cloud-Edge Continuum Leveraging ONNX and WASM," in *Proc. 2024 IEEE Symposium on Computers and Communications (ISCC)*, 2024, pp. 1–7.

[15] J. Bachmeier, V. Yussupov, J. Henß, and H. Koziolek, "WebAssembly with WASI-NN for Edge Machine Learning Inference: Experiences and Lessons Learned," in *European Conference on Software Architecture*, Cham, Switzerland: Springer Nature Switzerland, 2025, pp. 343–359.

[16] D. Van Thang, A. Volkov, A. Muthanna, I. A. Elgendy, R. Alkanhel, D. N. K. Jayakody, and A. Koucheryavy, "A framework

integrating federated learning and fog computing based on client sampling and dynamic thresholding techniques," *IEEE Access*, 2025, doi: 10.1109/ACCESS.2025.

[17] "Introduction – Wasmtime," *Wasmtime Documentation*. [Online]. Available: https://docs.wasmtime.dev/ (accessed Oct. 28, 2025).

[18] J. Bosamiya, W. S. Lim, and B. Parno, "Provably-safe multilingual software sandboxing using WebAssembly," in *Proc. 31st USENIX Security Symp. (USENIX Security '22)*, Boston, MA, Aug. 2022, pp. 1975–1992.

[19] P. Gackstatter, P. A. Frangoudis, and S. Dustdar, "Pushing serverless to the edge with WebAssembly runtimes," in *Proc. 22nd IEEE/ACM Int. Symp. Cluster, Cloud Internet Comput. (CCGrid)*, 2022, pp. 140–149, doi: 10.1109/CCGrid54584.2022.00023.

[20] S. Berard, "Part One: Exploring WebAssembly to Power Secure, Portable Applications Spanning the Cloud to Tiny Edge Devices," *Atym*. [Online]. Available: https://www.atym.io/post/exploring-webassembly-to-power-secure-portable-applications-spanning-the-cloud-to-tiny-edge-devices (accessed Oct. 28, 2025).

[21] Hoffman, Kevin. *"Programming WebAssembly with Rust: unified development for web, mobile, and embedded applications,"* 2019: 1-220.

[22] L. Wagner, M. Mayer, A. Marino, A. Soldani Nezhad, H. Zwaan, and I. Malavolta, "On the Energy Consumption and Performance of WebAssembly Binaries across Programming Languages and Runtimes in IoT," in *Proc. 27th International Conference on Evaluation and Assessment in Software Engineering*, 2023, pp. 72–82.

[23] A. Prabakar and R. Kiran, "WebAssembly Performance Analysis: A Comparative Study of C++ and Rust Implementations," 2024.

[24] A. N. Volkov, A. S. A. Muthanna, A. E. Koucheryavy, A. S. Borodin, A. I. Paramonov, S. S. Vladimirov, et al., "Perspective research on networks and services 2030 in the 6G Meganetlab laboratory of SPbSUT," *Elektrosvyaz*, no. 6, pp. 5-14, 2023, doi: 10.34832/ELSV.2023.43.6.001.

ФРЕЙМВОРК РАЗВЁРТЫВАНИЯ ФЕДЕРАТИВНОГО ОБУЧЕНИЯ НА ОСНОВЕ WEBASSEMBLY ДЛЯ СРЕД ТУМАННЫХ ВЫЧИСЛЕНИЙ

Данг Ван Тханг, *Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича (СПбГУТ), Санкт-Петербург, Россия,* ***dangvanthang_sut@gmail.com***

Кучерявый Андрей Евгеньевич, *Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича (СПбГУТ), Санкт-Петербург, Россия,* ***akouch@mail.ru***

Аннотация

Федеративное обучение в среде туманных вычислений для сетей 6G открывает широкие возможности в различных прикладных областях. Локальность данных используется для защиты конфиденциальной информации, снижения коммуникационных затрат и разгрузки центрального сервера. Вместе с тем развёртывание задач ФО на туманных узлах с гетерогенными аппаратными конфигурациями, различными операционными системами и архитектурами набора инструкций остаётся серьёзной технической проблемой. Жёсткие требования 6G к задержке и тесная интеграция с приложениями реального времени дополнительно усугубляют её. WebAssembly (Wasm) представляет собой переносимый байткод-формат, способный преодолеть указанные ограничения в среде туманных вычислений 6G. Безопасность исполнения обеспечивается изолированной средой и разграничением полномочий на основе возможностей; производительность близка к нативной, объём модулей компактен, а запуск возможен на любой платформе без перекомпиляции. В настоящей работе представлен фреймворк развёртывания федеративного обучения на основе Wasm в среде туманных вычислений для сетей 6G. Практическая применимость и производительность решения оценены в экспериментах на гетерогенном наборе устройств. Полученные результаты свидетельствуют о том, что предложенный фреймворк снижает потребление оперативной памяти и загрузку процессора, сохраняя при этом высокую точность модели, что соответствует требованиям к производительности сетей 6G.

Ключевые слова: *WebAssembly, федеративное обучение, туманные вычисления, сети 6G, гетерогенные устройства.*

Литература

1. *Iftikhar S., Gill S.S., Song C., Xu M., Aslanpour M.S., Toosi A.N.* et al. AI-Based Fog and Edge Computing: A Systematic Review, Taxonomy and Future Directions // Internet of Things. 2023. Vol. 21. Art. 100674. DOI: 10.1016/j.iot.2022.100674.

2. *Ateya A.A., El-Latif A.A.A., Muthanna A., Volkov A., Koucheryavy A.* Enabling Metaverse and Telepresence Services in 6G Networks. New York : River Publishers, 2025. 254 p. DOI: 10.1201/9788770046749.

3. *Stojmenovic I., Wen S., Huang X., Luan H.* An Overview of Fog Computing and Its Security Issues // Concurrency and Computation: Practice and Experience. 2016. Vol. 28, № 10, pp. 2991-3005. DOI: 10.1002/cpe.3485.

4. *Thang D.V., Volkov A., Muthanna A., Koucheryavy A., Ateya A.A., Jayakody D.N.K.* Future of Telepresence Services in the Evolving Fog Computing Environment: A Survey on Research and Use Cases // Sensors. 2025. Vol. 25, № 11. Art. 3488. DOI: 10.3390/s25113488.
5. *Li L., Fan Y., Tse M., Lin K.Y.* A Review of Applications in Federated Learning // Computers & Industrial Engineering. 2020. Vol. 149. Art. 106854. DOI: 10.1016/j.cie.2020.106854.
6. *Hasan B.T., Idrees A.K.* Federated Learning for IoT/Edge/Fog Computing Systems // Federated Learning. Apple Academic Press, 2024, pp. 47-75.
7. *Xia Q., Ye W., Tao Z., Wu J., Li Q.* A Survey of Federated Learning for Edge Computing: Research Problems and Solutions // High-Confidence Computing. 2021. Vol. 1, № 1. Art. 100008. DOI: 10.1016/j.hcc.2021.100008.
8. *Barona Lopez L.I., Borja Saltos T.* Heterogeneity Challenges of Federated Learning for Future Wireless Communication Networks // Journal of Sensor and Actuator Networks. 2025. Vol. 14, № 2. Art. 37. DOI: 10.3390/jsan14020037.
9. *Abreha H.G., Hayajneh M., Serhani M.A.* Federated Learning in Edge Computing: A Systematic Survey // Sensors. 2022. Vol. 22, № 2. Art. 450. DOI: 10.3390/s22020450.
10. Unleashing Edge AI with WebAssembly: Performance, Portability, and a Hands-On Guide [Электронный ресурс] // DEV Community. URL: https://dev.to/vaib/unleashing-edge-ai-with-webassembly-performance-portability-and-a-hands-on-guide-p7o (дата обращения: 28.10.2025).
11. *Zhang Y., Liu M., Wang H., Ma Y., Huang G., Liu X.* Research on WebAssembly Runtimes: A Survey // ACM Transactions on Software Engineering and Methodology. 2025. Vol. 34, № 8, pp. 1-47. DOI: 10.1145/3714465.
12. *Khelifa S.E., Bagaa M., Messaoud A.O., Ksentini A.* Case Study of WebAssembly Runtimes for AI Applications on the Edge // Proc. 2024 Global Information Infrastructure and Networking Symposium (GIIS). 2024. P. 1-6. DOI: 10.1109/GIIS59465.2024.10449907.
13. *Heinrich T., Will N.C., Obelheiro R.R., Maziero C.A.* Uso de Chamadas WASI para a Identifica??o de Amea?as em Aplicacoes WebAssembly // Proc. Simp?sio Brasileiro de Seguran?a da Informa??o e de Sistemas Computacionais (SBSeg). 2023, pp. 237-250. DOI: 10.5753/sbseg.2023.233111.
14. *Garofalo M., Colosi M., Catalfamo A., Villari M.* Web-Centric Federated Learning over the Cloud-Edge Continuum Leveraging ONNX and WASM // Proc. 2024 IEEE Symposium on Computers and Communications (ISCC). 2024. P. 1-7. DOI: 10.1109/ISCC61673.2024.10733614.
15. *Bachmeier J., Yussupov V., Henss J., Koziolek H.* WebAssembly with WASI-NN for Edge Machine Learning Inference: Experiences and Lessons Learned // Software Architecture. Cham : Springer Nature Switzerland, 2025, pp. 343-359. DOI: 10.1007/978-3-032-02138-0_23.
16. *Van Thang D., Volkov A., Muthanna A., Elgendy I.A., Alkanhel R., Jayakody D.N.K., Koucheryavy A.* A Framework Integrating Federated Learning and Fog Computing Based on Client Sampling and Dynamic Thresholding Techniques // IEEE Access. 2025. DOI: 10.1109/ACCESS.2025.
17. Introduction - Wasmtime [Электронный ресурс] // Wasmtime Documentation. URL: https://docs.wasmtime.dev/ (дата обращения: 28.10.2025).
18. *Bosamiya J., Lim W.S., Parno B.* Provably-Safe Multilingual Software Sandboxing using WebAssembly // Proc. 31st USENIX Security Symposium (USENIX Security '22). Boston : USENIX Association, 2022, pp. 1975-1992. URL: https://www.usenix.org/conference/usenixsecurity22/presentation/bosamiya.
19. *Gackstatter P., Frangoudis P.A., Dustdar S.* Pushing Serverless to the Edge with WebAssembly Runtimes // Proc. 22nd IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGrid). IEEE, 2022., pp. 140-149. DOI: 10.1109/CCGrid54584.2022.00023.
20. *Berard S.* Part One: Exploring WebAssembly to Power Secure, Portable Applications Spanning the Cloud to Tiny Edge Devices [Электронный ресурс] // Atym. URL: https://www.atym.io/post/exploring-webassembly-to-power-secure-portable-applications-spanning-the-cloud-to-tiny-edge-devices (дата обращения: 28.10.2025).
21. *Hoffman K.* Programming WebAssembly with Rust: Unified Development for Web, Mobile, and Embedded Applications. Raleigh : Pragmatic Bookshelf, 2019. 240 p. ISBN 978-1-68050-636-5.
22. *Wagner L., Mayer M., Marino A., Soldani Nezhad A., Zwaan H., Malavolta I.* On the Energy Consumption and Performance of WebAssembly Binaries across Programming Languages and Runtimes in IoT // Proc. 27th International Conference on Evaluation and Assessment in Software Engineering (EASE). 2023, p. 72-82. DOI: 10.1145/3593434.3593454.
23. *Prabakar A., Kiran R.* WebAssembly Performance Analysis: A Comparative Study of C++ and Rust Implementations : B.S. thesis. Karlskrona : Blekinge Institute of Technology, 2024. URL: https://urn.kb.se/resolve?urn=urn:nbn:se:bth-26632.
24. *Волков А.Н., Мутханна А.С.А., Кучерявый А.Е., Бородин А.С., Парамонов А.И., Владимиров С.С.* и др. Перспективные исследования сетей и услуг 2030 в лаборатории 6G Meganetlab СПбГУТ // Электросвязь. 2023. № 6. С. 5-14. DOI: 10.34832/ELSV.2023.43.6.001.

Информация об авторах:

Данг Ван Тханг, *аспирант кафедры передачи сигналов, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича (СПбГУТ), Санкт-Петербург, Россия, ORCID: https://orcid.org/0009-0009-2219-3767*

Кучерявый Андрей Евгеньевич, *профессор, заведующий кафедрой сетей связи и передачи данных, д.т.н., Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича (СПбГУТ), Санкт-Петербург, Россия, ORCID: https://orcid.org/0000-0003-4479-2479*

ITERATIVE DEMODULATION ALGORITHM BASED ON GRADIENT SEARCH METHOD FOR MU-MIMO SYSTEMS WITH LARGE ANTENNA ARRAYS

DOI: 10.36724/2072-8735-2026-20-6-63-72

Manuscript received 24 March 2026;
Accepted 03 June 2026

Ben Rezheb S.B.K.,
Moscow Technical University of Communications and Informatics, Moscow, Russia, ***sbenrezheb@yandex.ru***

Vitaly B. Kreyndelin,
Moscow Technical University of Communications and Informatics, Moscow, Russia, ***vitkrend@gmail.com***

Keywords: *MU-MIMO, Massive MIMO, MMSE, OGM2, Spatial correlation, interference immunity, computational complexity, Turbocode, QAM*

This paper proposes and examines an iterative demodulation algorithm based on the optimized gradient method (OGM2) for high-dimensional multi-user MIMO (MU-MIMO) systems, taking into account the spatial correlation of fading. The classic MMSE algorithm, which is effective for small antenna configurations, becomes inapplicable in Massive MU-MIMO systems due to its high computational complexity. The proposed iterative approach provides a significant reduction in computational costs while maintaining acceptable noise immunity in conditions close to those of real communication channels. The simulation results are given for the MU-MIMO system with antenna configurations 128x128, 256x256, and 512x512 for 4 and 8 users at the level of spatial correlation of fading equal to 0.2. It is shown that the proposed algorithm provides a reduction in computational complexity by 2-8 times compared to the traditional MMSE algorithm with losses in noise immunity not more than 0.1-0.2 dB in systems with turbo coding at the level *FER* = 10^{-2}. The results demonstrate the robustness of the algorithm to spatial correlation, which confirms its practical applicability in real MIMO systems.

Information about authors:
Ben Rezheb Sofien Ben Kamel, *postgraduate student, Moscow Technical University of Communications and Informatics, Moscow, Russia*
Kreindelin Vitaly Borisovich, *Head of the Department of Theory of Electrical Circuits, Professor, Doctor of Technical Sciences, Moscow Technical University of Communications and Informatics, Moscow, Russia*

Для цитирования:
Бен Режеб С.Б.К., Крейнделин В.Б. Итерационный алгоритм демодуляции на основе метода градиентного поиска для систем MU-MIMO с большим числом антенн // T-Comm: Телекоммуникации и транспорт. 2026. Том 20. №6. С. 63-72.

For citation:
Ben Rezheb S.B.K., V.B. Kreyndelin, "Gradient Search Iterative Demodulation Algorithm for Multi-Antenna MU-MIMO Systems," *T-Comm*, 2026, vol. 20, no. 6, pp. 63-72.

Introduction

Modern and future wireless communication systems impose increasingly stringent requirements on data rates and quality of service for a large number of simultaneously active users. Multi-user transmission with many transmit and many receive antennas (MU-MIMO, Multi-User Multiple-Input Multiple-Output) is one of the key solutions for meeting these goals, enabling a base station to serve multiple users simultaneously on the same time-frequency resources [1, 2].

The transition to Massive MU-MIMO systems, where the base station is equipped with hundreds of antennas, opens up new opportunities for increasing throughput [3, 4]. However, the practical implementation of such systems faces serious computational challenges, especially at the stage of demodulating the received signals. The classical linear MMSE (Minimum Mean Square Error) algorithm, widely used in small-scale MIMO systems, requires inversion of large-dimension matrices, which leads to computational complexity on the order of $O\left(L^3\right)$, where L – Is the number of antennas [5, 6].

An important factor that distinguishes real communication channels from idealized models is the presence of spatial correlation between antenna elements. Ignoring spatial correlation when developing and analyzing demodulation algorithms can lead to an inaccurate assessment of their actual effectiveness.

The aim of this work is to study the applicability of the proposed OGM2-based iterative demodulation algorithm for high-dimensional MU-MIMO systems taking into account spatial correlation of the fading. This will allow evaluating the algorithm's performance under conditions as close as possible to real communication channels and confirming its practical relevance for modern wireless communication systems.

1 MU-MIMO system model

Figure 1 shows the block diagram of an MU-MIMO system in the uplink[1],. Let us denote by L the total number of transmit antennas (the sum of the antennas of all users), and by K – the number of receive antennas at the base station.

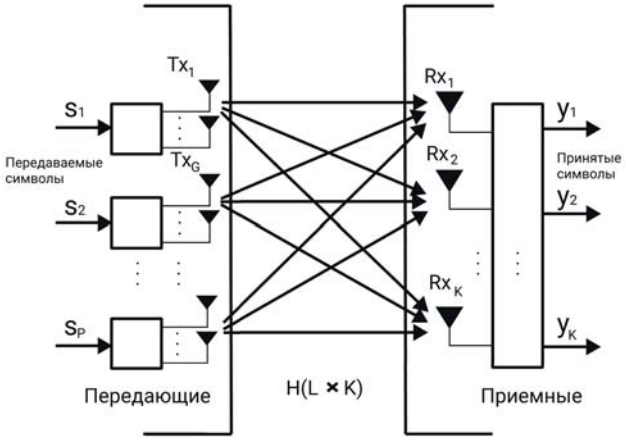


Fig. 1. Block diagram of the MU-MIMO system.

In the considered configuration, P users simultaneously transmit data to a base station equipped with K receive antennas; each user is equipped with G transmit antennas. For example, the $\mathrm{Tx}_1,...,\mathrm{Tx}_G$ antennas transmit the complex information symbols $\mathrm{s}_i(1)$ of the first user to the $\mathrm{Rx}_1,...,\mathrm{Rx}_K$ antennas. The signals received at the $\mathrm{Rx}_1,...,\mathrm{Rx}_K$ antennas are denoted by $\mathrm{y}_1,...,\mathrm{y}_K$.. The transmit antennas of all P users simultaneously send complex information symbols to the base station. Each receive antenna of the base station receives signals from all $L = P \cdot G$ transmit antennas of all users [7].

The MU-MIMO channel model in vector-matrix form is given by:

$$\mathbf{y} = \sum_{p=1}^{P} \mathbf{H}_p \mathbf{s}_p + \mathbf{n} \qquad (1)$$

In expression (1) the following notations are used:

- $\mathrm{s}_i(p)$, $i = 1,2,\ldots G$, – is the complex symbol transmitted by the i-th antenna of the p-th user;
- $\mathbf{s}_p = [\mathrm{s}_1(p),\ldots \mathrm{s}_i(p),...,\mathrm{s}_G(p)]^{\mathrm{T}}$, $p = 1,2,\ldots P$- is the complex column vector of transmitted information symbols of the p-th user with dimensions $\mathrm{G}\times 1$;
- $\mathbf{y} = [\mathrm{y}_1,...,\mathrm{y}_K]^{\mathrm{T}}$ – s the complex column vector of the signal received at the base station with dimensions $\mathrm{K}\times 1$.
- $\mathbf{n} = [\mathrm{n}_1,...,\mathrm{n}_K]^{\mathrm{T}}$ – is the complex column vector of additive white Gaussian noise (AWGN) with dimensions $\mathrm{K}\times 1$, whose elements are uncorrelated and each has zero mean and variance $2\sigma_n^2$;
- $\mathbf{H}_p$ - is the complex channel matrix of the p-th user with dimensions $\mathrm{K}\times\mathrm{G}$.

Expression (1) can be written in vector-matrix form by combining the channel matrices of all users into a single block matrix and the vectors of transmitted symbols into a single column vector:

$$\mathbf{y} = [\mathbf{H}_1 \quad \mathbf{H}_2 \quad \ldots \quad \mathbf{H}_P] \cdot \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \vdots \\ \mathbf{s}_P \end{bmatrix} + \mathbf{n}. \qquad (2)$$

By denoting the combined channel matrix as $\mathbf{H} = [\mathbf{H}_1 \quad \mathbf{H}_2 \quad \ldots \quad \mathbf{H}_P]$ of dimensions $\mathrm{K}\times\mathrm{L}$ and the combined vector of transmitted symbols as $\mathbf{s} = [\mathbf{s}_1^T \quad \mathbf{s}_2^T \quad \ldots \quad \mathbf{s}_P^T]^{\mathrm{T}}$ of dimensions $\mathrm{L}\times 1$, the following model of a multi-user MIMO system is obtained, which is formally similar to the model of a single-user MIMO system:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}. \qquad (1)$$

Thus, the demodulation problem in a MU-MIMO system can be reduced to the demodulation problem in an equivalent single-user MIMO system with a channel matrix of dimensions $\mathrm{K}\times\mathrm{L}$. This makes it possible to apply demodulation algorithms originally developed for single-user MIMO systems.

[1] On this line, the base station receives signals from many users.

2 Algorithm MMSE

A large number of different demodulation methods are known for model (1). Among them, a class of linear algorithms can be distinguished, which have significantly lower computational complexity compared to nonlinear optimal detection methods [3, 5].

The MMSE demodulation algorithm is a linear algorithm that is optimal according to the minimum mean square error criterion between the transmitted information symbols and their estimates at the receiver side. The MMSE algorithm performs linear processing of the received vector $\mathbf{y}$ in order to obtain an estimate $\hat{\mathbf{s}}^{MMSE}$ of the transmitted information symbol vector $\mathbf{s}$. The obtained estimate $\hat{\mathbf{s}}^{MMSE}$ generally does not belong to the set of possible symbols $\mathbf{\Theta}$ which is determined by the modulation scheme. Therefore, an additional nonlinear transformation of the estimate is required in order to map the obtained values to the nearest valid constellation symbols [8, 9].

The MMSE algorithm calculates the estimate $\hat{\mathbf{s}}^{MMSE}$ as follows [5]:

$$\hat{\mathbf{s}}^{\text{MMSE}} = \left(\mathbf{H}'\mathbf{H} + 2\sigma_n^2\mathbf{1}\right)^{-1}\mathbf{H}'\mathbf{y}\,, \qquad (4)$$

where $\mathbf{1}$ – is the identity matrix of dimensions L×L .

It should be noted that the calculation of the estimate $\hat{\mathbf{s}}^{MMSE}$ according to expression (4) requires a significant number of arithmetic operations, with computational complexity on the order of ($O\left(L^3\right)$ for $L = K$). For a large number of transmit and receive antennas, which is typical for high-dimensional MIMO (Massive MIMO) systems, the computational complexity of the MMSE algorithm becomes excessively high. This significantly complicates or even prevents its practical implementation in such systems.

3 Iterative demodulation algorithm

Let us now consider approximate linear iterative demodulation methods. In this case, it is necessary to optimize a scalar function $f(\mathbf{s})$, hat depends on the vector of complex information symbols $\mathbf{s}$. As such a function $f(\mathbf{s})$ we choose the following quadratic form. Let us turn to approximate linear iterative demodulation methods:

$$f(\mathbf{s}) = \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 + 2\sigma_n^2 \cdot \|\mathbf{s}\|^2 \quad , \qquad (5)$$

where $\|...\|$ – denotes the Euclidean norm of a vector.

Let us now find the minimum of function (5). Due to its quadratic form, this function has a unique minimum, which can be found by setting its gradient equal to zero. Then, taking expression (5) into account, we obtain:

$$grad\left(f(\mathbf{s})\right) = \frac{df(\mathbf{s})}{d\mathbf{s}} = 2\cdot\left(\mathbf{H}'\mathbf{H} + 2\sigma_n^2\cdot\mathbf{1}\right)\cdot\mathbf{s} - 2\cdot\mathbf{y} = \qquad (6)$$
$$= 2\cdot\left(\mathbf{H}'\cdot\left(\mathbf{H}\cdot\mathbf{s}\right) + 2\sigma_n^2\cdot\mathbf{s}\right) - 2\cdot\mathbf{y} = 0$$

By solving equation (6) with respect to the unknown vector $\mathbf{s}$, the following estimate of $\hat{\mathbf{s}}$ is obtained:

$$\hat{\mathbf{s}} = \left(\mathbf{H}'\mathbf{H} + 2\sigma_n^2\mathbf{1}\right)^{-1}\mathbf{H}'\mathbf{y}\,. \qquad (7)$$

Comparing expressions (4) and (7), it can be seen that the exact solution of the minimization problem for the quadratic function $f(\mathbf{s})$ leads to the MMSE estimate given in (4). As noted earlier, due to the high computational complexity of calculating the exact MMSE estimate in systems with a large number of transmit and receive antennas, direct implementation of expression (4) becomes difficult or even impractical [5, 9].

Therefore, it is reasonable to avoid the exact solution in (4) and instead use approximate iterative methods to minimize the function $f(\mathbf{s})$. A large number of iterative optimization methods are known for solving such problems [10-12]. Among the most widely used methods are:

- the Heavy Ball Method, which introduces a momentum term and uses gradient information from previous iterations in order to accelerate convergence [13];
- the Momentum Method, which forms an exponentially weighted moving average of past gradients [14];
- Nesterov's Accelerated Gradient method, which provides an optimal convergence rate for smooth convex optimization problems 11];
- the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA), which demonstrates high convergence speed in large-scale optimization problems [15];
- the Conjugate Gradient Method, which also provides high convergence speed in cases where the eigenvalues of the system matrix have a small spread [16];
- quasi-Newton methods, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm, which approximate the Hessian matrix of second derivatives [17, 18].

One of the most efficient iterative optimization methods is the OGM2 method (Optimized Gradient Method 2) [11].

The OGM2 algorithm is an improved version of accelerated gradient descent designed for minimizing convex functions. Compared with the conventional gradient descent method, OGM2 provides a higher convergence rate, allowing the solution to be obtained in a smaller number of iterations.

The algorithm belongs to the class of first-order optimization methods, since it uses only gradient information and does not require computation of second derivatives or inversion of large matrices. As a result, OGM2 has relatively low computational complexity and is well suited for high-dimensional problems.

In massive MU-MIMO signal demodulation problems, the use of OGM2 is particularly effective because the computational complexity of conventional methods such as MMSE increases significantly with the number of antennas. In contrast, OGM2 is implemented as a sequence of iterative matrix-vector multiplication operations, which reduces computational complexity and improves the efficiency of hardware implementation.

The operation of the proposed demodulation algorithm based on the OGM2 method includes the following steps:

Initial conditions: $\mathbf{s}_0 = \mathbf{0}$ – is the initial value of the vector $\mathbf{s}$; L_0 – is the algorithm parameter corresponding to the Lipschitz constant of the function $f(\mathbf{s})$) [19]; I – is the total number of iterations of the algorithm; $i = 0$ – is the initial value of the current iteration index i; $\theta_0 = 1$ – is the initial value of the acceleration parameter.

1. alculation of the gradient of the function $f(\mathbf{s})$ at the current point $\mathbf{s}_i$:

$$grad\left(f\left(\mathbf{s}_i\right)\right) = 2\cdot\left(\mathbf{H}'\cdot\mathbf{H}+2\sigma_n^2\cdot\mathbf{1}\right)\cdot\mathbf{s}_{i-1} - 2\cdot\mathbf{H}'\cdot\mathbf{y}. \quad (8)$$

2. alculation of the Lipschitz constant:

$$L_0 = 8\cdot\left(\|\mathbf{H}\|_F^2 + 5\cdot 2\sigma_n^2\right), \quad (9)$$

where $\|\mathbf{H}\|_F$ – is the Frobenius norm of the matrix $\mathbf{H}$ [20].

3. Calc lation of the temporary (intermediate) point:

$$\mathbf{y}_{i+1} = \mathbf{s}_i - \frac{1}{L_0} grad\left(f\left(\mathbf{s}_i\right)\right). \quad (10)$$

4. Update of the acceleration pa meter θ_{i+1}:

$$\theta_{i+1} = \frac{1+\sqrt{1+6\theta_i^2}}{2}, i = 0;\ldots I-1 \quad (11)$$

5. orrection using a weighted sum of gradients (i.e., with accumulation of gradients from all previous iterations):

$$\mathbf{z}_{i+1} = \mathbf{s}_i - \alpha\cdot\frac{1}{L_0}\sum_{k=0}^{i}\theta_k grad\left(f\left(\mathbf{s}_k\right)\right) \quad (12)$$

where α – is a tunable algorithm parameter whose value is determined experimentally during simulation depending on the antenna configuration dimensions of the MIMO system.

6. Calculation of the estimate at the next point. This estimate is determined as a weighted sum of $\mathbf{y}_{i+1}$ and $\mathbf{z}_{i+1}$:

$$\mathbf{s}_{i+1} = \left(1-\frac{1}{\theta_{i+1}}\right)\mathbf{y}_{i+1} + \frac{1}{\theta_{i+1}}\mathbf{z}_{i+1} \quad . \quad (13)$$

7. $i = i+1$.

Steps 1–7 are repeated for all values of $i = 1\ldots I$, As a result, an estimate $\hat{\mathbf{s}}$ of the complex information symbol vector $\mathbf{s}$ is obtained at the output:

$$\hat{\mathbf{s}} = \mathbf{s}_I. \quad (14)$$

8. Next, the decoding parameter D is calculated:

$$D = 0.22\cdot\left(1+4.3\cdot 2\sigma_n^2\right)\cdot 2\sigma_n^2. \quad (15)$$

9. Decoding, i.e., obtaining the vector of estimated transmitted bits $\hat{\mathbf{b}}$, is then performed using a soft-input turbo decoder.

$$\hat{\mathbf{b}} = \boldsymbol{\varphi}\left(\hat{\mathbf{s}}, D\right), \quad (16)$$

where $\boldsymbol{\varphi}(\ldots)$ - is the function describing the turbo decoding algorithm.

The proposed demodulation algorithm differs from the original OGM2 algorithm in the specific computation of the Lipschitz constant L_0 according to expression (9), the computation of the decoding parameter D according to expression (15), and the introduction of the tunable parameter α. During simulation, the optimal values of parameter α, providing the best convergence performance of the algorithm, were determined. For the 128×128 antenna configuration, the parameter value is $\alpha = 38$, while for the 256×256 and 512×512 antenna configurations, the optimal value is $\alpha = 40$.

The demodulation algorithm described by expressions (8)…(16), achieves accelerated convergence due to the additional correction introduced by expression (12) for $\mathbf{z}_{i+1}$, which takes into account the gradients accumulated during previous iterations. This modification considers the contribution of all previous gradients, thereby improving the convergence rate of the algorithm.

4 Computational complexity analysis

Let us compare the computational complexity of the proposed demodulation algorithm (expressions (8)…(16), hereinafter referred to as OGM2) and the conventional MMSE algorithm (4). Computational complexity is understood as the number of elementary arithmetic operations (multiplications and additions) required to execute the corresponding algorithm. In digital signal processing systems, computational complexity directly determines the hardware resource requirements, power consumption, and data processing latency. The main computational operations in demodulation algorithms for MU-MIMO systems are:

- matrix-vector multiplication with computational complexity $O(KL)$;
- multiplication of two matrices with computational complexity $O\left(K^2L\right)$;
- inversion of an $L\times L$ matrix with computational complexity $O\left(L^3\right)$.

The conventional MMSE algorithm (4) requires inversion of an $L\times L$ matrix, which results in a cubic dependence of the number of operations on the system dimension. In contrast, the proposed iterative OGM2-based algorithm performs only matrix-vector operations at each iteration, completely avoiding matrix inversion. Even when several tens of iterations are performed, the total computational complexity remains significantly lower.

Table 1 presents the results of computational complexity analysis for the OGM2 algorithm (8)…(16) and the MMSE algorithm (4) for antenna configurations of 128×128, 256×256 and 512×512. It is assumed that the number of transmit and receive antennas is equal, i.e., $K = L$.

Table 1

Computational complexity of the OGM2 and MMSE algorithms

Number of iterations	32	-
	Algorithm OGM2	MMSE
Number of multiplications	$I(8L^2+16L+10)+4L^2+2L$	$4L^3+6L^2-L$
Number of multiplications for $L=128$ antennas	4,323,952	8,486,784
Number of multiplications for $L=256$ antennas	17,171,264	67,501,824
Number of multiplications for $L=512$ antennas	68,420,928	538,443,264
Number of additions	$I(8L^2+8L+5)+2L(2L-$	$4L^3+3L^2-3L$

Number of additions for 128 antennas	4,291,712	8,437,376
Number of additions for 256 antennas	17,104,544	67,304,704
Number of additions for 512 antennas	68,287,648	537,655,808
Total number of operations for 128 antennas	8,615,664	16,924,289
Total number of operations for 256 antennas	34,275,808	134,806,528
Total number of operations for 512 antennas	136,708,576	1,076,099,072

On Fig. 2 a graphical representation of the computational complexity is shown.

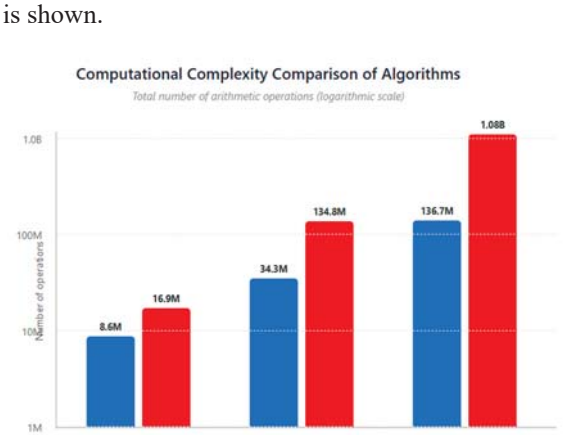


Fig. 2. Diagram of computational complexity for the OGM2 and MMSE algorithms for 128, 256, and 512 antennas.

Based on the results presented in Table 1 and Fig. 2, it can be concluded that the proposed iterative OGM2-based demodulation algorithm, described by expressions (8)…(16) exhibits high computational efficiency.

For a MU-MIMO system with a 128×128 antenna configuration, the proposed OGM2 algorithm requires 8,615,664 arithmetic operations when 32 iterations are performed. For the same configuration, the MMSE algorithm requires 16,924,289 operations. Thus, the use of OGM2 reduces computational complexity by approximately a factor of 2. This indicates that even for medium-scale systems, the iterative approach provides a noticeable reduction in computational cost while maintaining the required demodulation accuracy.

For a system with a 256×256 configuration, the difference between the two algorithms becomes even more pronounced. The OGM2 algorithm requires 34,275,808 arithmetic operations, whereas the MMSE implementation requires 134,806,528 operations. In this case, the reduction in computational complexity reaches approximately 3.9 times. This result shows that the growth in complexity of the OGM2 algorithm is significantly slower compared to MMSE, for which computational costs increase sharply due to the need for matrix inversion operations of large dimension.

The most significant advantage is observed for a MU-MIMO system with a 512×512. configuration. In this case, the OGM2 algorithm requires 136,708,576 arithmetic operations, while the MMSE algorithm requires 1,076,099,072 operations. Consequently, the reduction in computational complexity is approximately 7.9 times. This confirms the high scalability of the proposed iterative method and its effectiveness for use in large-scale massive MU-MIMO systems.

It should be noted that the advantage in computational complexity becomes more pronounced as the MU-MIMO system dimensionality increases. This is due to the fact that the MMSE algorithm requires matrix inversion, which leads to significant computational costs of order $O\left(L^3\right)$, whereas OGM2 has a substantially lower complexity of $O\left(L^2 I\right)$. Consequently, the proposed iterative demodulation algorithm significantly reduces the computational burden compared to the conventional MMSE approach, making it practical for large-scale MU-MIMO systems.

5 Turbo coding

Turbo coding is applied in the considered MIMO system to improve the noise immunity of data transmission and to reduce the probability of errors in signal reception. The use of turbo codes enables high transmission reliability even at low signal-to-noise ratio (SNR) values, which is especially important for multi-user MU-MIMO systems with a large number of transmit and receive antennas. Thanks to iterative decoding, turbo codes achieve performance close to the theoretical Shannon limit and are therefore widely used in modern digital communication systems.

Structurally, the turbo encoder consists of a parallel connection of two systematic convolutional encoders, between which an interleaver is used [21, 22]. The input to the turbo encoder is a vector of information bits, which is simultaneously fed:

- into the first systematic convolutional encoder;
- into the second convolutional encoder through an interleaver.

The first encoder processes the original bit sequence without changing its order and generates systematic and parity bits. In the second branch, the sequence is first permuted by the interleaver, which changes the ordering of bits according to a predefined rule. After that, the permuted sequence is passed to the second convolutional encoder. The use of the interleaver helps to remove correlation between errors and improves the efficiency of iterative decoding, since both encoders operate on different representations of the same message.

At the output, a stream of encoded bits is formed. To control redundancy, puncturing is applied in order to achieve the required code rate [23]. Decoding is performed using an iterative method with two MAP decoders exchanging a posteriori information [24].

6 Spatial correlation of fading

Let $\mathbf{H}_w$ – denote the uncorrelated Rayleigh fading MIMO channel matrix of dimensions $L \times K$, consisting of complex, independent Gaussian random variables with zero mean and identical variances. Let $\mathbf{H}$ – denote the correlated MIMO channel matrix, also of dimensions $L \times K$.

The following Kronecker model of a correlated MIMO channel is well known [25]:

$$\mathbf{H} = \mathbf{R}_r^{1/2}\mathbf{H}_w\mathbf{R}_t^{1/2}, \qquad (17)$$

where $\mathbf{R}_t$ – is the transmit correlation matrix of dimensions L×L; $\mathbf{R}_r$ – is the receive correlation matrix of dimensions K×K. The matrices $\mathbf{R}_t$ and $\mathbf{R}_r$ are positive definite Hermitian matrices with ones on the main diagonal.

For the sake of simplification, it is assumed that the correlation matrices $\mathbf{R}_t$ and $\mathbf{R}_r$ are tridiagonal matrices [3]:

$$\mathbf{R}_t = \begin{bmatrix} 1 & p_t & 0 & \cdots & 0 \\ p_t & 1 & p_t & \cdots & \vdots \\ 0 & p_t & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & p_t \\ 0 & \cdots & 0 & p_t & 1 \end{bmatrix}; \quad \mathbf{R}_r = \begin{bmatrix} 1 & p_r & 0 & \cdots & 0 \\ p_r & 1 & p_r & \cdots & \vdots \\ 0 & p_r & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & p_r \\ 0 & \cdots & 0 & p_r & 1 \end{bmatrix}; \tag{18}$$

The values of the correlation coefficients p_t and p_r can be selected to represent different types of channels.

It should be noted that in the presence of spatially correlated fading in the MU-MIMO model (1) the channel matrix $\mathbf{H}_p$ of each user can be generated using the Kronecker model (17).

7 Simulation results

Statistical simulations were carried out to investigate the noise immunity of the MU-MIMO system using the conventional MMSE demodulation algorithm (4) and the proposed OGM2 algorithm (8)…(16). The simulation conditions are as follows:

- Antenna configurations: – 128×128, 256×256 and 512×512.
- Number of iterations in the OGM2 algorithm – 32.
- Number of experiments – 2500.
- Spatial fading correlation coefficients at the transmitter and receiver sides – $p_t = p_r = 0.2$.
- Modulation scheme – QAM-16.
- Demodulation algorithms – MMSE и OGM2.
- Number of users – 4.
- Constraint length –4.
- Generator polynomials – (13,15) in octal notation.
- Frame length – 128 бит.
- Code rate – 1/2.

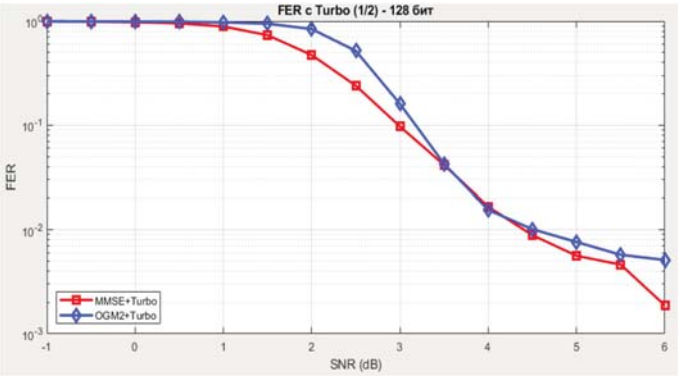


Fig. 3. Performance characteristics of the MU-MIMO system for 4 users with a 128×128 antenna configuration.

Let us consider the noise immunity characteristics of the MU-MIMO system using the conventional MMSE algorithm and the proposed iterative OGM2-based demodulation algorithm. The comparison is performed using two key performance metrics: the Bit Error Rate (BER), defined as the ratio of incorrectly received bits to the total number of received bits, and the Frame Error Rate (FER), which characterizes the proportion of frames containing at least one erroneously received bit. Both metrics are analyzed as functions of the Signal-to-Noise Ratio (SNR).

Figure 3 shows the dependence of the frame error rate (FER with turbo coding) on the signal-to-noise ratio for a MU-MIMO system with a 128×128 antenna configuration, using the MMSE and OGM2 algorithms for 4 users. As can be seen from Fig. 3, the performance loss of the OGM2 algorithm in terms of noise immunity at $FER = 10^{-2}$ is only about 0.1 dB.

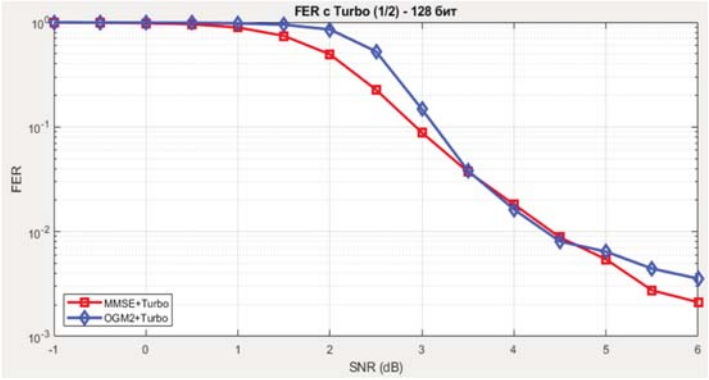


Fig. 4. Performance characteristics of the MU-MIMO system for 8 users with a 128×128 antenna configuration.

Figure 4 presents similar dependencies for 8 users. It can be observed that the noise-robustness loss of the OGM2 algorithm at $FER = 10^{-2}$ is negligible (practically zero).

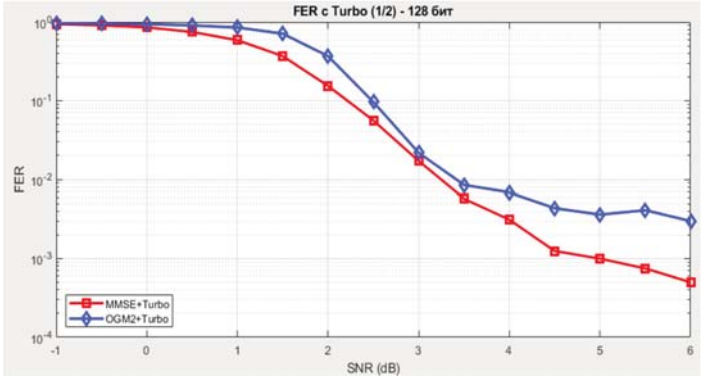


Fig. 5. Performance characteristics of the MU-MIMO system for 4 users with a 256×256 antenna configuration.

Figure 5 shows the dependence of the frame error rate (FER with turbo coding) on the signal-to-noise ratio for a MU-MIMO system with a 256×256 antenna configuration using the MMSE and OGM2 algorithms for 4 users. From the graph, it can be seen that the noise-robustness loss of the OGM2 algorithm at $FER = 10^{-2}$ is approximately 0.2 dB.

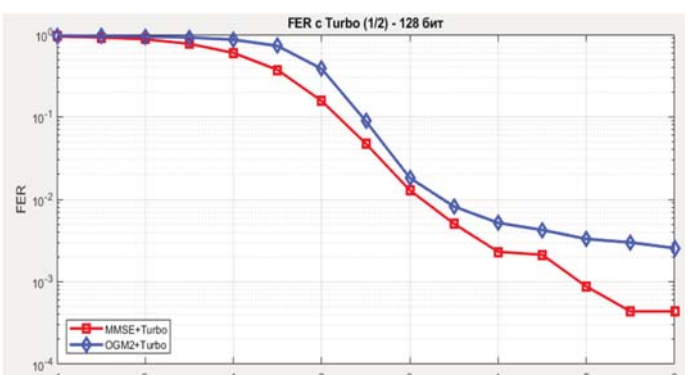


Fig. 6. Performance characteristics of the MU-MIMO system for 8 users with a 256×256 antenna configuration.

Figure 6 presents similar results for 8 users. It can be seen that the noise-robustness loss of the OGM2 algorithm at $FER=10^{-2}$ is also about 0.2 dB.

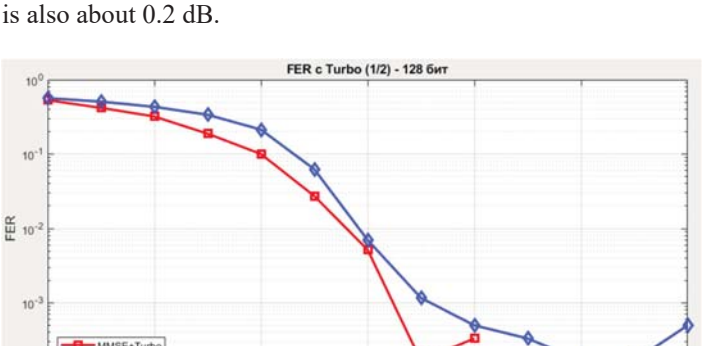


Fig. 7. Performance characteristics of the MU-MIMO system for 4 users with a 512×512 antenna configuration.

Figure 7 shows the dependence of the frame error rate (FER with turbo coding) on the signal-to-noise ratio for a MU-MIMO system with a 512×512 antenna configuration using the MMSE and OGM2 algorithms for 4 users. From the graph, it can be seen that the noise-robustness loss of the OGM2 algorithm at $FER=10^{-2}$ is also approximately 0.2 dB.

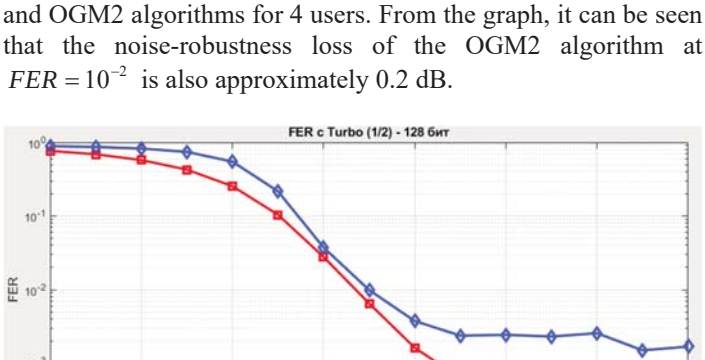


Fig. 8. Performance characteristics of the MU-MIMO system for 8 users with a 512×512 antenna configuration.

Figure 8 presents similar results for 8 users. It can be observed that the noise-robustness loss of the OGM2 algorithm at $FER=10^{-2}$ is also about 0.2 dB.

8 Conclusions

Based on the conducted statistical simulation of a high-dimensional MU-MIMO system with various antenna configurations and numbers of users, using the proposed linear iterative demodulation algorithm based on OGM2, the following conclusions can be drawn.

1. **Reduction of computational complexity.**The proposed OGM2-based demodulation algorithm demonstrates a significant advantage in computational complexity compared to the conventional MMSE algorithm. For a MU-MIMO system with a 128×128 antenna configuration, the gain is approximately 2.0 times; for a 256×256 – configuration, about 3.9 times; and for a 512×512 – configuration, up to 7.9 times. The obtained results show that the efficiency of the OGM2 algorithm increases with the system dimensionality. This makes the proposed algorithm promising for application in Massive MU-MIMO systems, where the use of the conventional MMSE detector becomes challenging due to high computational load and hardware implementation complexity.

2. **Acceptable degradation in noise immunity.** The simulation results show that the proposed iterative algorithm introduces only a minor loss in noise immunity compared to the optimal MMSE algorithm when turbo coding is employed. For different system configurations and numbers of users, the energy loss at $FER=10^{-2}$ s approximately in the range of (0.1-0.2) dB, which can be considered an acceptable trade-off for a significant reduction in computational complexity of 2 to 8 times. In some cases, for example in the 128×128 configuration with 8 users, the difference between OGM2 and MMSE is practically negligible, further confirming the high efficiency of the proposed approach.

3. **Robustness under spatially correlated fading.** The simulation results obtained for a spatial correlation level of $p_t=p_r=0.2$ demonstrate that the proposed algorithm maintains stable performance under correlated fading conditions typical of real wireless channels. This confirms the applicability of the algorithm not only in idealized channel models with independent fading, but also in more realistic MU-MIMO operating scenarios. Therefore, the algorithm can be effectively used in practical wireless communication systems.

4. **Performance under different numbers of users.**The analysis of simulation results for different numbers of users (4 and 8 users) shows that the proposed algorithm maintains high efficiency even as the number of simultaneously served users increases. This indicates good robustness against inter-user interference and confirms its suitability for multi-user systems with high traffic density.

5. **Scalability with system dimensionality.**The results obtained for MIMO systems with 128×128, 256×256 и 512×512 antenna configurations confirm that the computational gain of the proposed algorithm relative to MMSE increases with system dimensionality. This property is particularly important for future Massive MU-MIMO systems, where a further increase in the number of base station antennas is expected. Thus, the OGM2 algorithm demonstrates good scalability and is capable of efficient operation in large-scale systems.

Based on the obtained results, the use of the proposed iterative OGM2-based demodulation algorithm is well justified for high-dimensional multi-user MU-MIMO systems. The algorithm

provides an effective trade-off between computational complexity and noise immunity while maintaining robust performance under realistic conditions with spatial fading correlation and inter-user interference.

References

[1] V.B. Kreyndelin, V.A. Usachev, “LTE-Advanced Pro as the basis for new M2M scenarios,” *T-Comm*. 2017. Vol. 11, no. 3, pp. 28-32. (In Russian)

[2] A. Kishk, X. Chen, “ MIMO Communications – Fundamental Theory, Propagation Channels, and Antenna Systems,” IntechOpen, 2023, 342 p.

[3] D. Borges, P. Montezuma, R. Dinis, M. Beko, “Massive MIMO Techniques for 5G and Beyond – Opportunities and Challenges,” *Electronics*, 2021, 10, 1667.

[4] N. Jayaweera, K. B. S. Manosha, N. Rajatheva and M. Latva-aho, "Minimizing Energy Consumption in Cell-Free Massive MIMO Networks," in *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 13263-13277, Sept. 2024, doi: 10.1109/TVT.2024.3392790.

[5] M.G. Bakulin, V.B. Kreyndelin, D.Yu. Pankratov, “Technology in radio systems on the road to 5G,” Moscow: Hot Line-Telecom, 2018. 280 p. (In Russian)

[6] A. Osinsky, R. Bychkov, M. Kirichenko, V. Lyashev and A. Ivanov, "Justification for Performance Loss Caused by Round-Off Errors in Massive MIMO Detector Based on QR Decomposition," in *IEEE Wireless Communications Letters*, vol. 15, pp. 205-209, 2026, doi: 10.1109/LWC.2025.3624088.

[7] J. Tian, Y. Han, S. Jin, J. Zhang and J. Wang, "Analytical Channel Modeling: From MIMO to Extra Large-Scale MIMO," in *Chinese Journal of Electronics*, vol. 34, no. 1, pp. 1-15, January 2025, doi: 10.23919/cje.2023.00.418.

[8] H. J. Park, H. Chung, S. -W. Choi and J. W. Lee, "Coded Multi-User Extreme Massive MIMO Systems with Gauss Seidel-Aided MMSE-PIC," *2024 15th International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju Island, Korea, Republic of, 2024, pp. 1076-1077, doi: 10.1109/ICTC62082.2024.10826779.

[9] T. Abood, I. Hburi and H. F. Khazaal, "Massive MIMO: An Overview, Recent Challenges, and Future Research Directions," *2021 International Conference on Advance of Sustainable Engineering and its Application (ICASEA)*, Wasit, Iraq, 2021, pp. 43-48, doi: 10.1109/ICASEA53739.2021.9733081.

[10] D. Kim, J. A. Fessler, “Generalizing the optimized gradient method for smooth convex minimization,” *SIAM Journal on Optimization*. 2018. Vol. 28, no. 2, pp. 1920-1950. DOI: 10.1137/17M112124X.

[11] A.V. Gasnikov, “Modern numerical optimization methods. Universal gradient descent method,” Moscow: MCMNO, 2021. 272 p. (In Russian)

[12] V. G. Zhadan, “Optimization methods. Part II. Numerical Algorithms. Tutorial,” Moscow, MFTI, 2015, 320 p. (In Russian)

[13] Donghwan Kim, Jeffrey A. Fessler,”Adaptive Restart of the Optimized Gradient Method for Convex Optimization,” November 29, 2017, pp.1-18, https://arxiv.org/pdf/1703.04641.pdf

[14] Jorge Nocedal, Stephen J. Wright. Numerical Optimization. Second Edition. Springer Science Business Media, LLC, USA, 2006, 684 p.

[15] Chee-Khian Sim, “A First Order Algorithm on an Optimization Problem with Improved Convergence when Problem is Convex,” 14 August 2025. https://optimization-online.org/wp-content/uploads/2025/08/NonconvexFISTA_Ver6.pdf

[16] Ross, I. M., "An Optimal Control Theory for Accelerated Optimization," arXiv:1902.09004, 15 Mar. 2022. https://arxiv.org/pdf/1902.09004

[17] L. Li and J. Hu, "An Efficient Linear Detection Scheme Based on L-BFGS Method for Massive MIMO Systems," in *IEEE Communications Letters*, vol. 26, no. 1, pp. 138-142, Jan. 2022, doi: 10.1109/LCOMM.2021.3121445.

[18] L. Li and J. Hu, "Fast-Converging and Low-Complexity Linear Massive MIMO Detection With L-BFGS Method," in *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10656-10665, Oct. 2022, doi: 10.1109/TVT.2022.3185967.

[19] Yu.E. Nesterov. Introduction to Convex Optimization. Moscow: MCMNO, 2010. 262 p. (In Russian)

[20] J. Golub, Ch. Van Loan. Matrix computations. Moscow: Mir, 1999. 548 p. (In Russian)

[21] M. Bakulin, V. Kreyndelin, A. Rog, D. Petrov, S. Melnik, "Low-complexity iterative MIMO detection based on turbo-MMSE algorithm," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, 550-560, 10.1007/978-3-319-67380-6_51

[22] Y. Zhao, "Research and Application Analysis of Turbo Code Decoding Algorithms," *2024 IEEE 2nd International Conference on Image Processing and Computer Applications (ICIPCA)*, Shenyang, China, 2024, pp. 1-5, doi: 10.1109/ICIPCA61593.2024.10709190.

[23] T. Gendron, E. Boutillon, C. A. Nour and D. Gnaedig, "Revisiting augmented decoding techniques for LTE Turbo Codes," *2021 11th International Symposium on Topics in Coding (ISTC)*, Montreal, QC, Canada, 2021, pp. 1-5, doi: 10.1109/ISTC49272.2021.9594119.

[24] Rolf Johannesson; Kamil Sh. Zigangirov, "Turbo Coding," in *Fundamentals of Convolutional Coding*, IEEE, 2015, pp.567-592, doi: 10.1002/9781119098799.ch9.

[25] V. B. Kreindelin, D. Yu. Pankratov, “Analysis of the Radio Channel Capacity of a MIMO System under the Conditions of Spatially Correlated Fadings,” *Journal of Communications Technology and Electronics*, 2019, Vol. 64. No. 8, pp. 863-869.

[26] M.G. Bakulin, V.B. Kreyndelin, "The problem of spectral efficiency and capacity increase in perspective 6G communication systems", *T-Comm*, 2020, vol. 14, no.2, pp. 25-31. DOI: 10.36724/2072-8735-2020-14-2-25-31.

[27] M.G. Bakulin, T.B.C. Ben Rejeb, V.B. Kreyndelin, D.Y. Pankratov, A.E. Smirnov, "Code domain NOMA in 3GPP specifications: 5G or 6G?" *T-Comm*. 2022, vol 16, no. 1, pp. 4-14. DOI 10.36724/2072-8735-2022-16-1-4-14.

ИТЕРАЦИОННЫЙ АЛГОРИТМ ДЕМОДУЛЯЦИИ НА ОСНОВЕ МЕТОДА ГРАДИЕНТНОГО ПОИСКА ДЛЯ СИСТЕМ MU-MIMO С БОЛЬШИМ ЧИСЛОМ АНТЕНН

Бен Режеб Софиэн Бен Камель, *Московский технический университет связи и информатики, Москва, Россия,* ***sbenrezheb@yandex.ru***

Крейнделин Виталий Борисовичч, *Московский технический университет связи и информатики, Москва, Россия,* ***vitkrend@gmail.com***

Аннотация

В данной работе предлагается и исследуется итерационный алгоритм демодуляции на основе оптимизированного градиентного метода (OGM2) для многопользовательских систем MIMO (MU-MIMO) высокой размерности с учетом пространственной корреляции замираний. Классический алгоритм MMSE, эффективный для малых антенных конфигураций, становится неприменимым в системах Massive MU-MIMO из-за высокой вычислительной сложности. Предлагаемый итерационный подход обеспечивает значительное снижение вычислительных затрат при сохранении приемлемой помехоустойчивости в условиях, приближенных к условиям реальных каналов связи. Приводятся результаты имитационного моделирования для системы MU-MIMO с конфигурациями антенн 128x128, 256x256 и 512x512 для 4 и 8 пользователей при уровне пространственной корреляции замираний, равном 0.2. Показано, что предлагаемый алгоритм обеспечивает снижение вычислительной сложности в 2-8 раз по сравнению с традиционным алгоритмом MMSE при потерях в помехоустойчивости не более 0.1-0.2 дБ в системах с турбокодированием на уровне $FER = 10^{-2}$. Результаты демонстрируют устойчивость алгоритма к пространственной корреляции, что подтверждает его практическую применимость в реальных системах MIMO.

Ключевые слова: *MU-MIMO, Massive MIMO, MMSE, OGM2, пространственная корреляция, помехоустойчивость, вычислительная сложность, турбокод, QAM*

Литература

1. *Крейнделин В.Б., Усачев В.А.* LTE-Advanced Pro как основа для новых сценариев M2M // T-Comm: Телекоммуникации и транспорт. 2017. Т. 11, № 3. С. 28-32.

2. *Kishk A., Chen X.* MIMO Communications –- Fundamental Theory, Propagation Channels, and Antenna Systems; IntechOpen, 2023, 342 p.

3. *Borges D., Montezuma P., Dinis R., Beko M.* Massive MIMO Techniques for 5G and Beyond-Opportunities and Challenges. Electronics 2021, 10, 1667.

4. Jayaweera N., Manosha K. B. S., Rajatheva N., Latvaaho M. Minimizing Energy Consumption in Cell-Free Massive MIMO Networks // IEEE Transactions on Vehicular Technology, vol. 73, no. 9, pp. 13263-13277, Sept. 2024, doi: 10.1109/TVT.2024.3392790.

5. *Бакулин М.Г., Крейнделин В.Б., Панкратов Д.Ю.* Технологии в системах радиосвязи на пути к 5G. М.: Научно-техническое издательство "Горячая линия-Телеком", 2018. 280 с.

6. *Osinsky A., Bychkov R., Kirichenko M., Lyashev V., Ivanov A.* Justification for Performance Loss Caused by Round-Off Errors in Massive MIMO Detector Based on QR Decomposition // IEEE Wireless Communications Letters, vol. 15, pp. 205-209, 2026, doi: 10.1109/LWC.2025.3624088.

7. *Tian J., Han Y., Jin S., Zhang J., Wang J.* Analytical Channel Modeling: From MIMO to Extra Large-Scale MIMO // Chinese Journal of Electronics, vol. 34, no. 1, pp. 1-15, January 2025, doi: 10.23919/cje.2023.00.418.

8. *Park H. J., Chung H., Choi S. -W., Lee J. W.* Coded Multi-User Extreme Massive MIMO Systems with Gauss Seidel-Aided MMSE-PIC // 2024 15th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, Republic of, 2024, pp. 1076-1077, doi: 10.1109/ICTC62082.2024.10826779.

9. *Abood T., Hburi I., Khazaal H. F.* Massive MIMO: An Overview, Recent Challenges, and Future Research Directions // 2021 International Conference on Advance of Sustainable Engineering and its Application (ICASEA), Wasit, Iraq, 2021, pp. 43-48, doi: 10.1109/ICASEA53739.2021.9733081.

10. *Kim D., Fessler J. A.* Generalizing the optimized gradient method for smooth convex minimization // SIAM Journal on Optimization. 2018. Vol. 28, no. 2. - P. 1920-1950. - DOI: 10.1137/17M112124X.

11. *Гасников А. В.* Современные численные методы оптимизации. Метод универсального градиентного спуска. М.: МЦМНО 2021. 272 с.

12. *Жадан В. Г.* Методы оптимизации. Часть II. Численные алгоритмы," учебное пособие, Москва, МФТИ, 2015, 320 с.

13. *Donghwan Kim, Jeffrey A. Fessler.* Adaptive Restart of the Optimized Gradient Method for Convex Optimization. November 29, 2017, pp.1-18, https://arxiv.org/pdf/1703.04641.pdf

14. *Jorge Nocedal, Stephen J. Wright.* Numerical Optimization. Second Edition. Springer Science Business Media, LLC, USA, 2006, 684 p.

15. *Chee-Khian Sim.* A First Order Algorithm on an Optimization Problem with Improved Convergence when Problem is Convex. 14 August 2025. https://optimization-online.org/wp-content/uploads/2025/08/NonconvexFISTA_Ver6.pdf

16. *Ross I. M.* An Optimal Control Theory for Accelerated Optimization," arXiv:1902.09004, 15 Mar. 2022. https://arxiv.org/pdf/1902.09004

17. *Li L., Hu J.* An Efficient Linear Detection Scheme Based on L-BFGS Method for Massive MIMO Systems // IEEE Communications Letters, vol. 26, no. 1, pp. 138-142, Jan. 2022, doi: 10.1109/LCOMM.2021.3121445.

18. *Li L., Hu J.* Fast-Converging and Low-Complexity Linear Massive MIMO Detection With L-BFGS Method // IEEE Transactions on Vehicular Technology, vol. 71, no. 10, pp. 10656-10665, Oct. 2022, doi: 10.1109/TVT.2022.3185967.

19. *Нестеров Ю.Е.* Введение в выпуклую оптимизацию. М.: МЦНМО, 2010. 262 с.

20. *Голуб Дж., Ван Лоун Ч.* Матричные вычисления. Пер. с англ. М.: Мир, 1999. 548 с.

21. *Bakulin M., Kreyndelin V., Rog A., Petrov D., Melnik S.* Low-complexity iterative MIMO detection based on turbo-MMSE algorithm // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2017, 550-560, 10.1007/978-3-319-67380-6_51

22. *Zhao Y.* Research and Application Analysis of Turbo Code Decoding Algorithms // 2024 IEEE 2nd International Conference on Image Processing and Computer Applications (ICIPCA), Shenyang, China, 2024, pp. 1-5, doi: 10.1109/ICIPCA61593.2024.10709190.

23. *Gendron T., Boutillon E., Nour C. A., Gnaedig D.* Revisiting augmented decoding techniques for LTE Turbo Codes // 2021 11th International Symposium on Topics in Coding (ISTC), Montreal, QC, Canada, 2021, pp. 1-5, doi: 10.1109/ISTC49272.2021.9594119.

24. Rolf Johannesson; Kamil Sh. Zigangirov, "Turbo Coding," in Fundamentals of Convolutional Coding, IEEE, 2015, pp. 567-592, doi: 10.1002/9781119098799.ch9.

25. *Kreindelin V. B., Pankratov D. Yu.* Analysis of the Radio Channel Capacity of a MIMO System under the Conditions of Spatially Correlated Fadings // Journal of Communications Technology and Electronics 2019, Vol. 64. № 8, pp. 863-869.

26. *Бакулин М.Г., Крейнделин В.Б.* Проблема повышения спектральной эффективности и емкости в перспективных системах связи 6G // T-Comm: Телекоммуникации и транспорт. 2020. Т. 14. № 2. С. 25-31. DOI 10.36724/2072-8735-2020-14-2-25-31. EDN HHVLBC.

27. *Бакулин М. Г., Бен Режеб Т. Б. К., Крейнделин В. Б., Панкратов Д. Ю., Смирнов А. Э.* Технология NOMA с кодовым разделением в 3GPP: 5G или 6G? // T-Comm: Телекоммуникации и транспорт. 2022. Т. 16, № 1. С. 4-14. DOI 10.36724/2072-8735-2022-16-1-4-14. EDN ZBEDVU.

Информация об авторах:

Бен Режеб Софиэн Бен Камель, *аспирант, Московский технический университет связи и информатики, Москва, Россия*

Крейнделин Виталий Борисович, *заведующий кафедрой "Теория электрических цепей", профессор, д.т.н., Московский технический университет связи и информатики, Москва, Россия*