Московский Государственный Университет
имени М.В. Ломоносова
Российская Академия Наук
Международная Академия Технологических Наук
Российская Академия Естественных Наук

Интеллектуальные Системы.

Теория и приложения

ТОМ **30** ВЫПУСК 3 * **2026**

МОСКВА

УДК 519.95; 007:159.955
ББК 32.81

ISSN 2411–4448
Издаётся с 1996 г.

Главный редактор: д.ф.-м.н., профессор Э. Э. Гасанов

Редакционная коллегия:

к.ф.-м.н., доц. А. В. Галатенко **(зам. главного редактора)**
д.ф.-м.н., проф. А. А. Часовских **(зам. главного редактора)**

д.ф.-м.н., проф. В. В. Александров, д.ф.-м.н., проф. С. В. Алёшин, д.ф.-м.н., проф. А. Е. Андреев, д.ф.-м.н., проф. Д. Н. Бабин, к.ф.-м.н., доц. Г. В. Боков, проф. К. Вашик, проф. Я. Деметрович, академик РАН, д.ф.-м.н., проф. Ю. Л. Ершов, к.ф.-м.н., н.с. Г. В. Калачёв, проф. Г. Килибарда, д.ф.-м.н., проф. В. Н. Козлов, к.ф.-м.н., в.н.с. В. А. Носов, д.ф.-м.н., проф. А. С. Подколзин, д.ф.-м.н., проф. Ю. П. Пытьев, д.т.н., проф. А. П. Рыжов, к.ф.-м.н., доц. Э. М. Сайдаматов, академик РАН, д.т.н., проф. А. С. Сигов, к.ф.-м.н., доц. А. С. Строгалов, проф. Б. Тальхайм, проф. Ш. Ушчумлич, д.ф.-м.н., проф. А. В. Чечкин, к.ф.-м.н. Ш. Н. Шералиев, к.ф.-м.н. Р. Шчепанович.

Секретари редакции: И. О. Бергер, Е. В. Кузнецова

В журнале «Интеллектуальные системы. Теория и приложения» публикуются научные достижения в области теории и приложений интеллектуальных систем, новых информационных технологий и компьютерных наук.
Издание журнала осуществляется под эгидой МГУ имени М.В. Ломоносова, Научного Совета по комплексной проблеме «Кибернетика» РАН, Отделения «Математическое моделирование технологических процессов» МАТН.
Учредитель журнала: ООО «Интеллектуальные системы».
Журнал входит в список изданий, включенных ВАК РФ в реестр публикаций материалов по кандидатским и докторским диссертациям по математике и механике. Входит в дополнительный перечень научных изданий, в которых могут быть опубликованы результаты диссертаций, защищаемых в МГУ.

Индекс подписки на журнал: 64559 в каталоге НТИ «Роспечать».

Адрес редакции: 119991, Москва, ГСП-1, Ленинские Горы, д. 1, механико-математический факультет, комн. 12-01.
Адрес издателя: 115230, Россия, Москва, Хлебозаводский проезд, д. 7, стр. 9, офис 9. Тел. +7 (495) 939-46-37, e-mail: mail@intsysmagazine.ru

*) Прежнее название журнала: «Интеллектуальные системы».

© ООО «Интеллектуальные системы», 2026.

ОГЛАВЛЕНИЕ

Станислав Владимирович Алешин (к 85-летию со дня рождения) 5

Часть 1. Общие проблемы теории интеллектуальных систем

Артамонов С. А., Миронов А. М. Использование иерархического поиска ближайших соседей для заполнения пропущенных значений в табличных данных 8

Бергер И. О., Хлебникова А. А. Алгоритм определения типов восприятия трудных жизненных ситуаций на основе краткой версии опросника «Типы ориентаций в трудной ситуации» ТОРС-36 30

Грабовой А. В. От VC-размерности к законам масштабирования: эволюция теоретических подходов к оценке сложности и обобщающей способности 53

Часть 2. Специальные вопросы теории интеллектуальных систем

Давыдова Д. Н. Квантизация языковых моделей с помощью дифференцируемого поиска архитектуры нейронной сети 88

Ковалёва Е. С. Метод распознавания изображений на основе выпуклой линейной оболочки над конечным полем 113

Родин С. Б. О максимально сложно реализуемых отображениях 126

Часть 3. Математические модели

Бабин Д. Н. Почти одноместные базисы автоматов с операцией суперпозиции. 142

Гасанов Э. Э. О слабости одномерных идемпотентных клеточных автоматов с локаторами 151

Калашников М. Э. Проблема полноты в классе линейных p-адических автоматов 170

Юсуфова А. А., Калачев Г. В. Оценки характеристик кодов Таннера на графе проективного пространства 189

Часть 4. Материалы семинаров

Доклады семинара «Теория автоматов» 218

Станислав Владимирович Алешин (к 85-летию со дня рождения)

14 августа 2026 года исполняется 85 лет профессору кафедры математической теории интеллектуальных систем механико-математического факультета МГУ, доктору физико-математических наук Станиславу Владимировичу Алешину.

Редакционная коллегия журнала «Интеллектуальные системы. Теория и приложения» от всей души поздравляет Станислава Владимировича с этим замечательным юбилеем!

С.В. Алешин — выдающийся российский ученый-математик, автор фундаментальных работ в алгебраической теории автоматов и распознавания образов. С.В. Алешиным получен блестящий научный результат — средствами теории автоматов найдено решение одной из центральных проблем алгебры — проблемы Бернсайда о существовании бесконечной конечнопорожденной периодической группы. В теорию автоматов прочно вошло устойчивое сочетание «группы алешинского типа». В докторской диссертации С.В. Алешиным развит новый подход к построению динамических распознающих систем. Эти результаты вошли в учебники, широко известны мировой научной общественности и получили заслуженное признание. С.В. Алешин является автором таких монографий как «Алгебраические системы автоматов» и «Распознавание динамических образов». Книга «Введение в теорию автоматов», соавтором которой является С.В. Алешин, стала настольной для специалистов в области кибернетики и дискретной математики и до сих пор остается одной из лучших в мире монографий по теории автоматов.

Прикладные разработки С.В. Алешина относятся к области оптимального синтеза микросхем и вычислительной литографии. Разработанные методы позволили повысить производительность и эффективность промышленного производства микросхем, что подтверждается более чем 40 патентами США.

С.В. Алешин читает спецкурсы по теории автоматов и теории алгебраических систем автоматов, ведет специальные семинары «Дискретный анализ» и «Методы вычислительной литографии для создания чипов», а также руководит научной работой студентов и аспирантов.

С.В. Алешин — блестящий педагог и наставник молодежи. Среди учеников С.В. Алешина 2 доктора и 10 кандидатов физико-математических наук.

С.В. Алешин является членом редколлегии ВАКовского журнала «Интеллектуальные системы. Теория и приложения», входил в состав оргкомитетов многих научных конференций.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

В день Вашего 85-летия мы желаем Вам, дорогой Станислав Владимирович, крепкого здоровья, неиссякаемой творческой энергии для новых свершений и радости от общения с молодыми талантами! Пусть Ваша мудрость как путеводная звезда еще долгие годы освещает дорогу российской науке!

Редакционная коллегия журнала
«Интеллектуальные системы. Теория и приложения»

Часть 1
Общие проблемы теории интеллектуальных систем

Использование иерархического поиска ближайших соседей для заполнения пропущенных значений в табличных данных

С. А. Артамонов*, А. М. Миронов‡

Пропущенные значения являются распространённой проблемой в анализе данных, для решения которой разработано множество методов. Одним из наиболее популярных подходов является заполнение пропущенных значений с использованием метода k-ближайших соседей (KNN), основанного на информации от схожих наблюдений. Данный подход широко применяется в анализе больших данных, где объёмы выборок могут достигать миллиардов объектов, что делает задачу ускорения KNN особенно актуальной из-за его линейной сложности. В данной работе предлагается алгоритм ускоренного заполнения пропущенных значений на основе методов приближённого поиска ближайших соседей (ANN) и анализируется соотношение между скоростью работы и качеством заполнения. Экспериментальные результаты показывают, что использование ANN позволяет существенно сократить время вычислений при сохранении высокого качества заполнения пропущенных значений.

Ключевые слова: заполнение пропущенных значений в табличных данных, приближенный поиск ближайших соседей.

1. Введение

Проблема пропущенных значений при обработке табличных данных широко распространена, и обычно приводит к снижению качества обработки

* *Артамонов Сергей Александрович* — аспирант каф. математической теории интеллектуальных систем мех.-мат. факультета МГУ, e-mail: sergei.artamonov@math.msu.ru, ORCID: 0009-0001-6368-8266.

Artamonov Sergei Aleksandrovich — PhD student, Department of Mathematical Theory of Intelligent Systems, Faculty of Mechanics and Mathematics, Lomonosov Moscow State University.

‡ *Миронов Андрей Михайлович* — к.ф.-м.н., доцент каф. математической теории интеллектуальных систем мех.-мат. факультета МГУ, e-mail: andrew.mironov@math.msu.ru, ORCID: 0000-0002-9132-7804.

Mironov Andrei Mikhailovich — PhD, Associate Professor, Department of Mathematical Theory of Intelligent Systems, Faculty of Mechanics and Mathematics, Lomonosov Moscow State University.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

такого рода данных при использовании алгоритмов классического машинного обучения. Наличие пропущенных значений в данных создаёт серьёзные ограничения для применения ряда алгоритмов машинного обучения, особенно тех, которые основаны на функциях потерь, допускающих аналитическое представление, и требуют полного числового представления признаков. Пропущенные значения способны приводить к систематическому искажению распределений признаков и, как следствие, к смещению оценок параметров модели. Это происходит в тех случаях, когда механизм возникновения пропусков не является полностью случайным. Искажения в распределении признаковых описаний приводят к переобучению алгоритмов классического машинного обучения, поэтому пропущенные значения могут существенно влиять на точность и обобщающую способность моделей машинного обучения [1].

На сегодняшний день существует широкий спектр алгоритмов заполнения пропущенных значений в табличных данных, однако не существует единой зарекомендовавшей себя методики выбора наиболее надёжного алгоритма с учетом заданных ограничений по вычислительным ресурсам и времени. Иногда для достижения приемлемого качества достаточно использовать базовые методы — такие как заполнение пропущенных значений постоянной величиной, нулём, средним или медианой соответствующего признака.

Причины возникновения пропущенных данных разнообразны и включают ошибки на этапе сбора информации (например, при проведении опросов или измерений), неточности при ручном вводе данных, а также сбои при передаче информации. В некоторых случаях значения теряются или искажаются в процессе предобработки, либо вовсе не собираются. Пропущенные значения могут существенно влиять на точность и обобщающую способность моделей машинного обучения [1]. Хотя некоторые современные алгоритмы способны обрабатывать пропущенные значения без дополнительной предобработки, такие подходы нередко сопровождаются снижением качества модели. Один из известных примеров — алгоритм eXtreme Gradient Boosting [2], обладающий встроенными механизмами обработки пропущенных значений. В работе Aydin и соавт. [3] показано, что модели XGBoost сами по себе зачастую демонстрируют худшие результаты по сравнению с подходами, использующими специализированные методы заполнения пропущенных значений. Подобные подходы отличаются вычислительной эффективностью и простотой реализации. Однако более сложные методы могут обеспечить более высокую точность восстановления пропущенных значений, что сопровождается увеличением вычислительных затрат. Одним из наиболее широко используемых подходов является метод KNN Imputer [4], основанный на

алгоритме k ближайших соседей. Данный метод в настоящее время применяется в самых различных предметных областях. Несмотря на свою эффективность, этот подход имеет существенный недостаток — высокую вычислительную сложность. Как отмечают Murti и соавт. [5], Acuña и Rodríguez [6], а также многие другие исследователи, метод требует анализа всего набора данных для поиска схожих объектов, что делает его особенно ресурсоёмким при работе с большими объёмами данных.

В то же время существует класс алгоритмов приближённого поиска ближайших соседей (Approximate Nearest Neighbor, ANN), разработанных с целью ускорения стандартного метода k ближайших соседей путём разбиения пространства поиска и ограничения области подбора соседей для каждого запроса только релевантным подмножеством этого пространства [7–9]. В настоящее время известен широкий спектр методов и программных пакетов для реализации ANN-подходов. В работе Aumüller и соавт. [10] проведено сравнение наиболее распространённых алгоритмов приближённого поиска ближайших соседей.

В этой работе мы проводим анализ возможных рисков, связанных с применением алгоритма Hierarchical Navigable Small World для ускорения процесса заполнения пропущенных значений, а также проводим оценку этого ускорения на специально собранном с этой целью наборе данных.

2. Обзор литературы

2.1. Пропущенные значения

Проблема пропущенных значений исследуется с середины XX века; одними из первых значимых работ в этой области стали исследования Beale и Little [35], в которых проводилась сравнительная оценка ранних алгоритмов заполнения пропусков. Эти методы, как правило, интерпретировали заполнение как задачу регрессии на основе полных данных. Современные подходы сохраняют эту концептуальную основу, но используют более совершенные регрессионные алгоритмы. Одним из наиболее распространённых на практике методов остаётся KNN Imputer [4], реализующий регрессионный подход посредством алгоритма k ближайших соседей.

В последние годы было предложено множество более сложных методов восстановления пропущенных данных. Так, в 2018 году Yoon и соавт. представили подход GAIN, основанный на генеративно-состязательных сетях (GAN) [36]. В том же году Ma и соавт. предложили алгоритм импутации на основе байесовского вывода [37]. Среди других популярных методов можно отметить MICE [38], MissForest [39], а также методы, основанные на автоэнкодерах, такие как MIDA [40].

Несмотря на достижения в области обработки пропущенных данных, метод KNN Imputer остаётся широко используемым при работе с большими наборами данных — например, в банковской и медицинской отраслях — благодаря рациональному соотношению вычислительной эффективности, простоты реализации и масштабируемости. В данной работе мы не стремимся превзойти современные подходы к заполнению пропусков по качеству восстановления данных. Цель исследования состоит в том, чтобы предложить усовершенствование одного из наиболее распространённых методов и оценить его влияние на вычислительные затраты и точность заполнения пропусков.

Важно отметить, что не все пропущенные значения имеют одинаковую природу. Широко используемая классификация пропущенных данных основана на связи между отсутствующими значениями и наблюдаемыми или ненаблюдаемыми величинами в наборе данных, а также на факторах, вызывающих пропуски [17, 18]. Выделяют три основных типа:

- **Missing completely at random (MCAR)**: В этом случае пропуски возникают полностью случайно и не зависят ни от наблюдаемых, ни от ненаблюдаемых значений в наборе данных. Вероятность отсутствия значения одинакова для всех наблюдений. MCAR могут возникать из-за случайных технических проблем или внешних факторов, не связанных с данными.
- **Missing at random (MAR)**: Здесь вероятность пропуска связана с другими наблюдаемыми переменными, но не зависит от самого отсутствующего значения. Например, данные могут чаще отсутствовать для людей определённого возраста или профессии, то есть пропуски объяснимы известными признаками.
- **Missing not at random (MNAR)**: В данном случае вероятность пропуска зависит от значения самого отсутствующего элемента. Например, если респондент не указывает свой доход в налоговой декларации, чтобы избежать раскрытия информации, вероятность пропуска напрямую зависит от значения величины. Работа с MNAR-данными особенно сложна, так как механизм возникновения пропусков не наблюдаем.

2.2. Приближенный поиск ближайших соседей

Методы приближённого поиска ближайших соседей (Approximate Nearest Neighbors, ANN) представляют собой эффективный способ ускорения поиска ближайших соседей и находят широкое применение в различных областях науки. В частности, в компьютерном зрении поиск ближайших

соседей активно используется для идентификации схожих изображений в больших коллекциях [11][12]. Аналогично, в задачах обработки естественного языка эти методы применяются для извлечения релевантной информации из крупных текстовых корпусов, что характерно для систем типа RAG [13]. В связи с этим методы приближённого поиска ближайших соседей стали популярными как средство значительного повышения вычислительной эффективности систем, основанных на поиске ближайших соседей [14].

В данной работе мы используем один из хорошо известных методов приближенного поиска ближайших соседей — алгоритм Hierarchical Navigable Small World (HNSW), предложенный Malkov и Yashunin в 2016 году [19]. Метод основан на построении иерархической структуры графов Small World [20], где элементы набора данных представлены вершинами графа. На сегодняшний день HNSW является одним из наиболее широко используемых методов для работы с наборами данных среднего размера [26]. В нашем исследовании применяется наиболее распространённая реализация алгоритма, доступная через программный пакет HNSWlib.

Выбор рассматриваемого метода был основан на соображениях, изложенных в работе [26], где предлагается стратегия подбора индексных структур в зависимости от ключевых характеристик данных. В рамках данного подхода методы индексирования классифицируются на плоские индексы, графовые индексы и индексы на основе инвертированных файлов (IVF). Метод HNSW является наиболее распространённым и практически значимым представителем графовых индексных структур для приближённого поиска ближайших соседей и рассматривается как основной метод для задачи заполнения пропусков в датасетах умеренного размера, не относящихся ни к малым, ни к крайне крупным выборкам.

3. Предлагаемый метод

В данной работе предлагается новый метод заполнения пропущенных значений, основанный на общей идее метода k ближайших соседей (KNN). Основной целью является устранение одного из наиболее существенных недостатков классического подхода KNN при обработке больших массивов данных — длительного времени работы при применении к большим наборам данных, обусловленного линейной сложностью данного алгоритма. Необходимость вычисления расстояний между всеми парами объектов в наборе данных становится серьёзным ограничением, особенно при работе с промышленными датасетами умеренного и крупного размера.

Для преодоления данного ограничения в работе применяется приближённый поиск ближайших соседей, позволяющий существенно снизить вычислительные затраты по сравнению с точным KNN-поиском. Стандартная процедура поиска k ближайших соседей заменяется приближённым поиском на основе индексной структуры Hierarchical Navigable Small World.

Схематическое описание предлагаемого метода представлено в виде следующего алгоритма:

Входные данные: неполная матрица данных $X \in \mathbb{R}^{N \times d}$.

Выходные данные: матрица с заполненными пропущенными значениями $\tilde{X} \in \mathbb{R}^{N \times d}$.

1. Инициализировать результирующую матрицу $\tilde{X}$ на основе исходной матрицы X.
2. Заменить все пропущенные значения (NaN) в матрице X нулевыми значениями, получив матрицу $\bar{X}$.
3. Построить ANN-индекс I в соответствии с выбранной стратегией приближённого поиска ближайших соседей.
4. Для каждого объекта $i = 1, 2, \ldots, N$ выполнить следующие действия:
   а) Обозначить через x_i i-ю строку матрицы $\bar{X}$.
   б) С использованием ANN-индекса I найти k приближённых ближайших соседей для вектора x_i:
   $$X_{\text{nearest}} = I(x_i, k).$$
   в) Для каждого пропущенного значения в i-й строке матрицы $\tilde{X}$:
      i. Исключить из множества X_{nearest} элементы, содержащие пропуск в искомом столбце и заменить эти элементы на следующих соседей, в соответствии с ANN-индексом I, пока не будет набрано нужное количество приближённых ближайших соседей

         При практической реализации удобно выполнять этот пункт, беря небольшой запас по k на шаге 4.б
      ii. вычислить среднее значение соответствующего признака по множеству X_{nearest}, исключая пропущенные значения (NaN);
      iii. заменить пропущенное значение полученным средним.
5. Вернуть матрицу $\tilde{X}$ как результат работы алгоритма.

На первом этапе все пропущенные значения в исходном наборе данных заменяются нулевыми значениями. Данный шаг предварительной обработки соответствует подходу, используемому в наиболее распространённой реализации метода заполнения пропущенных значений на основе KNN, представленной в библиотеке scikit-learn [15]. Такое заполнение необходимо для обеспечения корректного вычисления расстояний между объектами.

Далее на основе полученной матрицы данных строится индекс HNSW, который используется для выполнения приближённого поиска ближайших соседей. После построения индекса для каждой строки матрицы X определяется множество из K ближайших соседей. Следует отметить, что данный этап допускает наличие некоторой погрешности, что является характерной особенностью приближённых методов поиска ближайших соседей.

На заключительном этапе пропущенные значения в каждой строке восстанавливаются путём замены их на среднее значение соответствующего признака, вычисленное по найденному множеству соседей. При этом значения NaN, присутствующие у самих соседей, исключаются из расчёта среднего.

4. Данные

4.1. Искусственное создание пропущенных значений

Для оценки качества заполнения пропущенных значений в экспериментах использовались наборы данных, не содержащие исходных пропусков. В связи с этим пропущенные значения вводились искусственно, что потребовало корректного моделирования различных механизмов возникновения значений NaN. В соответствии с общепринятыми определениями были реализованы различные стратегии внесения пропусков, соответствующие каждому типу пропущенных значений.

MCAR (Missing Completely At Random). В случае MCAR пропущенные значения возникают полностью случайным образом и не зависят ни от наблюдаемых, ни от ненаблюдаемых переменных. Соответственно, стратегия внесения NaN заключается в случайном выборе индексов в каждом столбце с последующей заменой соответствующих значений на NaN. Каждое значение в наборе данных имеет одинаковую вероятность быть удалённым, при этом общая доля пропусков определяется параметрами эксперимента. Формально данный процесс может быть описан

следующим образом:

$$\mathrm{NaN}_{i,j} \sim \mathrm{Bern}(p), \tag{1}$$

где p — фиксированный параметр, задающий вероятность замены значения на NaN, i и j — индексы строки и столбца соответственно, а $\mathrm{NaN}_{i,j}$ — бинарная переменная, указывающая на наличие пропущенного значения.

MAR (Missing At Random). В сценарии MAR вероятность появления пропущенного значения зависит от значений других признаков в наборе данных. Для моделирования данного типа пропусков предполагается, что значения NaN вводятся в некоторый столбец j, а вероятность пропуска определяется функцией от остальных признаков в той же строке. В частности, вычисляется нормированная сумма всех признаков строки, за исключением целевого столбца j, которая используется для задания вероятности внесения пропуска:

$$\begin{cases} \mathrm{NaN}_{i,j} \sim \mathrm{Bern}(p), \\ p \propto X_{i,1} + X_{i,2} + \cdots + X_{i,j-1} + X_{i,j+1} + \ldots \end{cases} \tag{2}$$

Здесь i и j обозначают индексы строки и столбца, а $X_{i,k}$ — значение признака k в строке i. В рамках данной модели вероятность появления пропущенного значения в столбце j возрастает с увеличением суммы остальных признаков в соответствующей строке.

MNAR (Missing Not At Random). В случае MNAR вероятность пропуска зависит от самого значения, которое становится пропущенным. Для моделирования данного механизма вероятность появления NaN задаётся как величина, пропорциональная значению соответствующего элемента:

$$\begin{cases} \mathrm{NaN}_{i,j} \sim \mathrm{Bern}(p), \\ p \propto X_{i,j}. \end{cases} \tag{3}$$

Данный подход отражает типичную ситуацию MNAR, например, когда экстремальные значения с большей вероятностью оказываются пропущенными вследствие эффектов цензурирования или систематических искажений при сборе данных.

Следует отметить, что в рамках настоящего исследования пропущенные значения рассматриваются исключительно в числовых признаках. Такой подход соответствует распространённой практике в смежных исследованиях и позволяет избежать дополнительной сложности, связанной с обработкой категориальных переменных.

4.2. Наборы данных

В экспериментальной части используются следующие наборы данных, к которым применяются описанные выше методы искусственного внесения пропущенных значений.

Набор данных 1 (регрессия): California Housing Dataset [16]. Набор данных содержит информацию о жилой недвижимости в штате Калифорния, полученную на основе переписи населения США 1990 года. Он включает 20 640 наблюдений с 8 числовыми признаками.

Набор данных 2 (классификация): Loan Approval Dataset [21]. Данный набор данных содержит финансовую информацию о клиентах банка и соответствующее решение о выдаче кредита. Он включает 45 000 наблюдений с 13 признаками (8 числовых и 5 категориальных) и одну бинарную целевую переменную, принимающую значение 1 в случае одобрения кредита и 0 в противном случае.

Набор данных 3 (классификация): Air Quality Dataset [22]. Набор данных состоит из 5 000 наблюдений, описывающих качество воздуха в различных регионах, и содержит 9 числовых признаков. Целевая переменная категоризует качество воздуха на четыре класса: хорошее, умеренное, плохое и опасное.

Набор данных 4 (классификация): MAGIC Gamma Telescope Dataset [25]. Синтетический набор данных, содержащий 19 020 наблюдений с 10 числовыми признаками, полученными на основе характеристик гамма-событий, зарегистрированных наземным черенковским телескопом. Задача состоит в классификации событий на сигналы гамма-частиц и адронный фон; целевая переменная является бинарной.

Набор данных 5 (классификация): Mushrooms Dataset [23]. Набор данных включает 54 000 наблюдений с 8 числовыми признаками, описывающими характеристики грибов. Целевая переменная указывает на съедобность гриба: значение 0 соответствует съедобным, а значение 1 — ядовитым.

Набор данных 6 (регрессия): Paris Housing Dataset [24]. Синтетический набор данных, включающий 10 000 наблюдений, описывающих цены на жильё в городской среде, смоделированной по аналогии с Парижем.

5. Численные эксперименты

5.1. Описание численного эксперимента

Потенциальные преимущества предлагаемого метода могут быть оценены через анализ соотношения между качеством заполнения пропущенных значений и временем вычислений. Соответственно, наши эксперименты направлены на количественную оценку этих двух аспектов и сравнение их с базовым методом KNN-заполнения пропусков.

В работе проводятся два типа экспериментов:

1. Прямая оценка качества заполнения пропущенных значений
2. Оценка через метрики качества задачи регрессии или классификации (далее - **базовая задача**)

Во втором случае для каждого варианта датасета обучалась модель градиентного бустинга (XGBoost) с одинаковыми гиперпараметрами для всех версий, чтобы обеспечить сопоставимость результатов.

Численный эксперимент описывается следующей последовательностью действий:

1. Ввести $p\%$ пропущенных значений в набор данных с использованием подходящей стратегии для выбранного типа пропусков.
2. Применить метод заполнения пропущенных значений и сохранить все полученные версии датасета.
3. Для каждой версии выполнить прямую оценку качества заполнения.
4. Для каждой версии обучить модель XGBoost и оценить показатели на соответствующей базовой задаче

Такой эксперимент зависит от нескольких ключевых параметров:

- Количество ближайших соседей K, используемых при заполнении пропусков.
- Доля пропущенных значений p.
- Тип пропусков: **MCAR**, **MAR** или **MNAR**.

Мы исследовали все возможные комбинации этих параметров, выполняя описанную процедуру для каждой конфигурации. Сетка значений параметров определялась следующим образом:

- Доля пропусков (p): 5%, 10%, 15%, 20%, 30%

- Тип пропусков: **MCAR**, **MAR**, **MNAR**
- Количество ближайших соседей (K): нелинейная сетка:
  - От 3 до 30 с шагом 1
  - От 30 до 100 с шагом 5
  - От 100 до 1000 с шагом 25

Также мы зафиксировали параметры, определяющие алгоритм поиска при помощи HNSW. Основные параметры - это количество двунаправленных связей, создаваемых для каждого нового элемента при построении графа (M) и количество кандидатов, которые учитываются при построении графа (ef_construction). Мы зафиксировали эти значения: M = 16, ef_construction = 64.

Поскольку неизвестно, существуют ли стабильные зависимости между качеством заполнения пропусков и результатами решения соответствующих базовых задач, а также как эти зависимости связаны с типом пропусков, мы используем методологию оценки, учитывающую оба аспекта.

Для оценки решения базовых задач применяются различные метрики в зависимости от типа задачи: для регрессии используется коэффициент детерминации (R^2) [27], а для классификации — F_1-мера [28]. Одновременно мы оцениваем качество заполнения пропущенных значений напрямую, сравнивая исходные данные с восстановленными данными после применения каждого метода.

Однако не существует общепринятого способа оценки качества заполнения пропусков. Shadbahr et al. [29] выделяют три принципиально различных подхода, ни один из которых на сегодняшний день не является общепринятым. Следуя современным подходам [30, 31], мы используем наиболее простой из методов, описанных в [29], известный как оценка расхождения по образцам (sample-wise discrepancy). Этот метод сравнивает заполненные значения с исходными на уровне каждой строки, используя коэффициент детерминации (R^2).

Также представляют интерес изменения ключевых характеристик решения при различных значениях числа ближайших соседей K. Ожидается, что время заполнения пропусков будет расти с увеличением K, однако важно оценить, как качество восстановления данных и показатели качества базовых задач реагируют на изменения K. Поскольку оптимальное значение K заранее неизвестно для каждой задачи, для каждого тестируемого значения K фиксируются время заполнения пропусков, метрики решения базовых задач и качество восстановления данных.

Численные эксперименты выполнялись на CPU без использования GPU. Расчёты запускались на вычислительном кластере на базе серверных процессоров Intel Xeon с 56 физическими ядрами на узел. Для одного независимого эксперимента выделялся один поток; внутреннее распараллеливание алгоритмов не использовалось. Параллельный запуск применялся только между независимыми экспериментами.

5.2. Результаты численных экспериментов

В каждом эксперименте мы оцениваем три ключевых показателя: **(а)** качество заполнения пропущенных значений в соответствии с ранее описанной методологией оценки; **(б)** время заполнения в секундах; и **(в)** качество решения соответствующей базовой задачи — измеряемое с помощью коэффициента детерминации R^2 для задач регрессии и метрики F_1 для задач классификации. Поскольку оптимальное число ближайших соседей (K) заранее неизвестно для конкретного набора данных, мы усредняем значение рассматриваемых характеристик по всему рассматриваемому набору значений K.

Таблица 1: Среднее время заполнения пропусков для различных наборов данных

P	Датасет	T_{KNN}	T_{HNSW}
5	CH	10.1	**0.72**
5	Loan	44.06	**1.52**
5	Air	10.17	**0.69**
5	Mag	3.92	**0.49**
5	PH	59.13	**1.58**
5	Mus	0.63	**0.16**
10	CH	18.49	**0.94**
10	Loan	89.72	**2.92**
10	Air	18.12	**0.92**
10	Mag	6.81	**0.67**
10	PH	109.79	**2.13**
10	Mus	1.13	**0.22**
15	CH	25.35	**1.11**
15	Loan	122.24	**3.16**
15	Air	24.12	**1.14**
15	Mag	9.16	**0.84**
15	PH	150.73	**2.64**
15	Mus	1.56	**0.27**
20	CH	31.23	**1.34**
20	Loan	146.95	**3.99**
20	Air	29.53	**1.36**
20	Mag	11.59	**1.01**
20	PH	190.47	**3.24**
20	Mus	1.87	**0.33**
30	CH	47.34	**1.75**
30	Loan	194.96	**4.48**
30	Air	39.32	**1.80**
30	Mag	15.88	**1.34**
30	PH	253.50	**4.27**
30	Mus	2.50	**0.43**

Таблица 2: Среднее изменение качества заполнения пропущенных значений на уровне отдельных наблюдений при замене KNN на ANN на основе HNSW (%)

P%	Dataset	MAR	MNAR	MCAR
5	CH	-1.28	-0.54	-1.02
5	Loan	0.17	0.28	0.08
5	Air	-1.05	-0.90	-0.50
5	Mag	-0.92	-1.08	-1.06
5	PH	0.16	0.12	0.18
5	Mus	-0.99	-1.28	-1.40
10	CH	-1.45	-0.19	-1.23
10	Loan	0.19	0.81	0.32
10	Air	-1.75	-1.02	-1.93
10	Mag	-2.04	-2.38	-1.89
10	PH	0.27	-0.23	0.37
10	Mus	-1.84	-2.23	-2.13
15	CH	-0.09	1.48	-0.01
15	Loan	0.60	1.03	0.69
15	Air	-3.62	-2.86	-2.18
15	Mag	-2.89	-3.47	-2.38
15	PH	0.39	-0.28	0.39
15	Mus	-1.48	-2.02	-2.53
20	CH	2.98	4.24	1.18
20	Loan	1.36	1.32	1.17
20	Air	-4.74	-5.69	-3.75
20	Mag	-3.32	-4.20	-3.24
20	PH	1.10	-0.41	0.59
20	Mus	-0.94	-1.02	-1.25
30	CH	6.98	15.04	7.36
30	Loan	2.42	3.87	2.15
30	Air	-9.26	-13.7	-4.07
30	Mag	-4.61	-5.61	-3.44
30	PH	1.94	-4.18	1.85
30	Mus	2.35	2.76	1.06

Таблица 3: Среднее изменение качества решения базовой задачи при замене KNN на ANN на основе HNSW (%)

P%	Dataset	MAR	MNAR	MCAR
5	CH	-6.32	-4.80	-6.53
5	Loan	0.14	-0.25	0.10
5	Air	-0.47	-0.72	-0.40
5	Mag	-2.13	-1.10	-1.83
5	PH	0.18	0.31	0.55
5	Mus	2.02	2.20	3.00
10	CH	-3.13	-4.57	-6.07
10	Loan	0.33	0.14	-0.05
10	Air	-0.82	-1.15	-0.58
10	Mag	-3.22	-3.06	-3.33
10	PH	0.54	0.34	0.24
10	Mus	2.32	0.38	1.79
15	CH	-3.47	-3.58	-7.40
15	Loan	0.69	1.02	0.63
15	Air	-0.61	-0.87	-0.50
15	Mag	-4.26	-4.36	-4.06
15	PH	0.80	0.91	1.22
15	Mus	2.60	1.73	3.70
20	CH	-1.08	-6.49	-7.27
20	Loan	1.66	1.22	0.91
20	Air	-1.43	-2.28	-0.88
20	Mag	-3.66	-3.64	-3.84
20	PH	1.00	1.20	0.50
20	Mus	0.45	-0.17	0.91
30	CH	12.44	-5.57	2.42
30	Loan	3.34	2.31	3.00
30	Air	-0.34	-0.48	-0.62
30	Mag	1.04	-1.86	-1.37
30	PH	1.18	1.49	1.09
30	Mus	-0.60	0.40	-0.55

В таблице 1 приведено среднее время заполнения пропущенных значений для каждого из рассматриваемых наборов данных и каждого значения параметра p, агрегированное по всем типам механизмов пропусков. Основная цель предложенного метода заключается в ускорении традиционного процесса заполнения пропусков при помощи KNN Imputer. Существенных различий во времени заполнения между случаями MAR, MNAR и MCAR выявлено не было.

Следует также отметить, что время построения индекса является относительно небольшим по сравнению со временем поиска. В случае HNSW медианное время построения индекса составляет приблизительно 8% от времени поиска в среднем по всем наборам данных.

В таблице 1 показано ускорение, достигаемое за счёт замены точного поиска ближайших соседей на заполнение на основе ANN. В среднем заполнение пропусков с использованием HNSW оказывается в **17.9** раза быстрее по сравнению с классическим KNN Imputer при доле пропущенных значений 5%, в **23.8** раза быстрее при 10%, в **26.5** раза при 15%, в **26.5** раза при 20% и в **28.4** раза быстрее при 30% пропущенных значений.

В таблицах 2 и 3 представлены результаты, характеризующие влияние замены классического KNN на ANN-подход на основе HNSW на качество заполнения пропущенных значений и на качество решения базовой задачи. Таблица 2 отражает средний относительный прирост или потерю качества заполнения пропущенных значений на уровне отдельных наблюдений для различных механизмов пропусков (MAR, MNAR и MCAR), а таблица 3 показывает соответствующие изменения показателей качества решения базовых задач. Ускорение процедуры заполнения пропущенных значений достигается ценой неоднозначных изменений качества: для одних наборов данных и сценариев пропусков наблюдаются умеренные потери, тогда как в других случаях фиксируется сопоставимый или даже положительный эффект, особенно при высоких долях пропусков. Таким образом, результаты, представленные в таблицах 2 и 3, важны в контексте интерпретации полученного ускорения, поскольку они позволяют количественно оценить, к каким именно изменениям качества приводит использование ANN-методов. Для удобства анализа, результаты, представленные в этих таблицах также отображены на графиках 1, 2 и 3 (с усреднением по всем p). Хорошо видно, что методы измерения качества заполнения пропусков рассогласованы - в ряде случаев знаки изменения качества разнонаправлены. Наиболее важным критерием является изменение в оценке качества решения базовой задачи.

В целом наблюдается тенденция к увеличению относительных различий между алгоритмами по мере роста доли пропущенных данных (p). В частности (по таблице 2):

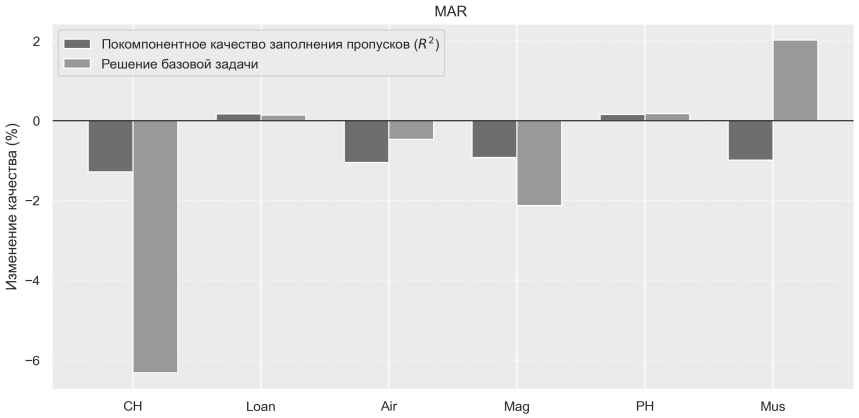


Рис. 1: Изменение качества заполнения пропущенных значений и решения соответствующей базовой задачи при переходе от KNN к ANN (MAR)

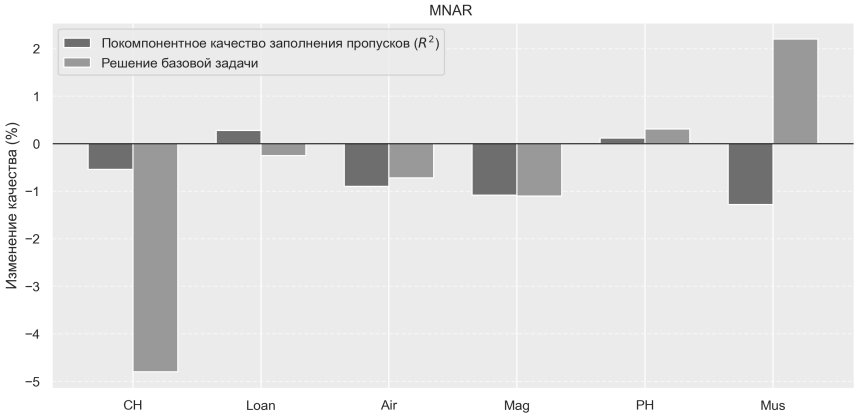


Рис. 2: Изменение качества заполнения пропущенных значений и решения соответствующей базовой задачи при переходе от KNN к ANN (MNAR)

- при $p = 5\%$ отклонение варьируется от -1.28% до $+0.29\%$;
- при $p = 10\%$ — от -1.89% до $+0.84\%$;
- при $p = 15\%$ — от -3.77% до $+1.48\%$;
- при $p = 20\%$ — от -5.69% до $+4.24\%$;
- при $p = 30\%$ — от -13.7% до $+15.04\%$.

С точки зрения изменения качества решения базовых задач, наибольшая зафиксированная потеря качества составляет -7.4%, тогда как максимальный прирост достигает $+12.44\%$. Эти эффекты существенно

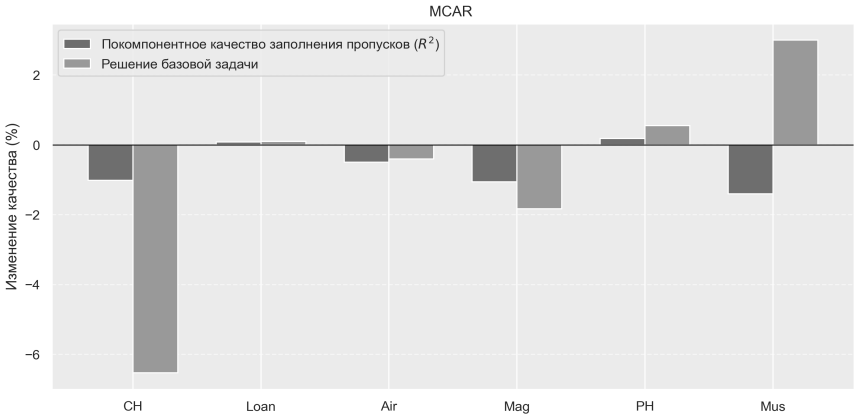


Рис. 3: Изменение качества заполнения пропущенных значений и решения соответствующей базовой задачи при переходе от KNN к ANN (MCAR)

зависят как от конкретного набора данных, так и от доли пропущенных значений. Так, для наборов данных Loan Classification, Mushrooms и Paris Housing в большинстве случаев наблюдаются положительные изменения показателей, тогда как для набора California Housing преимущественно фиксируются отрицательные изменения значения метрики F_1. В прикладных задачах допустимость различий в качестве порядка 5–6% может существенно варьироваться в зависимости от контекста применения: в одних случаях такие отклонения являются критичными, тогда как в других приоритет отдается вычислительной эффективности. Если основным ограничением является время обработки, ANN-подход к заполнению пропущенных значений, в частности метод на основе HNSW, позволяет добиться значительного ускорения. Однако для задач, чувствительных даже к небольшим изменениям качества данных, целесообразно проводить дополнительную эмпирическую проверку перед заменой классических KNN-методов на ANN-решения.

6. Заключение

Эксперименты показали, что методы на основе HNSW позволяют существенно ускорить процесс заполнения пропущенных значений на основании алгоритма KNN IMputer, достигая прироста скорости до 28× при сохранении сопоставимого уровня точности практически во всех случаях. Вопреки ожиданиям, использование ANN-подхода не всегда сопровождается снижением качества заполнения пропущенных значений: изменения

в показателях могут быть как положительными, так и отрицательными и в значительной степени зависят от конкретного набора данных. Это указывает на то, что отдельные наборы данных более или менее подходят для применения методов на основе ANN, независимо от доли и типа пропусков. В то же время ключевые особенности данных, определяющие оптимальный подход к заполнению пропусков, пока остаются неясными и представляют собой перспективное направление для дальнейших исследований.

Кроме того, мы отметили, что с ростом доли пропущенных значений в данных увеличивается относительная вариативность разницы в качестве между ANN- и KNN-методами. Эта разница может быть как положительной, так и отрицательной. При этом тип пропусков, по-видимому, не оказывает значимого влияния на такую зависимость. Несмотря на возможные риски, предложенный ANN-подход хорошо подходит для задач, требующих быстрого заполнения пропущенных значений в больших наборах данных при ограничениях по времени. В случаях, когда критична максимальная точность модели, рекомендуется провести дополнительную валидацию или использовать классический KNN-подход.

Список литературы

[1] Acock, Alan, “What to do about missing Values”, *APA Handbook of Research Methods in Psychology*, 2012, 27–50.

[2] Chen, Tianqi and Guestrin, Carlos, “XGBoost: A Scalable Tree Boosting System”, 2016, 785–794. DOI: 10.1145/2939672.2939785.

[3] Ergul Aydin, Zeliha and Kamisli Ozturk, Zehra, “Performance Analysis of XGBoost Classifier with Missing Data”, 2021.

[4] Troyanskaya, Olga and Cantor, Mike and Hastie, Trevor and Tibshirani, Rob and Botstein, David and Altman, Russ, “Missing Value Estimation Methods for DNA Microarrays”, *Bioinformatics*, **17** (2001), 520–525. DOI: 10.1093/bioinformatics/17.6.520.

[5] Murti, Della and Pujianto, Utomo and Wibawa, Aji and Akbar, Muhammad, “K-Nearest Neighbor (K-NN) based filling of missing values”, 2019, 83–88. DOI: 10.1109/ICSITech46713.2019.8987530.

[6] Acuna, Edgar and Rodriguez, Caroline, “The Treatment of Missing Values and its Effect on Classifier Accuracy”, *Journal of Classification*, 2004, 639–647. DOI: 10.1007/978-3-642-17103-1_60.

[7] Liu, Ting and Moore, Andrew and Gray, Alexander and Yang, Ke, “An Investigation of Practical Approximate Nearest Neighbor Algorithms”, 2004.

[8] Arya, Sunil and Mount, David M. and Netanyahu, Nathan S. and Silverman, Ruth and Wu, Angela Y., “An optimal algorithm for approximate nearest neighbor searching fixed dimensions”, *J. ACM*, **45** (1998), 891–923.

[9] Andoni, Alexandr and Indyk, Piotr and Razenshteyn, Ilya, "Approximate Nearest Neighbor Search in High Dimensions", 2019, 3287–3318. DOI: 10.1142/9789813272880_0182.

[10] Aumüller, Martin and Bernhardsson, Erik and Faithfull, Alexander John, "ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms", *Inf. Syst.*, **87** (2018).

[11] Datta, Ritendra and Joshi, Dhiraj and Li, Jia and Wang, James Ze, "Image retrieval: Ideas, influences, and trends of the new age", *ACM Comput. Surv.*, **40** (2008), 5:1–5:60.

[12] Hays, James and Efros, Alexei A., "Scene Completion Using Millions of Photographs", *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2007.

[13] Şakar, Tolga and Emekci, Hakan, "Maximizing RAG efficiency: A comparative analysis of RAG methods", *Natural Language Processing*, **31** (2024), 1–25. DOI: 10.1017/nlp.2024.53.

[14] Chen, Wei and Chen, Jincai and Zou, Fuhao and Li, Yuan-Fang and Lu, Ping and Zhao, Wei, *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, 2019.

[15] Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E., "Scikit-learn: Machine Learning in Python", *Journal of Machine Learning Research*, **12** (2011), 2825–2830.

[16] Pace, R. Kelley and Barry, Ronald P., "Sparse spatial autoregressions", *Statistics & Probability Letters*, **33** (1997), 291–297.

[17] Schafer, Joseph and Graham, John, "Missing Data: Our View of the State of the Art", *Psychological Methods*, **7** (2002), 147–177. DOI: 10.1037/1082-989X.7.2.147.

[18] Du, Jinghan and Hu, Minghua and Zhang, Weining, "Missing Data Problem in the Monitoring System: A Review", *IEEE Sensors Journal*, **20**:23 (2020), 13984–13998. DOI: 10.1109/JSEN.2020.3009265.

[19] Malkov, Yury and Yashunin, Dmitry A., "Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **42** (2016), 824–836.

[20] Lietz, Haiko, "Watts, Duncan J./Strogatz, Steven H. (1998). Collective Dynamics of » Small-World « Networks. Nature 393, S. 440–442.", 2019, 551–553. DOI: 10.1007/978-3-658-21742-6_130.

[21] Sharma, Rohit, "Loan Approval Dataset", 2024. DOI: 10.34740/KAGGLE/DSV/8173310.

[22] Mateen, Mujtaba, "Air Quality and Pollution Assessment", 2024. DOI: 10.34740/KAGGLE/DS/6197184.

[23] Sawhney, Prisha, "Mushroom Dataset", 2024.

[24] "Paris Housing Price Prediction Dataset", 2024.

[25] Bock, R., "MAGIC Gamma Telescope", 2004. DOI: 10.24432/C52C8B.

[26] Douze, Matthijs and Guzhva, Alexandr and Deng, Chengqi and Johnson, Jeff and Szilvasy, Gergely and Mazaré, Pierre-Emmanuel and Lomeli, Maria and

Hosseini, Lucas and Jégou, Hervé, "The faiss library", *arXiv preprint*, **240** (2024).

[27] Skovgaard, Lene, "Applied regression analysis. 3rd edn. N. R. Draper and H. Smith, Wiley, New York, 1998. No. of pages: xvii+706. Price: £45. ISBN 0-471-17082-8", *Statistics in Medicine - STAT MED*, **19** (2000), 3136–3139.

[28] Sasaki, Yutaka, "The truth of the F-measure", *Teach Tutor Mater*, 2007.

[29] Shadbahr, Tolou and Roberts, Michael and Stanczuk, Jan and Gilbey, Julian and Teare, Philip and Dittmer, Sören and Thorpe, Matthew and Viñas Torné, Ramon and Sala, Evis and Lio, Pietro and Patel, Mishal and Preller, Jacobus and Rudd, James and Mirtti, Tuomas and Rannikko, Antti and Aston, John and Tang, Jing and Schönlieb, Carola-Bibiane, "The impact of imputation quality on machine learning classifiers for datasets with missing values", *Communications medicine*, **3** (2023), 139. DOI: 10.1038/s43856-023-00356-z.

[30] Jadhav, Anil and Pramod, Dhanya and Ramanathan, Krishnan, "Comparison of Performance of Data Imputation Methods for Numeric Dataset", *Applied Artificial Intelligence*, **33** (2019), 1–21. DOI: 10.1080/08839514.2019.1637138.

[31] Thurow, Maria and Dumpert, Florian and Ramosaj, Burim and Pauly, Markus, "Imputing missings in official statistics for general tasks – our vote for distributional accuracy", *Statistical Journal of the IAOS*, **37** (2021), 1–12. DOI: 10.3233/SJI-210798.

[32] Rivolli, Adriano and Garcia, Luís P.F. and Soares, Carlos and Vanschoren, Joaquin and de Carvalho, André C.P.L.F., "Meta-features for meta-learning", *Knowledge-Based Systems*, **240** (2022), 108101. DOI: 10.1016/j.knosys.2021.108101.

[33] Akiba, Takuya and Sano, Shotaro and Yanase, Toshihiko and Ohta, Takeru and Koyama, Masanori, "Optuna: A next-generation hyperparameter optimization framework", 2019, 2623–2631.

[34] Altmann, Andre and Tolosi, Laura and Sander, Oliver and Lengauer, Thomas, "Permutation importance: A corrected feature importance measure", *Bioinformatics (Oxford, England)*, **26** (2010), 1340–1347. DOI: 10.1093/bioinformatics/btq134.

[35] Beale, E. M. L. and Little, R. J. A., "Missing Values in Multivariate Analysis", *Journal of the Royal Statistical Society: Series B (Methodological)*, **37** (1975), 129–145. DOI: 10.1111/j.2517-6161.1975.tb01037.x.

[36] Yoon, Jinsung and Jordon, James and Schaar, Mihaela, "Gain: Missing data imputation using generative adversarial nets", 2018, 5689–5698.

[37] Ma, Zhihua and Chen, Guanghui, "Bayesian methods for dealing with missing data problems", *Journal of the Korean Statistical Society*, **47** (2018), 297–313.

[38] Zhang, Zhongheng, "Multiple imputation with multivariate imputation by chained equation (MICE) package", *Annals of translational medicine*, **4**:2 (2016), 30.

[39] Stekhoven, Daniel J and Bühlmann, Peter, "MissForest—non-parametric filling of missing values for mixed-type data", *Bioinformatics*, **28**:1 (2012), 112–118.

[40] Gondara, Lovedeep and Wang, Ke, "Mida: Multiple imputation using denoising autoencoders", 2018, 260–272.

Статья поступила 21 марта 2026 г.

Hierarchical Nearest Neighbor Search for Missing Value Imputation in Tabular Data

S. A. Artamonov, A. M. Mironov

Missing data represent one of the most frequent difficulties encountered in data science. A wide range of effective techniques for dealing with missing values has been developed to date. One commonly adopted approach relies on nearest neighbor methods, which perform imputation by leveraging information from the most similar observations. Such techniques are particularly prevalent in big data analysis, where datasets can comprise billions of instances. Under these conditions, improving the efficiency of k-Nearest Neighbors (KNN) algorithms becomes crucial, as their computational cost can be substantial. In this study, we investigate approaches for accelerating missing value imputation based on KNN. We introduce a novel algorithm that employs approximate nearest neighbor (ANN) techniques and examine the relationship between computational efficiency and imputation accuracy. The experimental results demonstrate that widely used ANN algorithms are capable of significantly reducing runtime while preserving a high level of imputation quality.

Keywords: missing values filling, approximate nearest neighbors.

Received on March 21, 2026

Алгоритм определения типов восприятия трудных жизненных ситуаций на основе краткой версии опросника «Типы ориентаций в трудной ситуации» ТОРС-36

И. О. Бергер*, А. А. Хлебникова‡

Статья посвящена валидации краткой версии опросника «Типы ориентаций в трудных ситуациях» (ТОРС-36) в качестве инструмента классификации типов восприятия трудных жизненных ситуаций и оценке роли анализа текстовых описаний с привлечением больших языковых моделей в определении этих типов. Решаются следующие задачи: разработка и валидация алгоритма классификации типов восприятия трудных ситуаций на основе профиля ТОРС-36 с привлечением больших языковых моделей для анализа текстовых описаний; сопоставление результатов алгоритмической классификации с экспертной оценкой; сравнение эффективности различных модификаций алгоритма. В выборку исследования вошли 200 текстовых описаний трудных ситуаций, полученных от 200 респондентов (131 женщина, 69 мужчин) в возрасте от 14 до 54 лет. Эталонная классификация выполнялась экспертами-психологами на основе совместного анализа количественных показателей шкал опросника ТОРС-36 и качественных данных (текстовых описаний трудных ситуаций). Проведена серия из шести экспериментов. В экспериментах 1–4 последовательно возрастала роль количественных данных и формальных правил классификации. В эксперименте 5 оценивалась точность классификации при разметке качественных данных с помощью большой языковой модели, в эксперименте 6 – при разметке, выполненной экспертами-психологами. Наиболее эффективным среди алгоритмов, использующих большую языковую модель, оказался гибридный, в котором пороговые правила применяются к количественным показателям, а большая языковая модель анализирует

* *Бергер Ирина Олеговна* — м.н.с. каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: ioberger@ya.ru, ORCID: 0000-0003-0123-1014.

Berger Irina Olegovna — Junior Researcher, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intellectual Systems.

‡ *Хлебникова Алёна Андреевна* — аспирант каф. общей психологии ф. психологии, МГУ имени М.В. Ломоносова, e-mail: alena.epochta@gmail.com, ORCID: 0000-0002-7427-8844.

Khlebnikova Alena Andreevna — PhD Student, Department of General Psychology, Faculty of Psychology, Lomonosov Moscow State University.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

качественные данные. Его точность составила 85% на полной выборке. При этом максимальная точность классификации в серии экспериментов – 85,5% – достигнута при использовании того же алгоритма с экспертной разметкой качественных данных. Полученные результаты подтверждают возможность использования опросника ТОРС-36 как инструмента классификации типов восприятия трудных ситуаций и позволяют рассматривать большие языковые модели как средство поддержки эксперта при классификации типов восприятия трудных ситуаций.

Ключевые слова: копинг (совладание), трудная жизненная ситуация, восприятие ситуации, ТОРС-36, алгоритм определения типов восприятия трудных ситуаций, большие языковые модели, смешанный дизайн исследования.

1. Введение

Изучение механизмов восприятия жизненных трудностей остается одной из центральных задач психологии копинга. То, как человек воспринимает и интерпретирует трудную ситуацию, определяет не только выбор им конкретных копинг-стратегий, но и общую эффективность совладания. Однако в большинстве эмпирических исследований фокус внимания смещен либо на объективные параметры стрессогенных воздействий, либо на устойчивые личностные диспозиции, предопределяющие копинг, как показывает Е.В. Битюцкая [1]. При этом целостный образ трудной ситуации, конституирующий субъективное переживание, редко становится предметом самостоятельного анализа.

С точки зрения методологии, во многом это связано с доминированием количественных методов исследования, в частности, стандартизированных опросников [1]. При всех достоинствах опросных методик – стандартизации, экономичности, возможности количественного анализа данных – их использование при изучении восприятия трудных жизненных ситуаций (далее ТЖС) неизбежно сводит многоуровневую смысловую реальность переживания к ограниченному набору шкал, заданных исследователем. Альтернативу представляют качественные методы, ориентированные на понимание смысловой структуры и субъективной реальности человека, и позволяющие понять, как человек организует свое восприятие. Методология смешанного дизайна, объединяющая качественные и количественные методы исследования, позволяет компенсировать ограничения каждого из подходов [10] и в исследованиях копинга значительно обогащает понимание того, как человек взаимодействует с ТЖС.

Количественным инструментом в наших исследованиях копинга выступает опросник «Типы ориентаций в трудных ситуациях» (ТОРС),

описанный в статье Е.В. Битюцкой и А.А. Корнеева [4] и его краткая версия TOPC-36, представленная в статье Е.В. Битюцкой, А.А. Корнеева, А.А. Хлебниковой [6]. Методики опираются на концептуальную модель ориентаций в трудных ситуациях [3], где ориентация понимается как единство когнитивных, эмоциональных и мотивационных компонентов, задающее готовность к определенному типу совладающих усилий. Полная версия TOPC содержит 65 пунктов и 8 шкал, соответствующих ориентациям; шкалы сгруппированы в два типа: отражающие приближение к трудности и отражающие уход от нее [4].

Для масштабных исследований и включения в тестовые батареи была разработана сокращенная версия TOPC-36, включающая 36 пунктов и 7 шкал [6]. Шкалы TOPC-36 распределяются по трем группам: (1) шкалы ориентаций приближения, (2) шкалы ориентаций ухода и (3) отдельная шкала «Упреждение проблемы». Последняя описывает активность субъекта, направленную на упреждение трудности и положительно коррелирует со шкалами ориентаций приближения, но не обнаруживает значимых связей со шкалами ориентаций ухода.

К ориентациям приближения относятся «Стремление к трудности», «Усложнение задачи» и «Достижение результата». К ориентациям ухода относятся «Избегание трудности», «Облегчение задачи» и «Отстранение от проблемы».

Качественные данные собираются с помощью «Методики структурированного описания ситуации» [2], предполагающей ответы респондентов на открытые вопросы. Совместный анализ выраженности шкал TOPC-36 и качественных данных позволяет диагностировать пять типов восприятия ТЖС, описанных в работе Е.В. Битюцкой, Э.Э. Гасанова, Н.А. Патрашкина, К.В. Хазовой [5]: драйвовый, максимальный, оптимальный, амбивалентный и отстраненный.

Драйвовый тип восприятия ТЖС характеризуется переживанием прилива сил, положительными эмоциями и оптимистичным прогнозом, уверенностью в себе и готовностью к активным действиям.

Максимальный тип восприятия ТЖС характеризуется склонностью к многозадачности и ориентацией на перфекционистски высокие цели ценой значительных усилий.

Оптимальный тип восприятия ТЖС предполагает достижение трудной цели усилиями, соразмерными требованиям задачи, с опорой на социальные ресурсы.

Амбивалентный тип восприятия ТЖС характеризуется внутренним конфликтом между приближением к трудности и уходом от нее, проявляющийся в неустойчивости мотивации, противоречивых эмоциях и колебаниях активности.

Отстраненный тип восприятия ТЖС характеризуется доминированием пассивно-избегающей позиции, уходом от решения трудной жизненной задачи [5].

Практическое применение процедуры диагностики типа восприятия ТЖС ограничено трудоемкостью ручной классификации [7], зависимостью от квалификации экспертов и субъективностью интерпретации качественных данных. Эти ограничения затрудняют использование типологии в масштабных исследованиях и прикладных задачах, требующих быстрой, воспроизводимой и стандартизированной классификации. В связи с этим возникает необходимость разработки формализованного алгоритма, который мог бы автоматически классифицировать типы восприятия ТЖС, моделируя процедуру экспертной оценки. Ранее была установлена эффективность современных больших языковых моделей (LLM) при решении задач анализа качественных данных [8], в частности, их способность выявлять смысловые категории и индикаторы, сопоставимые с экспертной интерпретацией текстовых данных [7].

Цель настоящего исследования — валидация опросника ТОРС-36 в качестве инструмента классификации типов восприятия ТЖС и оценка роли анализа текстовых описаний трудных ситуаций с привлечением LLM в определении этих типов. Задачами исследования стали:

– разработка и валидация алгоритма классификации типов восприятия ТЖС на основе профиля опросника ТОРС-36 с привлечением LLM для анализа текстовых описаний трудных ситуаций;

– сопоставление результатов алгоритмической классификации на основе профилей опросника ТОРС-36 с экспертной оценкой;

– сравнение эффективности различных модификаций алгоритма.

2. Критерии экспертной оценки

В основу эталонной классификации, использованной для разработки и валидации алгоритма определения типов восприятия ТЖС, положена процедура экспертной оценки. Два эксперта-психолога, обученные правилам классификации типов восприятия ТЖС и имеющие опыт качественного анализа данных, независимо друг от друга определяли тип восприятия ТЖС на основе совместного анализа двух источников данных: профиля респондента по опроснику ТОРС-36 [6] и его ответов на открытые вопросы «Методики структурированного описания ситуации» [2], и присваивали текстовому описанию ТЖС один из пяти типов восприятия ТЖС. При совпадении оценок итоговый тип восприятия ТЖС фиксировался без дополнительного обсуждения; при несовпадении проводилось обсужде-

ние до достижения консенсуса. Качественные данные анализировались с опорой на кодировочную инструкцию, разработанную Е.В. Битюцкой и Н.Г. Малышевой, и принципы разметки, предложенные Э.Э. Гасановым и Е.В. Битюцкой [5]. Каждый источник данных вносил свой вклад в итоговое заключение о типе восприятия ТЖС: если профиль респондента по опроснику ТОРС-36 и анализ качественных данных указывали на один и тот же тип, решение считалось однозначным; при расхождении предпочтение отдавалось качественным данным, поскольку они отражали смысловую структуру переживания. Исключение составляли чрезмерно краткие описания (в общей выборке настоящего исследования – не более 1,5% случаев), не позволявшие сделать надежное суждение: в этих ситуациях решение принималось на основе профиля по опроснику ТОРС-36. Отдельно фиксировались случаи, когда и профиль, и качественные признаки с равной степенью уверенности указывали на разные типы; при этом итоговое решение также принималось с опорой на качественные данные при условии их достаточной информативности.

Оценка профиля по опроснику ТОРС-36 проводилась путем сопоставления баллов по каждой шкале с пороговым значением 1,5 балла (из трех возможных): баллы ниже порога интерпретировались как незначимая представленность ориентации, баллы, равные порогу или превышающие его, – как актуальное присутствие ориентации в профиле.

При формировании эталонной классификации эксперты-психологи опирались на следующие формальные критерии, которые уточняют содержательные характеристики типов и были разработаны на основе результатов предшествующих исследований [5]:

– драйвовый тип: в профиле респондента по опроснику ТОРС-36 наиболее выражена шкала «Стремление к трудности» при выраженности других шкал ориентаций приближения и отсутствии выраженности шкал ориентаций ухода; в качественных данных этот тип проявляется через описания положительных оценок и эмоций, стремления к саморазвитию, повышения энергии и ориентации на достижение высоких результатов;

– максимальный тип: в профиле респондента по опроснику ТОРС-36 выражена шкала «Усложнение задачи», которая может сочетаться с другими выраженными шкалами ориентаций приближения при отсутствии выраженности шкал ориентаций ухода; в качественных данных наиболее значимыми признаками выступают указания на высокие достижения и многозадачность;

– оптимальный тип: в профиле по опроснику ТОРС-36 выражены шкалы ориентаций приближения, среди которых доминирует по выраженности шкала «Достижение результата», при отсутствии выраженности шкал ориентаций ухода; в качественных описаниях респондентов часто

упоминаются планомерный копинг, положительная переоценка ситуации, цели приближения и нейтральная оценка трудности;

– амбивалентный тип: характеризуется одновременной выраженностью одной, двух или трех шкал ориентаций приближения в сочетании с одной, двумя или тремя шкалами ориентаций ухода в профиле респондента по опроснику TOPC-36; в качественных описаниях часто упоминаются отрицательные эмоциональные состояния, требующие затрат сил и времени на их преодоление, при этом описываются цели двух типов: достижение привлекательного результата и избавление от неприятного;

– отстраненный тип: в профиле респондента по опроснику TOPC-36 отмечается выраженность одной, двух или трех шкал ориентаций ухода при отсутствии выраженности шкал ориентаций приближения; в качественных данных представлены описания интенсивных отрицательных эмоций переживания безысходности, использования избегающих копинг-стратегий и постановки целей, связанных с избеганием трудности.

В случаях, когда в профиле выражены шкалы «Стремление к трудности» и «Усложнение задачи» и не выражены шкалы ориентаций ухода (что характерно для драйвового и максимального типов), качественные данные дополнительно анализировались на наличие индикаторов категории угрозы жизни или здоровью (самого респондента, его близких или питомцев). Если такие индикаторы присутствовали, тип определялся как оптимальный, поскольку ранее было замечено, что оптимальный тип чаще встречается в ситуациях, связанных со здоровьем, тогда как драйвовый и максимальный – в контексте задач, связанных с достижениями целей.

Таким образом, эталонная классификация выборки была выполнена по единым правилам, что позволило в дальнейшем корректно оценивать точность алгоритмических решений.

3. Исходные данные

3.1. Полная выборка

Выборку исследования составили 200 текстовых описаний трудных ситуаций, полученных от 200 респондентов (131 женщина и 69 мужчин) в возрасте от 14 до 54 лет (M = 23,46; SD = 7,70). Возраст женщин варьировался от 14 до 54 лет (M = 23,29; SD = 7,85), мужчин — от 18 до 54 лет (M = 23,77; SD = 7,44). Выборка формировалась двумя способами: основной массив составили студенты московских вузов, приглашённые авторами в устной форме на занятиях своих учебных групп (участникам разъяснялась добровольность участия в исследовании и гарантиро-

валось отсутствие академических последствий отказа); дополнительно применялась стратегия снежного кома — студентов-участников просили порекомендовать исследование знакомым, не входящим в студенческую выборку, что позволило привлечь добровольцев старшего возраста. Сбор данных осуществлялся в 2025 году. На основании экспертной оценки текстовые описания ТЖС были распределены по пяти типам восприятия ТЖС: драйвовый (n = 17), максимальный (n = 14), оптимальный (n = 34), амбивалентный (n = 125) и отстраненный (n = 10).

3.2. Выборка пилотажного исследования

Выборку пилотажного исследования составили 100 текстовых описаний трудных ситуаций, полученных от 100 респондентов из полной выборки (67 женщин, 33 мужчины) в возрасте от 18 до 47 лет (M = 21,59; SD = 6,15). Возраст женщин варьировал от 18 до 44 лет (M = 21,01; SD = 5,40), мужчин – от 18 до 47 лет (M = 22,76; SD = 7,40). На основании экспертной оценки текстовые описания ТЖС были распределены по пяти типам восприятия ТЖС: драйвовый (n = 10), максимальный (n = 9), оптимальный (n = 16), амбивалентный (n = 59) и отстраненный (n = 6).

3.3. Условные обозначения

Тип, код – буквенно-цифровой код, представляющий номер класса, характеризующего тип восприятия ТЖС респондента на основании экспертной оценки, идентификатор случая и инициалы эксперта-психолога.

Тип – показатель, характеризующий тип восприятия ТЖС респондента на основании экспертной оценки: 1 – драйвовый, 2 – максимальный, 3 – оптимальный, 4 – амбивалентный, 5 – отстраненный.

Ф6 – показатель, фиксирующий наличие или отсутствие в качественных данных указаний на связь ТЖС с угрозой жизни или здоровью самого респондента, его близких либо питомцев. Принимает значение 1 в случае наличия такой связи, 0 в противном случае.

Ситуация – описание ТЖС, данное респондентом, текстовое поле.

Стремление к трудности – показатель шкалы «Стремление к трудности» опросника TOPC-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

Усложнение задачи – показатель шкалы «Усложнение задачи» опросника TOPC-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

Достижение результата – показатель шкалы «Достижение результата» опросника ТОРС-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

Избегание трудности – показатель шкалы «Избегание трудности» опросника ТОРС-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

Облегчение задачи – показатель шкалы «Облегчение задачи» опросника ТОРС-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

Отстранение от проблемы – показатель шкалы «Отстранение от проблемы» опросника ТОРС-36 [6]. Принимает действительные значения в диапазоне от 0 до 3.

A1 («Позитивные интенсивные эмоции», весовой коэффициент 3), **A2** («Позитивные неинтенсивные эмоции», 3), **E3** («Высокий уровень энергии», 3), **E4** («Энергия растет», 3), **1A2** («Положительная оценка ситуации», 3), **1A5** («Амбивалентная оценка», 1), **1B4** («Масштабность», 3), **1B7** («Вызов», 3), **1K1** («Контроль над ситуацией», 2), **1D6** («Положительная переоценка», 2), **2A5** («Развитие, увеличение», 3), **3A2** («Отмечается, что возможностей много», 2), **3A8** («Саморазвитие как возможность», 3), **3B2** («Отсутствие ограничений», 2), **5A3** («Невозможность неуспеха», 2), **6B5** («Фантазийный успех», 1) – ключевые признаки драйвового типа, представленные в качественных данных. Для определения данного типа эксперту-психологу достаточно зафиксировать наличие хотя бы нескольких из признаков. Весовые коэффициенты были определены в результате контент-анализа качественных описаний ТЖС, выполненного экспертами-психологами на основе кодировочной инструкции, разработанной Е.В. Битюцкой и Н.Г. Малышевой. Показатель каждого признака принимает значение 1 при наличии признака в качественных данных и 0 при его отсутствии. Описание признаков представлено в статье Е.В. Битюцкой, Э.Э. Гасанова, К.В. Хазовой, Н.А. Патрашкина [9].

Б1 («Время упоминается», 2), **Б4** («Временная, проходящая ситуация», 1), **E2** («Низкий уровень энергии», 1), **E6** («Необходимость затрат энергии», 1), **C4** («Суметь сделать все», 3), **C5** («Нужно успеть», 3), **C6** («Нужно достичь максимального результата», 1), **Ф4.6** («Распределение времени», 3), **1B9** («Необходимость», 3), **1D5** («Борьба», 1), **1D8** («Подбадривать себя», 1), **2A4** («Сохранение имеющегося», 1), **3B4** («Ограничения графика работы / учебы», 1), **3B5** («Множество дел и невозможность отказа от множественной задачи», 2), **5B4** («Перенос результата на более поздний срок», 1), **6B8** («Содержание успеха — завершить», 1) – ключевые признаки максимального типа, весовые коэффициенты признаков

определены тем же образом, что и для признаков драйвового типа. Показатель каждого признака принимает значение 1 при наличии признака в качественных данных и 0 при его отсутствии. Описание признаков представлено в статье Е.В. Битюцкой, Э.Э. Гасанова, К.В. Хазовой, Н.А. Патрашкина [9].

4. Описание алгоритма определения типов восприятия ТЖС

В основу разработанного алгоритма определения типов восприятия ТЖС положено решение, ранее примененное для валидации полной версии опросника ТОРС Е.В. Битюцкой, Э.Э. Гасановым, Н.А. Патрашкиным, К.В. Хазовой [5]. Данный подход был адаптирован для краткой версии ТОРС-36. Алгоритм включает три последовательных этапа, на каждом из которых производилось поэтапное отнесение ответов респондентов к одному из пяти типов восприятия ТЖС. Процедура применялась к каждой строке данных (отдельному текстовому описанию ТЖС) независимо.

На вход алгоритм получал таблицу данных `data`, содержащую баллы по шкалам опросника ТОРС-36 и результаты контент-анализа качественных описаний ТЖС, выполненного экспертами-психологами на основе кодировочной инструкции [5]. Результатом работы являлся целочисленный массив `answer` размерности, равной количеству строк исходной таблицы. Его элементы соответствовали номерам типов: 1 – драйвовый, 2 – максимальный, 3 – оптимальный, 4 – амбивалентный, 5 – отстраненный; значение 0 использовалось для случаев, которые не удалось классифицировать.

На первом этапе выделялись записи, относившиеся к амбивалентному (4) и отстраненному (5) типам. Соответствующим элементам массива `answer` присваивались значения 4 и 5. Записи, не отнесенные к этим типам, получали временную метку 123, означающую необходимость дальнейшей обработки. Записи, не удовлетворявшие ни одному из критериев, получали значение 0.

На втором этапе для записей с меткой 123 производилась дифференциация оптимального типа (3): если условия выполнены, элементу массива присваивалось значение 3, в противном случае метка заменялась на 12, и указывала на принадлежность к драйвовому (1) или максимальному (2) типам. После этого массив `answer` содержал значения из множества {12, 3, 4, 5, 0}.

На третьем этапе для записей с меткой 12 выполнялась дифференциация драйвового (1) и максимального (2) типов. В результате итоговый массив `answer` содержал значения из множества $\{0, 1, 2, 3, 4, 5\}$.

4.1. Дифференциация амбивалентного и отстраненного типов восприятия ТЖС от типов восприятия ТЖС, ориентированных на приближение

Первый шаг алгоритма обозначим через A_1. В нем значения ориентаций, вычисленных на основе ТОРС-36, подставлялись в логические выражения, и на основе результата вычислений формировался ответ.

Определение. Ориентация называется *выраженной*, если ее значение $r_{orientation_name}$ больше или равно пороговому значению $l_{orientation_name}$.

Выраженностью ориентации назовем функцию

$$expr_{orientation_name}(x) = \begin{cases} \text{Истина}, & \text{если } x \geqslant l_{orientation_name}, \\ \text{Ложь}, & \text{если } x < l_{orientation_name}. \end{cases}$$

Пусть даны значения $r_{\text{стр к трудн}}$, $r_{\text{услож зад}}$, $r_{\text{дост рез}}$, $r_{\text{избег трудн}}$, $r_{\text{облегч зад}}$, $r_{\text{отстр от пробл}}$. Они соответствуют столбцам «Стремление к трудности», «Усложнение задачи», «Достижение результата», «Избегание трудности», «Облегчение задачи», «Отстранение от проблемы».

Обозначим через P логическое выражение

$$\begin{aligned} P(x_1, x_2, x_3) = \\ = \big(expr_{\text{стр к трудн}}(x_1) = \text{Истина}\big) \vee \\ \vee \big(expr_{\text{услож зад}}(x_2) = \text{Истина}\big) \vee \\ \vee \big(expr_{\text{дост рез}}(x_3) = \text{Истина}\big), \end{aligned} \tag{1}$$

И через Q логическое выражение

$$\begin{aligned} Q(x_1, x_2, x_3) = \\ = \big(expr_{\text{избег трудн}}(x_1) = \text{Истина}\big) \vee \\ \vee \big(expr_{\text{облегч зад}}(x_2) = \text{Истина}\big) \vee \\ \vee \big(expr_{\text{отстр от пробл}}(x_3) = \text{Истина}\big). \end{aligned} \tag{2}$$

$$A_1(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}, r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}}) =$$

$$= \begin{cases} 123, & \text{если } P(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}) \wedge \\ & \wedge \neg Q(r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}}), \\ 4, & \text{если } P(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}) \wedge \\ & \wedge Q(r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}}), \\ 5, & \text{если } \neg P(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}) \wedge \\ & \wedge Q(r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}}), \\ 0, & \text{иначе,} \end{cases} \tag{3}$$

где $A_1(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}, r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}})$ — ответ первого шага алгоритма.

Определим пороговые значения:

- $l_{\text{стр к трудн}} = l_{\text{услож зад}} = l_{\text{дост рез}} = l_{\text{избег трудн}} = l_{\text{отстр от пробл}} = 1.5$,
- $l_{\text{облегч зад}} = 1.75$.

Данные значения получены следующим образом. Готовая выборка ответов респондентов была размечена экспертами-психологами, определен тип восприятия ТЖС для каждого текстового описания трудной ситуации, после чего запущен алгоритм перебора пороговых значений. Были зафиксированы пороговые значения, при которых алгоритм показал наивысший процент ответов, совпадающих с экспертной оценкой, а именно 92.91%. Наблюдалось повышение точности по сравнению с результатами, полученными при использовании полной версии опросника ТОРС (88, 5%).

4.2. Дифференциация оптимального типа восприятия ТЖС от максимального и драйвового типов

Обозначим через S логическое выражение

$$S(x_1, x_2, x_3) = \big(x_1 \geqslant x_3\big) \vee \big(x_2 \geqslant x_3\big) \tag{4}$$

Пусть даны значения $r_{\text{стр к трудн}}$, $r_{\text{услож зад}}$, $r_{\text{дост рез}}$, $r_{\text{избег трудн}}$, $r_{\text{облегч зад}}$, $r_{\text{отстр от пробл}}$ такие, что

$$A_1(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}, r_{\text{избег трудн}}, r_{\text{облегч зад}}, r_{\text{отстр от пробл}}) = 123.$$

Тогда:

$$A_2(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}, \Phi_6) = \\ = \begin{cases} 12, & \text{если } S(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}) \wedge \neg\Phi_6, \\ 3, & \text{иначе,} \end{cases} \tag{5}$$

где $A_2(r_{\text{стр к трудн}}, r_{\text{услож зад}}, r_{\text{дост рез}}, \Phi_6)$ — ответ второго шага алгоритма.

Значение Φ_6 определено на основе исходных данных (размеченных экспертами) либо вычислено с помощью обращения к LLM.

4.3. Дифференциация драйвового и максимального типов восприятия ТЖС с помощью взвешенных сумм показателей признаков типов

Рассмотрим множество признаков, перечисленных в пункте 3.3. Признаки принимали в соответствующих столбцах A1–6B8 бинарные значения (0 или 1). Вероятность отнесения ситуации к драйвовому типу возрастала с увеличением числа признаков драйвового типа, имеющих значение 1; аналогично вероятность отнесения ситуации к максимальному типу возрастала с числом признаков максимального типа, равных 1. Обозначим степени значимости признаков, указанные в пункте 3.3, через c_s^i, где $s \in S_i, i \in \{1, 2\}$ — буквенно–цифровой индекс признака, S_i — множество буквенно-цифровых индексов i-го словаря.

Рассмотрим выражение

$$M_i(x^i) = \sum_{s \in S_i} c_s^i x_s^i, i \in \{1, 2\}, \tag{6}$$

где $x_s^i, s \in S_i, i \in \{1, 2\}$ — значение признака с буквенно-цифровым индексом s.

Тогда:

$$A_3(x^1, x^2) = \begin{cases} 1, & \text{если } M_1(x^1) > M_2(x^2), \\ 2, & \text{иначе.} \end{cases} \tag{7}$$

5. Эмпирическое исследование возможностей классификации типов восприятия ТЖС

Для проверки возможности алгоритмического определения типов восприятия ТЖС на основе совместного анализа профилей опросника ТОРС-36 и текстовых описаний трудных ситуаций было проведено эмпирическое исследование, включавшее 6 последовательных экспериментов. Результаты автоматизированной обработки данных сопоставлялись с эталонными экспертными оценками. Первые три исследования носили пилотажный характер и проводились на одной и той же пилотажной выборке. Эксперименты были направлены на последовательное уточнение условий, при которых достигается наибольшая точность классификации типов восприятия ТЖС.

В качестве технического инструментария классификации использовались две LLM: GPT-4.1 (OpenAI), доступ к которой осуществлялся через OpenAI API (эксперименты 1-4), и deepseek-chat (DeepSeek, эксперименты 4-5), доступ к которой обеспечивался через DeepSeek API. Для модели GPT-4.1 параметр генерации «температура» был зафиксирован на значении 0 (t = 0), что соответствует минимальной вариативности ответов. Для модели deepseek-chat параметр «температура» составил 0,3 (t = 0,3); такое значение обычно способствует высокой предсказуемости, фактической точности и воспроизводимости ответов, однако некоторая степень вариативности при этом сохраняется.

5.1. Эксперимент 1. Оценка точности классификации типов восприятия ТЖС на основе качественных данных с использованием LLM

Первый эксперимент был направлен на оценку возможности LLM определения типа восприятия ТЖС исключительно на основе качественных данных – текстовых описаний трудных ситуаций, без привлечения профилей по опроснику ТОРС-36. LLM предъявлялся специализированный промпт, содержащий описания признаков пяти типов. Точность классификации составила 43%, что свидетельствует о недостаточности одних лишь качественных данных для надежного различения типов восприятия ТЖС с помощью LLM.

5.2. Эксперимент 2. Оценка точности классификации типов восприятия ТЖС на основе качественных данных и профилей по опроснику ТОРС-36 без указания пороговых значений с использованием LLM

Во втором эксперименте в предъявляемый LLM набор данных были дополнительно включены профили респондентов по опроснику ТОРС-36, однако пороговые значения выраженности шкал LLM не сообщались. Предполагалось, что наличие количественной информации позволит LLM самостоятельно выявить закономерности соотнесения профилей с типами. Точность классификации возросла до 59%. Несмотря на улучшение по сравнению с первым экспериментом, результат остался достаточно умеренным, что указывает на ограниченную способность модели к имплицитному выведению правил категоризации без явных критериев.

5.3. Эксперимент 3. Оценка точности классификации типов восприятия ТЖС на основе качественных данных и профилей ТОРС-36 с указанием пороговых значений выраженности шкал

В третьем эксперименте в промпт наряду с признаками типов были включены пороговые значения выраженности шкал. Учет этих формальных критериев позволил LLM более точно соотносить структуру профиля ТОРС-36 с соответствующим типом восприятия ТЖС. Точность классификации достигла 75%, что превышает показатели предыдущих серий и подтверждает ведущую роль пороговых правил в интерпретации количественных данных.

5.4. Эксперимент 4. Оценка точности классификации типов восприятия ТЖС на основе гибридного алгоритма определения типов восприятия ТЖС

Четвертый эксперимент предполагал разработку и апробацию гибридного алгоритма, интегрирующего количественные и качественные данные на основе двухступенчатой процедуры. Алгоритм был реализован на языке Python. На первом этапе с использованием шести шкал опросника ТОРС-36 («Стремление к трудности», «Усложнение задачи», «Достижение результата», «Избегание трудности», «Облегчение задачи», «Отстранение от проблемы») и пороговых значений $1,5$ осуществлялась первичная категоризация респондентов в три группы: группу «Приближение» (драйвовый, максимальный, оптимальный типы), группу «Уход» (отстраненный тип) и группу «Амбивалентное восприятие» (амбивалентный тип). На втором

этапе внутри группы «Приближение» решалась задача дифференциации драйвового и максимального типов. Для этого применялся LLM-анализ текстовых описаний ТЖС, в рамках которого LLM предъявлялся специализированный промпт, содержащий словарь значимых для данных типов признаков. Текстовые описания ТЖС, отнесенные к группе «Приближение», но не удовлетворяющие критериям драйвового или максимального типов, были отнесены к оптимальному типу.

Итоговая точность гибридного алгоритма составила 83%, что превышает результаты всех предшествующих версий. Высокая точность гибридного алгоритма, достигнутая в пилотажном исследовании с использованием модели GPT-4.1, послужила основанием для его проверки на полной выборке с применением модели deepseek-chat с пороговым значением шкалы «Облегчение задачи», равным $1,75$; итоговая точность в этом случае составила 85% (первые два шага соответствовали описанным в разделе 4, запрос к LLM содержал описание драйвового и максимального типов с инструкцией определить, к какому из них относится описание ТЖС). Детальный анализ точности на отдельных этапах алгоритма показал, что на первом этапе (распределение в группы «Приближение», «Уход», «Амбивалентное восприятие») процент совпадений с экспертной оценкой достиг 90,5% (число текстовых описаний ТЖС, отнесенных к неопределенному типу в этом и последующих экспериментах — 7), а на втором этапе (дифференциация драйвового и максимального типов внутри группы «Приближение») – 87,5%. Точность классификации при использовании единого запроса для дифференциации максимального и драйвового типов составила 85%.

Таблица 1: Шаг 1 (одинаков для четвертого и последующих экспериментов)

Тип	1, 2, 3	4	5	Правильные ответы
1	14	2	0	87,5%
2	12	1	0	92,3%
3	29	5	0	85,29%
4	3	117	0	97,5%
5	0	1	9	90%

Таблица 2: Шаг 2 для определения признака Ф6 с помощью deepseek-chat

Тип	1, 2	3	4	5	Правильные ответы
1	14	0	2	0	87,5%
2	12	0	1	0	92,3%
3	6	23	5	0	67,65%
4	1	2	117	0	97,5%
5	0	0	1	9	90%

Таблица 3: Шаг 3 для цельного запроса для дифференциации драйвового и максимального типов для deepseek-chat

Тип	1	2	3	4	5	Правильные ответы
1	10	3	1	2	0	62,5%
2	1	6	5	1	0	46,15%
3	0	1	28	5	0	82,35%
4	0	1	2	117	0	97,5%
5	0	0	0	1	9	90%

5.5. Эксперимент 5. Оценка точности классификации на основе разметки LLM

Ранее было показано, что LLM могут выступать в качестве инструмента поддержки исследователя при первичном анализе больших массивов текстовых данных, позволяя существенно сократить временные затраты без снижения качества разметки, как показано в статье А.А. Хлебниковой, Е.В. Битюцкой, Г.В. Калачева, Э.Э. Гасанова [7]. Это стало основанием для использования LLM при разметке качественных данных в настоящем исследовании: такая разметка, необходимая для выполнения шагов 2 и 3 алгоритма, осуществлялась посредством запросов к LLM. Для каждого случая выполнялись три запроса: один – для выявления показателя Ф6, два – для оценки 16 признаков драйвового и максимального типа. В запросе для выявления показателя Ф6 перечислялись типы ситуаций, связанных с угрозой жизни и здоровью для самого респондента, его близких либо питомцев. Для выявления 16-ти признаков типа LLM были предоставлены описания признаков, после чего был запрошен вывод в формате «$(d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}, d_{11}, d_{12}, d_{13}, d_{14}, d_{15}, d_{16})$», где d_i принимал значения «1», если признак с номером i встречался в тексте, и значение «0», если не встречался.

Таблица 4: Шаг 3 для дифференциации драйвового и максимального типов с помощью взвешенных сумм на основе разметки LLM deepseek-chat

Тип	1	2	3	4	5	Правильные ответы
1	13	1	0	2	0	81,25%
2	4	8	0	1	0	61,54%
3	4	2	23	5	0	67,65%
4	0	1	2	117	0	97,5%
5	0	0	0	1	9	90%

Можно заметить, что в этом случае процент ошибок для 3-го класса на $14,7\%$ выше, чем в случае гибридной обработки. Это связано с тем, что при едином запросе алгоритмическая классификация в случае неопределенности присваивала значение 3; соответственно процент ошибок для 1-го и 2-го классов был ниже.

Полученные результаты позволяют заключить, что совмещение формализованной обработки количественных показателей (на основе пороговых значений) с анализом качественных описаний является наиболее эффективной стратегией автоматической классификации типов восприятия ТЖС среди рассмотренных вариантов, использующих LLM.

Итоговая точность алгоритма определения типов восприятия ТЖС на основе разметки LLM составила 85%. Следует отметить, что гибридный алгоритм (с единым запросом для дифференциации драйвового и максимального типов) и алгоритм на основе взвешенных сумм показателей признаков дали одинаковую точность — 85%. Выбор в пользу гибридного подхода обусловлен его большей простотой: он требует одного запроса к LLM, тогда как алгоритм на основе взвешенных сумм предполагает предварительное вычисление 16 бинарных признаков и их весовых коэффициентов.

5.6. Эксперимент 6. Оценка точности классификации на основе разметки, производимой экспертами

Данный эксперимент является репликацией эксперимента, описанного в статье Е.В. Битюцкой, Э.Э. Гасанова, Н.А. Патрашкина, К.В. Хазовой [5], и выполнен на материале опросника TOPC-36.

В ходе работы экспертами была произведена ручная разметка ответов респондентов по шестнадцати признакам. На основе этих данных был проведен эксперимент с шагами 2 и 3 алгоритма, вычисляемыми с помощью данной разметки.

Итоговая точность алгоритма на основе разметки экспертов и полной выборки составила 85,5%. На шаге 2 точность составляет 87% в случае, если признак Ф6 определяет эксперт-психолог.

Показатель 85,5% является максимальным в серии экспериментов. Однако разница с результатом при LLM-разметке (85%) составляет 0,5 процентного пункта и не является статистически значимой. Этот результат следует интерпретировать как свидетельство сопоставимости экспертной и LLM-разметки. Кроме того, данный вывод может быть обусловлен и тем, что в качестве эталонной классификации использовались экспертные оценки.

Таблица 5: Шаг 2 для разметки экспертами

Тип	1, 2	3	4	5	Правильные ответы
1	14	0	2	0	87,5%
2	12	0	1	0	92,3%
3	7	22	5	0	64,7%
4	3	0	117	0	97,5%
5	0	0	1	9	90%

Таблица 6: Шаг 3 для дифференциации драйвового и максимального типов с помощью взвешенных сумм показателей признаков типов на основе разметки экспертами

Тип	1	2	3	4	5	Правильные ответы
1	14	0	0	2	0	87,5%
2	3	9	0	1	0	69,23%
3	3	4	22	5	0	64,7%
4	1	2	0	117	0	97,5%
5	0	0	0	1	9	90%

6. Заключение

В настоящем исследовании проверялась возможность использования опросника ТОРС-36 в качестве инструмента классификации типов восприятия ТЖС, а также оценивалась роль анализа текстовых описаний с привлечением LLM. Для достижения цели были поставлены три задачи: разработка и валидация алгоритма определения типа на основе

профиля ТОРС-36, сопоставление результатов алгоритмической классификации с экспертной оценкой и сравнение эффективности различных модификаций.

В рамках первой задачи был разработан и апробирован гибридный алгоритм, реализованный на языке Python и включающий два этапа: первичную категоризацию на основе пороговых правил, применяемых к шести шкалам ТОРС-36, и последующую дифференциацию драйвового и максимального типов посредством LLM-анализа текстовых описаний.

Вторая задача решалась на материале всей серии из шести экспериментов. Точность гибридного алгоритма составила 83% на пилотажной выборке (модель GPT-4.1) и 85% на полной (модель deepseek-chat). На первом этапе совпадение с экспертной оценкой достигает 90,5%, на втором — 87,5%; при использовании единого запроса для разграничения драйвового и максимального типов — 85%. Последовательное усложнение алгоритма позволило оценить вклад различных источников информации и методов обработки: использование исключительно качественных данных обеспечило 43% точности, что недостаточно для надежного различения пяти типов; добавление профилей опросника ТОРС-36 без указания пороговых значений повысило показатель точности до 59%; включение пороговых значений шкал опросника ТОРС-36 в промпт позволило достичь 75%, что подтверждает ведущую роль формальных критериев в интерпретации количественных данных.

В рамках третьей задачи наиболее результативным среди подходов, использующих LLM, признан гибридный, сочетающий формализованную обработку количественных показателей с анализом качественных описаний (85% на полной выборке). При этом алгоритм на основе взвешенных сумм показателей признаков дал ту же точность (85%); выбор гибридного алгоритма обусловлен его большей простотой. Максимальная точность в серии (85,5%) достигнута при использовании того же алгоритма с экспертной разметкой, что на 0,5 процентного пункта превышает результат LLM-разметки; это различие не является статистически значимым, что не позволяет говорить о решающем преимуществе экспертной разметки.

Таким образом, цель исследования достигнута: подтверждена возможность использования опросника ТОРС-36 в качестве инструмента классификации типов восприятия ТЖС, а анализ текстовых описаний ТЖМ с привлечением LLM может применяться как средство поддержки эксперта-психолога при решении задачи анализа качественных описаний ТЖС. Разработанный гибридный алгоритм определения типов восприятия ТЖС обеспечивает точность, сопоставимую с экспертной, и может служить основой для масштабируемой, воспроизводимой и стандартизированной диагностики.

Перспективы дальнейшей работы включают расширение выборки, оптимизацию пороговых значений, а также интеграцию предложенного подхода в практику психологической диагностики.

Список литературы

[1] Е. В. Битюцкая, “Современные подходы к изучению совладания с трудными жизненными ситуациями”, *Вестник Московского университета*, **14**:1 (2011), 100–111.

[2] Е. В. Битюцкая, Е. В. Карцева, “Рисуночная методика «Моя трудная жизненная ситуация» как инструмент диагностики восприятия трудной ситуации”, *Журнал практического психолога*, **4** (2013), 102–132.

[3] Е. В. Битюцкая, “Типы ориентаций в трудных ситуациях”, *Вопросы психологии*, **5** (2018), 41–53.

[4] Е. В. Битюцкая, А. А. Корнеев, “Диагностика восприятия жизненных трудностей: ситуационный опросник «Типы ориентаций в трудной ситуации»”, *Вестник Московского государственного областного университета*, **4** (2020), 141–163. DOI: 10.18384/2224-0209-2020-4-1047.

[5] Е. В. Битюцкая, Э. Э. Гасанов, Н. А. Патрашкин, К. В. Хазова, “Валидизация опросника «Типы ориентаций в трудной ситуации» на основе методов компьютерного моделирования и контент-анализа”, *Психология. Журнал Высшей школы экономики*, **21**:4 (2024), 729–752. DOI: 10.17323/1813-8918-2024-4-729-752.

[6] Е. В. Битюцкая, А. А. Корнеев, А. А. Хлебникова, “Разработка и психометрическая проверка краткой версии опросника «Типы ориентаций в трудных ситуациях» ТОРС-36”, *Психологические исследования*, **20**:111 (2026).

[7] А. А. Хлебникова, Е. В. Битюцкая, Г. В. Калачев, Э. Э. Гасанов, “Автоматизация разметки текстов о жизненных трудностях с использованием больших языковых моделей”, *Интеллектуальные системы. Теория и приложения*, **29**:3 (2025), 53–75.

[8] C. F. Barros, B. B. Azevedo, V. V. Graciano Neto, M. Kassab, M. Kalinowski, H. A. D. do Nascimento, M. C. G. S. P. Bandeira, “Large language model for qualitative research: A systematic mapping study”, *2025 IEEE/ACM International Workshop on Methodological Issues with*

Empirical Studies in Software Engineering (WSESE), IEEE, 2025. DOI: 10.1109/WSESE66602.2025.00015.

[9] E. V. Bityutskaya, E. E. Gasanov, K. V. Khazova, N. A. Patrashkin, "Classifying the Perception of Difficult Life Tasks: Machine Learning and/or Modeling of Logical Processes", *Psychology in Russia: State of the Art*, **17**:2 (2024), 64–84. DOI: 10.11621/pir.2024.0205.

[10] J. W. Creswell, *A concise introduction to mixed methods research*, SAGE, 2015.

Благодарности

Авторы выражают глубокую благодарность кандидату психологических наук, доценту кафедры общей психологии факультета психологии МГУ имени М.В. Ломоносова Битюцкой Екатерине Владиславовне и доктору физико-математических наук, заведующему кафедрой математической теории интеллектуальных систем механико-математического факультета МГУ имени М.В. Ломоносова Гасанову Эльяру Эльдаровичу за научное руководство, ценные рекомендации и постоянную поддержку на всех этапах исследования.

Отдельную признательность авторы выражают Анне Георгиевне Докучаевой за квалифицированную помощь в обработке качественных данных.

Исследование выполнено за счет гранта Российского научного фонда № 25-18-00737, https://rscf.ru/project/25-18-00737/.

Статья поступила 20 сентября 2026 г.

An algorithm for identifying types of perception of difficult life situations based on the short version of the "Types of Orientations in Difficult Situation" (TODS-36) questionnaire.

I. O. Berger, A. A. Khlebnikova

The article is devoted to the validation of a short version of the questionnaire "Types of orientations in difficult situations" (TODS-36) as a tool for classifying types of perception of difficult situations and evaluating the role of text description analysis involving large language models in determining these types. The following tasks are being solved: development and validation of an algorithm for classifying types of perception of difficult life situations based on the TODS-36 profile using large language models to analyze text descriptions; comparison of the

results of algorithmic classification with expert assessment; comparison of the effectiveness of various algorithm modifications. The study sample included 200 text descriptions of difficult situations received from 200 respondents (131 women, 69 men) aged 14 to 54 years. The reference classification was performed by expert psychologists based on a joint analysis of the quantitative indicators of the TODS-36 questionnaire scales and qualitative data (text descriptions of difficult situations). A series of six experiments was conducted. In experiments 1–4, the role of quantitative data and formal classification rules gradually increased. In experiment 5, classification accuracy was assessed when qualitative data were coded by a large language model; in experiment 6, it was assessed when the coding was performed by expert psychologists. The most effective among the algorithms using a large language model was the hybrid one, in which threshold rules are applied to quantitative indicators and the large language model analyzes qualitative data. Its accuracy was 85% on the full sample. At the same time, the highest accuracy in the series of experiments – 85.5% – was obtained using the same algorithm with expert coding of qualitative data. The results obtained confirm the possibility of using the TODS-36 questionnaire as a tool for classifying types of perception of difficult situations and allow us to consider large language models as a means of supporting an expert in classifying types of perception of difficult situations.

Keywords: coping, difficult life situation, situation perception, TODS-36, an algorithm for determining the types of perception of difficult situations, large language models, mixed-methods research.

Funding. The study was funded by a grant from the Russian Science Foundation, project number № 25-18-00737, https://rscf.ru/project/25-18-00737/.

References

[1] Bityutskaya, E. V., “Modern approaches to the study of coping with difficult life situations”, *Moscow University Bulletin*, **14**:1 (2011), 100–111 (In Russian).

[2] Bityutskaya, E. V., Kartseva E. V., “The "My Difficult Life Situation" Drawing Technique as a Tool for Diagnosing the Perception of a Difficult Situation”, *Journal of the Practicing Psychologist*, **4** (2013), 102–132 (In Russian).

[3] Bityutskaya, E. V., “Tipy orientatsiy v trudnykh situatsiyakh [Types of orientation in difficult situations].”, *Voprosy Psikhologii*, **5** (2018), 41–53 (In Russian).

[4] Bityutskaya, E. V., Korneev, A. A, “Diagnostics of perception of life events: the situational version of the questionnaire “types of orientations in difficult situation””, *Vestnik Moskovskogo Gosudarstvennogo Oblastnogo Universiteta*, **4** (2020), 141–163. DOI: 10.18384/2224-0209-2020-4-1047 (In Russian).

[5] Bityutskaya, E. V., Gasanov, E. E., Patrashkin, N.A., Khazova, K.V., "Validation of the Questionnaire "Types of Orientations in Difficult Situations" Based on Computer Modeling and Content Analysis Methods", *Psychology. Journal of the Higher School of Economics*, **21**:4 (2024), 729–752. DOI: 10.17323/1813-8918-2024-4-729-752 (In Russian).

[6] Bityutskaya, E. V., Korneev, A.A., Khlebnikova A.A., "Development and Psychometric Validation of the Short Version of the "Types of Orientations in Difficult Situation" Questionnaire (TODS-36)", *Psychological research*, **20**:111 (2026) (In Russian).

[7] Khlebnikova A. A., Bityutskaya E. V., Kalachev G. V., Gasanov E. E., "The Use of Language Models in Automated Markup of Texts on Life Difficulties", *Intelligent Systems. Theory and Applications*, **29**:3 (2025), 53–75 (In Russian).

[8] C. F. Barros, B. B. Azevedo, V. V. Graciano Neto, M. Kassab, M. Kalinowski, H. A. D. do Nascimento, M. C. G. S. P. Bandeira, "Large language model for qualitative research: A systematic mapping study", *2025 IEEE/ACM International Workshop on Methodological Issues with Empirical Studies in Software Engineering (WSESE), IEEE*, 2025. DOI: 10.1109/WSESE66602.2025.00015.

[9] Ekaterina V. Bityutskaya, Elyar E. Gasanov, Kseniia V. Khazova, Nikita A. Patrashkin, "Classifying the Perception of Difficult Life Tasks: Machine Learning and/or Modeling of Logical Processes", *Psychology in Russia: State of the Art*, **17**:2 (2024), 64–84. DOI: 10.11621/pir.2024.0205.

[10] J. W. Creswell, *A concise introduction to mixed methods research*, SAGE, 2015.

Received on September 20, 2026

От VC-размерности к законам масштабирования: эволюция теоретических подходов к оценке сложности и обобщающей способности

А. В. Грабовой*

В работе представлен обзор современного состояния исследований, посвященных теоретическим методам оценки сложности моделей машинного обучения и их связи с обобщающей способностью. Методологически работа оформлена как обзор предметной области. Указываются источники библиографического поиска, этапы дедупликации и полуавтоматического сужения корпуса документов, экспертный отбор и состав итогового аналитического набора публикаций. Основной вклад обзора состоит в анализе классических мер сложности из статистической теории обучения и их ограниченности для объяснения поведения современных перепараметризованных глубоких моделей машинного обучения. Показано, что VC-размерность, радемахеровские и PAC-байесовские оценки сохраняют значение как теоретический результат, однако характеризуют мощность класса гипотез в худшем случае и лишь ограниченно учитывают локальные геометрические характеристики решения, траекторию оптимизации и архитектурные особенности. Основное внимание уделено современным направлениям, включающим анализ ландшафта функции потерь, спектральных характеристик гессиана, методов оптимизации и эмпирических законов масштабирования. В результате предложена интерпретация текущего состояния предметной области, согласно которой современная теория сложности должна описывать не только класс моделей, но и эффективную сложность решений, учитывающих как локальные характеристики модели, так и данные, на которых она обучается.

Ключевые слова: теория сложности моделей, обобщающая способность, глубокое обучение, VC-размерность, радемахеровская сложность, ландшафт функции потерь, гессиан, двойной спуск, законы масштабирования.

**Грабовой Андрей Валериевич* — кандидат физико-математических наук, старший научный сотрудник, Институт проблем управления им. В. А. Трапезникова РАН, Москва, Россия, e-mail: grabovoy@ipu.ru, ORCID: 0000-0002-4031-0025.

Grabovoy Andrey Valerievich — Candidate of Physical and Mathematical Sciences, Senior Researcher, Institute of Control Sciences of the Russian Academy of Sciences, Moscow, Russia.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

1. Введение

Теория сложности моделей в машинном обучении — раздел статистической теории обучения и инструмент для анализа моделей машинного обучения. Теория отвечает на прикладные вопросы, связанные с анализом моделей. Например, какой объем данных нужен для обучения, почему архитектуры с близким числом параметров дают разное обобщение, какие методы оптимизации связаны с устойчивыми решениями и как интерпретировать рост качества при масштабировании моделей и данных. Для моделей глубокого обучения это существенно, так как современные модели сочетают сильную перепараметризацию с высокой эмпирической эффективностью. В данной работе под обзором предметной области (англ. *scoping review*) предполагается формат систематизированного описания области и сопоставления исследовательских трендов.

В рамках классической сложности основным объектом анализа выступает класс гипотез. В этом VC-размерность [1] и PAC-обучаемость [2] стали базовыми инструментами контроля обобщающей способности. Позднее это дополнили ростовые функции и оценки [3, 5], зависящие от выборки (англ. data-dependent). Однако развитие глубоких нейронных сетей показало, что одной лишь мощности класса моделей недостаточно для объяснения поведения моделей машинного обучения [30]. Модель с миллионами или миллиардами параметров может как переобучиться на заданный корпус данных, так и демонстрировать высокое качество обобщения [16]. При этом один и тот же класс функций способен вести себя по-разному в зависимости от алгоритма обучения, архитектурных ограничений и структуры данных [16, 30].

Наблюдается разрыв между числом параметров и эффективной сложностью решения. В рамках классической теории рост параметризации повышает риск переобучения [2]. Однако современные эмпирические результаты демонстрируют более сложную зависимость. Увеличение размера модели может ухудшать обобщение только при отдельных методах оптимизации, тогда как в более широком перепараметризованном режиме обучения качество нередко вновь улучшается. Следовательно, число параметров выступает лишь грубым методом оценки сложности и само по себе не определяет, какие функции реализуются после обучения. Существенными факторами становятся архитектурные ограничения, свойства данных, неявная регуляризация оптимизатора и геометрия найденного минимума [12].

По этой причине теория сложности для глубокого обучения постепенно смещается от анализа вместимости класса гипотез к анализу эффективной сложности конкретных решений [13]. В современных иссле-

дованиях это выражается в геометрии ландшафта функции потерь [12] и спектральным характеристикам гессиана [14]. Параллельно усилилось внимание к режимам обучения [49] и эмпирическим законам масштабирования [13, 18]. В данном обзоре эта эволюция рассматривается как единый исследовательский тренд. От классических верхних оценок к подходам, ориентированным на описание свойств решений, достигаемых в процессе обучения.

В данной работе выводы строятся с использованием заранее зафиксированной процедуры построения литературного корпуса. В анализ включены классические источники статистической теории обучения и современные теоретические, методологические и обзорные публикации по геометрии ландшафта, методам обучения и масштабированию. Такой дизайн позволяет сопоставить классические и современные исследовательские тренды на широком и явно описанном наборе источников.

2. Методология обзора теории сложности моделей машинного обучения

Задача данной работы состоит в том, чтобы охватить основные направления теоретической литературы о сложности и обобщении в методах машинного обучения, зафиксировать процедуру построения корпуса и обеспечить воспроизводимость отбора на уровне источников данных и критериев включения. Данный обзор не заменяет систематический обзор с полной регистрацией протокола по стандартам PRISMA (англ. Preferred Reporting Items for Systematic Reviews and Meta-Analyses), однако позволяет явно разделить широкий поисковый охват и содержательный анализ.

Первичный библиографический сбор публикаций по тематике исследования выполняется по трем открытым агрегаторам научных публикаций — OpenAlex[1], arXiv[2] и Crossref[3]. Поиск выполняется посредством однотипных по смыслу поисковых запросов. Запросы охватывают классические и современные направления исследований. К классическим относятся VC-размерность и рademахеровская сложность. К современным — PAC-байесовские и информационно-теоретические границы обобщения, алгоритмическая устойчивость, ландшафт функции потерь, спектр гессиана,

[1] OpenAlex API: `https://developers.openalex.org/api-reference/introduction#api-overview`.

[2] arXiv API: `https://info.arxiv.org/help/api/index.html`.

[3] Crossref REST API: `https://www.crossref.org/documentation/retrieve-metadata/rest-api/`.

двойной спуск, нейронное касательное ядро, режимы обучения, законы масштабирования, а также архитектуры трансформеров и диффузионных моделей. Для каждой формулировки запроса из каждого источника извлекается по 100 записей. Затем результаты объединяются в единый список публикаций с заголовками и, по возможности, аннотациями.

Записи с разных источников сопоставляются по идентификаторам DOI, совпадению нормализованных заголовков и близким полям описания. Дубликаты объединяются, причем для каждой уникальной работы сохраняется информация о числе поисковых попаданий и перечне запросов, которые ведут к включению работы в список. Такой подход учитывает устойчивость темы к различным формулировкам запроса, не подменяя содержательный отбор автоматически собранными показателями.

Полный дедуплицированный массив по объему не подлежит содержательному анализу целиком. Следовательно, на следующем этапе к каждой записи применяется правило-ориентированная процедура ранжирования, использующая данные заголовка и аннотации. Процедура учитывает совпадение с теоретически существенными лексическими шаблонами и принадлежность к заранее заданным тематическим группам, таким как классические методы, PAC-байесовские методы и информация, ландшафт и гессиан, режимы обучения нейронного касательного ядра (англ. Neural Tangent Kernel, NTK) и двойного спуска, масштабирование и архитектуры. Дополнительно учитывается число независимых поисковых попаданий и штрафы за доминирование узко прикладного контекста при слабой теоретической базе публикации. Все найденные публикации дополнительно распределяются по тематическим группам с ограничением перекоса, чтобы сокращенный перечень кандидатов сохранял представительность по исследовательским трендам, а не сводился к одной выделенной области исследований. В результате для экспертного разбора формируется сокращенный перечень из 500 релевантных записей.

На этапе ручного отбора из сокращенного списка кандидатов исключаются публикации, в которых термины обобщения и сложности использовались преимущественно в прикладном или узкоотраслевом значении без теоретического вклада в предметную область. Также исключаются работы, не добавляющие к обзору нового содержания относительно уже включенных источников. Предпочтение отдается публикациям, удовлетворяющим условиям: (1) формулируется фундаментальный теоретический результат или нетривиальная оценка сложности, (2) предлагается современный механизм, существенный для понимания сложности, (3) фиксируется устойчивый эмпирический результат, который теория сложности обязана учитывать.

Независимо от сокращенного списка кандидатов в итоговый перечень публикаций вручную включен набор фундаментальных источников по статистической теории обучения, устойчивости, ключевым архитектурам и режимам обучения. Такое дополнение методологически обосновано тем, что ранние канонические работы и отдельные базовые публикации по современным архитектурам не всегда устойчиво поднимаются в выдаче поиска по ключевым словам, ориентированного на недавнюю прикладную и вторичную литературу. Без явного включения этих источников корпус искажался бы в сторону современных обзоров.

Таблица 1. Тематическое распределение кандидатов и финального корпуса

Тематическая группа	**В 500**	**Доля, %**	**Auto**	**Manual+**	**В финале**
Классические границы	101	20.2	13	6	19
PAC-байесовские методы и информация	120	24.0	5	2	7
Ландшафт и гессиан	96	19.2	11	4	15
Режимы обучения	84	16.8	11	4	15
Масштабирование и архитектуры	45	9.0	7	4	11
Обзорные работы	54	10.8	9	0	9
Итого	**500**	**100.0**	**56**	**20**	**76**

Итоговый аналитический корпус включает 76 публикаций, из них 13 опорных работ по ключевым направлениям обзора, 44 дополнительных источников для контекста и сопоставления близких трендов и 19 зафиксированных фундаментальных работ. Из табл. 1 видно, что в списке из 500 кандидатов крупнейшие доли имеют вероятностно-информационная и классическая группы. Наблюдаемая неравномерность групп отражает свойства поисковой процедуры и текущую структуру доступной литературы, а не априорную иерархию важности направлений.

Несмотря на формализованную процедуру отбора, предложенная методология имеет ряд ограничений. Во-первых, поиск опирается на покрытие выбранных библиографических агрегаторов и может неполно отражать работы, индексируемые неравномерно или с задержкой. Во-вторых, ключевые запросы и лексические шаблоны неизбежно задают тематическое смещение, так как они преимущественно извлекают публикации, использующие устоявшийся словарь теории обобщения, по сравнению с работами с близким содержанием, но иной терминологией. В-третьих, правило-ориентированное ранжирование и последующий экспертный отбор снижают шум, но сохраняют интерпретационную зависимость от исследовательской рамки обзора. В-четвертых, обзор предметной области ориентирован на структурирование области и сравнительный анализ, а не

на метаанализ, поэтому выводы носят концептуальный, а не статистически агрегированный характер. Представленная процедура не претендует на полноту в смысле систематического обзора всех существующих публикаций по теме. Назначение работы — обеспечить прозрачный путь от широкого поиска к управляемому набору источников и зафиксировать границы обзорного анализа.

3. Классические меры сложности и их ограничения

Классическая теория обучения связывает обобщение с контролем мощности класса гипотез. VC-размерность измеряет способность класса реализовывать все возможные бинарные разметки на конечных выборках и тем самым служит критерием равномерной сходимости эмпирического риска к истинному [1,3]. В PAC-постановке это приводит к оценкам требуемого объема выборки через параметры точности и сложность класса [2]. Для классов гипотез умеренной сложности такая картина остается концептуально строгой: конечная VC-размерность означает обучаемость, а рост сложности увеличивает требования к данным. Для бинарной задачи это выражается оценкой обобщения через VC-размерность:

$$R(h) \leq \hat{R}_S(h) + \sqrt{\frac{d_{VC}\left(\log \frac{2n}{d_{VC}} + 1\right) + \log \frac{4}{\delta}}{n}}, \tag{1}$$

где $R(h)$ — истинный риск, $\hat{R}_S(h)$ — эмпирический риск на выборке S объема n, d_{VC} — VC-размерность класса, $\delta \in (0,1)$ — уровень доверия.

Для моделей с большим числом параметров VC-оценки становятся завышенными и перестают быть информативными. Поэтому для глубоких моделей машинного обучения VC-размерность сохраняет значение как формальное понятие статистической теории сложности, но не дает ответа на вопрос, почему обученная модель демонстрирует обобщающую способность. Проблема в том, что если две модели имеют одинаковую архитектуру, то их VC-размерность совпадает с точностью до порядка величины, однако после обучения они ведут себя по-разному. Следовательно, сама мощность класса не определяет тот объект, который фактически определяет качество модели.

Отдельно стоит упомянуть комбинаторный подход [4], который учитывает структуру выборки, а не только грубую оценку в худшем случае класса. В рамках этого подхода объектом исследования остается класс

гипотез, а не локальные геометрические характеристики решения в параметрическом пространстве.

Дальнейшим развитием теории сложности является радемахеровская сложность [5,6], которая зависит от рассматриваемой выборки и учитывает геометрию данных [31,32]. Для линейных моделей и некоторых специальных классов моделей радемахеровский подход дает более точные оценки. Данный подход применяется в глубоком обучении, поскольку оценки, зависящие от выборки, частично отделяют реально наблюдаемую сложность задачи от чисто комбинаторной мощности класса моделей [31,32]. Современные исследования развивают радемахеровскую сложность в нескольких направлениях. Например, для обобщения классических оценок [47], а также оценок для сложных классов функций [46]. Различные обзорные работы, также подчеркивают, что радемахеровский подход для глубоких сетей связан не столько с поиском одной универсальной оценки, сколько с выявлением границ применимости моделей [42,75]. Эмпирическая радемахеровская сложность класса $\mathcal{H}$ на выборке $S = (x_i)_{i=1}^n$ определяется выражением:

$$\hat{\mathfrak{R}}_S(\mathcal{H}) = \mathbb{E}_\sigma \left[\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i h(x_i) \right], \tag{2}$$

где $\sigma_i \in \{-1, +1\}$ — независимые случайные знаки Радемахера. Для липшицевых потерь оценка принимает вид:

$$R(h) \leq \hat{R}_S(h) + 2\,\hat{\mathfrak{R}}_S(\mathcal{H}) + 3\sqrt{\frac{\log(2/\delta)}{2n}}. \tag{3}$$

Разница радемахеровского подхода с VC-размерностью (1) состоит в том, что сложность зависит от заданной выборки, а не только от мощности класса функций. Однако радемахеровские оценки по-прежнему характеризуют весь класс функций, тогда как обучение исследует лишь малую и структурированную его часть. Поэтому две модели одинаковой архитектуры, обученные разными оптимизаторами или с разными инициализациями, имеют сопоставимые оценки сложности сверху, но могут заметно отличаться по обобщающей способности.

С практической точки зрения важно, что оценки, зависящие от выборки, часто оказываются содержательными лишь после введения дополнительных структурных ограничений, таких как ограничение норм весов, липшицевых ограничений или специальных возмущений данных. Ввод специальных ограничений делает решение не полным, так как для получения оценки требуется заранее встраивать в модель информацию

о геометрии решения, то есть теория частично объясняет обобщение, а частично уже предполагает его.

С другой стороны, достоинство PAC-байесовских оценок состоит в том, что они связывают обобщение не только с функцией, но и с распределением над параметрами. Базовые постановки этого подхода даны в [7,8], а современные обобщения и обзорные интерпретации представлены в [22, 33, 34, 51]. На практике эти оценки оказываются содержательными среди современных оценок. Внутри этого же тренда появляются и более специализированные приложения, например анализ стохастического градиентного спуска в PAC-байесовском подходе [56, 66]. Однако и эти оценки зависят от выбора априорного распределения, формы возмущений и параметризации и потому задают скорее гибкую математическую рамку, чем прямое объяснение того, почему заданная модель глубокого обучения достигает решения с высокой обобщающей способностью. В одной из канонических форм PAC-байесовская оценка записывается как

$$\mathbb{E}_{\theta\sim Q} R(\theta) \le \mathbb{E}_{\theta\sim Q} \hat{R}_S(\theta) + \sqrt{\frac{\mathrm{KL}(Q\|P) + \log \frac{2\sqrt{n}}{\delta}}{2(n-1)}}, \tag{4}$$

где P — априорное распределение, Q — апостериорное, $\mathrm{KL}(Q\|P)$ — дивергенция Кульбака–Лейблера. В (4) видно, что сложность перенесена с класса функций на стоимость перехода от априорного распределения к апостериорному через KL-дивергенцию.

Задача в рамках информационно-теоретического подхода ставится иначе. Он связывает ошибку обобщения с количеством информации, извлекаемой алгоритмом из данных [9]. В глубоких моделях машинного обучения при этом сохраняются трудности с получением численных оценок. Примером такого результата является оценка:

$$\left|\mathbb{E}[R(W) - \hat{R}_S(W)]\right| \;\le\; \sqrt{\frac{2\sigma^2}{n}\, I(W;S)}, \tag{5}$$

где W — параметры после обучения, $I(W;S)$ — взаимная информация между параметрами и выборкой, σ — параметр субгауссовости потерь. Неравенство (5) отражает алгоритмическую сложность, то есть чем меньше информации, записанной в параметрах выборки, тем ниже ожидаемая ошибка обобщения.

В совокупности эти подходы указывают на методологический сдвиг. Теория постепенно отказывается от предположения, что существует одна универсальная мера сложности, одинаково адекватная для всех методов обучения. Вместо этого формируется набор частичных, но более предметных описаний: одни применимы к анализу класса функций, другие —

к стохастической динамике алгоритма, третьи — к локальной устойчивости уже найденного решения. В литературе этот сдвиг описывается как переход от единой статистической меры сложности к семейству взаимодополняющих описаний. Этот тезис последовательно проводится как в обзорных работах [43, 44, 48], так и в теоретических работах [70, 71, 73, 74].

Одним из важных направлений исследований теории сложности является теория устойчивости моделей машинного обучения [10, 11]. Данная теория занимает промежуточное положение между классической статистической теорией и современной алгоритмической постановкой вопроса. С одной стороны, устойчивость по-прежнему связана с обобщением через формальные оценки сверху. С другой стороны, устойчивость напрямую зависит от свойств алгоритма и потому анализирует реальный процесс обучения, а не только класс гипотез. Для глубокого обучения это связано с тем, что стохастический градиентный спуск является шумным и траекторно-зависимым алгоритмом. Устойчивость тесно связана с размером батча, длительностью обучения, кривизной ландшафта и типом регуляризации. Поэтому устойчивость оказывается полезной не столько как самостоятельная универсальная мера сложности, сколько как часть более широкой картины, связывающей оптимизацию и геометрию. В этой постановке теория устойчивости помогает понять, почему современные исследования все чаще рассматривают сложность как свойство процесса обучения модели, а не только класса функций модели. Она не решает проблему полностью, но указывает направление развития: от емкостных описаний к алгоритмозависимым описаниям. В канонической постановке для β-устойчивого алгоритма используется оценка вида

$$\left|R(A(S)) - \hat{R}_S(A(S))\right| \leq \beta, \tag{6}$$

где A — алгоритм обучения, S — обучающая выборка, β — параметр устойчивости. Запись (6) фиксирует формальную связь между устойчивостью алгоритма и ошибкой обобщения.

Ограниченность классических мер сложности проявляется не в том, что они неверны, а в том, что они описывают иной уровень абстракции по сравнению с тем, который требуется для анализа современных глубоких моделей. В перепараметризованных моделях важно не только то, насколько велик класс возможных функций, но и то, какое подмножество решений реально достижимо конкретным методом оптимизации, каковы локальные геометрические характеристики этих решений и какие индуктивные смещения встроены в архитектуру. Эти идеи проявились после работ, показавших, что глубокие сети способны идеально настраиваться на случайные метки, оставаясь при этом успешными на реальных

данных [30]. Отсюда следует вывод, что теория сложности для глубокого обучения не ограничивается контролем емкости класса гипотез. Она учитывает, по меньшей мере, три дополнительных аспекта: *алгоритм обучения*, *локальные геометрические характеристики решений* и *структуру данных, согласованную с архитектурой*. Это и мотивирует переход к современным геометрическим и ландшафтным подходам.

4. Ландшафт функции потерь и геометрические характеристики решений

Если классические меры оценивают пространство всех потенциально допустимых решений, то ландшафтные подходы ориентированы на анализ свойств локальных минимумов, к которым приводит обучение. Интерес к данному тренду усилился после работ по визуализации ландшафта потерь, показавших, что решения с высокой обобщающей способностью часто располагаются в более широких областях пространства параметров [12, 40, 62]. Идея плоских минимумов как возможного объяснения этого эффекта восходит к ранним работам по нейронным сетям [13]. В дальнейшем она была переосмыслена в терминах спектра гессиана, локальной устойчивости и неявной регуляризации алгоритмов обучения, а также в терминах нормализованных и параметризационно-устойчивых характеристик плоскости [57].

В данном тренде понятие сложности получает иную интерпретацию. Речь идет не о размере класса реализуемых функций, а о свойствах найденного решения в локальной геометрии. В этом смысле ландшафтная сложность оказывается ближе к эффективной сложности обучения, чем к абстрактной мощности параметрического семейства.

В этом направлении часто используется анализ гессиана функции потерь и его спектра. Эмпирические работы показывают типичную структуру спектра, где мало доминирующих направлений кривизны и много почти плоских направлений [14, 58, 60]. Это согласуется с гипотезой, что решения с высоким обобщением чаще лежат не в острых изолированных минимумах, а в более широких областях параметрического пространства. Локальная геометрия задается матрицей гессиана

$$H(\theta) = \nabla_\theta^2 \mathcal{L}(\theta), \tag{7}$$

а ее спектральные характеристики описываются собственными значениями

$$\lambda_1(\theta) \geq \lambda_2(\theta) \geq \cdots \geq \lambda_p(\theta) \geq 0. \tag{8}$$

Важными характеристиками ландшафта являются: спектральный радиус $\lambda_{\max}(H)$ и след $\mathrm{tr}(H) = \sum_i \lambda_i$, которые соответственно отражают максимальную локальную кривизну и интегральную кривизну в окрестности найденного решения.

С этой точки зрения плоские минимумы представляют интерес по двум причинам. Во-первых, они более устойчивы к малым возмущениям параметров и потому потенциально демонстрируют более стабильный перенос на новые данные. Во-вторых, они отражают не только свойства класса моделей, но и свойства траектории, посредством которой алгоритм достиг найденного решения. Поэтому связь между ландшафтом и обобщением тесно переплетается с анализом оптимизации.

Существуют и попытки формализовать эту интуицию в виде количественных мер. В частности, в недавних работах на основе гессиана предлагаются величины, характеризующие вариабельность кривизны по примерам и связывающие локальную геометрию с объемом выборки и устойчивостью обучения [15]. Значение таких подходов состоит в переводе обсуждения плоских и острых решений из качественной плоскости в количественную. В более широком контексте этот тренд включает также оценки, использующие представления о сложности через внутренние представления и предпочтение более простых функций в параметрическом отображении [63, 64].

Наряду с этим были предложены и алгоритмические методы, специально ориентированные на предпочтение широких минимумов. К таким работам относятся как анализ влияния размера батча и остроты минимума [37], так и методы, явно управляющие локальными геометрическими характеристиками решения, включая метод минимизации с учетом остроты (англ. Sharpness-Aware Minimization, SAM) и его более поздние обобщения на режим Edge of Stability (границы устойчивости) [38, 59]. Дополнительно есть работы, где предпочтение широких минимумов связывается с фильтрацией высокочастотных компонент стохастической динамики [67]. Это важно, поскольку влияние оптимизатора на локальные геометрические характеристики решения означает необходимость учитывать в теории сложности не только архитектуру модели, но и динамику алгоритма. Для SAM целевая функция имеет характерную мини-макс форму:

$$\min_{\theta} \max_{\|\varepsilon\|_2 \le \rho} \mathcal{L}(\theta + \varepsilon), \tag{9}$$

где ρ задает радиус локального возмущения. Выражение (9) формализует идею избегания острых минимумов в явной оптимизационной постановке и связывает выбор алгоритма с геометрией получаемого решения.

На стыке геометрии и оптимизации проявляется различие между классической и современной постановкой вопроса. В статистической теории обучения алгоритм часто рассматривался как вторичный элемент. Сначала задавался класс гипотез, затем для него выводились общие оценки. В глубоком обучении ситуация принципиально иная. Один и тот же класс гипотез, обученный разными вариантами стохастического градиентного спуска, может приводить к решениям с различной локальной кривизной и различным качеством обобщения. Поэтому шум стохастической оптимизации, размер батча и свойства траектории становятся частью самой теории сложности, а не внешней технической деталью [10, 11].

С этой точки зрения, оптимизатор действует как механизм отбора решений внутри большого перепараметризованного пространства. Даже если класс допустимых функций чрезвычайно богат, реальная траектория SGD не исследует его равномерно. Вместо этого она систематически приводит к решениям с определенными геометрическими характеристиками, чувствительностью к возмущениям и структурой признаков. Следовательно, эффективная сложность оказывается результатом совместного действия архитектуры, данных и алгоритма оптимизации. Современные работы описывают этот механизм на разных уровнях: от диффузионной аппроксимации, в которой SGD экспоненциально предпочитает плоские минимумы [39], до анализа динамической устойчивости, границы устойчивости и детальных оценок устойчивости самого стохастического градиентного спуска [61, 65, 72].

Этот подход связывает локальный анализ гессиана с более глобальными вопросами стабильности обучения. Если решение лежит в области, устойчивой к шуму оптимизации и малым возмущениям параметров, то его свойства могут согласовываться с более высоким обобщением. Тем самым ландшафтный анализ связывает геометрию минимумов и алгоритмическую теорию устойчивости. Такая интерпретация согласуется с работами о фрактальной структуре стохастической оптимизации, где свойства траектории и свойства обобщения рассматриваются совместно [68].

В анализе плоских минимумов необходимо учитывать методологическую проблему. Геометрия в пространстве параметров не всегда напрямую соответствует геометрии в пространстве функций. Для нейронных сетей характерны перенормировки, симметрии слоев и другие преобразования параметров, которые могут заметно менять локальную кривизну без изменения реализуемой функции. Поэтому простая интерпретация вида «более плоский минимум означает более высокое обобщение» требует оговорок. Отсюда вытекают два следствия. Во-первых, ландшафтные меры нужно по возможности строить так, чтобы они были более устой-

чивы к параметризационным артефактам или хотя бы явно учитывали их наличие. Во-вторых, эмпирическая полезность таких мер не отменяет необходимости осторожной причинной интерпретации. Если плоскость зависит от выбора параметризации, то она, вероятно, является не самостоятельной причиной высокого обобщения, а наблюдаемым маркером более глубокой структуры. Практическое направление решения этой проблемы связано с инвариантными или квази-инвариантными мерами. В литературе используются нормализованные варианты остроты минимума, в которых локальная кривизна масштабируется параметрами слоя или нормы весов [57]. Указанные ограничения не отменяют использование ландшафтного подхода, а уточняют границы его применимости. В наиболее строгой постановке речь идет не о тезисе плоских минимумов вообще, а о связи обобщения с устойчивыми геометрическими свойствами решений, достигаемых конкретной динамикой обучения. Такая формулировка согласуется с современными эмпирическими данными и необходимостью учитывать архитектурную специфику.

Основное достоинство ландшафтного подхода состоит в том, что он описывает свойства уже найденного решения. Это позволяет частично компенсировать основной недостаток классических мер сложности — их слабую чувствительность к алгоритму и режиму обучения. Кроме того, геометрические характеристики естественно связываются с такими современными явлениями, как неявная регуляризация, влияние размера батча и различия между оптимизаторами [10, 11]. С другой стороны, у данного направления есть ряд недостатков. Во-первых, вычисление гессиана или даже его устойчивых спектральных характеристик остается ресурсоемкой задачей для больших моделей. Во-вторых, причинная интерпретация плоских минимумов далека от окончательной ясности: решения с высоким обобщением могут быть плоскими не потому, что плоскость сама по себе вызывает обобщение, а потому, что и плоскость, и хорошее обобщение являются следствиями более глубокой структуры данных и оптимизации. В-третьих, локальная геометрия фиксирует свойства конкретной точки, но не заменяет более широкую теорию методов обучения и масштабирования. Поэтому ландшафтный анализ следует понимать не как окончательную замену классических мер сложности, а как компонент более общей теории, в которой учитываются геометрия решения, алгоритм его поиска и структура данных.

5. Эмпирические оценки сложности и сходимости

Эмпирически показано, что простая монотонная связь между сложностью и ошибкой обобщения не выполняется. Это видно в классических работах по двойному спуску [16, 49], а также в последующих уточнениях для различных режимов и моделей [41, 45, 50]. Если в классических мерах сложности рост сложности модели после некоторой точки должен лишь ухудшать обобщение, то в реальности часто наблюдается другое. Сначала ошибка падает, затем возрастает, а затем снова уменьшается. Явление двойного спуска не образует завершенной теории, однако остается эмпирическим фактом. Оно показывает, что сложность в глубоком обучении не сводится к единственной скалярной величине вроде числа параметров или грубой мощности класса гипотез. Причем в разных режимах обучения одна и та же архитектура может функционировать по-разному, а качество обобщения определяется тем, в каком режиме обучения и при какой параметризации она работает. Для описания феномена двойного спуска используется кусочно-заданная форма тестовой ошибки

$$\mathcal{E}_{\text{test}}(m) \approx \begin{cases} \downarrow, & m < m_{\text{int}}, \\ \uparrow, & m \approx m_{\text{int}}, \\ \downarrow, & m > m_{\text{int}}, \end{cases} \tag{10}$$

где m — параметр сложности модели, m_{int} — интерполяционный порог. Запись (10) формализует немонотонность зависимости «сложность–ошибка».

В рамках режимной интерпретации двойного спуска обычно выделяются три фазы. До интерполяционного порога доминирует режим недообученной аппроксимации, и рост сложности чаще снижает смещение. Вблизи порога m_{int} усиливается чувствительность к шуму и нестабильности оптимизации, что приводит к локальному росту тестовой ошибки. После порога в сильно перепараметризованной области начинает доминировать режим селективной оптимизации, в котором алгоритм чаще находит решения с более благоприятной локальной геометрией.

Современная интерпретация этой проблемы часто формулируется в терминах двух режимов обучения. В достаточно широких нейросетевых моделях обучение может быть близко к линеаризированному режиму, где динамика хорошо описывается через нейронное касательное ядро [17, 36, 53]. В этом случае модель ведет себя почти как ядерный метод, а часть классических гарантий становится более применимой. В этом режиме используется линеаризированная динамика предсказаний вида

$$f_{t+1}(x) \approx f_t(x) - \eta\, K_{\theta_0}(x, X)\, \nabla_f \mathcal{L}(f_t(X), y)\,, \tag{11}$$

где K_{θ_0} — NTK в точке инициализации θ_0, η — шаг обучения. Формула (11) задает формальный переход к ядерной аппроксимации в широких сетях.

Однако на практике в моделях глубокого обучения доминирует другой режим — обучение признаков, в котором внутренняя репрезентация данных существенно меняется по ходу оптимизации. Этот режим в большей степени согласуется с современными эмпирическими результатами, но в меньшей степени поддается классическому анализу. Ограничения NTK-приближений и роль эволюции признаков обсуждаются в [35, 55], а также в более поздних работах [54, 76]. В таком режиме эффективная сложность определяется не только архитектурой, но и траекторией перестройки признакового пространства, а значит снова выводит нас к геометрии решения и динамике оптимизации. Режимы обучения служат связующим звеном между классической статистической теорией и ландшафтным анализом. Они позволяют объяснить, почему одна и та же модель может подчиняться различным схемам в зависимости от размера модели, данных и метода оптимизации.

Формально различие между режимами можно зафиксировать через динамику касательного ядра. В NTK-режиме выполняется приближение

$$\|K_{\theta_t} - K_{\theta_0}\| \ll \|K_{\theta_0}\|, \tag{12}$$

поэтому обучение близко к линеаризованной модели с почти фиксированными признаками. В режиме обучения признаков существенной является противоположная ситуация

$$\|K_{\theta_t} - K_{\theta_0}\| = \mathcal{O}(1), \tag{13}$$

и тогда ключевым объектом анализа становится дрейф представлений. Соотношения (12)–(13) задают математический критерий, почему одна и та же архитектура требует разных мер сложности в разных режимах обучения.

Режимный анализ важен не только как теоретическая классификация, но и как средство интерпретации экспериментальных результатов. Когда модель находится ближе к ядерному режиму, ее поведение оказывается сравнительно более предсказуемым, а роль оптимизации сводится к подбору коэффициентов в почти фиксированном признаковом пространстве. Когда же обучение протекает в режиме обучения признаков, ключевым объектом анализа становится эволюция представлений, а следовательно, сложность должна описывать уже не только выходную функцию, но и динамику внутренней репрезентации.

Здесь проявляется причина, по которой одна универсальная мера сложности не получает убедительного теоретического обоснования. Показатель, эффективный в почти линеаризированном режиме, может оказаться малоинформативным в режиме активного обучения признаков. Следовательно, корректнее говорить не о единственной численной сложности, а о семействе взаимодополняющих описаний, чья применимость зависит от фазы обучения и архитектурного режима. Режимы обучения рассматриваются как объяснительный слой между общими теоретическими оценками и практикой больших нейронных сетей.

Еще одно современное направление связано с эмпирическими законами масштабирования. Базовые результаты для языковых моделей даны в [18, 19], а последующие обобщения и поправки обсуждаются в [29, 69]. Эти результаты показывают, что качество моделей часто изменяется по степенным законам при росте числа параметров, объема данных и вычислительного бюджета. Соответствующие зависимости оказываются практически полезными для проектирования больших моделей, однако сами по себе не дают объяснения механизма обобщения. В литературе этот блок рассматривается как отдельный макроуровень описания, связывающий теорию обобщения с инженерной практикой построения больших моделей [43, 44, 48]. В упрощенной форме такие зависимости записываются как

$$\mathcal{L}(N, D, C) \approx \mathcal{L}_\infty + aN^{-\alpha} + bD^{-\beta} + cC^{-\gamma}, \tag{14}$$

где N — число параметров, D — объем данных, C — вычислительный бюджет, $\alpha, \beta, \gamma > 0$ — эмпирические показатели масштабирования. Формула (14) удобна как макроописание, но не задает сама по себе механизма, через который именно эти показатели возникают в разных режимах обучения.

Для трансформеров и диффузионных моделей наблюдается следующая картина. Эти архитектуры продемонстрировали масштабируемость, имеющую прикладное значение [20,21]. В то же время для них отсутствует единая теория сложности, которая одновременно учитывала бы архитектурную специфику, оптимизацию, геометрию решения и режимы обучения. В этой постановке законы масштабирования задают описательный макроуровень, тогда как ландшафтные и режимные подходы ориентированы на объяснительный уровень. В работе [23] масштабирование связывается с анализом кривизны и полных спектральных характеристик гессиана для архитектуры трансформера. Статистико-аппроксимационные и архитектурно-специфические версии представлены в работах [24–26]. Эта траектория исследований соединяет макроэмпирические зависимости с локальной геометрией архитектурно-специфических моделей.

Для трансформеров одна из трудностей состоит в том, что их успешность определяется не только размером модели, но и специфическими индуктивными смещениями механизма внимания, глубиной сети, структурой токенизации и масштабом предварительного обучения. Поэтому перенос классических оценок обучаемости в этот контекст редко дает содержательное объяснение. На уровне практики законы масштабирования помогают выбирать баланс между параметрами, данными и вычислениями, но дают ограниченную информацию о том, почему архитектура механизма внимания оказывается столь эффективной в режиме обучения признаков.

Для большей детализации полезно разделять архитектурные механизмы по их влиянию на сложность. Для трансформеров это, прежде всего, механизм самовнимания, остаточные связи и нормализация слоев: вместе они стабилизируют оптимизацию глубоких стеков и меняют эффективную кривизну задачи обучения. На уровне анализа сложности это означает, что параметр N в законах масштабирования (14) агрегирует разные по роли степени свободы. Практически эквивалентный рост числа параметров может сопровождаться разной динамикой обобщения в зависимости от того, где добавляется емкость: в ширине блоков внимания, в числе слоев, в MLP-части или в словаре токенизации. Этот вывод согласуется с работами, где законы масштабирования выводятся из статистико-аппроксимационного анализа и из динамики обучения признаков [24, 25, 27].

Для диффузионных моделей проблема имеет иную структуру. Здесь качество зависит не только от выразительности аппроксимирующей сети, но и от самой многошаговой стохастической процедуры генерации из шума. Следовательно, сложность должна описывать не статическую функцию, а целую динамическую схему восстановления сигнала. Это демонстрирует ограниченность единой скалярной меры сложности. Даже если параметризация сети фиксирована, изменение режима шума, числа шагов и процедуры обучения способно существенно менять как стабильность, так и качество генерации [21].

Для диффузионных архитектур это особенно заметно: эффективная сложность определяется совместно параметризацией денойзера и расписанием шума. Поэтому даже при близких кривых законов масштабирования финальная генеративная способность может различаться из-за разных траекторий обучения по временным шагам и разной чувствительности к ошибкам на ранних и поздних уровнях шума [28].

Связь геометрического и масштабного уровней удобно описывать как трехуровневую цепочку: локальные метрики формируют режим обучения, а режим обучения определяет вид наблюдаемой кривой закона масштаби-

рования при росте N, D, C. Ключевое методологическое следствие состоит в том, что одинаковая макрокривая качества не гарантирует одинаковый механизм обобщения.

Из этого следует вывод, что теория сложности глубокого обучения не может ограничиться ни одним из этих методов по отдельности. Классические оценки задают базовый язык, ландшафтные меры описывают локальные геометрические характеристики решений, анализ режимов обучения объясняет качественные переходы между динамическими фазами, а законы масштабирования фиксируют крупномасштабные эмпирические закономерности.

6. Сравнительный анализ и открытые вопросы

Рассмотренные работы в табл. 2 дают сравнительную картину современных подходов к сложности в глубоком обучении. Ключевой вывод состоит в том, что разные меры сложности отвечают на разные исследовательские вопросы. Классические меры описывают вместимость класса гипотез. Вероятностные и информационные методы частично учитывают данные и алгоритм. Геометрические и ландшафтные методы описывают свойства найденного решения. Режимные и масштабные подходы фиксируют качественные переходы и крупномасштабные закономерности.

Видно, что сохраняется разрыв между глобальной сложностью класса моделей и эффективной сложностью реально найденного решения. Первая необходима для формальной теории обучаемости, однако в глубоких сетях она недостаточна для объяснения практического обобщения. Вторая пока не имеет общепринятого формального описания, при этом современные геометрические и алгоритмические подходы в основном ориентированы на ее анализ. Классическая теория отвечает на вопрос о том, когда обучение в принципе возможно, тогда как современная практика требует ответа на вопрос почему из множества допустимых решений обучение систематически приводит к подмножеству решений с благоприятными эмпирическими свойствами. Если первый вопрос относится к глобальной структуре класса гипотез, то второй — к селективности алгоритма и геометрии достижимых минимумов. Поэтому современные исследования движутся к многокомпонентному описанию сложности, а не к поиску единственной универсальной формулы.

Современная теория должна быть не только формально строгой, но и эмпирически релевантной. Иначе говоря, качество меры сложности сегодня определяется не только ее математической строгостью, но и тем, насколько она позволяет согласовать классические гарантии с реальными

Таблица 2. Сравнение подходов к анализу сложности моделей глубокого обучения

Подход	Что измеряет	Сильные стороны	Ограничения для глубокого обучения
VC- и PAC-подходы	Мощность класса гипотез в худшем случае	Строгие критерии обучаемости и ясный статистический язык	Классическая оценка сложности через мощность класса функций и гарантии обучаемости в вероятностной постановке; пессимистичны в перепараметризованных режимах, не учитывают алгоритм и найденное решение
Радемахеровские и нормовые оценки	Емкость класса функций, зависящая от данных	Более тонкие оценки сверху, чувствительность к выборке и нормам	Уточняют обобщение за счет геометрии выборки и ограничений на параметры; по-прежнему описывают весь класс, а не подмножество реально достижимых решений
PAC-байесовские и информационные границы	Связь обобщения с распределением параметров и информацией об обучении	Учитывают алгоритмический аспект и допускают адаптивный анализ	Формируют вероятностно-информационную рамку, связывающую обобщение с распределением параметров и информацией из данных; зависят от выбора априорных распределений, параметризации и часто остаются численно слабыми
Ландшафтные и гессиановые меры	Локальные геометрические характеристики решения	Отражают свойства минимумов, устойчивость и связь с оптимизацией	Геометрический анализ свойств найденного решения и его устойчивости к возмущениям; дороги вычислительно, причинная интерпретация плоскости и спектра не всегда однозначна
Режимы обучения и законы масштабирования	Качественные фазы обучения и крупномасштабные эмпирические зависимости	Описывают современную практику больших моделей	Описывают фазы обучения и закономерности качества при росте параметров, данных и вычислений; дают неполное объяснение механизма обобщения и требуют стыковки с геометрией и оптимизацией

режимами обучения больших сетей. Ландшафт и оптимизация в этом случае выступают как аналитические компоненты, так как позволяют анализировать сложность не только теоретически, но и применительно к решениям, возникающим в современных вычислительных режимах.

Таблица 3. Уровни объяснения и типы теоретических утверждений

Уровень объяснения	Тип утверждений	Типичный вопрос	Основной риск неверной интерпретации
Глобальный, статистический	Гарантии обучаемости, оценки сверху на ошибку	Когда обучение в принципе возможно?	Перенос оценок худшего случая на конкретную обученную модель без учета алгоритма
Алгоритмический	Устойчивость, неявная регуляризация, селективность траектории	Почему из многих решений выбираются именно эти?	Подмена причинного механизма постфактум корреляциями
Геометрический	Спектральные и ландшафтные характеристики найденного минимума	Какие свойства решения связаны с обобщением?	Игнорирование параметризационных инвариантностей и вычислительной чувствительности метрик
Эмпирический	Режимные переходы и законы масштабирования	Как меняется качество при росте ресурсов?	Объяснение механизма только по форме масштабной кривой

В табл. 3 представлены ключевые различия между различными направлениями исследований. Статистические оценки нередко интерпретируются так, будто они обязаны предсказывать локальную геометрию решения, а ландшафтные наблюдения интерпретируются так, как будто они автоматически дают универсальные гарантии обучаемости. Методологически корректно рассматривать эти уровни как взаимодополняющие. С этой точки зрения содержательный вклад обзора состоит не только в перечислении направлений, но и в реконструкции их совместимой архитектуры. Классические оценки ограничивают область допустимых объяснений, алгоритмические и геометрические подходы описывают механизм селекции решений внутри этой области, режимный и масштабный анализ тестируют устойчивость механизма при изменении масштаба моделей, данных и вычислений. Такая многоуровневая схема дает более строгий язык для формулировки открытых задач.

Из этого анализа выделяется четыре вывода. Во-первых, классические меры, основанные на емкости класса, не исчезают из современной теории, но они дают базовые ориентиры, а не окончательные оценки. Во-вторых, алгоритм обучения в глубоких сетях является частью самой сложности. В-третьих, геометрические свойства найденного решения сопоставимы

по роли со свойствами класса допустимых функций. В-четвертых, крупномасштабные эмпирические закономерности, включая двойной спуск и законы масштабирования, выступают не альтернативой строгой теории, а ее тестом. Современные обзорные и синтетические работы подтверждают многоуровневую картину, в которой неявная регуляризация, геометрия ландшафта и режимы масштабирования рассматриваются как взаимодополняющие части одного исследовательского тренда. Это видно в обзорных публикациях [43, 44, 48] и в работах, ориентированных на теоретический анализ [52, 70, 73].

Исходя из анализа, выделяются следующие направления исследований теории сложности моделей машинного обучения.

1. Эффективная сложность достижимых решений. Требуется теория, описывающая не только весь класс параметрически допустимых функций, но и подмножество решений, которое достигается обучением из заданных инициализаций при заданных алгоритмах. На этом уровне соединяются язык классической емкости класса и язык геометрии найденных минимумов.

2. Совместное описание оптимизации и ландшафта. Ландшафтные меры рассматриваются после обучения, а теория оптимизации — отдельно. Для глубокого обучения этого уже недостаточно. Требуется теория, в которой свойства спектра, плоскости минимумов, размер батча, шум и выбор оптимизатора описываются как части единого механизма формирования обобщающего решения.

3. Теория для современных больших архитектур. Трансформеры, диффузионные модели и другие современные архитектуры демонстрируют устойчивые эмпирические закономерности, однако соответствующая теория сложности остается фрагментарной. Особенно важна задача согласовать законы масштабирования с более локальными геометрическими и динамическими характеристиками.

4. Согласование уровней объяснения и критериев верификации. Открытой проблемой остается построение единого метода сопоставления результатов разных типов. От статистических оценок до геометрических и режимных индикаторов. Для прогресса необходимы работы, в которых для одной и той же постановки задачи одновременно фиксируются глобальные оценочные характеристики, алгоритмические показатели

устойчивости, ландшафтные метрики и эмпирические зависимости качества.

5. Переход от ретроспективного объяснения к прогностической теории. Значительная часть современных методов успешно объясняет уже наблюденные эффекты, но в меньшей степени дает проверяемые прогнозы до проведения полного вычислительного эксперимента. Соответственно, одним из направлений дальнейших исследований является разработка предсказательных критериев, позволяющих на ранних этапах обучения оценивать вероятный класс обобщающего решения и ожидаемую чувствительность к изменению масштаба. Это согласуется с результатами, где динамика спектра гессиана в раннем и среднем обучении коррелирует с последующей устойчивостью решения [58, 60]. Она также согласуется с работами, связывающими стабильность траектории SGD и итоговое обобщение [61, 65].

Остается открытым вопрос о форме будущей объединяющей теории. Вероятно, она не будет одной формулой или одной универсальной оценкой. Более обоснованным представляется многоуровневый подход: на верхнем уровне — укрупненные законы масштабирования и фазовые переходы, на среднем — режимы обучения и роль оптимизатора, на локальном — геометрия минимумов, спектр гессиана и устойчивость. Такое расслоение в большей степени соответствует наблюдаемой практике глубокого обучения, чем попытка свести все эффекты к одной численной характеристике. Эти проблемы показывают, что тренд развития теории сложности в глубоком обучении лежит не в отказе от классических результатов, а в их переинтерпретации и расширении. Классическая теория остается необходимым основанием, но ее нужно дополнять анализом эффективной геометрии решений и алгоритмических режимов.

7. Заключение

Теория сложности моделей глубокого обучения смещается от анализа абстрактной мощности класса гипотез к анализу эффективной сложности реально достигаемых решений. Статья опирается на воспроизводимую процедуру построения корпуса публикаций и сопоставление классических, вероятностно-информационных, ландшафтных и режимно-масштабных линий исследований. Классические меры — VC-размерность, радемахеровские оценки, PAC-байесовские и информационные границы — остаются фундаментальной частью языка теории обучения, однако в перепараметризованных режимах уже не дают единственного объяснения обобщающей

способности. Современные подходы описывают сложность через совместный анализ геометрии ландшафта, динамики оптимизации, режимов обучения и масштабных закономерностей. Ключевой вывод состоит в том, что архитектура, данные и алгоритм обучения должны анализироваться совместно. Только в этом случае теория сохраняет и формальную строгость, и эмпирическую релевантность для современных больших моделей.

Список литературы

[1] V. N. Vapnik, A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities", *Theory of Probability and its Applications*, **16**:2 (1971), 264–280. DOI: 10.1137/1116025.

[2] L. G. Valiant, "A theory of the learnable", *Communications of the ACM*, **27**:11 (1984), 1134–1142. DOI: 10.1145/1968.1972.

[3] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.

[4] К. В. Воронцов, "Комбинаторный подход к оценке качества обучаемых алгоритмов", *Математические вопросы кибернетики*, **13** (2004), 5–51.

[5] P. L. Bartlett, S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results", *Journal of Machine Learning Research*, **3** (2002), 463–482.

[6] V. Koltchinskii, "Rademacher penalties and structural risk minimization", *IEEE Transactions on Information Theory*, **47**:5 (2001), 1902–1914. DOI: 10.1109/18.930926.

[7] D. A. McAllester, "Some PAC-Bayesian theorems", *Machine Learning*, **37**:3 (1999), 355–363. DOI: 10.1023/A:1007618624809.

[8] G. K. Dziugaite, D. M. Roy, "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data", *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*, 2017.

[9] A. Xu, M. Raginsky, "Information-theoretic analysis of generalization capability of learning algorithms", *Advances in Neural Information Processing Systems*, **30**, 2017.

[10] O. Bousquet, A. Elisseeff, "Stability and generalization", *Journal of Machine Learning Research*, **2** (2002), 499–526.

[11] M. Hardt, B. Recht, Y. Singer, "Train faster, generalize better: stability of stochastic gradient descent", *Proceedings of the 33rd International Conference on Machine Learning*, Proceedings of Machine Learning Research, **48**, 2016, 1225–1234.

[12] H. Li, Z. Xu, G. Taylor, C. Studer, T. Goldstein, "Visualizing the Loss Landscape of Neural Nets", *Advances in Neural Information Processing Systems*, **31**, 2018, 6389–6399.

[13] S. Hochreiter, J. Schmidhuber, "Flat minima", *Neural Computation*, **9**:1 (1997), 1–42. DOI: 10.1162/neco.1997.9.1.1.

[14] L. Sagun, L. Bottou, Y. LeCun, "Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond", arXiv: 1611.07476.

[15] N. S. Kiselev, A. V. Grabovoy, “Unraveling the Hessian: A Key to Smooth Convergence in Loss Function Landscapes”, *Doklady Mathematics*, **110**:Suppl. 1 (2024), S49–S61. DOI: 10.1134/S1064562424601987.

[16] M. Belkin, D. Hsu, S. Ma, S. Mandal, “Reconciling modern machine-learning practice and the classical bias–variance trade-off”, *Proceedings of the National Academy of Sciences*, **116**:32 (2019), 15849–15854. DOI: 10.1073/pnas.1903070116.

[17] A. Jacot, F. Gabriel, C. Hongler, “Neural tangent kernel: Convergence and generalization in neural networks”, *Advances in Neural Information Processing Systems*, **31**, 2018.

[18] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, “Scaling laws for neural language models”, arXiv: 2001.08361.

[19] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, O. Vinyals, J. W. Rae, L. Sifre, “Training Compute-Optimal Large Language Models”, *Advances in Neural Information Processing Systems*, **35**, 2022, 30016–30030. DOI: 10.52202/068431-2176.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, “Attention is all you need”, *Advances in Neural Information Processing Systems*, **30**, 2017, 5998–6008.

[21] J. Ho, A. Jain, P. Abbeel, “Denoising diffusion probabilistic models”, *Advances in Neural Information Processing Systems*, **33**, 2020, 6840–6851.

[22] F. Hellström, G. Durisi, B. Guedj, M. Raginsky, “Generalization Bounds: Perspectives from Information Theory and PAC-Bayes”, *Foundations and Trends in Machine Learning*, **18**:1 (2025), 1–223. DOI: 10.1561/2200000112.

[23] E. Petrov, N. Kiselev, V. Meshkov, A. Grabovoy, “Closing the Curvature Gap: Full Transformer Hessians and Their Implications for Scaling Laws”, arXiv: 2504.12516.

[24] A. Havrilla, W. Liao, “Understanding Scaling Laws with Statistical and Approximation Theory for Transformer Neural Networks on Intrinsically Low-dimensional Data”, arXiv: 2411.06646.

[25] M. K. Pasupuleti, “Scaling Laws for Transformers on Low-Dimensional Data: A Statistical and Approximation Theory Perspective”, *International Journal of Academic and Industrial Research Innovations*, **2**:1 (2022), 191–202. DOI: 10.62311/nesx/rp-5-01-2022.

[26] M. Mayaki, “Generalization and Scaling Laws for Mixture-of-Experts Transformers”, arXiv: 2604.09175.

[27] Z. Yao, R. Wu, T. Gao, “Understanding Scaling Laws in Deep Neural Networks via Feature Learning Dynamics”, arXiv: 2512.21075.

[28] Z. Liang, H. He, C. Yang, B. Dai, “Scaling Laws For Diffusion Transformers”, arXiv: 2410.08184.

[29] N. Sardana, J. Portes, S. Doubov, J. Frankle, “Beyond Chinchilla-Optimal: Accounting for Inference in Language Model Scaling Laws”, arXiv: 2401.00448.

[30] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, “Understanding deep learning (still) requires rethinking generalization”, *Communications of the ACM*, **64**:3 (2021), 107–115. DOI: 10.1145/3446776.

[31] L. V. Truong, “On Rademacher Complexity-based Generalization Bounds for Deep Learning”, arXiv: 2208.04284.

[32] S. Sachs, T. van Erven, L. Hodgkinson, R. Khanna, U. Şimşekli, “Generalization Guarantees via Algorithm-dependent Rademacher Complexity”, arXiv: 2307.02501.

[33] W. Zhou, V. Veitch, M. Austern, R. Adams, P. Orbanz, “Non-Vacuous Generalization Bounds at the ImageNet Scale: A PAC-Bayesian Compression Approach”, arXiv: 1804.05862.

[34] L. Hodgkinson, C. Heide, R. Salomone, F. Roosta, M. Mahoney, “A PAC-Bayesian Perspective on the Interpolating Information Criterion”, arXiv: 2311.07013.

[35] S. Arora, S. Du, W. Hu, Z. Li, R. Wang, “Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks”, *Proceedings of the 36th International Conference on Machine Learning*, 2019, 322–332.

[36] B. Adlam, J. Pennington, “The Neural Tangent Kernel in High Dimensions: Triple Descent and a Multi-Scale Theory of Generalization”, arXiv: 2008.06786.

[37] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P. Tang, “On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima”, arXiv: 1609.04836.

[38] P. Foret, A. Kleiner, H. Mobahi, B. Neyshabur, “Sharpness-Aware Minimization for Efficiently Improving Generalization”, *International Conference on Learning Representations*, 2021, arXiv: 2010.01412.

[39] Z. Xie, I. Sato, M. Sugiyama, “A Diffusion Theory For Deep Learning Dynamics: Stochastic Gradient Descent Exponentially Favors Flat Minima”, arXiv: 2002.03495.

[40] L. Wu, Z. Zhu, W. E, “Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes”, arXiv: 1706.10239.

[41] S. Mei, A. Montanari, “The generalization error of random features regression: Precise asymptotics and double descent curve”, arXiv: 1908.05355.

[42] G. Valle-Perez, A. Louis, “Generalization bounds for deep learning”, arXiv: 2012.04115.

[43] X. Hu, L. Chu, J. Pei, W. Liu, J. Bian, “Model Complexity of Deep Learning: A Survey”, arXiv: 2103.05127.

[44] F. He, D. Tao, “Recent advances in deep learning theory”, arXiv: 2012.10931.

[45] M. Advani, A. Saxe, H. Sompolinsky, “High-dimensional dynamics of generalization error in neural networks”, *Neural Networks*, **132** (2020), 428–446. DOI: 10.1016/j.neunet.2020.08.022.

[46] V. Cherkassky, X. Shao, F. Mulier, V. Vapnik, “Model complexity control for regression using VC generalization bounds”, *IEEE Transactions on Neural Networks*, **10**:5 (1999), 1075–1089. DOI: 10.1109/72.788648.

[47] A. Hole, “Vapnik-Chervonenkis Generalization Bounds for Real Valued Neural Networks”, *Neural Computation*, **8**:6 (1996), 1277–1299. DOI: 10.1162/neco.1996.8.6.1277.

[48] Y. Dar, V. Muthukumar, R. Baraniuk, “A Farewell to the Bias-Variance Tradeoff? An Overview of the Theory of Overparameterized Machine Learning”, arXiv: 2109.02355.

[49] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, I. Sutskever, "Deep double descent: Where bigger models and more data hurt", *Journal of Statistical Mechanics: Theory and Experiment*, **2021**:12 (2021), 124003. DOI: 10.1088/1742-5468/ac3a74.

[50] E. H. Lee, V. Cherkassky, "Understanding Double Descent Using VC-Theoretical Framework", *IEEE Transactions on Neural Networks and Learning Systems*, **35**:12 (2024), 18838–18847. DOI: 10.1109/TNNLS.2024.3388873.

[51] B. Ghojogh, A. Ghodsi, *PAC Learnability and Information Bottleneck in Deep Learning: Tutorial and Survey*, HAL preprint hal-04705310, 2024.

[52] Z. Johannsen, "Implicit Regularization and Generalization in Overparameterized Neural Networks", arXiv: 2604.07603.

[53] S. Vakili, M. Bromberg, J. Garcia, D.-s. Shiu, A. Bernacchia, "Uniform Generalization Bounds for Overparameterized Neural Networks", arXiv: 2109.06099.

[54] G. Bachmann, T. Hofmann, A. Lucchi, "Generalization Through The Lens Of Leave-One-Out Error", arXiv: 2203.03443.

[55] T. Suzuki, S. Akiyama, "Benefit of deep learning with non-convex noisy gradient descent: Provable excess risk bound and superiority to kernel methods", arXiv: 2012.03224.

[56] K. Sakamoto, I. Sato, "Analyzing Lottery Ticket Hypothesis from PAC-Bayesian Theory Perspective", arXiv: 2205.07320.

[57] Y. Tsuzuku, I. Sato, M. Sugiyama, "Normalized Flat Minima: Exploring Scale Invariant Definition of Flat Minima for Neural Networks using PAC-Bayesian Analysis", arXiv: 1901.04653.

[58] M. Wei, D. Schwab, "How noise affects the Hessian spectrum in overparameterized neural networks", arXiv: 1910.00195.

[59] P. Long, P. Bartlett, "Sharpness-Aware Minimization and the Edge of Stability", arXiv: 2309.12488.

[60] Z. Xie, Q.-Y. Tang, Y. Cai, M. Sun, P. Li, "On the Power-Law Hessian Spectrums in Deep Learning", arXiv: 2201.13011.

[61] L. Wu, W. Su, "The Implicit Regularization of Dynamical Stability in Stochastic Gradient Descent", arXiv: 2305.17490.

[62] P. Zhou, J. Feng, "The Landscape of Deep Learning Algorithms", arXiv: 1705.07038.

[63] P. Natekar, M. Sharma, "Representation Based Complexity Measures for Predicting Generalization in Deep Learning", arXiv: 2012.02775.

[64] G. Valle-Perez, C. Camargo, A. Louis, "Deep learning generalizes because the parameter-function map is biased towards simple functions", arXiv: 1805.08522.

[65] Y. Lei, Y. Ying, "Fine-Grained Analysis of Stability and Generalization for Stochastic Gradient Descent", arXiv: 2006.08157.

[66] B. London, "A PAC-Bayesian Analysis of Randomized Learning with Application to Stochastic Gradient Descent", *Advances in Neural Information Processing Systems*, **30**, 2017.

[67] D. Bisla, J. Wang, A. Choromanska, "Low-Pass Filtering SGD for Recovering Flat Optima in the Deep Learning Optimization Landscape", arXiv: 2201.08025.

[68] A. Camuto, G. Deligiannidis, M. Erdogdu, M. Gurbuzbalaban, U. Simsekli, L. Zhu, "Fractal Structure and Generalization Properties of Stochastic Optimization Algorithms", arXiv: 2106.04881.

[69] J. Wang, A. Choromanska, "A Survey of Optimization Methods for Training DL Models: Theoretical Perspective on Convergence and Generalization", arXiv: 2501.14458.

[70] V. Nagarajan, "Explaining generalization in deep learning: progress and fundamental limits", arXiv: 2110.08922.

[71] K. Kawaguchi, L. Kaelbling, Y. Bengio, "Generalization in Deep Learning", arXiv: 1710.05468.

[72] S. Arora, Z. Li, A. Panigrahi, "Understanding Gradient Descent on Edge of Stability in Deep Learning", arXiv: 2205.09745.

[73] T. Poggio, A. Banburski, Q. Liao, "Theoretical Issues in Deep Networks: Approximation, Optimization and Generalization", arXiv: 1908.09375.

[74] T. Poggio, K. Kawaguchi, Q. Liao, B. Miranda, L. Rosasco, X. Boix, J. Hidary, H. Mhaskar, "Theory of Deep Learning III: explaining the non-overfitting puzzle", arXiv: 1801.00173.

[75] N. Golowich, A. Rakhlin, O. Shamir, "Size-Independent Sample Complexity of Neural Networks", *Information and Inference: A Journal of the IMA*, **9**:2 (2020), 473–504. DOI: 10.1093/imaiai/iaz007.

[76] N. Vyas, Y. Bansal, P. Nakkiran, "Limitations of the NTK for Understanding Generalization in Deep Learning", arXiv: 2206.10012 .

Статья поступила 5 мая 2026 г.

From VC Dimension to Scaling Laws: The Evolution of Theoretical Approaches to Complexity and Generalization

A. V. Grabovoy

This paper presents a review of the current state of research on theoretical approaches to complexity in deep learning and their relation to generalization. Methodologically, the review is structured as a scoping review: it documents bibliographic sources, deduplication and semi-automated corpus narrowing, expert screening, and the composition of a final analytical publication set. The central question is why classical complexity measures, despite their foundational role in statistical learning theory, are insufficient for explaining the behavior of modern overparameterized neural networks. VC dimension, Rademacher complexity, PAC-Bayesian bounds, and information-theoretic approaches remain indispensable as a baseline analytical language, yet they mainly characterize worst-case hypothesis-class capacity and only partially reflect the geometry of the learned solution, the optimization trajectory, and architectural inductive biases. The main focus is therefore placed on contemporary approaches based on the loss landscape, Hessian spectra, optimization methods, and empirical scaling laws. The review develops a synthetic interpretation of the field according to which a useful theory of complexity for deep learning should describe not only the expressive

capacity of a class, but also the effective complexity of the solutions actually reached by concrete algorithms on concrete data.

Keywords: complexity theory, generalization, deep learning, VC dimension, Rademacher complexity, loss landscape, Hessian, double descent, scaling laws.

References

[1] V. N. Vapnik, A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities", *Theory of Probability and its Applications*, **16**:2 (1971), 264–280. DOI: 10.1137/1116025.

[2] L. G. Valiant, "A theory of the learnable", *Communications of the ACM*, **27**:11 (1984), 1134–1142. DOI: 10.1145/1968.1972.

[3] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.

[4] K. V. Vorontsov, "A combinatorial approach to assessing the quality of learning algorithms", *Mathematical Questions of Cybernetics*, **13** (2004), 5–51 (In Russian).

[5] P. L. Bartlett, S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results", *Journal of Machine Learning Research*, **3** (2002), 463–482.

[6] V. Koltchinskii, "Rademacher penalties and structural risk minimization", *IEEE Transactions on Information Theory*, **47**:5 (2001), 1902–1914. DOI: 10.1109/18.930926.

[7] D. A. McAllester, "Some PAC-Bayesian theorems", *Machine Learning*, **37**:3 (1999), 355–363. DOI: 10.1023/A:1007618624809.

[8] G. K. Dziugaite, D. M. Roy, "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data", *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*, 2017.

[9] A. Xu, M. Raginsky, "Information-theoretic analysis of generalization capability of learning algorithms", *Advances in Neural Information Processing Systems*, **30**, 2017.

[10] O. Bousquet, A. Elisseeff, "Stability and generalization", *Journal of Machine Learning Research*, **2** (2002), 499–526.

[11] M. Hardt, B. Recht, Y. Singer, "Train faster, generalize better: stability of stochastic gradient descent", *Proceedings of the 33rd International Conference on Machine Learning*, Proceedings of Machine Learning Research, **48**, 2016, 1225–1234.

[12] H. Li, Z. Xu, G. Taylor, C. Studer, T. Goldstein, “Visualizing the Loss Landscape of Neural Nets”, *Advances in Neural Information Processing Systems*, **31**, 2018, 6389–6399.

[13] S. Hochreiter, J. Schmidhuber, “Flat minima”, *Neural Computation*, **9**:1 (1997), 1–42. DOI: 10.1162/neco.1997.9.1.1.

[14] L. Sagun, L. Bottou, Y. LeCun, “Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond”, arXiv: 1611.07476.

[15] N. S. Kiselev, A. V. Grabovoy, “Unraveling the Hessian: A Key to Smooth Convergence in Loss Function Landscapes”, *Doklady Mathematics*, **110**:Suppl. 1 (2024), S49–S61. DOI: 10.1134/S1064562424601987.

[16] M. Belkin, D. Hsu, S. Ma, S. Mandal, “Reconciling modern machine-learning practice and the classical bias–variance trade-off”, *Proceedings of the National Academy of Sciences*, **116**:32 (2019), 15849–15854. DOI: 10.1073/pnas.1903070116.

[17] A. Jacot, F. Gabriel, C. Hongler, “Neural tangent kernel: Convergence and generalization in neural networks”, *Advances in Neural Information Processing Systems*, **31**, 2018.

[18] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, “Scaling laws for neural language models”, arXiv: 2001.08361.

[19] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, O. Vinyals, J. W. Rae, L. Sifre, “Training Compute-Optimal Large Language Models”, *Advances in Neural Information Processing Systems*, **35**, 2022, 30016–30030. DOI: 10.52202/068431-2176.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, “Attention is all you need”, *Advances in Neural Information Processing Systems*, **30**, 2017, 5998–6008.

[21] J. Ho, A. Jain, P. Abbeel, “Denoising diffusion probabilistic models”, *Advances in Neural Information Processing Systems*, **33**, 2020, 6840–6851.

[22] F. Hellström, G. Durisi, B. Guedj, M. Raginsky, “Generalization Bounds: Perspectives from Information Theory and PAC-Bayes”, *Foundations and Trends in Machine Learning*, **18**:1 (2025), 1–223. DOI: 10.1561/2200000112.

[23] E. Petrov, N. Kiselev, V. Meshkov, A. Grabovoy, “Closing the Curvature Gap: Full Transformer Hessians and Their Implications for Scaling Laws”, arXiv: 2504.12516.

[24] A. Havrilla, W. Liao, “Understanding Scaling Laws with Statistical and Approximation Theory for Transformer Neural Networks on Intrinsically Low-dimensional Data”, arXiv: 2411.06646.

[25] M. K. Pasupuleti, “Scaling Laws for Transformers on Low-Dimensional Data: A Statistical and Approximation Theory Perspective”, *International Journal of Academic and Industrial Research Innovations*, **2**:1 (2022), 191–202. DOI: 10.62311/nesx/rp-5-01-2022.

[26] M. Mayaki, “Generalization and Scaling Laws for Mixture-of-Experts Transformers”, arXiv: 2604.09175.

[27] Z. Yao, R. Wu, T. Gao, “Understanding Scaling Laws in Deep Neural Networks via Feature Learning Dynamics”, arXiv: 2512.21075.

[28] Z. Liang, H. He, C. Yang, B. Dai, “Scaling Laws For Diffusion Transformers”, arXiv: 2410.08184.

[29] N. Sardana, J. Portes, S. Doubov, J. Frankle, “Beyond Chinchilla-Optimal: Accounting for Inference in Language Model Scaling Laws”, arXiv: 2401.00448.

[30] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, “Understanding deep learning (still) requires rethinking generalization”, *Communications of the ACM*, **64**:3 (2021), 107–115. DOI: 10.1145/3446776.

[31] L. V. Truong, “On Rademacher Complexity-based Generalization Bounds for Deep Learning”, arXiv: 2208.04284.

[32] S. Sachs, T. van Erven, L. Hodgkinson, R. Khanna, U. Şimşekli, “Generalization Guarantees via Algorithm-dependent Rademacher Complexity”, arXiv: 2307.02501.

[33] W. Zhou, V. Veitch, M. Austern, R. Adams, P. Orbanz, “Non-Vacuous Generalization Bounds at the ImageNet Scale: A PAC-Bayesian Compression Approach”, arXiv: 1804.05862.

[34] L. Hodgkinson, C. Heide, R. Salomone, F. Roosta, M. Mahoney, “A PAC-Bayesian Perspective on the Interpolating Information Criterion”, arXiv: 2311.07013.

[35] S. Arora, S. Du, W. Hu, Z. Li, R. Wang, “Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks”, *Proceedings of the 36th International Conference on Machine Learning*, 2019, 322–332.

[36] B. Adlam, J. Pennington, “The Neural Tangent Kernel in High Dimensions: Triple Descent and a Multi-Scale Theory of Generalization”, arXiv: 2008.06786.

[37] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P. Tang, “On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima”, arXiv: 1609.04836.

[38] P. Foret, A. Kleiner, H. Mobahi, B. Neyshabur, “Sharpness-Aware Minimization for Efficiently Improving Generalization”, *International Conference on Learning Representations*, 2021, arXiv: 2010.01412.

[39] Z. Xie, I. Sato, M. Sugiyama, “A Diffusion Theory For Deep Learning Dynamics: Stochastic Gradient Descent Exponentially Favors Flat Minima”, arXiv: 2002.03495.

[40] L. Wu, Z. Zhu, W. E, “Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes”, arXiv: 1706.10239.

[41] S. Mei, A. Montanari, “The generalization error of random features regression: Precise asymptotics and double descent curve”, arXiv: 1908.05355.

[42] G. Valle-Perez, A. Louis, “Generalization bounds for deep learning”, arXiv: 2012.04115.

[43] X. Hu, L. Chu, J. Pei, W. Liu, J. Bian, “Model Complexity of Deep Learning: A Survey”, arXiv: 2103.05127.

[44] F. He, D. Tao, “Recent advances in deep learning theory”, arXiv: 2012.10931.

[45] M. Advani, A. Saxe, H. Sompolinsky, “High-dimensional dynamics of generalization error in neural networks”, *Neural Networks*, **132** (2020), 428–446. DOI: 10.1016/j.neunet.2020.08.022.

[46] V. Cherkassky, X. Shao, F. Mulier, V. Vapnik, “Model complexity control for regression using VC generalization bounds”, *IEEE Transactions on Neural Networks*, **10**:5 (1999), 1075–1089. DOI: 10.1109/72.788648.

[47] A. Hole, “Vapnik-Chervonenkis Generalization Bounds for Real Valued Neural Networks”, *Neural Computation*, **8**:6 (1996), 1277–1299. DOI: 10.1162/neco.1996.8.6.1277.

[48] Y. Dar, V. Muthukumar, R. Baraniuk, “A Farewell to the Bias-Variance Tradeoff? An Overview of the Theory of Overparameterized Machine Learning”, arXiv: 2109.02355.

[49] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, I. Sutskever, “Deep double descent: Where bigger models and more data hurt”, *Journal of Statistical Mechanics: Theory and Experiment*, **2021**:12 (2021), 124003. DOI: 10.1088/1742-5468/ac3a74.

[50] E. H. Lee, V. Cherkassky, “Understanding Double Descent Using VC-Theoretical Framework”, *IEEE Transactions on Neural Networks and Learning Systems*, **35**:12 (2024), 18838–18847. DOI: 10.1109/TNNLS.2024.3388873.

[51] B. Ghojogh, A. Ghodsi, *PAC Learnability and Information Bottleneck in Deep Learning: Tutorial and Survey*, HAL preprint hal-04705310, 2024.

[52] Z. Johannsen, “Implicit Regularization and Generalization in Overparameterized Neural Networks”, arXiv: 2604.07603.

[53] S. Vakili, M. Bromberg, J. Garcia, D.-s. Shiu, A. Bernacchia, “Uniform Generalization Bounds for Overparameterized Neural Networks”, arXiv: 2109.06099.

[54] G. Bachmann, T. Hofmann, A. Lucchi, “Generalization Through The Lens Of Leave-One-Out Error”, arXiv: 2203.03443.

[55] T. Suzuki, S. Akiyama, “Benefit of deep learning with non-convex noisy gradient descent: Provable excess risk bound and superiority to kernel methods”, arXiv: 2012.03224.

[56] K. Sakamoto, I. Sato, "Analyzing Lottery Ticket Hypothesis from PAC-Bayesian Theory Perspective", arXiv: 2205.07320.

[57] Y. Tsuzuku, I. Sato, M. Sugiyama, "Normalized Flat Minima: Exploring Scale Invariant Definition of Flat Minima for Neural Networks using PAC-Bayesian Analysis", arXiv: 1901.04653.

[58] M. Wei, D. Schwab, "How noise affects the Hessian spectrum in overparameterized neural networks", arXiv: 1910.00195.

[59] P. Long, P. Bartlett, "Sharpness-Aware Minimization and the Edge of Stability", arXiv: 2309.12488.

[60] Z. Xie, Q.-Y. Tang, Y. Cai, M. Sun, P. Li, "On the Power-Law Hessian Spectrums in Deep Learning", arXiv: 2201.13011.

[61] L. Wu, W. Su, "The Implicit Regularization of Dynamical Stability in Stochastic Gradient Descent", arXiv: 2305.17490.

[62] P. Zhou, J. Feng, "The Landscape of Deep Learning Algorithms", arXiv: 1705.07038.

[63] P. Natekar, M. Sharma, "Representation Based Complexity Measures for Predicting Generalization in Deep Learning", arXiv: 2012.02775.

[64] G. Valle-Perez, C. Camargo, A. Louis, "Deep learning generalizes because the parameter-function map is biased towards simple functions", arXiv: 1805.08522.

[65] Y. Lei, Y. Ying, "Fine-Grained Analysis of Stability and Generalization for Stochastic Gradient Descent", arXiv: 2006.08157.

[66] B. London, "A PAC-Bayesian Analysis of Randomized Learning with Application to Stochastic Gradient Descent", *Advances in Neural Information Processing Systems*, **30**, 2017.

[67] D. Bisla, J. Wang, A. Choromanska, "Low-Pass Filtering SGD for Recovering Flat Optima in the Deep Learning Optimization Landscape", arXiv: 2201.08025.

[68] A. Camuto, G. Deligiannidis, M. Erdogdu, M. Gurbuzbalaban, U. Simsekli, L. Zhu, "Fractal Structure and Generalization Properties of Stochastic Optimization Algorithms", arXiv: 2106.04881.

[69] J. Wang, A. Choromanska, "A Survey of Optimization Methods for Training DL Models: Theoretical Perspective on Convergence and Generalization", arXiv: 2501.14458.

[70] V. Nagarajan, "Explaining generalization in deep learning: progress and fundamental limits", arXiv: 2110.08922.

[71] K. Kawaguchi, L. Kaelbling, Y. Bengio, "Generalization in Deep Learning", arXiv: 1710.05468.

[72] S. Arora, Z. Li, A. Panigrahi, "Understanding Gradient Descent on Edge of Stability in Deep Learning", arXiv: 2205.09745.

[73] T. Poggio, A. Banburski, Q. Liao, “Theoretical Issues in Deep Networks: Approximation, Optimization and Generalization”, arXiv: 1908.09375.

[74] T. Poggio, K. Kawaguchi, Q. Liao, B. Miranda, L. Rosasco, X. Boix, J. Hidary, H. Mhaskar, “Theory of Deep Learning III: explaining the non-overfitting puzzle”, arXiv: 1801.00173.

[75] N. Golowich, A. Rakhlin, O. Shamir, “Size-Independent Sample Complexity of Neural Networks”, *Information and Inference: A Journal of the IMA*, **9**:2 (2020), 473–504. DOI: 10.1093/imaiai/iaz007.

[76] N. Vyas, Y. Bansal, P. Nakkiran, “Limitations of the NTK for Understanding Generalization in Deep Learning”, arXiv: 2206.10012.

Received on May 5, 2026

Часть 2
Специальные вопросы теории интеллектуальных систем

Квантизация языковых моделей с помощью дифференцируемого поиска архитектуры нейронной сети

Д. Н. Давыдова*

Поиск архитектуры нейронной сети – это метод, позволяющий автоматически подобрать наилучшую архитектуру нейронной сети для данной задачи на представленном алгоритму наборе данных. Одной из разновидностей поиска архитектуры нейронной сети является DARTS – differentiable neural architecture search. DARTS позволяет перейти от дискретного пространства поиска к непрерывному и использовать метод градиентного спуска для подбора параметров, отвечающих за архитектуру сети. В статье будет рассмотрена возможность использования метода DARTS для подбора схемы квантизации и разрядности различных компонентов генеративных языковых моделей и описаны результаты проведенных экспериментов. Код для обучения и оценки квантизованной модели доступен по ссылке: https://github.com/daria1d/Darts-QAT.

Ключевые слова: обработка естественного языка, поиск архитектуры нейронной сети, квантизация, DARTS.

1. Введение

Метод поиска архитектуры нейронной сети позволяет автоматизировать процесс подбора архитектуры различных нейросетевых моделей. Подбор оптимальной архитектуры нейронной сети вручную может занимать значительное количество времени и требовать проведения большого количества экспериментов, в то время как поиск архитектуры нейронной сети позволяет существенно ускорить создание наиболее эффективной модели для данной задачи.

Среди методов поиска архитектуры нейронной сети можно выделить эволюционные алгоритмы, методы обучения с подкреплением, дифференцируемый поиск архитектуры нейронной сети, поиск архитектуры с использованием больших языковых моделей.

*Давыдова Дарья Николаевна — аспирант каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: d.dawydowa2017@yandex.ru, ORCID: 0009-0009-4338-9734.

Davydova Daria Nikolaevna — PhD student, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intellectual Systems.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

Первые подходы поиска архитектуры нейронной сети были основаны на эволюционных алгоритмах. Подбор архитектурных параметров с помощью эволюционных алгоритмов осуществляется за счет модификации набора потенциальных решений (популяции) на основе ряда правил (операторов). На первом этапе эволюционный алгоритм инициализирует набор случайных сетей для создания первичной популяции. Далее каждая архитектура оценивается с помощью функции приспособленности, которая отражает её качество (например, точность классификации). По этой оценке отбираются наиболее подходящие архитектуры, к которым применяются итеративные процедуры мутации, скрещивания и селекции [1] для создания новых архитектур-кандидатов. Эволюционный процесс продолжается до тех пор, пока не будет достигнут критерий завершения, который обычно определяется максимальным количеством допустимых поколений.

Генетические алгоритмы требуют большого количества вычислительных ресурсов и не способны использовать информацию о градиенте, в то время как RL-алгоритмы более эффективны и могут использовать информацию с прошлых шагов для более точного поиска архитектуры. В основе поиска архитектуры нейронной сети с использованием RL [2] лежит наблюдение, что широкий класс архитектур нейронных сетей можно описать с помощью строк переменной длины. Ключевая идея метода – обучить модель-контроллер – агента, который пошагово генерирует архитектуру нейронной сети, выбирая компоненты из заданного пространства. Затем производительность модели, построенной по сгенерированному описанию, оценивается на валидационном наборе, и полученная метрика (например, точность) используется в качестве награды для обучения контроллера. Таким образом, генерируя все больше сетей, контроллер обучается строить более производительные модели.

С появлением больших языковых моделей (LLM) популярными стали методы поиска архитектуры нейронной сети, в которых для подбора различных компонентов архитектуры моделей используются LLM. В некоторых статьях [3] исследуются способности LLM в написании кода: LLMatic формирует описание архитектуры в виде кода или псевдокода с помощью LLM, используя заранее заданные шаблоны (промпты). Затем полученные архитектуры оцениваются по качеству и разнообразию с использованием принципов Quality-Diversity оптимизации: отбираются не только точные, но и структурно различные модели. Каждая архитектура оценивается на точность и производительность. Отобранные кандидаты далее служат основой для новых запросов к LLM, что позволяет итеративно улучшать архитектуру.

Другой метод LLaMA-NAS [4] находит оптимальные по Парето варианты архитектур на базе LLaMA2-7B. В основе метода лежит концепция one-shot суперсети – большой, переопределённой модели, которая содержит множество возможных подархитектур, например, с меньшим числом трансформерных слоёв или меньшими размерами скрытых слоёв. Во время обучения из этой суперсети совместно выбираются и дообучаются несколько подархитектур, что позволяет использовать общие параметры и эффективно оценивать модели. Поиск в пространстве потенциальных подархитектур в LLaMA-NAS осуществляется с помощью генетического алгоритма. В результате получаются оптимальные по Парето варианты архитектур – набор моделей, оптимальных по размеру, временным затратам и точности.

Дифференцируемый поиск архитектуры нейронной сети (DARTS) [5] позволяет достигать качества, сравнимого с SOTA-моделями, затрачивая при этом меньше вычислительных ресурсов. В то время как большинство методов поиска архитектуры нейронной сети предполагают поиск по дискретному набору архитектур-кандидатов, DARTS сокращает пространство поиска, делая его непрерывным. Это позволяет обучать параметры архитектуры с использованием градиентного спуска. Такой подход помогает существенно сократить вычислительные затраты при сохранении качества, сравнимого с лучшими подобранными вручную архитектурами.

Далее будут проведены эксперименты, адаптирующие метод DARTS для подбора наиболее оптимальной схемы квантизации и степени сжатия для каждого компонента модели и описаны полученные результаты.

2. Алгоритм дифференцируемого поиска архитектуры нейронной сети

Метод DARTS [5] был предложен в статье DARTS: Differentiable Architecture Search, опубликованной в 2018 году. Целью работы было сделать процесс автоматического поиска архитектур нейронных сетей более эффективным и доступным, заменив дорогие дискретные поисковые методы, требующие сотен GPU, на дифференцируемый, градиентно-оптимизируемый подход.

Алгоритм предполагает поиск ячейки архитектуры сети – повторяющегося базового элемента, служащего шаблоном для построения всей модели. Найденную ячейку можно укладывать последовательно для формирования итоговой архитектуры или сочетать с другими компонентами нейронной сети.

Каждая ячейка представляет собой ориентированный граф без циклов, содержащий N промежуточных вершин, две входных и одну выходную. Каждая вершина графа соответствует скрытому представлению $x(i)$ (например, карте признаков в случае сверточной нейронной сети), а каждое ребро – операции $o^{(i,j)}$, которая преобразует выход узла $x^{(i)}$ и передаёт результат в узел $x^{(j)}$. Выходные данные ячейки получаются путем применения операции сокращения (например, конкатенации) ко всем промежуточным узлам.

Каждая промежуточная вершина вычисляется с учетом всех своих предков:

$$x^{(j)} = \sum_{i<j} o^{(i,j)}(x^{(i)})$$

Специальная нулевая операция указывает на отсутствие ребра между двумя вершинами. Задача подбора ячейки таким образом сводится к подбору операций на ребрах.

Для перехода от дискретного к непрерывному пространству поиска выбор операции определяется как функция Softmax над множеством всех операций-кандидатов:

$$\overline{o}^{(i,j)}(x) = \sum_{o \in O} \frac{exp(\alpha_o^{(i,j)})}{\sum_{o' \in O} exp(\alpha_{o'}^{(i,j)})} o(x),$$

где веса смешанной операции для пары вершин (i, j) определяются вектором $\alpha^{(i,j)}$ размерности $|O|$.

Задача поиска архитектуры тогда сводится к обучению множества непрерывных переменных $\alpha = \{\alpha^{(i,j)}\}$. После процедуры поиска дискретную архитектуру можно получить заменой каждой смешанной операции $o'^{(i,j)}$ на наиболее вероятную операцию, то есть, $o^{(i,j)} = argmax_{o \in O} \alpha_o^{(i,j)}$.

В процессе поиска архитектуры DARTS решает задачу двухуровневой оптимизации, где параметры весов w и архитектуры α обучаются попеременно на разных подмножествах данных. На внутреннем уровне минимизируется ошибка на обучающей выборке:

$$w^*(\alpha) = \arg\min_w \mathcal{L}_{train}(w, \alpha).$$

На внешнем уровне оптимизируются архитектурные параметры по ошибке на валидации:

$$\min_\alpha \mathcal{L}_{val}(w^*(\alpha), \alpha).$$

На каждой итерации алгоритм обновляет w градиентным спуском по $\mathcal{L}_{train}$, фиксируя α, затем обновляет α по $\mathcal{L}_{val}$, фиксируя w. После завершения поиска полученная архитектура дискретизируется (выбираются операции с наибольшими коэффициентами α), и сеть обучается заново, оптимизируя только веса w.

$$\min_{\alpha} \mathcal{L}_{val}(w^*(\alpha), \alpha)$$
$$w^*(\alpha) = argmin_w \mathcal{L}_{train}(w, \alpha)$$

.

3. DARTS для квантизации языковых моделей

В [6] было проведено исследование по применению метода DARTS для подбора оптимальной степени сжатия различных компонентов модели в сверточных нейронных сетях. Архитектура ячейки сети (количество слоев, размерность фильтра свертки для каждого слоя) является фиксированной для всех вариантов квантизации. Слои ячейки связаны несколькими параллельными ребрами, представляющими операцию свертки, параметры и активации которой приводятся к разным разрядностям с помощью DARTS. Как и в оригинальном DARTS, для обновления параметров используется двухуровневая оптимизация, а в функцию потерь для архитектурных параметров включен также штраф за выбор высокой разрядности в двух вариантах. В первом случае авторы делают акцент на сжатии модели, а во втором – на ускорение и оптимизацию энергопотребления:

$$\mathcal{L} = \mathrm{CE} + C \cdot \sum_{e_{ij} \in E_k} m_{ij} \, \#\mathrm{PARAM}(e_{ij}) \, \mathrm{bit}_w(e_{ij}),$$

$$\mathcal{L} = \mathrm{CE} + C \cdot \sum_{e_{ij} \in E_k} m_{ij} \, \#\mathrm{FLOP}(e_{ij}) \, \mathrm{bit}_w(e_{ij}) \, \mathrm{bit}_a(e_{ij}).$$

где CE – кросс-энтропия, e_{ij} – ребро в графе ячейки, m_{ij} – маска выбора ребра, $m_{ij} \in [0, 1]$, C – коэффициент, определяющий вклад штрафа за вычислительную стоимость в функцию потерь, E_k – множество ребер-кандидатов рассматриваемой ячейки.

Использование DARTS для моделей ResNet [7], MobileNet [8], VGG-small [9] на задаче классификации изображений позволило превзойти результаты SOTA-методов квантизации и существенно сжать модель.

Авторы исследования [10] также используют DARTS для подбора схем квантизации для сверточных нейронных сетей MobileNet [8] и ResNet

[7]. Метод сохраняет идею двухуровневой оптимизации из оригинального DARTS и позволяет избегать использования прокси-задачи на больших моделях. В исследовании параметры квантизации (тип операции и битность) и параметры модели оптимизируются совместно по одной функции потерь, в которую включен штраф за высокую разрядность:

$$L[F] = CE + \eta \sum_{f \in F} \mathbb{E}[b_f]\, \mathbb{E}[b_a]\, \frac{|f|\, w\, h}{s^2},$$

$$\mathbb{E}[b_f] = \sum_i \pi_i\, \alpha_i^{b_f}, \qquad \mathbb{E}[b_a] = \sum_j \pi_j\, \beta_j^{b_a},$$

где CE – кросс-энтропия, f – операция (сверточный или линейный фильтр), w, h – ширина и высота выхода фильтра, s – шаг операции f, b_f – битность весов, b_a – битность активаций, $\mathbb{E}[b_f]$ – ожидаемая битность весов фильтра, $\mathbb{E}[b_a]$ – ожидаемая битность активаций, π_i – вероятность выбора i-й операции, η – коэффициент, определяющий вклад штрафа за вычислительную стоимость, $|f|$ – число параметров операции f, i и j – индексы кандидатов разрядности весов и активаций соответственно.

В результате удалось сжать исходные модели в несколько раз с точностью, близкой к исходным 32-битным моделям.

В статье [11] предложен метод ∇QDARTS, в котором с помощью дифференциируемого поиска подбирается не только архитектура, но и разрядность для различных компонентов сети. Исследования проводились на ResNet, MobileNet [8] и EfficientNet [12] для задач классификации изображений. В методе ∇QDARTS процесс обучения происходит на двух уровнях: на внешнем уровне оптимизируются архитектурные параметры, а на внутреннем – параметры разрядности и параметры самой сети. Для поощрения модели к снижению битности на внешнем уровне вводится штраф за высокую разрядность, что позволяет добиться баланса между точностью и эффективностью модели.

$$\mathcal{L}[F] = R_E(F) + \nu \sum_{i=1}^{N_{\mathrm{OP}}} \mathrm{OP}_i \cdot \mathrm{BitA}_i \cdot \mathrm{BitW}_i$$

$$\mathrm{BitA}_i = \gamma_i^p a_i, \quad \mathrm{BitW}_i = \gamma_i^q w_i,$$

где $R_E(F)$ – эмпирическая ошибка модели F, $\nu \geqslant 0$ – обучаемый коэффициент при штрафе за битность, OP_i – вычислительная стоимость i-й операции, N_{OP} – число элементарных вычислений слоя, a_i, w_i – активации и веса, γ_i^p, γ_i^q – параметры, отвечающие за отбор битности.

На задачах классификации изображений метод обеспечил до 2–3% прироста точности при снижении вычислительных затрат и размера модели на 30–50% по сравнению с методом, представленным в [6].

В исследовании [13] авторы использовали метод DARTS для поиска оптимальной схемы квантизации трансформерной модели BERT [14]. В [13] архитектурные параметры α отвечают за подбор разрядности весов и активаций линейных компонентов модели. Параметры α подбираются для различных групп весов и активаций (для отдельной матрицы, для всего механизма внимания или для целых блоков модели). Авторы используют двухуровневую оптимизацию – параметры модели оптимизируются с помощью кросс-энтропии, а параметры подбора разрядности оптимизируются с помощью функции потерь, сочетающей кросс-энтропию и штраф за высокую битность:

$$\mathcal{L} = CE + \lambda\, L_{\text{size}},$$

где λ – коэффициент штрафа за размер модели, L_{size} – штраф за размер модели.

Штраф за размер модели должен обеспечить низкую разрядность модели, сохранив при этом достаточно высокую точность. Для этого вводится желаемый размер модели, и сеть штрафуется как за слишком высокую разрядность для группы параметров, так и за слишком низкую:

$$L_{\text{size}} = \begin{cases} 0, & C_O \in [(1-\epsilon)V,\, (1+\epsilon)V] \\ \log \mathbb{E}[CO_{i,j}], & C_O > (1+\epsilon)V \\ -\log \mathbb{E}[CO_{i,j}], & C_O < (1-\epsilon)V, \end{cases}$$

где C_O – фактический размер модели с распределением разрядностей O, V – желаемый размер модели, $\mathbb{E}[CO_{i,j}]$ – ожидаемый вклад группы параметров в размер модели, i – индекс слоя, j – индекс группы параметров внутри слоя.

Эксперименты на модели BERT с задачами из GLUE [15] показали, что предложенный метод сохраняет до 99% точности исходной модели при заметном сокращении вычислительной сложности и объёма памяти, демонстрируя, что принципы DARTS могут быть эффективно адаптированы для задач смешанной квантизации и сжатия трансформеров.

4. Эксперимент по применению DARTS для квантизации генеративной трансформерной модели

В статье будет описан эксперимент по подбору схемы квантизации и разрядности для весов и активаций различных компонентов генеративной трансформерной модели с помощью метода DARTS и представлены полученные результаты.

В отличие от прошлых исследований [13], метод DARTS будет адаптирован для генеративной трансформерной модели, состоящей только из блоков декодера. Метод подбирает битность для каждого линейного слоя модели из блоков декодера и учитывает ошибку квантизации для уменьшения потерь в качестве работы модели при снижении разрядности.

4.1. Подбираемые схемы квантизации

Для эксперимента в качестве кандидатов были выбраны следующие схемы квантизации: PACT – квантизация активаций с обучаемым порогом отсечения [16], LSQ+ – квантизация с обучаемым шагом [17] и равномерная симметричная квантизация [18].

Равномерная симметричная квантизация [18] – это представление непрерывных значений весов и активаций дискретными значениями с фиксированным шагом. Диапазон значений квантизации при этом симметричен относительно 0. Таким образом, нулевая точка квантизации всегда находится в 0, а все уровни квантизации имеют один и тот же шаг.

Квантизованное целочисленное значение веса или активации вычисляется по формуле:

$$\bar{x} = \text{clamp}\Big(\text{round}\big(x/s\big), n, p\Big),$$

где s – шаг квантизации, n и p – границы диапазона квантизации. Восстановленное деквантизованное значение вычисляется по формуле:

$$\hat{x} = \bar{x} \times s$$

Для обратного прохода по сети используется straight-through-estimator (STE) [19].

$$\frac{\partial \hat{x}}{\partial x} = s\frac{\partial \bar{x}}{\partial x} \simeq \begin{cases} 1, & ns < x < ps, \\ 0, & \text{иначе.} \end{cases}$$

Квантизация методом LSQ+ [17] – усовершенствованная версия метода LSQ с обучаемым шагом квантизации и пересчетом градиента с учётом

диапазона квантизации, что позволяет лучше адаптировать метод для квантизации к низким разрядностям.

Для квантизации активаций предлагается асимметричная схема, в которой обучается не только шаг квантизации, но и сдвиг, что позволяет корректно обрабатывать смещённое относительно 0 распределение активаций.

$$\bar{x} = round\Big(\text{clamp}\Big(\frac{x-\beta}{s},\, n,\, p\Big)\Big)$$

$$\hat{x} = \bar{x} \times s + \beta,$$

где s – шаг квантизации, β – сдвиг квантизации, n, p – границы диапазона квантизации, $\hat{x}$ – восстановленные (деквантизованные) активации, $\bar{x}$ – квантованные активации.

Обновление градиента s вычисляется по следующей формуле:

$$\frac{\partial \hat{x}}{\partial s} = \frac{\partial \bar{x}}{\partial s} s + \bar{x} \simeq \begin{cases} -\frac{x-\beta}{s} + round\Big(\frac{x-\beta}{s}\Big), & \text{если } n < \frac{x-\beta}{s} < p \\ n \text{ или } p, & \text{иначе} \end{cases}$$

Обновление градиента β вычисляется по формуле:

$$\frac{\partial \hat{x}}{\partial \beta} = \frac{\partial \bar{x}}{\partial \beta}\, s + 1 \simeq \begin{cases} 0, & \text{если } n < \dfrac{x-\beta}{s} < p \\ 1, & \text{иначе} \end{cases}$$

В обоих случаях для вычисления $\frac{\partial \bar{x}}{\partial \beta}$ и $\frac{\partial \bar{x}}{\partial s}$ используется straight-through-estimator (STE) [19].

Для весов используется симметричная квантизация со знаком, поскольку на практике распределение весов слоя обычно симметрично относительно нуля. Асимметричная квантизация активаций не вносит дополнительной вычислительной стоимости на этапе инференса, так как сдвиг квантизации может быть предварительно вычислен и включён в значение смещения слоя при запуске.

$$\hat{w}\hat{x} = (\bar{w} \times s_w)\,(\bar{x} \times s_x + \beta) = \bar{w}\bar{x}\, s_w s_x + \underbrace{\beta s_w \bar{w}}_{\text{смещение слоя}},$$

где $\hat{w}$ – восстановленные (деквантизованные) веса, $\bar{w}$ – квантизованные веса.

Также в качестве кандидата схем квантизации рассматривается метод квантизации активаций PACT (Parameterized Clipping Activation

for Quantized Neural Networks) [16]. Проблема, которую решает PACT – потеря точности при квантизации активаций в связи с тем, что при квантизации выходов таких функций, как ReLU, большинство значений близки к нулю, и появляется несколько очень больших значений-выбросов. В статье используется асимметричный вариант PACT с обучаемыми нижней и верхней границами. При квантизации с использованием PACT вводятся обучаемые параметры α_l, α_h, задающие нижнюю и верхнюю границу активации:

$$y = \mathrm{PACT}(x) = \mathrm{clip}(x, \alpha_l, \alpha_h) = \begin{cases} \alpha_l, & \text{если } x < \alpha_l, \\ x, & \text{если } \alpha_l \leqslant x < \alpha_h, \\ \alpha_h, & \text{если } x \geqslant \alpha_h. \end{cases}$$

После обрезки значения равномерно квантуются к k бит в диапазоне $[\alpha_l, \alpha_h]$:

$$y_q = \mathrm{round}\Big(\frac{y - \alpha_l}{s}\Big) \cdot s + \alpha_l, \quad \text{где } s = \frac{\alpha_h - \alpha_l}{2^k - 1}.$$

В процессе обратного распространения ошибки градиенты $\frac{\partial y_q}{\partial \alpha_l}$, $\frac{\partial y_q}{\partial \alpha_h}$ вычисляются с использованием суррогатной производной: градиент распространяется через операцию отсечения: $y = \mathrm{clip}(x, \alpha_l, \alpha_h)$, а остальные зависимости квантизованного значения y_q от границ α_l и α_h в обратном проходе не учитываются.

$$\frac{\partial y_q}{\partial \alpha_l} \simeq \frac{\partial y_q}{\partial y}\frac{\partial y}{\partial \alpha_l} = \begin{cases} 1, & x < \alpha_l \\ 0, & x \geqslant \alpha_l \end{cases}$$

$$\frac{\partial y_q}{\partial \alpha_h} \simeq \frac{\partial y_q}{\partial y}\frac{\partial y}{\partial \alpha_h} = \begin{cases} 0, & x < \alpha_h \\ 1, & x \geqslant \alpha_h \end{cases}$$

4.2. Смешанная операция линейного слоя

При поиске схемы квантизации с помощью DARTS подбираются разрядности для всех линейных слоев модели, за исключением слоя эмбеддингов и выходного слоя логитов, которые наиболее чувствительны к понижению разрядности.

Для каждого линейного слоя ℓ задаётся набор квантизаторов-кандидатов для весов $\{Q_i^w\}_{i=1}^{N_w}$ и активаций $\{Q_j^a\}_{j=1}^{N_a}$ с битностями b_i^w

и b_j^a соответственно. Подбор квантизатора и разрядности для весов и активаций осуществляется с помощью параметров $\boldsymbol{\alpha}^w, \boldsymbol{\alpha}^a$, обучаемых градиентным спуском.

Для перехода к непрерывному пространству поиска параметров, отвечающих за подбор схемы квантизации и битности, выбран Softmax с температурой, позволяющий во время обучения исследовать различные сочетания квантизаторов и разрядностей, а затем перейти к более жесткому выбору битности для каждого слоя.

$$p_i(T) = \frac{\exp\left(\frac{\alpha_i}{T}\right)}{\sum\limits_{j=1}^{N} \exp\left(\frac{\alpha_j}{T}\right)},$$

где α_i – параметры квантизаторов, T – температура.

Активации и веса квантизуются выбранными кандидатами-квантизаторами с заданными битностями по формулам:

$$\tilde{x} = \sum_{j=1}^{N_a} p_j^a \cdot Q_j^a(x)$$

$$\tilde{W} = \sum_{i=1}^{N_w} p_i^w \cdot Q_i^w(W)$$

Таким образом, полная смешанная операция на линейных слоях модели задается следующей формулой:

$$Y = \left(\sum_{i=1}^{N_a} p_i^a \, Q_i^a(X)\right) \cdot \left(\sum_{j=1}^{N_w} p_j^w \, Q_j^w(W)\right)^{\mathsf{T}} + b$$

4.3. Оптимизация и функция потерь

Как и в классическом DARTS, оптимизация параметров модели и параметров квантизаторов проводится на двух уровнях. На внутреннем уровне оптимизируются параметры сети с помощью кросс-энтропии, на внешнем – параметры квантизаторов.

Цель поиска схемы квантизации и разрядности с помощью DARTS – уменьшить размер и вычислительные затраты модели, сохранив при этом ее качество. Поэтому функция потерь внешнего уровня состоит из кросс-энтропии и функции, отвечающей за вычислительные затраты, которые необходимо минимизировать.

Для определения вычислительных затрат необходимо оценить среднюю разрядность слоя и количество операций, выполняемых в данном слое. Для подсчета средней разрядности вычисляется ожидаемая битность весов и активаций при смешанной квантизации на данном линейном слое:

$$\mathbb{E}[b_w] = \sum_{i=1}^{N_w} p_i^w b_i^w, \qquad \mathbb{E}[b_a] = \sum_{j=1}^{N_a} p_j^a b_j^a$$

Количество операций подсчитывается как количество MAC (Multiply–Accumulate) – базовых вычислительных операций, состоящих из двух действий: умножения двух чисел и последующего прибавления результата к накопленной сумме.

$$a = a + (b \cdot c)$$

Тогда количество операций для линейного слоя внутри многослойного персептрона будет вычисляться по формуле:

$$MACs_L = d_{in} \cdot d_{out} \cdot L,$$

где d_{in}, d_{out} – входная и выходная размерность слоя соответственно, L – средняя длина последовательности в корпусе.

Итоговая вычислительная стоимость линейного слоя может быть определена формулой:

$$Cost_L = MACs_L \cdot \mathbb{E}[b_w] \cdot \mathbb{E}[b_a]$$

Стоимость вычислений линейных преобразований для матриц Q, K, V и выходной проекции в механизме внимания можно вычислить по формуле:

$$MACs_{Att} = L \cdot d_{model}^2$$

Стоимость вычислений для операций QK^T и $Softmax(QK^T)V$ определяется следующим образом:

$$MACs_{Att} = L^2 \cdot d_{model}$$

Тогда итоговая вычислительная стоимость слоя внимания может быть определена формулой:

$$Cost_{Att} = L \cdot d_{model}^2 \cdot \Big(\mathbb{E}[b_{w_Q}] \cdot \mathbb{E}[b_{a_Q}] + \mathbb{E}[b_{w_K}] \cdot \mathbb{E}[b_{a_K}] + \mathbb{E}[b_{w_V}] \cdot \mathbb{E}[b_{a_V}] + \\ + \mathbb{E}[b_{w_{out}}] \cdot \mathbb{E}[b_{a_{out}}]\Big) + 2L^2 \cdot d_{model}$$

Суммарная вычислительная стоимость по всем слоям определяется следующим образом:

$$\mathcal{L}_{cost} = \sum_{\ell=1}^{N} \mathrm{Cost}_{L_\ell} + \sum_{\ell=1}^{N} \mathrm{Cost}_{Att_\ell},$$

где N - число трансформерных блоков модели.

Дополнительно вводится штраф за ошибку квантизации, предотвращающий слишком резкий переход модели к низким битностям. Он определяется среднеквадратической ошибкой между исходными весами и активациями и их квантизованными версиями, домноженными на соответствующие параметры квантизации:

$$\mathcal{L}_{\mathrm{err}} = \sum_{\ell=1}^{N_L} \left(\sum_{i=1}^{N_w} p_{w,i}^{(\ell)} \frac{1}{|W_\ell|} \left\| W_\ell - Q_i(W_\ell) \right\|^2 + \sum_{j=1}^{N_a} p_{a,j}^{(\ell)} \frac{1}{|X_\ell|} \left\| X_\ell - Q_j(X_\ell) \right\|^2 \right).$$

Итоговая функция потерь, учитывающая точность модели и ее вычислительную сложность, задается формулой:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{cost} \cdot \mathcal{L}_{cost} + \lambda_{err} \cdot \mathcal{L}_{\mathrm{err}}$$

где $\mathcal{L}_{CE}$ кросс-энтропия для задачи языкового моделирования, а $\lambda_{cost}, \lambda_{err}$ – коэффициенты, балансирующие точность модели и ее вычислительную стоимость.

4.4. Результаты эксперимента

Эксперимент проводился на модели GPT-Neo-125m [20] – трансформерной декодерной модели, содержащей стандартный механизм внимания и многослойный персептрон, состоящий из 2 линейных слоев. Модель была предобучена на задачу генерации текста на корпусе The Pile [21], содержащем около 800 ГБ текста, включая научные статьи, код, книги, энциклопедии, форумы.

Квантизация проводилась на входных активациях и весах матриц Q, K, V и выходной проекции механизма внимания, а также входного и выходного полносвязного линейного слоя многослойного персептрона. Были выбраны следующие квантизаторы-кандидаты и разрядности для весов: квантизатор LSQ+ и симметричный равномерный квантизатор с разрядностями 32, 16, 8 и 4 бит, и LSQ+, PACT и симметричный равномерный квантизатор с разрядностями 32, 16, 8, и 6 бит для активаций.

Размер батча был задан равным 16, длина последовательности 512, использовался оптимизатор AdamW с шагом обучения $2 \cdot 10^{-5}$ для оптимизации параметров модели, AdamW с шагом обучения 10^{-4} для архитектурных параметров. На каждой итерации сначала обновлялись параметры модели по обучающему батчу, затем архитектурные параметры по отдельному валидационному батчу. В эксперименте использовались значения $\lambda_{cost} = 0.02$, $\lambda_{err} = 0.05$. В ходе обучения температура Softmax линейно уменьшалась от 1.5 до 1 по шагам оптимизации. Для токенизации текстов использовался оригинальный токенизатор модели GPT-Neo-125m. Модель обучалась в течение 1 эпохи на подмножестве корпуса The Pile [21], содержащем 20000 текстов. Для внутреннего уровня оптимизации использовалось 60% выборки, для внешнего - 40%. Полное распределение разрядностей и квантизаторов по слоям, полученное в результате поиска DARTS, приведено в приложении.

Качество исходной и квантизованной модели оценивалось с помощью перплексии – обратной вероятности последовательности токенов, нормализованной по ее длине:

$$PPL(W) = \sqrt[N]{\frac{1}{P(w_1, w_2, ..., w_N)}},$$

где W – упорядоченная последовательность токенов тестовой коллекции, N – количество токенов в этой последовательности.

Перплексия замерялась на следующих корпусах:

- wikitext-2 [22], содержащем англоязычные статьи из Википедии размером около 2 миллионов токенов, для оценки использовалась подвыборка test;
- wikitext-103 [22], содержащем статьи из Википедии размером 103 миллиона токенов, для оценки использовалась подвыборка validation;
- FineWeb-EDU [23], содержащем извлечённые из Common Crawl и отфильтрованные по качеству тексты образовательной направленности размером 1.3 триллионов токенов, для оценки использовалась подвыборка train;
- OpenWebText [24] – аналоге корпуса WebText [25], на котором обучалась модель GPT-2 [25], размером 9 миллиардов токенов, для оценки использовалась подвыборка train;
- PG-19 [26], содержащем длинные художественные тексты из Project Gutenberg [27] размером около 2 миллиардов токенов, для оценки использовалась подвыборка validation.

Далее метод поиска схемы квантизации с помощью DARTS сравнивается с другими методами квантизации. Первые 3 из них предполагают квантизацию в ходе обучения (QAT), оставшиеся 2 – квантизацию после обучения (PTQ).

- Квантизация весов по равномерной симметричной схеме к 8 бит;
- Квантизация весов по равномерной симметричной схеме, а активаций по асимметричной схеме к 8 бит по подходу LLM-QAT [28];
- Квантизация весов методом nf4 [29], используемым в QLoRA, адаптированным для QAT, к 4 бит;
- Квантизация весов методом LLM.int8(), введенным в [30], к 8 бит;
- Квантизация весов методом nf4 [29] к 4 бит.

Для оценки степени сжатия модели рассчитывается средняя битность параметров модели с учетом как квантизуемых параметров, так и параметров, сохраняемых в полной разрядности:

$$\text{AvgBits} = \frac{32N_{\text{fp}} + \sum_{i=1}^{N_q} N_i b_i}{N_{\text{fp}} + \sum_{i=1}^{N_q} N_i},$$

где N_{fp} - количество элементов матриц весов, оставшихся в разрядности 32 бит, N_q - количество квантизуемых матриц весов, N_i - число элементов в i-й квантизуемой матрице, b_i - ее разрядность.

Смещения линейных слоев и параметры нормализации при вычислении AvgBits не учитываются, разделяемая матрица весов эмбеддингов и выходного слоя учитывается при подсчете один раз.

Оценка вычислительной сложности модели определяется следующим образом:

$$BitOps = \sum_{i=1}^{n} MACs_i * b_{w_i} * b_{a_i} + 32^2 \sum_{j=1}^{m} MACs_j,$$

где n – количество квантизуемых линейных слоев, b_{w_i} – битность весов i-го слоя, b_{a_i} – битность активаций i-го слоя, m – количество слоев, оставляемых в полной разрядности.

При оценке вычислительной сложности учитывались линейные проекции Q, K, V, O в механизме внимания, два линейных слоя в многослойном персептроне, матричные умножения $QK^\top$ и AV, где A - матрица весов внимания, а также выходной линейный слой.

После проведения экспериментов получены следующие значения перплексии исходной модели GPT-Neo-125m и квантизованных версий на корпусах:

Таблица 1: Результаты экспериментов на модели GPT-Neo-125m

	Wikitext-2	Wikitext-103	FineWeb-EDU	OpenWebText	PG-19	AvgBit	BitOps (G)
GPT-Neo-125m	24.11	25.50	23.28	21.75	22.31	32	68477.21
QAT w8-a32	54.86	50.75	24.81	22.14	23.43	15.72	35079.55
QAT w8-a8	63.52	58.82	28.37	25.32	27.99	15.72	26730.13
QAT w4-a32	129.17	114.55	51.67	44.75	47.18	13.0	29513.27
PTQ w8-a32	60.38	43.28	24.55	22.43	23.59	15.72	35079.55
PTQ w4-a32	86.71	67.30	34.13	31.31	31.73	13.0	29513.27
DARTS+QAT	54.54	50.34	24.80	22.13	23.68	15.64	28141.03

Эксперимент показал сохранение работоспособности модели GPT-Neo-125m несмотря на ее небольшой размер при сжатии модели в 2.05 раза. Результаты показывают высокий потенциал использования методов архитектурного поиска для подбора схем квантизации, в том числе, для моделей больших размеров, таких как LLaMA-7B, Qwen3-8B и других. Кроме того, полученная схема квантизации интерпретируема и показывает, какие из слоев оказываются более чувствительными к ошибке квантизации, а какие допускают более сильное сжатие: для большинства матриц весов выбирается разрядность 8 бит, тогда как отдельные проекции механизма внимания допускают более сильное сжатие до 4 бит.

Список литературы

[1] A. E. Eiben, J. E. Smith, *Introduction to evolutionary computing*, Springer, Berlin, Heidelberg, 2015, ISBN: 9783662448731,3662448748. DOI: 10.1007/978-3-662-44874-8, 287 pp.

[2] B. Zoph, Q. V. Le, "Neural architecture search with reinforcement learning", *International Conference on Learning Representations (ICLR)*, 2017 https://arxiv.org/abs/1611.01578.

[3] M. U. Nasir, S. Earle, J. Togelius, S. James, C. Cleghorn, "LLMatic: Neural architecture search via large language models and quality diversity optimization", *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Association for Computing Machinery, New York, NY, USA, 2024, 1110–1118. DOI: 10.1145/3638529.3654017.

[4] A. Sarah, S. N. Sridhar, M. Szankin, S. Sundaresan, "LLaMA-NAS: Efficient neural architecture search for large language models", *European Conference on Computer Vision (ECCV)*, Springer Nature Switzerland, Cham, Switzerland, 2024, 67–74. DOI: 10.48550/arXiv.2405.18377.

[5] H. Liu, K. Simonyan, Y. Yang, “DARTS: Differentiable architecture search”, *International Conference on Learning Representations (ICLR)*, 2019 https://arxiv.org/abs/1806.09055.

[6] B. Wu, Y. Wang, P. Zhang, Y. Tian, “Mixed precision quantization of convnets via differentiable neural architecture search”, *arXiv preprint*, 2018 https://arxiv.org/abs/1812.00090.

[7] K. He, X. Zhang, S. Ren, J. Sun, “Deep residual learning for image recognition”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, 770–778. DOI: 10.48550/arXiv.1512.03385.

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications”, *arXiv preprint*, 2017 https://arxiv.org/abs/1704.04861.

[9] K. Simonyan, A. Zisserman, “Very deep convolutional networks for large-scale image recognition”, *International Conference on Learning Representations (ICLR)*, 2015 https://arxiv.org/abs/1409.1556.

[10] Z. Cai, N. Vasconcelos, “Rethinking differentiable search for mixed-precision neural networks”, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 2349–2358. DOI: 10.48550/arXiv.2004.05795.

[11] P. Behnam, U. Kamal, S. Vijay Ganesh, Z. Li, M. A. Jurado, A. Khare, I. Fedorov, G. Liu, A. Tumanov, “∇QDARTS: Quantization as an Elastic Dimension to Differentiable NAS”, *Transactions on Machine Learning Research*, 2025 https://openreview.net/forum?id=ubrOSWyTS8.

[12] M. Tan, Q. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks”, *International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research, 2019, 6105–6114 https://arxiv.org/abs/1905.11946.

[13] C. Zhao, T. Hua, Y. Shen, Q. Lou, H. Jin, “Automatic mixed-precision quantization search of BERT”, *arXiv preprint*, 2021 https://arxiv.org/abs/2112.14938.

[14] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019 https://arxiv.org/abs/1810.04805.

[15] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding", *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, 2018, 353–355. DOI: 10.18653/v1/W18-5446.

[16] J. Choi, Z. Wang, S. Venkataramani, P. Chuang, V. Srinivasan, K. Gopalakrishnan, "PACT: Parameterized clipping activation for quantized neural networks", *arXiv preprint*, 2018 https://arxiv.org/abs/1805.06085.

[17] Y. Bhalgat, J. Lee, M. Nagel, T. Blankevoort, N. Kwak, "LSQ+: Improving low-bit quantization through learnable offsets and better initialization", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, 696–697. DOI: 10.1109/CVPRW50498.2020.00356.

[18] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, 2704–2713. DOI: 10.1109/CVPR.2018.00286.

[19] Y. Bengio, N. Léonard, A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation", *arXiv preprint*, 2013 https://arxiv.org/abs/1308.3432.

[20] S. Black, G. Leo, P. Wang, C. Leahy, S. Biderman, "GPT-Neo: Large scale autoregressive language modeling with mesh-tensorflow", *Zenodo*, 2021. DOI: 10.5281/zenodo.5297715.

[21] L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, C. Leahy, "The Pile: An 800GB dataset of diverse text for language modeling", *arXiv preprint*, 2020 https://arxiv.org/abs/2101.00027.

[22] S. Merity, C. Xiong, J. Bradbury, R. Socher, "Pointer sentinel mixture models", *arXiv preprint*, 2016 https://arxiv.org/abs/1609.07843.

[23] "FineWeb-EDU dataset", *Hugging Face Datasets*, 2024 https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu.

[24] "OpenWebText Corpus", *GitHub repository*, 2019 https://github.com/skylion007/OpenWebTextCorpus.

[25] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, "Language models are unsupervised multitask learners", *OpenAI Technical Report*, 2019 https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[26] J. W. Rae, A. Potapenko, S. M. Jayakumar, C. Hillier, "Compressive transformers for long-range sequence modelling", *International Conference on Learning Representations (ICLR)*, 2020 https://arxiv.org/abs/1911.05507.

[27] M. Hart, "Project Gutenberg", 1971 https://www.gutenberg.org.

[28] Z. Liu, B. Oguz, C. Zhao, E. Chang, P. Stock, Y. Mehdad, Y. Shi, R. Krishnamoorthi, V. Chandra, "LLM-QAT: Data-free quantization aware training for large language models", *Findings of the Association for Computational Linguistics: ACL 2024*, 2024, 467–484. DOI: 10.18653/v1/2024.findings-acl.26.

[29] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs", *Advances in Neural Information Processing Systems*, **36**, 2023, 10088–10115 https://arxiv.org/abs/2305.14314.

[30] T. Dettmers, M. Lewis, Y. Belkada, L. Zettlemoyer, "LLM.int8(): 8-bit matrix multiplication for transformers at scale", *arXiv preprint*, 2022. DOI: 10.48550/arXiv.2208.07339.

Приложение А. Результаты подбора разрядностей для трансформерной генеративной модели

Автоматически подобранные битности весов и активаций для линейных проекций модели GPT-Neo-125m:

```
module.transformer.h.0.attn.attention.k_proj: W=8 (uniform) | A=8 (
    uniform)
module.transformer.h.0.attn.attention.v_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.0.attn.attention.q_proj: W=8 (uniform) | A=8 (
    lsq)
module.transformer.h.0.attn.attention.out_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.0.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.0.mlp.c_proj: W=8 (uniform) | A=8 (lsq)
```

```
module.transformer.h.1.attn.attention.k_proj: W=8 (lsq) | A=8 (lsq)
module.transformer.h.1.attn.attention.v_proj: W=4 (uniform) | A=8 (
    pact)
module.transformer.h.1.attn.attention.q_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.1.attn.attention.out_proj: W=4 (uniform) | A=8
    (lsq)
module.transformer.h.1.mlp.c_fc: W=8 (lsq) | A=8 (pact)
module.transformer.h.1.mlp.c_proj: W=8 (uniform) | A=8 (uniform)

module.transformer.h.2.attn.attention.k_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.2.attn.attention.v_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.2.attn.attention.q_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.2.attn.attention.out_proj: W=8 (uniform) | A=8
    (uniform)
module.transformer.h.2.mlp.c_fc: W=8 (uniform) | A=8 (lsq)
module.transformer.h.2.mlp.c_proj: W=8 (uniform) | A=8 (uniform)

module.transformer.h.3.attn.attention.k_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.3.attn.attention.v_proj: W=8 (lsq) | A=8 (lsq)
module.transformer.h.3.attn.attention.q_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.3.attn.attention.out_proj: W=8 (uniform) | A=8
    (uniform)
module.transformer.h.3.mlp.c_fc: W=8 (uniform) | A=8 (uniform)
module.transformer.h.3.mlp.c_proj: W=8 (uniform) | A=8 (uniform)

module.transformer.h.4.attn.attention.k_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.4.attn.attention.v_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.4.attn.attention.q_proj: W=8 (uniform) | A=8 (
    lsq)
module.transformer.h.4.attn.attention.out_proj: W=8 (uniform) | A=8
    (uniform)
module.transformer.h.4.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.4.mlp.c_proj: W=8 (lsq) | A=32 (identity)

module.transformer.h.5.attn.attention.k_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.5.attn.attention.v_proj: W=8 (lsq) | A=8 (lsq)
```

```
module.transformer.h.5.attn.attention.q_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.5.attn.attention.out_proj: W=8 (uniform) | A=8
    (uniform)
module.transformer.h.5.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.5.mlp.c_proj: W=8 (uniform) | A=8 (uniform)

module.transformer.h.6.attn.attention.k_proj: W=8 (lsq) | A=8 (pact)
module.transformer.h.6.attn.attention.v_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.6.attn.attention.q_proj: W=8 (uniform) | A=8 (
    lsq)
module.transformer.h.6.attn.attention.out_proj: W=4 (uniform) | A=8
    (uniform)
module.transformer.h.6.mlp.c_fc: W=8 (uniform) | A=8 (uniform)
module.transformer.h.6.mlp.c_proj: W=8 (lsq) | A=8 (lsq)

module.transformer.h.7.attn.attention.k_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.7.attn.attention.v_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.7.attn.attention.q_proj: W=8 (uniform) | A=8 (
    pact)
module.transformer.h.7.attn.attention.out_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.7.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.7.mlp.c_proj: W=8 (uniform) | A=8 (lsq)

module.transformer.h.8.attn.attention.k_proj: W=8 (lsq) | A=8 (lsq)
module.transformer.h.8.attn.attention.v_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.8.attn.attention.q_proj: W=8 (lsq) | A=8 (pact)
module.transformer.h.8.attn.attention.out_proj: W=4 (uniform) | A=8
    (lsq)
module.transformer.h.8.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.8.mlp.c_proj: W=8 (lsq) | A=8 (pact)

module.transformer.h.9.attn.attention.k_proj: W=8 (uniform) | A=32 (
    identity)
module.transformer.h.9.attn.attention.v_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.9.attn.attention.q_proj: W=8 (lsq) | A=32 (
    identity)
module.transformer.h.9.attn.attention.out_proj: W=8 (uniform) | A=8
    (lsq)
module.transformer.h.9.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
```

```
module.transformer.h.9.mlp.c_proj: W=8 (lsq) | A=8 (uniform)

module.transformer.h.10.attn.attention.k_proj: W=8 (lsq) | A=8 (lsq)
module.transformer.h.10.attn.attention.v_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.10.attn.attention.q_proj: W=8 (lsq) | A=8 (pact)

module.transformer.h.10.attn.attention.out_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.10.mlp.c_fc: W=8 (lsq) | A=8 (uniform)
module.transformer.h.10.mlp.c_proj: W=8 (uniform) | A=8 (uniform)

module.transformer.h.11.attn.attention.k_proj: W=8 (lsq) | A=8 (pact)

module.transformer.h.11.attn.attention.v_proj: W=8 (lsq) | A=8 (pact)

module.transformer.h.11.attn.attention.q_proj: W=8 (lsq) | A=8 (pact)

module.transformer.h.11.attn.attention.out_proj: W=8 (lsq) | A=8 (
    uniform)
module.transformer.h.11.mlp.c_fc: W=8 (lsq) | A=32 (identity)
module.transformer.h.11.mlp.c_proj: W=8 (lsq) | A=32 (identity)
```

Статья поступила 26 апреля 2026 г.

Quantization of language models via differentiable neural architecture search

D. N. Davydova

Neural architecture search is a method for automatic selection of an optimal neural network architecture for a given task based on a provided dataset. One of the approaches to neural architecture search is Differentiable architecture search (DARTS). DARTS transforms the discrete search space into a continuous one, enabling the use of gradient-based optimization to learn the architectural parameters of the network. In this paper, we investigate the applicability of DARTS for selecting quantization schemes and bit-widths for different components of generative language models, and present the results of the conducted experiments. The code for training and evaluation of the quantized model is available at: https://github.com/daria1d/Darts-QAT.

Keywords: natural language processing, neural architecture search, quantization, DARTS.

References

[1] A. E. Eiben, J. E. Smith, *Introduction to evolutionary computing*, Springer, Berlin, Heidelberg, 2015, ISBN: 9783662448731,3662448748. DOI: 10.1007/978-3-662-44874-8, 287 pp.

[2] B. Zoph, Q. V. Le, "Neural architecture search with reinforcement learning", *International Conference on Learning Representations (ICLR)*, 2017 https://arxiv.org/abs/1611.01578.

[3] M. U. Nasir, S. Earle, J. Togelius, S. James, C. Cleghorn, "LLMatic: Neural architecture search via large language models and quality diversity optimization", *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Association for Computing Machinery, New York, NY, USA, 2024, 1110–1118. DOI: 10.1145/3638529.3654017.

[4] A. Sarah, S. N. Sridhar, M. Szankin, S. Sundaresan, "LLaMA-NAS: Efficient neural architecture search for large language models", *European Conference on Computer Vision (ECCV)*, Springer Nature Switzerland, Cham, Switzerland, 2024, 67–74. DOI: 10.48550/arXiv.2405.18377.

[5] H. Liu, K. Simonyan, Y. Yang, "DARTS: Differentiable architecture search", *International Conference on Learning Representations (ICLR)*, 2019 https://arxiv.org/abs/1806.09055.

[6] B. Wu, Y. Wang, P. Zhang, Y. Tian, "Mixed precision quantization of convnets via differentiable neural architecture search", *arXiv preprint*, 2018 https://arxiv.org/abs/1812.00090.

[7] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, 770–778. DOI: 10.48550/arXiv.1512.03385.

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications", *arXiv preprint*, 2017 https://arxiv.org/abs/1704.04861.

[9] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *International Conference on Learning Representations (ICLR)*, 2015 https://arxiv.org/abs/1409.1556.

[10] Z. Cai, N. Vasconcelos, "Rethinking differentiable search for mixed-precision neural networks", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, 2349–2358. DOI: 10.48550/arXiv.2004.05795.

[11] P. Behnam, U. Kamal, S. Vijay Ganesh, Z. Li, M. A. Jurado, A. Khare, I. Fedorov, G. Liu, A. Tumanov, "∇QDARTS: Quantization as an Elastic Dimension to Differentiable NAS", *Transactions on Machine Learning Research*, 2025 https://openreview.net/forum?id=ubrOSWyTS8.

[12] M. Tan, Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks", *International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research, 2019, 6105–6114 https://arxiv.org/abs/1905.11946.

[13] C. Zhao, T. Hua, Y. Shen, Q. Lou, H. Jin, "Automatic mixed-precision quantization search of BERT", *arXiv preprint*, 2021 https://arxiv.org/abs/2112.14938.

[14] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019 https://arxiv.org/abs/1810.04805.

[15] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding", *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, 2018, 353–355. DOI: 10.18653/v1/W18-5446.

[16] J. Choi, Z. Wang, S. Venkataramani, P. Chuang, V. Srinivasan, K. Gopalakrishnan, "PACT: Parameterized clipping activation for quantized neural networks", *arXiv preprint*, 2018 https://arxiv.org/abs/1805.06085.

[17] Y. Bhalgat, J. Lee, M. Nagel, T. Blankevoort, N. Kwak, "LSQ+: Improving low-bit quantization through learnable offsets and better initialization", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, 696–697. DOI: 10.1109/CVPRW50498.2020.00356.

[18] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, 2704–2713. DOI: 10.1109/CVPR.2018.00286.

[19] Y. Bengio, N. Léonard, A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation", *arXiv preprint*, 2013 https://arxiv.org/abs/1308.3432.

[20] S. Black, G. Leo, P. Wang, C. Leahy, S. Biderman, "GPT-Neo: Large scale autoregressive language modeling with mesh-tensorflow", *Zenodo*, 2021. DOI: 10.5281/zenodo.5297715.

[21] L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, C. Leahy, "The Pile: An 800GB dataset of diverse text for language modeling", *arXiv preprint*, 2020 https://arxiv.org/abs/2101.00027.

[22] S. Merity, C. Xiong, J. Bradbury, R. Socher, "Pointer sentinel mixture models", *arXiv preprint*, 2016 https://arxiv.org/abs/1609.07843.

[23] "FineWeb-EDU dataset", *Hugging Face Datasets*, 2024 https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu.

[24] "OpenWebText Corpus", *GitHub repository*, 2019 https://github.com/skylion007/OpenWebTextCorpus.

[25] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, "Language models are unsupervised multitask learners", *OpenAI Technical Report*, 2019 https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[26] J. W. Rae, A. Potapenko, S. M. Jayakumar, C. Hillier, "Compressive transformers for long-range sequence modelling", *International Conference on Learning Representations (ICLR)*, 2020 https://arxiv.org/abs/1911.05507.

[27] M. Hart, "Project Gutenberg", 1971 https://www.gutenberg.org.

[28] Z. Liu, B. Oguz, C. Zhao, E. Chang, P. Stock, Y. Mehdad, Y. Shi, R. Krishnamoorthi, V. Chandra, "LLM-QAT: Data-free quantization aware training for large language models", *Findings of the Association for Computational Linguistics: ACL 2024*, 2024, 467–484. DOI: 10.18653/v1/2024.findings-acl.26.

[29] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs", *Advances in Neural Information Processing Systems*, **36**, 2023, 10088–10115 https://arxiv.org/abs/2305.14314.

[30] T. Dettmers, M. Lewis, Y. Belkada, L. Zettlemoyer, "LLM.int8(): 8-bit matrix multiplication for transformers at scale", *arXiv preprint*, 2022. DOI: 10.48550/arXiv.2208.07339.

Received on April 26, 2026

Метод распознавания изображений на основе выпуклой линейной оболочки над конечным полем

Е. С. Ковалёва*

Метод построения выпуклых линейных оболочек конечного множества точек часто используется в задачах распознавания. В $n-$мерном пространстве для этого перебираются n элементные подмножества точек. Если оставшиеся точки лежат по одну сторону от гиперплоскости, содержащей выбранные точки, то эта гиперплоскость называется гранью. Проверка новой точки на принадлежность выпуклой линейной оболочке сводится к проверке выполнения системы линейных неравенств, образованных гранями.

Предлагается решать огрублённую задачу, когда координаты точек принимают три значения $0, 1, -1$, интерпретируемые как элементы поля из трёх элементов. При этом линейная функция равна нулю в точках, лежащих на гиперплоскости, равна 1 с одной стороны от неё, и равна -1 с другой стороны. Проверка принадлежности точки выпуклой линейной оболочке сводится к проверке условий $S_a(x) \in \{0, \lambda_a\}$ для всех граней S_a выпуклой линейной оболочки, где значение 0 соответствует принадлежности точки гиперплоскости S_a, а λ_a является общим значением линейной формы на точках множества, не принадлежащих S_a. Таким образом, принадлежность новой точки выпуклой линейной оболочке определяется её положением относительно всех граней оболочки.

Ключевые слова: выпуклая линейная оболочка, конечные поля, пространство E_3^n, линейные формы, распознавание образов, классификация изображений.

1. Введение

Задачи распознавания образов и компьютерного зрения являются одним из наиболее активно развивающихся направлений современной информатики и прикладной математики [1–4]. Одной из важных задач данной области является определение состояния глаз человека по цифровому

* *Ковалёва Елена Сергеевна* — аспирант каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: e.kovaleva.msu@yandex.ru, ORCID: 0009-0004-7275-3076.

Kovaleva Elena Sergeevna — PhD student, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intellectual Systems.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

изображению. Распознавание открытых и закрытых глаз применяется в системах контроля внимания водителей, биометрической идентификации личности, анализе поведения человека и интеллектуальных интерфейсах взаимодействия человека с вычислительными системами [5–7].

Предложенный подход не требует обучения параметров модели, настройки весовых коэффициентов или решения задач оптимизации. После построения оболочек классификация выполняется посредством проверки принадлежности объектов соответствующим выпуклым линейным оболочкам. Вычислительный эксперимент показал точность распознавания 96.05% для класса закрытых глаз и 97.92% для класса открытых глаз, что свидетельствует о возможности практического применения предложенного метода в задачах распознавания состояний глаз.

2. Основные определения

Пусть $E_3 = \mathbb{Z}_3 = \{0, 1, 2\}$ является конечным полем из трёх элементов, а E_3^n — n-мерным линейным пространством над полем E_3.

Определение 1. *Гиперплоскостью в пространстве* E_3^n называется множество точек

$S_a = \{x = (x_1, \ldots, x_n) \in E_3^n \mid a_1x_1 + \cdots + a_nx_n + a_{n+1} \equiv 0 \pmod 3\}$, где $a = (a_1, \ldots, a_n, a_{n+1}) \in E_3^{n+1}$ и $(a_1, \ldots, a_n) \neq (0, \ldots, 0)$.

Определение 2. Пусть $X \subset E_3^n$ — конечное множество точек, S_a - гиперплоскость в пространстве E_3^n, $X \setminus S_a = \{x \in X \mid x \notin S_a\}$. S_a называется *гранью выпуклой линейной оболочки* множества X, если выполняются следующие условия:

1. на гиперплоскости S_a лежит не менее n точек множества X, то есть $|X \cap S_a| \geqslant n$;
2. если $X \setminus S_a \neq \varnothing$, то существует число $\lambda_a \in \{1, 2\}$ такое, что для любой точки $x = (x_1, \ldots, x_n) \in X$ выполняется уравнение $a_1x_1 + \cdots + a_nx_n + a_{n+1} \in \{0, \lambda_a\} \pmod 3$;
3. если $X \setminus S_a = \varnothing$, то для любой точки $x = (x_1, \ldots, x_n) \in X$ выполняется уравнение $a_1x_1 + \cdots + a_nx_n + a_{n+1} \equiv 0 \pmod 3$.

Точки $z \in E_3^n$, удовлетворяющие условию

$$S_a(z) = \lambda_a$$

для каждой грани S_a выпуклой линейной оболочки множества X, будем называть *внутренними точками* выпуклой линейной оболочки множества X.

Определение 3. *Выпуклой линейной оболочкой* конечного множества точек $X \subset E_3^n$ называется множество

$$\mathrm{conv}(X) = \{x \in E_3^n : S_a(x) \in \{0, \lambda_a\} \text{ для каждой грани } S_a\}.$$

Определение 4. Выпуклая линейная оболочка $\mathrm{conv}(X)$ называется *непустой*, если её внутренность непуста. В противном случае выпуклая линейная оболочка называется *пустой*.

Введем обозначения.

Пусть S - множество граней выпуклой линейной оболочки,

S_i – $i-$я грань из множества S,

$Z_i = X \setminus S_i$ — множество точек множества X, не принадлежащих грани S_i,

$S_i(Z_i)-$ значение точек из множества Z_i в уравнении грани S_i,

$Z = (Z_1, Z_2, ..., Z_p)$, где $p-$ количество граней,

$S(x) = (S_1(x), S_2(x), \ldots, S_p(x))$, где $x \in E_3^n$.

Рассмотрим примеры выпуклых линейных оболочек.

Пример 1. Рассмотрим множество точек $X = \{(0,0), (0,1), (0,2), (1,0)\}$.

Кандидатами на грани являются гиперплоскости

$x_1 \equiv 0 \pmod 3$

$x_2 \equiv 0 \pmod 3$

$x_1 + x_2 + 2 \equiv 0 \pmod 3$

$x_1 + 2x_2 + 2 \equiv 0 \pmod 3$

Проверка по определению 2 показывает, что условию удовлетворяет только гиперплоскость $x_1 \equiv 0 \pmod 3$. Следовательно, получаем кортеж граней $S = (x_1 \equiv 0 \pmod 3)$.

Проверим оболочку на пустоту.

Для единственной грани S_1 имеем $Z_1 = X \setminus S_1 = \{(1,0)\}$. Следовательно, $S_1(Z_1) = \lambda_1 = 1$.

Рассмотрим множество кандидатов $Y = \{(1,1), (1,2), (2,0), (2,1), (2,2)\}$ на внутренние точки выпуклой линейной оболочки множества X.

Грань S_i	$S_i(Z_i)$	$S_i(1,1)$	$S_i(1,2)$	$S_i(2,0)$	$S_i(2,1)$	$S_i(2,2)$
x_1	1	1	1	2	2	2

Так как $S_1(Z_1) = \lambda_1 = 1$ и $S_1(1,1) = S_1(1,2) = 1 = \lambda_1$ а для остальных точек множества Y $S_1(2,0) = S_1(2,1) = S_1(2,2) = 2 \neq \lambda_1$, то внутренними точками оболочки являются $(1,1)$ и $(1,2)$. Следовательно, выпуклая линейная оболочка множества $\{(0,0), (0,1), (0,2), (1,0)\}$ является непустой.

Пример 2. Рассмотрим множество

$$X = \{(0,0),(0,1),(0,2),(1,0),(1,1),(1,2)\}.$$

Кандидатами на грани являются гиперплоскости
$x_1 \equiv 0 \pmod 3$
$x_1 + 2 \equiv 0 \pmod 3$
$x_2 \equiv 0 \pmod 3$
$x_2 + 2 \equiv 0 \pmod 3$
$x_2 + 1 \equiv 0 \pmod 3$
$x_1 + x_2 \equiv 0 \pmod 3$
$x_1 + x_2 + 2 \equiv 0 \pmod 3$
$x_1 + x_2 + 1 \equiv 0 \pmod 3$
$x_1 + 2x_2 \equiv 0 \pmod 3$
$x_1 + 2x_2 + 2 \equiv 0 \pmod 3$
$x_1 + 2x_2 + 1 \equiv 0 \pmod 3$
Проверка по определению 2 показывает, что гранями являются
$S_1 = x_1$
$S_2 = x_1 + 2$
Следовательно, кортеж граней выпуклой линейной оболочки
$S = (x_1 \equiv 0 \pmod 3,\ x_1 + 2 \equiv 0 \pmod 3)$.
Проверим оболочку на пустоту.

Для первой грани S_1 имеем $Z_1 = X \setminus S_1 = \{(1,0),(1,1),(1,2)\}$. Следовательно, $S_1(Z_1) = \lambda_1 = 1$.

Для второй грани S_2 имеем $Z_2 = X \setminus S_2 = \{(0,0),(0,1),(0,2)\}$. Следовательно, $S_2(Z_2) = \lambda_2 = 2$.

Следовательно, $S(Z) = \big(S_1(Z_1), S_2(Z_2)\big) = (1,2)$.

Рассмотрим множество кандидатов $Y = \{(2,0),(2,1),(2,2)\}$ на внутренние точки выпуклой линейной оболочки множества X.

Грань S_i	$S_i(Z_i)$	$S_i(2,0)$	$S_i(2,1)$	$S_i(2,2)$
x_1	1	2	2	2
$x_1 + 2$	2	1	1	1

Для первой грани S_1 имеем $\lambda_1 = 1$. Для всех точек $y \in Y$ выполняется $S_1(y) = 2 \neq \lambda_1$.

Для второй грани S_2 имеем $\lambda_2 = S_2(Z_2) = 2$. Для всех точек $y \in Y$ выполняется $S_2(y) = 1 \neq \lambda_2$.

Следовательно, для каждой грани S_i ни одна точка $y \in Y$ не удовлетворяет условию $S_i(y) = \lambda_i \Rightarrow$ выпуклая линейная оболочка множества

$$X = \{(0,0),(0,1),(0,2),(1,0),(1,1),(1,2)\}$$

является пустой.

3. Вычислительный эксперимент

Для проверки работоспособности предложенного метода был проведён вычислительный эксперимент по распознаванию состояний глаз. В качестве исходных данных использовались изображения двух классов: открытые глаза и закрытые глаза. Размеры обучающих выборок составляли $|X_{\text{open}}| = 96$ и $|X_{\text{close}}| = 93$. Предварительная обработка изображений выполнялась с использованием библиотеки `dlib` [8] и алгоритма локализации характерных точек лица. Пример разметки глаз с помощью библиотеки `dlib` показан на рисунке 1.



Рис. 1: Пример разметки характерных точек глаз с использованием библиотеки `dlib`

Шаг 1. Формирование обучающих и тестовых выборок.

Для всех обучающих и тестовых изображений [9] выполнялась предварительная обработка. Изображения переводились в цветовое пространство RGB, после чего с использованием библиотеки `dlib` [8] осуществлялось обнаружение лица и определение координат 68 антропометрических точек. Изображения, на которых лицо не было обнаружено, исключались из дальнейшего рассмотрения. В эксперименте таких изображений обнаружено не было. Для формирования признакового описания использовались характерные точки области глаз с номерами 37, 38, 40, 41, 43, 44, 47 и 48, определяемые с помощью библиотеки dlib.

Координаты каждой ключевой точки предварительно нормировались.

Для каждого изображения ширины W и высоты H координаты характерных точек глаз предварительно нормируются относительно размеров

изображения:

$$\widetilde{x}_i = \frac{x_i}{W}, \qquad \widetilde{y}_i = \frac{y_i}{H},$$

где (x_i, y_i) — исходные координаты i-й характерной точки, полученной с помощью библиотеки dlib.

В работе используются точки с номерами

$$37,\ 38,\ 40,\ 41,\ 43,\ 44,\ 47,\ 48.$$

На основе нормированных координат формируется вектор

$$\Delta = (\Delta_1, \Delta_2, \ldots, \Delta_8),$$

где

$$\begin{aligned} \Delta_1 &= \widetilde{x}_{41} - \widetilde{x}_{37}, \quad \Delta_2 = \widetilde{y}_{41} - \widetilde{y}_{37}, \\ \Delta_3 &= \widetilde{x}_{40} - \widetilde{x}_{38}, \quad \Delta_4 = \widetilde{y}_{40} - \widetilde{y}_{38}, \\ \Delta_5 &= \widetilde{x}_{47} - \widetilde{x}_{43}, \quad \Delta_6 = \widetilde{y}_{47} - \widetilde{y}_{43}, \\ \Delta_7 &= \widetilde{x}_{48} - \widetilde{x}_{44}, \quad \Delta_8 = \widetilde{y}_{48} - \widetilde{y}_{44}. \end{aligned}$$

Далее для каждого изображения вычисляются

$$\Delta_{\min} = \min_{1 \leqslant j \leqslant 8} \Delta_j, \qquad \Delta_{\max} = \max_{1 \leqslant j \leqslant 8} \Delta_j.$$

Если $\Delta_{\max} \neq \Delta_{\min}$, то каждый компонент вектора линейно нормируется на отрезок $[-1, 1]$ по формуле

$$z_j = -1 + 2\frac{\Delta_j - \Delta_{\min}}{\Delta_{\max} - \Delta_{\min}}, \qquad j = 1, \ldots, 8.$$

Если $\Delta_{\max} = \Delta_{\min}$, то каждый компонент вектора линейно нормируется на отрезок $[-1, 1]$ по формуле

$$z_j = 0, \qquad j = 1, \ldots, 8.$$

После этого каждое значение z_j округляется до ближайшего целого числа: $q_j = \mathrm{round}(z_j), \qquad q_j \in \{-1, 0, 1\}$.

Наконец, для перехода к элементам конечного поля $E_3 = \mathbb{Z}_3 = \{0, 1, 2\}$ производится замена $-1 \longmapsto 2$.

Таким образом, каждому изображению ставится в соответствие вектор

$$q = (q_1, q_2, \ldots, q_8) \in E_3^8.$$

По сформированным векторам были построены обучающие матрицы признаков $train_o$ и $train_c$ для изображений с открытыми и закрытыми глазами соответственно, а также тестовые матрицы $test_o$ и $test_c$.

В результате эксперимента для обучающей выборки закрытых глаз была получена следующая матрица уникальных векторов признаков:

$$train_{\text{close}} = \begin{pmatrix} 0 & 2 & 1 & 2 & 0 & 2 & 2 & 2 \\ 0 & 2 & 1 & 2 & 1 & 2 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 2 & 2 \\ 1 & 0 & 1 & 2 & 1 & 0 & 2 & 0 \\ 1 & 0 & 1 & 2 & 1 & 0 & 2 & 2 \\ 1 & 2 & 1 & 2 & 1 & 0 & 2 & 0 \\ 1 & 2 & 1 & 2 & 1 & 0 & 2 & 2 \\ 1 & 2 & 1 & 2 & 1 & 2 & 2 & 0 \\ 1 & 2 & 1 & 2 & 1 & 2 & 2 & 2 \\ 1 & 2 & 1 & 2 & 2 & 2 & 0 & 0 \end{pmatrix}.$$

И для обучающей выборки открытых глаз:

$$train_{\text{open}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 2 & 2 & 0 \\ 0 & 2 & 1 & 0 & 1 & 2 & 2 & 0 \\ 0 & 2 & 1 & 2 & 1 & 0 & 2 & 0 \\ 0 & 2 & 1 & 2 & 1 & 2 & 2 & 0 \\ 0 & 2 & 1 & 2 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 0 & 1 & 1 & 2 & 0 \\ 1 & 0 & 1 & 0 & 1 & 2 & 2 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 2 & 0 \\ 1 & 2 & 1 & 0 & 0 & 2 & 2 & 2 \\ 1 & 2 & 1 & 0 & 1 & 2 & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 & 2 & 2 & 0 \\ 1 & 2 & 1 & 1 & 1 & 2 & 2 & 0 \\ 1 & 2 & 1 & 2 & 1 & 2 & 0 & 0 \end{pmatrix}.$$

Шаг 2. Построение выпуклых линейных оболочек.

В рассматриваемой задаче размерность пространства признаков составляла $n = 8$. На первом этапе эксперимента для каждого класса независимо строилось множество граней выпуклой линейной оболочки.

Пусть $X = \{x^{(1)}, x^{(2)}, \ldots, x^{(N)}\} \subset E_3^n$.

Для построения граней выпуклой линейной оболочки перебираются векторы коэффициентов $a = (a_1, \ldots, a_n, a_{n+1}) \in E_3^{n+1}$, задающие гиперплоскости $S_a = \{x = (x_1, \ldots, x_n) \in E_3^n : a_1x_1 + \cdots + a_nx_n + a_{n+1} = 0$

(mod 3)}. Затем для каждой гиперплоскости S_a проверялось выполнение условий, определяющих грань выпуклой линейной оболочки множества X. Для этого для каждой точки $x \in X$ вычислялось значение линейной формы, задающей гиперплоскость S_a, по модулю 3.

В результате были получены множества $G_{\text{open}} = G(train_{\text{open}})$, содержащее 57 граней и $G_{\text{close}} = G(train_{\text{close}})$, содержащее 63 грани, задающие выпуклые линейные оболочки соответствующих классов.

Шаг 3. Проверка непустоты построенных оболочек.

После построения множеств граней для каждого класса выполнялся поиск внутренних точек выпуклых линейных оболочек, не принадлежащих исходным обучающим множествам. Результаты показали, что такие точки существуют для обоих классов.

По результатам эксперимента число внутренних точек выпуклой линейной оболочки открытых глаз составляет 23, а число внутренних точек выпуклой линейной оболочки закрытых глаз составляет 31.

Шаг 4. Проверка пересечения классов точек открытых и закрытых глаз.

Было проведено пересечение оболочек открытых и закрытых глаз для того, чтобы избежать случаев вхождения точек из тестовых выборок одновременно в обе оболочки. В рассматриваемом вычислительном эксперименте общих точек выпуклых линейных оболочек классов открытых и закрытых глаз обнаружено не было.

Случаи, когда классифицируемый объект одновременно принадлежит обеим оболочкам или не принадлежит ни одной из них, в рамках настоящей работы не рассматривались. Разработка правила принятия решения для таких случаев является направлением дальнейшего развития предложенного метода.

Шаг 5. Вычисление точности распознавания открытых и закрытых глаз.

Для каждой тестовой точки вычислялись значения линейных форм, задающих грани соответствующей выпуклой линейной оболочки, после чего выполнялась проверка совпадения полученных значений со значениями, наблюдаемыми на обучающих объектах данного класса.

Тестирование проводилось на независимых выборках изображений. Для класса закрытых глаз использовалось 76 тестовых изображений, из которых 73 были распознаны как принадлежащие выпуклой линейной оболочке класса закрытых глаз. Следовательно, точность распознавания составила $Acc_{\text{close}} = \frac{73}{76} \cdot 100\% = 96.05\%$. Для класса открытых глаз использовалось 48 тестовых изображений, из которых 47 были распознаны как принадлежащие выпуклой линейной оболочке класса открытых глаз.

Таким образом, точность распознавания составила $Acc_{\text{open}} = \frac{47}{48} \cdot 100\% = 97.92\%$.

Для исследования геометрических свойств построенных выпуклых линейных оболочек был проведён дополнительный эксперимент. В рамках эксперимента для каждого класса вычислялись две характеристики: доля правильно классифицированных тестовых объектов, лежащих внутри выпуклой линейной оболочки соответствующего класса, и доля правильно классифицированных тестовых объектов, совпадающих с элементами обучающей выборки.

Для класса закрытых глаз знаменателем при вычислении этих долей являлось число правильно классифицированных тестовых объектов, равное 73. Результаты для класса закрытых глаз представлены в таблице 1.

Доля правильно классифицированных точек из test_{close}, лежащих внутри выпуклой линейной оболочки обучающего множества train_{close}	64.38%
Доля правильно классифицированных точек из test_{close}, совпадающих с точками обучающего множества train_{close}	35.62%

Таблица 1: Принадлежность правильно классифицированных тестовых объектов выпуклой линейной оболочке для класса закрытых глаз

Для класса открытых глаз знаменателем при вычислении этих долей являлось число правильно классифицированных тестовых объектов, равное 47. Результаты для класса открытых глаз представлены в таблице 2.

Доля правильно классифицированных точек из test_{open}, лежащих внутри выпуклой линейной оболочки обучающего множества train_{open}	55.32%
Доля правильно классифицированных точек из test_{open}, совпадающих с точками обучающего множества train_{open}	44.68%

Таблица 2: Принадлежность правильно классифицированных тестовых объектов выпуклой линейной оболочке для класса открытых глаз

Полученные результаты имеют принципиальное значение для обоснования предложенного метода. Если бы распознавание фактически сводилось к сравнению тестовых объектов с элементами обучающей выборки, то доля объектов, принадлежащих выпуклой линейной оболочке, совпадала бы с долей объектов, присутствующих в обучающей выборке.

Однако результаты эксперимента показывают существенное различие между этими величинами. Для класса закрытых глаз разница составляет $64.38 - 35.62 = 28.76$ процентных пунктов, а для класса открытых глаз — $55.32 - 44.68 = 10.64$ процентных пунктов.

Следовательно, значительная часть тестовых объектов принадлежит выпуклым линейным оболочкам, не совпадая при этом с объектами обучающей выборки. Это означает, что алгоритм использует не только обучающие точки, но и дополнительные точки пространства E_3^8, принадлежащие внутренней области построенных выпуклых линейных оболочек.

Совместный анализ результатов проверки непустоты оболочек и результатов дополнительного эксперимента показывает, что тестовые объекты располагаются как на гранях построенных оболочек, так и внутри них. Таким образом, в данном примере принятие решения осуществляется на основе геометрической структуры выпуклой линейной оболочки, а не сводится к проверке совпадения с конечным набором обучающих объектов.

Полученные результаты подтверждают работоспособность предложенного метода распознавания и демонстрируют наличие у построенных выпуклых линейных оболочек обобщающей способности по отношению к новым объектам. Кроме того, предложенный подход сохраняет полную геометрическую интерпретируемость процесса принятия решения, поскольку классификация определяется свойствами граней выпуклой линейной оболочки и расположением объектов относительно них.

4. Выводы

В работе предложен метод распознавания открытых и закрытых глаз, основанный на использовании выпуклых линейных оболочек в пространстве E_3^n. Были введены определения гиперплоскости в дискретном пространстве, грани выпуклой линейной оболочки и выпуклой линейной оболочки конечного множества точек. На основе данных определений разработан алгоритм распознавания изображений глаз. Предложенный подход применён к задаче классификации состояний глаз.

Научная новизна работы заключается в построении модели выпуклой линейной оболочки над конечным полем E_3, а также в использовании данной модели для решения прикладной задачи распознавания изображений.

Проведённый вычислительный эксперимент показал работоспособность предложенного метода. Для класса закрытых глаз была получена точность 96.05%, а для класса открытых глаз — 97.92%. Обозначим через

N_o и N_c количества различных дискретизированных векторов объектов обучающей выборки открытых и закрытых глаз соответственно, непосредственно используемых при построении граней, а через n — размерность пространства признаков. Вычислительная сложность построения граней для двух классов оценивается как $O\left((N_o + N_c)n3^{n+1}\right)$. В проведённом эксперименте $n = 8$, $N_o = 13$ и $N_c = 11$, поэтому оценка принимает вид $O\left((13 + 11) \cdot 8 \cdot 3^9\right)$. При фиксированном n сложность линейна относительно числа различных обучающих векторов $N_o + N_c$, однако относительно размерности n она является экспоненциальной. Сильной стороной метода является отсутствие этапа обучения параметров, детерминированность вычислений и возможность геометрической интерпретации результата. Ограничением является экспоненциальная зависимость от размерности пространства признаков, вследствие чего применение метода наиболее целесообразно для задач с малой или фиксированной размерностью.

Список литературы

[1] R. Duda, P. Hart, D. Stork, *Pattern Classification. 2nd ed.*, John Wiley & Sons, New York, NY, USA, 2001, 654 pp.

[2] C. Bishop, *Pattern Recognition and Machine Learning*, Springer New York, NY, New York, NY, USA, 2006, 778 pp.

[3] R. Szeliski, *Computer Vision: Algorithms and Applications. 1st ed.*, Springer, London, 2011, 812 pp.

[4] R. Gonzalez, R. Woods, *Digital Image Processing*, Prentice Hall, Upper Saddle River, NJ, USA, 2008, 954 pp.

[5] K. Kim, H. Hong, G. Nam, K. Park, "A Study of Deep CNN-Based Classification of Open and Closed Eyes Using a Visible Light Camera Sensor", *Sensors*, **17**:7 (2017), 1534. DOI: 10.3390/s17071534.

[6] I. Kayadibi, G. E. Güraksın, U. Ergün, N. Özmen Süzme, "An Eye State Recognition System Using Transfer Learning: AlexNet-Based Deep Convolutional Neural Network", *International Journal of Computational Intelligence Systems*, **15** (2022), 49. DOI: 10.1007/s44196-022-00108-2.

[7] R. Florez, F. Palomino-Quispe, R. Coaquira-Castillo, J. Herrera-Levano, T. Paixao, A. Alvarez, "A CNN-Based Approach for Driver Drowsiness Detection by Real-Time Eye State Identification", *Applied Sciences*, **13**:13 (2023), 7849. DOI: 10.3390/app13137849.

[8] Davis E. King, *Dlib C++ Library*, documentation, accessed 10.03.2026, `http://dlib.net/`.

[9] E. Kovaleva, *Faces with Open/Closed Eyes*, accessed 10.03.2026, `https://www.kaggle.com/datasets/helen647135486/faces-with-open-closed-eyes`.

Статья поступила 28 июня 2026 г.

A finite-field convex linear hull-based recognition method

E. S. Kovaleva

The method of constructing convex linear hulls of a finite set of points is often used in recognition problems. In an n-dimensional space, n-element subsets of points are considered for this purpose. If the remaining points lie on the same side of the hyperplane containing the selected points, then this hyperplane is called a face. Checking whether a new point belongs to the convex linear hull is reduced to verifying a system of linear inequalities determined by the faces.

We propose considering a coarse-grained problem in which the coordinates of the points take three values, $0, 1, -1$, interpreted as elements of a field with three elements. In this setting, the linear form is equal to zero at points lying on the hyperplane, equal to 1 on one side of it, and equal to -1 on the other side. Checking whether a point belongs to the convex linear hull is reduced to verifying the conditions $S_a(x) \in \{0, \lambda_a\}$ for all faces S_a of the convex linear hull, where the value 0 corresponds to the point lying on the hyperplane S_a, while λ_a is the common value of the linear form at the points of the set that do not belong to S_a. Thus, whether a new point belongs to the convex linear hull is determined by its position relative to all faces of the hull.

Keywords: convex linear hull, finite fields, E_3^n space, linear forms, pattern recognition, image classification.

References

[1] R. Duda, P. Hart, D. Stork, *Pattern Classification. 2nd ed.*, John Wiley & Sons, New York, NY, USA, 2001, 654 pp.

[2] C. Bishop, *Pattern Recognition and Machine Learning*, Springer New York, NY, New York, NY, USA, 2006, 778 pp.

[3] R. Szeliski, *Computer Vision: Algorithms and Applications. 1st ed.*, Springer, London, 2011, 812 pp.

[4] R. Gonzalez, R. Woods, *Digital Image Processing*, Prentice Hall, Upper Saddle River, NJ, USA, 2008, 954 pp.

[5] K. Kim, H. Hong, G. Nam, K. Park, "A Study of Deep CNN-Based Classification of Open and Closed Eyes Using a Visible Light Camera Sensor", *Sensors*, **17**:7 (2017), 1534. DOI: 10.3390/s17071534.

[6] I. Kayadibi, G. E. Güraksın, U. Ergün, N. Özmen Süzme, "An Eye State Recognition System Using Transfer Learning: AlexNet-Based Deep Convolutional Neural Network", *International Journal of Computational Intelligence Systems*, **15** (2022), 49. DOI: 10.1007/s44196-022-00108-2.

[7] R. Florez, F. Palomino-Quispe, R. Coaquira-Castillo, J. Herrera-Levano, T. Paixao, A. Alvarez, "A CNN-Based Approach for Driver Drowsiness Detection by Real-Time Eye State Identification", *Applied Sciences*, **13**:13 (2023), 7849. DOI: 10.3390/app13137849.

[8] Davis E. King, *Dlib C++ Library*, documentation, accessed 10.03.2026, http://dlib.net/.

[9] E. Kovaleva, *Faces with Open/Closed Eyes*, accessed 10.03.2026, https://www.kaggle.com/datasets/helen647135486/faces-with-open-closed-eyes.

Received on June 28, 2026

О максимально сложно реализуемых отображениях

С. Б. Родин*

Данная работа продолжает изучение вопроса сложности реализуемости автоматов посредством кодирований алфавита состояний. Ранее изучался вопрос «простой реализуемости» как автоматов, так и отображений на конечном множестве. В данной работе изучаются «сложно реализуемые» отображения на конечном множестве, т. е. такие отображения, что любое неизбыточное кодирование приводит к булеву оператору максимальной сложности с точки зрения степени полинома Жегалкина. Доказано, что для любой мощности множества $n = 2^k$ существуют «сложно реализуемые» отображения. Существование таких отображений позволит в дальнейшем доказать существование «сложно реализуемых» автоматов.

Ключевые слова: теория автоматов, переходные системы, подстановка, кодирование, сложность, булев оператор.

1. Введение

На практике часто приходится решать задачу перехода от автоматного описания функционирования на язык схем. Например, при логическом синтезе чипов на первом этапе функционирование чипа описывается как конечный автомат. Переход к описанию на языке схем осуществляется с помощью кодирования алфавита состояний, входного алфавита и выходного алфавита в алфавите $E_2 = \{0, 1\}$. В результате кодирования возникает булев оператор. При этом автомат может обладать тем свойством, что каждое кодирование порождает оператор, отличный от оператора, порождаемого любым другим кодированием [1].

Возникающий в результате кодирования булев оператор можно рассматривать как набор булевых функций. Сложность такого оператора можно определить как максимальную сложность получающихся булевых функций. Как известно [2], каждой булевой функции единственным

* *Родин Сергей Борисович* — к.ф.-м.н., старший научный сотрудник каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: sergei_rodin@mail.ru.

Rodin Sergei Borisovich — Candidate of Physical and Mathematical Sciences, Senior research scientist, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intelligent Systems.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

образом соответствует полином Жегалкина. В статье [3] было предложено определить сложность как максимальную из сложностей полиномов Жегалкина функций, задающих этот оператор, т. е. как максимальную степень полиномов. Тогда простейшими с точки зрения такой сложности являются такие операторы, что соответствующие полиномы Жегалкина имеют первую степень, или линейные булевы функции. Интересно заметить, что максимальность мощности множества возникающих для автомата посредством кодирований операторов [1] не гарантирует существование «простой» в указанном выше смысле, реализации автомата [4].

В статье [3] был доказан критерий линейной реализуемости нумерованной переходной системы $V = (E_2, E_n, \varphi)$ посредством неизбыточного кодирования F. Данный критерий был сформулирован в терминах порождающих внутренней полугруппы переходной системы. Обозначим через p_0 отображение на n-элементном множестве, индуцированное входным символом 0, а через p_1 отображение на n-элементном множестве, индуцированное входным символом 1. Для линейной реализуемости переходной системы необходимыми и достаточными условиями являются, во-первых, линейная реализуемость отображений p_0 и p_1, во-вторых, выполнение свойства «аддитивного сдвига» [5] на отображениях p_0 и p_1 [3]. В работах [6] и [7] изучался вопрос линейной реализуемости отображений и было показано, что при удлинении кода любое отображение является линейно реализуемым. Однако, если ограничить длину кода минимально возможной, то сложность реализации отображения увеличится и, в общем случае, будет определяться выбором кодирования.

Данная работа посвящена вопросу существования «максимально» сложных отображений, т. е. таких что при любом кодировании оператор, построенный по отображению, имеет максимальную сложность в смысле полинома Жегалкина.

2. Основные понятия и определения

Основным объектом изучения являются отображения на множестве $E_n = \{0, \ldots, n-1\}$, где $n = 2^k$.

2.1. Булев оператор и его сложность

Сначала введем понятия, связанные с булевым оператором, и определим его сложность.

Определение 1. Пусть $\varphi : E_2^m \to E_2^k$ — булев оператор. Его можно рассматривать как набор k булевых функций, зависящих от m пе-

ременных, а именно, если $\varphi(\alpha_0, \alpha_1, \ldots, \alpha_{m-1}) = (\beta_0, \beta_1, \ldots, \beta_{k-1})$, то $f_j(\alpha_0, \alpha_1, \ldots, \alpha_{m-1}) = \beta_j$, где $0 \leqslant j \leqslant k-1$. Обозначим этот набор через $\mathcal{F}_\varphi = \big(f_0, f_1, \ldots, f_{k-1}\big)$.

Пример 1. Рассмотрим оператор φ, заданный таблицей

x_0	x_1	x_2	y_0	y_1	y_2
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	0	1	1

Тогда последние три столбца y_0, y_1, y_2 можно рассматривать как булевы функции f_0, f_1, f_2. Эти функции имеют следующий вид

$$
\begin{aligned}
f_0(x_0, x_1, x_2) =& x_0 \oplus x_1 \cdot x_2 \\
f_1(x_0, x_1, x_2) =& x_2 \oplus x_0 \cdot x_1 \oplus x_0 \cdot x_2 \oplus x_1 \cdot x_2 \oplus x_0 \cdot x_1 \cdot x_2 \\
f_2(x_0, x_1, x_2) =& 1 \oplus x_1 \oplus x_2 \oplus x_0 \cdot x_1 \oplus x_0 \cdot x_2 \oplus x_1 \cdot x_2 \oplus x_0 \cdot x_1 \cdot x_2
\end{aligned}
$$

Определение 2. Пусть $\mathcal{F} = \big(f_0, f_1, \ldots, f_{k-1}\big)$ — набор булевых функций, зависящих от m переменных. Данный набор определяет булев оператор $\varphi_\mathcal{F} : E_2^m \to E_2^k$ по правилу

$$
\begin{aligned}
\varphi_\mathcal{F}(\alpha_0, \alpha_1, \ldots, \alpha_{m-1}) =& (f_0(\alpha_0, \alpha_1, \ldots, \alpha_{m-1}), \\
& f_1(\alpha_0, \alpha_1, \ldots, \alpha_{m-1}), \\
& \ldots \\
& f_{k-1}(\alpha_0, \alpha_1, \ldots, \alpha_{m-1})),
\end{aligned}
$$

где $\alpha_i \in E_2$.

Пример 2. Пусть дана пара функций $f_0(x_0, x_1, x_2) = x_0 \oplus x_1 \cdot x_2$

x_0	x_1	x_2	f_0
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

и $f_1(x_0, x_1, x_2) = x_0 \cdot x_1 \oplus x_1 \cdot x_2 \oplus x_2$

x_0	x_1	x_2	f_1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Данные функции определяют булев оператор φ, задаваемый таблицей

x_0	x_1	x_2	y_0	y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	1

Определение 3. Пусть $\varphi : E_2^m \to E_2^k$ — булев оператор. *Сложностью оператора* назовем максимальную степень полиномов Жегалкина функций $\mathcal{F}_\varphi$ или $L_{\deg}(\varphi) = \max_{f_i \in \mathcal{F}_\varphi}\{\deg f_i\}$.

Заметим, что сложность оператора из примера 1 равна 3 и сложность оператора из примера 2 равна 2.

2.2. Реализуемость отображения посредством кодирования

От отображения к булеву оператору можно перейти с помощью кодирования. Сначала определим кодирование, а затем покажем как с помощью кодирования получается булев оператор.

Определение 4. *Кодированием множества* $E_n = \{0, \dots, n-1\}$ назовем взаимно-однозначное отображение (вложение) $F : \{0, \dots, n-1\} \to E_2^k$, где $k \geqslant]\log_2 n[$.

В данной работе будут изучаться только неизбыточные кодирования, т. е. кодирования, у которых $k =]\log_2 n[$.

Пример 3. В качестве примера неизбыточного кодирования можно рассмотреть следующее отображение E_8 в E_2^3:

q	0	1	2	3	4	5	6	7
$F(q)$	001	010	000	100	101	110	111	011

Выделим из всех кодирований «стандартное » кодирование.

Определение 5. Кодирование $F_0 : \{0, \dots, n-1\} \to E_2^k$ назовем *стандартным*, если код элемента есть его двоичное представление.

Пример 4. В качестве примера стандартного кодирования можно рассмотреть следующее отображение E_8 в E_2^3:

q	0	1	2	3	4	5	6	7
$F_0(q)$	000	001	010	011	100	101	110	111

Каждому кодированию F можно сопоставить подстановку s_F на множестве $E_n = \{0, \dots, n-1\}$ по правилу $s_F(i) = F_0^{-1}(F(i))$.

Кодированию F из примера 3 соответствует подстановка

$$s_F = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 2 & 0 & 4 & 5 & 6 & 7 & 3 \end{pmatrix}$$

Каждой подстановке $s : E_n \to E_n$ сопоставим кодирование F_s по правилу $F_s(i) = F_0(s(i))$.

Замечание 1. *Заметим, что для неизбыточного кодирования F верно $F_{s_F} = F$.*

Замечание 2. *Заметим, что для подстановки s верно $s_{F_s} = s$.*

Пример 5. Пусть задана подстановка

$$s = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 2 & 0 & 4 & 5 & 6 & 7 & 3 \end{pmatrix}$$

Данной подстановке соответствует кодирование

q	0	1	2	3	4	5	6	7
$F(q)$	001	010	000	100	101	110	111	011

Определение 6. Пусть $p : E_n \to E_n$ — отображение множества $E_n = \{0, \ldots, n-1\}$ в себя, где $n = 2^k$. Неизбыточное кодирование $F : E_n \to E_2^k$ множества E_n сопоставляет отображению p булев оператор $\varphi_p^F : E_2^k \to E_2^k$, по правилу

$$\varphi_p^F(\alpha_0, \ldots, \alpha_{k-1}) = F(p(F^{-1}(\alpha_0, \ldots, \alpha_{k-1}))),$$

где $\alpha_0, \ldots, \alpha_{k-1} \in E_2$.

Пример 6. Пусть задано отображение

$$p = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 5 & 6 & 7 & 2 & 3 & 4 \end{pmatrix}$$

Рассмотрим кодирование

q	0	1	2	3	4	5	6	7
$F(q)$	000	001	010	100	110	011	101	111

Построим булев оператор по отображению p с использованием кодирования F. Запишем отображение p в табличном виде.

i	$p(i)$
0	0
1	1
2	5
3	6
4	7
5	2
6	3
7	4

Заменим в таблице элементы множества E_8 на их коды, определяемые кодированием F. В результате получится следующий булев оператор φ

x_0	x_1	x_2	y_0	y_1	y_2
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
1	0	0	1	0	1
1	1	0	1	1	1
0	1	1	0	1	0
1	0	1	1	0	0
1	1	1	1	1	0

Определение 7. Пусть задано отображение $p : E_n \to E_n$ и неизбыточное кодирование F множества E_n. Степенной сложностью $L_{\deg}(p, F)$ реализации отображения p посредством кодирования F назовем сложность булева оператора φ_p^F, т. е. $L_{\deg}(p, F) = L_{\deg}(\varphi_p^F)$.

Определение 8. Пусть задано отображение $p : E_n \to E_n$. Обозначим через $\mathbb{F}^{irr}$ множество всех неизбыточных кодирований множества E_n. Степенной сложностью реализации отображения p назовем минимум среди степенных сложностей реализации отображения p посредством всех неизбыточных кодирований, т. е. $L_{\deg}^{irr}(p) = \min_{F \in \mathbb{F}^{irr}}\{L_{\deg}(p, F)\}$.

Если $n = 2^k$, то отображения множества $E_n = \{0, \ldots, n-1\}$ в себя могут быть представлены как многочлены над полем Галуа $\mathbb{F}_n$ [8].

Замечание 3. *Умножение отображений осуществляется «слева направо», т. е. если заданы отображения p_1 и p_2, то значение их произведения на элементе i определяется равенством $(p_1 \cdot p_2)(i) = p_2(p_1(i))$.*

3. Основные результаты

В работе [3] была доказана лемма

Лемма 1. *Пусть задано отображение p множества $E_n = \{0, \ldots, n-1\}$ в себя, где $n = 2^k$. Булев оператор, сопоставляемый кодированием F отображению p, совпадает с булевым оператором, который кодирование F_0 сопоставляет отображению $s_F^{-1} \cdot p \cdot s_F$.*

Данная лемма утверждает, что для получения всех булевых операторов для данного отображения вместо перебора всевозможных кодирований можно рассмотреть только стандартное кодирование для отображений, полученных сопряжением с помощью всевозможных перестановок, т. е. перестановка, задающая сопряжение, и кодирование в некотором смысле «эквивалентны». С помощью этой леммы докажем утверждение, что для двух отображений, являющихся сопряжением друг друга, можно найти кодирования, которые сопоставляют им одинаковый булев оператор, т. е. такие отображения порождают «одинаковое» множество операторов.

Лемма 2. *Пусть заданы отображение p на множестве E_n, взаимно-однозначное отображение $s : E_n \to E_n$ и неизбыточное кодирование F множества E_n. Тогда верно, что булев оператор φ_p^F, получаемый по отображению p посредством кодирования F, равен булеву оператору $\varphi_{s^{-1}\cdot p\cdot s}^{\tilde{F}}$, получаемому по отображению $s^{-1}\cdot p\cdot s$ посредством кодирования $\tilde{F} = F_{s^{-1}\cdot s_F}$.*

Доказательство. Согласно лемме 1 булев оператор, сопоставляемый неизбыточным кодированием F отображению p, совпадает с булевым оператором, который кодирование F_0 сопоставляет отображению $s_F^{-1}\cdot p\cdot s_F$.

Также согласно этому утверждению булев оператор, сопоставляемый неизбыточным кодированием $\tilde{F} = F_{s^{-1}\cdot s_F}$ отображению $s^{-1}\cdot p\cdot s$ совпадает с булевым оператором, который кодирование F_0 сопоставляет отображению $s_{\tilde{F}}^{-1}\cdot s^{-1}\cdot p\cdot s\cdot s_{\tilde{F}}$. Заметим, что $s_{\tilde{F}} = s^{-1}\cdot s_F$, так как кодирование $\tilde{F}$ определяется перестановкой $s^{-1}\cdot s_F$ (см. замечание 1).

Следовательно, верно, что

$$s_{\tilde{F}}^{-1}\cdot s^{-1}\cdot p\cdot s\cdot s_{\tilde{F}} = (s^{-1}\cdot s_F)^{-1}\cdot s^{-1}\cdot p\cdot s\cdot s^{-1}\cdot s_F =$$

$$= s_F^{-1}\cdot s\cdot s^{-1}\cdot p\cdot s\cdot s^{-1}\cdot s_F = s_F^{-1}\cdot p\cdot s_F.$$

Следовательно, булев оператор, сопоставляемый неизбыточным кодированием $\tilde{F} = F_{s^{-1}\cdot s_F}$ отображению $s^{-1}\cdot p\cdot s$ совпадает с булевым оператором, который кодирование F_0 сопоставляет отображению $s_F^{-1}\cdot p\cdot s_F$. Отсюда следует утверждение леммы. □

Пусть задано отображение p на множестве E_n. Обозначим через Φ_p^{irr} множество всех операторов, сопоставляемых всевозможными неизбыточными кодированиями отображению p.

Следствие 1. *Пусть заданы отображение p на множестве $E_n = \{0,\dots,n-1\}$ и взаимно-однозначное отображение $s : E_n \to E_n$. Множества булевых операторов Φ_p^{irr} и $\Phi_{\tilde{p}}^{irr}$, где $\tilde{p} = s^{-1}\cdot p\cdot s$, совпадают.*

Доказательство. Поскольку $\tilde{p} = s^{-1} \cdot p \cdot s$, то согласно лемме 2 $\Phi_p^{irr} \subseteq \Phi_{\tilde{p}}^{irr}$. Поскольку $p = s \cdot \tilde{p} \cdot s^{-1}$, то согласно лемме 2 $\Phi_{\tilde{p}}^{irr} \subseteq \Phi_p^{irr}$. А следовательно эти два множества равны. □

Определение 9. Весом булевой функции $f(x_0, x_1, \ldots, x_{k-1}) : E_2^k \to E_2$ назовем число наборов, на которых она принимает значение 1.

Лемма 3. *Полином Жегалкина булевой функции* $f(x_0, x_1, \ldots, x_{k-1}) : E_2^k \to E_2$ *веса 1 имеет степень* k.

Доказательство. Поскольку вес функции f равен 1, то она принимает значение 1 на одном наборе. Обозначим этот набор через $(\alpha_0, \alpha_1, \ldots, \alpha_{k-1})$. Тогда СДНФ функции f имеет вид

$$f(x_0, x_1, \ldots, x_{k-1}) = x_0^{\alpha_0} \cdot x_1^{\alpha_1} \cdot \ldots \cdot x_{k-1}^{\alpha_{k-1}}.$$

Каждый множитель есть либо x_i, если $\alpha_i = 1$, либо $x_i \oplus 1$, если $\alpha_i = 0$. Обозначим через $(\beta_0, \beta_1, \ldots, \beta_{k-1})$ такой набор, что $\beta_i = \overline{\alpha_i},\ i \in \{0, \ldots, k-1\}$. Тогда формулу для f можно переписать как

$$f(x_0, x_1, \ldots, x_{k-1}) = (x_0 \oplus \beta_0) \cdot (x_1 \oplus \beta_1) \cdot \ldots \cdot (x_{k-1} \oplus \beta_{k-1}).$$

Многочлен такого вида содержит слагаемое $x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1}$. Данное утверждение можно доказать по индукции.

База индукции: Если $k = 1$, то многочлен имеет вид либо x, либо $x \oplus 1$ и утверждение верно.

Шаг индукции: Пусть утверждение доказано для $k-1$, тогда произведение первых $k-1$ скобок можно записать как $x_0 \cdot x_1 \cdot \ldots \cdot x_{k-2} \oplus g(x_0, x_1, \ldots, x_{k-2})$, где $\deg(g(x_0, x_1, \ldots, x_{k-2})) < k-1$. Произведение всех k скобок можно записать как

$$(x_0 \cdot x_1 \cdot \ldots \cdot x_{k-2} \oplus g(x_0, x_1, \ldots, x_{k-2})) \cdot (x_{k-1} \oplus \beta_{k-1}),$$

где $\deg(g(x_0, x_1, \ldots, x_{k-2})) < k-1, \beta_{k-1} \in \{0, 1\}$. В результате раскрытия скобок получим выражение

$$x_0 \cdot x_1 \cdot \ldots \cdot x_{k-2} \cdot x_{k-1} \oplus g(x_0, x_1, \ldots, x_{k-2}) \cdot x_{k-1} \oplus$$
$$\oplus \beta_{k-1} \cdot x_0 \cdot x_1 \cdot \ldots \cdot x_{k-2} \oplus \beta_{k-1} \cdot g(x_0, x_1, \ldots, x_{k-2}).$$

При этом степень второго, третьего и четвертого слагаемых меньше чем k, а следовательно полученное выражение содержит слагаемое $x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1}$, т. е. полином Жегалкина функции f имеет степень k. □

Следствие 2. *Полином Жегалкина булевой функции* $f(x_0, x_1, \ldots, x_{k-1}) : E_2^k \to E_2$ *веса* $2^k - 1$ *имеет степень* k.

Доказательство. Поскольку вес функции f равен $2^k - 1$, то функция f принимает значение 0 только на одном наборе, а на остальных наборах — значение 1. Тогда функция $\overline{f}$ принимает значение 1 только на одном наборе, а на остальных наборах — значение 0. Обозначим ее через g. Согласно лемме 3 полином Жегалкина функции g содержит слагаемое $x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1}$, т. е.

$$g(x_0, x_1, \ldots, x_{k-1}) = x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1} \oplus h(x_0, x_1, \ldots, x_{k-1}),$$

где $deg(h) < k$. Поскольку $g(x_0, x_1, \ldots, x_{k-1}) = \overline{f}(x_0, x_1, \ldots, x_{k-1})$, то верно равенство

$$\begin{aligned} f(x_0, x_1, \ldots, x_{k-1}) &= g(x_0, x_1, \ldots, x_{k-1}) \oplus 1 = \\ &= x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1} \oplus h(x_0, x_1, \ldots, x_{k-1}) \oplus 1, \end{aligned}$$

где $deg(h) < k$. □

Теорема 1. *Пусть задано отображение* p *на множестве* $E_n = \{0, \ldots, n-1\}$, *где* $n = 2^k$, *соответствующее полиному Галуа* $c \cdot x^{n-1} + a$, *где* $c, a \in E_n, c \neq 0$. *Степенная сложность реализации отображения* p *посредством любого неизбыточного кодирования* F *равна* k, *т. е.* $L_{deg}^{irr}(p) = k$.

Доказательство. Заметим, что согласно операции умножения в поле Галуа, $0^{n-1} = 0$. Поскольку ненулевые элементы поля Галуа образуют относительно операции умножения циклическую группу порядка $n-1$, любой ненулевой элемент поля Галуа в степени $n-1$ есть 1. Следовательно, многочлен x^{n-1} соответствует отображению

$$p_{x^{n-1}} = \begin{pmatrix} 0 & 1 & 2 & \ldots & n-1 \\ 0 & 1 & 1 & \ldots & 1 \end{pmatrix}.$$

Многочлен $c \cdot x^{n-1} + a$ соответствует отображению

$$p_{c \cdot x^{n-1} + a} = \begin{pmatrix} 0 & 1 & 2 & \ldots & n-1 \\ a & b & b & \ldots & b \end{pmatrix},$$

где $b = c + a$. Заметим, что поскольку $c \neq 0$, то $b \neq a$.

Пусть задано неизбыточное кодирование $F : E_n \to E_2^k$. Введем обозначения: $F(0) = (\gamma_0, \gamma_1, \ldots, \gamma_{k-1})$, $F(a) = (\alpha_0, \alpha_1, \ldots, \alpha_{k-1})$, $F(b) = (\beta_0, \beta_1, \ldots, \beta_{k-1})$. По отображению $p_{c \cdot x^{n-1} + a}$ и кодированию F построим таблицу

x_0	x_1	$\ldots$	x_{k-1}	y_0	y_1	$\ldots$	y_{k-1}
0	0	$\ldots$	0	β_0	β_1	$\ldots$	β_{k-1}
0	0	$\ldots$	1	β_0	β_1	$\ldots$	β_{k-1}
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
γ_0	γ_1	$\ldots$	γ_{k-1}	α_0	α_1	$\ldots$	α_{k-1}
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
1	1	$\ldots$	1	β_0	β_1	$\ldots$	β_{k-1}

Множество индексов $\{0, 1, \ldots, k-1\}$ можно разделить на два подмножества. Первое множество M_1 содержит такие i, что $\alpha_i = \beta_i$. Второе множество M_2 содержит такие i, что $\alpha_i \neq \beta_i$. Множество M_2 не пусто в силу определения кодирования и неравенства $b \neq a$.

Для $i \in M_1$ верно, что $y_i = \alpha_i$, т. е. функция константа.

Если $i \in M_2$, то функция $y_i = f_i(x_0, \ldots, x_{k-1})$ обладает тем свойством, что на одном наборе она принимает значение α_i, на остальных β_i, причем $\alpha_i \neq \beta_i$. Согласно лемме 3 и следствию 2 полином Жегалкина функции $f_i(x_0, \ldots, x_{k-1})$ содержит слагаемое $x_0 \cdot x_1 \cdot \ldots \cdot x_{k-1}$. А значит $L_{deg}(p_{c \cdot x^{n-1}+a}, F) = k$. В силу произвольности выбора кодирования F верно, что $L^{irr}_{deg}(p_{c \cdot x^{n-1}+a}) = k$. □

Следствие 3. *Пусть задано отображение p на множестве $E_n = \{0, \ldots, n-1\}$, где $n = 2^k$, соответствующее полиному Галуа $c \cdot x^{n-1} + a$, $c \neq 0$. Тогда для любого взаимно-однозначного отображения $s : E_n \to E_n$ степенная сложность реализации отображения $s^{-1} \cdot p \cdot s$ равна k.*

Доказательство. Утверждение следует из теоремы 1 и следствия 1. □

4. Заключение

В работе изучался вопрос степенной сложности реализации отображения посредством неизбыточных кодирований. В ходе исследования были получены следующие основные результаты:

- Доказана лемма об эквивалентности множеств булевых операторов для сопряженных отображений. Показано, что если два отображения изоморфны с точностью до перестановки элементов (являются сопряженными), то порождаемые ими классы булевых операторов полностью совпадают, а их минимальные степенные сложности равны ($L^{irr}_{\deg}(p) = L^{irr}_{\deg}(s^{-1} \cdot p \cdot s)$). Это свойство позволяет существенно сузить пространство поиска при классификации отображений по их сложности.

- На основе полученных свойств веса функций найдена точная нижняя оценка сложности для специального класса мономов над конечными полями Галуа $\mathbb{F}_n$ (где $n = 2^k$). Доказано, что для любого отображения, представимого в виде полинома $c \cdot x^{n-1} + a$ ($c \neq 0$), степенная сложность реализации равна k при *любом* выборе неизбыточного двоичного кодирования.

Предметом дальнейшего исследования будет задача нахождения всех отображений, имеющих максимальную сложность реализации, а также поиск критерия данного свойства.

Заметим, что для задания переходной системы с одним бинарным входом достаточно указать пару отображений: одно — соответствующее входному символу 0, другое — соответствующее входному символу 1. Сложность реализации переходной системы будет не ниже, чем сложность каждого из отображений. А потому представляется возможным доказать существование «максимально сложных» переходных систем с использованием построенных отображений.

Также планируется продолжить исследование в области линейной реализации отображений. Ранее в работах [6] и [7] было показано, что при использовании кода длины n, любое отображение на n-элементном множестве является линейно реализуемым. Следовательно, и отображения, построенные в данной работе и имеющие максимальную степенную сложность при неизбыточном кодировании, также являются линейно реализуемыми при использовании кода длины n.

Рассмотрим отображение $p_{x^{n-1}}$ для $n = 8$

$$p_{x^7} = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Можно заметить, что при использовании кодирования F, задаваемого таблицей:

q	0	1	2	3	4	5	6	7
$F(q)$	1111	0000	0010	0100	0110	1000	1010	1100

отображение p_{x^7} является линейно реализуемым, так как соответствующий оператор имеет вид:

x_0	x_1	x_2	x_3	y_0	y_1	y_2	y_3
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0
1	0	1	0	0	0	0	0
1	1	1	1	1	1	1	1

и может быть доопределен как:

$$\begin{aligned} y_0 &= x_3, \\ y_1 &= x_3, \\ y_2 &= x_3, \\ y_3 &= x_3. \end{aligned}$$

Соответственно, возникает задача найти минимальное удлинение кода, при котором отображение допускает линейную реализуемость.

В заключение автор выражает благодарность Алёшину Станиславу Владимировичу и Носову Михаилу Васильевичу за многочисленные обсуждения и советы, которые позволили получить результаты, изложенные в данной работе.

Список литературы

[1] Родин С.Б., “Переходные системы с максимальной вариантностью относительно кодирования состояний”, *Интеллектуальные системы*, **4**:3-4 (1999), 335–352.

[2] Яблонский С.В., *Введение в дискретную математику*, Наука, Москва, 1979, 272 с.

[3] Родин С.Б., “Линейно реализуемые автоматы”, *Дискретная математика*, **29**:1 (2017), 59–79.

[4] Родин С.Б., “О связи линейно реализуемых автоматов и автоматов с максимальной вариативностью относительно кодирования состо-

яний", *Интеллектуальные системы. Теория и приложения*, **20**:2 (2016), 337–347.

[5] Родин С.Б., "О свойстве аддитивного сдвига для линейной реализуемости автоматов", *Интеллектуальные системы. Теория и приложения*, **28**:2 (2024), 52–66.

[6] Родин С.Б., "О свойствах кодирований состояний автомата", *Интеллектуальные системы. Теория и приложения*, **21**:1 (2017), 97–111.

[7] Родин С.Б., "О свойстве линейной реализуемости отображений", *Интеллектуальные системы. Теория и приложения*, **28**:1 (2024), 107–119.

[8] Р. Лидл, Г. Нидеррайтер, *Конечные поля*. Т. 1, Мир, Москва, 1988, 430 с.

Статья поступила 28 апреля 2026 г.

Mappings with maximum implementation complexity

S. B. Rodin

The paper continues the investigation into the complexity of automata implementation via state encoding. Previous research was focused on the identification of "simple implementations" of automata and mappings on a finite set. This paper is devoted to mappings with "complex implementations", i.e., mappings for which any irredundant encoding yields a Boolean operator of maximum complexity in terms of the Zhegalkin polynomial degree. It is proved that for any set cardinality $n = 2^k$ such mappings exist. These mappings can be used in future research to prove that automata admitting only "complex implementations" also exist.

Keywords: Automata theory, semiautomata, transition systems, assignment, state encoding, complexity, boolean operator.

References

[1] Rodin S.B., "The most variable semiautomata with respect to the states encoding", *Intelligent systems*, **4**:3-4 (1999), 335–352 (in Russian).

[2] Yablonskij S.V., *Introduction to the discrete math*, Nauka, Moscow, 1979 (in Russian), 272 pp.

[3] Sergey B. Rodin., "Linearly realizable automata", *Discrete Mathematics and Applications*, **27**:6 (2017), 387–402. DOI: 10.1515/dma-2017-0039.

[4] Rodin S.B., "On relation between the linearly realizable automata and the most variable automata with respect to the states encoding", *Intelligent systems. Theory and Applications*, **20**:2 (2016), 337–347 (in Russian).

[5] Rodin S.B., "On the additive shift property for the linear realizability of automata", *Intelligent systems. Theory and Applications*, **28**:2 (2024), 52–66 (in Russian).

[6] Rodin S.B., "On automata states encoding properties", *Intelligent systems. Theory and Applications*, **21**:1 (2017), 97–111 (in Russian).

[7] Rodin S.B., "On the linear realizability property of the mappings", *Intelligent systems. Theory and Applications*, **28**:1 (2024), 107–119 (in Russian).

[8] R. Lidl, H. Niederreiter, *Finite fields*. V. 1, Mir, Moscow, 1988 (in Russian), 430 pp.

Received on April 28, 2026

Часть 3
Математические модели

Почти одноместные базисы автоматов с операцией суперпозиции.

Д. Н. Бабин*

Для систем автоматных функций, образующих замкнутый класс важно определить, базисом от какого минимального числа переменных они могут быть получены операциями суперпозиции. Особый интерес представляют классы, в которых «собственно» автоматы из базиса имеют одну входную переменную, это ограничение не распространяется на булевы функции из базиса, которые являются автоматами с одним состоянием. Примером такого класса является класс всех автоматов. Автор изучает подклассы автоматов, у которых функции переходов и выходов принадлежат некоторому классу Поста.

Ключевые слова: арность, автомат, булевая функция, суперпозиция, замкнутый класс.

Понятие суперпозиции возникает сразу за понятием функции, как в математическом анализе, так и в алгебре. Относительно суперпозиции решаются главные задачи: полнота и выразимость. При этом под выразимостью понимается возможность получения функций одного множества через функции другого множества, а под полнотой — выразимость всех функций через заданные. Вычисление автоматной функции $\mathbf{f}$ задаётся своими каноническими уравнениями

$$\begin{cases} q(1) = q_0, \\ q(t+1) = \varphi(q(t), a(t)) \\ b(t) = \psi(q(t), a(t)) \end{cases} \tag{1}$$

Здесь q — это состояние a-функции, q_0 её начальное состояние, буквы $a(t)$ и $b(t)$ называются входной и выходной буквами, $t = 1, 2, ...$, а сверхслова $a(1)a(2)...$ и $b(1)b(2)...$ - входными и выходными сверхсловами, соответственно. Функции φ и ψ называются функцией переходов и выходной функцией, соответственно, а шестерка

$$(A, Q, B, \varphi, \psi, q_0) \tag{2}$$

* *Бабин Дмитрий Николаевич* — д.ф.-м.н., проф. каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: d.n.babin@mail.ru.

Babin Dmitry Nikolaevich — Doctor of Physical and Mathematical Sciences, Prof., Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intellectual Systems.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

называется инициальным автоматом, порождающим a-функцию $\mathbf{f}$. При этом предполагается, что шестёрка (2) минимальна по числу состояний среди всех подобных шестёрок для автоматной функции $\mathbf{f}$ (1). Это так называемый "приведенный" автомат, в котором все состояния достижимы из начального и нет эквивалентных состояний. При этом будем предполагать, что $A \subseteq E_2^n, Q \subseteq E_2^s, B \subseteq E_2$. Это так называемый структурный автомат с n входами.

Автоматная функция работает в дискретном времени, и при каждом такте работы меняет своё активное состояние под действием входной буквы. На выходе в этот такт возникает выходная буква, определяемая выходной функцией. Таким образом, автомат реализует отображение слов входного алфавита в слова выходного алфавита.

Входной сложностью автомата (2) с $A \subseteq E_2^n, Q \subseteq E_2^s, B \subseteq E_2$ называется число его входов n. Входной сложностью множества автоматов называется максимальная входная сложность (или её точная верхняя грань) автоматов этого множества, арностью замкнутого класса автоматов называется минимальная входная сложность порождающей системы этого класса.

Для непрерывных функций задача минимальной арности порождающей системы была решена А.Н. Колмогоровым в статье "О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения."[1]. Для автоматных функций подобная теорема была получена автором [2]. Оказалось, что система автоматов с одним входом и универсальная булева функция - это полная система в классе автоматов с операцией суперпозиции.

Назовём автоматную функцию *тривиальной*, если она может быть реализована автоматом с одним состоянием. Тривиальная автоматная функция — это, фактически, булева функция. Назовём замкнутый класс автоматов классом *арности почти один*, если в нём найдётся базис, все нетривиальные автоматы которого имеют один вход.

Представляют интерес классы автоматов *арности почти один*, так как это минимальные по арности классы автоматов, имеющие в своём замыкании автоматы со многими входами. А также интересны классы автоматов не являющиеся таковыми.

Пусть в автомате (2) с каноническими уравнениями (1) при всех $a \in A$ и всех $q_1 \neq q_2$ выполнено $\varphi(q_1, a) \neq \varphi(q_2, a)$, то есть каждая входная буква a задаёт биекцию

$$\varphi_a : Q \to Q$$

множества состояний. Здесь $\varphi_a(q) = \varphi(q, a)$. Такие автоматы называются *групповыми*. Множество перестановок, порождённое образующими

из $\{\varphi_a | a \in A\}$ называется *полугруппой автомата* (2), для группового автомата полугруппа является группой.

Не трудно убедиться, что множество групповых автоматов замкнуто относительно суперпозиции, а также, что приведенный автомат группового автомата тоже групповой. Автомат с циклической группой, назовём циклическим групповым автоматом. Известно, что замыкание множества циклических групповых автоматов — это собственный подкласс класса групповых автоматов — множество групповых автоматов с разрешимыми группами. Булевы функции можно считать циклическими групповыми автоматами.

Рассмотрим маленький

Пример.

Пусть G конечный коммутативный моноид с нейтральным элементом 0 и операцией $\oplus$, и

$$0 \in A_1 \subseteq G, 0 \in A_2 \subseteq G,$$

системы образующих, содержащие 0. Пусть автоматы таковы:

$$(A_1, G, G, \oplus, \psi_0, 0), \qquad (A_2, G, G, \oplus, \psi_0, 0), \tag{3}$$

где $\psi_0(q, a) \equiv q$. Задаваемые этими автоматами автоматные функции обозначим через $\mathbf{f_1}$ и $\mathbf{f_2}$, соответственно. Автоматная функция $\mathbf{f_1} \oplus \mathbf{f_2}$ (рис.1) порождается автоматом

$$(A_1 \times A_2, G \times G, G, (\oplus, \oplus), \psi_0 \oplus \psi_0, (0, 0)), \tag{4}$$

Состояния (q_1, q_2) и (q_3, q_4), при $q_1 \oplus q_2 = q_3 \oplus q_4 = q$ эквивалентны потому что, по входной букве (a_1, a_2) они перейдут в эквивалентные состояния (q'_1, q'_2) и (q'_3, q'_4), где

$$(q'_1 \oplus q'_2) = (q_1 \oplus a_1) \oplus (q_2 \oplus a_2) = (q_1 \oplus q_2) \oplus (a_1 \oplus a_2) = q \oplus (a_1 \oplus a_2).$$

$$(q'_3 \oplus q'_4) = (q_3 \oplus a_1) \oplus (q_4 \oplus a_2) = (q_3 \oplus q_4) \oplus (a_1 \oplus a_2) = q \oplus (a_1 \oplus a_2).$$

Приведенный автомат для $\mathbf{f_1} \oplus \mathbf{f_2}$ (рис. 1) — это

$$(A_1 \times A_2, G, G, \varphi_0, \psi_0, 0), \tag{5}$$

где $\varphi_0(q, a_1, a_2) = q \oplus (a_1 \oplus a_2)$

Автомат (5) уже имеет множество разных преобразований, задаваемых входными буквами из множества

$A_1 \oplus A_2 = \{a_1 \oplus a_2 \mid a_1 \in A_1, a_2 \in A_2\}$.

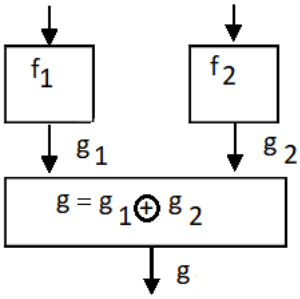


Рис. 1

Применим эту схему к автоматам с циклическими группами и одним входом, пусть $GG = G \setminus \{0\}, |GG| = t$ множество ненулевых элементов циклической группы G и автоматная функция $\mathbf{f_i}$ задаётся автоматом

$$(\{0,1\}, G, G, \varphi_i, \psi_0, 0), i = 1, \ldots, t,$$

где $\varphi_i(g,0) = g, \varphi_i(g,1) = g \oplus a_i$.

Рассмотрим автомат $\mathbf{F} = (\mathbf{f_1} \oplus \ldots \oplus \mathbf{f_t})(\mathbf{\Theta(a)})$ (рис.2).

Здесь $\Theta : {E_2}^p \to {E_2}^t$, имеет в качестве входной буквы $a \in G$ в некоторой двоичной кодировке.

$$\Theta(a_i) = 0 \ldots 1 \ldots 0,$$

где 1 стоит на i-том месте, $i = 1, \ldots, t$. $\Theta(0) = 0^t$. Здесь p это целая часть от $\log_2(t+1) + 1$. Для неиспользуемого кода x положим $\Theta(x) = 0^t$. В дальнейшем мы будем аналогично использовать не бинарные входы и выходы автоматов.

Схема (рис.2) из состояния $g = g_1 \oplus g_2 \oplus \ldots \oplus g_t$ под действием входной буквы a_i перейдёт в состояние $g_1 \oplus \ldots \oplus g_i \oplus a_i \oplus \ldots \oplus g_t = g \oplus a_i$. Таким образом, схема реализует автомат

$$(G, G, G, \oplus, \psi_0, 0),$$

из которого и булевых функций можно получить любой автомат с циклической группой G. Известно [3], что замыкание множества автоматов с циклическими группами даёт класс автоматов с разрешимыми группами.

Кроме того, сам класс групповых автоматов не порождается своими одноместными автоматами и булевыми функциями [4]. Получилась

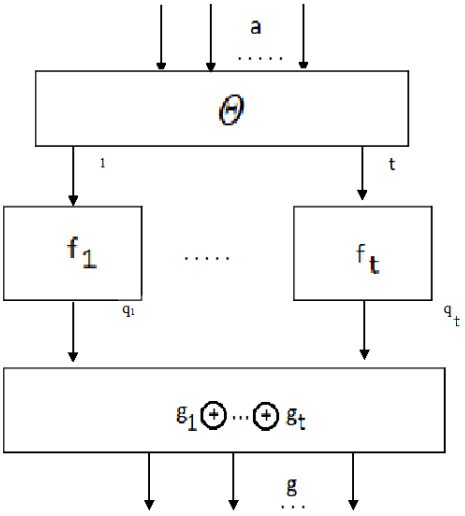


Рис. 2

Теорема 1.

1. Класс групповых автоматов с разрешимыми группами имеет арность почти 1.

2. Класс групповых автоматов не имеет арность почти 1.

В рассмотренном примере требование совпадения операции $\oplus$ с функцией переходов автомата является избыточным, достаточно потребовать, чтобы в автоматах (3) было выполнено

$$\varphi(q, a) = \Phi(q) \oplus a, \tag{6}$$

где $\Phi(q_1 \oplus q_2) = \Phi(q_1) \oplus \Phi(q_2)$

В самом деле, тогда

$$\varphi(q_1, a_1) \oplus \varphi(q_2, a_2) = (\Phi(q_1) \oplus a_1) \oplus (\Phi(q_2) \oplus a_2) = \Phi(q_1 \oplus q_2) \oplus (a_1 \oplus a_2),$$

Под это условие подпадает класс линейных автоматов. Автомат (1) называется линейным, если его функции переходов и выходов линейны. Нас интересует возможность выразимости через одноместные автоматы и булевые линейные функции линейного автомата, выдающего код своего состояния. Его канонические уравнения имеют вид

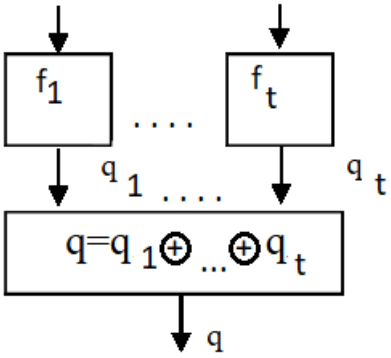


Рис. 3

$$\begin{cases} q(1) = 0, \\ q(t+1) = Aq(t) + Ba(t) \\ b(t) = q(t) \end{cases} \tag{7}$$

Здесь в качестве $\oplus$ используется $+$ сложение по модулю 2, $a \in {E_2}^t$. Произвольный линейный автомат получается из (7) суперпозицией с линейными булевыми функциями. Пусть в (7) матрица B это строка из столбцов $(B_1 B_2 \dots B_t)$, тогда автомат (7) получается схемой (рис.3) из одноместных автоматов (8) вида

$$(\{0,1\}, Q, Q, \varphi_i, \psi_0, 0), i = 1, \dots, t,$$

где $\varphi_i(q,x) = Aq + B_i x, \psi_0(q,x) = q, x \in \{0,1\}, i = 1, \dots t.$

$$\begin{cases} q(1) = 0, \\ q(t+1) = Aq(t) + B_i a(t) \\ b(t) = q(t) \end{cases} \tag{8}$$

Схема (рис.3) из состояния $q = q_1 + q_2 + \dots + q_t$ под действием входной буквы $x_1 \dots x_t$ перейдёт в состояние

$Aq_1 + B_1 x_1 + \dots + Aq_i + B_i x_i + \dots + Aq_t + B_t x_t = Aq + B(x_1 \dots x_t)^T$. Здесь $(x_1 \dots x_t)^T$ обозначает транспонированную строку или вектор-столбец.

Схема (рис.3) реализует автомат (7). Таким образом, получилась известная

Теорема 2. *Класс линейных автоматов имеет арность почти 1.* [5]

Заметим, что теорема 2 верна для линейных автоматов над любым конечным полем.

Если в качестве операции $\oplus$ взять дизъюнкцию $\vee$, то получаем класс автоматов с идемпотентным сложением $x \vee y = y \vee x, x \vee x = x$ и каноническими уравнениями

$$\begin{cases} q(1) = q_0, \\ q(t+1) = Aq(t) \vee Ba(t) \\ b(t) = Dq(t) \vee La(t) \end{cases} \tag{9}$$

Доказательство выразимости автомата (9) через одноместные и дизъюнкцию проводится аналогично линейному случаю. При этом знак $+$ заменяется на $\vee$ и при умножении строки $a_1 a_2 \dots a_m \in {E_2}^m$ на столбец $q_1 q_2 \dots q_m \in {E_2}^m$ используется формула

$$(a_1 \cdot q_1) \vee \dots \vee (a_m \cdot q_m)$$

Имеет место

Теорема 3.
Класс дизъюнктивных автоматов имеет арность почти 1.

Теорема обобщается для автоматов над E_k, если вместо функции $\vee$ использовать максимум $\max(x, y)$.

Если в качестве операции $\oplus$ взять конъюнкцию $\&$

$$x\&y = y\&x, x\&x = x,$$

то получаем замкнутый класс автоматов с каноническими уравнениями

$$\begin{cases} q(1) = q_0, \\ q(t+1) = Aq(t)\&Ba(t) \\ b(t) = Dq(t)\&La(t) \end{cases} \tag{10}$$

Доказательство выразимости автомата (10) через одноместные и конъюнкцию проводится аналогично линейному случаю. При этом знак $+$

заменяется на & и при умножении строки $a_1a_2\dots a_m \in {E_2}^m$ на столбец $q_1q_2\dots q_m \in {E_2}^m$ используется формула

$$({q_1}^{a_1}\&\dots\&{q_m}^{a_m})$$

где $q^0 = 1, q^1 = q$.

Класс автоматов с каноническими уравнениями (10) назовём конъюнктивными автоматами. Не трудно убедиться, что множество конъюнктивных автоматов замкнуто относительно суперпозиции, а также, что приведенный автомат конъюнктивного автомата тоже конъюнктивный. Аналогично получаем теорему

Теорема 4.

Класс конъюнктивных автоматов имеет арность почти 1.

Теорема обобщается для автоматов над E_k, если вместо функции & использовать минимум $\min(x, y)$.

Список литературы

[1] А. Н. Колмогоров., "О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения", *Докл. АН СССР*, **114**:5 (1957), 953–956.

[2] Бабин Д.Н., "О полноте двухместных о.д.-функций относительно суперпозиции", *Дискретная математика*, **1**:4 (1989), 86–91.

[3] Арбиб М., *Алгебраическая теория автоматов языков и полугрупп*, Статистика, Москва, 1975.

[4] Бабин Д.Н., "Вербальные подавтоматы и задача полноты", *Вестник Московского университета*, 1985, № 3, Математика и механика, 82-85.

[5] Кудрявцев В.Б., Алешин С.В., Подколзин А.С., *Введение в теорию автоматов.*, Наука, Москва, 1985.

Статья поступила 30 марта 2026 г.

Almost unary bases of automata under superposition.

D. N. Babin

For systems of automaton functions that form a closed class, it is important to determine the basis from the minimal number of variables from which they can be obtained by superposition operations. Of particular interest are classes in which the 'proper' automata from the basis have a single input variable; this restriction does not apply to Boolean functions from the basis, which are automata with a single state. An example of such a class is the class of all automata. The author studies subclasses of automata whose transition and output functions belong to a certain Post class.

Keywords: arity, automaton, Boolean function, superposition, closed class.

References

[1] A. N. Kolmogorov, "On the representation of continuous functions of several variables in the form of superpositions of continuous functions of one variable and addition (in Russian)", *Proc. USSR Acad. Sci.*, **114**:5 (1957), 953–956.

[2] Babin D.N., "On the completeness of binary o.d. functions with respect to superposition (in Russian)", *Discrete Mathematics*, **1**:4 (1989), 86–91.

[3] Arbib M., *Algebraic theory of automata, languages, and semigroups (in Russian)*, Statistika, Moscow, 1975.

[4] Babin D.N., "Verbal subautomata and the completeness problem (in Russian)", *Bulletin of Moscow University*, 1985, №3, Mathematics and Mechanics, 82-85.

[5] Kudryavtsev V.B., Aleshin S.V., Podkolzin A.S., *Introduction to Automata Theory (in Russian)*, Nauka, Moscow, 1985.

Received on March 30, 2026

О слабости одномерных идемпотентных клеточных автоматов с локаторами

Э. Э. Гасанов*

Рассматриваются одномерные клеточные автоматы с локаторами, у которых полугруппа, заданная на алфавите вещания, является идемпотентным моноидом. Ранее было показано, что такие сложные задачи для клеточных автоматов как задача синхронизации стрелков или задача управления движением точки на прямой в идемпотентных клеточных автоматах с локаторами решаются совсем просто и задачи фактически вырождаются. В данной работе показано, что для задач, где надо сравнивать расстояния, идемпотентные клеточные автоматы с локаторами не дают преимущества по порядку по сравнению с обычными клеточными автоматами. Тогда как использование неидемпотентных клеточных автоматов дает принципиальный выигрыш по времени.

Ключевые слова: идемпотентные клеточные автоматы с локаторами, сравнение и сложение целых чисел.

1. Введение

Понятие клеточного автомата было введено Дж. фон Нейманом [1–3] для описания процессов самовоспроизведения в биологии и технике. Эта модель развивалась и использовалась в работах А. Беркса [4], Э. Мура [5], В. Б. Кудрявцева, А. С. Подколзина, А. А. Болотова [6] и других исследователей [7, 8].

Клеточный автомат — это математический объект с дискретными пространством и временем. Пространство поделено на клетки, в каждой из которых находится элементарный автомат. Такты времени задаются натуральными числами. Состояние автомата каждой пространственной клетки определяется очень простыми правилами взаимодействия, которые предписывают изменения состояния автомата каждой клетки в следующем такте времени в ответ на текущее состояние автоматов соседних клеток.

Гасанов Эльяр Эльдарович — д.ф.-м.н., зав. каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: el_gasanov@mail.ru, ORCID: 0000-0002-0739-0967.

Gasanov Elyar Eldarovich — Doctor of Physical and Mathematical Sciences, Head of the Chair of Mathematical Theory of Intelligent Systems, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

В работе Э. Э. Гасанова [9] было введено понятие клеточного автомата с локаторами, которое отличается от понятия обычного клеточного автомата тем, что допускает передачу информации не только между соседними ячейками, но и на любое расстояние посредством передачи сигналов из алфавита вещания в эфир.

В данной работе рассматриваются одномерные клеточные автоматы с локаторами, у которых полугруппа, заданная на алфавите вещания, является идемпотентным моноидом. В работе [9] показано, что такие сложные задачи для клеточных автоматов как задача синхронизации стрелков [10] или задача управления движением точки на прямой [11] в идемпотентных клеточных автоматах с локаторами решаются совсем просто и задачи фактически вырождаются. В данной работе показано, что для задач, где надо сравнивать расстояния, идемпотентные клеточные автоматы с локаторами не дают преимущества по порядку по сравнению с обычными клеточными автоматами и время решения задачи линейно зависит от расстояния. Хотя, как показано в работе [12], использование неидемпотентных клеточных автоматов дает принципиальный выигрыш, и время решения пропорционально логарифму от расстояния.

В настоящей работе рассматривается следующая задача. На прямой заданы 2 числа, одно из которых положительное, а второе отрицательное. Надо понять какое из них меньшее по модулю. Эта задача перекликается с одномерной задачей поиска ближайшего соседа, рассмотренной в работах [12, 13]. В этих работах показано, что эту задачу с помощью клеточного автомата с локаторами можно решить за время, равное по порядку логарифму от расстояния до ближайшей точки, причем в работе [12] получена верхняя оценка, а в работе [13] — нижняя. При этом при получении верхней оценки по существу используется сложение по модулю 2 по одной из компонент алфавита вещания, т.е. полугруппа, заданная на алфавите вещания, не является идемпотентным моноидом.

В данной работе показано, что задача сравнения по модулю целых чисел не может быть решена быстрее чем за время, пропорциональное расстоянию до ближайшей точки, что равно по порядку времени решения этой задачи обычным клеточным автоматом. Полученная линейная нижняя оценка автоматически дает линейную нижнюю оценку и для одномерной задачи поиска ближайшего соседа, и для задачи сложения целых чисел.

Автор выражает благодарность Г.В.Калачеву за ценные замечания и советы.

2. Постановка задачи и формулировка результатов.

Как уже было сказано, в [9] было введено понятие клеточного автомата с локаторами. В работе Г. В. Калачева [14] были выявлены некоторые неточности приведенного в [9] определения. Наиболее точное определение клеточного автомата с локаторами дано в работах [15, 16].

В настоящей работе мы не будем приводить общее понятие клеточного автомата с локаторами, а определим только его сужение на одномерный случай, приведенное в работе [13].

В определении клеточного автомата с локаторами используется понятие рационального телесного угла, но в одномерном случае имеются только 3 рациональных телесных угла, которые мы обозначим $\nu_{-1} = \{x \in \mathbb{R} : x < 0\}$, $\nu_1 = \{x \in \mathbb{R} : x > 0\}$, $\nu_\Omega = \nu_{-1} \cup \nu_1$. Телесный угол ν_Ω называется *полным*.

Одномерным клеточным автоматом с локаторами (КАсЛ) называется восьмерка $\sigma = (\mathbb{Z}, Q, V, G, +, \mathcal{L}, \varphi, \psi)$, где $\mathbb{Z}$ — множество одномерных векторов с целыми координатами; Q — некоторое конечное множество, называемое *множеством состояний*; в множестве Q выделено одно состояние q_0, называемое *состоянием покоя*; $V = \{-1, 1\}$ — множество из двух векторов из $\mathbb{Z}$; $(G, +)$ — конечная коммутативная полугруппа с нейтральным элементом e и с коммутативной полугрупповой операцией $+$; $\mathcal{L} = \{\nu_{-1}, \nu_1\}$ — множество из описанных выше рациональных телесных углов в $\mathbb{R}$; φ — функция, зависящая от переменных $x_{-1}, x_0, x_1, z_{-1}, z_1, z_\Omega$; $\varphi : Q^3 \times G^2 \to Q$, $\varphi(\mathbf{q_0}, \mathbf{e}) = q_0$; $\mathbf{q_0} = (q_0, q_0, q_0) \in Q^3$, $\mathbf{e} = (e, e) \in G^2$; ψ — функция, зависящая от переменных $x_{-1}, x_0, x_1, z_{-1}, z_1, z_\Omega$; $\psi : Q^3 \times G^2 \to G$; $\psi(\mathbf{q_0}, \mu) = e$, $\mu \in G^2$. Элементы множества $\mathbb{Z}$ называются *ячейками* клеточного автомата σ, элементы множества Q — *состояниями ячейки* клеточного автомата σ. Множество V называется *шаблоном соседства* клеточного автомата σ и состоит из двух соседей — ближайшего соседа слева и ближайшего соседа справа. Множество G называется *алфавитом вещания*, а его элементы называются *сигналами вещания*. Множество $\mathcal{L}$ называется *шаблоном локаторов* клеточного автомата σ и состоит из двух локаторов — смотрящего влево и смотрящего вправо. Функция φ называется *локальной функцией переходов* автомата σ, функция ψ — *функцией вещания* автомата σ. Переменные x_{-1}, x_0, x_1 принимают значения из Q, переменные z_{-1}, z_1, z_Ω — из G. Состояние q_0 интерпретируется как *состояние покоя*, а условие $\varphi(\mathbf{q_0}, \mathbf{e}) = q_0$ — как *условие сохранения состояния покоя*. Ячейки, находящиеся в состоянии, отличном от q_0, будем называть *активными*. Условие $\psi(\mathbf{q_0}, \mu) = e$ означает, что ячейка в состоянии покоя, не имеющая

активных соседей, посылает в эфир нейтральный элемент, что можно интерпретировать как то, что она не посылает сигналы в эфир.

Если $a \in \mathbb{Z}$, ν — телесный угол из $\mathcal{L}$, то через $\nu(a)$ обозначим телесный угол, полученный параллельным переносом угла ν в точку a, т.е. $\nu_{-1}(a) = \{x \in \mathbb{R} : x < a\}$, $\nu_1(a) = \{x \in \mathbb{R} : x > a\}$, $\nu_\Omega(a) = \nu_{-1}(a) \cup \nu_1(a)$.

Если $a \in \mathbb{Z}$ — ячейка клеточного автомата σ, то множество $V(a) = \{a-1, a, a+1\}$ называется *окрестностью ячейки* a, а множество $\mathcal{L}(a) = \bigcup\limits_{\nu \in \mathcal{L}} \nu(a)$ — *множеством локаторов ячейки* a.

Состоянием клеточного автомата с локаторами σ назовем пару (g, f), где g — произвольная функция, определенная на множестве $\mathbb{Z}$, принимающая значения из G и называемая *состоянием эфира*, f — произвольная функция, определенная на множестве $\mathbb{Z}$, принимающая значения из Q и называемая *распределением состояний клеточного автомата с локаторами* σ. Такую функцию можно интерпретировать как некую мозаику, возникающую в одномерном пространстве в результате приписывания каждой точке с целочисленными координатами некоторого сигнала из множества G и некоторого состояния из множества Q. Множество всевозможных состояний клеточного автомата с локаторами обозначим Σ.

Если $a \in \mathbb{Z}$, (g, f) — состояние клеточного автомата с локаторами σ, то значение $g(a)$ называем *сигналом ячейки* a, *определяемым состоянием* (g, f), а значение $f(a)$ — *состоянием ячейки* a, *определяемым состоянием* (g, f). Значения

$$\phi^g_{-1}(a) = \sum_{b \in \nu_{-1}(a) \cap \mathbb{Z}} g(b), \ \phi^g_1(a) = \sum_{b \in \nu_1(a) \cap \mathbb{Z}} g(b) \tag{1}$$

называем *значениями локаторов* ν_{-1} *и* ν_1 *ячейки* a, *определяемыми состоянием* (g, f). Здесь суммирование сигналов осуществляется с помощью определяющей операции $+$ полугруппы G. Отметим, что в формулах (1) используются формально бесконечные суммы, и, чтобы они были определены, мы либо будем считать, что только конечное число слагаемых в суммах отлично от нейтрального элемента, либо предположим, что полугруппа $(G, +)$ является *идемпотентным моноидом*, т.е. для любого $h \in G$ выполнено $h + h = h$.

Отметим также, что если в шаблоне локаторов $\mathcal{L}$ есть локатор, соответствующий полному телесному углу, то его значение равно

$$\phi^g_\Omega(a) = \sum_{b \in \nu_\Omega(a) \cap \mathbb{Z}} g(b) = \phi^g_{-1}(a) + \phi^g_1(a),$$

т.е. его значение полностью определяется значениями локаторов ν_{-1} и ν_1, поэтому полный локатор является избыточным в одномерном случае.

На множестве Σ определим *глобальную функцию переходов* Φ_σ клеточного автомата с локаторами σ, полагая $\Phi_\sigma(g,f) = (g',f')$, где $(g,f),(g',f') \in \Sigma$ и для любой ячейки $a \in \mathbb{Z}$ выполняются тождества

$$g'(a) = \psi(f(a-1), f(a), f(a+1), \phi^g_{-1}(a), \phi^g_1(a)), \tag{2}$$

$$f'(a) = \varphi(f(a-1), f(a), f(a+1), \phi^g_{-1}(a), \phi^g_1(a)). \tag{3}$$

Содержательная интерпретация отображения Φ_σ такова, что сигнал каждой ячейки и состояние каждой ячейки "после перехода" определяются по состоянию окрестности ячейки и по значениям локаторов "до перехода" с помощью законов ψ и φ одинаково для всех ячеек.

Поведениями клеточного автомата с локаторами σ называем такие последовательности (g_0,f_0), (g_1,f_1), $(g_2,f_2),\ldots$ его состояний, для которых выполняется $(g_{i+1},f_{i+1}) = \Phi_\sigma(g_i,f_i)$ для всех $i = 0,1,2,\ldots$, причем (g_i,f_i) называется *состоянием клеточного автомата с локаторами* σ *в момент* i, а (g_0,f_0) — *начальным состоянием клеточного автомата с локаторами* σ.

Для одномерных КА с локаторами состояние удобно записывать в виде слова, бесконечного в оба конца. При этом, чтобы сохранять привязку, удобно в этом слове отметить фиксированную позицию (не обязательно начало координат). Пусть $\vec{\alpha} = (\alpha_1,\alpha_2,\ldots)$ — обозначение для сверхслова α в обычной записи слева направо, когда хочется подчеркнуть направление записи, $\overleftarrow{\alpha} = (\ldots,\alpha_2,\alpha_1)$ — симметричное обозначение для сверхслова α, записанное справа налево. Введём обозначение $\left(\overleftarrow{\alpha} \,\middle|_i\, \vec{\beta}\right)$, где $\alpha,\beta \in (G\times Q)^\infty$, $i \in \mathbb{Z}$.

$$\left(\overleftarrow{\alpha} \,\middle|_i\, \vec{\beta}\right) := (g,f) \in \Sigma, \tag{4}$$

где $(g(i+k-1), f(i+k-1)) = \beta_k$,$(g(i-k), f(i-k)) = \alpha_k$, $k \in \mathbb{N}$.

Пусть $q \in Q$, $e \in G$ — нейтральный элемент полугруппы G, тогда обозначим $\dot{q} = (e,q)$.

Сформулируем задачу сравнения по модулю целых чисел $S(l,r)$, $l,r \in \mathbb{N}$. Пусть на $\mathbb{Z}$ задано начальное состояние $I_0(l,r) = (g_0,f_0)$ клеточного автомата, удовлетворяющее следующим условиям:

1) ячейке с координатой 0 присвоено состояние C, т.е. $f_0(0) = C$;

2) ячейке с координатой $-l$ присвоено состояние L, т.е. $f_0(-l) = L$;

3) ячейке с координатой r присвоено состояние R, т.е. $f_0(r) = R$;

4) остальным ячейкам присвоено состояние q_0, являющееся состоянием покоя;

5) все ячейки не посылают в эфир сигналы, т.е. для любой ячейки $a \in \mathbb{Z}$ выполнено $g_0(a) = e$.

В нотации (4)

$$I_0(l,r) = \left(\dot{q_0}^{\infty} \dot{L} \dot{q_0}^{l-1} \dot{C} \;\middle|_1\; \dot{q_0}^{r-1} \dot{R} \dot{q_0}^{\infty} \right). \tag{5}$$

Решением задачи сравнения по модулю целых чисел $S(l,r)$, соответствующей начальному состоянию $I_0(l,r)$, назовем состояние $I_F(l,r) = (g_F, f_F)$ автомата, удовлетворяющее следующим условиям:

1) если $l < r$, то ячейкам из отрезка $[-l, 0]$ присваиваются значения L_F, т.е. для любой ячейки $a \in [-l, 0]$ выполнено $f_F(a) = L_F$;

2) если $l \geq r$, то ячейкам из отрезка $[0, r]$ присваиваются значения R_F, т.е. для любой ячейки $a \in [0, r]$ выполнено $f_F(a) = R_F$;

3) остальные ячейки находятся в состоянии покоя q_0;

4) все ячейки не посылают в эфир сигналы, т.е. для любой ячейки $a \in \mathbb{Z}$ выполнено $g_F(a) = e$.

В нотации (4) если $l < r$, то

$$I_F(l,r) = \left(\dot{q_0}^{\infty} \dot{L_F}^{l+1} \;\middle|_1\; \dot{q_0}^{\infty} \right), \tag{6}$$

если $l \geq r$, то

$$I_F(l,r) = \left(\dot{q_0}^{\infty} \;\middle|_0\; \dot{R_F}^{r+1} \dot{q_0}^{\infty} \right). \tag{7}$$

Скажем, что клеточный автомат с локаторами σ *решает задачу сравнения по модулю целых* чисел, если для любых натуральных чисел l, r из начального состояния $I_0(l,r)$, соответствующего задаче $S(l,r)$ и задаваемого выражением (5), он переходит в финальное состояние $I_F(l,r)$, описываемое выражениями (6–7), после достижения финального состояния оно больше не меняется, и ни одно промежуточное состояние не содержит ячеек в состояниях L_F и R_F.

Пусть $l, r \in \mathbb{N}$. Обозначим через $T(\sigma, l, r)$ — время, за которое произвольный клеточный автомат с локаторами σ решает задачу сравнения по модулю целых чисел $S(l,r)$, (положим $T(\sigma,l,r) = \infty$, если автомат σ не решает задачу $S(l,r)$).

Назовем клеточный автомат с локаторами *идемпотентным*, если полугруппа его алфавита вещания является идемпотентным моноидом.

Следующая лемма является аналогом леммы о накачке, которая говорит, что можно сколь угодно накручивать периодическую часть без качественного изменения результата.

Лемма 1 (о накачке)**.** *Пусть* $\sigma = (\mathbb{Z}, Q, V, G, +, L, \varphi, \psi)$ *— одномерный идемпотентный КАсЛ. Для любых* $\alpha, \beta \in (G \times Q)^{\infty}$, $\gamma \in (G \times Q)^{+}$, $t \in \mathbb{N}$

существуют такие $\alpha', \beta' \in (G \times Q)^\infty$, $\gamma' \in (G \times Q)^{|\gamma|}$, *что для любого* $k \in \mathbb{N} \cup \{0\}$, $i \in \mathbb{Z}$ *выполнено*

$$\Phi_\sigma^t \left(\overleftarrow{\alpha} \gamma^t \;\middle|_i\; \gamma^k \gamma^t \vec{\beta} \right) = \left(\overleftarrow{\alpha'} \;\middle|_i\; \gamma'^k \vec{\beta'} \right).$$

Теорема 1. *Если* $l, r \in \mathbb{N}$, *то для любого одномерного идемпотентного клеточного автомата с локаторами* σ *выполнено* $T(\sigma, l, r) \geq \frac{\min(l,r)}{2}$.

Теорема 2. *Существует одномерный идемпотентный клеточный автомат с локаторами* σ, *такой, что он решает задачу сравнения по модулю целых чисел и для любых натуральных чисел* l, r *выполнено* $T(\sigma, l, r) = \lfloor \frac{\min(l,r)-1}{2} \rfloor + 2 = \lceil \frac{\min(l,r)}{2} \rceil + 1$.

Здесь $\lfloor \frac{a}{2} \rfloor$ — целая часть от деления a на 2, $\lceil \frac{a}{2} \rceil$ — наименьшее целое, не меньшее $a/2$.

3. Доказательство леммы о накачке

Сначала докажем утверждение леммы для случая $t = 1$, т.е. докажем равенство

$$\Phi_\sigma \left(\overleftarrow{\alpha} \gamma \;\middle|_i\; \gamma^k \gamma \vec{\beta} \right) = \left(\overleftarrow{\alpha'} \;\middle|_i\; \gamma'^k \vec{\beta'} \right). \tag{8}$$

Обозначим исходное состояние $(g, f) = \left(\overleftarrow{\alpha} \gamma \;\middle|_i\; \gamma^k \gamma \vec{\beta} \right)$.

Это состояние состоит из левого хвоста $\overleftarrow{\alpha} \gamma$, правого хвоста $\gamma \vec{\beta}$ и периодической части γ^k (если $k > 0$).

Обозначим через $T = |\gamma|$ — длину периода слова γ^k. Поскольку $\gamma \in (G \times Q)^+$, то $T > 0$.

Если $a = (r, q) \in G \times Q$, то обозначим $s(a) = q$, $w(a) = r$. Если $\alpha \in (G \times Q)^*$ или $\alpha \in (G \times Q)^\infty$, т.е. α — слово или сверхслово, то обозначим $w(\alpha) = \sum_{a \in \alpha} w(a)$. Обозначим $B = w(\gamma) = \sum_{p=1}^{T} w(\gamma_p)$. Сигнал B — это суммарный сигнал слова γ.

Рассмотрим случай, когда $k = 0$.

Пусть

$$\Phi_\sigma \left(\overleftarrow{\alpha} \gamma \;\middle|_i\; \gamma \vec{\beta} \right) = \left(\overleftarrow{\alpha'} \;\middle|_i\; \vec{\beta'} \right). \tag{9}$$

для некоторых α' и β', которые назовем соответственно левым и правым хвостами результата.

Сначала покажем, что для любого $k \in \mathbb{N}$

$$\Phi_\sigma \left(\overleftarrow{\alpha} \gamma \;\middle|_i\; \gamma^k \gamma \vec{\beta} \right) = \left(\overleftarrow{\alpha'} \;\middle|_i\; \gamma''_k \vec{\beta'} \right), \tag{10}$$

где $|\gamma''_k| = kT$, а α' и β' те же самые, что и в равенстве (9).

Обозначим $(g', f') = \left(\overleftarrow{\alpha}\gamma \;\middle|_i\; \gamma\vec{\beta}\right)$.

Рассмотрим левый хвост. Согласно формулам (2) и (3) значения функций перехода и вещания определяются состояниями соседей и значениями левого и правого локаторов.

Рассмотрим произвольную ячейку j из левого хвоста, т.е. $j < i$. По определению для любого $n < i$ выполняется $f(n) = f'(n)$. Для ячейки $i-1$ правым соседом является ячейка i, но $f(i) = f'(i) = s(\gamma_0)$. Таким образом состояния ячеек, по которым определяются значения функций переходов и вещания для левого хвоста полностью одинаковые.

Значение левого локатора ячейки j определяется значениями сигналов, передаваемых ячейками, находящимися слева от j, и равно

$$\phi^g_{-1}(j) = \sum_{p=-\infty}^{j-1} g(p) = \sum_{p=-\infty}^{j-1} g'(p) = \phi^{g'}_{-1}(j),$$

т.е. значения левого локатора одинаковые для g и g'.

Для значения правого локатора ячейки j имеем

$$\phi^g_1(j) = \sum_{p=j+1}^{\infty} g(p) = \sum_{p=j+1}^{i-1} g(p) + B + kB + w(\beta) = \\ = \sum_{p=j+1}^{i-1} g'(p) + B + w(\beta) = \phi^{g'}_1(j).$$

Иными словами сумма, определяющая значение правого локатора для g, отличается от суммы для g' на слагаемое kB, но поскольку в сумме уже есть слагаемое B, то в силу идемпотентности полугруппы G добавление слагаемого kB значения суммы не меняет.

Полностью аналогичные рассуждения проходят и для правого хвоста.

Иллюстрацию этих рассуждений можно найти на рисунке 1.

Тем самым значения хвостов не зависят от k и равны всегда α' и β'.

Остается доказать периодичность с периодом T слова γ''_k из равенства (10).

Пусть $k = 1$ и $\Phi_\sigma\left(\overleftarrow{\alpha}\gamma \;\middle|_i\; \gamma\gamma\vec{\beta}\right) = \left(\overleftarrow{\alpha'} \;\middle|_i\; \gamma'\vec{\beta'}\right)$.

Покажем, что для любого $k > 1$ справедливо $\gamma''_k = \gamma'^k$.

Обозначим $(g'', f'') = \left(\overleftarrow{\alpha}\gamma \;\middle|_i\; \gamma\gamma\vec{\beta}\right)$.

Рассмотрим произвольную ячейку $j \in [i, i+kT-1]$. Пусть $a = \lfloor \frac{j-i}{T} \rfloor$ — целая часть от деления $j-i$ на T, $b = (j-i) \mod T$ — остаток от деления $j-i$ на T, т.е. $j = i + aT + b$.

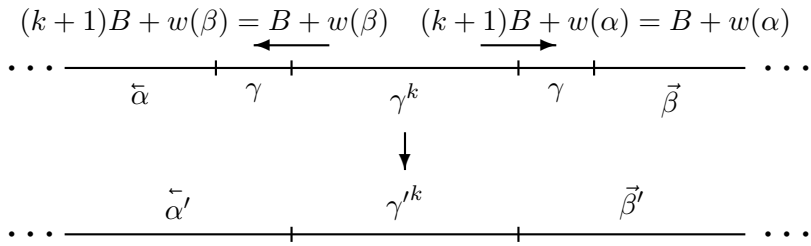


Рис. 1: Преобразование хвостов не зависит от k.

Понятно, что $f(j-1) = f''(b-1)$, $f(j) = f''(b)$ и $f(j+1) = f''(b+1)$, т.е. окрестность ячейки b для f'' и окрестность ячейки j для f совпадают. Посмотрим на значения правого и левого локаторов ячейки j.

$$\phi^g_{-1}(j) = \sum_{p=-\infty}^{j-1} g(p) = w(\alpha) + B + aB + \sum_{p=1}^{b} w(\gamma_p) =$$

$$= w(\alpha) + B + \sum_{p=1}^{b} w(\gamma_p) = \sum_{p=-\infty}^{b-1} g''(p) = \phi^{g''}_{-1}(b).$$

$$\phi^g_1(j) = \sum_{p=j+1}^{\infty} g(p) = \sum_{p=b+1}^{T} w(\gamma_p) + (k-1-a)B + B + w(\beta) =$$

$$= \sum_{p=b+1}^{T} w(\gamma_p) + B + w(\beta) = \phi^{g''}_1(b).$$

Иными словами сумма, определяющая значение как левого, так и правого локатора ячейки j для g, отличается от суммы ячейки b для g'' на слагаемое кратное B, но поскольку в сумме уже есть слагаемое B, то в силу идемпотентности полугруппы G добавление слагаемого кратного B значения суммы не меняет.

Тем самым значения состояний окрестности и значения левого и правого локаторов ячейки j для g совпадает с соответствующими значениями ячейки b для g'', откуда сразу следует, что $\gamma''_k = \gamma'^k$. Иллюстрацию этих рассуждений можно найти на рисунке 2.

Равенство (8) полностью доказано.

Чтобы доказать утверждение леммы для случая $t > 1$ достаточно применить равенство (8) t раз. Лемма доказана.

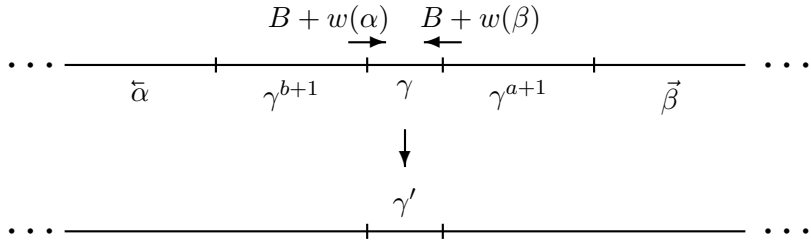


Рис. 2: Преобразование периодической части не зависит от a и b, поскольку $w(\gamma^{a+1}) = (a+1)B = B$ при $a > 0$.

4. Доказательство теоремы 1

Доказательство теоремы 1 является творческим переизложением нижней оценки из работы [13] на случай идемпотентного моноида.

Пусть КАсЛ σ решает задачу $S(l, r)$ за время t.

Рассмотрим случай $l < r$. Допустим, $2t \le l - 1$. Тогда $2t < r - 1$, т.е. $r - 1 - 2t \ge 1$.

Поскольку σ решает задачу $S(l, r)$ за время t, то

$$\Phi^t_\sigma(I_0(l,r)) = \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\, \dot{q_0}^{l-1}\, \dot{C} \,\middle|_1\, \dot{q_0}^{r-1} \dot{R}\, \dot{q_0}^\infty\right) =$$
$$\Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L} \dot{q_0}^{l-1}\, \dot{C} \,\middle|_1\, \dot{q_0}^t\, \dot{q_0}^{r-1-2t}\, \dot{q_0}^t\, \dot{R}\, \dot{q_0}^\infty\right) = \left(\dot{q_0}^\infty\, \dot{L_F}^{l+1} \,\middle|_1\, \dot{q_0}^\infty\right).$$

Тогда по лемме 1 выполнено

$$\Phi^t_\sigma(I_0(l,l)) = \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\, \dot{q_0}^{l-1}\, \dot{C} \,\middle|_1\, \dot{q_0}^t\, \dot{q_0}^{l-1-2t}\, \dot{q_0}^t\, \dot{R}\, \dot{q_0}^\infty\right) =$$
$$= \left(\dot{q_0}^\infty\, \dot{L_F}^{l+1} \,\middle|_1\, \dot{q_0}^\infty\right).$$

Но это противоречит тому, что σ решает задачу $S(l, l)$, поскольку $I_F(l,l) = \left(\dot{q_0}^\infty \,\middle|_0\, \dot{R_F}^{l+1}\, \dot{q_0}^\infty\right)$. Отсюда $2t \ge l$.

Аналогично рассмотрим случай $l \ge r$. Допустим, $2t \le r - 1$, т.е. $r - 1 - 2t \ge 0$.

Поскольку σ решает задачу $S(l,r)$ за время t, то

$$\begin{aligned}\Phi^t_\sigma(I_0(l,r)) &= \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\,\dot{q_0}^{l-1}\,\dot{C}\;\middle|_1\;\dot{q_0}^{r-1}\dot{R}\,\dot{q_0}^\infty\right) = \\ &= \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\,\dot{q_0}^{l-1}\,\dot{C}\;\middle|_1\;\dot{q_0}^{t}\,\dot{q_0}^{r-1-2t}\,\dot{q_0}^{t}\,\dot{R}\,\dot{q_0}^\infty\right) = \\ &= \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\,\dot{q_0}^{l-1}\,\dot{C}\,\dot{q_0}^{t}\;\middle|_{t+1}\;\dot{q_0}^{r-1-2t}\,\dot{q_0}^{t}\,\dot{R}\,\dot{q_0}^\infty\right) = \\ &= \left(\dot{q_0}^\infty\;\middle|_0\;\dot{R_F}^{r+1}\,\dot{q_0}^\infty\right) = \left(\dot{q_0}^\infty\dot{R_F}^{t+1}\;\middle|_{t+1}\;\dot{R_F}^{r-1-2t}\,\dot{R_F}^{t+1}\,\dot{q_0}^\infty\right).\end{aligned}$$

Тогда по лемме 1 выполнено

$$\begin{aligned}\Phi^t_\sigma(I_0(l,l+1)) &= \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\,\dot{q_0}^{l-1}\,\dot{C}\;\middle|_1\;\dot{q_0}^{t}\,\dot{q_0}^{l-2t}\,\dot{q_0}^{t}\,\dot{R}\,\dot{q_0}^\infty\right) = \\ &= \Phi^t_\sigma\left(\dot{q_0}^\infty \dot{L}\,\dot{q_0}^{l-1}\,\dot{C}\,\dot{q_0}^{t}\;\middle|_{t+1}\;\dot{q_0}^{l-2t}\,\dot{q_0}^{t}\,\dot{R}\,\dot{q_0}^\infty\right) = \\ &= \left(\dot{q_0}^\infty\,\dot{R_F}^{t+1}\;\middle|_{t+1}\;x^{l-2t}\,\dot{R_F}^{t+1}\dot{q_0}^\infty\right).\end{aligned}$$

для некоторого $x \in G \times Q$, причем если $r-1-2t>0$, то $x = \dot{R_F}$.

Но это противоречит тому, что σ решает задачу $S(l,l+1)$, поскольку состояние R_F не должно появляться до финального состояния $I_F(l,l+1) = \left(\dot{q_0}^\infty\,\dot{L_F}^{l+1}\;\middle|_1\;\dot{q_0}^\infty\right)$. Отсюда $2t \geq r$.

Теорема 1 доказана.

5. Доказательство теоремы 2

Дадим неформальное описание идемпотентного клеточного автомата с локаторами, который будет решать задачу сравнения по модулю целых чисел.

Пусть l, r — натуральные числа. Ячейку с координатой 0 назовем *центральной*, и она в начальный момент будет находиться в состоянии C. Ячейку с координатой $-l$ назовем *левой*, и она в начальный момент будет находиться в состоянии L. Ячейку с координатой r назовем *правой*, и она в начальный момент будет находиться в состоянии R. Остальные ячейки в начальный момент находятся в состоянии покоя q_0.

Алфавит состояний $Q = \{q_0, L, L_1, L_2, L_R, C, R, R_1, R_2, R_L, L_F, R_F\}$, q_0 — состояние покоя, алфавит вещания $G = \{0,1,2\}^2$, полугрупповая операция — по-компонентный максимум, $e = (0,0)$ — нейтральный элемент полугруппы G, хоть шаблон локаторов стандартный $\mathcal{L} = \{\nu_{-1}, \nu_1\}$, но для упрощения изложения будем считать, что шаблон локаторов содержит только один полный локатор ν_Ω, просто его значения будут получаться суммированием значений локаторов ν_{-1}, ν_1.

От центральной ячейки C вправо отправляется сигнал R_L, влево — L_R, от L вправо отправляется волна из ячеек L, от R влево — волна из ячеек R. Причем по мере распространения волн ячейки, находящиеся в состояниях L, R, L_R и R_L, в этих состояниях и остаются. В момент, когда волна L встречается с волной L_R (ячейка L_R или C справа от L), то ячейка в состоянии L, столкнувшаяся с L_R, отправляет в эфир сигнал $(2,0)$ и переходит в состояние L_2; если эти волны встречаются не вплотную (ячейка L слева, а L_R или C справа от q_0), то эта ячейка в состоянии q_0 отправляет в эфир сигнал $(1,0)$ и переходит в L_1. Аналогично, в момент, когда волна R встречается с волной R_L (ячейка R_L или C слева от R), ячейка в состоянии R, столкнувшаяся с R_L, отправляет в эфир сигнал $(0,2)$ и переходит в состояние R_2; если эти волны встречаются не вплотную (ячейка R справа, а R_L или C слева от q_0), то эта ячейка в состоянии q_0 отправляет в эфир сигнал $(0,1)$ и переходит в R_1. Во всех остальных случаях ячейки молчат.

Таким образом, первый ненулевой сигнал в эфире появляется, когда либо слева, либо справа происходит столкновение волн. Все ячейки слышат одинаковые сигналы из эфира, кроме ячеек, отправивших сигнал, но ячейки, отправившие сигнал, знают, что они отправили, из своих состояний. При этом $l < r$, только если в эфире появляются сигналы $(2,0)$, $(1,0)$ (столкновение произошло только слева) или $(2,1)$ (столкновение волн произошло с двух сторон, но справа столкновение не вплотную). В остальных случаях при ненулевом сигнале в эфире $l \geq r$.

Как только в эфире появляется ненулевой сигнал, то если этот сигнал равен $(2,0)$, $(1,0)$ или $(2,1)$, то все ячейки в состояниях R, R_L, R_1, R_2 переходят в состояние покоя q_0, а ячейки в состояниях L, L_R, L_1, L_2, C — в состояние L_F. В остальных случаях при ненулевом сигнале в эфире все ячейки в состояниях L, L_R, L_1, L_2 переходят в состояние покоя q_0, а ячейки в состояниях R, R_L, R_1, R_2, C — в состояние R_F. При ненулевом сигнале в эфире ячейки ничего в эфир не отправляют, и финальная конфигурация получается на следующем такте после появления в эфире ненулевого сигнала.

Оценим общее время алгоритма. На такте $\lfloor \frac{\min(l,r)-1}{2} \rfloor$ происходит столкновение волн, и в эфир отправляется ненулевой сигнал, который ячейки слышат на следующем такте, после чего переходят в финальное состояние. Таким образом, финальное состояние получается через 2 такта после столкновения волн, в момент $\lfloor \frac{\min(l,r)-1}{2} \rfloor + 2 = \lceil \frac{\min(l,r)}{2} \rceil + 1$.

Приведем формальное описание функций перехода и вещания.

Правила перехода ($*$ обозначает любой символ, в скобках — пояснение, где конфигурация встречается):

1. Распространение волн:
   а) $\varphi(q_0, q_0, x, e) = x$, $x \in \{L_R, R\}$ (распространение волны влево);
   б) $\varphi(q_0, q_0, C, e) = L_R$ (запуск волны влево от центра);
   в) $\varphi(x, q_0, q_0, e) = x$, $x \in \{R_L, L\}$ (распространение волны вправо);
   г) $\varphi(C, q_0, q_0, e) = R_L$ (запуск волны вправо от центра);
2. Столкновение волн:
   а) $\varphi(L, q_0, x, e) = L_1$, $x \in \{C, L_R\}$ (чётное l);
   б) $\varphi(*, L, x, e) = L_2$, $x \in \{C, L_R\}$ (нечётное l);
   в) $\varphi(x, q_0, R, e) = R_1$, $x \in \{C, R_L\}$ (чётное r);
   г) $\varphi(x, R, *, e) = R_2$, $x \in \{C, R_L\}$ (нечётное r);
3. Сравнение и переход в финальное состояние:
   а) $\varphi(*, x, *, (a, b)) = L_F$, $a > b$, $x \in \{C, L, L_R\}$ (ячейки слева от центра при $l < r$);
   б) $\varphi(*, L_a, *, (*, b)) = L_F$, $a > b$ (ячейка, запустившая сигнал слева от центра при $l < r$);
   в) $\varphi(*, x, *, (a, b)) = R_F$, $a \leq b$, $b \neq 0$, $x \in \{C, R, R_L\}$ (ячейки справа от центра при $l \geq r$);
   г) $\varphi(*, R_b, *, (a, *)) = R_F$, $a \leq b$ (ячейка, запустившая сигнал справа от центра при $l \geq r$);
4. Оставшиеся случаи:
   а) $\varphi(*, x, *, e) = x$, $x \in \{L, R, L_R, R_L, C, L_F, R_F\}$ за исключением случаев 2 б) и 2 г) (сохранение состояний);
   б) Во всех остальных случаях $\varphi = q_0$.

Функция вещания определяется следующим образом:

$$\psi(x, y, z, g) = \begin{cases} (a, 0), & q' = L_a, a \in \{1, 2\}, \\ (0, b), & q' = R_b, b \in \{1, 2\}, \\ (0, 0) & \text{иначе}, \end{cases} \qquad \text{где } q' = \varphi(x, y, z, g).$$

Идемпотентный клеточный автомат с локаторами, решающий задачу сравнения по модулю целых чисел полностью описан.

Теорема 2 доказана.

Легко построить обычный клеточный автомат, который будет решать задачу сравнения по модулю целых чисел $S(l, r)$ за время $O(l + r)$. Для

этого достаточно послать волны от левой и правой ячеек к центральной ячейке, и какая из волн придет к центральной ячейке быстрее, тот отрезок и короче, т.е. через $\min(l, r)$ тактов мы будем знать результат и нам остается построить финальную конфигурацию. Для этого мы посылаем от центральной ячейки к более далекому краю стирающую волну, а на более коротком отрезке можно запустить задачу синхронизации стрелков, чтобы состояния R_F (или L_F) появились одновременно. Это займет времени, равное 3 длинам отрезка. Но к тому времени дальний край еще может быть не стерт, поэтому лучше дойти до дальнего края, стереть его, вернуться к центральной ячейке и только после этого запустить решение задачи синхронизации стрелков. Например, если $l < r$, то суммарное время работы нашего клеточного автомата будет $l + 2r + 3l$.

Тем самым, если считать, что l и r имеют одинаковый порядок, то и клеточный автомат, и идемпотентный клеточный с локаторами будут решать задачу за линейное время, и тем самым идемпотентный клеточный с локаторами не имеет преимущества по порядку по сравнению с обычным клеточным автоматом.

6. Перенос нижней оценки на другие задачи

Рассмотрим задачу, которую назовем модифицированной задачей сравнения по модулю целых чисел и обозначим $S_M(l, r)$. Начальная конфигурация в этой задаче такая же, что и в задаче сравнения по модулю целых чисел $S(l, r)$. Также считается, что в сигнале вещания имеется специальная компонента, которая отвечает за результат задачи. А именно, если по этой компоненте выдается 0, то решения еще нет, если по этой компоненте появилась 1, то решение получено и $l < r$, если по этой компоненте выдается 2, то решение получено и $l \geq r$. После получения решения состояния автомата не меняются и автомат молчит (выдает в эфир нейтральный сигнал).

Обозначим через $T_M(\sigma, l, r)$ — время, за которое произвольный клеточный автомат с локаторами σ решает модифицированную задачу сравнения по модулю целых чисел $S_M(l, r)$, (положим $T_M(\sigma, l, r) = \infty$, если автомат σ не решает задачу $S_M(l, r)$).

Утверждение 1. *Если $l, r \in \mathbb{N}$, то для любого одномерного идемпотентного клеточного автомата с локаторами σ выполнено $T_M(\sigma, l, r) \geq \frac{\min(l,r)}{2} - 1$.*

Доказательство. Рассмотрим произвольную модифицированную задачу сравнения по модулю целых чисел $S_M(l, r)$.

Предположим, что некоторый одномерный идемпотентный клеточный автомат с локаторами σ решает модифицированную задачу за время t и $t < \frac{s}{2} - 1$, где $s = \min(l, r)$. На базе этого автомата построим автомат σ', который будет решать обычную задачу сравнения по модулю целых чисел за время меньшее чем $\frac{s}{2}$, что будет противоречить теореме 1, и докажет неверность предположения.

Автомат σ' будет надстройкой над автоматом σ. Сначала также, как это сделано в автомате, описанном в доказательстве теоремы 2, добавим к автомату σ компоненту вещания (и возможно добавим состояния для запоминания позиций левой, правой и центральной ячеек), с помощью которой разделим всю прямую на левый и правый хвосты и левый и правый отрезок. В момент, когда автомат σ сообщит о результате через компоненту результата сигнала вещания, мы используем этот результат, чтобы получить финальную конфигурацию задачи сравнения по модулю целых чисел. Таким образом автомат σ' выдаст финальную конфигурацию на такт позже сигнала результата автомата σ.

Утверждение 1 доказано. □

Тем самым мы показали, что хоть мы в доказательстве теоремы 1 по существу опирались на особенности финальной конфигурации, а именно, что она представляет собой непрерывный отрезок активных клеток от центральной до ближайшей левой или правой ячеек, но оказывается это не имеет принципиального значения.

Рассмотрим задачу поиска ближайшего соседа описанную в работах [12, 13]. В этой задаче в начальной конфигурации на прямой разбросано некоторое количество точек. Надо найти точку ближайшую к началу координат.

Покажем, что эту задачу нельзя решить одномерным идемпотентным клеточным автоматом с локаторами за время, меньшее чем $\frac{s}{2} - 1$, где s — расстояние до ближайшего соседа. Если бы могли это сделать, то взяв в качестве начальной конфигурации конфигурацию, в которой только одна точка находится левее начала координат, и одна точка — правее начала координат, мы фактически получим начальную конфигурацию модифицированной задачи сравнения по модулю целых чисел. И решив, ее за время, меньшее чем $\frac{s}{2} - 1$, получим противоречие с утверждением 1.

Рассмотрим задачу сложения целых векторов, описанную в работе [15]. В этой задаче в начальной конфигурации отмечены 3 ячейки: ячейка с координатой 0 как начало координат, и две ячейки с координатами a и b интерпретируемые как слагаемые. В финальной конфигурации только 2 ячейки должны быть не в состоянии покоя: ячейка с координатой 0 в состоянии C_F и ячейка с координатой $a + b$ в состоянии R_F, причем

в промежуточных состояниях состояния C_F и R_F не появляются. В случае, если $a = -b$, считаем, что в финальной конфигурации ячейка с координатой 0 оказывается в состоянии C_{F0}, а остальные ячейки — в состоянии покоя.

Покажем, что эту задачу нельзя решить одномерным идемпотентным клеточным автоматом с локаторами за время меньшее, чем $\frac{s}{2} - 2$ тактов, где $s = \min(|a|, |b|)$. Если бы мы могли решать эту задачу за время $t < \frac{s}{2} - 2$, то мы могли бы решить модифицированную задачу сравнения по модулю целых чисел $S_M(l, r)$ следующим образом. Зададим начальную конфигурацию задачи сложения чисел так, что $a = -l$, а $b = r$. После этого решаем задачу сложения чисел за t тактов. В момент своего появления ячейка в состоянии C_F что-нибудь кричит в эфир, в зависимости от того в какой локатор ячейка в состоянии R_F слышит этот сигнал, она понимает с какой стороны от начала координат она находится, и на следующий такт выдает финальный сигнал 2, если она находится левее начала координат, и — сигнал 1, если — правее. Если $l = r$, то в момент появления состояния C_{F0} можно сразу подавать финальный сигнал 2. Получается, что мы решили задачу сравнения чисел за время $t + 1 < \frac{s}{2} - 1$, что противоречит утверждению 1.

Остается заметить, что каждую из задач из этого раздела можно решить обычным клеточным автоматом за линейное время (задачу поиска ближайшего соседа с некоторыми оговорками). И отсюда получаем, что для всех этих задач одномерный идемпотентный клеточный автомат с локаторами не дает преимущества по порядку по сравнению с обычным клеточным автоматом.

Список литературы

[1] фон Дж. Нейман, *Теория самовоспроизводящихся автоматов*, Мир, М, 1971.

[2] von J. Neumann, *Collected Works*, New York, 1961–1963.

[3] von J. Neumann, *Theory of self-reproducing automata*, London, 1966.

[4] A. Burks, *Essays on Cellular Automata*, University of Illinois Press, 1971.

[5] Э. Ф. Мур, “Математические модели самовоспроизведения”, *Математические проблемы в биологии*, Мир, М, 1966.

[6] В. Б. Кудрявцев, А. С. Подколзин, А. А. Болотов, *Основы теории однородных структур*, Наука, М, 1971, 296 с.

[7] В. Б. Кудрявцев, Э. Э. Гасанов, А. С. Подколзин, *Теория интеллектуальных систем: в 4 т. Т. 4. Теория автоматов*, Издательский дом МГУ, М, 2026, 305 с.

[8] Г. В. Калачев, Е. Е. Титова, "О мере множества законов движения точки, реализуемых клеточными автоматами", *Интеллектуальные системы. Теория и приложения*, **22**:3 (2018), 105–125.

[9] Э. Э. Гасанов, "Клеточные автоматы с локаторами", *Интеллектуальные системы. Теория и приложения*, **24**:2 (2020), 120–133.

[10] F. R. Moore, G. G. Langdon, "A generalized firing squad problem", *Information and Control*, **12**:3 (1968), 212–220.

[11] Е. Е. Титова, "Конструирование движущихся изображений клеточными автоматами", *Интеллектуальные системы. Теория и приложения*, **18**:1 (2014), 153–180.

[12] Д. И. Васильев, "Поиск ближайшего соседа на прямой с помощью клеточного автомата с локаторами", *Интеллектуальные системы. Теория и приложения*, **24**:3 (2020), 99–119.

[13] Д. И. Васильев, Э. Э. Гасанов, "Нижняя оценка сложности задачи поиска ближайшего соседа на прямой с помощью клеточного автомата с локаторами", *Вестник Московского университета. Серия 1: Математика. Механика*, 2023, № 5, 33–39. DOI: 10.55959/MSU0579-9368-1-64-5-5.

[14] Г. В. Калачев, "Замечания к определению клеточного автомата с локаторами", *Интеллектуальные системы. Теория и приложения*, **24**:3 (2020), 47–56.

[15] Д. Э. Ибрагимова, "Сложение векторов на прямой с помощью клеточного автомата с локаторами", *Интеллектуальные системы. Теория и приложения*, **26**:4 (2022), 134–162.

[16] Э. Э. Гасанов, "Клеточные автоматы с локаторами как модель устройств с беспроводной связью", *Математические вопросы кибернетики*, Вып 21, ФИЗМАТЛИТ, М, 2023, 5–51. DOI: 10.20948/mvk-2023-5.

Статья поступила 16 мая 2026 г.

On the Weakness of One-Dimensional Idempotent Cellular Automata with Locators

E. E. Gasanov

We consider one-dimensional cellular automata with locators for which the semigroup defined on the broadcast alphabet is an idempotent monoid. It has previously been shown that complex cellular automata problems such as the firing squad synchronization problem or the problem of controlling the motion of a point on a line are solved quite simply in idempotent cellular automata with locators, and the problems essentially degenerate. In this paper, we show that for problems that require distance comparison, idempotent cellular automata with locators provide no asymptotic advantage over conventional cellular automata. However, the use of non-idempotent cellular automata provides a significant time advantage.

Keywords: Idempotent cellular automata with locators, comparison and addition of integers.

References

[1] von J. Neumann, *Theory of self-reproducing automata*, Mir, M, 1971 (in Russian).

[2] von J. Neumann, *Collected Works*, New York, 1961–1963.

[3] von J. Neumann, *Theory of self-reproducing automata*, London, 1966.

[4] A. Burks, *Essays on Cellular Automata*, University of Illinois Press, 1971.

[5] E. F. Moore, "Mathematical models of self-reproduction", *Mathematical problems in biology*, Mir, M, 1966 (in Russian).

[6] V. B. Kudryavtsev, A. S. Podkolzin, A. A. Bolotov, *Fundamentals of the theory of homogeneous structures*, Nauka, M, 1971 (in Russian), 296 pp.

[7] V. B. Kudryavtsev, E. E. Gasanov, A. S. Podkolzin, *Theory of Intelligent Systems: In 4 vol. Vol. 4. Automata Theory*, Moscow University Press, M, 2026 (in Russian), 305 pp.

[8] G. V. Kalachev, E. E. Titova, "On the measure of the set of laws of motion of a point realized by cellular automata", *Intelligent Systems: Theory and Applications*, **22**:3 (2018), 105–125 (in Russian).

[9] E. E. Gasanov, "Cellular automata with locators", *Intelligent Systems: Theory and Applications*, **24**:2 (2020), 120–133 (in Russian).

[10] F. R. Moore, G. G. Langdon, "A generalized firing squad problem", *Information and Control*, **12**:3 (1968), 212–220.

[11] E. E. Titova, “Construction of moving images using cellular automata”, *Intelligent Systems: Theory and Applications*, **18**:1 (2014), 153–180 (in Russian).

[12] D. I. Vasilev, “Finding the nearest neighbor on a line using a cellular automaton with locators”, *Intelligent Systems: Theory and Applications*, **24**:3 (2020), 99–119 (in Russian).

[13] D. I. Vasilev, E. E. Gasanov, “A Lower Bound on Complexity of a Locator Cellular Automaton Solution for the Closest Neighbor Search Problem”, *Moscow University Mathematics Bulletin*, **78**:5 (2023), 244-252. DOI: 10.3103/S0027132223050078 (in Russian).

[14] G. V. Kalachev, “Notes on the definition of a cellular automaton with locators”, *Intelligent Systems: Theory and Applications*, **24**:3 (2020), 47–56 (in Russian).

[15] D. E. Ibragimova, “Vector addition on a line using a cellular automaton with locators”, *Intelligent Systems: Theory and Applications*, **26**:4 (2022), 134–162 (in Russian).

[16] E. E. Gasanov, “Cellular automata with locators as a model for wireless communication devices”, *Matematicheskie voprosy kibernetiki*, Issue 21, FIZMATLIT, M, 2023, 5–51. DOI: 10.20948/mvk-2023-5 (in Russian).

Received on May 16, 2026

Проблема полноты в классе линейных p-адических автоматов

М. Э. Калашников*

В работе исследована проблема K-полноты в классе линейных p-адических автоматов для нечётного простого числа p. Получены критерии полноты для однородных и неоднородных линейных p-адических функций. Полученные результаты обобщают известные свойства линейных 2-адических автоматов на случай произвольного нечётного простого основания. Кроме того, построена континуальная строго возрастающая цепочка K-замкнутых классов I_α, $\alpha \in (0, 1]$.

Ключевые слова: конечный автомат, p-адическое число, линейный p-адический автомат, операции композиции, обратная связь, проблема полноты, замкнутый класс.

Введение

В настоящее время активно изучаются свойства класса конечных автоматов и его многочисленных подклассов. В частности, в работах [2–4] рассматривается задача K-полноты для линейных автоматов над конечными полями.

Важным подклассом множества конечных автоматов является класс p-адических автоматов, систематическое изучение которого началось с работы [5]. Для исследования таких автоматов используется аппарат p-адических чисел.

В работе [1] была решена проблема K-полноты для класса линейных 2-адических автоматов: описаны все предполные классы и доказана алгоритмическая разрешимость проверки полноты конечных систем.

В настоящей работе полученные результаты обобщаются на случай линейных p-адических автоматов для произвольного нечётного простого числа p.

* *Калашников Максим Эдуардович* — аспирант каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: maxkalash98@gmail.com.

Kalashnikov Maksim Eduardovich — Postgraduate student, Chair of Mathematical Theory of Intelligent Systems, Faculty of Mechanics and Mathematics, Lomonosov Moscow State University.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

1. Начальные установки

В работе изучается класс L_p линейных p-адических автоматов, вычисляющих линейные функции на кольце целых p-адических чисел $\mathbb{Z}_p$ с коэффициентами из подкольца

$$\mathbb{Z}_{(p)} = \mathbb{Z}_p \cap \mathbb{Q}.$$

Здесь и далее p обозначает произвольное нечётное простое число. Множество $E_p = \{0, 1, \ldots, p-1\}$ представляет собой используемый алфавит.

Определение 1. Конечный инициальный автомат с входным алфавитом E_p^n и выходным алфавитом E_p называется линейным p-адическим автоматом, если для определяемого им отображения $f(x_1, \ldots, x_n)$ существуют числа $r_0, r_1, \ldots, r_n \in \mathbb{Z}_{(p)}$ такие, что для любых $\alpha_1, \ldots, \alpha_n \in \mathbb{Z}_p$

$$f(\alpha_1, \ldots, \alpha_n) = \sum_{i=1}^{n} r_i \alpha_i + r_0.$$

В этом случае будем говорить, что функция f имеет разложение

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0. \tag{1}$$

Всюду в работе будут использоваться следующие обозначения и соглашения. Запись Fb_x обозначает операцию применения обратной связи к переменной x.

Пусть A — числовое множество и $k \in \mathbb{Q}$. Положим

$$kA = \{ka \mid a \in A\}, \qquad k + A = \{k + a \mid a \in A\}.$$

Если $k \neq 0$, то также положим

$$\frac{A}{k} = \left\{ \frac{a}{k} \;\middle|\; a \in A \right\}.$$

Для простого числа q определим q-адическую оценку

$$v_q \colon \mathbb{Q} \longrightarrow \mathbb{Z} \cup \{+\infty\}.$$

Если $r \neq 0$, то единственным образом

$$r = q^k \frac{u}{v}, \qquad k \in \mathbb{Z}, \quad u, v \in \mathbb{Z}, \quad v > 0, \quad (u, v) = 1, \quad q \nmid uv.$$

Положим $v_q(r) = k$. Кроме того, $v_q(0) = +\infty$.

Для $r \in \mathbb{Z}_{(p)}$ через $\overline{r}$ будем обозначать его остаток от деления на p. Другими словами, $r \in \overline{r} + p\mathbb{Z}_{(p)}, \overline{r} \in E_p$.

Переменная x_i называется непосредственной, если для соответствующего коэффициента $v_p(r_i) = 0$. Операция обратной связи применима к переменной тогда и только тогда, когда соответствующая переменная не является непосредственной. Через L_p^0 будем обозначать подмножество L_p, состоящее из однородных функций (свободный член в разложении (1) равен нулю).

Для функции

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0$$

положим

$$U(f) = \{r_i \mid 1 \leq i \leq n\},$$

причем, при использовании представителей из $U(f)$ договоримся брать дроби в несократимом виде. Переменная x_i называется существенной, если $r_i \neq 0$.

Для $M \subseteq L_p$ через $K(M)$ обозначим замыкание M относительно отождествления и переименования переменных, подстановки и обратной связи. Множество M называется K-замкнутым, если $K(M) = M$, и K-полным в L_p, если $K(M) = L_p$.

2. Некоторые свойства L_p

Начнём с леммы, описывающей действие операций композиции на линейные p-адические функции. Аналогичное утверждение для 2-адического случая доказано в работе [1].

Лемма 1. *Пусть для функций f и f' выполнено:*

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0,$$

$$f'(x_{n+1}, \ldots, x_{n+n'}) = \sum_{i=n+1}^{n+n'} r_i x_i + r'_0.$$

Через f_1, f_2, f_3 обозначим функции, получаемые, соответственно, отождествлением переменных x_{n-1} и x_n функции f (при $n \geq 2$), переименованием переменных функции f с $x_1, \ldots, x_n$ на $x'_1, \ldots, x'_n$, подстановкой функции f вместо переменной $x_{n+n'}$ функции f'. Если к переменной x_n

функции f применима обратная связь, то через f_4 обозначим результат применения обратной связи к этой переменной. Тогда выполнены следующие равенства:

$$f_1(x_1,\dots,x_{n-1}) = \sum_{i=1}^{n-2} r_i x_i + (r_{n-1}+r_n)x_{n-1} + r_0,$$

$$f_2(x'_1,\dots,x'_n) = \sum_{i=1}^{n} r_i x'_i + r_0,$$

$$f_3(x_1,\dots,x_{n+n'-1}) = \sum_{i=n+1}^{n+n'-1} r_i x_i + \sum_{i=1}^{n} r_i r_{n+n'} x_i + r'_0 + r_{n+n'} r_0,$$

$$f_4(x_1,\dots,x_{n-1}) = \sum_{i=1}^{n-1} \frac{r_i}{1-r_n} x_i + \frac{r_0}{1-r_n}.$$

Далее покажем, что класс линейных p-адических функций порождается простым сумматором и константой.

Определение 2. Автомат, реализующий p-адический сумматор $x_1 + x_2$, задаётся следующей системой канонических уравнений:

$$\begin{aligned} q(0) &= 0,\\ q(t+1) &= \left\lfloor \frac{a(t)+b(t)+q(t)}{p} \right\rfloor,\\ y(t) &= a(t)+b(t)+q(t) - p\,q(t+1). \end{aligned}$$

Лемма 2.

$$K(\{x_1+x_2, 1\}) = L_p.$$

Доказательство. Положим $M = \{x_1+x_2, 1\}$. Будем последовательно получать всё более сложные функции из $K(M)$.

Последовательно применяя к сумматору x_1+x_2 операции подстановки и переименования переменных, получаем

$$x_1 + \dots + x_n \in K(M),$$

а значит, отождествляя переменные, имеем

$$\forall n \in \mathbb{N}: \quad nx \in K(M).$$

В частности, получаем $py \in K(M)$ и $(p-1)x \in K(M)$, следовательно,

$$(p-1)x + py \in K(M).$$

Применим обратную связь по переменной y:

$$\mathrm{Fb}_y\big((p-1)x+py\big)=\frac{p-1}{1-p}\,x=-x\in K(M).$$

Подставляя функцию nx в функцию $-x$, для каждого $n\in\mathbb{N}$ получаем функцию $-nx$. Следовательно,

$$\forall k\in\mathbb{Z}:\quad kx\in K(M).$$

Пусть $r\in\mathbb{Z}_{(p)}$. Тогда $r=\dfrac{a}{b}$, где $a\in\mathbb{Z}$, $b\in\mathbb{N}$ и $p\nmid b$. Так как $(p,b)=1$, по алгоритму Евклида существуют $k,\ell\in\mathbb{Z}$ такие, что

$$kp+\ell b=1,\qquad\text{то есть}\qquad b=\frac{1-kp}{\ell}.$$

Рассмотрим функцию

$$g(x,y)=a\ell\,x+kp\,y\in K(M).$$

Применяя обратную связь к переменной y, получаем

$$\mathrm{Fb}_y(g)=\frac{a\ell}{1-kp}\,x=\frac{a}{(1-kp)/\ell}\,x=\frac{a}{b}\,x=rx.$$

Значит,

$$\forall r\in\mathbb{Z}_{(p)}:\quad rx\in K(M).$$

Пусть $f\in L_p$, то есть

$$f(x_1,\dots,x_n)=\sum_{i=1}^{n}r_ix_i+r_0,\qquad r_i,r_0\in\mathbb{Z}_{(p)}.$$

По доказанному выше для каждого i имеем $r_ix_i\in K(M)$, а также $r_0x_0\in K(M)$. Тогда

$$\sum_{i=1}^{n}r_ix_i+r_0x_0\in K(M),$$

и, подставив 1 вместо x_0, получаем $f\in K(M)$. Значит, $L_p\subseteq K(M)$.

Обратное включение также выполняется, поскольку $M\subseteq L_p$ и класс L_p замкнут относительно операций композиции, то есть $K(M)\subseteq L_p$. Следовательно, $K(M)=L_p$. □

Теперь определим некоторые важные классы функций в L_p:

Определение 3.

$$T_a = \{f \in L_p \mid f(\alpha_1, \ldots, \alpha_n) \in a + p\mathbb{Z}_p \text{ для всех } \alpha_1, \ldots, \alpha_n \in a + p\mathbb{Z}_p\}.$$

$$V_1 = \{f \in L_p \mid f \text{ имеет не более одной непосредственной переменной }\}.$$

$$V_o = \left\{ f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0 \;\middle|\; \sum_{i=1}^{n} r_i \equiv 1 \pmod{p} \right\}.$$

$$I = \left\{ f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0 \;\middle|\; \sum_{i=1}^{n} |r_i| \leq 1 \right\}.$$

$$R_c(q) = \{f \in L_p \mid f \text{ имеет не одну существенную переменную, и } \forall \frac{u}{v} \in U(f), (u,v) = 1 : (u,q) = q,\ (q,v) = 1\} \cup$$

$$\cup \{f \in L_p \mid f \text{ имеет одну существенную переменную, и } \forall \frac{u}{v} \in U(f) \setminus \{0\}, (u,v) = 1 : (v,q) = 1\},\quad q \text{ - простое, не равное } p.$$

$$R_d(q) = \{f \in L_p \mid f \text{ имеет не одну непосред. переменную, и } \forall \frac{u}{v} \in U(f), (u,v) = 1 : (u,q) = q,\ (q,v) = 1\} \cup$$

$$\cup \{f \in L_p \mid f \text{ имеет одну непосред. переменную, и } \forall \frac{u}{v} \in U(f) \cap p\mathbb{Z}_{(p)}, (u,v) = 1 : (u,q) = q, \forall \frac{u}{v} \in U(f), \frac{u}{v} \notin p\mathbb{Z}_{(p)}, (u,v) = 1 : (v,q) = 1\}, q \text{ - простое, } q \neq p.$$

Лемма 3. *Классы T_a ($a \in E_p$), V_1, V_o, I, а также $R_c(q)$ и $R_d(q)$ для всех простых $q \neq p$ являются попарно различными замкнутыми классами в L_p.*

Доказательство. Доказательство замкнутости может быть проведено аналогично 2-адическому случаю [1]. Доказательство попарной взаимной невключенности также аналогично 2-адическому случаю. □

3. Теорема о полноте для линейных p-адических автоматов

3.1. Случай однородных функций

Положим

$$J_p^0 = \{V_o \cap L_p^0, V_1 \cap L_p^0, I \cap L_p^0\} \cup \{R_c(\pi) \cap L_p^0, R_d(\pi) \cap L_p^0 \mid \pi \neq p - \text{простое}\}.$$

Далее нам понадобится теорема Коши-Дэвенпорта:

Теорема 1 (Дэвенпорт, [6]). *Пусть p — простое число, а A, B — непустые подмножества поля $\mathbb{F}_p$. Тогда*

$$|A+B| \geq \min\{p,\ |A|+|B|-1\}.$$

Лемма 4. *Пусть*

$$M \subseteq L_p^0, \qquad M \not\subseteq V_1, \qquad M \not\subseteq V_o.$$

Тогда

$$0 \in K(M).$$

Доказательство. Так как $M \not\subseteq V_o$, по определению, найдётся функция

$$g(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i \in M$$

такая, что

$$\sum_{i=1}^{n} \overline{r_i} \neq 1.$$

Отождествим все переменные функции g и получим одноместную однородную функцию $g_1(x) = qx$.

Если $\overline{q} = 0$, то $q \in p\mathbb{Z}_{(p)}$. Следовательно, к переменной x функции qx применима обратная связь. Так как свободный член функции qx равен нулю, получаем

$$\mathrm{Fb}_x(qx) = \frac{0}{1-q} = 0 \implies 0 \in K(M).$$

Остаётся рассмотреть случай $\overline{q} \neq 0, \qquad \overline{q} \neq 1$.

Так как $M \not\subseteq V_1$, существует функция

$$h(x_1, \ldots, x_n) = \sum_{i=1}^{n} a_i x_i \in M$$

имеющая не менее двух непосредственных переменных. Перенумеруем переменные функции h так, чтобы $x_1, \ldots, x_m$ были непосредственными, где $m \geq 2$. Тогда

$$h(x_1, \ldots, x_n) = \sum_{i=1}^{m} a_i x_i + p \sum_{i=m+1}^{n} b_i x_i, \qquad \overline{a_i} \neq 0.$$

Положим

$$S = \sum_{i=1}^{m} \overline{a_i}.$$

Если $S = 0$, положим

$$\alpha = a_1, \qquad \beta = \sum_{i=2}^{m} a_i + p \sum_{i=m+1}^{n} b_i.$$

Тогда

$$\overline{\alpha} = \overline{a_1} \neq 0, \qquad \overline{\beta} = S - \overline{a_1} = -\overline{a_1} \neq 0.$$

Пусть теперь $S \neq 0$. Если существует $j \in \{1, \ldots, m\}$ такой, что

$$\overline{a_j} \neq S,$$

то положим

$$\alpha = a_j, \qquad \beta = \sum_{\substack{i=1 \\ i \neq j}}^{m} a_i + p \sum_{i=m+1}^{n} b_i.$$

В этом случае

$$\overline{\alpha} = \overline{a_j} \neq 0, \qquad \overline{\beta} = S - \overline{a_j} \neq 0.$$

Остаётся случай, когда

$$\overline{a_j} = S \qquad \text{для всех } j \in \{1, \ldots, m\}.$$

Тогда

$$S = \sum_{i=1}^{m} \overline{a_i} = mS.$$

Поскольку $S \neq 0$, имеем

$$m \equiv 1 \pmod{p}.$$

Следовательно, $m \geq p + 1$. Положим

$$\alpha = a_1 + a_2, \qquad \beta = \sum_{i=3}^{m} a_i + p \sum_{i=m+1}^{n} b_i.$$

Так как в рассматриваемом случае $p > 2$, получаем

$$\overline{\alpha} = 2S \neq 0, \qquad \overline{\beta} = (m-2)S = -S \neq 0.$$

Таким образом, соответствующим отождествлением переменных функции h получаем

$$B(x,y) = \alpha x + \beta y \in K(M), \qquad \overline{\alpha} \neq 0, \quad \overline{\beta} \neq 0.$$

Теперь рассмотрим множество

$$C = \left\{ \overline{c} \in \mathbb{F}_p \;\middle|\; c \in \mathbb{Z}_{(p)}, \quad cx \in K(M) \right\}.$$

Иными словами, C состоит из остатков коэффициентов при x у одноместных однородных функций, выразимых из M.

Так как $g_1(x) = qx \in K(M)$, имеем $\overline{q} \in C$. Кроме того, подставляя функцию qx в саму себя, получаем $q^2 x \in K(M)$, поэтому $\overline{q}^2 \in C$.

Поскольку $\overline{q} \neq 0$ и $\overline{q} \neq 1$, имеем $\overline{q}^2 \neq \overline{q}$. Следовательно, $|C| \geq 2$.

Предположим, что $|C| < p$. Поскольку $B(x,y) \in K(M)$, множество C удовлетворяет включению

$$\overline{\alpha} C + \overline{\beta} C \subseteq C.$$

Так как $\overline{\alpha} \neq 0$, $\overline{\beta} \neq 0$, умножение на $\overline{\alpha}$ и $\overline{\beta}$ является биекцией поля $\mathbb{F}_p$. Поэтому

$$|\overline{\alpha} C| = |C|, \qquad |\overline{\beta} C| = |C|.$$

По теореме Коши–Дэвенпорта имеем

$$|\overline{\alpha} C + \overline{\beta} C| \geq \min\{p,\ 2|C| - 1\}.$$

Если $2|C| - 1 \leq p$, то $|\overline{\alpha} C + \overline{\beta} C| \geq 2|C| - 1 > |C|$ (так как $|C| > 1$).

Если $2|C| - 1 \geq p$, то $|\overline{\alpha} C + \overline{\beta} C| \geq p > |C|$ (по предположению).

Таким образом, имеем $|\overline{\alpha} C + \overline{\beta} C| > |C|$. С другой стороны, из включения $\overline{\alpha} C + \overline{\beta} C \subseteq C$ следует $|\overline{\alpha} C + \overline{\beta} C| \leq |C|$. Пришли к противоречию. Следовательно, $|C| = p$ и, в частности, $0 \in C$.

Значит, существует функция $cx \in K(M)$ такая, что $\overline{c} = 0$, то есть $c \in p\mathbb{Z}_{(p)}$. К переменной x функции cx применима обратная связь, и, поскольку функция однородна, получаем $\mathrm{Fb}_x(cx) = 0$. Следовательно,

$$0 \in K(M).$$

Лемма доказана. □

Следующие четыре леммы описывают основные этапы доказательства теоремы о полноте. Их доказательство полностью аналогично 2-адическому случаю [1].

Лемма 5 (Получение масштабированного сумматора)**.** *Пусть*

$$M \subseteq L_p^0, \qquad 0 \in K(M), \qquad M \not\subseteq V_1.$$

Тогда существует число

$$s \in \mathbb{Z}_{(p)} \setminus p\mathbb{Z}_{(p)}, \qquad s > 0,$$

такое, что

$$sx_1 + sx_2 \in K(M).$$

Лемма 6 (Получение множителя, большего единицы)**.** *Пусть*

$$M \subseteq L_p^0, \qquad 0 \in K(M), \qquad M \not\subseteq I.$$

Тогда существует число $r \in \mathbb{Z}_{(p)}$, $r > 1$, такое, что

$$rx \in K(M).$$

Лемма 7 (Получение дробного множителя с простым q в знаменателе)**.** *Пусть $q \neq p$ — простое число. Пусть*

$$M \subseteq L_p^0, \qquad 0 \in K(M),$$

и пусть для некоторого $s \in \mathbb{Z}_{(p)} \setminus p\mathbb{Z}_{(p)}$ выполнено

$$sx_1 + sx_2 \in K(M).$$

Если $M \not\subseteq R_c(q)$ и $M \not\subseteq R_d(q)$, то существует число $\rho_q \in \mathbb{Z}_{(p)}$ такое, что

$$v_q(\rho_q) < 0 \qquad \text{и} \qquad \rho_q x \in K(M).$$

Лемма 8 (Переход от масштабированного сумматора к точному сумматору)**.** *Пусть*

$$M \subseteq L_p^0, \qquad 0 \in K(M).$$

Предположим, что в $K(M)$ содержатся функции

$$sx_1 + sx_2, \qquad rx,$$

где

$$s, r \in \mathbb{Z}_{(p)}, \qquad s \notin p\mathbb{Z}_{(p)}, \quad s > 0, \qquad r > 1.$$

Пусть для каждого простого $q \neq p$, делящего числитель числа s в несократимой записи, в $K(M)$ содержится функция $\rho_q x$ со свойством $v_q(\rho_q) < 0$. Тогда

$$x_1 + x_2 \in K(M).$$

Теорема 2 (Теорема полноты для однородных линейных p-адических автоматов)**.** *Пусть*

$$M \subseteq L_p^0.$$

Если для каждого класса $\Theta \in J_p^0$ *выполняется* $M \nsubseteq \Theta$, *тогда*

$$K(M) = L_p^0.$$

Доказательство. По условию $M \nsubseteq V_1, M \nsubseteq V_o$. По лемме о получении нуля имеем $0 \in K(M)$.

Так как $M \nsubseteq V_1$, по лемме о масштабированном сумматоре существует $s \notin p\mathbb{Z}_{(p)}, s > 0$, такое, что $sx_1 + sx_2 \in K(M)$.

Так как $M \nsubseteq I$, по лемме о функции вне I существует $r \in \mathbb{Z}_{(p)}, r > 1$, такое, что $rx \in K(M)$.

Запишем

$$s = \frac{u}{v}, \qquad (u, v) = 1, \qquad u > 0, \qquad v > 0.$$

Пусть $q_1, \ldots, q_m$ — все простые делители числителя u. Так как $s \notin p\mathbb{Z}_{(p)}$, ни один из этих простых не равен p. Для каждого q_j из условия теоремы имеем

$$M \nsubseteq R_c(q_j), \qquad M \nsubseteq R_d(q_j).$$

По лемме о дробном множителе получаем

$$\rho_{q_j} x \in K(M), \qquad v_{q_j}(\rho_{q_j}) < 0.$$

Теперь применяем лемму о переходе от масштабированного сумматора к точному. Получаем $x_1 + x_2 \in K(M)$. Следовательно, вспоминая доказательство, что $K(\{x_1 + x_2,\ 1\}) = L_p$:

$$L_p^0 \subseteq K(M).$$

Обратное включение очевидно, поскольку класс L_p^0 замкнут относительно операций композиции.

Следовательно,

$$K(M) = L_p^0.$$

□

Так как все классы из J_p^0 являются собственными несовпадающими замкнутыми подклассами (лемма 3), обратное условие также верно: из полноты некоторой системы M следует ее невключение во все классы из J_p^0. Следовательно, J_p^0 является критериальной системой предполных классов.

3.2. Неоднородный вариант

Положим

$$J_p = \{T_a \mid a \in E_p\} \cup \{V_o, V_1, I\} \cup \{R_c(\pi), R_d(\pi) \mid \pi \neq p \text{ - простое}\}.$$

Теорема 3 (Теорема полноты для линейных p-адических автоматов)**.** *Пусть $M \subseteq L_p$. Если для каждого класса $\Theta \in J_p$ выполнено $M \not\subseteq \Theta$, то*

$$K(M) = L_p.$$

Доказательство. Для функции

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0$$

обозначим через f^0 её однородную часть. Положим

$$M^0 = \{f^0 \mid f \in M\} \subseteq L_p^0.$$

Из условия $M \not\subseteq \theta$ для каждого $\theta \in J_p$ следует, что $M^0 \not\subseteq \theta_0$ для каждого $\theta_0 \in J_p^0$. Следовательно, выполнены условия теоремы о полноте в однородном случае и

$$K(M^0) = L_p^0.$$

Для любой однородной функции $h \in L_p^0$ существует функция $\widetilde{h} \in K(M)$ такая, что $\widetilde{h}^0 = h$. Иначе говоря, $\widetilde{h}$ имеет вид $\widetilde{h} = h + c_h$ с некоторым $c_h \in \mathbb{Z}_{(p)}$.

В частности, множество констант, выразимых из M, непусто. Обозначим его через

$$C = \{c \in \mathbb{Z}_{(p)} \mid c \text{ как константная функция принадлежит } K(M)\}.$$

Непустота C следует из только что доказанного утверждения при $h = 0$.

Покажем, что

$$C = \mathbb{Z}_{(p)}.$$

По доказанному утверждению, применённому к однородной функции $h(x_1, x_2) = x_1 - x_2$, найдётся функция

$$D(x_1, x_2) = x_1 - x_2 + d \in K(M)$$

с некоторым $d \in \mathbb{Z}_{(p)}$. Выберем любую константу $c \in C$. Тогда $D(c, c) = d$, откуда $d \in C$.

Рассмотрим множество

$$G = C - d = \{c - d \mid c \in C\}.$$

Докажем, что G является идеалом кольца $\mathbb{Z}_{(p)}$.

Так как $d \in C$, то $0 = d - d \in G$. Пусть

$$u = c_1 - d, \qquad v = c_2 - d,$$

где $c_1, c_2 \in C$. Тогда

$$D(c_1, c_2) = c_1 - c_2 + d \in C.$$

Следовательно,

$$u - v = c_1 - c_2 = (c_1 - c_2 + d) - d \in G.$$

Значит, G является аддитивной подгруппой кольца $\mathbb{Z}_{(p)}$.

Теперь пусть $r \in \mathbb{Z}_{(p)}$. По доказанному выше утверждению, примененному к однородной функции rx, найдется функция

$$A_r(x) = rx + a_r \in K(M)$$

с некоторым $a_r \in \mathbb{Z}_{(p)}$. Если $c \in C$, то

$$A_r(c) \in C, \qquad A_r(d) \in C.$$

Поскольку разность любых двух элементов из C принадлежит G, имеем

$$A_r(c) - A_r(d) = r(c - d) \in G.$$

Следовательно,

$$rG \subseteq G \qquad \text{для всех } r \in \mathbb{Z}_{(p)}.$$

Итак, G является идеалом кольца $\mathbb{Z}_{(p)}$.

Предположим, что $G \subseteq p\mathbb{Z}_{(p)}$. Тогда все элементы множества $C = d + G$ имеют один и тот же остаток по модулю p. Обозначим этот остаток через $a \in \mathbb{F}_p$. Тогда $C \subseteq a + p\mathbb{Z}_{(p)}$.

Возьмем произвольную функцию $f \in M$ и произвольную константу $c \in C$. Так как $f \in M \subseteq K(M)$ и $c \in C$, то после подстановки c во все переменные получаем константу $f(c, \ldots, c) \in C$. Следовательно, $c \equiv a \pmod p$ и $f(c, \ldots, c) \equiv a \pmod p$. Но $f(c, \ldots, c) = r_0 + c\sum_{i=1}^{n} r_i$. Поэтому $r_0 + a\sum_{i=1}^{n} r_i \equiv a \pmod p$, то есть $r_0 + a\,(\sum_{i=1}^{n} r_i - 1) \equiv 0 \pmod p$. Это в точности означает, что $f \in T_a$. Так как $f \in M$ было выбрано произвольно, получаем $M \subseteq T_a$, что противоречит условию теоремы.

Следовательно,

$$G \not\subseteq p\mathbb{Z}_{(p)}.$$

Значит, G содержит элемент $g \notin p\mathbb{Z}_{(p)}$. Такой элемент обратим в кольце $\mathbb{Z}_{(p)}$. Поскольку G является идеалом, содержащим обратимый элемент, получаем

$$G = \mathbb{Z}_{(p)}.$$

Отсюда

$$C = d + G = \mathbb{Z}_{(p)}.$$

Таким образом, все константы из $\mathbb{Z}_{(p)}$ принадлежат $K(M)$. В частности,

$$1 \in K(M).$$

Осталось получить сумматор $x_1 + x_2$. По доказанному выше утверждению, примененному к однородной функции

$$x_1 + x_2 + x_3,$$

существует функция

$$S(x_1, x_2, x_3) = x_1 + x_2 + x_3 + s \in K(M)$$

с некоторым $s \in \mathbb{Z}_{(p)}$. Так как уже доказано, что все константы принадлежат $K(M)$, имеем

$$-s \in K(M).$$

Подставляя константу $-s$ вместо переменной x_3, получаем

$$S(x_1, x_2, -s) = x_1 + x_2.$$

Следовательно,

$$x_1 + x_2 \in K(M).$$

Итак,

$$x_1 + x_2 \in K(M), \qquad 1 \in K(M).$$

По лемме о порождении класса L_p сумматором и константой 1 имеем окончательно

$$K(M) = L_p.$$

□

Так как все классы из J_p являются собственными несовпадающими замкнутыми подклассами (лемма 3), обратное условие также верно: из полноты некоторой системы M следует ее невключение во все классы из J_p. Следовательно, J_p является критериальной системой предполных классов.

4. Континуальная цепочка замкнутых классов I_α

В этом разделе будет построена нетривиальная цепочка вложенных замкнутых подклассов L_p, имеющая мощность континуума.

Для удобства введем следующее обозначение: для автомата

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0$$

положим

$$\|f\| = \sum_{i=1}^{n} |r_i|.$$

Для каждого $\alpha \in (0, 1]$ определим класс

$$I_\alpha = \{f \in L_p \mid \|f\| \leq \alpha\}.$$

Отметим, что свободный член r_0 в определении класса I_α не участвует. Очевидно, что I_1 совпадает с ранее определённым классом I.

Теорема 4. *Для любого $\alpha \in (0, 1]$ класс I_α является K-замкнутым:*

$$K(I_\alpha) = I_\alpha.$$

Доказательство. Достаточно проверить, что класс I_α сохраняется при всех операциях композиции.

Пусть

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} r_i x_i + r_0, \qquad g(y_1, \ldots, y_m) = \sum_{j=1}^{m} s_j y_j + s_0$$

и $f, g \in I_\alpha$.

Переименование переменных не меняет коэффициенты при переменных, поэтому не меняет величину $\|f\|$.

Рассмотрим отождествление переменных. Если в автомате f отождествляются переменные x_{n-1} и x_n, то новый коэффициент при полученной переменной равен $r_{n-1} + r_n$. Поэтому по неравенству треугольника имеем

$$\sum_{i=1}^{n-2} |r_i| + |r_{n-1} + r_n| \leq \sum_{i=1}^{n} |r_i| = \|f\| \leq \alpha.$$

Следовательно, результат отождествления снова принадлежит I_α.

Рассмотрим подстановку. Подставим f вместо переменной y_m автомата g. Тогда коэффициенты при переменных автомата f умножаются на

s_m, а остальные коэффициенты автомата g сохраняются. Поэтому для результата h имеем

$$\|h\| = \sum_{j=1}^{m-1} |s_j| + |s_m| \sum_{i=1}^{n} |r_i| = \|g\| - |s_m| + |s_m|\|f\|.$$

Так как $f \in I_\alpha$ и $\alpha \le 1$, то

$$\|f\| \le \alpha \le 1.$$

Следовательно,

$$\|h\| = \|g\| - |s_m| + |s_m|\|f\| \le \|g\| \le \alpha.$$

Значит, результат подстановки также лежит в I_α.

Осталось проверить обратную связь. Пусть обратная связь применяется к переменной x_n автомата f. Результат обратной связи имеет вид

$$f_{\text{fb}}(x_1, \ldots, x_{n-1}) = \sum_{i=1}^{n-1} \frac{r_i}{1 - r_n} x_i + \frac{r_0}{1 - r_n}.$$

Положим

$$A = \sum_{i=1}^{n-1} |r_i|, \qquad B = |r_n| \qquad \Longrightarrow \; A + B \le \alpha \le 1.$$

Поскольку обратная связь применима к x_n, имеем

$$r_n \in p\mathbb{Z}_{(p)}.$$

Если бы $B = |r_n| = 1$, то рациональное число r_n было бы равно 1 или -1, что невозможно для элемента $p\mathbb{Z}_{(p)}$. Поэтому $B < 1$, и

$$|1 - r_n| \ge 1 - |r_n| = 1 - B > 0.$$

Поэтому

$$\|f_{\text{fb}}\| = \sum_{i=1}^{n-1} \left| \frac{r_i}{1 - r_n} \right| \le \frac{A}{1 - B}.$$

Из $A + B \le \alpha$ получаем $A \le \alpha - B$. Значит,

$$\|f_{\text{fb}}\| \le \frac{\alpha - B}{1 - B}.$$

Так как $\alpha \leq 1$, выполнено

$$\frac{\alpha - B}{1 - B} \leq \alpha.$$

Следовательно,

$$\|f_{\text{fb}}\| \leq \alpha,$$

что и требовалось.

Итак, класс I_α сохраняется при переименовании переменных, отождествлении переменных, подстановке и обратной связи. Поэтому $K(I_\alpha) \subseteq I_\alpha$. Обратное же включение следует из определения замыкания. Следовательно,

$$K(I_\alpha) = I_\alpha.$$

Теорема доказана. □

Теорема 5. *Если* $0 < \alpha < \beta \leq 1$, *то*

$$I_\alpha \subsetneq I_\beta.$$

Доказательство. Нестрогое включение непосредственно следует из определения. Если $f \in I_\alpha$, то $\|f\| \leq \alpha < \beta$, следовательно, $\|f\| \leq \beta$, и потому $f \in I_\beta$. Значит,

$$I_\alpha \subseteq I_\beta.$$

Докажем строгость включения. Выберем натуральное число N, не делящееся на p, настолько большим, что

$$\frac{1}{N} < \beta - \alpha.$$

Тогда для некоторого $m \in \mathbb{Z}$ выполнено

$$\alpha < \frac{m}{N} < \beta.$$

Положим $r = m/N$. Поскольку $p \nmid N$, имеем $r \in \mathbb{Z}_{(p)}$.

Рассмотрим функцию

$$f_r(x) = rx.$$

Тогда

$$\|f_r\| = |r| = r,$$

поэтому $f_r \in I_\beta \setminus I_\alpha$. Следовательно,

$$I_\alpha \subsetneq I_\beta.$$

Теорема доказана. □

Следствие 1. *Семейство*

$$\{I_\alpha \mid \alpha \in (0,1]\}$$

образует строго возрастающую цепочку K-замкнутых классов:

$$\alpha < \beta \quad \Longrightarrow \quad I_\alpha \subsetneq I_\beta.$$

В частности, множество попарно различных классов этой цепочки имеет мощность континуума.

Заключение

В работе исследована проблема K-полноты в классе линейных p-адических автоматов для нечётного простого числа p. Получены критерии полноты для однородных и неоднородных линейных p-адических функций. Полученные результаты обобщают известные свойства линейных 2-адических автоматов на случай произвольного нечётного простого основания. Кроме того, построена континуальная строго возрастающая цепочка K-замкнутых классов I_α, $\alpha \in (0,1]$.

Автор выражает благодарность профессору А. А. Часовских за помощь в постановке задачи и содействие в подготовке статьи.

Список литературы

[1] А. А. Часовских, “О полноте в классе линейных 2-адических автоматов”, *Интеллектуальные системы. Теория и приложения*, **20**:4 (2016), 209–227.

[2] А. А. Часовских, “Об алгоритмической разрешимости проблемы полноты для линейных автоматов”, *Вестн. Моск. ун-та. Сер. 1. Матем., мех.*, 1986, № 3, 82–84.

[3] А. А. Часовских, “Проблема полноты для класса линейно-автоматных функций”, *Дискрет. матем.*, **27**:2 (2015), 134–151. DOI: 10.4213/dm1330; *Discrete Math. Appl.*, **26**:2 (2016), 89–104. DOI: 10.1515/dma-2016-0007.

[4] А. А. Часовских, “Проблема полноты в классах линейных автоматов”, *Интеллектуальные системы. Теория и приложения*, **22**:2 (2018), 151–153.

[5] А. Г. Лунц, “Конечные p-адические автоматы”, *Докл. АН СССР*, **150**:4 (1963), 755–758.

[6] Г. Дэвенпорт, “О сложении классов вычетов”, *УМН*, 1940, № 7, 90–92.

Статья поступила 28 августа 2026 г.

The completeness problem in the class of linear p-adic automata

M. E. Kalashnikov

The K-completeness problem in the class of linear p-adic automata is studied for an odd prime p. Completeness criteria are obtained for homogeneous and inhomogeneous linear p-adic functions. The results generalize known results for linear 2-adic automata to arbitrary odd prime bases. In addition, a strictly increasing chain of K-closed classes I_α, $\alpha \in (0, 1]$, of continuum cardinality is constructed.

Keywords: finite automaton, p-adic number, linear p-adic automaton, composition operations, feedback, completeness problem, closed class.

References

[1] A. A. Chasovskikh, "Completeness in the class of linear 2-adic automata", *Intelligent Systems. Theory and Applications*, **20**:4 (2016), 209–227 (In Russian).

[2] A. A. Chasovskikh, "On the algorithmic decidability of the completeness problem for linear automata", *Vestn. Mosk. Univ. Ser. 1. Mat. Mekh.*, 1986, № 3, 82–84 (In Russian).

[3] A. A. Chasovskikh, "Completeness problem for the class of linear automata functions", *Diskr. Mat.*, **27**:2 (2015), 134–151. DOI: 10.4213/dm1330; *Discrete Math. Appl.*, **26**:2 (2016), 89–104. DOI: 10.1515/dma-2016-0007 (In Russian).

[4] A. A. Chasovskikh, "The problem of completeness in the class of linear automata", *Intelligent Systems. Theory and Applications*, **22**:2 (2018), 151–153 (In Russian).

[5] A. G. Lunts, "Finite p-adic automata", *Dokl. Akad. Nauk SSSR*, **150**:4 (1963), 755–758 (In Russian).

[6] H. Davenport, "On the addition of residue classes", *Uspekhi Mat. Nauk*, 1940, № 7, 90–92 (In Russian).

Received on August 28, 2026

Оценки характеристик кодов Таннера на графе проективного пространства

А. А. Юсуфова*, Г. В. Калачев‡

В работе рассматриваются коды Таннера, построенные на графе инцидентности точек и гиперплоскостей проективного пространства, с проективными кодами Рида-Маллера в качестве компонентных кодов. Сопоставление символов локальных кодов рёбрам производится таким образом, чтобы получаемый код Таннера обладал симметрией проективного пространства. В работе получены нижние оценки минимального расстояния и размерности таких кодов. В частности, показано, что при определённых значениях параметров эти коды являются асимптотически хорошими.

Ключевые слова: коды Таннера, граф инцидентности, проективное пространство, минимальное расстояние, размерность кода.

1. Введение

В данной работе изучаются коды Таннера, построенные на графе инцидентности точек и гиперплоскостей проективного пространства $\mathbb{P}_q^D$ над конечным полем $\mathbb{F}_q$. В качестве локальных кодов используются проективные коды Рида–Маллера, а сопоставление символов локальных кодов рёбрам графа производится таким образом, чтобы получаемый код обладал симметрией проективного пространства: группа изометрий кода включает в себя группу проективных преобразований $\mathrm{PGL}(D{+}1, q)$. Такая симметрия важна для потенциальных применений этих кодов в качестве локальных кодов в симплициальных комплексах с соответствующими локальными симметриями.

Одна из основных мотиваций для изучения таких кодов связана с задачей построения локально тестируемых и квантовых LDPC кодов

Юсуфова Алина Алмасовна — студент магистратуры, филиал МГУ в г. Ташкенте, e-mail: yusufova.alinochka@mail.ru, ORCID: 0009-0007-4090-8215.

Yusufova Alina Almasovna — Master's student, Lomonosov Moscow State University branch in Tashkent.

‡*Калачев Глеб Вячеславович* — канд. физ.-мат. наук, науч. сотр. каф. математической теории интеллектуальных систем мех.-мат. ф-та МГУ, e-mail: gleb.kalachev@yandex.ru, ORCID: 0000-0003-2695-3179.

Kalachev Gleb Vyacheslavovich — Candidate of Physical and Mathematical Sciences, Researcher, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of Mathematical Theory of Intellectual Systems.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

на основе симплициальных комплексов Рамануджана типа $\widetilde{A}_d$ [1]. Графы инцидентности проективных пространств возникают как локальные структуры (линки) в таких комплексах, а коды Таннера, определённые на линках, играют ключевую роль при определении глобального кода на комплексе. Для построения хорошего глобального кода необходимо, чтобы локальные коды Таннера обладали необходимой симметрией, достаточной размерностью, минимальным расстоянием, а также свойством тестируемости на согласованность (agreement testability). Симплициальные комплексы Рамануджана потенциально могли бы дать семейства хороших локально тестируемых или квантовых LDPC кодов с высокой симметрией, чего пока не удаётся достичь с использованием конструкций, основанных на произведениях кодов (например, [2]). В данной работе мы фокусируемся на исследовании размерности и минимального расстояния.

Двумерный случай ($D = 2$), то есть конструкции на графах инцидентности аффинных и проективных плоскостей, изучен достаточно подробно. Оценки размерности кодов Таннера на таких графах получены в [3–5], а оценки минимального расстояния — в [3, 6, 7]. В [8] для кодов на аффинной плоскости, помимо расстояния и размерности, исследуется также свойство тестируемости на согласованность, важное для построения локально тестируемых кодов. Многомерный случай изучался только для локальных кодов степени один [9]. Настоящая работа рассматривает одно из естественных обобщений: проективные пространства произвольной размерности с локальными кодами более высокой степени.

Основные результаты работы состоят в следующем. Для кода Таннера на графе инцидентности проективного пространства с произвольными локальными кодами с достаточно большим минимальным расстоянием получена нижняя оценка минимального расстояния глобального кода, основанная на спектральном расширении графа и оценке Сипсера–Спилмана. Для кодов с проективными кодами Рида–Маллера в качестве локальных кодов получена нижняя оценка размерности, а также нижние оценки минимального расстояния как в случае проективной плоскости, так и в больших размерностях. Из полученных оценок следует, что при определённых соотношениях между параметрами эти коды являются асимптотически хорошими. Также проверено, что предложенная конструкция действительно наследует симметрию проективного пространства.

Статья организована следующим образом. В разделе 2 приведены основные определения и обозначения, относящиеся к кодам Таннера, построенным на графе инцидентности проективного пространства, и формулируются основные результаты работы. Раздел 3 содержит вспомогательные сведения о кодах Рида–Маллера, графах-расширителях и спектре графа инцидентности, необходимые для дальнейших доказа-

тельств. В разделе 4 приведены доказательства оценок минимального расстояния и размерности рассматриваемых кодов, а также исследуется их инвариантность относительно действия групп симметрий. В разделе 5 сформулированы основные выводы. В приложении А приведены результаты численного эксперимента, посвящённого вычислению размерности кода в двумерном случае.

2. Основные определения и результаты

2.1. Линейные коды

Обозначим через $\mathbb{F}_q^n$ — n-мерное векторное пространство над $\mathbb{F}_q$. Ниже приведём основные определения (см., например, [10, стр. 198]).

Линейным кодом длины n и размерности k (сокращённо $[n,k]$-кодом) называется произвольное линейное k-мерное подпространство n-мерного векторного пространства $\mathbb{F}_q^n$. Элементы линейного кода $\mathcal{C} \subseteq \mathbb{F}_q^n$ называются *кодовыми словами*. Отношение $R = k/n$ линейного кода $\mathcal{C} \subseteq \mathbb{F}_q^n$ называется *скоростью кода (code rate)*. *Вес вектора* $v \in \mathbb{F}_q^n$ — количество ненулевых элементов вектора. Вес будем обозначать через $|v|$. *Расстояние Хэмминга* между векторами $x, y \in \mathbb{F}_q^n$ определим как

$$d(x,y) = |x - y|,$$

то есть как число позиций, в которых x и y не совпадают. Минимальное расстояние между различными кодовыми словами кода $\mathcal{C} \subseteq \mathbb{F}_q^n$ называется *расстоянием кода*. Обозначим его через $d_{\min}(\mathcal{C})$. Иногда удобнее использовать *относительное минимальное расстояние* $\delta_{\min}(\mathcal{C}) := d_{\min}(\mathcal{C})/n$. Легко показать, что в линейном коде $\mathcal{C}$ кодовое расстояние равно весу его минимального ненулевого элемента:

$$d_{\min}(\mathcal{C}) = \min_{c \in \mathcal{C} \setminus \{0\}} |c|.$$

2.2. Коды Таннера

Нас будет интересовать класс линейных кодов, которые определяются на графах через набор локальных проверок. В дальнейшем будем следовать изложению определения таких кодов из работы [11].

Пусть A и B — произвольные множества. Через A^B понимаются множества всех функций (отображений), определённых на B и принимающих значения в A.

Определение 1 (Код Таннера)**.** Пусть $G = (V, E)$ — w-регулярный граф, в котором все рёбра, выходящие из вершины v, занумерованы от 1 до w, обозначим их $e_{v,1}, \ldots, e_{v,w}$. Пусть также есть набор линейных кодов $\mathcal{C} = (\mathcal{C}_v)_{v \in V}$, $\mathcal{C}_v \subseteq \mathbb{F}_q^w$. *Код Таннера* $\mathcal{T}(G, \mathcal{C})$ — это линейный код, состоящий из всех кодовых слов $c \in \mathbb{F}_q^E$, таких что $c_v \in \mathcal{C}_v$ для всех вершин $v \in V$, где $c_v = (c(e_{v,1}), \ldots, c(e_{v,w}))$.

2.2.1. Граф инцидентности проективного пространства

Мы будем исследовать коды Таннера на двудольном графе $\Gamma_{D,q}$ инцидентности прямых и гиперплоскостей $(D+1)$-мерного пространства $\mathbb{F}_q^{D+1}$ (который можно также интерпретировать как граф инцидентности точек и гиперплоскостей D-мерного проективного пространства над полем $\mathbb{F}_q$). Определим этот граф в том виде, как нам будет удобно его использовать, приводя заодно некоторые вспомогательные определения из [12, стр. 9]. Напомним, что *степенью вершины* v в графе G называется число инцидентных ей рёбер. Граф G называется *регулярным (или однородным)* графом, если степени всех его вершин равны.

Нам будет удобно задавать двудольный граф тройкой $G = (V_0, V_1, E)$, где V_0, V_1 — множества вершин первой и второй доли, $E \subseteq V_0 \times V_1$ — множество рёбер.

Обозначим $\mathsf{Gr}_q(D, k) := \{V \subseteq \mathbb{F}_q^D \mid \dim V = k\}$ — множество всех k-мерных подпространств D-мерного пространства над полем $\mathbb{F}_q$.

Тогда $\Gamma_{D,q} = (\mathsf{Gr}_q(D+1, 1), \mathsf{Gr}_q(D+1, D), E)$, где

$$E = \{(\ell, h) \in \mathsf{Gr}_q(D+1, 1) \times \mathsf{Gr}_q(D+1, D) \mid \ell \subseteq h\}.$$

Определим скалярное произведение $\langle \alpha, \beta \rangle := \sum_{i=0}^{D} \alpha_i \beta_i$ векторов $\alpha, \beta \in \mathbb{F}_q^{D+1}$. Каждую прямую $\ell \in \mathsf{Gr}_q(D+1, 1)$ можно однозначно задать ненулевым вектором (*представителем*) $\hat{\ell} \in \mathbb{F}_q^{D+1}$ на этой прямой, а каждую гиперплоскость $h \in \mathsf{Gr}_q(D+1, D)$ можно задать ненулевым вектором (*представителем*) $\hat{h}$, ортогональным данной гиперплоскости. Поскольку представители $\hat{\ell}$ и $\hat{h}$ определены с точностью до умножения на константу, для определённости потребуем, чтобы у представителей первая ненулевая координата была равна 1.

Представители прямых и гиперплоскостей можно естественным образом взаимно-однозначно отождествить с точками проективного пространства

$$\mathbb{P}_q^D = (\mathbb{F}_q^{D+1} \setminus \{0\}) / \sim,$$

где $x \sim y$, если $x = \gamma y$ для некоторого $\gamma \in \mathbb{F}_q \setminus \{0\}$ для $x, y \in \mathbb{F}_q^{D+1} \setminus \{0\}$. Каждой точке $x \in \mathbb{P}_q^D$ сопоставляется вектор длины $D+1$ её однородных координат.

Теперь мы можем ввести чуть более удобное определение $\Gamma_{D,q} = (\mathbb{P}_q^D, \mathbb{P}_q^D, E)$, где

$$E = \{(x, y) \in \mathbb{P}_q^D \times \mathbb{P}_q^D \mid \langle x, y \rangle = 0\}.$$

2.2.2. Код на графе проективного пространства

Введём ещё несколько необходимых определений и обозначений. Пусть $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ — полином, тогда его *степень* $\deg_{x_{i_1}, x_{i_2}, \ldots, x_{i_m}} f$ *по переменным* $x_{i_1}, x_{i_2}, \ldots, x_{i_m}$ определим как максимальную степень его монома по этим переменным, где степень монома определяется как

$$\deg_{x_{i_1}, \ldots, x_{i_m}} \left(c x_1^{d_1} x_2^{d_2} \cdots x_n^{d_n} \right) = d_{i_1} + d_{i_2} + \cdots + d_{i_m}.$$

В частности, *степенью* полинома f называется степень по всем его переменным:

$$\deg f = \deg_{x_1, \ldots, x_n} f.$$

Полином называется *однородным по переменным* $x_{i_1}, \ldots, x_{i_m}$, если для любого монома, входящего в данный полином, его степень по переменным $x_{i_1}, \ldots, x_{i_m}$ равна одной и той же величине d. Полином называется *однородным степени* d, если степень всех его мономов равна d.

Введём несколько обозначений.

1. $\mathbb{F}_q[x_1, \ldots, x_n]$ — кольцо полиномов от переменных $x_1, \ldots, x_n$ над полем $\mathbb{F}_q$;
2. $\mathbb{F}_q[x_1, \ldots, x_n]_{\leq \nu}$ — множество полиномов степени не выше ν над полем $\mathbb{F}_q$;
3. $\mathbb{F}_q[x_1, \ldots, x_n]_{\nu}$ — множество однородных полиномов степени ν над полем $\mathbb{F}_q$, включающее в себя нулевой полином.

Зафиксируем $D \geq 2$, $m < (q-1)D$. Определим код $\mathcal{C}_{D,q,m}$ как множество функций $c : E(\Gamma_{D,q}) \to \mathbb{F}_q$ таких, что для любого $\gamma \in \mathbb{P}_q^D$ существуют однородные полиномы f_γ, $f^\gamma \in \mathbb{F}_q[x_0, \ldots, x_D]_m$ такие, что $c(\alpha, \beta) = f_\alpha(\beta) = f^\beta(\alpha)$ для всех рёбер $(\alpha, \beta) \in E(\Gamma_{D,q})$.

Замечание. Код $\mathcal{C}_{D,q,m}$ является кодом Таннера на графе $\Gamma_{D,q}$, в котором для различных вершин могут использоваться различные локальные коды.

2.3. Полученные результаты

Теорема 1. *Пусть для каждой вершины $v \in V$ графа $\Gamma_{D,q}$ выбран локальный код $\mathcal{C}_v \subseteq \mathbb{F}_q^w$, $w = \frac{q^D-1}{q-1}$. Обозначим за $\mathcal{C} = \{\mathcal{C}_v\}_{v\in V}$ — семейство таких кодов, где длина каждого кода составляет w и расстояние не меньше εw. Тогда относительное минимальное расстояние кода $\mathcal{T} = \mathcal{T}(\Gamma_{D,q}, \mathcal{C})$ удовлетворяет оценке:*

$$\delta_{\min}(\mathcal{T}) \geq \varepsilon \left(\varepsilon - \sqrt{q^{1-D}}\right).$$

Теорема 2. *Пусть $0 \leq m < q$. Тогда $\delta_{\min}(\mathcal{C}_{2,q,m}) \geq \varepsilon^3$, где $\varepsilon = \frac{q-m}{q}$.*

Теорема 3. *Пусть $m < q$, $D \geq 3$. Тогда*

$$\delta_{\min}(\mathcal{C}_{D,q,m}) \geq \frac{(q-m)^2(q-2)}{q^3}.$$

Теорема 4. *Пусть $1 \leq m < q/2$. Тогда*

$$\dim \mathcal{C}_{D,q,m} \geq \frac{2m+D}{D}\binom{m+D-1}{D-1}^2.$$

Таким образом, для случая проективной плоскости ($D = 2$) получаем следствие

Следствие 1. *Пусть $1 \leq m < q/2$. Тогда*

$$\dim \mathcal{C}_{2,q,m} \geq (m+1)^3. \tag{1}$$

Отметим, что аналогичные результаты для кодов на графах аффинных плоскостей получены в работах [3–5,8]. В [5] также без доказательства упомянуто, что оценки для аффинного случая переносятся на проективный, однако это не всегда выполняется и требует дополнительного исследования[1]. Более того, на основании результатов численного эксперимента, в котором размерность кода $\mathcal{C}_{2,q,m}$ оценивалась по рангу его проверочной матрицы, можно также выдвинуть следующее предположение:

Гипотеза 1. *Пусть $q/2 \leq m < q$. Тогда*

$$\dim \mathcal{C}_{2,q,m} \geq (2m+1-q)(q^2+q+1) + (q-m)^3.$$

[1] в частности, при $q = 9, m = 3$ в аффинном случае размерность 66 [4], а в проективном 64 (см. Приложение А)

Заметим, что в случае $q = 9, m = 4$ размерность равна 131, что выше оценки (1). В [3, 4] детально исследован аффинный случай и показано, что для кодов на графе аффинной плоскости для $q = 9, k = 4$ (в наших обозначениях это соответствует $m = 3$): размерность равна 66, что выше общей нижней оценки $k^3 = 64$.

Для линейных кодов $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathbb{F}_q^n$ линейное отображение $\varphi : \mathcal{C}_1 \to \mathcal{C}_2$ называется *изометрией* кодов $\mathcal{C}_1$ и $\mathcal{C}_2$, если $\varphi(\mathcal{C}_1) = \mathcal{C}_2$ и φ сохраняет вес Хэмминга, то есть $|\varphi(x)| = |x|$ для любого $x \in \mathcal{C}_1$. Коды $\mathcal{C}_1$ и $\mathcal{C}_2$ называются *эквивалентными*, если между ними существует изометрия. *Мономиальным преобразованием* пространства $\mathbb{F}_q^n$ называется композиция перестановки координат и умножения каждой координаты на ненулевой элемент поля. Любое мономиальное преобразование является изометрией кодов.

Следующее утверждение показывает, что код $\mathcal{C}_{D,q,m}$ обладает определённой симметрией.

Для любого вектора $\alpha = (\alpha_0, \ldots, \alpha_D) \in \mathbb{F}_q^{D+1} \setminus \{0\}$ обозначим через $\hat{\alpha}$ нормированный вектор, где первая ненулевая компонента равна 1. Здесь

$$\hat{\alpha} = \frac{\alpha}{\alpha_i} = \Big(0, \ldots, 0, \underbrace{1}_{i}, \tfrac{\alpha_{i+1}}{\alpha_i}, \ldots, \tfrac{\alpha_D}{\alpha_i}\Big),$$

где $i = \min\{k \in \{0, \ldots, D\} \mid \alpha_k \neq 0\}$. Таким образом, $\hat{\alpha}$ — однородные координаты точки проективного пространства $\mathbb{P}_q^D$, соответствующей прямой, проходящей через точку α.

Утверждение 1. *Пусть $D \geq 2$, $1 \leq m < (q-1)D$, $E = E(\Gamma_{D,q})$, $\mathcal{C} = \mathcal{C}_{D,q,m}$. Для любой матрицы $A \in \mathrm{GL}(D+1, q)$ существует набор ненулевых констант $a = (a_{(\alpha,\beta)})_{(\alpha,\beta)\in E} \in \mathbb{F}_q^E$ такой, что для любого кодового слова $c \in \mathcal{C}$ существует кодовое слово $c' \in \mathcal{C}$, удовлетворяющее условию:*

$$c'(\alpha, \beta) = a_{(\alpha,\beta)} c\big(\widehat{A\alpha}, \widehat{A^*\beta}\big) \qquad \text{для всех } (\alpha, \beta) \in E,$$

где $A^* = (A^T)^{-1}$.

Из данного утверждения, в частности, следует, что группа изометрий кода $\mathcal{C}_{D,q,m}$ включает в себя группу проективных преобразований $\mathrm{PGL}(D+1, q)$.

3. Вспомогательные результаты

3.1. Коды Рида-Маллера

Для доказательства основных результатов нам понадобятся известные результаты о размерности и минимальном расстоянии недвоичных ко-

дов Рида-Маллера. Пусть $\mathbb{F}_q^D = \{\alpha_1, \ldots, \alpha_{q^D}\}$. Кодом Рида-Маллера порядка ν длины q^D над полем $\mathbb{F}_q$ называется множество

$$\mathrm{RM}_q(\nu, D) := \left\{ (p(\alpha_i)_{i=1}^{q^D}) \;\middle|\; p \in \mathbb{F}_q[x_1, \ldots, x_D]_{\leq \nu} \right\}.$$

Известно [13, стр. 1568], что при $\nu < q$ коды Рида-Маллера имеют следующие параметры:

$$\dim \mathrm{RM}_q(\nu, D) = \binom{\nu + D}{D},$$
$$d_{\min}(\mathrm{RM}_q(\nu, D)) = (q - \nu) q^{D-1}.$$

Теперь определим проективные коды Рида-Маллера. Пусть $\mathbb{P}_q^D = (P_1, \ldots, P_{n_D})$, где $n_D := |\mathbb{P}_q^D| = \frac{q^{D+1}-1}{q-1}$, P_i — однородные координаты i-й точки проективного пространства. Проективным кодом Рида-Маллера порядка ν длины n_D называется код

$$\mathrm{PRM}_q(\nu, D) := \left\{ \big(p(P_i)\big)_{i=1}^{n_D} \;\middle|\; p \in \mathbb{F}_q[x_0, x_1, \ldots, x_D]_\nu \right\}$$

Известно [13, Раздел III, Теорема 1, стр. 1568–1569], что при $1 \leq \nu < q$ проективные коды Рида-Маллера имеют следующие параметры:

$$\dim \mathrm{PRM}_q(\nu, D) = \binom{\nu + D}{D}, \tag{2}$$
$$d_{\min}(\mathrm{PRM}_q(\nu, D)) = (q - \nu + 1) q^{D-1}. \tag{3}$$

При этом при $\nu = 0$ код содержит только полиномы нулевой степени, то есть константы, поэтому $\mathrm{PRM}_q(0, D)$ — это просто код с повторением длины n_D.

При $D = 1$ коды Рида-Маллера соответствуют расширенным кодам Рида-Соломона длины q, где значение полинома дополнительно вычисляется в точке 0. Проективные коды Рида-Маллера при $D = 1$ в точности соответствуют проективным кодам Рида-Соломона, где значение полинома дополнительно вычисляется в 0 и в бесконечно удалённой точке (старший коэффициент полинома), т.е.

$$\mathrm{PRS}(q, k) = \mathrm{PRM}_q(k - 1, 1).$$

В частности,

$$\dim \mathrm{PRS}(q, k) = k,$$
$$d_{\min}(\mathrm{PRS}(q, k)) = q - k + 2.$$

Как и обычные коды Рида-Соломона, проективные коды Рида-Соломона являются MDS кодами, то есть для них достигается оценка Синглтона.

3.2. Графы–расширители (expander graphs)

Приведём определения разных видов расширения графов (см. [12, стр. 9], [14, стр. 7]).

Пусть дан регулярный граф G степени d с n вершинами, все вершины которого пронумерованы от 1 до n. *Матрицей смежности* $A(G)$ порядка $n \times n$ графа G называется симметричная бинарная матрица вида:

$$A_{ij} = \begin{cases} 1, & \text{если вершины } i \text{ и } j \text{ смежны;} \\ 0, & \text{в противном случае.} \end{cases}$$

Все собственные значения матрицы $A(G)$ упорядочим в виде:

$$\lambda_1 = d \ \geq \ \lambda_2 \geq \ldots \ \geq \ \lambda_{n-1} \ \geq \ \lambda_n.$$

Второе по абсолютной величине собственное значение $\max(|\lambda_2|, |\lambda_n|)$ характеризует спектральное расширение графа. Для двудольных регулярных графов $\lambda_n = -d$, и таким образом $\lambda(G) = d$. Поэтому в этом случае используется ослабленный вариант спектрального расширения, которое характеризуется вторым собственным значением λ_2.

Пусть V — множество вершин данного графа G и пусть $S, T \subseteq V$. Обозначим:

$$E(S,T) = \{(s,t) \in E \mid s \in S, t \in T\}, \qquad E(S) := |E(S,S)|/2.$$

Заметим, что в $E(S,S)$ каждое ребро $\{u,v\}$ входит два раза: как (u,v) и как (v,u); поэтому при подсчёте числа рёбер $E(S)$, индуцированных на множестве вершин S, мы делим число пар в $E(S,S)$ пополам.

Нам потребуется следующая односторонняя версия леммы Алона-Чанга.

Лемма 1 (теорема 9.2.1 в [15])**.** *Пусть G — d-регулярный граф порядка n, второе по убыванию собственное значение которого равно λ. Пусть $X \subset V(G)$ — произвольное подмножество вершин, такое что $|X| = \gamma n$ для некоторого $\gamma \in [0,1]$. Тогда число рёбер, индуцированных на X, удовлетворяет неравенству*

$$E(X) \leq \frac{dn}{2}\left(\gamma^2 + \frac{\lambda}{d}\,\gamma\,(1-\gamma)\right).$$

3.2.1. Спектр графа инцидентности проективного пространства

Лемма 2. *Матрица смежности графа $\Gamma_{D,q}$ имеет следующие собственные значения:*

1. $\pm w$ *кратности* 1, *где* $w = \frac{q^D-1}{q-1}$ — *степень вершины;*
2. $\pm q^{\frac{D-1}{2}}$ *кратности* $n-1$, *где* $n = \frac{q^{D+1}-1}{q-1}$ — *число вершин в каждой доле.*

Доказательство. Матрица смежности графа $\Gamma_{D,q}$ имеет вид

$$A = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix},$$

где $M = (M_{\alpha,\beta})_{\alpha,\beta\in\mathbb{P}_q^D}$,

$$m_{\alpha,\beta} = \begin{cases} 1, & \langle\alpha,\beta\rangle = 0, \\ 0, & \text{иначе.} \end{cases}$$

Здесь строки и столбцы матрицы M проиндексированы точками проективного пространства $\mathbb{P}_q^D$.

Для $\alpha \in \mathbb{P}_q^D$ введём обозначение

$$\alpha^\perp := \{\beta \in \mathbb{P}_q^D \mid \langle\alpha,\beta\rangle = 0\}.$$

Заметим, что $\alpha^\perp$ представляет собой $(D-1)$-мерное проективное пространство, поэтому $|\alpha^\perp| = \frac{q^D-1}{q-1} = w$. Для различных $\alpha, \alpha' \in \mathbb{P}_q^D$ множество $\alpha^\perp \cap \alpha'^\perp$ представляет собой $(D-2)$-мерное проективное пространство. Отсюда

$$|\alpha^\perp| = \frac{q^D-1}{q-1} = w, \tag{4}$$

$$|\alpha^\perp \cap \alpha'^\perp| = \frac{q^{D-1}-1}{q-1} =: s. \tag{5}$$

Вычислим матрицу MM^T, используя (4) и (5):

$$(MM^T)_{\alpha,\alpha'} = \big|\{\beta : \langle\alpha,\beta\rangle = \langle\alpha',\beta\rangle = 0\}\big| = |\alpha^\perp \cap \alpha'^\perp| = \begin{cases} w, & \alpha = \alpha', \\ s, & \alpha \neq \alpha'. \end{cases}$$

Таким образом,

$$MM^T = sJ_n + (w-s)I_n,$$

где J_n — $n \times n$ матрица из единиц, I_n — единичная матрица $n \times n$.

Пусть $\mathbf{1}$ — вектор длины n, все компоненты которого равны 1. Легко видеть, что $J_n\mathbf{1} = n\mathbf{1}$, а также $J_n x = 0$ для любого вектора x, ортогонального $\mathbf{1}$. Таким образом, матрица J_n имеет однократное собственное значение n и $(n-1)$-кратное собственное значение 0. Значит, MM^T имеет спектр:

$$\lambda_2 = \cdots = \lambda_n = w - s = \frac{q^D - q^{D-1}}{q-1} = q^{D-1},$$
$$\lambda_1 = sn + (w-s) = \frac{(q^{D+1}-1)(q^{D-1}-1)}{(q-1)^2} + q^{D-1} = \frac{q^{2D} - 2q^D + 1}{(q-1)^2} = w^2.$$

Поскольку матрица M симметричная, то модули её собственных значений равны корню из собственных значений матрицы MM^T:

$$\lambda_1(M) = w, \quad |\lambda_2(M)| = \cdots = |\lambda_n(M)| = \sqrt{q^{D-1}}.$$

Осталось заметить, что матрица

$$A = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix}$$

имеет в спектре значения $\pm\lambda_i(M), i = 1, \ldots, n$. Лемма доказана. □

4. Доказательство основных результатов

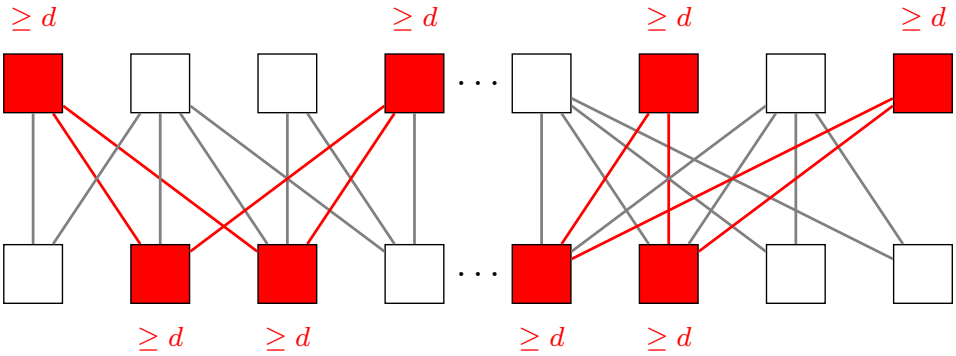
4.1. Оценка минимального расстояния

Семейство кодов назовём *асимптотически хорошим*, если минимальное расстояние пропорционально длине кода и скорость не меньше некоторого $R > 0$. В свою очередь, это означает, что для каждого кода из семейства минимальное расстояние $d_{\min}$ растёт линейно с длиной n, обычно так, что $d_{\min} \geq \alpha n$, где $\alpha > 0$ — фиксированная константа, $k \geq Rn$.
Приведём нижнюю оценку минимального расстояния, также упоминавшуюся в работах [3, 5, 7] для графовых кодов. Здесь мы приведём явное доказательство для графа проективной плоскости $\Gamma_{2,q}$.

Лемма 3. *Пусть для каждой вершины $v \in V$ графа $\Gamma_{2,q}$ выбран локальный код $\mathcal{C}_v \subseteq \mathbb{F}_q^{q+1}$. Через $\mathcal{C} = \{\mathcal{C}_v\}_{v\in V}$ обозначим семейство таких кодов с расстоянием не меньше d. Будем считать, что в каждой вершине графа $\Gamma_{2,q}$ фиксирована нумерация инцидентных рёбер. Тогда код $\mathcal{T}(\Gamma_{2,q}, \mathcal{C})$ обладает минимальным расстоянием не меньше:*

$$d_{min}\Big(\mathcal{T}\left(\Gamma_{2,q}, \mathcal{C}\right)\Big) \geq d(d(d-1)+1).$$

В частности, при выборе $\mathcal{C} \equiv \mathcal{C}_0$ для всех $v \in V$ получаем конструкцию кода Таннера вида $\mathcal{T}(\Gamma_{2,q}, \mathcal{C}_0)$.



Доказательство. Для краткости обозначим $\Gamma := \Gamma_{2,q}$, $E := E(\Gamma)$ — множество рёбер графа Γ. Рассмотрим ненулевое кодовое слово $c \in \mathcal{T}(\Gamma, \mathcal{C})$ минимального веса. Напомним, что кодовое слово кода $\mathcal{T}(\Gamma, \mathcal{C})$ — это функция, определённая на множестве рёбер графа Γ. *Активными* назовём те рёбра $e \in E$, для которых $c(e) \neq 0$, таким образом $d(\mathcal{T}(\Gamma, \mathcal{C})) = |c|$ равно числу активных рёбер. Вершину графа Γ назовём *активной*, если из неё исходит хотя бы одно активное ребро. Напомним, что каждой вершине графа соответствует точка или прямая проективной плоскости. Точку или прямую будем называть *активной*, если соответствующая вершина графа $\Gamma_{2,q}$ активна.

Так как код Таннера $\mathcal{T}(\Gamma, \mathcal{C})$ состоит из всех кодовых слов $c \in \mathbb{F}_q^E$ таких, что $c_v \in \mathcal{C}_v$ для всех вершин $v \in V$, где $c_v = (c(e_{v,1}), \ldots, c(e_{v,w}))$, то рассмотрим произвольную активную прямую, из которой исходит не менее d активных рёбер. Поскольку эта прямая инцидентна по меньшей мере d активным точкам, то на ней расположены как минимум d активных точек. В свою очередь, через каждую активную точку рассматриваемой прямой проходит ещё не менее $d-1$ активных прямых по построению. Тогда всего активных прямых будет не менее $d(d-1)+1$. Как уже упоминалось ранее, из соответствующих вершин графа исходит как минимум d активных рёбер, таким образом общее число активных рёбер не меньше $d(d(d-1)+1)$. □

Лемма 4. *Пусть G — w-регулярный граф порядка n со вторым по величине собственным значением $\lambda \geq 0$. Пусть для каждой вершины $v \in V$ графа G сопоставлен линейный код $\mathcal{C}_v$ с длиной w и минимальным*

расстоянием не меньше εw. Обозначим за $\mathcal{C} = \{\mathcal{C}_v\}_{v\in V}$ — семейство таких кодов, тогда выполнена оценка:

$$\delta_{\min}(\mathcal{T}(G,\mathcal{C})) \geq \varepsilon\left(\varepsilon - \frac{\lambda}{w}\right).$$

Эта лемма аналогична [14, Lemma 15], но мы не предполагаем, что все локальные коды одинаковые. Приведём здесь доказательство, идея которого полностью аналогична той, что использовалась в [14, Lemma 15].

Доказательство. При $\lambda/w \geq \varepsilon$ утверждение леммы тривиально. Поэтому далее рассматриваем случай $\lambda/w < \varepsilon$.

Пусть $c \in \mathcal{T}(G,\mathcal{C})$ — некоторое ненулевое кодовое слово минимального веса. Обозначим за S множество рёбер графа G с ненулевыми пометками и A — множество вершин, инцидентных рёбрам из S. Пусть

$$\gamma := \frac{|A|}{n}.$$

По лемме 1 выполнено

$$|S| \leq \frac{wn}{2}\Big(\gamma^2 + \tfrac{\lambda}{w}(\gamma - \gamma^2)\Big).$$

Поскольку каждое ребро из S инцидентно по крайней мере двум вершинам, значит в среднем на одну вершину из A приходится рёбер

$$\frac{2|S|}{|A|} \leq \frac{wn\big(\gamma^2 + \frac{\lambda}{w}(\gamma - \gamma^2)\big)}{\gamma n} = w\Big(\gamma + \tfrac{\lambda}{w}(1-\gamma)\Big).$$

По условию, из каждого локального кода $\mathcal{C}_v$ для $v \in A$ исходит не менее εw рёбер с ненулевыми пометками, поэтому

$$\gamma + \frac{\lambda}{w}(1-\gamma) \geq \varepsilon \quad\Longrightarrow\quad \gamma \geq \frac{\varepsilon - \frac{\lambda}{w}}{1 - \frac{\lambda}{w}} \geq \varepsilon - \frac{\lambda}{w}.$$

Также

$$|S| \geq \frac{1}{2}\varepsilon w|A| = \gamma\varepsilon\frac{wn}{2} = \gamma\varepsilon|E(G)|.$$

Таким образом, относительное число рассматриваемых рёбер будет не меньше

$$\frac{|S|}{|E(G)|} \geq \varepsilon\left(\varepsilon - \frac{\lambda}{w}\right),$$

что и доказывает утверждение леммы. □

Теорема 1. *Пусть для каждой вершины $v \in V$ графа $\Gamma_{D,q}$ выбран локальный код $\mathcal{C}_v \subseteq \mathbb{F}_q^w$, $w = \frac{q^D-1}{q-1}$. Обозначим за $\mathcal{C} = \{\mathcal{C}_v\}_{v\in V}$ — семейство таких кодов, где длина каждого кода составляет w и расстояние не меньше εw. Тогда относительное минимальное расстояние кода $\mathcal{T} = \mathcal{T}(\Gamma_{D,q}, \mathcal{C})$ удовлетворяет оценке:*

$$\delta_{\min}(\mathcal{T}) \geq \varepsilon\left(\varepsilon - \sqrt{q^{1-D}}\right).$$

Доказательство. По лемме 2 второе собственное значение графа $\Gamma_{D,q}$ равно $\lambda = \sqrt{q^{D-1}}$. Согласно лемме 4, для $\mathcal{T}$ справедливо:

$$\delta_{\min}(\mathcal{T}) \geq \varepsilon\left(\varepsilon - \frac{\lambda}{w}\right) = \varepsilon\left(\varepsilon - \frac{\sqrt{q^{D-1}}}{w}\right).$$

Учитывая, что $w = \frac{q^D-1}{q-1} > q^{D-1}$, имеем

$$\delta_{\min}(\mathcal{T}) \geq \varepsilon\left(\varepsilon - \frac{\sqrt{q^{D-1}}}{q^{D-1}}\right) = \varepsilon\left(\varepsilon - \sqrt{q^{1-D}}\right). \qquad \square$$

Лемма 5. *Для кода $\mathcal{C}_{D,q,m}$ в каждой вершине $v \in V(\Gamma_{D,q})$ локальный код $\mathcal{C}_v$ эквивалентен проективному коду Рида-Маллера* $\mathrm{PRM}_q(m, D-1)$.

Доказательство. Обозначим $w := \frac{q^D-1}{q-1}$ — степень вершины $\Gamma_{D,q}$. В каждой вершине v графа $\Gamma_{D,q}$, соответствующей точке α проективного пространства, по определению локальный код $\mathcal{C}_v \subset \mathbb{F}_q^w$ задаётся следующим образом:

$$\mathcal{C}_v = \left\{ (f_\alpha(\beta))_{\beta\in\mathbb{P}_q^D:\langle\alpha,\beta\rangle=0} \;\middle|\; f_\alpha \in \mathbb{F}_q[a_0,\ldots,a_D]_m \right\}.$$

Поскольку условие $\langle\alpha, x\rangle = \sum_{i=0}^D \alpha_i x_i = 0$ является линейным относительно переменных $x_0,\ldots,x_D$, то, без ограничения общности полагая $\alpha_0 \neq 0$, можно выразить x_0 через $x_1,\ldots,x_D$:

$$x_0 = -\sum_{i=1}^{D} \frac{\alpha_i}{\alpha_0} x_i.$$

Подставляя это выражение в $f_\alpha(x_0,\ldots,x_D)$, получаем функцию

$$g_\alpha(x_1,\ldots,x_D) := f_\alpha\Big(-\sum_{i=1}^{D} \frac{\alpha_i}{\alpha_0} x_i,\ x_1,\ldots,x_D\Big).$$

Так как исходный полином f_α являлся однородным полиномом степени m от $D+1$ переменных, то полученный полином g_α является однородным полиномом степени m от D переменных. Таким образом, $\mathcal{C}_v$ является множеством векторов значений полиномов от D переменных, вычисленных в точках $(D-1)$-мерного проективного пространства, но не на однородных координатах, а на точках, отличающихся в константу раз. Поэтому $\mathcal{C}_v$ изометричен коду $\mathrm{PRM}_q(m, D-1)$, где изометрия задаётся умножением координат на константы, обусловленные выбором представителей точек $\mathbb{P}_q^{D-1}$.

В силу симметричности конструкции такие же рассуждения применимы для вершин графа $\Gamma_{D,q}$, соответствующих гиперплоскостям проективного пространства. □

Используя известную оценку минимального расстояния проективного кода Рида-Маллера (3), при $m < q$ получаем следствие.

Следствие 2. *Пусть $1 \le m < q$, $D \ge 2$. Для кода $\mathcal{C}_{D,q,m}$ в каждой вершине $v \in V(\Gamma_{D,q})$ минимальное расстояние локального кода $\mathcal{C}_v$ равно $(q-m+1)q^{D-2}$.*

Теорема 2. *Пусть $0 \le m < q$. Тогда $\delta_{\min}(\mathcal{C}_{2,q,m}) \ge \varepsilon^3$, где $\varepsilon = \frac{q-m}{q}$.*

Доказательство. Рассмотрим случай при $m = 0$. В этом случае все локальные коды — коды с повторением, а значит и весь код $\mathcal{C}_{2,q,m}$ — код с повторением, значит его относительное минимальное расстояние равно 1; кроме того, $\varepsilon = \frac{q-m}{q} = 1$, таким образом утверждение теоремы выполнено.

Теперь рассмотрим основной случай $m \ge 1$. По лемме 5 все локальные коды кода $\mathcal{C}_{2,q,m}$ изометричны кодам Рида-Маллера $\mathrm{PRM}_q(m, 1)$, то есть кодам Рида-Соломона $\mathrm{PRS}(q, m+1)$. Значит по следствию 2 минимальное расстояние всех локальных кодов равно

$$d_0 = q + 1 - m = \varepsilon q + 1.$$

Согласно лемме 3:

$$d\big(\mathcal{C}_{2,q,m}\big) \ge d_0\Big(d_0(d_0-1)+1\Big).$$

Так как $d_0 = \varepsilon\, q + 1$, то получим:

$$d\big(\mathcal{C}_{2,q,m}\big) \ge (\varepsilon q+1)\Big((\varepsilon q+1)(\varepsilon q+1-1)+1\Big) = (\varepsilon q+1)\Big((\varepsilon q+1)\varepsilon q+1\Big) =$$

$$= (\varepsilon q+1)\Big(\varepsilon^2 q^2 + \varepsilon q + 1\Big) \ge \varepsilon^3 (q+1)\Big(q^2+q+1\Big) = \varepsilon^3 n,$$

где $n = (q+1)(q^2+q+1)$ — длина кода $\mathcal{C}_{2,q,m}$. □

Теперь докажем оценку расстояния для размерности $D \geq 3$, используя теорему 1.

Теорема 3. *Пусть $m < q$, $D \geq 3$. Тогда*

$$\delta_{\min}(\mathcal{C}_{D,q,m}) \geq \frac{(q-m)^2(q-2)}{q^3}.$$

Доказательство. Заметим, что при $m = 0$ все локальные коды — коды с повторением, а значит и весь код $\mathcal{C}_{D,q,m}$ — код с повторением, значит его относительное минимальное расстояние равно 1, и утверждение теоремы выполнено. Далее предполагаем, что $m \geq 1$.

Пусть $w := \frac{q^D-1}{q-1}$ — степень вершины графа $\Gamma_{D,q}$. По следствию 2 все локальные коды кода $\mathcal{C}_{D,q,m}$ имеют минимальное расстояние

$$d_0 = (q-m+1)q^{D-2}.$$

Значит относительное минимальное расстояние можно оценить следующим образом:

$$\varepsilon = \frac{d_0}{w} = \frac{(q+1-m)q^{D-2}}{\frac{q^D-1}{q-1}} \geq \frac{(q+1-m)(q-1)}{q^2} \geq \frac{q-m}{q}.$$

Здесь мы использовали, что $(q+1-m)(q-1) = (q-m)q+m-1 \geq (q-m)q$. По теореме 1 получим оценку

$$\delta_{\min}(\mathcal{C}_{D,q,m}) \geq \varepsilon\left(\varepsilon - \sqrt{q^{1-D}}\right).$$

Поскольку $D \geq 3$, то $\sqrt{q^{1-D}} \leq 1/q$, значит

$$\varepsilon - \sqrt{q^{1-D}} \geq \frac{(q+1-m)(q-1)}{q^2} - \frac{1}{q} = \frac{(q+1-m)(q-1)-q}{q^2}$$

или

$$\varepsilon - \sqrt{q^{1-D}} \geq \frac{(q-m)(q-1)-1}{q^2} \geq \frac{(q-m)(q-2)}{q^2}.$$

Отсюда

$$\delta_{\min}(\mathcal{C}_{D,q,m}) \geq \frac{(q-m)}{q} \cdot \frac{(q-m)(q-2)}{q^2} \geq \frac{(q-m)^2(q-2)}{q^3}. \qquad \square$$

4.2. Оценка размерности

Далее будем считать, что размер поля q, параметр $m \in \mathbb{N}$ и размерность пространства $D \geq 2$ фиксированы. Далее опустим индексы и положим $\Gamma := \Gamma_{D,q}$, $E := E(\Gamma)$, $\mathcal{C} := \mathcal{C}_{D,q,m}$.

Введём 2 группы переменных $x = (x_0, \ldots, x_D)$ и $a = (a_0, \ldots, a_D)$. Через P обозначим кольцо полиномов от $2(D+1)$ переменных над конечным полем $\mathbb{F}_q$:

$$P = \mathbb{F}_q[x_0, \ldots, x_D, a_0, \ldots, a_D].$$

Обозначим $B(x, a) := \langle x, a \rangle = \sum_{i=0}^{D} a_i x_i$ — полином, задающий скалярное произведение.

Для $s \geq 0$ через P_s обозначим множество полиномов из P, однородных по переменным $x_0, \ldots, x_D$ и по переменным $a_0, \ldots, a_D$, таких, что $\deg_x(p) = \deg_a(p) = s$; также включим в P_s нулевой полином, чтобы оно было линейным пространством.

Рассмотрим отображение:

$$\sigma : P_m \to \mathbb{F}_q^E,$$

где

$$\sigma(p) = p\big|_E, \qquad \sigma(p; \alpha, \beta) = p(\alpha, \beta),$$

где $(\alpha, \beta) \in E$.

Лемма 6. *Пусть $1 \leq m < q/2$. Тогда для любого $p \in P_m$ выполняется $\sigma(p) = 0$ тогда и только тогда, когда $B \mid p$. Другими словами,* $\ker \sigma = P_{m-1} \cdot B$.

Доказательство.

Достаточность. Докажем, что если для $p \in P_m$ выполнено $B \mid p$, то $\sigma(p) = 0$. Напомним, что множество E представлялось в виде:

$$E = \{(\alpha, \beta) \in \mathbb{P}_q^D \times \mathbb{P}_q^D \mid B(\alpha, \beta) = 0\}.$$

Поскольку $p = Br$ для некоторого $r \in P$, то в силу делимости имеем:

$$p\big|_E = \underbrace{B\big|_E}_{\equiv 0} \cdot r\big|_E \equiv 0.$$

Необходимость. Докажем, что если для $p \in P_m$ верно, что $\sigma(p) = 0$, то $B \mid p$. Зафиксируем произвольный многочлен $p \in P_m$ такой, что $\sigma(p) = 0$, то есть $p|_E = 0$.

Домножим данный многочлен на a_0^m и получим:

$$a_0^m p(x,a)$$

Введём замену:

$$u = B(x,a).$$

Выразим:

$$a_0 x_0 = u - \sum_{i=1}^{D} a_i x_i. \tag{6}$$

Заметим, что

$$\deg_{x_0} p \le \deg_x p = m.$$

Поскольку $\deg_{x_0} p \le m$, то в каждом мономе $a_0^m p$ степень a_0 не меньше, чем степень x_0. Поэтому существует такой многочлен $\hat{p} \in \mathbb{F}_q[u, x_1, \ldots, x_D, a_0, \ldots, a_D]$, что

$$\hat{p}(B(x,a), x_1, \ldots, x_D, a_0, \ldots, a_D) = a_0^m p(x,a).$$

Каждому моному $\prod_{s=0}^{D} x_s^{i_s} a_s^{j_s}$ многочлена p соответствует слагаемое

$$a_0^{m-i_0}(u - \sum_{s=1}^{D} a_s x_s)^{i_0} a_0^{j_0} \prod_{s=1}^{D} x_s^{i_s} a_s^{j_s}$$

в полиноме $\hat{p}$. Отсюда

$$\deg_{u,x_1,\ldots,x_D} \hat{p} = \deg_x p = m.$$

Кроме того, поскольку $\sum_{s=0}^{D} j_s = \deg_a p = m$, то

$$\deg_a a_0^{m-i_0}(u - \sum_{s=1}^{D} a_s x_s)^{i_0} a_0^{j_0} \prod_{s=1}^{D} x_s^{i_s} a_s^{j_s} = m - i_0 + i_0 + \sum_{s=0}^{D} j_s = 2m < q,$$

значит $\deg_a \hat{p} < q$. Таким образом, степень $\hat{p}$ по каждой переменной не выше $q-1$.

Положим

$$\tilde{E} := \Big\{(u, x_1, \ldots, x_D, a_0, \ldots, a_D) \in \mathbb{F}_q^{2D+2} \;\Big|\; u = 0\Big\}.$$

Тогда для любой точки

$$(0, x_1, \ldots, x_D, a_0, \ldots, a_D) \in \tilde{E}, \qquad a_0 \neq 0$$

возьмём

$$x_0 = -a_0^{-1} \sum_{i=1}^{D} a_i x_i,$$

тогда точка (x, a) лежит на конусе, образованном прямыми из E. Отсюда $p(x, a) = 0$, и по определению многочлена $\hat{p}$ имеем

$$\hat{p}(0, x_1, \ldots, x_D, a_0, \ldots, a_D) = a_0^m p(x, a) = 0.$$

Осталось рассмотреть случай $a_0 = 0$. Покажем, что $\hat{p}$ делится на a_0. Сгруппируем в полиноме p все мономы, где x_0 входит в степени m, не содержащие a_0:

$$p(x, a) = x_0^m h(a_1, \ldots, a_D) + a_0 q(x, a) + \sum_{s=1}^{D} x_s q_s(x, a),$$

где h — однородный полином степени m по переменным $a_1, \ldots, a_D$, а $q, q_1, \ldots, q_D$ — некоторые полиномы.

Покажем, что $h \equiv 0$. Предположим противное. Тогда найдётся набор

$$(a_1, \ldots, a_D) \neq 0$$

такой, что

$$h(a_1, \ldots, a_D) \neq 0.$$

Рассмотрим точку

$$x = (1, 0, \ldots, 0), \qquad a = (0, a_1, \ldots, a_D).$$

Для неё

$$B(x, a) = a_0 x_0 + \sum_{s=1}^{D} a_s x_s = 0 \cdot 1 + \sum_{s=1}^{D} a_s \cdot 0 = 0,$$

поэтому $(x, a) \in E$. Так как $\sigma(p) = 0$, имеем

$$p(x, a) = 0.$$

С другой стороны, из приведённого разложения следует

$$p(1, 0, \ldots, 0,\ 0, a_1, \ldots, a_D) = h(a_1, \ldots, a_D) \neq 0,$$

поскольку $a_0 = 0$ и $x_1 = \cdots = x_D = 0$. Получено противоречие. Значит, $h \equiv 0$. Тогда в каждом слагаемом полинома $\hat{p}$

$$a_0^{m-i_0} \Big(u - \sum_{s=1}^{D} a_s x_s \Big)^{i_0} a_0^{j_0} \prod_{s=1}^{D} x_s^{i_s} a_s^{j_s}$$

либо $i_0 < m$, либо $j_0 > 0$. Следовательно, каждое слагаемое полинома $\hat{p}$ содержит множитель a_0, значит

$$a_0 \mid \hat{p}.$$

Таким образом, в случае $a_0 = 0$, также имеем

$$\hat{p}(0, x_1, \ldots, x_D, a_0, \ldots, a_D) = 0.$$

Следовательно, если $p\big|_E = 0$, то и $\hat{p}\big|_{\tilde{E}} = 0$.

Покажем, что $u \mid \hat{p}$. Пусть $c_{i_0,\ldots,i_D,j_0,\ldots,j_D} \in \mathbb{F}_q$ — коэффициент при мономе $u^{i_0} a_0^{j_0} \prod_{s=1}^{D} x_s^{i_s} a_s^{j_s}$ в полиноме $\hat{p}$. Тогда

$$\hat{p}(0, x_1, \ldots, x_D, a_0, \ldots, a_D) = \sum_{i_1,\ldots,i_D,j_0,\ldots,j_D} c_{0,i_1,\ldots,i_D,j_0,\ldots,j_D} \cdot a_0^{j_0} \prod_{s=1}^{D} x_s^{i_s} a_s^{j_s}. \quad (7)$$

Предположим, что существует хотя бы один коэффициент

$$c_{0,i_1,\ldots,i_D,j_0,\ldots,j_D} \neq 0 \quad \text{при некоторых } 0 \leq i_1, \ldots, i_D, j_0, \ldots, j_D < q.$$

Зная, что степень $\hat{p}$ по каждой переменной не выше $q-1$ и используя тот факт, что любая функция, заданная на конечном множестве $\mathbb{F}_q$, однозначно представима в виде многочлена степени не выше $q-1$, следует, что многочлен (7) не может быть тождественно равен нулю, если хотя бы один коэффициент $c_{0,i_1,\ldots,i_D,j_0,\ldots,j_D}$ не равен нулю. Это противоречит условию $\hat{p}|_{\tilde{E}} = 0$, а значит все эти коэффициенты должны быть нулевыми, и, следовательно, в $\hat{p}$ нет мономов, не содержащих u. Таким образом, $\hat{p} = u \cdot r$ для некоторого $r \in \mathbb{F}_q[u, x_1, \ldots, x_D, a_0, \ldots, a_D]$, а значит

$$a_0^m p = B \cdot r(B(x, a), x_1, \ldots, x_D, a_0, \ldots, a_D).$$

Поскольку B неприводим и $B \nmid a_0^m$, то $B \mid p$, что и требовалось доказать. □

Лемма 7. *Если $p \in P_m$, то $\sigma(p) \in \mathcal{C}$, где $0 \leq m \leq q-1$.*

Доказательство. Напомним, что код $\mathcal{C}$ определялся как множество функций $c : E \to F_q$, для которых для каждой точки $\alpha \in \mathbb{P}_q^D$ существуют однородные полиномы f_α и f^α степени m от $D+1$ переменной, такие что $f_\alpha(\beta) = c(\alpha, \beta) = f^\beta(\alpha)$ для всех $(\alpha, \beta) \in E$, то есть удовлетворяющих условию $\langle \alpha, \beta \rangle = 0$.

Для произвольного $\alpha \in \mathbb{P}_q^D$ определим функции $f_\alpha \in \mathbb{F}_q[a_0, \ldots, a_D]$, $f^\alpha \in \mathbb{F}_q[x_0, \ldots, x_D]$:

$$f_\alpha(a) := p(\alpha, a), \qquad f^\alpha(x) := p(x, \alpha).$$

Тогда для всех $\alpha, \beta \in \mathbb{P}_q^D$, таких что $\langle \alpha, \beta \rangle = 0$ верно:

$$\sigma(p)(\alpha, \beta) = f_\alpha(\beta) = f^\beta(\alpha).$$

Кроме того, поскольку p однороден по группам переменных a и x и имеет степень m по каждой из групп, то f_α и f^α также являются однородными полиномами степени m. Таким образом, отображение $\sigma(p)$ удовлетворяет обоим условиям определения кода $\mathcal{C}$ и, следовательно,

$$\sigma(p) \in \mathcal{C},$$

что и требовалось доказать. □

Теорема 4. *Пусть* $1 \leq m < q/2$. *Тогда*

$$\dim \mathcal{C}_{D,q,m} \geq \frac{2m+D}{D} \binom{m+D-1}{D-1}^2.$$

Доказательство. Оценим размерность пространства полиномов P_m. Базисом линейного пространства полиномов над $\mathbb{F}_q$ является множество мономов

$$\Big\{ \prod_{s=0}^{D} a_s^{i_s} x_s^{j_s} \;\Big|\; \sum_{s=0}^{D} i_s = \sum_{s=0}^{D} j_s = m \Big\}.$$

Есть $\binom{m+D}{D}$ способов разбить число m в сумму $D+1$ целых неотрицательных слагаемых, а значит мощность базиса и размерность пространства P_m равны

$$\dim P_m = \binom{m+D}{D}^2.$$

По лемме 6 выполнено $\ker \sigma = B \cdot P_{m-1}$, значит

$$\begin{aligned} \dim \operatorname{Im} \sigma &= \dim P_m - \dim \ker \sigma = \\ &= \dim P_m - \dim P_{m-1} = \\ &= \binom{m+D}{D}^2 - \binom{m-1+D}{D}^2. \end{aligned}$$

Учитывая, что $\binom{m+D}{D} = \frac{m+D}{D}\binom{m+D-1}{D-1}$, $\binom{m+D-1}{D} = \frac{m}{D}\binom{m+D-1}{D-1}$, имеем

$$\begin{aligned} \dim \operatorname{Im} \sigma &= \left(\left(\frac{m+D}{D} \right)^2 - \left(\frac{m}{D} \right)^2 \right) \binom{m+D-1}{D-1}^2 = \\ &= \frac{m^2 + 2mD + D^2 - m^2}{D^2} \binom{m+D-1}{D-1}^2 = \\ &= \frac{2m+D}{D} \binom{m+D-1}{D-1}^2. \end{aligned}$$

По лемме 7 выполнено $\sigma(P_m) \subseteq \mathcal{C}_{D,q,m}$, значит

$$\dim \mathcal{C}_{D,q,m} \geq \dim \operatorname{Im} \sigma \geq \frac{2m+D}{D} \binom{m+D-1}{D-1}^2. \qquad \square$$

4.3. Симметрия кода $\mathcal{C}_{D,q,m}$

Автоморфизмом линейного кода $\mathcal{C} \subseteq \mathbb{F}_q^n$ называется частный случай изометрии, когда $\mathcal{C}' = \mathcal{C}$. Можно показать, что любой автоморфизм задаётся мономиальным преобразованием $\varphi\colon \mathbb{F}_q^n \to \mathbb{F}_q^n$, переводящим $\mathcal{C}$ в себя.

Напомним, что $\Gamma_{D,q} = (\mathbb{P}_q^D, \mathbb{P}_q^D, E)$ — двудольный граф с множеством рёбер

$$E = \{(\alpha, \beta) \in \mathbb{P}_q^D \times \mathbb{P}_q^D \mid \langle \alpha, \beta \rangle = 0\}.$$

Код $\mathcal{C} \subseteq \mathbb{F}_q^E$ понимается как некоторое линейное множество функций $c : E \to \mathbb{F}_q$.

Каждая невырожденная матрица $A \in \mathrm{GL}(D+1, q)$ определяет проективное преобразование $\alpha \mapsto \widehat{A\alpha}$ и $\beta \mapsto \widehat{A^*\beta}$, где $A^* = (A^T)^{-1}$, $\alpha \in \mathbb{P}_q^D$, $\beta \in \mathbb{P}_q^D$. Введём отображения

$$\varphi_A : (\alpha, \beta) \mapsto (A\alpha, A^*\beta)$$
$$\hat{\varphi}_A : (\alpha, \beta) \mapsto (\widehat{A\alpha}, \widehat{A^*\beta}).$$

Для любых $(\alpha, \beta) \in E$ выполнено

$$\langle A\alpha, A^*\beta \rangle = (A\alpha)^T((A^T)^{-1}\beta) = \alpha^T A^T (A^T)^{-1}\beta = \alpha^T\beta = \langle \alpha, \beta \rangle = 0,$$

а значит и $\langle \widehat{A\alpha}, \widehat{A^*\beta} \rangle = 0$, то есть $\hat{\varphi}_A(\alpha, \beta) \in E$. Таким образом, $\hat{\varphi}_A$ задаёт автоморфизм графа $\Gamma_{D,q}$.

Покажем, что любое такое преобразование задаёт некоторый автоморфизм кода $\mathcal{C}_{D,q,m}$.

Утверждение 1. *Пусть $D \geq 2$, $1 \leq m < (q-1)D$, $E = E(\Gamma_{D,q})$, $\mathcal{C} = \mathcal{C}_{D,q,m}$. Для любой матрицы $A \in \mathrm{GL}(D+1, q)$ существует набор ненулевых констант $a = (a_{(\alpha,\beta)})_{(\alpha,\beta)\in E} \in \mathbb{F}_q^E$ такой, что для любого кодового слова $c \in \mathcal{C}$ существует кодовое слово $c' \in \mathcal{C}$, удовлетворяющее условию:*

$$c'(\alpha, \beta) = a_{(\alpha,\beta)} c\big(\widehat{A\alpha}, \widehat{A^*\beta}\big) \qquad \text{для всех } (\alpha, \beta) \in E,$$

где $A^* = (A^T)^{-1}$.

Доказательство. Для вектора $\alpha \in \mathbb{F}_q^{D+1} \setminus \{0\}$ определим величину $\kappa(\alpha)$, равную первой ненулевой координате в векторе α. Тогда $\alpha = \kappa(\alpha)\hat{\alpha}$.

Для каждой пары $(\alpha, \beta) \in E$ положим

$$a_{(\alpha,\beta)} := \big(\kappa(A\alpha) \cdot \kappa(A^*\beta)\big)^m.$$

Зафиксируем кодовое слово $c \in \mathcal{C}$ и определим c' как в условии теоремы:

$$c'(\alpha, \beta) := a_{(\alpha,\beta)} c\big(\widehat{A\alpha}, \widehat{A^*\beta}\big) \qquad \text{для всех } (\alpha, \beta) \in E.$$

Покажем, что $c' \in \mathcal{C}$. Рассмотрим проекции f'_α (соотв., f'^α) функции c' на окрестности вершин α левой (соотв., правой) доли графа $\Gamma_{D,q}$, тогда

$$f'_\alpha(\beta) = c'(\alpha, \beta) = f'^\beta(\alpha) \qquad \text{для всех } (\alpha, \beta) \in E.$$

Убедимся, что для всех $\alpha \in \mathbb{P}_q^D$ функции f'_α и f'^α являются кодовыми словами локальных кодов, то есть задаются однородными полиномами степени m. Зафиксируем произвольное $\alpha \in \mathbb{P}_q^D$, тогда для всех β таких, что $\langle \alpha, \beta \rangle = 0$, выполнено

$$f'_\alpha(\beta) = c'(\alpha, \beta) = a_{(\alpha,\beta)}\, c(\widehat{A\alpha}, \widehat{A^*\beta}) = \kappa(A\alpha)^m \cdot \kappa(A^*\beta)^m \cdot f_{\widehat{A\alpha}}(\widehat{A^*\beta}).$$

Поскольку $c \in \mathcal{C}$, то $f_{\widehat{A\alpha}}$ — однородный полином степени m, значит

$$\kappa(A^*\beta)^m \cdot f_{\widehat{A\alpha}}(\widehat{A^*\beta}) = f_{\widehat{A\alpha}}(\underbrace{\kappa(A^*\beta) \cdot \widehat{A^*\beta}}_{A^*\beta}) = f_{\widehat{A\alpha}}(A^*\beta).$$

Поскольку преобразование $x \mapsto A^*x$ задаёт линейную подстановку, то

$$\deg f'_\alpha = \deg f_{\widehat{A\alpha}}(A^*x) = \deg f_{\widehat{A\alpha}} = m.$$

Аналогично для фиксированного $\beta \in \mathbb{P}_q^D$ и всех $\alpha \in \mathbb{P}_q^D$ таких, что $\langle \alpha, \beta \rangle = 0$, получим

$$f'^\beta(\alpha) = c'(\alpha, \beta) = a_{(\alpha,\beta)}\, c\big(\widehat{A\alpha}, \widehat{A^*\beta}\big) = \kappa(A\alpha)^m \cdot \kappa(A^*\beta)^m \cdot f^{\widehat{A^*\beta}}\big(\widehat{A\alpha}\big).$$

Поскольку $f^{\widehat{A^*\beta}}$ является также однородным полиномом степени m, то

$$\kappa(A\alpha)^m \cdot f^{\widehat{A^*\beta}}\big(\widehat{A\alpha}\big) = f^{\widehat{A^*\beta}}\big(\underbrace{\kappa(A\alpha) \cdot \widehat{A\alpha}}_{A\alpha}\big) = f^{\widehat{A^*\beta}}(A\alpha).$$

В силу линейности преобразования $x \mapsto Ax$ имеем

$$\deg f'^\beta = \deg f^{\widehat{A^*\beta}}(Ax) = \deg f^{\widehat{A^*\beta}} = m.$$

Это означает, что c' удовлетворяет всем локальным ограничениям кода $\mathcal{C}$, значит $c' \in \mathcal{C}$. □

5. Заключение

В настоящей работе была рассмотрена конструкция кодов Таннера на графе инцидентности точек и гиперплоскостей D-мерного проективного пространства $\mathbb{P}^D(\mathbb{F}_q)$, где в роли локальных проверок — проективные коды Рида–Маллера. Получены нижние оценки на минимальное расстояние таких кодов и их размерности, а также сформулирована гипотеза для более широкого диапазона параметров на основе результатов численного эксперимента.

Также было показано, что описанные коды инвариантны при действии группы $\mathrm{PGL}(D+1, \mathbb{F}_q)$, что позволит использовать их как локальные коды при построении кодов с высокой симметрией на комплексах Рамануджана.

Список литературы

[1] A. Lubotzky, “High dimensional expanders”, *Proceedings of the International Congress of Mathematicians (ICM 2018, Rio de Janeiro)*, **2**, World Scientific, 2019, 705–730. DOI: 10.1142/9789813272880_0027.

[2] Г. В. Калачев, П. А. Пантелеев, “Семейство асимптотически хороших квантовых и локально тестируемых классических LDPC-кодов”, *Математические вопросы кибернетики*, Вып 21, ФИЗМАТЛИТ, М, 2023, 111–155. DOI: 10.20948/mvk-2023-111.

[3] F. Piñero, *An algebraic approach to graph codes*, PhD thesis, Technical University of Denmark, 2015.

[4] P. Beelen, T. Høholdt, F. Piñero, J. Justesen, “On the Dimension of Graph Codes with Reed–Solomon Component Codes”, *2013 IEEE International Symposium on Information Theory Proceedings*, IEEE, 2013, 1227–1231. DOI: 10.1109/ISIT.2013.6620422.

[5] T. Høholdt, J. Justesen, “Graph codes with Reed–Solomon component codes”, *2006 IEEE International Symposium on Information Theory*, IEEE, 2006, 2022–2026. DOI: 10.1109/ISIT.2006.261904.

[6] H. Janwa, A. K. Lal, “On Tanner Codes: Minimum Distance and Decoding”, *Applicable Algebra in Engineering, Communication and Computing*, **13** (2003), 335–347. DOI: 10.1007/s00200-003-0098-4.

[7] T. Høholdt, J. Justesen, “The Minimum Distance of Graph Codes”, *Coding and Cryptology (IWCC 2011)*, Lecture Notes in Computer Science, **6639**, Springer, 2011, 201–212. DOI: 10.1007/978-3-642-20901-7_12.

[8] I. Dinur, S. Liu, R. Y. Zhang, “New codes on high dimensional expanders”, *40th Computational Complexity Conference (CCC 2025)*, Leibniz International Proceedings in Informatics, **339**, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 27:1–27:42. DOI: 10.4230/LIPIcs.CCC.2025.27.

[9] F. Rodier, “Codes from flag varieties over a finite field”, *Journal of Pure and Applied Algebra*, **178**:2 (2003), 203–214. DOI: 10.1016/S0022-4049(02)00188-3.

[10] А. В. Чашкин, *Лекции по дискретной математике*, Физматлит, М, 2007, 416 с.

[11] R. Tanner, "A recursive approach to low complexity codes", *IEEE Transactions on Information Theory*, **27**:5 (1981), 533–547. DOI: 10.1109/TIT.1981.1056404.

[12] В. А. Емельичев, О. И. Мельников, В. И. Сарванов, Р. И. Тышкевич, *Лекции по теории графов*, Наука, М, 1990, 384 с.

[13] A. B. Sorensen, "Projective Reed–Muller Codes", *IEEE Transactions on Information Theory*, **37**:6 (1991), 1567–1576. DOI: 10.1109/18.104317.

[14] M. Sipser, D. A. Spielman, "Expander codes", *IEEE Transactions on Information Theory*, **42**:6 (1996), 1710–1722. DOI: 10.1109/18.556667.

[15] N. Alon, J. H. Spencer, *The Probabilistic Method*, 3, John Wiley & Sons, 2008. DOI: 10.1002/9780470277331.

Приложение А. Численный эксперимент

В данном разделе приведены результаты вычисления размерности кода $\mathcal{C}_{2,q,m}$. Исходный код, который использовался в данном эксперименте, доступен по ссылке: `https://github.com/Alinemsu/Tanner_matrix`

Результаты подсчёта характеристик проверочных матриц и параметров кода $\mathcal{C}_{2,q,m}$ для $q \in \{2, 3, 4, 5, 7, 8, 9, 11\}$ приведены в таблице ниже.

q	m	Ранг	Размерность	Высота	Ширина
2	0	20	1	28	21
2	1	13	8	14	21
3	0	51	1	78	52
3	1	44	8	52	52
3	2	25	27	26	52
4	0	104	1	168	105
4	1	97	8	126	105
4	2	76	29	84	105
4	3	41	64	42	105
5	0	185	1	310	186
5	1	178	8	248	186
5	2	159	27	186	186
5	3	116	70	124	186
5	4	61	125	62	186
7	0	455	1	798	456
7	1	448	8	684	456

7	2	429	27	570	456
7	3	392	64	456	456
7	4	315	141	342	456
7	5	220	236	228	456
7	6	113	343	114	456
8	0	656	1	1168	657
8	1	649	8	1022	657
8	2	630	27	876	657
8	3	593	64	730	657
8	4	520	137	584	657
8	5	411	246	438	657
8	6	284	373	292	657
8	7	145	512	146	657
9	0	909	1	1638	910
9	1	902	8	1456	910
9	2	883	27	1274	910
9	3	846	64	1092	910
9	4	779	**131**	910	910
9	5	664	246	728	910
9	6	519	391	546	910
9	7	356	554	364	910
9	8	181	729	182	910
11	0	1595	1	2926	1596
11	1	1588	8	2660	1596
11	2	1569	27	2394	1596
11	3	1532	64	2128	1596
11	4	1471	125	1862	1596
11	5	1380	216	1596	1596
11	6	1205	391	1330	1596
11	7	1000	596	1064	1596
11	8	771	825	798	1596
11	9	524	1072	532	1596
11	10	265	1331	266	1596

Статья поступила 29 апреля 2026 г.

Bounds on the parameters of Tanner codes on the graph of projective space

A. A. Yusufova, G. V. Kalachev

This paper considers Tanner codes constructed on the incidence graph of points and hyperplanes of a projective space, with projective Reed–Muller codes as component codes. The assignment of symbols of local codes to edges is performed so that the resulting Tanner code inherits the symmetry of the projective space. Lower bounds on the minimum distance and dimension of such codes are obtained. In particular, it is shown that, for certain parameter values, these codes are asymptotically good.

Keywords: Tanner codes, incidence graph, projective space, minimum distance, code dimension.

References

[1] A. Lubotzky, "High dimensional expanders", *Proceedings of the International Congress of Mathematicians (ICM 2018, Rio de Janeiro)*, **2**, World Scientific, 2019, 705–730. DOI: 10.1142/9789813272880_0027.

[2] G. V. Kalachev, P. A. Panteleev, "A family of asymptotically good quantum and locally testable classical LDPC codes", *Matematicheskie voprosy kibernetiki*, Issue 21, FIZMATLIT, M, 2023, 111–155 (in Russian. DOI: 10.20948/mvk-2023-111).

[3] F. Piñero, *An algebraic approach to graph codes*, PhD thesis, Technical University of Denmark, 2015.

[4] P. Beelen, T. Høholdt, F. Piñero, J. Justesen, "On the Dimension of Graph Codes with Reed–Solomon Component Codes", *2013 IEEE International Symposium on Information Theory Proceedings*, IEEE, 2013, 1227–1231. DOI: 10.1109/ISIT.2013.6620422.

[5] T. Høholdt, J. Justesen, "Graph codes with Reed–Solomon component codes", *2006 IEEE International Symposium on Information Theory*, IEEE, 2006, 2022–2026. DOI: 10.1109/ISIT.2006.261904.

[6] H. Janwa, A. K. Lal, "On Tanner Codes: Minimum Distance and Decoding", *Applicable Algebra in Engineering, Communication and Computing*, **13** (2003), 335–347. DOI: 10.1007/s00200-003-0098-4.

[7] T. Høholdt, J. Justesen, "The Minimum Distance of Graph Codes", *Coding and Cryptology (IWCC 2011)*, Lecture Notes in Computer Science, **6639**, Springer, 2011, 201–212. DOI: 10.1007/978-3-642-20901-7_12.

[8] I. Dinur, S. Liu, R. Y. Zhang, "New codes on high dimensional expanders", *40th Computational Complexity Conference (CCC 2025)*, Leibniz International Proceedings in Informatics, **339**, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 27:1–27:42. DOI: 10.4230/LIPIcs.CCC.2025.27.

[9] F. Rodier, "Codes from flag varieties over a finite field", *Journal of Pure and Applied Algebra*, **178**:2 (2003), 203–214. DOI: 10.1016/S0022-4049(02)00188-3.

[10] A. V. Chashkin, *Lektsii po diskretnoi matematike*, Fizmatlit, M, 2007 (in Russian), 416 pp.

[11] R. Tanner, "A recursive approach to low complexity codes", *IEEE Transactions on Information Theory*, **27**:5 (1981), 533–547. DOI: 10.1109/TIT.1981.1056404.

[12] V. A. Emelichev, O. I. Melnikov, V. I. Sarvanov, R. I. Tyshkevich, *Lektsii po teorii grafov*, Nauka, M, 1990 (in Russian), 384 pp.

[13] A. B. Sorensen, "Projective Reed–Muller Codes", *IEEE Transactions on Information Theory*, **37**:6 (1991), 1567–1576. DOI: 10.1109/18.104317.

[14] M. Sipser, D. A. Spielman, "Expander codes", *IEEE Transactions on Information Theory*, **42**:6 (1996), 1710–1722. DOI: 10.1109/18.556667.

[15] N. Alon, J. H. Spencer, *The Probabilistic Method*, 3, John Wiley & Sons, 2008. DOI: 10.1002/9780470277331.

Received on April 29, 2026

Часть 4
Материалы семинаров

Доклады семинара «Теория автоматов»

В первом полугодии 2026 года на научном семинаре «Теория автоматов» под руководством профессора Эльяра Эльдаровича Гасанова состоялось 18 докладов.

11 февраля 2026 года

О конфигурации паросочетаний, ее свойствах и приложении в криптографии

инж. 1 кат. А. И. Болотников

В рамках доклада автор представит новую конструкцию — конфигурацию паросочетаний, и проведет обзор свойств конфигурации паросочетаний и ее характеристического многочлена. Также будет сформулирована задача о неправильной функции весов, показана ее NP-полнота и построение на ее основе системы шифрования Болотникова-Ирматова (криптосистемы паросочетаний).

18 февраля 2026 года

Сложность проблемы сюръективного гомоморфизма для циклов

инж. 1 кат. Н. П. Корчагин

Для графа H в задаче Surj-Hom(H) по данному графу G требуется определить, существует ли сюръективный гомоморфизм из G на H. Доклад посвящен сложности задачи Surj-Hom для графов, которые получаются из неориентированных циклов добавлением ориентации некоторым ребрам и в которых каждая вершина содержит петлю. Будет представлен обзор задачи, включая современные достижения и результаты докладчика. Задача рассматривается как частный случай массовой задачи сюръективного удовлетворения ограничениям SCSP. Вводится ряд свойств, которые для множества отношений Γ позволяют определять сложность SCSP(Γ) с помощью сведения или путем анализа функций, сохраняющих эти отношения. Введенные свойства используются, чтобы определить сложность задачи для всех рассматриваемых графов, кроме шести циклов длины 4, 5 и 6.

25 февраля 2026 года

О трех начальных приближениях к формальному определению визуального образа в произвольной визуальной среде

проф. В. Н. Козлов

Распознавание визуальных образов — одна из центральных задач для интеллектуальных систем. Продвижению здесь математических, теоремных методов исследования в немалой степени мешает отсутствие полного и приемлемого формального определения понятия визуального образа в произвольной визуальной среде. На содержательном, интуитивном уровне ясно, что зрительный образ — понятие сложное, многоплановое и многоуровневое, и вряд ли определение образа может быть дано некоторым «единовременным» усилием. В докладе будет представлена идея подхода к определению образа последовательными приближениями, и описаны три начальных приближения.

4 марта 2026 года

Формальная верификация протоколов консенсуса с византийской отказоустойчивостью

асп. К. В. Зиборов

Протоколы консенсуса являются фундаментом блокчейн-технологий, обеспечивая согласованность данных в распределенных сетях. В условиях византийских отказов, когда часть узлов может вести себя злонамеренно, ошибки в дизайне таких протоколов приводят к нарушениям критических свойств согласованности и живучести. В рамках работы проработана методология формальной спецификации и верификации BFT-протоколов с использованием метода проверки модели, позволяющая получать более строгие гарантии корректности.

Помимо классических гарантий, вводится строгая формализация устойчивости к цензуре для BFT-протоколов с лидером как ограниченной гарантии качества цепочки; показано, что стандартная ротация лидера может приводить к нарушению этой гарантии. Предлагается и формально верифицируется правило выбора лидера, обеспечивающее устойчивость к цензуре без ослабления исходных гарантий безопасности и живучести. Предложенная методология проверена на примерах индустриальных протоколов IBFT и SmartBFT.

11 марта 2026 года

Преодоление ограничений обучения сиамских нейронных сетей в задаче оптического распознавания символов

асп. А. К. Мокин

Метрическое обучение находит широкое применение в современных задачах, таких как оптическое распознавание символов (OCR), идентификация лиц и рекомендательные системы. Доклад посвящен преодолению возникающих в ходе решения таких задач практических проблем, связанных с обучением сиамских нейронных сетей. В качестве примера будет рассмотрена задача OCR и ограничения существующих методов и подходов, из-за которых сиамские сети могут работать хуже и чаще ошибаться, например, при распознавании рукописного текста с большой вариативностью или визуально похожих классов. Будут представлены способы адаптации обучения для достижения высокой точности при небольших данных и преодоления ограничений известных методов и подходов.

18 марта 2026 года

Проблема полноты в классе линейных дефинитных автоматов

асп. И. В. Молдованов

Рассматриваемый в докладе класс линейных дефинитных автоматов возникает путем замыкания по операциям суперпозиции системы, состоящей из сумматора и задержек с заданным начальным состоянием. Этот класс представляет интерес тем, что, в отличие от класса линейных автоматов, содержит конечные полные системы относительно операций суперпозиции. Ранее для конечнопорожденных подмножеств этого класса, содержащих нулевую константу, был получен критерий полноты в терминах системы замкнутых классов, а также предложен алгоритм распознавания полноты. В докладе этот результат обобщается: путем расширения системы замкнутых классов получен критерий полноты для произвольных подмножеств класса линейных дефинитных автоматов. На его основе может быть построен алгоритм распознавания полноты произвольных конечных подмножеств данного класса.

25 марта 2026 года

Методы построения представлений объектов с использованием внешней информации о свойствах наборов объектов

м.н.с. А. М. Колосов

В докладе рассматриваются методы построения представлений объектов с использованием внешней информации о свойствах наборов объектов. Актуальность работы связана с тем, что задачи построения, снижения размерности, бинаризации и улучшения качества векторных представлений играют важную роль в машинном обучении и анализе данных. В центре внимания находится постановка задачи обобщенного неметрического многомерного шкалирования, в которой в качестве входных данных используются не сами близости между объектами, а сведения о порядке близостей в парах пар объектов. На этой основе предлагаются методы построения представлений, а также методы снижения размерности, бинаризации и улучшения качества уже имеющихся векторных представлений. Отдельное внимание уделяется использованию методов машинного обучения, позволяющих получить единый подход к решению этого круга задач. Представленные результаты показывают практическую значимость предложенных методов и их применимость к различным типам данных.

1 апреля 2026 года

Об одном необходимом условии устойчивости криптосистемы паросочетаний

доц. А. А. Ирматов, инж. 1 кат. А. И. Болотников

В докладе рассматривается составная часть секретного ключа криптосистемы паросочетаний — функция весов, для которой задача о чередующемся взвешенном пути решается за полиномиальное время. Значительная часть координат векторов весов данной функции используется в качестве «шума» для усложнения взлома всей системы шифрования. В докладе будет представлено условие на «шум», необходимое для устойчивости системы к атакам, опирающимся на размерность специальных подмножеств векторов весов.

Подбор оптимальной схемы квантизации нейронных сетей с помощью градиентных методов

с.н.с. В. С. Половников, асп. Д. Н. Давыдова

Квантизация является одной из основных техник оптимизации и ускорения больших языковых моделей, но подбор разрядности вручную или перебором для каждого слоя нейронной сети был бы слишком долгим и затрачивал бы слишком много вычислительных ресурсов. Доклад будет посвящен применению метода дифференцируемого поиска архитектуры нейронной сети (DARTS) для подбора оптимальной схемы квантизации генеративной языковой модели, будут приведены результаты экспериментов. В отличие от оригинального DARTS, вместо архитектурных параметров с помощью градиентного спуска подбираются оптимальный метод квантизации и разрядность для каждого линейного слоя модели.

Устойчивость нечеткой классификации пользователя для определения уровня физических нагрузок

проф. А. П. Рыжов, студ. А. С. Гусев

В работе исследуется устойчивость нечеткого вывода Мамдани относительно параметров функций принадлежности. Рассматривается ситуация, когда функции принадлежности заданы с некоторой возможной ошибкой, и оценивается влияние этой ошибки на результат вывода. Такая ситуация характерна для экспертного задания функций принадлежности и проблемы холодного старта. На примере классификации физического состояния пользователя по данным pose estimation показано, что нечеткий вывод позволяет корректно интерпретировать данные, полученные трекингом ключевых точек тела. Установлено, что малые ошибки в заданиях функций принадлежности не приводят к существенным ошибкам вывода, что гарантирует корректность работы систем нечеткого вывода в рассматриваемых условиях.

Нечеткая агрегация рисков на деревьях с использованием t-норм

проф. А. П. Рыжов, студ. Ф. А. Федотов

В работе рассматривается задача нечеткой агрегации рисков в иерархических структурах, моделируемых ориентированными деревьями, с использованием t-норм. Исследуются строгие архимедовы t-нормы и их представление через аддитивные генераторы, что позволяет получить аналитические формулы для итоговой агрегированной оценки и для

ее чувствительности к изменениям листовых значений. На этой основе вводятся и анализируются характеристики устойчивости к шуму, селективности инцидентов и уровневой детекции событий. Также обсуждаются общие подходы к выделению и конструированию классов операторов агрегации, согласованных с требованиями робастности и чувствительности.

8 апреля 2026 года

Аналог цифровой подписи для буквенных текстов на основе инвариантов аффинных преобразований для алфавитных изображений

проф. В. Н. Козлов

Как правило, сердцевиной цифровой подписи является функция, у которой ее значение при заданном значении аргумента вычисляется легко, а обратное, т.е. вычисление значения аргумента при известном значении функции, трудно. В докладе будет представлен аналог цифровой подписи для буквенных текстов на другой принципиальной основе, с использованием вводимого понятия алфавитного изображения и кодов изображений, определяющих их с точностью до аффинных преобразований.

15 апреля 2026 года

Предполные классы в классе полиномов четырехзначной логики, содержащие все полиномиальные функции одной переменной

студ. И. М. Янушкевич

В работе исследовались полиномы в четырехзначной логике. Было получено взаимно-однозначное соответствие между полиномами в четырехзначной логике и булевыми функциями той же арности. С помощью полученного соответствия были описаны два предполных класса, содержащие все полиномиальные функции одной переменной. Один из них можно считать аналогом класса Слупецкого.

Реализация булевых функций и операции сложения в клеточных автоматах с локаторами

студ. Ч. А. Надиров

Клеточный автомат с локаторами — это обобщение классического клеточного автомата, в котором ячейки могут обмениваться информацией не только с соседями, но и на расстоянии с помощью сигналов. В работе

разработан алгоритм сложения нескольких натуральных чисел в этой модели, работающий за время, пропорциональное логарифму от числа слагаемых. Также доказано, что почти все булевые функции можно реализовать клеточными автоматами с локаторами со сложностью 2^n, причем эта сложность неулучшаема по порядку, а время вычисления значения функции занимает всего 2 такта.

22 апреля 2026 года

Задача подсчета точек клеточными автоматами с локаторами

студ. А. Е. Бобров

В работе исследуются клеточные автоматы с локаторами. Предложены алгоритмы подсчета количества точек на плоскости и извлечения квадратного корня числа, время выполнения которых логарифмически зависит от величины обрабатываемого множества/входного значения.

О сложности булевых операторов в некоторых классах клеточных схем

студ. Д. Э. Каламбетов

Клеточные схемы — структуры, где элементы расположены по решетке и могут взаимодействовать только с соседними элементами. В работе рассматриваются клеточные схемы с бинарными значениями входов и выходов, но с элементами арифметического сложения и без ограничения на значность передаваемых через проводники сигналов. Найдены порядки роста сложности дешифратора в классе клеточных схем из линейных элементов, элемента Хэвисайда и элементов, вычисляющих определенные разряды целых неотрицательных чисел, а также двоичного умножителя в классе клеточных схем из кусочно-линейных элементов, в том числе в конечном базисе.

Об алгоритмах построения групп автоморфизмов нагруженных графов, мультиграфов и гиперграфов

студ. Е. И. Яковенко

В докладе будет рассмотрен алгоритм МакКея, используемый для построения систем образующих групп автоморфизмов графов. Алгоритм распространен на нагруженные графы, мультиграфы и гиперграфы, возникающие при анализе молекулярных структур в химии. Рассмотрены примеры применения разработанного алгоритма к молекулярным графам.

6 мая 2026 года

Минионы и алгоритмы для решения задачи удовлетворения ограничениям

с.н.с. Д. Н. Жук

Открытия 2025 года показали, что почти любой универсальный алгоритм для задачи удовлетворения ограничениям задается абстрактным минионом, а применимость алгоритма зависит от существования гомоморфизма минионов. В докладе мы покажем, как абстрактные минионы возникают из замкнутых классов функций k-значной логики, продемонстрируем на примерах, как минионы задают конкретные алгоритмы. На примере классификации Шефера мы покажем, как наличие гомоморфизма минионов связано с применимостью алгоритма. В конце мы сформулируем конкретные (почти комбинаторные) вопросы, решение которых станет прорывом в теории алгоритмов.

К сведению авторов публикаций в журнале «Интеллектуальные системы. Теория и приложения»

В соответствии с требованиями ВАК РФ к изданиям, входящим в перечень ведущих рецензируемых научных журналов и изданий, в которых могут быть опубликованы основные научные результаты диссертаций на соискание ученой степени доктора и кандидата наук, статьи в журнал «Интеллектуальные системы. Теория и приложения» предоставляются авторами в следующей форме:

1. Статьи, набранные в пакете LaTeX, предоставляются к загрузке через WEB-форму https://intsysmagazine.ru/submit .
2. В преамбуле статьи должны содержаться название статьи на русском и английском языках, аннотация на русском и английском языках (не более 150 слов), список ключевых слов на русском и английском языках (не более 12 слов), информация об авторах: Ф.И.О. полностью, место работы, должность, ученая степень и/или звание (если имеется), для аспирантов ФИО научного руководителя, e-mail, ORCID.
3. Список литературы оформляется в едином формате, установленном системой Российского индекса научного цитирования. Списки на русском и английском языках приводятся в конце файла с текстом статьи, и должны быть оформлены с помощью пакета `amsbib`.
4. За публикацию статей в журнале «Интеллектуальные системы. Теория и приложения» с авторов (в том числе аспирантов высших учебных заведений) статей, рекомендованных к публикации, плата не взимается. Авторам бесплатно предоставляется номер журнала, в котором вышла статья. Журнал распространяется по подписке, экземпляры журнала рассылаются подписчикам наложенным платежом. Условия подписки публикуются в каталоге НТИ «Роспечать», индекс журнала 64559.
5. Электронная версия журнала находится в открытом доступе. Все материалы распространяются на условиях лицензии Creative Commons Attribution 4.0 International (CC BY 4.0). Электронные версии всех выпущенных номеров размещаются на сайте

https://intsysmagazine.ru

и доступны для свободного чтения и скачивания. Авторы сохраняют авторские права на свои статьи.

Подписано в печать: 21.09.2026
Дата выхода: 25.09.2026
Тираж: 200 экз.
Цена свободная
Свидетельство о регистрации СМИ: ПИ № ФС77–58444 от 25 июня 2014 г., выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор).