

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
НАУКИ ИНСТИТУТ СИСТЕМНОГО ПРОГРАММИРОВАНИЯ
ИМ. В.П. ИВАННИКОВА РОССИЙСКОЙ АКАДЕМИИ НАУК

На правах рукописи

Бабенко Михаил Григорьевич

**МАТЕМАТИЧЕСКИЕ МОДЕЛИ, МЕТОДЫ И
АЛГОРИТМЫ ОБРАБОТКИ ЗАШИФРОВАННЫХ
ДАННЫХ В РАСПРЕДЕЛЕННЫХ СРЕДАХ**

Специальность 2.3.5 (05.13.11) —
Математическое и программное обеспечение вычислительных систем,
комплексов и компьютерных сетей

Диссертация на соискание учёной степени
доктора физико-математических наук

Научный консультант:
Академик РАН,
доктор физико-математических наук, профессор РАН
Аветисян Арутюн Ишханович

Москва — 2022

Оглавление

| | |
|---|----|
| ВВЕДЕНИЕ | 8 |
| Глава 1. СОВРЕМЕННЫЕ МЕТОДЫ ОБЕСПЕЧЕНИЯ | |
| БЕЗОПАСНОСТИ В РАСПРЕДЕЛЕННЫХ | |
| СИСТЕМАХ ОБРАБОТКИ ДАННЫХ | |
| 1.1 Классификация угроз информационной безопасности | 27 |
| 1.2 Структурная модель распределенной обработки данных в виртуальных средах | 34 |
| 1.3 Проблемы использования облачных вычислений в условиях неопределенности возникновения киберугроз | 36 |
| 1.4 Подходы к обеспечению надежного и безопасного хранения и обработки данных на основе структур доступа | 44 |
| 1.5 Современные подходы к построению систем обработки конфиденциальных данных в закодированном виде | 46 |
| 1.5.1 Частично гомоморфные вычисления | 48 |
| 1.5.2 В некоторой степени гомоморфные вычисления | 49 |
| 1.5.3 Полностью гомоморфные вычисления | 52 |
| 1.5.3.1 СККС схема для работы с закодированными вещественными числами | 59 |
| 1.5.3.2 BFV схема для работы с закодированными целыми числами | 60 |
| 1.6 Выводы по первой главе | 61 |
| Глава 2. МОДЕЛИ И МЕТОДЫ ОБРАБОТКИ ДАННЫХ С | |
| ИСПОЛЬЗОВАНИЕМ ГОМОМОРФНЫХ | |
| ВЫЧИСЛЕНИЙ | |
| 2.1 Гомоморфные вычисления над кольцом вычетов с делителями нуля, основанные на избыточной системе остаточных классов | 64 |
| 2.2 Схема WA-MRC-RRNS надежного и безопасного хранения и обработки данных на основе структуры доступа | 67 |
| 2.2.1 Вероятность потери данных при хранении с использованием схемы WA-MRC-RRNS | 71 |

| | | |
|---------|--|-----|
| 2.2.2 | Стратегии распределенного хранения данных | 75 |
| 2.2.3 | Сравнение производительности структур доступа | 80 |
| 2.3 | Проблема сговора в облачных сервисах при распределенном хранении и обработке данных | 87 |
| 2.4 | Атака открытым текстом на гомоморфные коды | 92 |
| 2.4.1 | Атака открытым текстом на HORNS | 92 |
| 2.4.2 | Атака открытым текстом на схему Asmuth-Bloom | 94 |
| 2.5 | Схема AC-RRNS и ее свойства | 95 |
| 2.5.1 | Вычислительная безопасность AC-RRNS | 97 |
| 2.5.2 | Свойства AC-RRNS | 103 |
| 2.5.2.1 | Избыточность данных | 103 |
| 2.5.2.2 | Вероятность получения несанкционированного доступа к данным посредством облачного сговора | 104 |
| 2.6 | Выводы по второй главе | 107 |

Глава 3. РАЗРАБОТКА МЕТОДОВ И АЛГОРИТМОВ ОПРЕДЕЛЕНИЯ ЗНАКА И СРАВНЕНИЯ ГОМОМОРФНО ЗАКОДИРОВАННЫХ ЧИСЕЛ НАД КОЛЬЦОМ ВЫЧЕТОВ С ДЕЛИТЕЛЯМИ НУЛЯ

| | | |
|-------|---|-----|
| 3.1 | Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с делителями нуля | 110 |
| 3.1.1 | Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с четным диапазоном | 111 |
| 3.1.2 | Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с нечетным диапазоном | 112 |
| 3.2 | Подходы к сравнению чисел в кольце вычетов \mathbb{Z}_m | 114 |
| 3.3 | Методы сравнения чисел, основанные на переводе чисел из RNS в двоичную систему счисления | 120 |
| 3.3.1 | Китайская теорема об остатках | 123 |
| 3.3.2 | Обобщенная позиционная система счисления | 123 |
| 3.3.3 | Приближенный метод | 124 |
| 3.4 | Методы сравнения чисел в RNS с использованием диагональной функции | 125 |
| 3.5 | Функция ядра Акушского и ее свойства | 128 |

| | | |
|--------|--|-----|
| 3.6 | Сравнение чисел с помощью функции ядра Акушского | 135 |
| 3.6.1 | Проблема монотонности функции ядра | 135 |
| 3.6.2 | Метод построения функций ядра Акушского, не содержащих критических ядер | 138 |
| 3.6.3 | Функция Pirlo и Impedovo | 142 |
| 3.7 | Сравнение чисел на основе алгоритма определения знака числа . | 144 |
| 3.8 | Модифицированная диагональная функция | 146 |
| 3.9 | Модификация алгоритма сравнения чисел в RNS | 151 |
| 3.9.1 | Сравнение чисел в RNS с нечетным диапазоном | 151 |
| 3.9.2 | Сравнение чисел в RNS, содержащий модуль, равный степени двойки | 154 |
| 3.10 | Оценка производительности алгоритмов сравнения чисел в RNS . | 156 |
| 3.10.1 | Анализ алгоритмов сравнения чисел в RNS | 156 |
| 3.10.2 | Анализ и оценка сложности алгоритмов сравнения чисел в RNS | 159 |
| 3.11 | Выводы по третьей главе | 168 |

Глава 4. РАЗРАБОТКА И ОПТИМИЗАЦИЯ МЕТОДОВ И АЛГОРИТМОВ ОПРЕДЕЛЕНИЯ ЗНАКА И СРАВНЕНИЯ ГОМОМОРФНО

| | | |
|-------|---|------------|
| | ЗАКОДИРОВАННЫХ ЧИСЕЛ НАД ПОЛЕМ | 171 |
| 4.1 | Интерполяция функции знака числа над полем \mathbb{Z}_m | 171 |
| 4.2 | Сравнение чисел над полями характеристики m | 172 |
| 4.3 | Матрицы специального вида над \mathbb{Z}_m и их свойства | 174 |
| 4.4 | Полиномиальная интерполяция функции сравнения чисел над простым полем | 177 |
| 4.5 | Коэффициенты многочлена, определенного для интерполяционной функции сравнения чисел над простым полем | 181 |
| 4.6 | Аппроксимация функции определения знака закодированного числа над полем \mathbb{R} | 188 |
| 4.7 | Об оценке точности полиномиальной аппроксимации функции определения знака закодированного числа над полем \mathbb{R} | 195 |
| 4.8 | О наилучшем приближении функции определения знака закодированного числа многочленом над полем \mathbb{R} | 209 |
| 4.8.1 | Норма и ее свойства | 211 |

| | | |
|-------|---|-----|
| 4.8.2 | Приближения функции определения знака закодированного числа над полем \mathbb{R} многочленами Бернштейна | 214 |
| 4.8.3 | Свойства многочлена наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R} . . | 216 |
| 4.8.4 | Количество многочленов наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R} являющихся нечетными функциями | 224 |
| 4.8.5 | Количество многочленов наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R} являющихся функциями общего вида | 230 |
| 4.9 | Нейросетевой метод определения знака закодированного числа над полем \mathbb{R} | 237 |
| 4.10 | Выводы по четвертой главе | 242 |

Глава 5. ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ

АЛГОРИТМОВ ОБНАРУЖЕНИЯ И

ИСПРАВЛЕНИЯ АРИФМЕТИЧЕСКИХ ОШИБОК

ОБРАБОТКИ ЗАКОДИРОВАННЫХ ДАННЫХ С

ИСПОЛЬЗОВАНИЕМ СВОЙСТВ РАНГА ЧИСЛА 244

| | | |
|-------|--|-----|
| 5.1 | Ранг числа и его свойства | 245 |
| 5.2 | Представление ранга числа в виде алгебраического многочлена над \mathbb{Z}_p | 247 |
| 5.3 | Разработка методов обнаружения и исправления ошибок арифметических операций обработки закодированных данных с использованием свойств ранга числа | 255 |
| 5.3.1 | Разработка методов обнаружения и исправления ошибок арифметических операций с использованием свойств ранга числа $r(X)$ | 255 |
| 5.3.2 | Разработка методов обнаружения и исправления ошибок арифметических операций с использованием свойств нормализованного ранга числа $\hat{r}(X)$ | 263 |
| 5.4 | Разработка методов вычисления ранга числа с использованием приближенного метода | 265 |
| 5.5 | Выводы по пятой главе | 276 |

| | |
|--|------------|
| Глава 6. РАЗРАБОТКА МЕТОДОВ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМ ОБРАБОТКИ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ДВУХУРОВНЕВОЙ RRNS | 278 |
| 6.1 Подходы к повышению надежности и безопасности обрабатываемых конфиденциальных данных | 279 |
| 6.2 Обеспечение надежности и конфиденциальности данных с использованием двухуровневой RRNS | 282 |
| 6.2.1 Одноуровневая RRNS | 284 |
| 6.2.2 Кодирование и декодирование данных в двухуровневой RRNS | 287 |
| 6.3 Двухуровневая RRNS | 288 |
| 6.3.1 Алгоритм коррекции ошибок с использованием двухуровневой RRNS | 289 |
| 6.3.2 Разработка методов и алгоритмов коррекции ошибок с использованием двухуровневой RRNS | 292 |
| 6.3.3 Корректирующие свойства двухуровневой RRNS с обратным распространением ошибки | 297 |
| 6.4 Разработка алгоритмов кодирования и декодирования в двухуровневой RRNS | 298 |
| 6.4.1 Алгоритмы кодирования 2Lbp-RRNS | 300 |
| 6.4.2 Алгоритмы декодирования 2Lbp-RRNS | 304 |
| 6.5 Анализ производительности двухуровневой RRNS | 307 |
| 6.5.1 Скорость кодирования и декодирования в двухуровневой RRNS | 308 |
| 6.5.2 Скорость обработки данных в двухуровневой RRNS | 311 |
| 6.6 Выводы по шестой главе | 313 |
| ЗАКЛЮЧЕНИЕ | 315 |
| СПИСОК ЛИТЕРАТУРЫ | 322 |
| Статьи автора в журналах, рекомендованных ВАК РФ, Scopus, Web of Science | 322 |
| Другие публикации автора по теме диссертации | 327 |
| Свидетельства о государственной регистрации программ для ЭВМ | 336 |
| Патенты на изобретения | 341 |

| | |
|---|------------|
| Цитируемая литература | 344 |
| СПИСОК РИСУНКОВ | 373 |
| СПИСОК ТАБЛИЦ | 376 |
| Приложение А. ПАТЕНТЫ | 378 |
| Приложение Б. СВИДЕТЕЛЬСТВА О ГОСУДАРСТВЕННОЙ РЕГИСТРАЦИИ ПРОГРАММ ДЛЯ ЭВМ | 390 |

ВВЕДЕНИЕ

Актуальность темы исследования. Облачные вычисления способны обеспечить значительные преимущества при организации удаленного распределенного хранения и обработки данных в виде доступности, масштабируемости, энергоэффективности, почти нулевых предварительных инвестиций в инфраструктуру, своевременного предоставления услуг и т.д. Однако, вместе с преимуществами возникают дополнительные проблемы, связанные с потерей, искажением, кражей данных [354]. Аутсорсинг данных подразумевает делегирование прямого управления данными и их обработки, что увеличивает риски кражи информации в случае недобросовестного поведения провайдера облачных услуг (Cloud Solution Provider – CSP).

Проблема надежности является критической для сохранения целостности и доступности данных в облачной среде. Разработка методов проектирования надежных сервисов, использующих распределенные системы обработки данных – важнейшее направление исследований в области аутсорсинга вычислений. Повышение эффективности необходимо для повышения качества облачных сервисов, но при этом не должно критически влиять на безопасность доверенных сервису данных. Стандартным подходом к обеспечению конфиденциальности данных является использование традиционных алгоритмов, основанных на гомоморфных вычислениях. Общая идея решения данной проблемы в контексте облачных вычислений состоит в том, чтобы делегировать обработку данных, не предоставляя к ним прозрачный доступ.

Гомоморфные вычисления, используемые, в частности, для полностью гомоморфного шифрования (Fully Homomorphic Encryption – FHE), способны решить описанную проблему [213, 215]. Гомоморфные вычисления позволяют третьей (возможно ненадежной) стороне обрабатывать закодированную информацию без раскрытия исходных данных. Гомоморфизм групп позволяет применять основные математические операции непосредственно к новой алгебраической структуре, сохраняя результаты данных операций с точностью до обратного гомоморфного преобразования. Другими словами, гомоморфные вычисления обеспечивают совместимость двух критических для аутсорсинга данных факторов: вычислений и конфиденциальности.

Основным ограничивающим фактором для построения безопасных и надежных систем обработки данных является высокая вычислительная сложность алгоритмов. Многочисленные попытки оптимизации существующих схем гомоморфных вычислений имели лишь незначительный успех и не решают указанную проблему. Требуется комплексный подход к уменьшению вычислительной сложности, включающий проработку всех этапов проектирования схемы обработки данных, начиная с построения модели, соответствующей требованиям, предъявляемым к современным распределенным вычислительным средам, включая возможность реализации механизмов обеспечения надежности и конфиденциальности обрабатываемых данных, и заканчивая разработкой эффективных алгоритмов реализации функционала схем обработки конфиденциальных данных.

Анализ современных систем распределенной обработки конфиденциальных данных, теоретических и практических исследований ведущих российских и зарубежных ученых, позволяет сделать вывод, что на данный момент проблема снижения вычислительной сложности алгоритмов обработки данных остается открытой. Таким образом, научная проблема, на решение которой направлена данная работа, заключается в разработке фундаментальных основ для проектирования систем обработки и хранения конфиденциальных данных в гетерогенных средах. Для решения поставленной общей научной проблемы проведена ее декомпозиция на ряд частных задач:

- Разработка теории сравнения зашифрованных чисел и определения их знака над различными алгебраическими структурами.
- Модификация методов контроля выполнения арифметических операций с зашифрованными данными с использованием ранга числа.
- Разработка конфигурируемой масштабируемой двухуровневой структуры доступа на основе избыточной системы остаточных классов, допускающей реализацию гомоморфных вычислений и позволяющей осуществлять параллельную обработку данных с сохранением их конфиденциальности.
- Разработка алгоритма обнаружения и исправления ошибок в двухуровневой избыточной системе остаточных классов с использованием расстояния Хемминга.

Существует множество перспективных теоретических решений, реализующих схемы гомоморфных вычислений, однако, большинство из них ориенти-

рованы на использование единого сервиса хранения и обработки данных, что сильно ограничивает предоставляемые пользователю вычислительные возможности. Для расширения вычислительных возможностей, помимо оптимизации вычислительной сложности основных операций, предлагается программно объединить вычислительные возможности различных облачных сервисов в рамках парадигмы мультиоблачного хранения и обработки данных. Схема, реализующая мультиоблачный подход, должна обеспечивать надежность и конфиденциальность хранимых и обрабатываемых данных в условиях повышенной неопределенности, связанной с использованием различных облачных сервисов, каждый из которых характеризуется динамическим изменением основных свойств и параметров. Другими словами, с объединением ресурсов различных облачных сервисов объединяются и риски, связанные с потерей данных и утратой их конфиденциальности, характерные для каждого из них, что необходимо учитывать при построении мультиоблачных моделей распределенного хранения и обработки данных.

Успешные теоретические решения в области построения безопасных и надежных распределенных систем хранения и обработки данных предложили R.L. Rivest, T. Elgamal, A. Shamir, L. Adleman, C. Gentry, Z. Brakerski, V. Vaikuntanathan, A. Badawi, S. Halevi, A. Khedr, G. Gulak, И.Я. Акушский, В.М. Амербаев, Д.И. Юдицкий, Н.И. Червяков, А.Л. Стемпковский, А.А. Коляда, В.В. Князев, В.А. Торгашев, И.Т. Пак, Л.К. Бабенко и другие авторы.

Целью исследования является разработка теоретических основ, эффективных методов и алгоритмов определения знака числа, сравнения зашифрованных чисел, кодов обнаружения и исправления ошибок данных и арифметических операций, позволяющих повысить надежность хранения и эффективность обработки конфиденциальных данных в открытых распределенных средах.

Объектом исследования являются теория обеспечения надежности и конфиденциальности данных.

Предметом исследования выступают модели и методы распределенной обработки данных с использованием гомоморфных вычислений.

Методы исследования. При решении поставленных задач использовались методы теории чисел, теории алгоритмов, теории вероятностей и математической статистики, комбинаторики, арифметики конечных полей, нейросетевых моделей над кольцом вычетов, отказоустойчивого кодирования.

Научную новизну диссертации представляют следующие основные научные результаты, расширяющие существующий базис теории и практики обработки конфиденциальных данных в распределенных средах.

- Разработана теория построения многочленов наилучшего приближения функции определения знака числа, что улучшает и расширяет известные результаты.
- Предложен метод вычисления многочленов наилучшего приближения и решена задача об их количестве.
- Разработана теория сравнения зашифрованных чисел и определения их знака над кольцом с делителями нуля.
- Выделен класс монотонных функций ядра Акушского. Решена проблема возникновения критических ядер.
- Модифицированы методы контроля выполнения арифметических операций с зашифрованными данными с использованием ранга числа.
- Разработан метод обнаружения и исправления ошибок в двухуровневом СОК с использованием расстояния Хемминга.
- Предложены оригинальные методы и алгоритмы повышения надежности и безопасности хранимых и обрабатываемых данных в распределенных средах.
- Построены многочлены, использующие интерполяционные многочлены Лагранжа, позволяющие определять знак числа и сравнивать числа над полем \mathbb{Z}_m , уточнены их степени.
- Предложена 2Lbp-RRNS конфигурируемая масштабируемая двухуровневая структура доступа на основе избыточной системе остаточных классов (ИСОК), допускающая реализацию гомоморфных вычислений и позволяющая осуществлять параллельную обработку данных с сохранением их конфиденциальности.
- Разработаны алгоритмы кодирования и декодирования данных в 2Lbp-RRNS для улучшения эффективности обработки данных в распределенных средах.

Практическая и теоретическая значимость. Работа носит теоретический характер. Полученные в ней результаты позволяют проектировать распределенные системы обработки конфиденциальных данных с использованием гомоморфных вычислений. Предложены новые модели построения подобных

систем, а также эффективные реализации вычислительно сложных операций и алгоритмов кодирования, декодирования.

Применение вышеперечисленных результатов диссертационного исследования обеспечивает повышение эффективности систем распределенной обработки конфиденциальных данных в современных распределенных вычислительных системах.

Практическая и теоретическая значимость полученных результатов и вклад диссертанта в развитие соответствующей отрасли знаний подтверждается цитированием результатов в международных изданиях: 1181 ссылка в Google Scholar (h-index = 17), 568 ссылок в Scopus (h-index = 14).

Основные положения, выносимые на защиту:

- Теория сравнения зашифрованных чисел и определения их знака над различными алгебраическими структурами.
- Методы контроля выполнения арифметических операций с зашифрованными данными с использованием ранга числа.
- Алгоритм обнаружения и исправления ошибок в двухуровневой избыточной системе остаточных классов с использованием расстояния Хемминга.
- Конфигурируемая, масштабируемая двухуровневая структура доступа на основе избыточной системы остаточных классов, допускающая реализацию гомоморфных вычислений и позволяющая осуществлять параллельную обработку данных с сохранением их конфиденциальности.
- Комплекс программ, зарегистрированных в Роспатенте РФ.

Основные результаты диссертационного исследования были использованы в рамках следующих научно-технических работ:

Министерство науки и высшего образования Российской Федерации

1. «Исследование и разработка передовых методов защиты информации, сохранения конфиденциальности и предотвращения утечки данных при обработке данных в распределенных средах» (Проект 075-15-2020-788), 2020-2022.
2. «Северо-Кавказский центр математических исследований» (Проект 075-02-2021-1749, 075-02-2022-892), 2021-2023.
3. «Фундаментальные алгоритмы, технологии глубокого обучения и безопасности для облачного хранения и обработки данных» (Проект 075-15-2021-1010), 2021.

4. «Разработка методов пространственного разделения и периодического обновления секрета на точках эллиптической кривой» (ФЦП, Проект: 14.В37.21.1128), 2012-2013.
5. «Разработка программного комплекса шифрования данных, на основе использования точек эллиптической кривой» (ФЦП, Проект: 07.Р20.11.0029), 2011.

Российский научный фонд

1. «Эффективная, безопасная и отказоустойчивая система распределенного хранения и обработки конфиденциальных данных с регулируемой избыточностью для проектирования мобильных облаков на маломощных вычислительных устройствах» (Проект: 19-71-10033, 19-71-10033-П), 2019-2024.

Российский фонд фундаментальных исследований

1. «Эффективная интеллектуальная система управления данными в краевых, туманных и облачных вычислениях с регулируемой отказоустойчивостью и безопасностью» (Проект: 20-37-51004 Научное наставничество), 2021-2022.
2. «Разработка методов и алгоритмов быстродействующего, отказоустойчивого математического сопроцессора для проектирования вычислительных систем с повышенным уровнем безопасности и низким энергопотреблением» (Проект: 20-37-70023 Стабильность), 2019-2020.
3. «Разработка новых отказоустойчивых мобильных систем связи с низким энергопотреблением на основе интеграции параллельной математики и искусственных нейронных сетей» (Проект: 18-07-00109-а), 2018-2019.
4. «Разработка и исследование концепции активной безопасности на точках эллиптической кривой» (Проект: 12-07-31087 мол_а), 2012.

Совет по грантам Президента Российской Федерации

1. «Безопасная и надежная распределенная система хранения больших данных с регулируемой избыточностью» (Проект: МК-341.2019.9), 2019-2020.
2. «Разработка методов и алгоритмов функционирования устройств для «Интернет вещей» с использованием модулярной арифметики» (Проект: СП-1215.2016.5), 2016-2018.

Достоверность и обоснованность полученных в диссертации результатов подтверждена корректным применением классических методов исследования, строгими доказательствами и анализом эффективности разработанных моделей и алгоритмов. Результаты согласуются с проведенными численными экспериментами.

Соответствие диссертации паспорту специальности. Тема и основные результаты диссертации соответствуют следующим областям исследований паспорта специальности ВАК 2.3.5 (05.13.11) – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей»:

1. Модели, методы, алгоритмы, языки и программные инструменты для организации взаимодействия программ и программных систем;
2. Модели и методы создания программ и программных систем для параллельной и распределенной обработки данных, языки и инструментальные средства параллельного программирования;
3. Модели, методы, алгоритмы и программная инфраструктура для организации глобально распределенной обработки данных.

Апробация работы. Все результаты диссертационного исследования прошли апробацию на научных мероприятиях в России и за рубежом. Выделим наиболее значимые из них.

Российские конференции:

1. International Siberian Conference on Control and Communications (SIBCON), 2015.
2. International Conference Engineering and Telecommunication (En&T), 2020, 2019, 2016, 2015, 2014.
3. Ivannikov Ispras Open Conference (ISPRAS), 2020, 2019.
4. IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), 2021, 2020, 2018, 2017, 2016.
5. International Conference «Marchuk Scientific Readings 2020», dedicated to the 95th anniversary of the birthday of RAS Academician Guri. I. Marchuk (MSR-2020), 2020.
6. International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE), 2021, 2020.
7. International Workshop on Data Mining and Knowledge Engineering (YRID), 2020.

8. International Conference Russian Supercomputing Days (RuSCDays), 2020.
9. Conference of Open Innovations Association (FRUCT), 2010.
10. International Conference Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS), 2017, 2016.
11. IEEE International Conference on Soft Computing and Measurements (SCM), 2017.
12. International Conference BOINC-Based High Performance Computing: Fundamental Research and Development (BOINC: FAST), 2017.
13. International Scientific Conference Intelligent Information Technologies for Industry (IITI), 2016.

Международные симпозиумы:

1. IEEE International Parallel and Distributed Processing Symposium Workshops, IPDPSW, 2021, 2019, 2018 (Core Rank A).
2. IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing, CCGrid, 2021 (Core Rank A).
3. International Conference on Optimization and Learning, OLA, 2021.
4. Latin American High Performance Computing Conference, CARLA, 2020, 2019, 2018, 2017.
5. International Conference on High Performance Computing and Simulation, HPCS, 2019, 2018 (Core Rank B).
6. International Workshop on Database and Expert Systems Applications, DEXA, 2017 (Core Rank B).
7. IEEE 8th International Conference on Application of Information and Communication Technologies (AICT).
8. 6th International Conference on Swarm Intelligence (ICSI) held in conjunction with the 2nd BRICS Congress on Computational Intelligence (CCI).

Публикации. По теме диссертации автором было опубликовано 89 статей, в том числе 36 статей в журналах из списка, рекомендованного ВАК, или индексируемых в международных базах Scopus и/или Web of Science [1–36], 53 работы – в сборниках трудов российских и международных конференций [37–89], получено 26 свидетельств о государственной регистрации программ для ЭВМ [90–115] и 12 патентов на изобретения [116–127].

Личный вклад автора. Диссертационная работа представляет собой многолетнее исследование автора, объединенное тематикой и методами исследования. Все выносимые на защиту результаты получены лично автором. Из совместных работ в диссертацию включены только те результаты, которые принадлежат непосредственно автору. В опубликованных совместных работах постановка и решение задач осуществлялись совместными усилиями соавторов при непосредственном участии соискателя. В статьях [1, 2, 4–6, 8, 10, 11, 15, 18, 29, 33, 39–41, 44–46, 48, 49, 52, 55, 57, 58, 61, 65, 69, 74, 77, 79–82, 84, 85, 88, 89] автором разработаны модели, методы и алгоритмы повышения надежности и безопасности распределенных систем хранения и обработки данных. В статьях [8, 11, 21] автором методы обнаружения, локализации и исправления ошибок в RNS. В статьях [4, 7, 9, 12, 14, 16, 20, 24, 26, 28, 31, 32, 42, 44, 48, 50, 51, 54, 56, 59, 60, 62–64, 67, 83] автором разработаны методы уменьшения вычислительной сложности алгоритмов выполнения операций определения знака, сравнения, деления закодированных чисел. В статьях [3, 4, 7–24, 26–31, 34–36, 39, 41, 42, 45, 47–55, 57–62, 68–87] автором исследованы свойства существующих схем, предложены и реализованы механизмы повышения их эффективности. В статьях [43, 56] автором разработаны методы обнаружения и исправления ошибок арифметических операций основанные на использовании свойств ранга числа. В статьях [25, 29, 33, 37, 38, 59, 63, 66] автором классифицированы существующие схемы и предложена структурная математическая модель обработки данных в распределенных средах. В статьях [7, 8, 23, 24, 26, 29, 59, 61, 69, 82] автором опубликованы результаты моделирования существующих и предложенных схем, полученные на основе разработанного комплекса программ.

Структура работы. Диссертация состоит из введения, 6 глав, заключения, библиографии из 378 наименований и 2 приложений. Общий объем основного текста работы – 321 страница, включая 35 таблиц и 43 рисунка.

Краткое содержание работы. В первой главе представлен проблемный обзор угроз информационной безопасности в современных распределенных средах хранения и обработки данных. Распределенные системы характеризуются высоким уровнем неопределенности, связанной с нестационарностью и динамическим изменением количества и состава их узлов и компонентов, что отрицательно влияет на эффективность вычислений, создавая дополнительные трудности в решении проблем планирования. Таким образом, требуется раз-

работка новых стратегий управления ресурсами для эффективного решения проблемы неопределенности.

В рамках сформулированной цели исследования построена структурная модель обработки данных в распределенных средах. Выделено пять уровней передачи, хранения и обработки данных. Для каждого из уровней выявлены основные угрозы безопасности данных и проанализированы современные методы уменьшения вероятности кражи, потери или искажения данных. Установлено, что в распределенных средах в условиях повышенной неопределенности фундаментальные подходы к снижению рисков конфиденциальности, целостности и доступности, использующие механизмы репликации данных, резервного копирования, структуры доступа, избыточную систему остаточных классов, коды стирания недостаточно эффективны и должны быть усовершенствованы. Предложено использование вышеперечисленных механизмов, адаптированных, оптимизированных и интегрированных в концепцию мультиоблачного хранения и обработки данных. Использование мультиоблачного подхода позволяет существенно повысить надежность распределенных систем и снизить вероятности потери, утечки информации, отказа в доступе в течение длительного времени.

Модель, наиболее адекватную мультиоблачному подходу с точки зрения организации распределенного хранения и обработки данных, реализуют пороговые структуры доступа. Однако, выбор оптимальной структуры доступа представляет собой сложную многокритериальную задачу, т.к. должен осуществляться не только на основе стандартных метрик, таких как сложность, скорость выполнения и т.д., но и учитывать особенности распределенной среды, связанные, в первую очередь, с высоким уровнем неопределенности. В работе приведено обоснование выбора алгоритмов реализации пороговых структур доступа с точки зрения обеспечения безопасности, надежности хранения, возможности осуществления контроля корректности операций с данными, гомоморфных вычислений и вводимой избыточности. Представлена модификация алгоритма реализации пороговой структуры доступа с учетом предложенных методов выполнения основных операций. Разработанные методы всесторонне протестированы, доказана и продемонстрирована их эффективность.

Возможность реализации гомоморфных вычислений является наиболее существенным аспектом при выборе параметров структуры доступа, т.к. помимо возможности распределенной обработки гомоморфные вычисления позволяют обеспечить безопасность обрабатываемых данных за счет обработки в закодиро-

ванном виде. Различают два подхода к реализации гомоморфных вычислений: над кольцом вычетов с делителями нуля и над полем.

Алгоритмы гомоморфных вычислений над кольцом вычетов с делителями нуля могут быть использованы при построении защищенной системы обработки данных, но при этом необходимо учитывать высокую вероятность взлома системы с помощью модифицированной атаки открытым текстом. Данная проблема, наряду с угрозой сговора облаков при использовании мультиоблачного подхода, является основной и успешно решается в рамках данного исследования. Алгоритмы гомоморфных вычислений над полем входят в проект стандарта по гомоморфным вычислениям от 2018 года и могут быть классифицированы либо как целочисленные, либо как вещественные, в зависимости от типа входных данных.

Существующие схемы, построенные с использованием гомоморфизма колец, позволяют выполнять арифметические операции сложения и умножения закодированных чисел. В зависимости от применяемой схемы гомоморфных вычислений меняются подходы к выполнению указанных операций. Однако, общей проблемой гомоморфных вычислений, независимо от используемого подхода (гомоморфные вычисления над кольцом вычетов с делителями нуля или над полем) и вида применяемых схем, является высокая сложность реализации и, как следствие, низкая скорость обработки данных. Наибольшие задержки наблюдаются при выполнении вычислительно сложных операций, к ним относятся операции определения знака числа и сравнения чисел. Эффективность выполнения указанных операций можно повысить путем разработки новых методов и оптимизации соответствующих алгоритмов вычисления приближенного (с необходимой точностью) или точного (когда это возможно) значения результата данных операций с сохранением свойства гомоморфности. Повышение эффективности вычисления результатов проблемных для гомоморфных вычислений операций равносильно ускорению процедуры кодирования/декодирования в целом, поэтому разработке методов выполнения операций определения знака числа и сравнения чисел уделено особое внимание в данном исследовании.

Вторая глава посвящена построению высокопроизводительной вычислительно стойкой структуры доступа, обладающей свойствами гомоморфизма колец, и обеспечивающей высокий уровень безопасности и надежности в нестационарной облачной среде. Предложена адаптивная распределенная служба хранения под названием WA-MRC-RRNS, которая реализует гомоморфное отобра-

жение и сочетает в себе функционал взвешенной пороговой структуры доступа и системы контроля корректности результатов обработки данных.

Использование взвешенной пороговой структуры доступа обусловлено доказанной теоремой о том, что вероятность потери данных при использовании взвешенной пороговой структуры доступа не превышает вероятности потери данных при использовании соответствующей классической пороговой структуры доступа. Показано, что в пессимистическом сценарии при настройке (3,4), вероятность потери данных при использовании WA-MRC-RRNS в 777.02 раза ниже, чем при использовании соответствующей классической пороговой структуры доступа MRC-RRNS. В среднем же вероятность потери данных при использовании WA-MRC-RRNS ниже в $9.23 \cdot 10^{17}$ раза.

Выбор избыточной системы остаточных классов (RRNS) в качестве основы для предложенной взвешенной пороговой структуры доступа обусловлен возможностью построения вычислительно стойкой схемы и реализации механизмов обнаружения/восстановления множественных ошибок данных. Кроме того, RRNS позволяет динамически настраивать параметры, чтобы справиться с различными объективными предпочтениями, рабочими нагрузками и свойствами облака.

Высокая производительность предложенной схемы достигается за счет разработанных алгоритмов кодирования/декодирования, основанных на переходе к представлению в обобщенной позиционной системе счисления (MRC), нейронной сети конечного кольца и их эффективной программной реализации. Сравнение предложенной схемы WA-MRC-RRNS с другой известной взвешенной схемой WA-AR-RRNS с точки зрения производительности дало следующие результаты: при кодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 13.73 раза, при декодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 385.07 раза. Отметим, что предложенная схема WA-MRC-RRNS так же превосходит классическую пороговую схему AR-RRNS с точки зрения производительности (в 4.83 раза при кодировании и в 120.04 раза при декодировании), проигрывая лишь классической пороговой схеме MRC-RRNS в 2.42 раза при кодировании и в 1.16 раза при декодировании. Данные потери в производительности абсолютно оправданы многократным повышением надежности и безопасности, достигаемым за счет использования взвешенной схемы WA-MRC-RRNS вместо классической пороговой схемы MRC-RRNS.

WA-MRC-RRNS – адаптивная схема, позволяющая динамически регулировать настройки (n_v, K, N) , чтобы справиться с отключениями, сбоями и изменением характеристик и параметров облачных сервисов. Настройки должны определяться экспериментально на основе накопленных статистических показателей. Статистический интервал времени должен быть установлен в соответствии с динамикой нестационарной среды и конфигурациями системы. Решение данных задач выходит за рамки данной работы и является предметом будущих исследований.

Для анализа предложенной схемы WA-MRC-RRNS с точки зрения безопасности данных, доказано утверждение, дающее оценку вероятности получения несанкционированного доступа к данным. Приведены вероятности получения несанкционированного доступа к данным для каждого из трех основных сценариев сговора: когда противоборствующая коалиция знает секретный ключ и не знает необходимое количество долей, не знает ни секретного ключа ни необходимого количества долей, а также не знает секретного ключа и знает необходимое количество долей. Для обеспечения безопасности данных предложено интегрировать WA-MRC-RRNS в разработанную конфигурируемую схему хранения данных AC-RRNS. Доказана вычислительная безопасность AC-RRNS. Сравнительный анализ предложенной схемы с известными структурами доступа, использующими аппарат RRNS, такими как схема HORNS, основанная на схеме Mignotte, и схема Asmuth-Bloom, дал следующие результаты: HORNS обладает меньшей избыточностью, но в отличие от предложенной схемы, не является вычислительно безопасной, уязвима для атаки открытым текстом и не может быть использована для решения проблемы сговора; схема Asmuth-Bloom является асимптотически идеальной, подходит для обеспечения безопасности данных при сговоре, но вводит избыточность в k раз превышающую избыточность предложенной схемы (k – параметр схемы Asmuth-Bloom). Кроме того, использование AC-RRNS многократно снижает вероятность неавторизованного доступа к данным коалиции из k злоумышленников по сравнению со схемами HORNS и Asmuth-Bloom.

Таким образом, предложенная схема превосходит ранее разработанные аналоги по многим параметрам и соответствует требованиям, предъявляемым к гомоморфным кодам, используемым в распределенных средах хранения и обработки данных.

В третьей главе исследованы различные подходы к выполнению операций определения знака и сравнения чисел, разработаны методы и алгоритмы реализации указанных операций, позволяющие повысить производительность гомоморфных вычислений над кольцом вычетов с делителями нуля.

Установлено, что результаты операций определения знака числа и сравнения чисел, заданных над кольцом вычетов с делителями нуля, невозможно вычислить с помощью многочленов.

Разработаны два алгоритма, реализующие определение знака числа для гомоморфных вычислений над кольцом вычетов с делителями нуля, основанных на RNS с четным и нечетным диапазоном.

Представлен обзор существующих и предложены новые методы сравнения чисел для гомоморфных вычислений над кольцом вычетов с делителями нуля на основе RNS.

Методы, основанные на переводе чисел из RNS в позиционную систему счисления, являются наиболее очевидными и наименее производительными способами сравнения чисел. С целью повышения производительности разработаны методы сравнения, основанные на вычислении позиционных характеристик, таких как диагональная функция, функция ядра Акушского, функция Pirlo и Impedovo. Все вышеперечисленные позиционные характеристики, кроме функции ядра Акушского, являются монотонными, и возможна ситуация, когда разные числа имеют одинаковую позиционную характеристику. В этом случае требуется выполнение дополнительных действий для сравнения чисел, представленных в RNS. Кроме того, для получения значений указанных позиционных характеристик используется ресурсозатратная операция вычисления остатка от деления на большой модуль. Для устранения этих недостатков введено понятие модифицированной диагональной функции, которое служит теоретической основой для разработки значительно более быстрого алгоритма сравнения. Модифицированная диагональная функция (MDF) является строго возрастающей и сочетает в себе преимущества диагональной функции и приближенного метода. Строгая монотонность MDF обеспечивает взаимоднозначное соответствие числа и его позиционной характеристики, поэтому не возникает ситуаций, когда требуется выполнение дополнительных действий для сравнения чисел. Кроме того, вместо операции нахождения остатка от деления на большое число, при вычислении MDF используются значительно более простые в реализации вычисления по модулю, равному степени числа 2.

Разработанное устройство сравнения на основе MDF и его наиболее эффективные известные аналоги, применяемые для сравнения чисел в RNS с модулями общего вида, были синтезированы для технологии 65 нм с использованием нескольких образцов наборов модулей. Согласно полученным оценкам производительности, предложенный подход обеспечивает снижение задержки на 11 – 75% (в зависимости от набора модулей) по сравнению с самыми быстрыми существующими реализациями известных методов сравнения чисел в RNS. Более того, наблюдается снижение аппаратных затрат (более чем на 41%) и значительное снижение энергопотребления, которое в ряде случаев превышает 100%. Таким образом, предложенный метод на основе MDF позволяет реализовывать наиболее эффективные на сегодняшний день устройства сравнения чисел, представленных в RNS с наборами модулей общего вида.

Особого внимания заслуживают функции ядра Акушского, свойства которой зависят от используемых при ее построении коэффициентов. Доказано, что для достижения монотонности при построении функции ядра необходимо использовать только неотрицательные коэффициенты. Показано, что уже известная диагональная функция, ранее предложенная для реализации сравнения чисел в RNS, есть не что иное, как частный случай функции ядра со всеми коэффициентами равными единице. Сформулированы условия, при которых обеспечивается минимальный диапазон функции ядра (необходимый для получения наилучших характеристик устройства сравнения чисел в RNS). Установлено, что монотонная функция ядра минимального диапазона (MMCF) имеет только один коэффициент, равный единице (соответствующий наибольшему модулю), все остальные коэффициенты равны нулю. Сформулирована и доказана теорема об условиях отсутствия критических ядер функции ядра Акушского, имеющая важное практическое значение для построения эффективных позиционных характеристик чисел, представленных в RNS. Представленное исследование позволяет сделать вывод, что функция ядра Акушского является обобщением позиционных характеристик чисел, представленных в RNS, ее изучение углубит понимание свойств позиционных характеристик и, следовательно, позволит разрабатывать более высокопроизводительные подходы и методы реализации операций над закодированными числами.

В четвертой главе исследованы различные подходы к выполнению операций определения знака и сравнения чисел, оптимизированы известные и разработаны новые методы и алгоритмы реализации указанных операций, позволя-

ющие повысить их эффективность в контексте гомоморфных вычислений над полем.

Гомоморфные вычисления над полем принято делить на два класса: целочисленные и вещественные, по формату обрабатываемых цифровых данных. Соответственно задачи определения знака числа и сравнения чисел следует рассматривать над полем \mathbb{Z}_m и над полем \mathbb{R} .

Построены многочлены, использующие интерполяционные многочлены Лагранжа, позволяющие определять знак числа и сравнивать числа над полем \mathbb{Z}_m . Вычислительная сложность алгоритмов определения знака и сравнения чисел при использовании целочисленных гомоморфных вычислений зависит от количества арифметических операций сложения и умножения, которые необходимо выполнить для вычисления интерполяционного многочлена. Целочисленные гомоморфные вычисления поддерживают ограниченное количество умножений, поэтому от мультипликативной глубины алгоритма, реализующего ту или иную операцию, во многом зависит производительность системы. Мультипликативная глубина вычисления многочлена зависит от его степени, что было показано в работе [203] на примере алгоритма Paterson–Stockmeyer. Доказана теорема, дающая оценку степени интерполяционного многочлена функции определения знака числа: показано, что степень многочлена равна $m - 2$. Доказана теорема, уточняющая оценку степени интерполяционного многочлена функции сравнения чисел: показано, что степень многочлена из работы [199], равная $2m - 2$, может быть уточнена до m .

Для аппроксимации функции определения знака числа над полем \mathbb{R} исследована проблема построения многочлена наилучшего приближения указанной функции. Показано, что если степень аппроксимирующего многочлена $n = 0$, то многочленами наилучшего приближения являются $Q_n(x) = a_0$, где $|a_0| \leq 1$. Доказано, что если степень аппроксимирующего многочлена $n \geq 1$, то не существует многочленов наилучшего приближения, являющихся четными функциями. Если степень аппроксимирующего многочлена $n \geq 1$, то существует единственный многочлен наилучшего приближения, являющийся нечетной функцией, который строится с помощью интерполяционной формулы Лагранжа, где в качестве узлов интерполяции используются нули многочлена Чебышева второго рода. Доказано, что если $n \geq 1$ и n – нечетное число, то не существует многочленов наилучшего приближения, являющихся функциями общего вида. Если $n \geq 1$ и n – четное число, то существует несчетное множество много-

членов наилучшего приближения, являющихся функциями общего вида. Для каждого рассмотренного случая построены аппроксимирующие многочлены и доказано, что каждый из них является многочленом наилучшего приближения. Для случаев, когда многочлена наилучшего приближения не существует, так же доказаны соответствующие теоремы.

Предложен модифицированный нейросетевой метод определения знака числа над полем \mathbb{R} , позволяющий более чем в 15.1 раза повысить точность указанной операции в окрестности проблемной точки $x = 0$.

Пятая глава посвящена разработке высокопроизводительных методов и алгоритмов вычисления ранга чисел, представленных в RNS. Основным приложением функции ранга числа, представленного в RNS, являются алгоритмы обнаружения и исправления ошибок арифметических вычислений, и от эффективности его вычисления во многом зависит производительность указанных алгоритмов.

Рассмотрены три формы ранга числа: классическая форма ранга, следующая из Китайской теоремы об остатках, нормализованный ранг числа и ранг числа, построенный с использованием функции ядра Акушского. Исследован вопрос об интерполяции функции ранга числа с помощью алгебраических многочленов. Доказан ряд теорем, позволяющих утверждать, что не существует многочлена, заданного над \mathbb{Z}_p , позволяющего вычислить ранг числа, представленного в RNS, вне зависимости от его формы.

Предложен эффективный метод вычисления ранга числа, основанный на использовании функции ядра Акушского, не содержащей критических ядер. Доказаны теоремы, дающие оценку верхней и нижней границ разрядности констант при использовании приближенного метода для вычисления ранга числа. Показано, что наборы модулей, удовлетворяющие полученной оценке, не являются компактной последовательностью. Предложенный метод позволяет сократить объем необходимых вычислений и увеличить скорость вычисления ранга числа по сравнению с приближенным методом: для нахождения ранга числа с использованием приближенного метода необходимо выполнить n операций с числами, превышающими значение модуля, тогда как в предлагаемом методе необходимо выполнить $\frac{n(n-1)}{2}$ операций с числами, не превышающими значение модуля.

Разработаны алгоритмы вычисления ранга числа, представленного в RNS. Доказаны теоремы, позволяющие осуществлять контроль результатов обработ-

ки закодированных чисел с использованием арифметических свойств классического и нормализованного рангов.

В шестой главе представлена конфигурируемая масштабируемая двухуровневая пороговая структура доступа на основе RRNS (2Lbp-RRNS), разработанная для надежного и безопасного хранения данных в мультиоблачных системах. Предложенная структура допускает использование гомоморфных вычислений и позволяет осуществлять параллельную обработку данных с сохранением их безопасности. Разработанная схема 2Lbp-RRNS является расширением классической схемы 2L-RRNS. Высокая эффективность и производительность 2Lbp-RRNS достигается за счет использования расстояния Хэмминга.

Получена верхняя граница для количества обнаруживаемых и исправляемых ошибок при использовании традиционных пороговых двухуровневых схем 2L-RRNS и предложенных пороговых двухуровневых схем 2Lbp-RRNS. Показано, что предложенная схема 2Lbp-RRNS обладает лучшими корректирующими свойствами по сравнению с традиционной схемой 2L-RRNS, позволяет обнаруживать в среднем в 1.58 раза и исправлять в среднем в 3.37 раза больше ошибок.

Предложены эффективные реализации алгоритмов кодирования и декодирования данных в 2Lbp-RRNS: эффективность при кодировании достигается за счет использования метода Паскаля и нейронной сети конечного кольца (FRNN), эффективность декодирования обусловлена использованием перехода к представлению в обобщенной позиционной системе счисления (MRC), FRNN и сверточной нейронной сети (CNN).

Отметим, что при кодировании/декодировании данных в 2Lbp-RRNS, для реализации обратного преобразования вариативно может быть использован один из алгоритмов: Mignotte (основанный на Китайской теореме об остатках), MRC8 или MRC16 (основанные на переходе к обобщенной позиционной системе счисления и отличающиеся лишь размером окна, 8 или 16 бит, при реализации FRNN). Для определения наиболее оптимального из перечисленных алгоритмов, выполнен сравнительный анализ производительности схем 2Lbp-RRNS, использующих указанные алгоритмы, учитывающий полный цикл хранения данных: кодирование-загрузка и выгрузка-декодирование для различных параметров облачных хранилищ. Результаты анализа показали, что производительность MRC16 при кодировании-загрузке колеблется в диапазоне 0.406-0.837 МБ/с, а при выгрузке-декодировании в диапазоне 0.54-1.093 МБ/с

в зависимости от параметров хранилищ. Для сравнения, показатели наиболее близкого по производительности алгоритма Mignotte при кодировании-загрузке колеблются в диапазоне 0.13-0.257 МБ/с, а при выгрузке-декодировании в диапазоне 0.16-0.292 МБ/с. Таким образом, MRC16 является более сбалансированным и быстрым алгоритмом, превосходящим MRC8 и Mignotte. Также показано преимущество в производительности, достигаемое за счет параллельной реализации предложенной схемы. Все экспериментальные данные получены для восьми реальных облачных хранилищ.

Глава 1. СОВРЕМЕННЫЕ МЕТОДЫ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ ОБРАБОТКИ ДАННЫХ

Управление информацией во многих областях, таких как умный город, умная промышленность, умная медицина и т.д., прочно связано с туманными-граничными-облачными вычислениями в рамках концепции Интернет вещей (Internet of Things – IoT). Концепция заключается в использовании существующих вычислительных ресурсов пространственно распределенных устройств, таких как датчики, измерительные устройства, устройства для мониторинга промышленного оборудования, игровых консолей, смартфонов, бытовых приборов, маршрутизаторов, коммутаторов и т.д., на всем пути передачи данных из физического мира в облако.

Интернет вещей создает наборы связанных и интегрированных в глобальную среду объектов. Полная взаимосвязь между устройствами важна для эффективного использования данных, их комплексного анализа, предоставления информационных услуг и оптимизации обработки и хранения данных. Эксперты рассматривают IoT как одну из ключевых информационных технологий будущего, ориентированных на эффективное выполнение многих функций.

1.1 Классификация угроз информационной безопасности

В литературе обсуждаются многие аспекты IoT, такие как методологические взгляды, технологические аспекты, коммуникационные платформы, ключевые алгоритмы, IT-услуги, интеграция устройств и т.д.

Несмотря на многие преимущества Интернета вещей, существует множество рисков, связанных с функциональностью, безопасностью и надежностью. Выделим некоторые из них.

1. *Граничная угроза:* физические повреждения, повышение привилегий, манипулирование услугами, мошенническая инфраструктура, потеря данных, уязвимости системы и приложений, кража, проблемы с общим доступом и т.д.

2. *Угрозы сети*: небезопасные API-интерфейсы, перехват конфиденциальной информации (подслушивание), прерывание законной передачи (глушение), фальсификация MAC-адреса (подмена MAC-адреса), кража MAC-идентификатора легитимного пользователя, имитация пары взаимодействующих узлов, внедрение устаревших сетевых команд и пакетов, фальсификация IP-адреса, имитация IP-адреса законного пользователя, Smurf-атака, TCP-флуд, UDP-флуд, подделка пакетов данных легитимного пользователя, вредоносный код, скрипты, активный контент и т.д.
3. *Угрозы инфраструктуры*: отказ в обслуживании, неправомерное использование ресурсов, утечка конфиденциальных данных, повышение привилегий, манипуляции с виртуальными машинами, проблемы контроля доступа, захват учетной записи, утечки данных, злонамеренный инсайдинг и т.д.
4. *Угрозы хранилища данных*: отказ в обслуживании, потеря данных, деструктивная сила, ошибки диска, несанкционированный доступ, изменение данных, сговор, кибератака и т.д.
5. *Угрозы окружающей среды*: землетрясения, наводнения, пожары и т.д.
6. *Умышленные угрозы*: перехват, хакерские атаки и т.д.
7. *Случайные угрозы*: ошибки ПК, вирусы, спам и т.д.
8. *Некомпетентность*: ошибки пользователей, невнимательность, любопытство, фальсификация, привилегированные инсайдеры и т.д.

Несмотря на обширные исследования проблем безопасности и надежности для различных активов туманных-граничных-облачных вычислений, безопасность и надежность в нестационарной среде остаются наиболее сложными проблемами. Для их решения требуется отказ от привычных вычислительных парадигм, адаптация текущих моделей Интернета вещей к нестационарной среде и разработка новых стратегий управления ресурсами.

Потеря и искажение данных являются следствием неспособности системы исправить ошибки. Ошибками также называют случаи получения данных неавторизованными пользователями.

Ошибки могут быть вызваны множеством факторов, приводящих к сбоям различных типов. Можно выделить следующие основные модели сбоев, характерные для IoT.

1. *Искажение и повреждение данных.* В процессе обработки, передачи и хранения данные могут быть изменены из-за несовершенного программного обеспечения и оборудования.
2. *Полная или частичная потеря данных.* Вероятность банкротства, природных и техногенных катастроф, вирусных атак и т.д. невелика, но приводит к полной или частичной потере данных. Например, из-за банкротства поставщика облачных услуг Nirvanix в 2011 году пользователи потеряли терабайты хранимых данных из-за отсутствия технической возможности перемещать данные к другому поставщику облачных хранилищ (Cloud Solution Provider – CSP) [211]. Из-за ошибки системного администратора в 2016 году было удалено безвозвратно 300 ГБ данных GitLab. Специалисты GitLab отметили, что в этом случае ни одна из пяти существующих систем резервного копирования не помогла восстановить данные [344].
3. *Краткосрочный или долгосрочный отказ в доступе.* При использовании Интернета вещей возрастает вероятность отказа в доступе. Это связано с повышенной уязвимостью Интернета вещей для DDoS-атак (Distributed Denial of Service) [264]. С повсеместным внедрением умной бытовой техники вероятность заражения вирусами для построения зомби-сетей значительно увеличивается. В настоящее время отсутствуют эффективные механизмы противодействия построению зомби-сетей из умной бытовой техники, т.к. технические возможности умных бытовых приборов не удовлетворяют техническим требованиям антивирусных приложений. Один из способов уменьшить вероятность отказа в доступе – использовать парадигму мультиоблака, которая позволяет пользователям получать доступ к данным из доступных CSP. В отчете о безопасности Kaspersky [264] представлена оценка ущерба, который может быть нанесен существующим интернет-службам в результате DDoS-атак.
4. *Неисправности API.* Облака предоставляют клиентам набор пользовательских интерфейсов (User Interface – UI) и API (Application Programming Interface – прикладной программный интерфейс) для управления облачными сервисами и взаимодействия с ними. Безопасность и доступность сервисов во многом зависят от безопасности этих API. Поскольку API используются в качестве "входной двери очень ве-

роятно, что они будут постоянно подвергаться атакам. Следовательно, требуется адекватная защита [354]. API интерфейсы должны защищать от случайных и злонамеренных попыток обойти политику безопасности. Плохо спроектированные, поврежденные, открытые и взломанные API могут стать причиной утечки данных.

5. *Ошибки дисковых накопителей.* Ошибки дисковых накопителей являются наиболее частой причиной сбоев облачных сервисов, поэтому для проектировщиков очень важно адекватно прогнозировать надежность и создавать механизмы для компенсации возможных потерь данных. Отказы, связанные с головками дисков, материалом носителя, температурой, влажностью, качеством воздуха, рабочими циклами, нагрузкой ввода-вывода и т.д., могут существенно повлиять как на надежность, так и на производительность жестких дисков. Для снижения негативного эффекта данных ошибок используется постоянный мониторинг состояния дисковых накопителей с последующей заменой вышедших из строя жестких дисков [314].
6. *Нарушение целостности.* Гарантия высокой доступности и масштабируемости безусловно привлекательный для пользователей фактор, однако, с повышением доступности и масштабируемости возрастают риски нарушения конфиденциальности и целостности данных. При этом в качестве нарушителей могут выступать как внешние, так и внутренние злоумышленники [278]. Облачный сервис не обязан уведомлять о повреждении и утечке данных, поэтому проверка целостности данных остается задачей пользователя. Существующие решения для верификации увеличивают накладные расходы на вычисления и хранение, и снижают пропускную способность, что приводит к увеличению общей стоимости эксплуатации облачного сервиса [307, 378].

Существует множество механизмов повышения надежности, минимизации рисков потери, искажения и нарушения безопасности данных: репликация, коды исправления ошибок, структуры доступа и т.д. [11, 142, 249, 292].

Репликация помогает повысить надежность при хранении и обработке данных, но приводит к высокой избыточности. Безусловное преимущество репликации – простота реализации. Однако, из всех вышеперечисленных подходов к повышению надежности, репликация является самым избыточным и требует дополнительных криптографических примитивов для обеспечения безопасно-

сти. Это приводит к увеличению стоимости хранения данных, дополнительной нагрузке, связанной с шифрованием, дешифрованием, передачей ключей шифрования и т.д. и, в конечном счете, к увеличению эксплуатационных затрат.

Butler [171] показал, что облачные провайдеры предлагают услуги с разной степенью надежности, т.к. повышение надежности сопряжено с дополнительными затратами. Поэтому при реализации концепции мультиоблачного хранения рекомендуется, чтобы объем данных, хранящихся у каждого поставщика облачных услуг, был пропорционален надежности облака.

Текущие решения Интернета вещей используют значительное количество датчиков, собирающих и передающих данные, измерительных устройств, вычислительных устройств, игровых консолей, смартфонов, устройств со встроенными процессорами, запускающими приложения и т.д. Умные вещи выполняют домашние функции, используются в сфере развлечений, здравоохранении, бытовой технике, управлении приложениями, промышленных устройствах и т.д.

Продолжаются исследования в области обеспечения конфиденциальности, безопасности и надежности, а также оценки рисков, слабых мест и требований к среде IoT. Данная среда характеризуется сильной взаимосвязанностью, например, данные, используемые бытовой техникой, подключенной к Интернету, можно использовать для составления досье на конкретного человека [291]. Для Интернета вещей характерны повышенные риски утечки конфиденциальных данных и DDoS-атак. Согласно анализу DDoS-атаки, произошедшей 21 октября 2016 г., она стала возможной благодаря использованию большого количества умной бытовой техники [304]. С момента появления концепции Интернета вещей в конце 1990-х годов эксперты по безопасности предупреждали об увеличении вероятности утечки данных и угрозах конфиденциальности для большого количества незащищенных устройств, подключенных к Интернету вещей. В декабре 2013 года компания Proofpoint, занимающаяся корпоративной безопасностью, обнаружила первую сеть ботов Интернета вещей, в которой более 25% ботов работали на устройствах, отличных от компьютеров. Среди них были смарт-телевизоры, детские мониторы и другая бытовая техника.

Обмен данными осуществляется по локальной беспроводной сети или через Интернет. Беспроводная сеть позволяет Интернету вещей управлять удаленными устройствами (лампочки, чайник, видеочамера и т.д.) без установки дополнительной коммуникационной инфраструктуры, но при этом предъявля-

ются повышенные требования для обеспечения надлежащего уровня безопасности.

Энергоэффективность криптографических алгоритмов – важная актуальная проблема для умных устройств, таких как RFID-метки или бесконтактные смарт-карты. Согласно стандарту ISO/IEC [236] пассивные RFID-метки должны иметь уровень энергопотребления не более 15 мкВт, чтобы гарантировать работу устройства в радиусе 1 м. Данный уровень энергопотребления сильно ограничивает выбор алгоритмов обеспечения безопасности.

Требования к алгоритмам для сред с низким уровнем ресурсов были определены в рамках международного стандарта информационных технологий - методов безопасности – облегченной криптографии (ISO/IEC FDIS 29192). Он определяет характеристики криптографических алгоритмов для сред с низким уровнем ресурсов, позволяющие обеспечить аутентификацию, идентификацию, доступ, обмен ключами и конфиденциальность данных. Важными требованиями, предъявляемыми к криптографическим алгоритмам, являются сложность и скорость. Последняя зависит не только от частоты процессора, но и от количества ядер, поскольку криптографические примитивы обычно очень хорошо распараллеливаются.

Сервисы безопасности могут быть реализованы с помощью комбинации криптографических механизмов, таких как блочные шифры, хэш-функции, алгоритмы цифровой подписи, и некриптографических механизмов.

Существуют различные варианты эффективной реализации защиты данных в средах с низким уровнем ресурсов: расширенный стандарт шифрования (Advanced Encryption Standard – AES), система остаточных классов (Residue Number System – RNS), облегченная криптография, проактивная безопасность и т.д.

Самая быстрая аппаратная реализация алгоритма AES с разворачиванием цикла и технологиями внешней циклической конвейерной обработки имеет пропускную способность от 30 до 70 Гбит/с. Для этого требуется более 250 000 GE [236]. Самая компактная реализация архитектуры 8-b AES обеспечивает функциональность как шифрования, так и дешифрования и занимает около 2645 GE с задержкой 226 циклов [155].

Программная реализация AES – битовое шифрование AES на графическом процессоре с поддержкой CUDA обеспечивает скорость 605,9 Гбит/с на Nvidia Tesla [293].

Аппаратно-программная реализация AES используется в семействе процессоров Intel в виде нового набора инструкций шифрования для поддержки AES. Аналогичное расширение генератора PadLock есть в микропроцессорах от VIA Technologies. Идея этого расширения – ускорить шифрование данных [315].

AES требует больших вычислительных ресурсов для защиты передачи данных по беспроводным сетям. Он используется для шифрования данных в граничном узле и дешифрования данных на туманном узле. Это критично, особенно когда туманный узел обрабатывает данные, полученные от множества умных вещей. Для парадигмы Интернета вещей предлагаются модели агрегирования данных с нескольких устройств или датчиков [29]. Альтернативный способ заключается в использовании других примитивов для обеспечения безопасности, корректности и надежности данных. Схема Mignotte и RNS удовлетворяют требованиям криптографического стандарта для сред с низким уровнем ресурсов.

RNS позволяет эффективно реализовывать криптографические алгоритмы с открытым ключом: RSA [154, 333], эллиптические кривые [150, 228, 321], XTR [15] и т.д. Более того, RNS, в дополнение к структурам доступа, обнаружению, локализации и исправлению ошибок, допускает реализацию гомоморфных вычислений [214, 299].

Важным механизмом обеспечения безопасности данных в мире интеллектуальных вещей является так называемая мало ресурсная криптография (Light Weight Cryptography – LWC). Одними из первых исследований в этой области являются работы [332, 340], в которых определены технические требования к LWC. Рабочие группы Интеллектуальной Сетевой Архитектуры и Компьютерной Безопасности Национального Института Стандартов и Технологий, включая группу Совместимости Интеллектуальных Сетей и группу Кибербезопасности, обеспокоены необходимостью проведения исследований в области LWC. Основная задача – разработать и реализовать криптографическую защиту миллионов устройств с ограниченными технологическими и вычислительными ресурсами. Из-за условий эксплуатации и реализуемого функционала, а также из-за удешевления цены, эти устройства имеют существенные ограничения по памяти, вычислительной мощности, энергопотреблению и т.д.

Эксперты по безопасности предупреждают о потенциальных опасностях подключения устройств Интернета вещей напрямую к облачным сервисам. Желательно использовать туманные узлы, которые создают безопасную подсеть

и уменьшают количество энергии, необходимое для передачи данных в облако. Использование туманных узлов снижает риски построения ботнет-сети с использованием устройств Интернета вещей. Например, согласно отчету о безопасности в 2013 году, устройства Интернета вещей (смарт-телевизоры, радионяня и другие бытовые приборы) использовались для создания ботнет-сети. В случае, когда вычислительные задачи не могут быть выполнены из-за технических ограничений граничных и туманных устройств, и/или если обработка требует большего потребления энергии, чем передача данных по сети, данные перемещаются в облака.

1.2 Структурная модель распределенной обработки данных в виртуальных средах

Существует несколько уровней нестационарности, связанных с Интернетом вещей: сенсор, обработка данных, обмен данными, хранение и т.д. Динамичность параметров и эластичность решений Интернета вещей положительно влияют на качество обслуживания; однако, они добавляют значительную неопределенность на различных уровнях вычислений, обмена данными и хранения. Нарушение безопасности и снижение надежности возможны на каждом из уровней. Каждый уровень характеризуется своим набором угроз. Наличие нестационарности на каждом из уровней существенно осложняет борьбу с угрозами и устранение последствий их воздействия.

Таким образом, обеспечение безопасности и надежности при организации Интернета вещей является комплексной, многоуровневой задачей, требующей решения ряда научных и технических проблем.

Для решения поставленной задачи были выделены пять уровней, характеризующих архитектуру Интернета вещей (рис. 1.1).

Уровень 1. Сенсоры, датчики, встроенные устройства, характеризующиеся очень низким энергопотреблением и низким уровнем защиты информации. Как правило, используются в смартфонах, нестандартных вычислительных устройствах, бытовой технике, дверных замках, холодильниках, RFID-метках (Radio Frequency Identification – радиочастотная идентификация) и т.д. Абсолютное

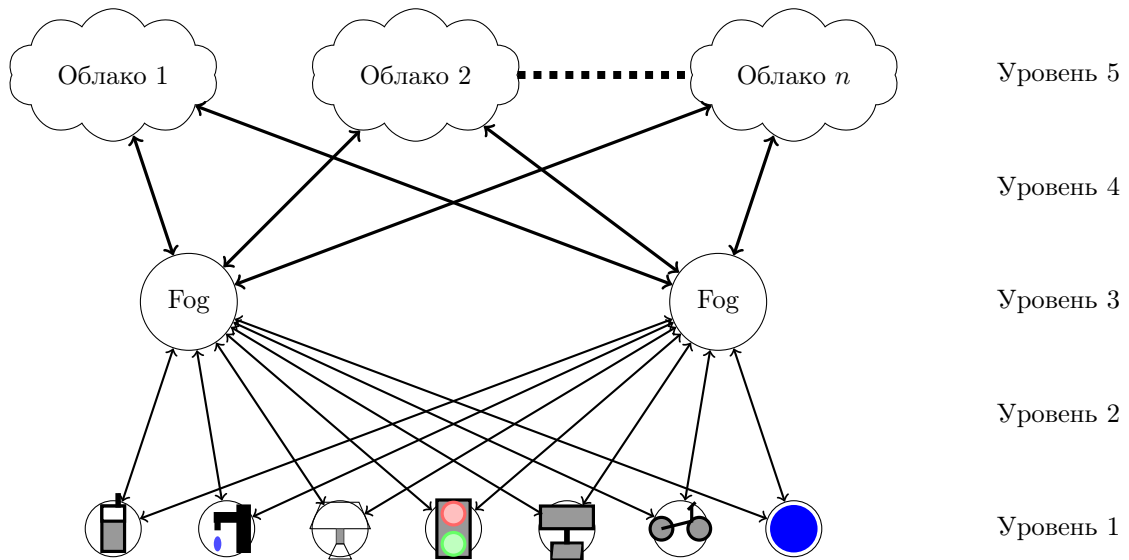


Рисунок 1.1 — Структурная схема работы Интернета вещей

большинство указанных устройств работают от батареи, поэтому проблема снижения энергопотребления наиболее актуальна для данного уровня.

Уровень 2. Беспроводные сети передачи данных. Процесс передачи реализуется устройствами с низким энергопотреблением, поэтому характеризуется высоким уровнем неопределенности, что затрудняет организацию безопасности при передаче.

Уровень 3. Вычислительная инфраструктура, объединяющая ресурсы миллиардов подключенных туманных устройств. Туманные устройства – это надежные и безопасные гетерогенные вычислительные устройства с ограниченными вычислительными возможностями. Очевидные преимущества туманных вычислений – возможность распределенной обработки данных, увеличение скорости работы приложений, минимизация задержек при передаче данных, улучшение качества обслуживания (Quality of Service – QoS) и качества взаимодействия (Quality of Experience – QoE). Основной проблемой на данном уровне является сложность организации распределенных вычислений.

Уровень 4. Всемирная сеть передачи данных Интернет. Характеризуется высоким уровнем безопасности и надежности.

Уровень 5. Распределенные центры хранения и обработки данных, характеризующиеся возможностью динамического масштабирования и адаптации параметров для обеспечения различных уровней надежности и снижения рисков нарушения конфиденциальности, потери и утечки конфиденциальных данных, DDoS-атак и т.д.

Для большинства облачных хранилищ имеет место проблема сохранения баланса между надежностью и избыточностью: для обеспечения высокого уровня надежности требуется большая избыточность как на аппаратном уровне, так и на уровне данных; с другой стороны, снижение избыточности в целях экономии ресурсов может привести к искажению или потере данных, что недопустимо в современных условиях и может привести к большим бизнес-потерям.

Согласно альянсу облачной безопасности [354], многие компании страдают от потери критически важных данных, хранящихся в облаках. Например, в апреле 2011 года из-за технических сбоев большинство клиентов Amazon EC2 потеряли свои данные. В ноябре 2014 года злоумышленники захватили личную информацию сотрудников Sony. В июне 2014 года из-за утечки конфиденциальной информации из Code Spaces его пришлось закрыть [354]. Amazon Simple Storage Service (S3) обнаружили проблему повреждения данных, вызванную ошибкой балансировщика нагрузки [252].

Ошибка используется как общий термин для обозначения отказа отдельного компонента системы, вызванного неисправным оборудованием, неисправным программным обеспечением, незаконным изменением данных, техническими неисправностями, DDoS-атаками, хакерскими атаками, ошибками, созданными человеком, банкротством провайдера, вирусами, катастрофами и т.д. [11, 260].

1.3 Проблемы использования облачных вычислений в условиях неопределенности возникновения киберугроз

Для построения распределенной системы хранения и обработки можно использовать различные подходы. Некоторые из них основаны на парадигмах облачных и грид-вычислений [360]. Эти инфраструктуры имеют как общие характеристики, так и принципиальные различия.

К облакам, используемым для хранения данных, предъявляется ряд требований, таких как безопасность, надежность и масштабируемость в условиях ограниченной пропускной способности интернет-соединения [146, 244, 355].

Для обеспечения быстрого доступа к распределенным данным и поддержания высокой степени надежности, доступности и масштабируемости [161] пред-

ложена система Bigtable, основанная на репликации незашифрованных данных без обеспечения конфиденциальности и безопасности данных.

Альтернативный механизм – Hadoop и MapReduce, основан на разделении набора данных на независимые блоки, которые обрабатываются параллельно, и сокращении их количества [191]. Однако, в работе [348] была показана его низкая эффективность.

Популярными становятся нереляционные базы данных (Not only Structured Query Language – NoSQL), учитывающие неоднородность неструктурированных данных [266]. Однако, две самые популярные базы данных NoSQL – Cassandra и MongoDB, имеют проблемы с безопасностью и конфиденциальностью данных [342].

Для решения этих проблем можно использовать классическое шифрование данных, но оно неприменимо в случае обработки данных. Невозможно проводить интенсивный анализ и обработку зашифрованных классическими алгоритмами данных, и требуются значительные вычислительные ресурсы для выполнения шифрования и дешифрования данных. Для устранения этого недостатка используются схемы, основанные на гомоморфных вычислениях [214, 299] и структуры доступа [10, 11, 223].

Существует множество подходов для интенсивных вычислений с распределенными данными [359].

Распределенная база данных (Distributed Data Base – DDB) хранит данные на различных сайтах компьютерной сети и использует логику для организации набора данных [302]. Есть два способа создания DDB: при нисходящем подходе база данных распределяется по различным сайтам, в то время как восходящий подход объединяет несколько отдельных баз данных в один интерфейс. Основная область применения DDB – это структурированное хранилище данных, поэтому оно не применимо к произвольным наборам данных, таким как Big Data.

Сеть доставки контента (Content Delivery Network – CDN) [219] представляет собой набор (не исходных) серверов, которые кэшируют данные, удовлетворяют запросы клиентов к базе данных и уменьшают рабочую нагрузку исходных серверов. Можно сформулировать следующие принципы CDN: балансировка нагрузки, сохранение пропускной способности и снижение временных задержек. Однако, на практике сети CDN не используются широко из-за отсутствия гибкости.

Основными принципами одноранговой P2P-сети (peer-to-peer) [301] являются масштабируемость и надежность, достигаемые за счет децентрализованной структуры и избыточности, совместного использования ресурсов и анонимности. P2P-сети эффективны в предоставлении быстрого доступа к файлам, группе одноранговых узлов. Но большинство сетей P2P не допускают интегрированных вычислений и служат средой распространения данных.

В последние годы растет потребность в разработке надежных распределенных систем хранения данных с использованием мультиоблачного подхода. Подобные распределенные системы состоят из набора хранилищ, принадлежащих разным CSP, с разными техническими характеристиками.

К преимуществам таких систем относятся автономность каждого CSP, повышенная стабильность и надежность, повышенная производительность, гибкость, уменьшение объемов обрабатываемых и хранимых данных в каждом CSP и т.д.

Автономность и разное географическое расположение центров обработки данных обеспечивают надежность хранения данных в случае различных глобальных технических неисправностей, сбоев, вызванных сейсмическими, гидрогеологическими, электрическими, техногенными и другими катастрофами.

Повышенная стабильность и надежность системы выражаются в сохранении функционала даже в случае выхода из строя отдельных крупных частей системы.

Повышенная производительность достигается за счет размещения распределенных данных в центре обработки, ближайшем к пользователю. Следовательно, нагрузка на сеть передачи данных и время отклика уменьшаются.

Повышенная гибкость реорганизации достигается за счет модульности системы. К недостаткам такого решения можно отнести:

1. *Повышенную сложность услуги:* требуется учитывать различные технические характеристики каждого CSP отдельно. Существующие публичные библиотеки общего назначения, такие как Apache jclouds и Apache Libcloud, не позволяют разрабатывать надежные мультиоблачные решения [286].
2. *Усложнение контроля целостности данных:* требуется организация контроля целостности данных расположенных в различных облачных хранилищах, рекомендуется использовать алгоритмы групповой под-

писи на основе структур доступа для контроля целостности данных в распределенных системах.

3. *Усложнение механизмов обеспечения безопасности данных*: требуется разработать новые механизмы обеспечения безопасности для решения основных проблем облачных вычислений, представленных в отчете о безопасности CSA (Cloud Security Alliance) [354].

Основными методами обеспечения надежности облачных хранилищ являются репликация, коды стирания, структуры доступа и коды исправления ошибок [161, 233].

Репликация обеспечивает высокий уровень надежности, но приводит к большой избыточности данных, что значительно увеличивает стоимость хранения и обслуживания.

Чтобы снизить вероятность ошибок, Ferru и др. [277] предложили структуру под названием CLOUDMF для управления несколькими облаками, но она не поддерживает межоблачные соединения.

Другие решения используют подход из работы Houidi и др. [181]. Авторы представили платформу Cloud Broker, которая позволяет передавать данные между двумя разными облаками с помощью технологии OpenFlow. Однако технология OpenFlow поддерживает ограниченное количество CSP.

Чтобы улучшить производительность мультиоблаков, Riteau [328] предложил подход, основанный на Nimbus, который позволяет передавать данные между облаками с высокой скоростью.

Облачные технологии позволяют расширить вычислительные возможности путем совместного использования удаленных аппаратных ресурсов. Преимущества использования облачных технологий побуждают организации и отдельных пользователей переносить свои данные, приложения и сервисы в облако. Однако стоит отметить, что кибератаки на облака в последние годы участились, и серьезно обострилась ситуация с нарушениями безопасности данных [185]. Согласно отчету о безопасности Cloud Security Alliance 2019, многие компании и пользователи облачных сред стали жертвами кибератак [354].

Для эффективного противодействия этим атакам требуется совместное использование интеллектуальных механизмов обеспечения безопасности и распределенных систем хранения, которые в совокупности обеспечивают безопасность, надежность и непрерывный доступ к данным [45].

Центр по Интернет-безопасности выпустил бенчмарки для облачных развертываний, чтобы проверить безопасность облачных провайдеров [173]. Предлагаемые бенчмарки позволяют оценить безопасность отдельного облачного сервиса, но не позволяют оценить влияние DDoS-атак и сговора в облаке на целостность и конфиденциальность данных.

Для обеспечения безопасности Aikat и др. [327] предложили взять за основу стандарт ISO/IEC 27001 от 2013 г. или публикацию NIST SP 800-53. Использование вышеперечисленных стандартов организациями и пользователями приведет к удорожанию облачных технологий.

Облачные хранилища обеспечивают высокий уровень безопасности, надежности и доступности за счет использования парадигмы распределенного хранения [11]. Как показано в работе [185], подобный подход позволяет противостоять классическим атакам, описанным в [354]. Атака на распределенное облачное хранилище заключается в одновременной атаке нескольких облачных провайдеров, что маловероятно, но возможно. Следовательно, требуются специальные методы обеспечения безопасности распределенного облачного хранилища. Наиболее известными и часто используемыми из них являются взвешенные структуры доступа и проактивная безопасность [147].

Существует несколько реализаций концепции проактивной безопасности, однако, ни одна из них не предназначена для явного использования нескольких облаков. Более того, они обеспечивают доступность данных, но не проверку целостности данных в режиме реального времени.

В таблице 1 представлены основные характеристики моделей распределенного хранения, а также сложность кодирования и декодирования соответствующих им методов. Здесь n – количество CSP, k – количество CSP, достаточное для восстановления данных, а L – длина сохраненных данных. Как видно из таблицы 1, далеко не все известные решения способны обеспечить одновременно надежное и безопасное распределенное хранение данных.

Надежность. Для обеспечения надежности применяются три основных метода [182]:

1. Репликация;
2. Коды стирания;
3. их модификации.

Chang и др. [161] представили модифицированный метод репликации данных, обеспечивающий высокую скорость кодирования и декодирования. Но он

Таблица 1 — Характеристики схем распределенного облачного хранения данных

| Схема распределенного хранения | Характеристики | | | | | | | | |
|--------------------------------|----------------|------------|-------------|--------------------|------------------|--------------------|-----------------|-------------------------------|-------------------------------|
| | Доступность | Надежность | Целостность | Конфиденциальность | Масштабируемость | Отказоустойчивость | Избыточность | Кодирование | Декодирование |
| [189] | • | • | | • | • | | $\frac{n}{k}$ | $O(L \cdot \log L)$ | $O(L \cdot \log L)$ |
| [192] | • | • | • | • | | | 2 | $O(L)$ | $O(L)$ |
| [145] | • | • | • | • | • | | $\frac{4n}{3k}$ | $O(L^2)$ | $O(L^2)$ |
| [11] | • | • | • | | • | • | $\frac{n}{k}$ | $O(n \cdot L)$ | $O(L \cdot \log L)$ |
| [336] | • | • | | • | • | | $\frac{n}{k}$ | $O(L^2)$ | $O(L^2)$ |
| [292] | • | • | | • | • | • | $\frac{n}{k}$ | $O(L^2)$ | $O(L^2)$ |
| [217] | • | • | | | • | • | 3 | $O(1)$ | $O(1)$ |
| [224] | | • | | | • | • | $\frac{n}{k}$ | $O(n \cdot L)$ | $O\left(\frac{L}{n}\right)$ |
| [255] | • | • | • | • | • | • | $\frac{n}{k}$ | $O(L^2)$ | $O(L^2)$ |
| [270] | • | • | • | • | | | $\frac{n}{k}$ | $O(n \cdot L^2 \cdot \log L)$ | $O(n \cdot L^2 \cdot \log L)$ |
| [337] | • | • | | | • | | $\frac{n}{k}$ | $O(n \cdot L)$ | $O\left(\frac{L}{n}\right)$ |
| [272] | • | • | | | • | • | $\frac{n}{k}$ | $O(n \cdot L)$ | $O\left(\frac{L}{n}\right)$ |
| [338] | | | • | • | | | 1 | $O(L)$ | $O(L)$ |
| [10] | • | • | • | • | • | • | $\frac{n}{k-1}$ | $O(n \cdot L)$ | $O(L \cdot \log L)$ |
| [371] | • | • | | • | • | | $\frac{n}{k}$ | $O(L^2)$ | $O(L^2)$ |
| [370] | • | • | | • | • | | $\frac{n}{k}$ | $O(k \cdot n \cdot L)$ | $O(L \cdot \log^2 L)$ |
| [198] | • | • | • | • | • | • | $\frac{n-1}{k}$ | $O(n \cdot L^2 \log L)$ | $O(k \cdot L^2 \log L)$ |

требует применения дополнительных криптографических примитивов для обеспечения безопасности и имеет высокую избыточность по сравнению с кодами стирания.

На данный момент предложен ряд различных модификаций кодов стирания для создания надежных методов распределенного хранения данных. Совместное использование кодов исправления ошибок и кодов стирания позволяет поддерживать работоспособность системы и минимизировать нагрузку на сеть передачи данных при восстановлении потерянных фрагментов [271, 292]. Коды стирания на основе избыточной системы остаточных классов (Redundant RNS) [320] позволяют обрабатывать данные в закодированной форме [11], и могут быть использованы как при проектировании маломощных устройств беспроводной передачи данных [320], так и систем распределенного хранения.

Безопасность. Безопасные системы распределенного хранения данных основаны на использовании криптографических примитивов – алгоритмов симметричного шифрования (AES) и цифровой подписи на основе RSA (Rivest, Shamir, Adleman) [338]. Преимуществами этих подходов являются высокая скорость шифрования и дешифрования и низкая избыточность данных. Недостаток – ошибка в зашифрованных данных приводит к их потере. Для устранения данного недостатка требуется применение дополнительных механизмов доступа к данным в течение длительного времени [267].

Безопасность и надежность. При построении безопасных и надежных облачных хранилищ используются следующие методы: эллиптическая криптография и коды стирания [198, 270], структуры доступа [10, 300, 343], коды исправления ошибок [11, 145], алгоритмы на основе графов и модифицированный алгоритм репликации данных [189], шифрование на основе атрибутов [347] и т.д.

В таблице 1 представлен сравнительный анализ основных известных методов организации распределенного облачного хранения данных с точки зрения обеспечения следующих характеристик: доступность, конфиденциальность, отказоустойчивость, целостность, избыточность, надежность и масштабируемость.

Из таблицы 1 видно, что наиболее эффективным с точки зрения вычислительной сложности является метод из работы [217]. Однако, данные хранятся в незашифрованном виде, что ограничивает его применимость для хранения конфиденциальных данных.

Альтернативный подход заключается в использовании восстанавливающих кодов [271], кодов стирания [292] и кодов исправления ошибок на основе RRNS. Однако, восстанавливающие коды и коды стирания не позволяют обрабатывать закодированные данные. Для обработки данных важным свойством кодов является гомоморфизм, поскольку гомоморфные вычисления позволяют обрабатывать закодированные данные без дополнительных вычислительных затрат на декодирование [299]. Значительный прорыв в области гомоморфных вычислений произошел благодаря работе Gentry [214]. Авторы предложили полностью гомоморфную схему для выполнения как сложения, так и умножения. Основными недостатками этого алгоритма являются значительная избыточность данных и отсутствие контроля над результатами арифметических операций.

Особого внимания заслуживает модель распределенного хранения данных, предложенная в работе [223], и гарантирующая безопасность, конфиденциальность, гомоморфизм, надежность и масштабируемость. Авторами предложены два подхода к построению систем на основе гомоморфных структур доступа в RRNS, причем модули RRNS используются в качестве секретных ключей, хранящихся у пользователей. Обработка данных приводит к экспоненциальному увеличению нагрузки на сеть и память, что делает данную модель неприменимой на практике в современных условиях.

Структуры доступа, предложенные в [151] и [280], обеспечивают безопасность и конфиденциальность данных. RRNS реализует тот же функционал, что и схема Mignotte, но позволяет контролировать результаты обработки данных.

Распределенные облачные хранилища также характеризуются рисками сговора [61]. Разработано несколько подходов к предотвращению сговора облаков [11, 277].

Как упоминалось выше, нестационарность облачной среды снижает эффективность, производительность, надежность и безопасность системы. Адаптивная парадигма снижает неопределенность, но редко применяется в облачных вычислениях [144, 172, 258].

Для решения проблем, связанных с неопределенностью, обычно используются стохастические и нечеткие методы, методы теории вероятностей и математической статистики [33]. Другие подходы основаны на методах машинного обучения (Machine Learning – ML) и используют информацию о характеристиках ранее реализованных распределенных систем для построения регрессионных моделей, деревьев решений и т.д. [143, 233]. Однако, поскольку характеристики

облачных сервисов быстро меняются, данных может не хватить для построения качественной модели, основанной на машинном обучении.

1.4 Подходы к обеспечению надежного и безопасного хранения и обработки данных на основе структур доступа

Рассмотрим следующий сценарий. Пользователь имеет конфиденциальные данные и решает не хранить их в едином облачном хранилище. Он делит их на несколько частей и хранит в разных облаках.

В данном сценарии имеет место несколько видов угроз безопасности.

1. К *преднамеренным угрозам* относятся несанкционированный доступ к информации, перехват, фальсификация, хакерские атаки и т.д. в одном или нескольких облаках.
2. К *случайным угрозам* относятся ошибки, сбои и т.д. Они могут привести к потере одного или нескольких фрагментов данных, несогласованности между разными копиями одних и тех же данных и/или невозможности восстановить исходные данные.
3. *Угрозы сговора* – это незаконное соглашение между двумя или более противниками (в контексте мультиоблачного хранения противниками выступают облачные сервисы) о получении полного доступа к личным данным.

Криптографические протоколы могут использоваться для уменьшения рисков преднамеренных угроз, но этого недостаточно для случайных угроз.

Для повышения безопасности и надежности систем хранения используют механизмы распределенного хранения на основе структур доступа и кодов исправления ошибок, которые распределяют данные по нескольким CSP и позволяют минимизировать вероятность кражи или потери информации в случае преднамеренных и случайных угроз. Примерами таких механизмов являются RACS [141], DepSky [192] и RRNS, использующая приближенный ранг (Approximate Rank RRNS – AR-RRNS) [11].

Далее рассматриваются четыре схемы совместного использования систем хранения, которые можно отнести к одному из двух классов (рис. 1.2):

1. Пороговые структуры доступа.

2. Взвешенные пороговые структуры доступа.

Первая структура порогового доступа (рис. 1.2*а*) – это классическая схема, когда каждое хранилище имеет одну долю данных одинакового размера. Пример такого хранилища – DepSky [192].

Вторая структура порогового доступа (рис. 1.2*б*) является расширением предыдущей схемы, предложенной Miranda-López и др. [45, 61], где у каждого хранилища одинаковое количество коротких долей. В обеих схемах данные могут быть восстановлены, если количество доступных долей превышает заданное количество (порог).

Разделение согласно традиционной взвешенной пороговой структуре доступа означает, что хранилища имеют одну долю разного размера [153] (рис. 1.2*в*)).

Tchernykh и др. [88] предложили взвешенную пороговую структуру доступа WA-RRNS (Weighted Access – RRNS) на основе избыточной системы остаточных классов, где каждое хранилище имеет несколько коротких долей (рис. 1.2*г*)). В той же работе была предложена более эффективная реализация WA-RRNS – WA-AR-RRNS, использующая приближенное значение ранга числа, представленного в RRNS, для ускорения процедуры декодирования.

В статье [29] рассматривается комбинация традиционной взвешенной схемы (рис. 1.2*в*)) и пороговой схемы коротких долей (рис. 1.2*б*)), разделяющая разное количество коротких долей одинакового размера между хранилищами (рис. 1.2*г*)).

При таком подходе данные могут быть восстановлены тогда и только тогда, когда сумма размеров долей не меньше заданного значения (порогового значения веса).

Далее будет показано, как размер и общее количество долей могут изменить надежность, уровень безопасности, скорость доступа и т.д. Указанные структуры уменьшают нагрузку на сеть передачи по сравнению с классическим механизмом репликации и снижают стоимость хранения данных.

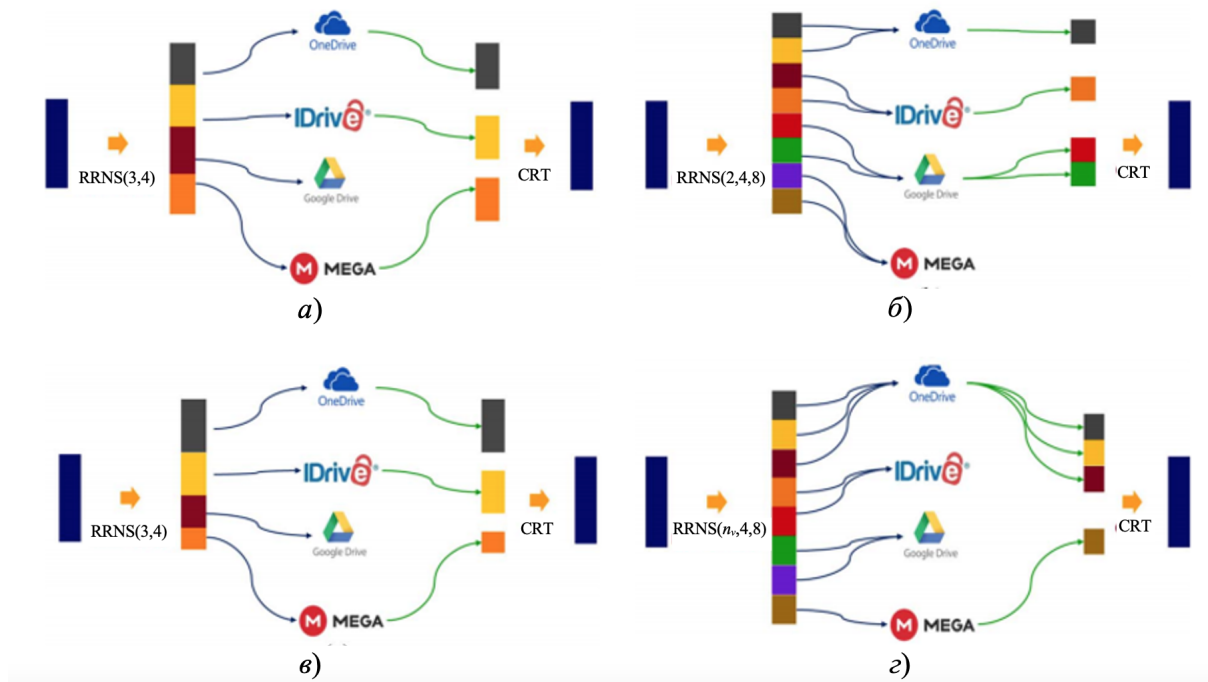


Рисунок 1.2 — Структуры доступа: *а*) пороговая структура доступа (одна доля на каждое хранилище); *б*) пороговая структура доступа (несколько коротких долей на каждое хранилище); *в*) взвешенная пороговая структура доступа (по одной доле разного размера на хранилище); *г*) взвешенная пороговая структура доступа (разное количество долей одинакового размера на хранилище)

1.5 Современные подходы к построению систем обработки конфиденциальных данных в закодированном виде

Rivest и др. в 1978 году [299] предложили новый механизм обработки закодированных данных, который лег в основу нового направления – гомоморфные вычисления. Термин гомоморфные вычисления (ГВ) определяет вид кодирования, при котором сохраняется способность вычислять определенные функции от закодированных данных. Выходные данные сохраняют особенности функции и формата ввода. Система, основанная на гомоморфных вычислениях, не имеет доступа к информации о закодированных данных и секретных ключах и использует только общедоступную информацию без риска утечки данных. Концепция гомоморфных вычислений базируется на отображении функций в пространстве открытых и закодированных сообщений. Гомоморфная функция, применяемая к закодированным данным, дает тот же результат (после декодирования), что и в случае ее применения к исходным незакодированным данным.

Пусть m_1 и m_2 – сообщения, c_1 и c_2 – соответствующие им закодированные сообщения. Операция \oplus в аддитивных гомоморфных вычислениях создает закодированное сообщение $c_{\oplus} \xleftarrow{\text{ГВ}} c_1 \oplus c_2$, которое можно декодировать до $m_1 + m_2$.

Аналогично, для операции \otimes в мультипликативных гомоморфных вычислениях генерируется закодированное сообщение $c_{\otimes} \xleftarrow{\text{ГВ}} c_1 \otimes c_2$, которое декодируется в $m_1 \times m_2$. В обоих случаях в гомоморфных вычислениях используются закодированные сообщения c_{\oplus} и c_{\otimes} , а не исходные сообщения m_1 и m_2 . В кодах, не обладающих свойством гомоморфизма, результаты операций $m_1 + m_2$ и $m_1 \times m_2$ не могут быть вычислены без предварительного декодирования c_1 и c_2 , но в этом случае пользователи жертвуют своей конфиденциальностью.

Гомоморфные вычисления классифицируются в соответствии со списком конкретных математических операций, выполняемых над закодированными данными. Эффективность и гибкость гомоморфных вычислений во многом зависят от количества операций в списке. Схема гомоморфных вычислений с большим числом операций более гибкая, но менее эффективная. И наоборот, схема с меньшим числом операций менее гибкая, но более эффективная.

Полностью гомоморфные вычисления – криптографический примитив, который реализует функцию кодирования, удовлетворяющую дополнительному требованию гомоморфности относительно двух арифметических операций над открытыми сообщениями (текстами): сложения и умножения.

Функция кодирования $E(k, m)$, где m – открытый текст, k – секретный ключ, гомоморфна относительно операции \odot над открытыми текстами, тогда и только тогда, когда существует эффективный алгоритм M , получающий на вход любую пару закодированных текстов вида $E(k, m_1)$, $E(k, m_2)$, а на выходе формирующий такой закодированный текст c_{\odot} , что при декодировании c_{\odot} будет получен открытый текст $m_1 \odot m_2$, где $\odot \in \{+, \times\}$ [133]. Если речь идет о гомоморфном коде, поддерживающем сразу две операции \oplus и \otimes , используется обозначение \odot .

Гомоморфные вычисления позволяют обрабатывать закодированные данные без их предварительного декодирования таким образом, что результат операций над закодированными данными эквивалентен после декодирования результату операции над открытыми данными [140]. При этом решается одна из проблем криптографии – проблема генерации, хранения и распространения общих сеансовых ключей и повышается уровень защищенности данных – сервер получает закодированные данные, обрабатывает их и возвращает закодирован-

ный результат, а открытые данные и секретные ключи не покидают безопасный сегмент при сетевом взаимодействии [131].

В следующих разделах представлен аналитический обзор трех типов гомоморфных вычислений: частично гомоморфные, в некоторой степени гомоморфные и полностью гомоморфные вычисления, обозначены их ограничения и возможности.

1.5.1 Частично гомоморфные вычисления

Частично гомоморфные вычисления (или частично гомоморфное кодирование, или частично гомоморфный код, или частично гомоморфная схема кодирования) поддерживают неограниченное количество операций одного типа. Например, аддитивный частично гомоморфный код допускает неограниченное количество сложений, но не поддерживает умножение.

Криптосистема, разработанная Ronald Rivest, Adi Shamir и Leonard Adleman (RSA), является первым мультипликативным частично гомоморфным кодом [329]. В общем, для двух сообщений m_1 и m_2 и соответствующих им кодов $c_1 = (m_1^e) \bmod n$ и $c_2 = (m_2^e) \bmod n$, где e выбирается так, что $\gcd(e, \phi) = 1$, $\phi = (q_1 - 1) \times (q_2 - 1)$, а q_1 и q_2 являются большими простыми числами, схема гомоморфного кодирования описывается следующим выражением

$$c_{\otimes} \stackrel{\text{ГВ}}{\leftarrow} (m_1 \times m_2)^e \bmod n = (m_1^e) \bmod n \times (m_2^e) \bmod n = c_1 \otimes c_2, \quad (1.1)$$

где $n = q_1 \times q_2$.

RSA не является семантически безопасным из-за детерминированного алгоритма шифрования. Примером релевантной мультипликативной частично гомоморфной схемы кодирования является схема Taher El-Gamal [201].

Криптосистема Shafi Goldwasser и Silvio Micali (GM) является первой аддитивной частично гомоморфной схемой кодирования [222]. Согласно схеме GM, сообщениям m_1 и m_2 ставятся в соответствие закодированные сообщения $c_1 = (b_1^2 \times e^{m_1}) \bmod n$ и $c_2 = (b_2^2 \times e^{m_2}) \bmod n$, где b_1^2 и b_2^2 – квадратичные невычеты, такие что $\gcd(b_1^2, n) = \gcd(b_2^2, n) = 1$, а e – квадратичный невычет по модулю n .

Схема GM является гомоморфной, кодирование $m_1 + m_2$ осуществляется согласно следующей формуле

$$\begin{aligned} c_{\oplus} \stackrel{\text{ГВ}}{\leftarrow} \left((b_1 \times b_2)^2 \times e^{m_1+m_2} \right) \bmod n &= \left((b_1^2 \times e^{m_1}) \times (b_2^2 \times e^{m_2}) \right) \bmod n \\ &= (b_1^2 \times e^{m_1}) \bmod n \times (b_2^2 \times e^{m_2}) \bmod n = c_1 \times c_2. \end{aligned} \quad (1.2)$$

Однако, GM не является эффективной схемой, поскольку закодированные тексты могут быть в несколько сотен раз больше, чем исходные открытые тексты.

Примерами релевантных аддитивных криптосистем являются частично гомоморфные коды, разработанные Josh (Cohen) Benaloh в 1994 году [159], David Naccache и Jacques Stern (NS) в 1997 году [288], Tatsuaki Okamoto и Shigenori Uchiyama (OU) в 1998 году [296], Pascal Paillier в 1999 году [303], Ivan Damgård и Mads Jurik (DJ) в 2001 году [190], Steven Galbraith в 2002 году [212] и Akinori Kawachi, Keisuke Tanaka и Keita Hagawa (KTX) в 2007 году [256].

Процесс кодирования в частично гомоморфных вычислениях не гарантирует заданный уровень безопасности и основан на добавлении «шума» к информации. Термин «шум» обозначает умеренное количество ошибок, вводимых в закодированное сообщение и порождающее неточное соотношение [281].

1.5.2 В некоторой степени гомоморфные вычисления

В некоторой степени гомоморфные вычисления (или в некоторой степени гомоморфное кодирование, или в некоторой степени гомоморфный код, или в некоторой степени гомоморфная схема кодирования) поддерживают заранее определенное ограниченное количество различных разрешенных гомоморфных операций. Каждая операция увеличивает основной шум, поэтому схема позволяет вычислять корректно лишь ограниченное число арифметических операций. Если шум превысил порог, то корректно декодировать результат вычислений нельзя.

Схема Dan Boneh, Eu-Jin Goh и Kobbi Nissim (BGN) [162] была первой гомоморфной криптосистемой, поддерживающей как сложение, так и умножение закодированных текстов постоянного размера. BGN основан на проблеме установления принадлежности элемента подгруппе G_p группы G порядка $n = q_1 \times q_2$ [218]. В BGN закодированные тексты $c_1 = g^{m_1} \times h^{e_1}$ и $c_2 = g^{m_2} \times h^{e_2}$

кодируют сообщения m_1 и m_2 , где g и u – два случайных числа из G , $h = u^{q_2}$ – случайный образующий элемент подгруппы G порядка q_1 , e_1 и e_2 – случайные числа из множества $\{0, 1, \dots, n-1\}$.

Кодирование $m_1 + m_2$ осуществляется по следующей формуле

$$\begin{aligned} c_{\oplus} &\stackrel{\text{ГВ}}{\longleftarrow} g^{m_1+m_2} \times h^{e_1+e_2+e} = (g^{m_1} \times h^{e_1}) \times (g^{m_2} \times h^{e_2}) \times h^e \\ &= c_1 \times c_2 \times h^e = c_1 \oplus c_2. \end{aligned} \quad (1.3)$$

Тем не менее, BGN непрактична с точки зрения выполнения операции умножения, поскольку она вычисляет c_{\otimes} только один раз, используя билинейное отображение $s: G \times G = G_1$, где G_1 – группа порядка $n = q_1 \times q_2$.

Пусть $g_1 = s(g, g)$ и $h_1 = s(g, h)$, где g_1 имеет порядок n , а h_1 имеет порядок q_1 . Таким образом, существует α такое, что $h = g^{\alpha q_2}$.

Кодирование $m_1 \times m_2$ осуществляется по следующей формуле

$$\begin{aligned} c_{\otimes} &\stackrel{\text{ГВ}}{\longleftarrow} g_1^{m_1 m_2} \times h_1^{m_1 e_2 + m_2 e_1 + \alpha q_2 e_1 e_2 + e} \\ &= g_1^{m_1 m_2} \times h_1^{m_1 e_2 + m_2 e_1 + \alpha q_2 e_1 e_2} \times h_1^e \\ &= g_1^{m_1 m_2} \times g_1^{\alpha q_2 (m_1 e_2 + m_2 e_1 + \alpha q_2 e_1 e_2)} \times h_1^e \\ &= g_1^{m_1 m_2 + \alpha q_2 (m_1 e_2 + m_2 e_1 + \alpha q_2 e_1 e_2)} \times h_1^e \\ &= s(g, g)^{(m_1 + \alpha q_2 e_1)(m_2 + \alpha q_2 e_2)} h_1^e \\ &= s(g^{m_1 + \alpha q_2 e_1}, g^{m_2 + \alpha q_2 e_2}) \times h_1^e \\ &= s(g^{m_1} \times g^{\alpha q_2 e_1}, g^{m_2} \times g^{\alpha q_2 e_2}) \times h_1^e \\ &= s(g^{m_1} \times h^{e_1}, g^{m_2} \times h^{e_2}) \times h_1^e \\ &= s(c_1, c_2) \times h_1^e = c_1 \otimes c_2, \end{aligned} \quad (1.4)$$

где $m_1 e_2 + m_2 e_1 + \alpha q_2 e_1 e_2 + e$ равномерно распределено в \mathbb{Z}_N , и c_{\otimes} – равномерное распределенное кодирование $(m_1 \times m_2) \bmod n$, но теперь в G_1 , а не в G . При этом, BGN все еще остается аддитивно гомоморфной в G_1 .

На рисунке 1.3 представлена хронология наиболее значимых изобретений в истории гомоморфных вычислений до появления первой схемы полностью гомоморфного кодирования Gentry в 2009 году [214].

Наиболее известные схемы в некоторой степени гомоморфных вычислений были предложены Andrew Yao [372] в 1982 году, Tomas Sander, Adam Young и Moti Yung (SYU) [331] в 1999 году, а также Yuval Ishai и Anat Paskin (IP) [253] в 2007 году. Каждая из них имеет свои преимущества и недостатки, касающиеся

количества операций, избыточности закодированных текстов, эффективности обработки и уязвимости для атак. Коротко резюмируя, указанные схемы либо небезопасны, либо непрактичны, но при этом они являются основой для полностью гомоморфного кодирования.

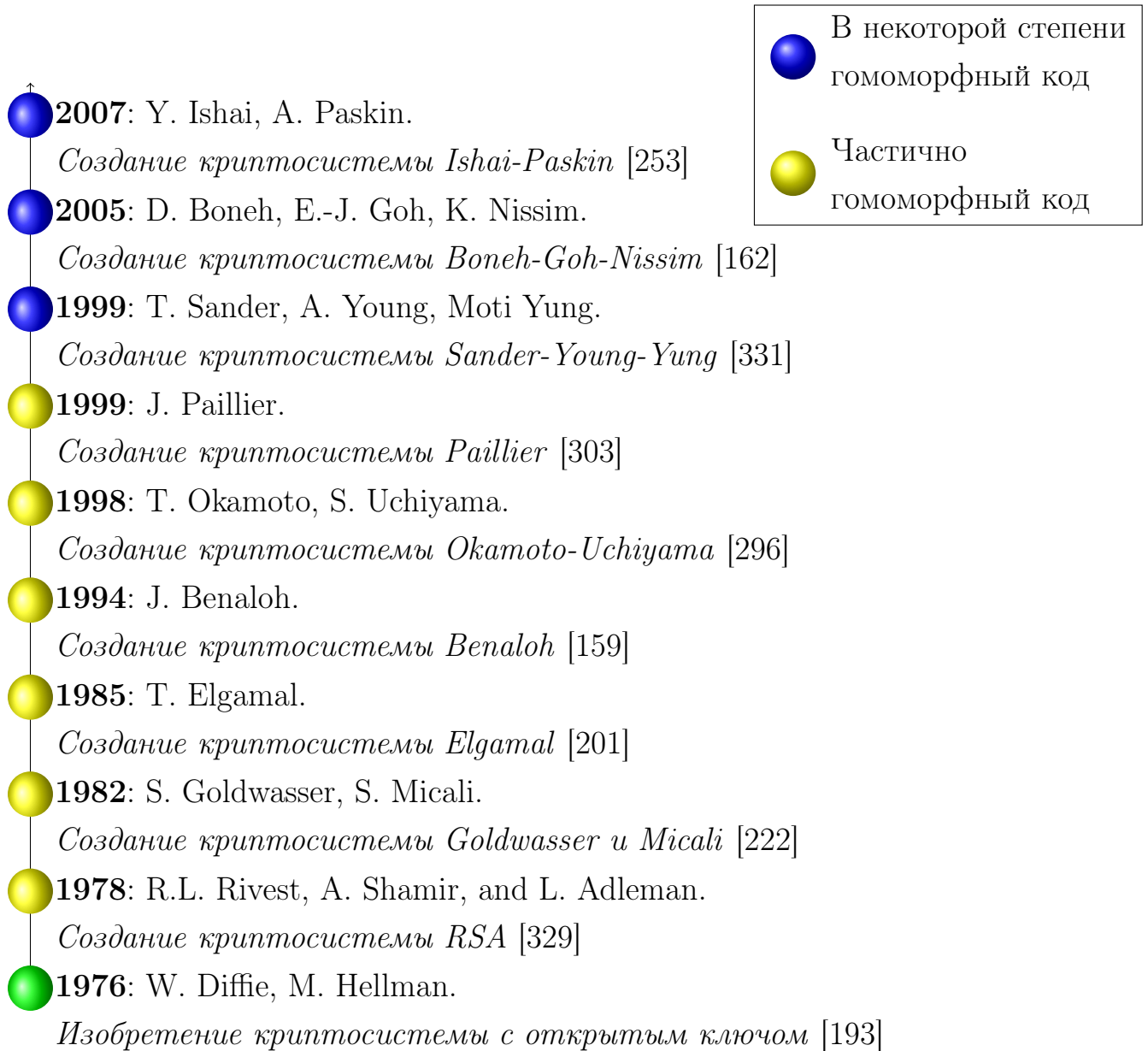


Рисунок 1.3 — Хронология изобретения гомоморфных кодов

1.5.3 Полностью гомоморфные вычисления

Первые полностью гомоморфные вычисления (или полностью гомоморфное кодирование, или полностью гомоморфный код, или полностью гомоморфная схема кодирования) были разработаны Gentry [214] в 2009 году, спустя более 30 лет исследований в данной области, начавшихся с изобретения криптографии с открытым ключом в 1976 году [193].

Основной проблемой, ограничивающей применение в некоторой степени гомоморфных вычислений, является bootstrapping – процедура для снижения уровня шума в закодированном тексте (сообщении, данных). Для в некоторой степени гомоморфных кодов bootstrapping является вычислительно сложной операцией из-за проблемы идеальных классов смежности. Схема может гомоморфно оценивать свою функцию декодирования. Конструкция в некоторой степени гомоморфных кодов использует понятие идеала в алгебре решеток. Идеал I в кольце $R = \mathbb{Z}[x] / (f(x))$, где $f(x)$ – это многочлен степени n со старшим коэффициентом единица, и $a + b \in I$, $r \times a \in I$ для всех $a, b \in I$ и $r \in R$.

Для шифрования используются два взаимно простых идеала I и J , т.е. выполняется соотношение $I + J = R$. Эти идеалы представляются соответствующими базисами B_I и B_J^{pk} (открытый ключ) и B_J^{sk} (секретный ключ), схема преобразует открытые тексты m_1 и m_2 в закодированные тексты $c_1 = (\psi_1) \bmod B_J^{pk}$ и $c_2 = (\psi_2) \bmod B_J^{pk}$, где $\psi_1 \xleftarrow{\text{ГВ}} \text{samp}(B_I, m_1)$ и $\psi_2 \xleftarrow{\text{ГВ}} \text{samp}(B_I, m_2)$ – выборки из смежных классов $I + m_1$ и $I + m_2$ соответственно.

Кодирование $m_1 + m_2$ осуществляется по формуле

$$c_{\oplus} \xleftarrow{\text{ГВ}} (\psi_1 + \psi_2) \bmod B_J^{pk} = (\psi_1) \bmod B_J^{pk} + (\psi_2) \bmod B_J^{pk} = c_1 \oplus c_2, \quad (1.5)$$

тогда как для $m_1 \times m_2$ используется формула

$$c_{\otimes} \xleftarrow{\text{ГВ}} (\psi_1 \times \psi_2) \bmod B_J^{pk} = (\psi_1) \bmod B_J^{pk} \times (\psi_2) \bmod B_J^{pk} = c_1 \otimes c_2. \quad (1.6)$$

Процедура bootstrapping снижает шум в закодированном тексте и может применяться неограниченное количество раз, что позволяет построить схему полностью гомоморфного кодирования.

Подход с использованием алгебры решеток для построения полностью гомоморфного кода Gentry сопряжен с большой вычислительной сложностью

арифметических операций, что существенно усложняет его реализацию и делает неприменимым на практике. Однако, данный подход является объектом различных оптимизаций и основой для новых подходов, решающих в той или иной мере проблемы производительности и технической реализации полностью гомоморфных вычислений [166, 168, 205, 209, 216].

С момента изобретения первого полностью гомоморфного кода Gentry было разработано множество новых полностью гомоморфных схем кодирования, объединяемых четырьмя семействами: полностью гомоморфные схемы на основе идеальной решетки, полностью гомоморфные схемы на основе целых чисел, полностью гомоморфные схемы на основе обучения с ошибками и полностью гомоморфные схемы на основе усеченного полиномиального кольца N -й степени (N^{th} -degree TRUncated polynomial ring – NTRU).

1. Схемы первого семейства следуют оригинальной идее Gentry, где безопасность базируется на вычислительной сложности поиска ближайшего элемента по алгебраической решетке.
2. Второе семейство базируется на вычислительно сложной задаче поиска наибольшего общего делителя при условии, что числа содержат ошибку [166, 209, 214].
3. Третье семейство включает схемы, основанные на обучении с ошибкой (Learning With Errors – LWE) [216] и кольцевом обучении с ошибкой (Ring LWE – RLWE) [166, 167, 202]. Оба подхода сводятся к решеточной задаче.
4. Четвертое семейство формируют схемы [149, 235] так же основанные на схеме NTRU [330]. Схема NTRU также связана с решетками, возникающими при рассмотрении колец многочленов над целыми числами.

На рисунке 1.4 представлена хронология разработки подходов к построению полностью гомоморфных схем кодирования: Craig Gentry [213] в 2009 году, Craig Gentry и Shai Halevi (GH) в 2011 году [215], Zvika Brakerski, Craig Gentry и Vinod Vaikuntanathan (BGV) [376] в 2012 году, вариант Fan-Vercauteren масштабно-инвариантной схемы Zvika Brakerski (BFV) [166] в 2012 году, и Jung Cheon, Andrey Kim, Miran Kim и Yongsoo Song (CKKS) [239] в 2017 году.

Появление первой схемы полностью гомоморфного кодирования оказало значительное влияние на разработку безопасных систем, но не на их реализацию. Высокий уровень безопасности полностью гомоморфных решений потенциально способен качественно повысить уровень многих технологий, например,

аутсорсинга вычислений в облачных средах, но эффективная реализация полностью гомоморфных вычислений на данный момент невозможна из-за ряда ограничений.

Термин конфиденциальный гомоморфизм был введен Rivest [299] для формального описания полностью гомоморфного кодирования: основная идея заключается в построении эффективных вычислений над закодированными данными без использования секретного ключа [213]. Концепции полностью гомоморфных вычислений не свойственна обфускация, т.е. схема не способна скрыть последовательность из l инструкций, называемых программой P , $P = \{I_1, I_2, \dots, I_l\}$. Таким образом, учитывая ввод открытого текста m и программы P , $O(P) = \hat{P}$ является преобразованием обфускации P , если только \hat{P} и P имеют одинаковое наблюдаемое поведение. Точнее, если P выходит из строя или завершается с ошибкой, то \hat{P} может завершиться, а может и не завершиться; в противном случае $P(m) = \hat{P}(m)$. Более подробная информация и дополнительные соображения по обфускации преобразований представлены в работе [183].

Третья сторона может обрабатывать $\hat{P}(m)$, не получая никакой информации о \hat{P} . Главный недостаток подхода – возможность узнать о связи между m и $\hat{P}(m)$, в отличие от полностью гомоморфного кодирования, где третья сторона может обрабатывать закодированную версию $P(m)$, но не может декодировать m и $\hat{P}(m)$.

Общая идея полностью гомоморфных вычислений заключается в том, что функцию f можно эффективно выразить как схему, обрабатывающую гомоморфно закодированные данные, например, программы, математические операции и т.д. [213].

Полностью гомоморфные вычисления считаются перспективным постквантовым инструментом [313]. Современные алгоритмы защиты данных с открытым ключом основаны на сложности решения таких проблем, как разложение на множители или дискретное логарифмирование. Считается, что эти широко изучаемые задачи трудноразрешимы на классическом оборудовании. Однако, противник, оснащенный достаточно мощным квантовым компьютером, сможет легко их решить. Несмотря на то, что квантового компьютера на сегодняшний день не существует, его потенциал считается угрозой.

- ↑
- **2020:** A. Al Badawi, J. Chao, J. Lin, C.F. Mun, S.J. Jie, B.H.M. Tan, N. Xiao, M.M.A. Khin, R.C. Vijay. *Реализация AlexNet с помощью нейронных сетей, сохраняющих конфиденциальность* [356,357]
 - **2019:** S. Wagh, D. Gupta, N. Chandran. *Создание системы SecureNN для обучения нейронных сетей, сохраняющих конфиденциальность* [361]
 - **2018:** L.T. Phong, Y. Aono, T. Hayashi, L. Wang, S. Moriai. *Создание модели глубокого обучения, сохраняющей конфиденциальность* [317]
 - **2017:** J.H. Cheon, A. Kim, M. Kim, Y. Song. *Создание криптосистемы CKKS* [239]
 - **2017:** H. Chabanne, A. de Wargny, J. Milgram, C. Morel, E. Prouff. *Создание глубокой нейронной сети, сохраняющей конфиденциальность* [316]
 - **2016:** Q. Zhang, L.T. Yang, Z. Chen. *Создание глубоких нейронных сетей, сохраняющих конфиденциальность при обработке больших данных* [374]
 - **2016:** H. Takabi, E. Hesamifard, M. Ghasemi. *Создание протокола мульти-партийных вычислений, сохраняющих конфиденциальность на базе гомоморфного кодирования* [352]
 - **2016** R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, J. Wernsing. *Создание системы CryptoNets* [184]
 - **2016:** A. Khedr, G. Gulak, V. Vaikuntanathan. *Создание алгоритма, сохраняющего конфиденциальность данных, обрабатываемых с помощью классификатора на базе дерева решений* [257]
 - **2012:** Z. Brakerski, C. Gentry, V. Vaikuntanathan. *Создание гомоморфного кода BGV* [376]
 - **2012:** Z. Brakerski [166] / J. Fan и F. Vercauteren [202]. *Создание гомоморфного кода BFV*
 - **2011:** M. Naehrig, K. Lauter, V. Vaikuntanathan. *Проектирование первого алгоритма логистической регрессии, сохраняющего конфиденциальность данных* [289]
 - **2011:** C. Gentry и S. Halevi. *Реализация первого гомоморфного кода* [215]
 - **2009:** C. Gentry. *Создание первого полностью гомоморфного кода* [213]
- ↓

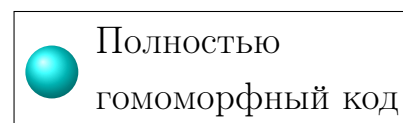


Рисунок 1.4 — Хронология разработки полностью гомоморфных кодов

Суть полностью гомоморфных вычислений состоит в получении закодированного текста $f(c)$ для любой желаемой функции f и закодированного текста c открытого текста m . Пока не произошла утечка информации о c , $f(c)$ и m , функцию f можно вычислять эффективно и безопасно. Ожидаемый функционал схемы полностью гомоморфного кодирования ϵ схож с функционалом классической модели черного ящика в компьютерных системах и поясняется на рисунке 1.5.

Наиболее сложная проблема при реализации схемы на рисунке 1.5 – найти подходящий механизм $Evaluate_\epsilon$, реализующий необходимый функционал за приемлемое время. Формально схема полностью гомоморфного кодирования

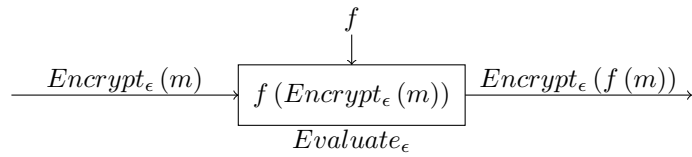


Рисунок 1.5 — Концепция гомоморфного кодирования

ϵ определяет обычную схему с открытым ключом содержащую четыре блока: $KeyGen_\epsilon$, $Encrypt_\epsilon$, $Decrypt_\epsilon$, и $Evaluate_\epsilon$ [213]. Вычислительная сложность всех операций в схеме ϵ должна быть полиномиальной по отношению к параметру безопасности λ , где

1. $KeyGen_\epsilon$ принимает λ в качестве входных данных и создает открытый ключ pk и секретный ключ sk в качестве выходных данных, где pk отображается из пространства открытого текста P в пространство закодированного текста \mathbb{C} , а sk в противоположном направлении.
2. $Encrypt_\epsilon$ использует pk и открытый текст $m \in P$ в качестве входных данных и формирует закодированный текст $c \in \mathbb{C}$, который представляет собой выходные данные.
3. $Decrypt_\epsilon$ определяет процесс, противоположный $Encrypt_\epsilon$, он получает sk и $c \in \mathbb{C}$ в качестве входных данных и выводит открытый текст $m \in P$.
4. $Evaluate_\epsilon$ принимает в качестве входных данных pk , операцию $\sigma \in \sigma_\epsilon$, и кортеж закодированных текстов $C = \langle c_1, \dots, c_t \rangle$, которые кодируют $M = \langle m_1, \dots, m_t \rangle$. C используется в качестве входных данных для σ , формирующей на выходе закодированный текст $C' \in \mathbb{C}$, такой что $Decrypt_\epsilon(sk, C') = \sigma(M)$.

Таким образом, для C , который кодирует M , желаемый функционал, реализуемый блоком $Evaluate_\epsilon$, заключается в получении закодированного текста $C' \stackrel{\Gamma B}{\longleftarrow} Evaluate_\epsilon(pk, \sigma, C)$, который кодирует $\sigma(M)$ под ключом pk , где $\sigma(M)$ определяет результат операции σ от множества незакодированных сообщений M . Свойства корректности и компактности являются основополагающими в формальном определении схемы полностью гомоморфного кодирования. Их можно достичь средствами четырех основных блоков, определенных выше.

Определение 1.5.1. Корректность. *Схема гомоморфных вычислений ϵ корректна для операций σ_ϵ , если для любой пары ключей $(sk, pk) \stackrel{\Gamma B}{\longleftarrow} KeyGen_\epsilon(\lambda)$, любой операции $\sigma \in \sigma_\epsilon$, любых закодированных текстов $C = \langle c_1, \dots, c_t \rangle$ и открытых текстов $M = \langle m_1, \dots, m_t \rangle$, где $c_i \stackrel{\Gamma B}{\longleftarrow} Encrypt_\epsilon(pk, m_i)$, выполняется следующее соотношение*

$$\text{Если } C' \stackrel{\Gamma B}{\longleftarrow} Evaluate_\epsilon(pk, \sigma, C), \text{ тогда } Decrypt_\epsilon(sk, C') = \sigma(M). \quad (1.7)$$

Определение 1.5.2. Компактность. *Схема гомоморфных вычислений ϵ является компактной, если существует такой многочлен f , что для любого значения параметра безопасности λ , результат формируемый на выходе блоком $Decrypt_\epsilon$ имеет размер не более чем $f(\lambda)$. В случае компактности ϵ и корректности для всех операций из σ_ϵ , говорят, что ϵ компактно вычисляет σ_ϵ .*

Определение 1.5.3. Полностью гомоморфное кодирование. *Схема гомоморфных вычислений ϵ полностью гомоморфна, если она компактно вычисляет все операции, т.е.*

$$Decrypt_\epsilon(sk, Evaluate_\epsilon(pk, \sigma, C)) = \sigma(M), \quad (1.8)$$

$\forall \sigma \in \sigma_\epsilon$.

Высокие накладные расходы на выполнение сложения и умножения делают полностью гомоморфные вычисления непрактичным в реальных приложениях.

В настоящее время стандарт по гомоморфным вычислениям [238] включает в себя следующие схемы: Brakerski-Gentry-Vaikuntanathan (BGV) [167], Brakerski [166] / Fan-Vercauteren [202] (BFV), Gentry-Sahai-Waters (GSW) [216], YASHE [248], Hoffstein-Pipher-Silverman (HPS) [237], López-Tromer-Vaikuntanathan (LTV) [274] и Cheon-Kim-Kim-Song (CKKS) [239].

Гомоморфные вычисления можно разделить на два класса в зависимости от типов обрабатываемых данных:

1. *Целые числа.* Гомоморфные схемы BGV и BFV поддерживают только целые числа, что затрудняет их использование для алгоритмов машинного обучения, нейронных сетей или компьютерного зрения. С другой стороны, они позволяют реализовать целочисленные вычисления для чувствительных к ошибкам алгоритмов обработки данных, таких как матричные операции [63].
2. *Числа с фиксированной запятой,* поддерживаемые CKKS. CKKS используется для проектирования нейронных сетей [356]. Однако, при использовании CKKS возникают ошибки, влияние которых невозможно устранить в процессе декодирования, поэтому применение CKKS для чувствительных к ошибкам алгоритмов может привести к значительному искажению результата, например, в матричных операциях. Как показано в [226], наличие ошибки равной 0.01 в одном значении матрицы третьего порядка может привести к тому, что декодированное при использовании CKKS значение определителя матрицы будет содержать ошибку, которая не позволит получить никакой информации об истинном значении определителя.

Для исследования схем гомоморфных вычислений была выбрана библиотека криптографии на решетках Lattigo [265], написанная на языке Golang. Указанная библиотека содержит необходимый для реализации схем гомоморфного кодирования набор функций, а ее структура позволяет выполнять всестороннее исследование схем, экспериментировать как с полной схемой (или схемами), так и с отдельными ее (их) компонентами. Все схемы удовлетворяют общепринятым стандартам безопасности, параметры которых представлены ниже.

N : размерность кольца, определяющая степень циклотомического многочлена и число коэффициентов многочленов открытого/закодированного текста. Кольцо всегда является степенью числа 2. Параметр N выполняет роль регулятора: с ростом N увеличивается безопасность, но уменьшается производительность, и наоборот. Настройка параметра N требует особого внимания, т.к. от него зависит корректность работы схемы.

Q : модуль закодированного текста. В Lattigo он выбирается как произведение малых взаимно простых модулей q_i , обеспечивающих $q_i \equiv 1 \pmod{2^N}$, что делает возможным его представление в RNS и NTT (Number Theoretical

Transform). Модули q_i выбираются из диапазона от 50-битных до 60-битных чисел, что обеспечивает наилучшую производительность. Q влияет как на безопасность, так и на производительность: если N фиксировано, то увеличение Q ведет к снижению как безопасности, так и производительности. Параметр Q тесно связан с N и должен тщательно выбираться, исходя из предполагаемых условий использования схемы.

σ^2 : дисперсия, используемая для многочленов ошибок. Этот параметр тесно связан с безопасностью схемы (с увеличением σ^2 возрастает безопасность).

Стоит также отметить, что все схемы, представленные в стандарте [238], имеют общую базу гомоморфных вычислений. Схемы реализуют арифметику над пространствами открытых и закодированных текстов. Пространство открытых текстов и пространство закодированных текстов совместно используют круговое кольцо, которое в данной работе обозначено как \mathbb{H} .

$$\mathbb{H} = \mathbb{Z}_Q[X] / (X^N + 1), \quad (1.9)$$

где $\mathbb{Z}_Q[X]$ – круговое кольцо, N – степень двойки.

1.5.3.1 СККС схема для работы с закодированными вещественными числами

Впервые схема СККС была предложена Cheon, Kim, Kim и Song в 2017 году. СККС является RNS-ускоренной версией схемы гомоморфного кодирования, реализующей арифметику приближенных чисел. Кроме того, данная схема может быть использована для реализации арифметики над $\mathbb{C}^{\frac{N}{2}}$, сопоставимой со структурой кругового кольца

$$\mathbb{C}^{\frac{N}{2}} \leftrightarrow \mathbb{H}, \quad (1.10)$$

однако, в случае СККС основным параметром является *scale*.

Scale определяет максимально возможное значение открытого текста. Так как в СККС комплексные числа кодируются с помощью многочлена с целыми коэффициентами, то исходные значения масштабируются во время кодирования, после чего округляются до ближайшего целого числа. Так же, как и N ,

scale равен степени двойки, на которую значения умножаются во время кодирования. Данный параметр влияет на точность вывода и на максимально допустимую глубину для используемого параметра безопасности. Конфигурация параметров для СККС сильно зависит от приложения. Требуется предварительный анализ схемы и закодированных данных, т.к. неправильно подобранная конфигурация параметров может снизить быстродействие схемы при выполнении определенных операций.

1.5.3.2 BFV схема для работы с закодированными целыми числами

Пакет BFV является RNS-ускоренной реализацией версии Fan-Vercauteren [202] масштабно-инвариантной схемы гомоморфного кодирования Brakerski [166]. Схема обеспечивает арифметику над \mathbb{Z}_t^N , поэтому пакетное кодирование данной структуры, аналогично схеме СККС, можно реализовать с использованием кругового кольца

$$\mathbb{Z}_t^N \leftrightarrow \mathbb{H}. \quad (1.11)$$

Если СККС имеет только один независимый параметр, то схема BFV характеризуется целым набором следующих независимых параметров:

P : расширенный модуль закодированного текста, который применяется исключительно для операции *Mul* (умножение) и подобных ей, определяется как произведение малых взаимно простых модулей p_j и выбирается таким образом, чтобы $P \cdot Q > Q^2$ с небольшим запасом (≈ 20 бит). Выполнение данного неравенства достигается посредством использования одного меньшего модуля в произведении Q . Отметим, что P не влияет на безопасность схемы.

t : модуль открытого текста, определяющий максимально возможное значение коэффициента открытого текста. Если вычисление приводит к более высокому по сравнению с t значению, то вычисленное значение уменьшается по модулю открытого текста. Параметр t должен удовлетворять условию $t \equiv 1 \pmod{2^N}$ и не влияет на безопасность.

1.6 Выводы по первой главе

В первой главе представлен проблемный обзор угроз информационной безопасности в современных распределенных средах хранения и обработки данных. Распределенные системы характеризуются высоким уровнем неопределенности, связанной с нестационарностью и динамическим изменением количества и состава их узлов и компонентов, что отрицательно влияет на эффективность вычислений, создавая дополнительные трудности в решении проблем планирования. Таким образом, требуется разработка новых стратегий управления ресурсами для эффективного решения проблемы неопределенности.

В рамках сформулированной цели исследования построена структурная модель обработки данных в распределенных средах, объединяющая в себе краевые, туманные и облачные вычисления. Выделено пять уровней передачи, хранения и обработки данных. Для каждого из уровней выявлены основные угрозы безопасности данных и проанализированы современные методы уменьшения вероятности кражи, потери или искажения данных. Установлено, что в распределенных средах в условиях повышенной неопределенности фундаментальные подходы к снижению рисков конфиденциальности, целостности и доступности, использующие механизмы репликации данных, резервного копирования, структуры доступа, избыточную систему остаточных классов, коды стирания, регенерационные коды недостаточно эффективны и должны быть усовершенствованы. Предложено использование вышеперечисленных механизмов, адаптированных, оптимизированных и интегрированных в концепцию мультиоблачного хранения и обработки данных. Показано, что использование мультиоблачного подхода позволяет существенно повысить надежность распределенных систем и снизить вероятности потери, утечки информации, отказа в доступе в течение длительного времени.

Модель, наиболее адекватную мультиоблачному подходу с точки зрения организации распределенного хранения и обработки данных, реализуют пороговые структуры доступа. Однако, выбор алгоритмов реализации пороговых структур доступа представляет собой сложную многокритериальную задачу, т.к. должен осуществляться не только на основе стандартных метрик, таких как сложность, скорость выполнения и т.д., но и учитывать особенности распределенной среды, связанные, в первую очередь, с высоким уровнем неопределенно-

сти. В работе приведено обоснование выбора алгоритмов реализации пороговой структуры доступа с точки зрения обеспечения конфиденциальности, надежности хранения, возможности осуществления контроля корректности операций с данными и вводимой избыточности. Представлена модификация пороговой структуры доступа с учетом предложенных, разработанных и реализованных методов выполнения основных операций. Разработанные методы всесторонне протестированы, доказана и продемонстрирована их эффективность.

Возможность реализации гомоморфных вычислений является наиболее существенным аспектом при выборе структуры доступа, т.к. помимо возможности распределенной обработки гомоморфный код позволяет обеспечить безопасность обрабатываемых данных за счет обработки в закодированном виде. Различают два подхода к реализации гомоморфных вычислений: над кольцом вычетов с делителями нуля и над полем.

Алгоритмы гомоморфных вычислений над кольцом вычетов с делителями нуля могут быть использованы при построении защищенной системы обработки данных, но при этом необходимо учитывать высокую вероятность взлома системы с помощью модифицированной атаки открытым текстом. Данная проблема, наряду с угрозой сговора облаков при использовании мультиоблачного подхода, является основной и успешно решается в рамках данного исследования. Алгоритмы гомоморфных вычислений над полем входят в проект стандарта по гомоморфным вычислениям от 2018 года и могут быть классифицированы либо как целочисленные, либо как вещественные, в зависимости от типа обрабатываемых данных.

Существующие схемы, построенные с использованием гомоморфизма колец, позволяют выполнять арифметические операции сложения и умножения закодированных чисел. В зависимости от применяемой схемы гомоморфных вычислений меняются подходы к выполнению указанных операций. Однако, общей проблемой гомоморфных вычислений, независимо от используемого подхода (гомоморфные вычисления над кольцом вычетов с делителями нуля или над полем) и вида применяемых схем, является высокая сложность реализации и, как следствие, низкая скорость обработки данных. Наибольшие задержки наблюдаются при выполнении вычислительно сложных операций, к ним относятся операции определения знака числа и сравнения чисел. Эффективность выполнения указанных операций можно повысить путем разработки новых методов и оптимизации соответствующих алгоритмов вычисления приближенного

(с необходимой точностью) или точного (когда это возможно) значения результата данных операций с сохранением свойства гомоморфности. Повышение эффективности вычисления результатов проблемных для гомоморфных вычислений операций равносильно ускорению процедуры кодирования/декодирования в целом, поэтому разработке методов выполнения операций определения знака числа и сравнения чисел уделено особое внимание в данном исследовании.

Глава 2. МОДЕЛИ И МЕТОДЫ ОБРАБОТКИ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ГОМОМОРФНЫХ ВЫЧИСЛЕНИЙ

2.1 Гомоморфные вычисления над кольцом вычетов с делителями нуля, основанные на избыточной системе остаточных классов

Концепция облачных вычислений предусматривает динамическое распределение ресурсов. Следовательно, помимо существующих проблем безопасности и надежности облачных и GRID-вычислений, возникают новые проблемы безопасности и надежности [298, 349], среди которых сговоры [223], атаки на виртуальную машину [244], атаки на ключи синхронизации [367] и т.д. Чтобы уменьшить неопределенность и минимизировать риски нарушения безопасности данных и отказа в доступе к данным, целесообразно использовать коды локализации и исправления ошибок основанные на Избыточной Системе Остаточных Классов (Redundant Residue Number System – RRNS).

В RRNS исходное число представляется остатками от деления на модули. Разрядность модулей меньше разрядности исходного числа, поэтому число разделяется на несколько меньших чисел, которые не зависят друг от друга. RRNS – непозиционная система счисления, поэтому обработка остатков может производиться независимо и одновременно, что упрощает и ускоряет вычисления.

Пусть p_i – попарно взаимно простые числа, называемые модулями RRNS, $i = \overline{1, n}$. Модули RRNS на два непересекающихся подмножества из k и r модулей, $n = k + r$. Наименьшие k модулей определяют рабочий диапазон RRNS $P = \prod_{i=1}^k p_i$. Оставшиеся r модулей называются контрольными и используются для обнаружения и исправления ошибок данных, представленных в RRNS.

Исходные данные можно представить в виде целого числа $S \in [0, P)$. S представляется в RRNS кортежем $S \xrightarrow{RRNS} (s_1, s_2, \dots, s_n)$, где $s_i = |S|_{p_i}$ – остаток от деления S на p_i . Схема (k, n) RRNS позволяет восстановить исходные данные S , используя любые k из n остатков, при условии, что все k остатков корректны. В случае искажения остатков, RRNS позволяет обнаружить r и исправить $\lfloor \frac{r}{2} \rfloor$ ошибок. Указанный метод введения избыточности позволяет построить надежную систему обработки данных с обнаружением и исправлением множественных ошибок [11, 85, 145, 179].

Одним из главных преимуществ RRNS является возможность выполнять сложение, умножение и вычитание параллельно и независимо для каждого остатка [351]. Арифметические операции выполняются без переносов между разрядами RRNS s_i , что, с одной стороны, позволяет производить параллельную обработку данных в облаках, а с другой – обеспечивает их конфиденциальность.

Использование контрольных модулей в RRNS обеспечивает надежность при долгосрочном хранении данных и позволяет осуществлять контроль результатов вычислений [145]. Для обнаружения и исправления ошибок в RRNS используется метод модулярных проекций [11], который эквивалентен преобразованию из RNS в двоичную систему счисления. Однако, количество модулярных проекций, которые необходимо вычислить, при увеличении кратности ошибки растет в геометрической прогрессии, тогда как вычисление даже одной проекции является сложным в вычислительном отношении алгоритмом.

Для оптимизации алгоритма [11] был предложен новый метод обнаружения и локализации ошибок в RRNS, основанный на приближенном ранге числа и характеристической функции. Предложенный метод позволяет снизить вычислительную сложность алгоритма декодирования, но требует большего объема памяти для хранения характеристической функции в табличной форме.

Если учесть, что RRNS – это не только код обнаружения, локализации и исправления ошибок, но и по-сути схема Mignotte, то RRNS можно использовать для обеспечения безопасности данных.

В работе 2011 года Gomathisankaran и др. [223] показали, что схема Mignotte является полностью гомоморфной и позволяет обрабатывать данные в закодированном виде.

Для решения проблемы сговора в облаках предлагается AC-RRNS (Anti-Collision – RRNS) на основе модификации схем Asmuth-Bloom [151] и Mignotte [280]. Достоинство предложенной схемы заключается в снижении избыточности данных по сравнению со схемой Asmuth-Bloom и обеспечении вычислительной безопасности. В отличие от схемы Mignotte, AC-RRNS позволяет обеспечить вычислительную безопасность данных с сохранением значений избыточности данных на том же уровне.

В предлагаемом решении для обеспечения надежности хранения данных используются корректирующие свойства RRNS. Большинство научных работ по RRNS рассматривают обнаружение и исправление только одной ошибки из-за высокой надежности современных компьютерных технологических решений.

Однако, этот тезис неприменим к распределенным системам хранения, особенно для больших данных, поскольку вероятность отказа нескольких вычислительных узлов высока. Следовательно, для построения надежных систем хранения требуется механизм исправления множественных (многократных) ошибок на основе RRNS. Эффективный механизм обнаружения, локализации и исправления многократных ошибок в RRNS, использующий приближенное значение ранга числа, был предложен в работе [11].

Как уже отмечалось, беззнаковое целое число S в диапазоне P может быть представлено остатками от деления (наименьшими неотрицательными вычетами) S на p_i . Чтобы представить знаковое целое число S в RNS, P делится на два поддиапазона [325]. Нижняя и верхняя половины диапазона используются для представления положительных и отрицательных целых чисел соответственно [351]. Представление чисел в RNS позволяет заменять операции над большими числами операциями над маленькими числами, которые обрабатываются параллельно и независимо.

Эффективность реализации систем на основе RNS во многом зависит от используемых наборов модулей. Есть несколько подходов к их выбору. Например, в работе [306] исследован набор модулей специального вида $\{2^n - 1, 2^n, 2^n + 1\}$. Такие наборы позволяют разрабатывать эффективные алгоритмы преобразования из двоичной системы счисления в RNS и обратно, а также эффективно выполнять арифметические операции. Phatak и Houston в 2016 году [308] представили подход к организации высокопроизводительных вычислений на основе RNS для графического процессора (Graphics Processing Unit – GPU), который использует простые числа определенной формы, хорошо подходящие для хранения и обработки больших данных. Однако, подобные наборы, как правило, ограничены 3-5 модулями и не позволяют эффективно масштабировать систему. Кроме того, арифметические операции для каждого модуля разрабатываются индивидуально, и агрегирование нового модуля требует оптимизации преобразования из RNS в двоичную систему счисления.

Среди известных алгоритмов преобразования из RNS в двоичную систему счисления можно выделить переход к представлению в смешанной позиционной системе счисления (Mixed Radix Conversion – MRC), Китайскую теорему об остатках (Chinese Remainder Theorem – CRT), гибрид CRT и MRC [160], новые версии Китайской теоремы об остатках (new CRT – nCRT) [362] и их модификации.

Высокая вычислительная сложность обратного преобразования – причина, по которой исследователи ищут методы, приближающие результат данной операции. Вот некоторые из них: приближенные версии CRT (approximate CRT – aCRT) [28, 358], функция ядра [170], функция частного [326], диагональная функция [195, 283], монотонная функция Pirlo и Impedovo [312].

2.2 Схема WA-MRC-RRNS надежного и безопасного хранения и обработки данных на основе структуры доступа

В диссертационном исследовании предлагается взвешенная пороговая структура доступа WA-MRC-RRNS – расширение WA-RRNS [88], использующая MRC для ускорения процедуры декодирования. MRC при декодировании использует малоразрядные операнды, что позволяет избежать использования арифметики многоразрядных чисел (длинной арифметики) при программной реализации, тогда как для WA-RRNS и WA-AR-RRNS длинная арифметика необходима для вычисления остатка от деления и умножения на рабочий диапазон RRNS соответственно. Таким образом, за счет ухода от длинной арифметики достигается преимущество в эффективности реализации WA-MRC-RRNS по сравнению с WA-RRNS и WA-AR-RRNS.

Согласно предложенной схеме, исходные данные раскладываются на набор коротких частей (долей) (рис. 1.2*г*). В WA-MRC-RRNS, в отличие от обычной WA-RRNS, каждое CSP имеет разное количество коротких долей. В этом разделе доказано, что такой подход снижает вероятность потери информации. Скорость кодирования/декодирования при этом снижается, однако, она все равно выше чем при использовании WA-AR-RRNS.

В таблице 2 введены необходимые для доказательства обозначения. В RRNS надежность системы зависит от параметров k и n , количества остатков, достаточного для восстановления, и общего количества остатков соответственно. Их выбор обеспечивает необходимый уровень вычислительной безопасности и конфиденциальности.

Червяков и др. [11] показали, что надежность системы зависит от r , где $r = n - k$, и от n . Чем больше значение n , тем надежнее система. С увеличе-

Таблица 2 — Обозначения используемые в схеме WA-RRNS

| | |
|---------------------------|--|
| N | Количество CSP |
| $n_i \geq 1$ | Количество долей, хранящееся в i -ом CSP |
| $n_v = (n_1, \dots, n_N)$ | Кортеж n_i |
| $n = \sum_{i=1}^N n_i$ | Общее количество модулей RRNS и долей |
| $k \leq n$ | Пороговое значение для классической схемы |
| $K \leq n$ | Пороговое значение для взвешенной схемы |
| $r = n - k$ | Количество контрольных (избыточных) модулей RRNS |
| (p_1, \dots, p_n) | Модули RRNS |
| $P = \prod_{i=1}^k p_i$ | Динамический диапазон RRNS |
| w_i | Вес i -го модуля RRNS |
| $W = \sum_{i=1}^K w_i$ | Пороговый вес |
| $Pr(k, n)$ | Вероятность потери данных пороговой схемы |
| $Pr(n_v, K, N)$ | Вероятность потери данных взвешенной пороговой схемы |

нием значения r , надежность системы так же возрастает за счет увеличения избыточности.

Взвешенная пороговая схема (n_v, K, N) определяется следующим образом.

Пусть для $N \geq 2$, $n_v = (n_1, n_2, \dots, n_N)$ – набор натуральных чисел, а K – натуральное число, удовлетворяющее условию $2 \leq K \leq \sum_{i=1}^N n_i$.

Кортеж n_v определяет количество долей в каждом из N хранилищ. Если все элементы n_v равны, т.е. количество долей в каждом CSP одинаково, то при обозначении схемы кортеж n_v заменяется значением n_i (как это сделано на рис. 1.2б)). Пороговое значение K – количество долей, необходимых для восстановления исходных данных, а N – общее количество CSP.

Поскольку количество простых чисел в диапазоне от 2 до x приблизительно равно $\pi(x) \approx \frac{x}{\ln x}$, достаточным условием существования набора модулей RNS для хранения данных по предложенной схеме является.

$$n = \sum_{i=1}^N n_i < \frac{l \cdot 2^{l-1} - 2^l}{l^2} = \frac{2^{l-1}(l-2)}{l^2}, \quad (2.1)$$

где l – длина модуля RNS в битах.

Данная оценка используется при $l \leq 32$, например, количество простых чисел длины 8 бит равно 23, 16 бит – 3030, 24 бита – 513708 и 32 бита – 98182656.

Если $l > 32$ используется следующая оценка

$$n = \frac{2^{l-1} (l - 2)}{l^2}. \quad (2.2)$$

Пусть i -е облачное хранилище имеет n_i долей и вероятность ошибки Pr_i . Тогда объем хранимых данных в i -м облаке пропорционален n_i .

Если существует $n_i \geq k$, то i -й облачный провайдер может восстановить закодированные данные и нарушить правила конфиденциальности. С другой стороны, использование большого количества долей позволяет реализовать процедуру исправления ошибок внутри CSP, что существенно увеличивает надежность системы в целом.

Задача состоит в том, чтобы распределить доли так, чтобы обеспечить максимальную надежность, не создав при этом угроз безопасности хранения данных. Учитывая ограничение $n_i < K$, продиктованное требованием конфиденциальности, и то, что с увеличением количества долей в каждом CSP надежность повышается, предлагается использовать взвешенную схему с параметрами $n_v = (K - 1, K - 1, \dots, K - 1)$.

Продемонстрируем преимущества данной схемы, доказав теорему, утверждающую, что вероятность потери данных при использовании предлагаемой взвешенной схемы меньше, чем вероятность потери данных при использовании традиционной пороговой схемы.

Теорема 2.2.1. $Pr(k, n) \geq Pr(n_v = (K - 1, K - 1, \dots, K - 1), K, N)$ для любого $k = K \geq 2$.

Доказательство. Для доказательства рассмотрим два случая, которые могут привести к потере данных. Отметим, что случаи 1-2 являются обобщающими, т.е. используемые значения вероятностей ошибок в случаях 1 и 2 гипотетически учитывают все возможные причины, приводящие к ним. Несмотря на то, что на этапе проектирования системы данные вероятностные значения получить практически невозможно (возможно лишь оценить их с некоторой точностью), они могут быть использованы при сравнении схем.

Случай 1. Потеря данных происходит в результате сбоя облачного сервиса. Обозначим вероятность возникновения ошибки на i -ом CSP через Pr_i , а вероятность потери данных по причине сбоя нескольких CSP (случай 1) через Pr_{C_1} .

Случай 2. Потеря данных происходит в результате ошибки одной из долей облачного сервиса. Обозначим вероятность потери данных в результате ошибки одной из долей i -го CSP (случай 2) через Pr_{C_2} .

Потеря данных в случае 1 может произойти в результате одновременного сбоя $N - 1$ CSP.

$$Pr_{C_1}(n_v, K, N) = \prod_{i=1}^n Pr_i + \sum_{i=1}^N (1 - Pr_i) \prod_{j=1, j \neq i}^N Pr_j. \quad (2.3)$$

Потеря данных в случае 2 может произойти, если

$$(K - 1) \cdot N - (K - 1) = (N - 1) \cdot (K - 1) \quad (2.4)$$

долей содержат ошибки.

Следовательно, вероятность потери данных

$$Pr_{C_2}(n_v, K, N) = \sum_{(N-1) \cdot (K-1)}^{N \cdot (K-1)} \binom{N \cdot (K-1)}{i} \cdot Pr_{ch}^i \cdot (1 - Pr_{ch})^{N \cdot (K-1) - i}, \quad (2.5)$$

где Pr_{ch} – вероятность ошибки в i -ой доле, а $\binom{n}{k} = C_n^k$ – число сочетаний из n по k .

Пусть в набор I входят CSP с индексами i_1, \dots, i_{k_0} , которые можно использовать для извлечения данных $\left(\sum_{j=1}^{k_0} n_{i_j} \geq K\right)$, а \bar{I} – набор CSP с индексами $\bar{i}_1, \dots, \bar{i}_{k_0}$, которые нельзя использовать для получения данных $\left(\sum_{j=1}^{k_0} n_{\bar{i}_j} < K\right)$, тогда имеют место следующие неравенства

$$Pr_{C_1}(k, n) \geq Pr_{C_1}(n_v, K, N) = \sum_I \prod_{i \in \bar{I}} (1 - Pr_i) \prod_{i \notin \bar{I}} Pr_i, \quad (2.6)$$

\bar{I} определяется пороговой структурой доступа. Для второго случая,

$$Pr_{C_2}(k, n) \geq Pr_{C_2}(n_v, K, N) = \sum_{i=n-k+1}^n \binom{n}{i} Pr_{ch}^i (1 - Pr_{ch})^{n-i}. \quad (2.7)$$

Из (2.6) и (2.7) следует, что

$$\begin{aligned} Pr(k, n) &= Pr_{C_1}(k, n) + Pr_{C_2}(k, n) \\ &\geq Pr_{C_1}(n_v, K, N) + Pr_{C_2}(n_v, K, N) \\ &= Pr(n_v, K, N). \end{aligned} \quad (2.8)$$

Теорема доказана. \square

Таблица 3 — Вероятность отказа CSP (CloudHarmony [324])

| i | Облако | T_D (мин.) | Pr_i | λ_i |
|-----|---------------|--------------|----------|-------------|
| 1 | Joyent | 34 | 0.000065 | 3.90E-03 |
| 2 | AWS | 150 | 0.000285 | 1.71E-02 |
| 3 | Azure | 649 | 0.001235 | 7.41E-02 |
| 4 | Google | 694 | 0.001320 | 7.92E-02 |
| 5 | Digital Ocean | 764 | 0.001454 | 8.72E-02 |
| 6 | Rackspace | 770 | 0.001465 | 8.79E-02 |
| 7 | IBM's | 1020 | 0.001941 | 1.16E-01 |
| 8 | CenturyLink | 1889 | 0.003594 | 2.16E-01 |

2.2.1 Вероятность потери данных при хранении с использованием схемы WA-MRC-RRNS

На основе информации о времени простоя поставщиков общедоступных облаков IaaS (Infrastructure as a Service), предоставленной CloudHarmony [171, 324], в таблице 3 резюмируется вероятность отказа различных облачных сервисов. CloudHarmony отслеживает состояние работоспособности поставщиков услуг, тестируя экземпляры рабочих нагрузок в общедоступных облаках и постоянно проверяя их исполняемость. Данная статистика не дает целостного представления о доступности облака, полной картины статистики простоев облака и информации о типах сбоев из-за ограниченной возможности мониторинга всех облачных сервисов зонами доступности лишь в нескольких регионах. Многие из основных сбоев, освещенных в СМИ, не отражены в отчете. Однако, он все равно представляет высокую ценность для анализа надежности.

Несмотря на то, что доступность облаков в последние годы увеличилась, простои неизбежны. В таблице 3 показано время T_D в минутах, когда услуги восьми облачных провайдеров были недоступны в течение года. Вероятность потери данных Pr_i вычисляется по геометрическому определению вероятности (соотношению мер) $Pr_i = \frac{T_D}{525600}$, где $525600 = 365 \cdot 24 \cdot 60$ – количество минут в году, T_D – общее время отказа в доступе, а λ_i – частота отказов в час.

Основываясь на этих вероятностях, можно приблизительно оценить надежность CSP и использовать данную оценку для анализа имеющихся и новых решений проблем безопасности.

Для оценки надежности системы хранения данных, как правило, используется вероятность отказа на заданном временном интервале. Вероятность безотказной работы (безотказности) i -го CSP за время t обозначается $Pr_i(t)$. Для его расчета используется закон экспоненциального распределения $Pr_i(t) = e^{-\lambda_i t}$, где λ_i – частота отказов в час, а t – время (количество часов).

Рисунок 2.1 демонстрирует вероятность безотказной работы $Pr_i(t)$, рассчитанную для восьми хранилищ на основе параметров в таблице 3. Из графика

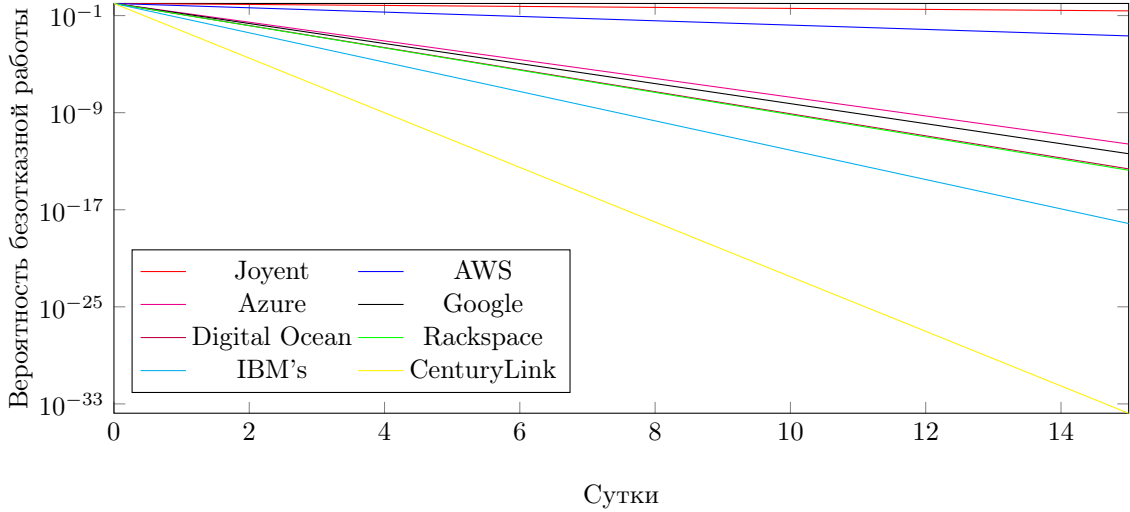


Рисунок 2.1 — Вероятность безотказной работы хранилищ

ков на рисунке 2.1 можно сделать вывод, что вероятность безотказной работы снижается экспоненциально. Таким образом, использование единого хранилища не гарантирует надежности в течение длительного периода времени. Например, через 15 дней вероятность безотказной работы CSP Joyent снижается до 0.247. Одновременный сбой сразу нескольких хранилищ является существенно менее вероятным событием чем сбой одного хранилища, согласно теореме о произведении вероятностей независимых событий, поэтому целесообразно использовать мультиоблачный подход для организации распределенного хранения данных.

Введем следующие обозначения. Пусть $I = \{i_1, i_2, \dots, i_k\}$ – набор из k CSP, доступных для загрузки. Каждое CSP характеризуется весом w_i , где $i = \overline{1, k}$. Любое из доступных для загрузки CSP может получить доступ к исходным данным, если вес w_i , равный сумме весов долей, хранящихся на i -ом CSP, равен или превышает пороговое значение W , $w_i \geq W$. Структура доступа — это набор всех возможных сочетаний элементов I , суммарный вес которых равен или превышает порог W , обозначается $\Gamma = \{I_1, I_2, \dots, I_l\}$. Например, для

классической пороговой схемы (2, 3), позволяющей восстанавливать данные с использованием любых двух ресурсов из трех $\Gamma = \{(1, 2), (1, 3), (2, 3), (1, 2, 3)\}$.

Вероятность безотказной работы всей системы за время t рассчитывается по следующей формуле

$$Pr_{\Gamma}(t) = \sum_{I \in \Gamma} \left(\prod_{i \in I} Pr_i(t) \prod_{i \notin I} (1 - Pr_i(t)) \right). \quad (2.9)$$

С учетом того, что $0 \leq Pr_i(t) \leq 1$, произведение $\prod_{i \notin I} (1 - Pr_i(t))$ удовлетворяет следующему неравенству

$$1 - \sum_{i \notin I} Pr_i(t) < \prod_{i \notin I} (1 - Pr_i(t)) < 1 - \sum_{i \notin I} Pr_i(t) + \frac{1}{2} \left(\sum_{i \notin I} Pr_i(t) \right)^2. \quad (2.10)$$

Следовательно, мы можем аппроксимировать значение $\prod_{i \notin I} (1 - Pr_i(t))$ следующим образом

$$\prod_{i \notin I} (1 - Pr_i(t)) \approx 1 - \sum_{i \notin I} Pr_i(t), \quad (2.11)$$

при этом из выражения (2.10) следует, что ошибка аппроксимации не превышает следующего значения

$$\frac{1}{2} \left(\sum_{i \notin I} Pr_i(t) \right)^2. \quad (2.12)$$

Подставляя (2.11) в (2.9), получаем

$$Pr_{\Gamma}(t) = \sum_{I \in \Gamma} \left(\prod_{i \in I} Pr_i(t) \left(1 - \sum_{i \notin I} Pr_i(t) \right) \right). \quad (2.13)$$

Учитывая, что $Pr_i(t) = e^{-\lambda_i t}$, вероятность безотказной работы хранилища

$$Pr_{\Gamma}(t) = \sum_{I \in \Gamma} \left(e^{-t \sum_{i \in I} \lambda_i} \left(1 - \sum_{i \notin I} e^{-\lambda_i t} \right) \right). \quad (2.14)$$

Учитывая, что частота отказов в час λ_i известна, вероятность отказа системы зависит от структуры доступа, которая однозначно определяется весами $\{w_1, w_2, \dots, w_N, W\}$

$$w_i = \frac{e^{-\lambda_i}}{\sum_{i=1}^N e^{-\lambda_i}} \cdot R, \quad (2.15)$$

Таблица 4 — Вероятность потери данных при использовании схемы MRC-RRNS в течение года

| | MRC-RRNS | R | $T_{MRC-RRNS}$ | $Pr_{MRC-RRNS}(\Gamma_{MRC-RRNS})$ | $\lambda_{MRC-RRNS}$ |
|---|----------|-----|----------------|------------------------------------|----------------------|
| 1 | (2, 2) | 1 | 1.84E+02 | 3.50E-04 | 2.10E-02 |
| 2 | (2, 3) | 1.5 | 2.37E-01 | 4.51E-07 | 2.71E-05 |
| 3 | (2, 4) | 2 | 3.25E-04 | 6.18E-10 | 3.71E-08 |
| 4 | (2, 5) | 2.5 | 4.88E-07 | 9.28E-13 | 5.57E-11 |
| 5 | (2, 6) | 3 | 7.00E-10 | 1.33E-15 | 7.98E-14 |

где R – избыточность данных.

Максимальный w_i обеспечивает максимальный уровень надежности, следовательно, в это хранилище можно распределить больше данных.

Оценим вероятность потери данных при использовании схем MRC-RRNS и WA-MRC-RRNS. Вычисленные значения представлены в таблицах 4 и 5 соответственно. Избыточность данных рассчитывается как $R \approx \frac{n}{k}$ [11]. Обозначим набор авторизованных пользователей MRC-RRNS как $I_{MRC-RRNS}$. Структура доступа $\Gamma_{MRC-RRNS}$ определяет все возможные наборы элементов из $I_{MRC-RRNS}$ с суммарным весом, равным или превышающим пороговое значение W . Вероятность потери данных $Pr_{MRC-RRNS}(\Gamma_{MRC-RRNS})$ вычисляется по следующей формуле

$$Pr_{MRC-RRNS}(\Gamma_{MRC-RRNS}) = \sum_{I \in \Gamma_{MRC-RRNS}} \left(\prod_{i \in I} Pr_i \prod_{i \notin I} (1 - Pr_i) \right). \quad (2.16)$$

Время отказа в доступе $T_{MRC-RRNS}(\Gamma_{MRC-RRNS})$ в минутах в год определяется с использованием геометрического определения вероятности

$$T_{MRC-RRNS}(\Gamma_{MRC-RRNS}) = Pr_{MRC-RRNS}(\Gamma_{MRC-RRNS}) \cdot m_{year}, \quad (2.17)$$

где $m_{year} = 365 \cdot 24 \cdot 60 = 525600$ – количество минут в году. Частота ошибок в час рассчитывается с использованием следующей формулы

$$\lambda_{MRC-RRNS} = Pr_{MRC-RRNS}(\Gamma_{MRC-RRNS}) \cdot m_{hour}, \quad (2.18)$$

где $m_{hour} = 60$ – количество минут в часе.

В таблице 5 представлены вероятности потери данных $Pr_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS})$ при использовании схемы WA-MRC-RRNS

Таблица 5 — Вероятность потери данных при использовании схемы WA-MRC-RRNS в течение года

| | WA-MRC-RRNS | R | $T_{WA-MRC-RRNS}$ | $Pr_{WA-MRC-RRNS}$ | $\lambda_{WA-MRC-RRNS}$ |
|---|-------------|-----|-------------------|--------------------|-------------------------|
| 1 | (1, 2, 2) | 1 | 5.94E+03 | 1.13E-02 | 6.78E-01 |
| 2 | (1, 2, 3) | 1.5 | 6.68E-02 | 1.27E-07 | 7.62E-06 |
| 3 | (1, 2, 4) | 2 | 8.13E-06 | 1.55E-11 | 9.30E-10 |
| 4 | (1, 2, 5) | 2.5 | 7.37E-08 | 1.40E-13 | 8.40E-12 |
| 5 | (1, 2, 6) | 3 | 5.26E-11 | 1.00E-16 | 6.00E-15 |

в год.

$$Pr_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS}) = \sum_{I \in \Gamma_{WA-MRC-RRNS}} \left(\prod_{i \in I} Pr_i \cdot \prod_{i \notin I} (1 - Pr_i) \right). \quad (2.19)$$

Время отказа в доступе $T_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS})$ в минутах в году определяется с использованием геометрического определения вероятности

$$T_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS}) = Pr_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS}) \cdot m_{year}. \quad (2.20)$$

Частота ошибок в час рассчитывается с использованием следующей формулы

$$\lambda_{WA-MRC-RRNS} = Pr_{WA-MRC-RRNS}(\Gamma_{WA-MRC-RRNS}) \cdot m_{hour}. \quad (2.21)$$

Сравнительный анализ таблиц 4 и 5 позволяет утверждать, что предложенная схема WA-MRC-RRNS более надежна по сравнению с известной пороговой структурой доступа MRC-RRNS.

2.2.2 Стратегии распределенного хранения данных

Обсудим, как выбор хранилищ, из списка доступных для загрузки данных, влияет на надежность хранения. Рассмотрим и предложенную взвешенную

пороговую схему WA-MRC-RRNS и классическую пороговую схему на основе RRNS.

В пороговых схемах (k, n) AR-RRNS и (k, n) MRC-RRNS исходные данные делятся на n долей с возможностью восстановления данных по k любым долям. На каждое хранилище приходится одна доля. Порог равен k .

В предложенной взвешенной пороговой схеме (n_v, K, N) WA-MRC-RRNS в каждом хранилище имеется $n_i = K - 1$ коротких долей. Порог равен K .

В таблице 6 представлено количество CSP (N), количество долей, необходимых для восстановления данных (k и K для классической и взвешенной пороговых схем соответственно), и общее количество долей (n) для обеих схем. Рассмотрим два сценария для хранения данных в облаках:

1. *Пессимистичный сценарий*, когда для хранения данных используются только CSP с высокой вероятностью ошибки.
2. *Оптимистичный сценарий*, когда для хранения данных используются CSP с низкой вероятностью ошибки.

На рисунке 2.2 и в таблице 7 представлены годовые вероятности потери данных $Pr(k, n)$ и $Pr(n_v, K, N)$ для схем MRC-RRNS и WA-MRC-RRNS соответственно в этих двух сценариях.

Схема WA-MRC-RRNS демонстрирует более низкую вероятность потери данных по сравнению с традиционной пороговой структурой доступа MRC-RRNS. В большинстве случаев, даже в пессимистичном сценарии схема WA-MRC-RRNS имеет меньшую вероятность потери данных, чем схема MRC-RRNS в оптимистичном.

Если $N = 8$, вероятность потери данных при пессимистичном и оптимистичном сценариях для каждой из схем одинакова, т.к. используются все доступные хранилища (табл. 7).

На рисунке 2.3 показано преимущество схемы WA-MRC-RRNS по сравнению со схемой MRC-RRNS с точки зрения вероятности потери данных в пессимистичном сценарии для различных значений (k, n) .

Вероятность потери данных в пессимистичном сценарии при использовании схемы WA-MRC-RRNS в 777.44 раза меньше, чем при использовании MRC-RRNS для установки $(3, 4)$. Для установки $(8, 8)$ вероятность потери данных при использовании схемы WA-MRC-RRNS в $9.19 \cdot 10^{17}$ раз ниже, чем при использовании MRC-RRNS.

Таблица 6 — Количество долей в схемах AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS

| | | AR-RRNS, MRC-RRNS | WA-AR-RRNS, WA-MRC-RRNS | | |
|-----|-----|-------------------|-------------------------|-----|-------------------|
| N | k | n | $n_i = K - 1$ | K | $n = N \cdot n_i$ |
| 4 | 2 | 4 | 1 | 2 | 4 |
| | 3 | 4 | 2 | 3 | 8 |
| | 4 | 4 | 3 | 4 | 12 |
| 5 | 2 | 5 | 1 | 2 | 5 |
| | 3 | 5 | 2 | 3 | 10 |
| | 4 | 5 | 3 | 4 | 15 |
| | 5 | 5 | 4 | 5 | 20 |
| 6 | 2 | 6 | 1 | 2 | 6 |
| | 3 | 6 | 2 | 3 | 12 |
| | 4 | 6 | 3 | 4 | 18 |
| | 5 | 6 | 4 | 5 | 24 |
| | 6 | 6 | 5 | 6 | 30 |
| 7 | 2 | 7 | 1 | 2 | 7 |
| | 3 | 7 | 2 | 3 | 14 |
| | 4 | 7 | 3 | 4 | 21 |
| | 5 | 7 | 4 | 5 | 28 |
| | 6 | 7 | 5 | 6 | 35 |
| | 7 | 7 | 6 | 7 | 42 |
| 8 | 2 | 8 | 1 | 2 | 8 |
| | 3 | 8 | 2 | 3 | 16 |
| | 4 | 8 | 3 | 4 | 24 |
| | 5 | 8 | 4 | 5 | 32 |
| | 6 | 8 | 5 | 6 | 40 |
| | 7 | 8 | 6 | 7 | 48 |
| | 8 | 8 | 7 | 8 | 56 |

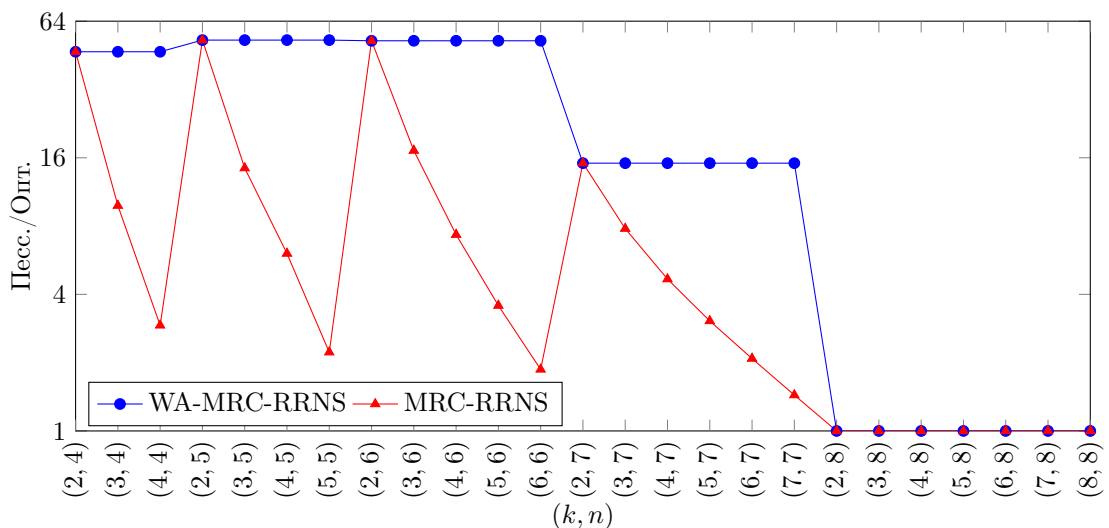
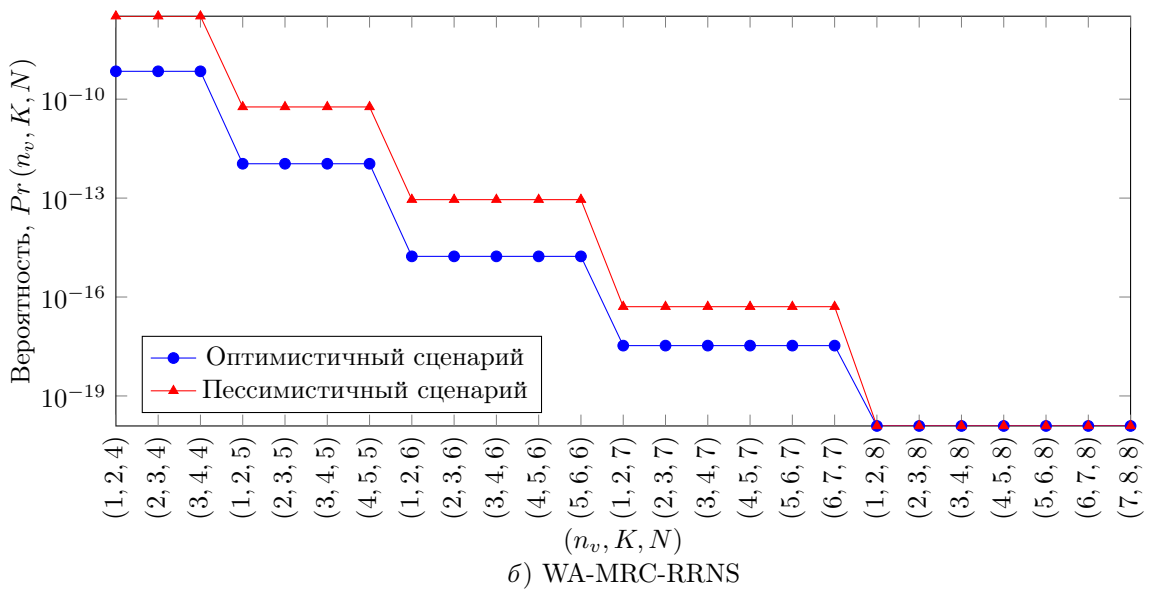
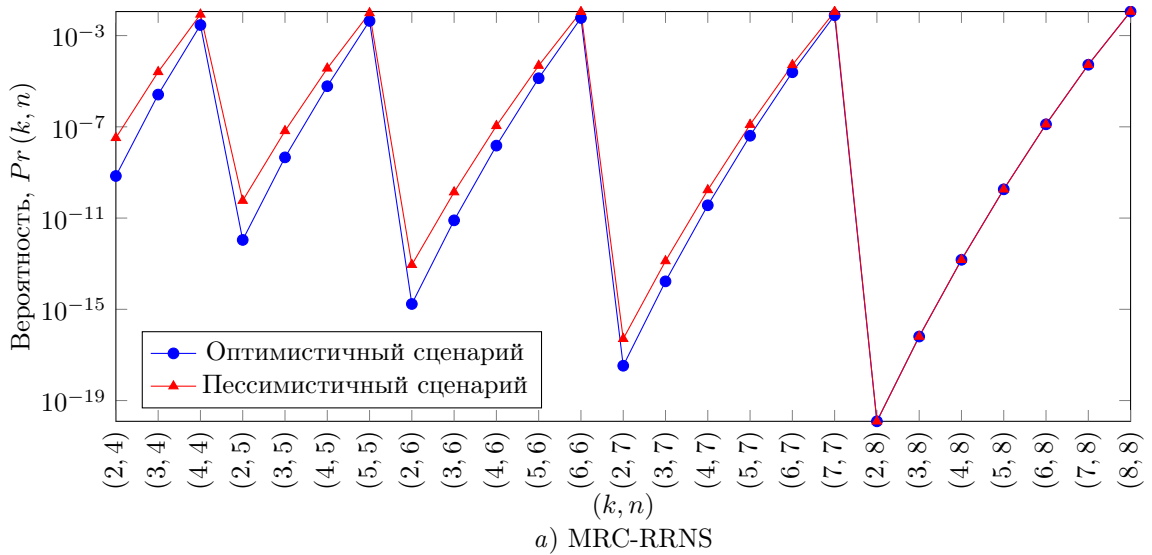


Рисунок 2.2 — а) Вероятность потери данных для схемы MRC-RRNS; б) Вероятность потери данных для схемы WA-MRC-RRNS; в) Отношение вероятностей потери данных при пессимистичном и оптимистичном сценариях для каждой из схем MRC-RRNS и WA-MRC-RRNS соответственно

Таблица 7 — Вероятность потери данных при использовании схем MRC-RRNS и WA-MRC-RRNS (пессимистичный (песс.) и оптимистичный (оптим.) сценарии) в течение года

| (k, n) | MRC-RRNS | | (n_v, K, N) | n | WA-MRC-RRNS | |
|----------|----------|----------|---------------|-----|-------------|----------|
| | Песс. | Оптим. | | | Песс. | Оптим. |
| (2, 4) | 3.28E-08 | 6.99E-10 | (1, 2, 4) | 4 | 3.28E-08 | 6.99E-10 |
| (3, 4) | 2.55E-05 | 2.59E-06 | (2, 3, 4) | 8 | 3.28E-08 | 6.99E-10 |
| (4, 4) | 8.47E-03 | 2.90E-03 | (3, 4, 4) | 12 | 3.28E-08 | 6.99E-10 |
| (2, 5) | 5.81E-11 | 1.10E-12 | (1, 2, 5) | 5 | 5.81E-11 | 1.10E-12 |
| (3, 5) | 6.60E-08 | 4.58E-09 | (2, 3, 5) | 10 | 5.81E-11 | 1.10E-12 |
| (4, 5) | 3.65E-05 | 6.03E-06 | (3, 4, 5) | 15 | 5.81E-11 | 1.10E-12 |
| (5, 5) | 9.76E-03 | 4.39E-03 | (4, 5, 5) | 20 | 5.81E-11 | 1.10E-12 |
| (2, 6) | 8.97E-14 | 1.71E-15 | (1, 2, 6) | 6 | 8.97E-14 | 1.71E-15 |
| (3, 6) | 1.37E-10 | 7.96E-12 | (2, 3, 6) | 12 | 8.97E-14 | 1.71E-15 |
| (4, 6) | 1.10E-07 | 1.50E-08 | (3, 4, 6) | 18 | 8.97E-14 | 1.71E-15 |
| (5, 6) | 4.82E-05 | 1.35E-05 | (4, 5, 6) | 24 | 8.97E-14 | 1.71E-15 |
| (6, 6) | 1.10E-02 | 5.89E-03 | (5, 6, 6) | 30 | 8.97E-14 | 1.71E-15 |
| (2, 7) | 5.09E-17 | 3.36E-18 | (1, 2, 7) | 7 | 5.09E-17 | 3.36E-18 |
| (3, 7) | 1.31E-13 | 1.68E-14 | (2, 3, 7) | 14 | 5.09E-17 | 3.36E-18 |
| (4, 7) | 1.70E-10 | 3.64E-11 | (3, 4, 7) | 21 | 5.09E-17 | 3.36E-18 |
| (5, 7) | 1.24E-07 | 4.06E-08 | (4, 5, 7) | 28 | 5.09E-17 | 3.36E-18 |
| (6, 7) | 5.15E-05 | 2.47E-05 | (5, 6, 7) | 35 | 5.09E-17 | 3.36E-18 |
| (7, 7) | 1.12E-02 | 7.78E-03 | (6, 7, 7) | 42 | 5.09E-17 | 3.36E-18 |
| (2, 8) | 1.23E-20 | 1.23E-20 | (1, 2, 8) | 8 | 1.23E-20 | 1.23E-20 |
| (3, 8) | 6.40E-17 | 6.40E-17 | (2, 3, 8) | 16 | 1.23E-20 | 1.23E-20 |
| (4, 8) | 1.48E-13 | 1.48E-13 | (3, 4, 8) | 24 | 1.23E-20 | 1.23E-20 |
| (5, 8) | 1.82E-10 | 1.82E-10 | (4, 5, 8) | 32 | 1.23E-20 | 1.23E-20 |
| (6, 8) | 1.29E-07 | 1.29E-07 | (5, 6, 8) | 40 | 1.23E-20 | 1.23E-20 |
| (7, 8) | 5.26E-05 | 5.26E-05 | (6, 7, 8) | 48 | 1.23E-20 | 1.23E-20 |
| (8, 8) | 1.13E-02 | 1.13E-02 | (7, 8, 8) | 56 | 1.23E-20 | 1.23E-20 |

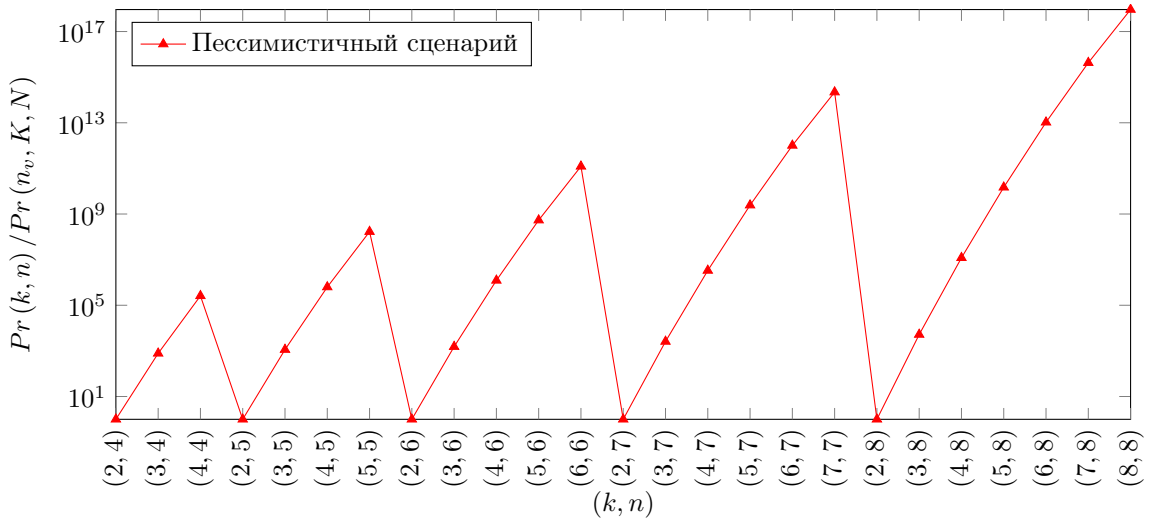


Рисунок 2.3 — Преимущество схемы WA-MRC-RRNS по сравнению с MRC-RRNS с точки зрения вероятности потери данных в пессимистичном сценарии

Далее представлен сравнительный анализ производительности схем AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS.

Для сравнения производительности вышеуказанных схем были разработаны соответствующие системы на языке программирования Java. Тестирование систем проводилось с использованием сервера Express x3650 M4 с двумя процессорами Xeon IvyBridge E5-2650v2 95 Вт, тактовой частотой по умолчанию 2,6 ГГц и симметричным подключением к Интернету со скоростью 300 Мбит/с. Каждый процессор имеет восемь ядер и по два потока на ядро (16 с гиперпоточностью), 32 КБ кэша уровня 1, 256 КБ уровня 2 и 20 МБ уровня 3. Так же использовались два домена NUMA по 32 ГБ каждый с общим объемом памяти 64 ГБ. Операционная система сервера – CentOS Linux версии 7.1.1503.

При моделировании системы использовались 16-битные модули, являющиеся простыми числами (табл. 8).

2.2.3 Сравнение производительности структур доступа

Если обратить внимание, например, на схемы, использующие все хранилища, т.е. при $n = 8$ для традиционных и $N = 8$ для взвешенных пороговых схем, то первые схемы (2, 8) для AR-RRNS и MRC-RRNS и (1, 2, 8) для WA-AR-

Таблица 8 — Простые 16-битные числа, используемые при моделировании структуры доступа на основе RRNS

| Модули RRNS | | | | | |
|------------------|------------------|------------------|------------------|------------------|------------------|
| $p_1 = 65521$ | $p_2 = 65519$ | $p_3 = 65497$ | $p_4 = 65479$ | $p_5 = 65449$ | $p_6 = 65447$ |
| $p_7 = 65437$ | $p_8 = 65423$ | $p_9 = 65419$ | $p_{10} = 65413$ | $p_{11} = 65407$ | $p_{12} = 65393$ |
| $p_{13} = 65381$ | $p_{14} = 65371$ | $p_{15} = 65357$ | $p_{16} = 65353$ | $p_{17} = 65327$ | $p_{18} = 65323$ |
| $p_{19} = 65309$ | $p_{20} = 65293$ | $p_{21} = 65287$ | $p_{22} = 65269$ | $p_{23} = 65267$ | $p_{24} = 65257$ |
| $p_{25} = 65239$ | $p_{26} = 65213$ | $p_{27} = 65203$ | $p_{28} = 65183$ | $p_{29} = 65179$ | $p_{30} = 65173$ |
| $p_{31} = 65171$ | $p_{32} = 65167$ | $p_{33} = 65147$ | $p_{34} = 65141$ | $p_{35} = 65129$ | $p_{36} = 65123$ |
| $p_{37} = 65119$ | $p_{38} = 65111$ | $p_{39} = 65101$ | $p_{40} = 65099$ | $p_{41} = 65089$ | $p_{42} = 65071$ |
| $p_{43} = 65063$ | $p_{44} = 65053$ | $p_{45} = 65033$ | $p_{46} = 65029$ | $p_{47} = 65027$ | $p_{48} = 65011$ |
| $p_{49} = 65003$ | $p_{50} = 64997$ | $p_{51} = 64969$ | $p_{52} = 64951$ | $p_{53} = 64937$ | $p_{54} = 64927$ |
| $p_{55} = 64921$ | | | $p_{56} = 64919$ | | |

RRNS и WA-MRC-RRNS кодируют по восемь долей одинакового размера. Следующая схема (3, 8) для AR-RRNS и MRC-RRNS кодирует все те же восемь долей, тогда как схема (2, 4, 8) для WA-AR-RRNS и WA-MRC-RRNS кодирует уже шестнадцать долей, поэтому размер долей для взвешенных схем (WA-AR-RRNS и WA-MRC-RRNS) будет уже в два раза меньше, чем для традиционных (AR-RRNS и MRC-RRNS) и т.д. Таким образом, с увеличением порогового значения размер коротких долей во взвешенных схемах уменьшается по сравнению с традиционными.

Рисунок 2.4 и таблица 9 показывают, что по мере увеличения N скорость кодирования для сравниваемых схем уменьшается, достигая минимального значения при $N = 8$. При этом скорость кодирования в WA-AR-RRNS в среднем приблизительно в три раза меньше, чем в AR-RRNS. Рисунок 2.5 и таблица 10 показывают, что скорость декодирования AR-RRNS превосходит WA-AR-RRNS в среднем приблизительно в четыре раза. При настройке (N, N) декодирование AR-RRNS достигает самых высоких скоростей, в то время как при той же настройке WA-AR-RRNS имеет самые низкие скорости декодирования. При использовании WA-AR-RRNS требуется больше коротких долей для восстановления данных, что приводит к снижению скоростей до диапазона 0.042-0.233 MB/s. Самое осязаемое различие между скоростями декодирования сравниваемых систем можно увидеть в последних строках таблицы 10. Для настройки (8, 8) скорость декодирования AR-RRNS равна 0.6219 MB/s, тогда как

Таблица 9 — Скорость кодирования (МБ/с) AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS

| (k, n) | Скорость кодирования (МБ/с) | | (n_v, K, N) | n | Скорость кодирования (МБ/с) | |
|----------|--------------------------------|----------|---------------|-----|--------------------------------|-------------|
| | AR-RRNS | MRC-RRNS | | | WA-AR-RRNS | WA-MRC-RRNS |
| (2, 4) | 2.7748 | 21.6309 | (1, 2, 4) | 4 | 2.8049 | 21.3903 |
| (3, 4) | 2.9335 | 32.8947 | (2, 3, 4) | 8 | 2.4514 | 21.7865 |
| (4, 4) | 3.4104 | 41.7014 | (3, 4, 4) | 12 | 2.1126 | 21.3265 |
| (2, 5) | 2.2739 | 18.1653 | (1, 2, 5) | 5 | 1.6032 | 17.8858 |
| (3, 5) | 3.2086 | 33.1235 | (2, 3, 5) | 10 | 0.8240 | 17.7336 |
| (4, 5) | 3.6810 | 36.5898 | (3, 4, 5) | 15 | 1.0581 | 16.7954 |
| (5, 5) | 3.9228 | 47.8011 | (4, 5, 5) | 20 | 1.1239 | 15.5836 |
| (2, 6) | 1.9525 | 14.1123 | (1, 2, 6) | 6 | 1.3505 | 13.8638 |
| (3, 6) | 2.7071 | 23.5849 | (2, 3, 6) | 12 | 0.6957 | 14.3472 |
| (4, 6) | 3.4888 | 29.0360 | (3, 4, 6) | 18 | 0.8776 | 13.7099 |
| (5, 6) | 3.8506 | 35.2485 | (4, 5, 6) | 24 | 0.9348 | 12.6214 |
| (6, 6) | 4.3254 | 40.4040 | (5, 6, 6) | 30 | 1.0578 | 12.2384 |
| (2, 7) | 1.6386 | 15.0127 | (1, 2, 7) | 7 | 1.1630 | 14.7929 |
| (3, 7) | 2.2921 | 20.9161 | (2, 3, 7) | 14 | 0.5993 | 12.7860 |
| (4, 7) | 3.0228 | 26.0960 | (3, 4, 7) | 21 | 0.7702 | 12.0091 |
| (5, 7) | 3.2084 | 34.5901 | (4, 5, 7) | 28 | 0.8208 | 10.4975 |
| (6, 7) | 3.4276 | 37.5798 | (5, 6, 7) | 35 | 0.8970 | 10.5630 |
| (7, 7) | 3.4981 | 43.0848 | (6, 7, 7) | 42 | 0.8066 | 10.4482 |
| (2, 8) | 1.4765 | 10.5152 | (1, 2, 8) | 8 | 1.0165 | 10.4231 |
| (3, 8) | 2.0894 | 17.7022 | (2, 3, 8) | 16 | 0.5271 | 11.1296 |
| (4, 8) | 2.7487 | 22.8623 | (3, 4, 8) | 24 | 0.6586 | 10.3659 |
| (5, 8) | 2.9209 | 28.9435 | (4, 5, 8) | 32 | 0.7178 | 9.4126 |
| (6, 8) | 3.0936 | 32.5203 | (5, 6, 8) | 40 | 0.7543 | 9.0358 |
| (7, 8) | 3.1647 | 37.2995 | (6, 7, 8) | 48 | 0.7091 | 8.6772 |
| (8, 8) | 3.5379 | 42.1230 | (7, 8, 8) | 56 | 0.7017 | 7.7724 |

Таблица 10 — Скорость декодирования (МБ/с) AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS

| (k, n) | Скорость декодирования (МБ/с) | | (n_v, K, N) | n | Скорость декодирования (МБ/с) | |
|----------|-------------------------------|----------|---------------|-----|-------------------------------|-------------|
| | AR-RRNS | MRC-RRNS | | | WA-AR-RRNS | WA-MRC-RRNS |
| (2, 4) | 0.2544 | 32.4254 | (1, 2, 4) | 4 | 0.2330 | 33.0033 |
| (3, 4) | 0.2834 | 33.8868 | (2, 3, 4) | 8 | 0.0903 | 32.9924 |
| (4, 4) | 0.3514 | 33.7837 | (3, 4, 4) | 12 | 0.1242 | 28.6861 |
| (2, 5) | 0.2077 | 33.7040 | (1, 2, 5) | 5 | 0.1712 | 33.8983 |
| (3, 5) | 0.2769 | 34.1064 | (2, 3, 5) | 10 | 0.0745 | 35.0140 |
| (4, 5) | 0.3266 | 34.1413 | (3, 4, 5) | 15 | 0.0942 | 30.5810 |
| (5, 5) | 0.4204 | 34.2818 | (4, 5, 5) | 20 | 0.1007 | 26.5041 |
| (2, 6) | 0.1703 | 33.8409 | (1, 2, 6) | 6 | 0.1590 | 33.5570 |
| (3, 6) | 0.2251 | 30.8928 | (2, 3, 6) | 12 | 0.0690 | 30.8737 |
| (4, 6) | 0.2791 | 34.1297 | (3, 4, 6) | 18 | 0.0864 | 29.5421 |
| (5, 6) | 0.3240 | 31.6455 | (4, 5, 6) | 24 | 0.0810 | 26.0348 |
| (6, 6) | 0.4467 | 30.6560 | (5, 6, 6) | 30 | 0.0798 | 23.9234 |
| (2, 7) | 0.1497 | 36.1271 | (1, 2, 7) | 7 | 0.1207 | 35.8937 |
| (3, 7) | 0.1917 | 35.2112 | (2, 3, 7) | 14 | 0.0547 | 34.5423 |
| (4, 7) | 0.2325 | 35.1617 | (3, 4, 7) | 21 | 0.0650 | 30.5530 |
| (5, 7) | 0.2859 | 33.8524 | (4, 5, 7) | 28 | 0.0643 | 26.9179 |
| (6, 7) | 0.3612 | 31.2207 | (5, 6, 7) | 35 | 0.0599 | 23.2991 |
| (7, 7) | 0.5035 | 29.4724 | (6, 7, 7) | 42 | 0.0697 | 20.9995 |
| (2, 8) | 0.1306 | 36.1010 | (1, 2, 8) | 8 | 0.0962 | 34.1180 |
| (3, 8) | 0.1637 | 33.4896 | (2, 3, 8) | 16 | 0.0440 | 34.9162 |
| (4, 8) | 0.1988 | 34.5065 | (3, 4, 8) | 24 | 0.0527 | 31.2207 |
| (5, 8) | 0.2471 | 32.8947 | (4, 5, 8) | 32 | 0.0543 | 27.2702 |
| (6, 8) | 0.3086 | 31.5159 | (5, 6, 8) | 40 | 0.0532 | 23.4137 |
| (7, 8) | 0.3976 | 29.3944 | (6, 7, 8) | 48 | 0.0572 | 21.0482 |
| (8, 8) | 0.6219 | 26.9251 | (7, 8, 8) | 56 | 0.0420 | 19.0295 |

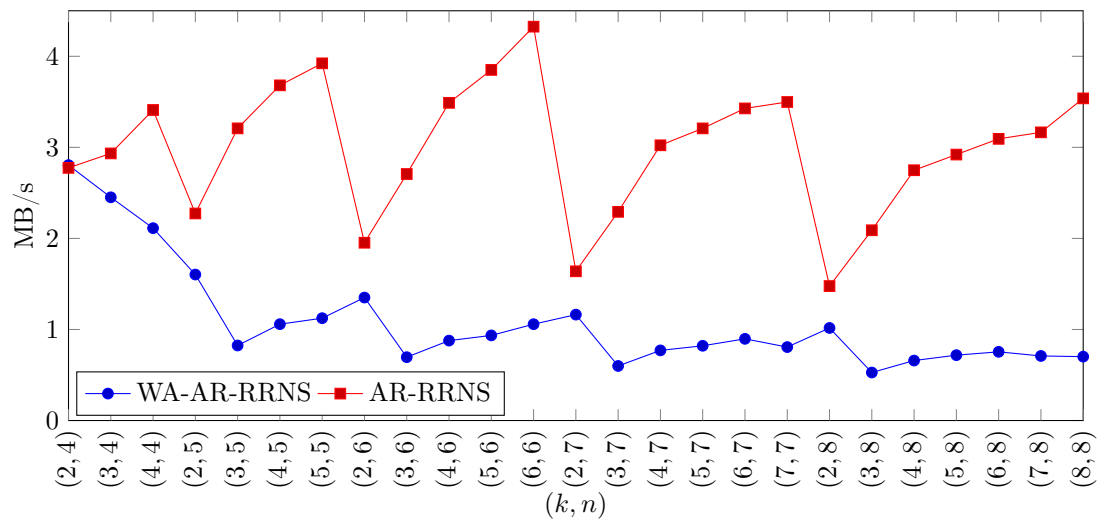


Рисунок 2.4 — Скорость кодирования WA-AR-RRNS и AR-RRNS

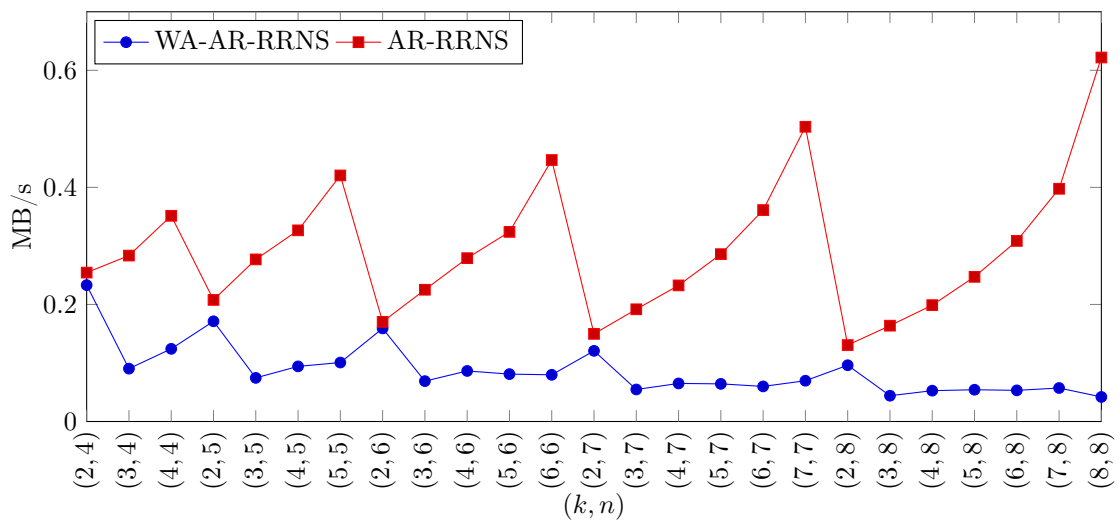


Рисунок 2.5 — Скорость декодирования WA-AR-RRNS и AR-RRNS

для соответствующей настройки $(7, 14, 8)$ с $n = 8 \cdot 7 = 56$ короткими долями, скорость декодирования WA-AR-RRNS составляет 0.042 MB/s, что почти в 15 раз медленнее по сравнению с AR-RRNS.

Схемы MRC-RRNS и WA-MRC-RRNS используют алгоритм кодирования-декодирования, основанный на переходе к смешанной позиционной системе счисления [243] и нейронной сети конечного кольца (Finite Ring Neural Network – FRNN) [373]. FRNN снижает вычислительную сложность алгоритмов кодирования и декодирования, поскольку промежуточные повторяющиеся результаты рассчитываются один раз для каждого набора модулей RRNS и сохраняются в нейронной сети как синаптические веса. Такой подход оптимизирует алгоритм MRC и увеличивает скорость кодирования/декодирования.

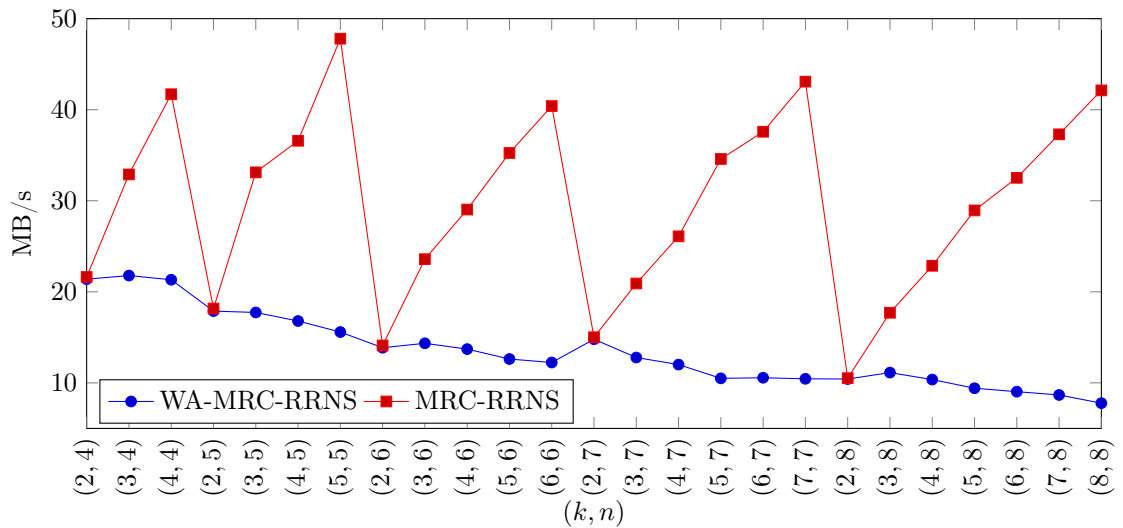


Рисунок 2.6 — Скорость кодирования WA-MRC-RRNS и MRC-RRNS

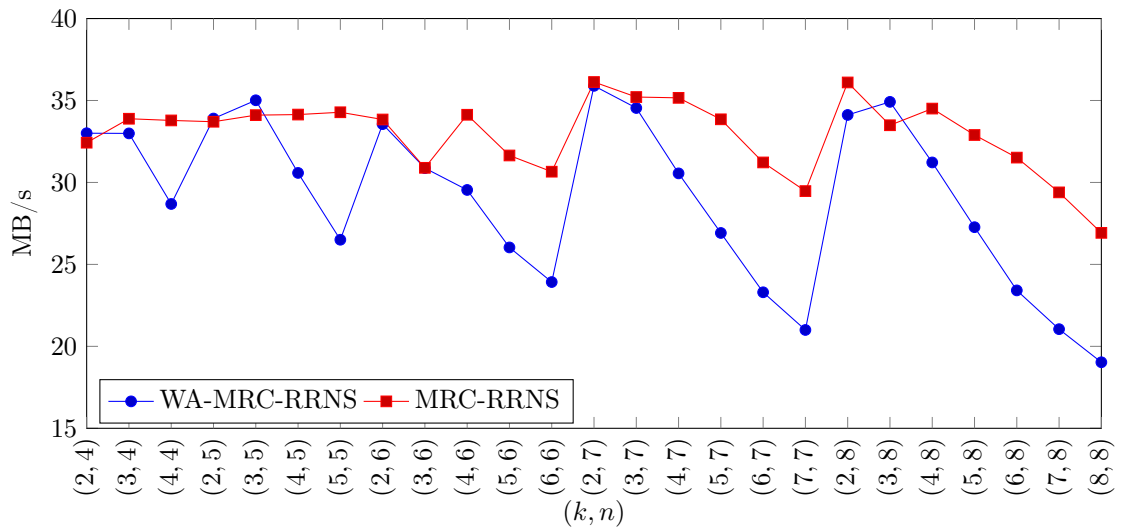


Рисунок 2.7 — Скорость декодирования WA-MRC-RRNS и MRC-RRNS

На рисунках 2.6 и 2.7 представлены скорость кодирования и декодирования сравниваемых схем соответственно. MRC-RRNS превосходит WA-MRC-RRNS в обоих случаях. Так же отметим, что скорость кодирования сравниваемых схем уменьшается с увеличением N . Рисунки 2.8 и 2.9 демонстрируют отношение скорости кодирования и декодирования WA-MRC-RRNS к соответствующим скоростям WA-AR-RRNS, AR-RRNS и MRC-RRNS для разных настроек (k, n) . На этапе кодирования для настройки $(3, 5)$ схема WA-MRC-RRNS в 21.52 раза быстрее, чем WA-AR-RRNS, в 5.53 раза быстрее, чем AR-RRNS, и в 1.87 раза медленнее, чем MRC-RRNS.

Наиболее ощутимое преимущество схемы WA-MRC-RRNS по сравнению со схемой WA-AR-RRNS на этапе декодирования достигает значения 792.77 для настройки $(3, 8)$. Если же сравнивать схемы WA-MRC-RRNS и AR-RRNS наи-

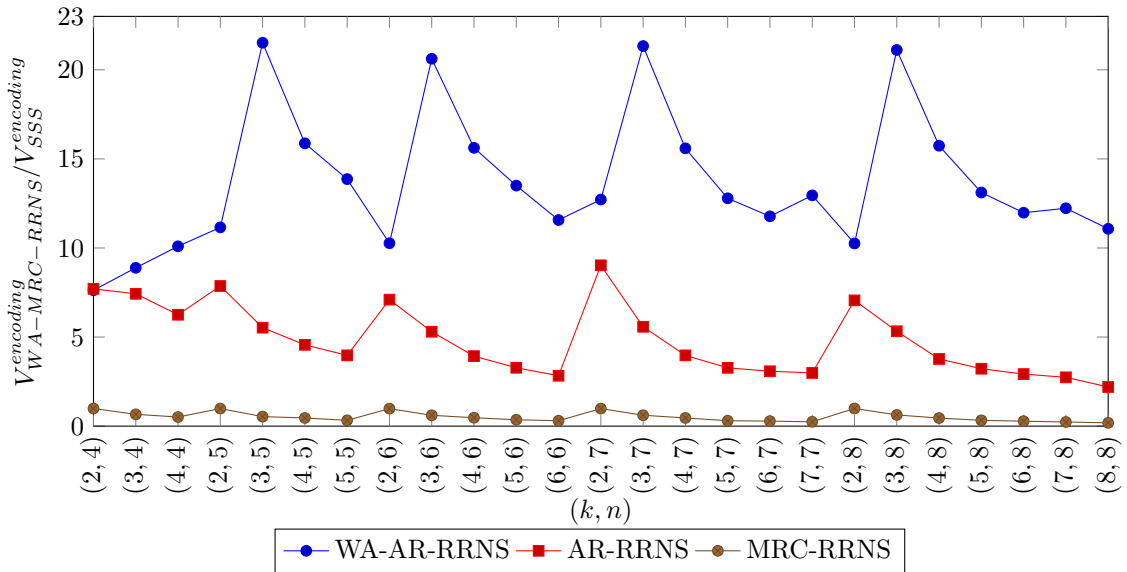


Рисунок 2.8 — Отношение скорости кодирования WA-MRC-RRNS к скорости кодирования WA-AR-RRNS, AR-RRNS и MRC-RRNS

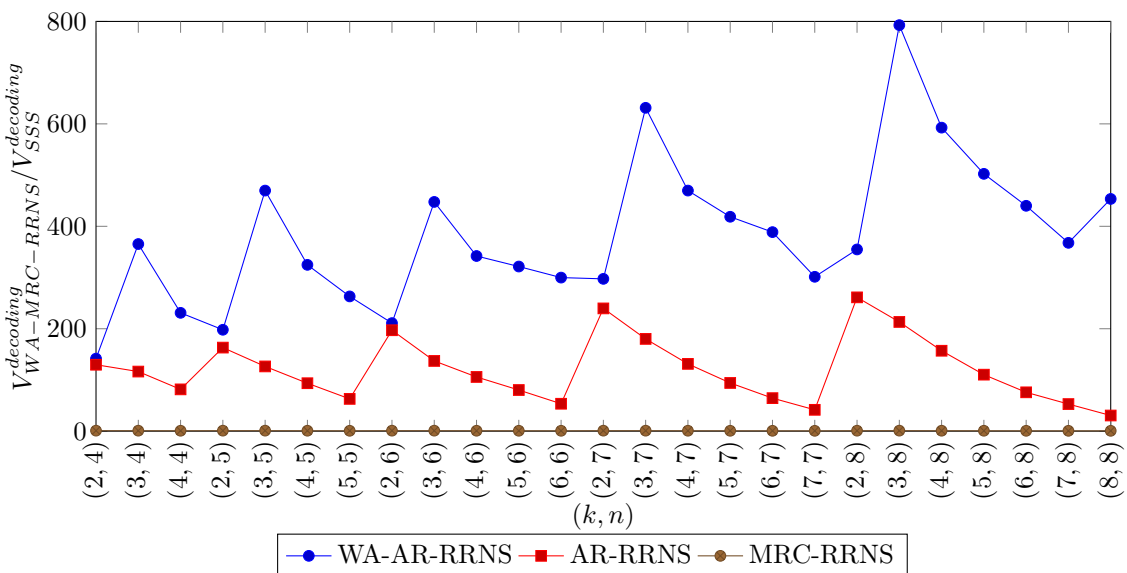


Рисунок 2.9 — Отношение скорости декодирования WA-MRC-RRNS к скорости декодирования WA-AR-RRNS, AR-RRNS и MRC-RRNS

Таблица 11 — Размеры долей AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS (в байтах)

| (k, n) | Размер долей | | (n_v, K, N) | n | Размер долей | |
|----------|--------------|----------|---------------|-----|--------------|-------------|
| | AR-RRNS | MRC-RRNS | | | WA-AR-RRNS | WA-MRC-RRNS |
| (2, 8) | 104857606 | 69905070 | (1, 2, 8) | 8 | 104857606 | 69905070 |
| (3, 8) | 69905072 | 41943042 | (2, 3, 8) | 16 | 52428806 | 29959318 |
| (4, 8) | 52428806 | 29959318 | (3, 4, 8) | 24 | 34952538 | 19065022 |
| (5, 8) | 41943046 | 23301692 | (4, 5, 8) | 32 | 26214406 | 13981016 |
| (6, 8) | 34952538 | 19065022 | (5, 6, 8) | 40 | 20971526 | 11037646 |
| (7, 8) | 29959320 | 16131942 | (6, 7, 8) | 48 | 17476272 | 9118056 |
| (8, 8) | 26214406 | 13981016 | (7, 8, 8) | 56 | 14979662 | 7767232 |

большее преимущество первой схемы наблюдается для настройки (2, 8) и составляет 261.28.

На этапе кодирования WA-MRC-RRNS превосходит по скорости схему WA-AR-RRNS в среднем в 13.73 раза, а схему AR-RRNS – в среднем в 4.83 раза. Если рассматривать этап декодирования, WA-MRC-RRNS превосходит по скорости схему WA-AR-RRNS в среднем в 385.06 раза, а схему AR-RRNS – в среднем в 120.04 раза.

В таблице 11 представлены размеры долей в байтах для AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS. Отметим, что размеры уменьшаются при возрастании порогового значения.

2.3 Проблема сговора в облачных сервисах при распределенном хранении и обработке данных

Данный раздел посвящен проблеме сговора CSP, в результате которого злоумышленники могут получить доступ к конфиденциальным данным в одном или нескольких хранилищах.

Для решения проблемы сговора используются три основные группы методов.

1. *Обнаружение сговора.* Для обнаружения сговора применяются механизмы, использующие публичные данные [163,200,335]. Публичные данные обрабатываются алгоритмами, относящимися к классу методов, предназначенных для их маркировки с целью проверки пользователей, которым эти данные были предоставлены [268]. Задача – обнаружить злоумышленника, участвующего в сговоре.
2. *Предотвращение сговора.* Основная идея методов предотвращения сговора заключается в блокировании доступа к данным для злоумышленников за счет распределения ключей [245], руководителей групп [375], структур доступа [375] и др. Слабым местом этих методов являются центры распределения ключей и руководители групп. В случае сговора с одним из них злоумышленник получает полный доступ к данным.
3. *Безопасность данных в случае сговора.* Для противодействия злоумышленникам, участвующим в сговоре, можно использовать добавление шума к конфиденциальным данным. Для добавления шума используются два основных метода: на основе секретного ключа и скрытых модулей RRNS. В работах [223,341] предложили модификацию структур доступа Asmuth-Bloom и Mignotte. Однако, как показано в работе Krawczyk [263], схема Asmuth-Bloom не применима на практике из-за большой избыточности данных.

Рассмотрим методы второй и третьей групп.

Для предотвращения облачного сговора, Nur в 2013 году [245] предложил схему, основанную на атрибутах секретных ключей, которые позволяют уменьшить количество открытых ключей. Автор использует распределенное депонирование ключей, состоящее из двух частей: центра генерации ключей и центра хранения.

Чтобы решить проблему сговора, Zhu и Jiang в 2016 году [375] предложили схему, использующую структуры доступа и протокол распределения ключей. В ее основе лежит билинейное спаривание в точках эллиптических кривых и дискретный логарифм Diffie-Hellman. Руководитель группы использует главный секретный ключ. В случае технических сбоев, приводящих к потере или искажению секретного мастер-ключа, работа всей системы нарушается. В случае кражи мастер-ключа злоумышленник может получить полный контроль над распределенной системой хранения данных. Кроме того, безопасность главного ключа напрямую зависит от вычислительной сложности решения задачи дис-

кретного логарифмирования Diffie-Hellman, поэтому предложенный алгоритм не является безопасным в вычислительном отношении.

Альтернативным решением проблемы сговора является использование структуры доступа на основе RRNS, обеспечивающих безопасность и надежность хранения и обработки данных в течение длительного промежутка времени.

Схемы RRNS Asmuth-Bloom и Mignotte обеспечивают надежное хранение данных. Главный недостаток схемы Asmuth-Bloom — большая избыточность данных, а схема Mignotte имеет низкий уровень безопасности данных. Для повышения уровня безопасности схемы Mignotte, Gomathisankaran и др. [223] предложили использовать модули RRNS в качестве секретного ключа. Однако, это приводит к увеличению избыточности.

Вычислительно безопасные схемы позволяют обеспечить безопасность хранимых данных, а так же уменьшить избыточность данных и нагрузку на сеть в k раз по сравнению со схемами Asmuth-Bloom, Shamir и т.д. [263].

В Разделе 2.5 предложена вычислительно безопасная схема, основанная на RRNS и структуре доступа. Она не позволяет злоумышленнику сопоставлять исходные данные с соответствующими долями, а также предотвращает атаки с использованием известного открытого текста, когда злоумышленник знает все данные, кроме секретного ключа. Кроме того, предложенная схема обладает свойством гомоморфизма и является одновременно инструментом обеспечения безопасности, надежности, конфиденциальности и обработки закодированных данных. Данные разбиваются на несколько более мелких закодированных частей, которые сохраняются на ресурсах разных провайдеров.

Выделяют три основных сценария сговора:

1. Злоумышленники знают требуемое количество долей исходных данных, но не знают секретного ключа.
2. Злоумышленники не знают ни требуемого количества долей исходных данных ни секретного ключа.
3. Злоумышленники не знают требуемого количества долей исходных данных, но знают секретный ключ.

Сравним схемы WA-MRC-RRNS и AR-RRNS с точки зрения устойчивости к облачному сговору.

Докажем следующие утверждения.

Утверждение 2.3.1. В предлагаемой схеме WA-MRC-RRNS вероятность получения данных на основе долей из t различных хранилищ $\bar{I} = \{i_1, i_2, \dots, i_t\}$ меньше или равна

$$Pr_C(L) \leq 2^{-(1-\frac{1}{W} \sum_{i \in \bar{I}} w_i) \cdot L} \cdot \prod_{i \in \bar{I}} Pr_{C_i} \cdot \prod_{i \notin \bar{I}} (1 - Pr_{C_i}), \quad (2.22)$$

где \bar{I} – множество злоумышленников (множество CSP, вступивших в сговор), $\sum_{(i \in \bar{I})} w_i < W$, L – размер исходных данных (в битах), а Pr_{C_i} – вероятность вступления в сговор i -го облака.

Доказательство. Рассмотрим случай, когда злоумышленники получают доступ ко всем долям, хранящимся в t CSP, но их общий вес меньше порога $\sum_{i \in \bar{I}} w_i < W$. Количество возможных вариантов, которые можно закодировать L битами равно 2^L .

Если i -ый злоумышленник имеет исходные данные в объеме $L \cdot \frac{w_i}{W}$, то все злоумышленники \bar{I} имеют объем данных $L \cdot \sum_{i \in \bar{I}} \frac{w_i}{W}$. Следовательно, количество неизвестных данных равно $L - L \cdot \sum_{i \in \bar{I}} \frac{w_i}{W} = (1 - \frac{1}{W} \sum_{i \in \bar{I}} w_i) \cdot L$.

Вероятность сговора t CSP равна

$$\prod_{i \in \bar{I}} Pr_{C_i} \cdot \prod_{i \notin \bar{I}} (1 - Pr_{C_i}). \quad (2.23)$$

Учитывая, что оставшийся неизвестный злоумышленникам текст представляет собой равновероятные неизвестные данные, вероятность получения данных \bar{I} злоумышленниками равна

$$Pr_C(L) = 2^{-(1-\frac{1}{W} \sum_{i \in \bar{I}} w_i) \cdot L} \cdot \prod_{i \in \bar{I}} Pr_{C_i} \cdot \prod_{i \notin \bar{I}} (1 - Pr_{C_i}). \quad (2.24)$$

Утверждение доказано. □

Для схемы WA-MRC-RRNS избыточность можно определить как

$$R = \frac{1}{W} \sum_{i=1}^N w_i, \quad (2.25)$$

тогда, подставляя (2.25) в (2.24), получим

$$Pr_C(L) = 2^{-\left(1 - \frac{R}{W} \cdot \frac{\sum_{i \in \bar{I}} e^{-\lambda_i}}{\sum_{i=1}^N e^{-\lambda_i}}\right) \cdot L} \cdot \prod_{i \in \bar{I}} Pr_{C_i} \cdot \prod_{i \notin \bar{I}} (1 - Pr_{C_i}). \quad (2.26)$$

Таблица 12 — Параметры MRC-RRNS и WA-MRC-RRNS

| R | (k, n) | $\{w_1, w_2, \dots, w_N\}$ | $\bar{I} = \{i_1, i_2, \dots, i_\xi\}$ |
|-----|----------|------------------------------------|--|
| 1 | (6, 6) | 0.18, 0.18, 0.17, 0.17, 0.16, 0.15 | 1, 2, 3, 4, 5 |
| 1.5 | (4, 6) | 0.27, 0.27, 0.25, 0.25, 0.24, 0.22 | 1, 4, 5, 6 |
| 2 | (3, 6) | 0.36, 0.36, 0.34, 0.33, 0.32, 0.29 | 4, 5, 6 |
| 2.5 | (2, 5) | 0.53, 0.52, 0.49, 0.49, 0.47 | 3, 4 |
| 3 | (2, 6) | 0.54, 0.53, 0.50, 0.50, 0.48, 0.44 | 3, 5 |

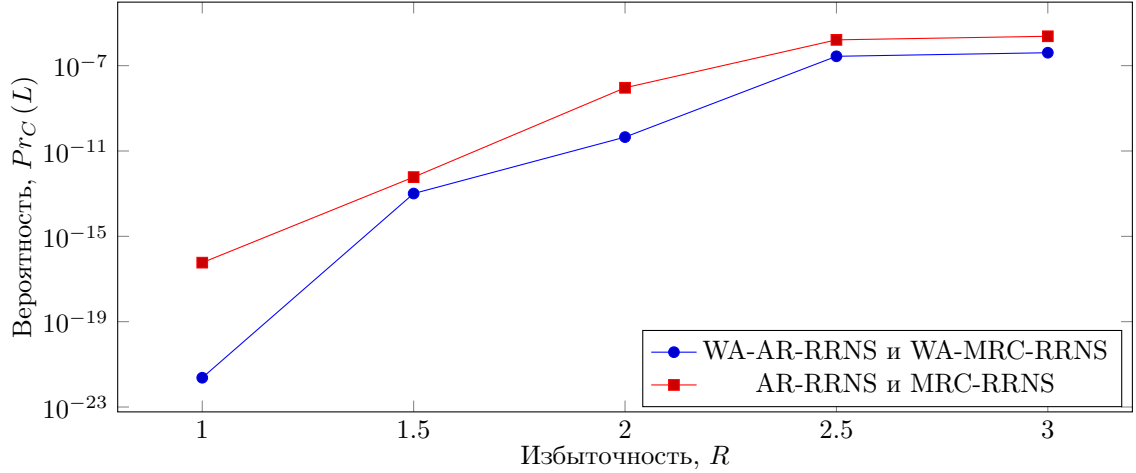


Рисунок 2.10 — Зависимость вероятности потери данных/получения несанкционированного доступа от избыточности данных, при $W = 1$ и $L = 128$ бит

Чтобы интерпретировать этот результат, рассмотрим пример с порогом $W = 1$ и длиной секретного ключа $L = 128$ бит. В таблице 12 показаны веса и индексы используемых хранилищ, соответствующие схемам с разной избыточностью. Избыточность данных в свою очередь вычисляется как $R \approx \frac{N}{K}$ (или $R \approx \frac{n}{k}$). Учитывая, что вероятность сговора на i -ом облаке равна вероятности утечки данных (табл. 3), $Pr_{C_1} = 0.000065$, $Pr_{C_2} = 0.000285$, $Pr_{C_3} = 0.001235$, $Pr_{C_4} = 0.001320$, $Pr_{C_5} = 0.001941$, $Pr_{C_6} = 0.003594$ и $\lambda = (0.0039, 0.0171, 0.0741, 0.0792, 0.1164, 0.2156)$. На рисунке 2.10 отражена зависимость вероятности получения злоумышленниками несанкционированного доступа к данным от их избыточности. Исходя из полученных графиков, можно сделать вывод, что предлагаемые схемы WA-AR-RRNS и WA-MRC-RRNS характеризуются более высоким уровнем защиты хранимых данных от облачных сговоров по сравнению с AR-RRNS и MRC-RRNS. Также отметим, что

увеличение избыточности приводит к снижению безопасности и, как было установлено ранее, повышению надежности (табл. 5).

В 2011 году Gomathisankaran и др. [223] предложили модифицированный с учетом проблемы сговора подход к построению схемы Mignotte под названием HORNS (A homomorphic encryption scheme for Cloud Computing using Residue Number System), основанный на использовании скрытых модулей RRNS. В Разделе 2.4.1 показано, что HORNS менее безопасна, чем схема Asmuth-Bloom. Так же в Разделе 2.4.2 доказываем вычислительная безопасность предложенной схемы AC-RRNS.

2.4 Атака открытым текстом на гомоморфные коды

2.4.1 Атака открытым текстом на HORNS

В HORNS в качестве секретного ключа используется набор модулей RRNS. Поскольку ключ неизвестен, результаты арифметических операций, выполненных с хранимыми долями, не преобразуются в остатки. Это увеличивает размер хранимых данных, но является необходимостью, т.к. если после атаки остатки становятся известны, злоумышленник может идентифицировать модули RRNS путем соответствующего перебора возможных модулей.

Основная идея метода вычисления скрытых модулей RRNS (атаки на схему HORNS) заключается в получении данных о секретном ключе на основе результатов деления нескольких наборов исходных данных S определенного специального содержания. Для произвольного целого t возможны три случая.

Случай 1: если $2^t < p_i$, то $|2^t|_{p_i} = 2^t$.

Случай 2: если $2^{t-1} < p_i \leq 2^t$, то $2^t = p_i + |2^t|_{p_i}$.

Случай 3: если $2^t \geq p_i$, то $|2^t|_{p_i} \neq 2^t$.

Нижняя граница t больше или равна 1. Чтобы найти верхнюю границу t , мы используем случай 3, где $2^t \geq p_i$.

Возьмем t из геометрического ряда с знаменателем 2: 1, 2, 4, 8, 16, 32, ... Теперь оценим более точно нижнюю и верхнюю границы: $t \in (2^{a-1}, 2^a]$, где $a = \lceil \log_2 \log_2 p_i \rceil$.

Таблица 13 — Вычисление полуинтервала для секретного ключа HORNS

| a | $S = 2^{2^a}$ | $ S _{p_1}$ | $ S _{p_2}$ | $ S _{p_3}$ | $ S _{p_4}$ |
|-----|------------------------|-------------|-------------|-------------|-------------|
| 0 | $2^{2^0} = 2$ | 2 | 2 | 2 | 2 |
| 1 | $2^{2^1} = 4$ | 4 | 4 | 4 | 4 |
| 2 | $2^{2^2} = 16$ | 16 | 16 | 16 | 16 |
| 3 | $2^{2^3} = 256$ | 256 | 256 | 256 | 256 |
| 4 | $2^{2^4} = 65536$ | 65536 | 65536 | 65536 | 65536 |
| 5 | $2^{2^5} = 4294967296$ | 1073741823 | 1073741817 | 1073741763 | 1073741747 |

Таблица 14 — Уточнение секретного ключа HORNS

| a | b | t | $S = 2^t$ | $ S _{p_1}$ | $ S _{p_2}$ | $ S _{p_3}$ | $ S _{p_4}$ |
|-----|-----|-----|------------|-------------|-------------|-------------|-------------|
| 16 | 32 | 24 | 16777216 | 16777216 | 16777216 | 16777216 | 16777216 |
| 24 | 32 | 28 | 268435456 | 268435456 | 268435456 | 268435456 | 268435456 |
| 28 | 32 | 30 | 1073741824 | 1073741824 | 1073741824 | 1073741824 | 1073741824 |
| 30 | 32 | 31 | 2147483648 | 2147483648 | 2147483648 | 2147483648 | 2147483648 |
| 31 | 32 | 32 | 4294967296 | 1073741823 | 1073741817 | 1073741763 | 1073741747 |

Уточнить t можно, применив случай 2 и бинарный поиск. Проиллюстрируем описанный выше метод на следующем примере.

Пример 2.4.1. Пусть структура доступа HORNS характеризуется параметрами $k = 2$, $n = 4$, а RRNS задается модулями: $p_1 = 3221225473$, $p_2 = 3221225479$, $p_3 = 3221225533$, $p_4 = 3221225549$, и $\{p_1, p_2, p_3, p_4\}$ – секретный ключ. В таблице 13 представлен результат работы схемы для различных исходных данных S , соответствующих различным значениям параметра a . Уточним, что модули RRNS неизвестны, известен лишь результат работы схемы, соответствующий разделяемому данным S . Поскольку для $a = 0, 1, 2, 3, 4$ $S = |S|_{p_i}$ (случай 1), а для $a = 5$ выполняется условие $S \neq |S|_{p_i}$ (случай 3), то $t \in (2^4, 2^5]$. Начальные значения a и b соответственно равны 2^4 и 2^5 .

Теперь мы можем оценить t более точно, используя бинарный поиск, т.е. формулу $t = \lceil \frac{a+b}{2} \rceil$ (табл. 14). Уточнение производится аналогично тому, как это было сделано выше. Согласно вычислениям, представленным в таблице 14, $t = 32$.

Используя случай 2, вычислим секретный ключ (модули RRNS):

$$\begin{aligned} p_1 &= 2^{32} - |S|_{p_1} = 2^{32} - 1073741823 = 3221225473, \\ p_2 &= 2^{32} - |S|_{p_2} = 2^{32} - 1073741817 = 3221225479, \\ p_3 &= 2^{32} - |S|_{p_3} = 2^{32} - 1073741763 = 3221225533, \\ p_4 &= 2^{32} - |S|_{p_4} = 2^{32} - 1073741747 = 3221225549. \end{aligned}$$

Пример 2.4.1 показывает, что подход, предложенный в работе [223] не является безопасным с вычислительной точки зрения, и секретный ключ может быть вычислен в результате атаки известным открытым текстом. Секретный ключ позволяет получить конфиденциальные данные пользователя. Таким образом, схема HORNS небезопасна с вычислительной точки зрения.

2.4.2 Атака открытым текстом на схему Asmuth-Bloom

В данном разделе анализируется атака с использованием известного открытого текста на схему Asmuth-Bloom. Подобно случаю атаки известным открытым текстом на схему HORNS, предполагается, что злоумышленник знает несколько наборов данных. Каждый набор с номером j содержит исходные данные $S^{(j)}$, набор модулей RRNS, и разделенные согласно схеме данные $C^{(j)} \xrightarrow{RRNS} (c_1^{(j)}, c_2^{(j)}, \dots, c_n^{(j)})$. Доли генерируются по следующей формуле $\forall i = \overline{1, n}$:

$$c_i^{(j)} = \left| S^{(j)} + p_0 \cdot rand_j \right|_{p_i}, \quad (2.27)$$

где $rand_j$ – случайное число, p_0 – секретный ключ, и $|x|_p$ – операция вычисления наименьшего неотрицательного остатка от деления x на модуль p . Чтобы оценить количество пар $(S^{(j)}, C^{(j)})$, необходимых для получения несанкционированного доступа к данным, докажем следующую теорему.

Теорема 2.4.1. *Зная $[k + k \cdot \log_2 p_0]$ пар $(S^{(j)}, C^{(j)})$, всегда можно узнать ключ p_0*

Доказательство. Из соотношения (2.27) мы получаем $c_i^{(j)} = S^{(j)} + p_0 \cdot rand_j - \alpha \cdot p_i$, где $\alpha \in \mathbb{Z}$. Зная пары $S^{(j)}, C^{(j)}$, получаем соответствующие значения

$A_j = p_0 \text{rand}_j$. Таким образом,

$$p_0 \cdot \text{rand}_j = c_i^{(j)} - S^{(j)} + \alpha \cdot p_i, \quad (2.28)$$

$$|p_0 \cdot \text{rand}_j|_{p_i} \equiv \left| c_i^{(j)} - S^{(j)} + \alpha \cdot p_i \right|_{p_i} \equiv \left| c_i^{(j)} - S^{(j)} \right|_{p_i}. \quad (2.29)$$

Пусть $\left| c_i^{(j)} - S^{(j)} \right|_{p_i} = a_i^{(j)}$, тогда

$$|p_0 \cdot \text{rand}_j|_{p_i} = a_i^{(j)}, \quad (2.30)$$

для всех $i \in [1, \dots, n]$. Зная остатки и модули, можно вычислить

$$A^{(j)} = p_0 \cdot \text{rand}_j \xrightarrow{RRNS} \left(a_1^{(j)}, a_2^{(j)}, \dots, a_n^{(j)} \right) \quad (2.31)$$

с помощью Китайской теоремы об остатках.

Из основной теоремы арифметики следует, что каждое целое число большее 1 либо само простое, либо является произведением простых чисел, и это представление единственно, если не учитывать порядок следования множителей. Следовательно, $A^{(j)} = m_1^{l_1} \cdot m_2^{l_2} \cdot \dots \cdot m_t^{l_t} \leq 2^k \cdot p_0^k$, где m_1, m_2, \dots, m_t — простые числа, а l_1, l_2, \dots, l_t — степени простых чисел соответственно.

Пусть $p_0^{(j)}$ — аппроксимация p_0 , полученная с j -ым набором данных. Она вычисляется как $p_0^{(j)} = \text{gcd} \left(p_0^{(j-1)}, A^{(j)} \right)$ для любого $j \in [2, \dots, N]$, $p_0^{(1)} = A^{(1)}$, а N — количество пар $(S^{(j)}, C^{(j)})$, используемых для аппроксимации p_0 . Из свойства функции $\text{gcd} \left(p_0^{(j-1)}, A^{(j)} \right)$ следует, что значения $p_0^{(j)}$ удовлетворяют следующему условию $p_0^{(1)} \geq p_0^{(2)} \geq \dots \geq p_0^{(N)} = p_0$.

В худшем случае, $p_0^{(0)} = m_1^{l_1} \cdot m_2^{l_2} \cdot \dots \cdot m_t^{l_t}$, $p_0^{(2)} = m_1^{l_1-1} \cdot m_2^{l_2} \cdot \dots \cdot m_t^{l_t}$, $p_0^{(l_1)} = m_1^{l_1-l_1} \cdot m_2^{l_2} \cdot \dots \cdot m_t^{l_t}$, $p_0^{(l_1+1)} = m_2^{l_2-1} \cdot \dots \cdot m_t^{l_t}$ и $p_0^{(l_1+l_2)} = m_2^{l_2-l_2} \cdot \dots \cdot m_t^{l_t}$, $p_0^{(l_1+l_2+\dots+l_t-1)} = m_t^{l_t}$. Поскольку $l_1 + l_2 + \dots + l_t - 1 \leq \lceil \log_2 (2^k \cdot p_0^k) \rceil = \lceil k + k \cdot \log_2 p_0 \rceil$, $N \leq \lceil k + k \cdot \log_2 p_0 \rceil$.

Теорема доказана. \square

2.5 Схема AC-RRNS и ее свойства

В предыдущих разделах было установлено, что подход со скрытыми модулями не следует использовать в качестве решения проблемы сговора облаков.

Это приводит к большой избыточности данных и не обеспечивает их безопасность. Для предотвращения сговора в облаке и обеспечения безопасности данных предлагается схема AC-RRNS. AC-RRNS представляет собой модификацию схемы Asmuth-Bloom [151], является вычислительно безопасной и позволяет снизить избыточность данных в k раз, где k – параметр схемы.

При использовании схемы AC-RRNS одна общая папка RRNS хранится в одном облачном провайдере. Для предотвращения сговора в облаке, используется секретный ключ p_0 , а для уменьшения избыточности данных предлагается ослабить условие Asmuth-Bloom $p_0 < p_1$, заменив его на $p_0 < P$.

Известно, что модули, равные степени двойки, небезопасны. Они позволяют злоумышленнику узнать, что часть конфиденциальных данных соответствует его проекции данных. Поэтому предлагается использовать модули разрядностью, равной разрядности машинного слова.

В RRNS вычислительная безопасность системы зависит от параметров k и n . За счет их выбора обеспечивается необходимый уровень вычислительной безопасности. Кроме того, корректирующие свойства RRNS позволяют обнаруживать и исправлять ошибки, возникающие из-за технических сбоев при передаче и хранении данных или из-за преднамеренной подделки данных при сговоре.

Червяков и др. [11] показали, что надежность системы зависит от r , где $r = n - k$. Чем больше значение r , тем надежнее система, однако, с ростом r растет избыточность. Учитывая современные объемы хранимых и обрабатываемых цифровых данных, избыточность становится ключевым фактором. После анализа избыточности таких систем, как RACS [141], DepSky [192] и др., становится ясно, что наиболее оптимальным условием для выбора r является $r < k$.

При использовании схемы AC-RRNS каждый облачный провайдер получает блок данных (долю), который состоит из идентификатора доли, свойств доли, проекции исходных данных, упрощенной цифровой подписи и модулей RRNS. Для генерации уникального секретного ключа используется хеш-функция на основе алгоритма SHA-3 [197].

В предлагаемой пороговой структуре используется концепция асимптотически совершенной схемы Asmuth-Bloom с нулевым разглашением [318].

Следуя [223], пусть параметр p_0 будет секретным ключом. Динамический диапазон системы, основанной на схеме Asmuth-Bloom, составляет $[0, p_0)$, что не подходит для построения защиты данных, участвующих в облачных вычисле-

ниях. Кроме того, размер каждой доли больше, чем размер исходных данных, что приводит к увеличению объема более чем в n раз.

Чтобы обезопасить систему от сговора в облаке, предлагается объединить подходы двух схем: Asmuth-Bloom [151] и Mignotte [280]. Для формализации предложенной схемы используем следующие обозначения: S – исходные данные, p_1, p_2, \dots, p_n – попарно взаимно простые числа (набор модулей RRNS), p_0 – целое число (адаптивный параметр, секретный ключ), взаимно простое с каждым из p_1, p_2, \dots, p_n .

Эти параметры удовлетворяют следующим трем условиям.

Условие 1. $p_0 > S$.

Условие 2. $\beta = \prod_{i=1}^k p_i > p_0 > \prod_{i=0}^{k-2} p_{n-i} = \alpha$.

Условие 3. $2^{l-1} < p_1 < p_2 < \dots < p_n < 2^l - 1$, где l – длина каждого модуля в битах.

Доли исходных данных c_i генерируются по следующей формуле

$$\forall i \in [1, \dots, n] : c_i = |S + p_0 \cdot rand|_{p_i}. \quad (2.32)$$

Условие 2 является модификацией условия Asmuth-Bloom $p_0 < p_1$. Динамический диапазон системы увеличивается в $2^{\lceil \log_2 \prod_{i=0}^{k-3} p_{n-i} \rceil}$ раз, где $2^{\lceil \log_2 \prod_{i=0}^{k-3} p_{n-i} \rceil} < \prod_{i=0}^{k-3} p_{n-i} \leq \frac{\prod_{i=0}^{k-2} p_{n-i}}{p_1}$. Увеличение динамического диапазона позволяет обрабатывать большие исходные данные с тем же размером долей, что позволяет снизить избыточность в k раз. С другой стороны, замена условия $p_0 < p_1$ в схеме Asmuth-Bloom условием 2 позволяет обеспечить вычислительную безопасность.

2.5.1 Вычислительная безопасность AC-RRNS

В данном разделе анализируется вычислительная безопасность схемы AC-RNNS в случае сговора. Условие 3 утверждает, что множество модулей RRNS представляет собой компактную последовательность, то есть $p_1 < p_2 < \dots < p_m < 2 \cdot p_1$. Следовательно, каждый облачный провайдер имеет примерно одинаковое количество информации об исходных данных. Покажем, что AC-RRNS минимизирует вероятность доступа к данным в результате сгово-

ра злоумышленников. Для этого докажем следующие утверждения, следствия и теорему.

Утверждение 2.5.1. *Если при использовании структуры доступа AC-RRNS с параметрами (k, n) , коалиция злоумышленников знает менее k долей и секретный ключ p_0 , то вероятность получения несанкционированного доступа к данным меньше $1/2^{l-1}$.*

Доказательство. Для множества $\bar{I} \subset \{1, 2, \dots, n\}$ мощностью меньше k можно вычислить значение S^* , удовлетворяющее равенству $S^* = |S|_{P_{\bar{I}}}$, где $P_{\bar{I}} = \prod_{i \in \bar{I}} p_i$. Следовательно, S можно представить как $S = S^* + P_{\bar{I}} \cdot w$, где целое число $w \in \left[0, \left\lfloor \frac{\beta}{P_{\bar{I}}} \right\rfloor\right]$. Каждому значению w соответствует значение C_w^* , вычисленное по следующей формуле $C_w^* = |S^* + P_{\bar{I}} \cdot w|_{p_0}$.

Принимая во внимание Условие 3, $P_{\bar{I}} \leq \prod_{i=0}^{k-2} p_{n-i}$. Следовательно, вероятность вычисления S при известном S^* удовлетворяет равенству $Pr(p(\bar{I})) \leq \frac{1}{\left\lfloor \frac{\beta}{P_{\bar{I}}} \right\rfloor} \leq \frac{1}{p_{n-k+1}} < \frac{1}{2^{l-1}}$.

Утверждение доказано. \square

Утверждение 2.5.2. *Если при использовании структуры доступа AC-RRNS с параметрами (k, n) $l > k$, вероятность получения несанкционированного доступа к данным на основе k или более известных долей без секретного ключа меньше $\frac{1}{2^{l \cdot (k-1)} \cdot (2^{l-k} - 1)}$.*

Доказательство. Из Условия 2, $\beta = \prod_{i=1}^k p_i > p_1^k$ и $\alpha = \prod_{i=0}^{k-2} p_{n-i} < (2p_1)^{k-1}$, отсюда следует, что мощность множества всех возможных секретных ключей p_0 удовлетворяет условию

$$\begin{aligned} \prod_{i=1}^k p_i - \prod_{i=0}^{k-2} p_{n-i} &= \beta - \alpha > p_1^k - 2p_1^{k-1} \\ &> 2^{(l-1)(k-1)} (2^{l-1} - 2^{k-1}) = 2^{l(k-1)} (2^{l-k} - 1). \end{aligned} \quad (2.33)$$

Тогда вероятность получения p_0 меньше чем $\frac{1}{2^{l \cdot (k-1)} \cdot (2^{l-k} - 1)}$.

Утверждение доказано. \square

Следствие 2.5.1. *Если $l > k$, то схема AC-RRNS с параметрами (k, n) удовлетворяет Условию 2.*

Доказательство. Условие 2 выполняется, если $\beta > \alpha$ (или $\beta - \alpha > 0$), таким образом, $\beta - \alpha > 2^{l(k-1)} (2^{l-k} - 1)$. Поскольку $l > k$, $2^{l-k} > 1$, отсюда $2^{l-k} - 1 > 0$, значит $\beta - \alpha > 0$.

Следствие доказано. \square

Следствие 2.5.2. *Если в структуре доступа AC-RRNS с параметрами (k, n) при $l > 2k$, противоборствующая коалиция знает менее k долей и не знает секретного ключа, то несанкционированный доступ к данным может быть получен с вероятностью меньше $\frac{1}{2^{l-k}}$, что эквивалентно полному перебору.*

Доказательство. Из Утверждений 2.5.1 и 2.5.2 следует, что вероятность получения несанкционированного доступа к данным без p_0 и k долей, следуя свойствам совместной вероятности (поскольку события независимы), меньше

$$\frac{1}{2^{l-1}} \cdot \frac{1}{2^{l \cdot (k-1)} (2^{l-k} - 1)} = \frac{1}{2^{l \cdot k-1} (2^{l-k} - 1)}. \quad (2.34)$$

Из условия 1 следует, что мощность множества всех возможных значений исходных данных S удовлетворяет условию

$$S < p_0 < \beta = \prod_{i=1}^k p_i < 2^{l \cdot k}. \quad (2.35)$$

Это означает, что вероятность получить несанкционированный доступ к данным перебором меньше $\frac{1}{2^{l \cdot k}}$.

Поскольку $2^{l \cdot k} = 2^{l \cdot k-1} \cdot 2 < 2^{l \cdot k-1} \cdot (2^k - 1) < 2^{l \cdot k-1} \cdot (2^{l-k} - 1)$, соответствующая вероятность выше, чем вероятность получить несанкционированный доступ к данным неполным набором долей.

Следовательно, вероятность получить несанкционированный доступ к данным равна

$$\max \left\{ \frac{1}{2^{l \cdot k}}, \frac{1}{2^{l \cdot k-1} \cdot (2^{l-k} - 1)} \right\} = \frac{1}{2^{l \cdot k}}. \quad (2.36)$$

Таким образом, в предложенной схеме (k, n) при $l > 2k$ сложность получения несанкционированного доступа к данным для противоборствующей коалиции с известными менее чем k долями и неизвестным секретным ключом эквивалентна сложности полного перебора.

Следствие доказано. \square

Докажем вычислительную безопасность предложенной схемы. Концепция вычислительной безопасности основана на следующей идее: информация не может быть эффективно восстановлена, если она не полна. Следовательно, схема является вычислительно безопасной, если злоумышленник зная исходные данные $S^{(1)}, S^{(2)}$ и неполные наборы долей $C^{(1)}, C^{(2)}$, не может однозначно отобразить $(S^{(1)}, C^{(1)})$ и $(S^{(2)}, C^{(2)})$.

Вычислительную безопасность для структуры доступа можно определить более строго [263] на основе концепции полиномиальной неразличимости [221]. Для любого распределения вероятностей $D(C, S)$ структура доступа является вычислительно безопасной, если для любой пары исходных данных $S^{(1)}, S^{(2)}$ и неполных наборов долей $C^{(1)}$ и $C^{(2)}$, распределения $D(C^{(1)}, S^{(1)})$ и $D(C^{(2)}, S^{(2)})$ полиномиально неразличимы, т.е. для любого вероятностного алгоритма A

$$\left| Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) - Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) \right| < \frac{1}{poly(n, k)}, \quad (2.37)$$

где $poly(n, k)$ – некоторый многочлен от количества возможных долей.

Теорема 2.5.1. *Схема AC-RRNS вычислительно безопасна.*

Доказательство. Для доказательства вычислительной безопасности предложенной схемы воспользуемся вспомогательным неравенством треугольника

$$\forall a, b, c \in \mathbb{R} : |a - b| \leq |a - c| + |b - c|. \quad (2.38)$$

Пусть $a = Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right)$, $b = Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right)$ и $c = Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right)$, получим

$$\begin{aligned} & \left| Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) - Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) \right| \\ & \leq \left| Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) - Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right) \right| \\ & \quad + \left| Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) - Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right) \right|, \quad (2.39) \end{aligned}$$

где $Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right)$ – вероятность получения несанкционированного доступа к данным с использованием менее k долей.

Для данной схемы шифрования не все варианты из множества общих исходов подходят. Нам подходят лишь те множества, которые определяют вычеты,

ограниченные величиной $p_1 \cdot \dots \cdot p_k$. Иными словами,

$$0 \geq S + p_0 \cdot rand < p_1 \cdot \dots \cdot p_k.$$

Нужно оценить число возможных "правильных" расширений неполной доли до полной. Общее количество исходов в данной схеме не превосходит $p_1 \cdot \dots \cdot p_k$. Любые k вычетов восстанавливают остальные, иначе любые k участника не смогли бы восстановить секрет.

$$Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) \leq \frac{\prod_{i=1}^k p_i}{\prod_{i=1}^n p_i} = \frac{1}{\prod_{i=k+1}^n p_i}, \quad (2.40)$$

$$Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) \leq \frac{\prod_{i=1}^k p_i}{\prod_{i=1}^n p_i} = \frac{1}{\prod_{i=k+1}^n p_i}, \quad (2.41)$$

$$Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right) = \frac{1}{\prod_{i=1}^k p_i}. \quad (2.42)$$

Из Условия 3 следует, что $p_0^k < \prod_{i=1}^k p_i < 2^k \cdot p_0^k$ и $p_0^{n-k} < \prod_{i=k+1}^n p_i < 2^{n-k} \cdot p_0^{n-k}$.

Таким образом,

$$\frac{1}{2^k p_0^k} < \frac{1}{\prod_{i=1}^k p_i} < \frac{1}{p_0^k}, \quad (2.43)$$

$$\frac{1}{2^{n-k} p_0^{n-k}} < \frac{1}{\prod_{i=k+1}^n p_i} < \frac{1}{p_0^{n-k}}. \quad (2.44)$$

Оценим члены в выражении (2.39)

$$\begin{aligned} & \left| Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) - Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right) \right| \\ & < \max \left\{ \frac{1}{p_0^{n-k}} - \frac{1}{2^k \cdot p_0^k}, \frac{1}{p_0^k} - \frac{1}{2^{n-k} p_0^{n-k}} \right\}, \quad (2.45) \end{aligned}$$

$$\begin{aligned} & \left| Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) - Pr \left(D \left(C^{(1)}, S^{(1)} \right) = 1 \right) \right| \\ & < \max \left\{ \frac{1}{p_0^{n-k}} - \frac{1}{2^k \cdot p_0^k}, \frac{1}{p_0^k} - \frac{1}{2^{n-k} p_0^{n-k}} \right\}. \quad (2.46) \end{aligned}$$

Подставляя (2.45) и (2.46) в (2.39), получаем

$$\begin{aligned} & \left| Pr \left(A \left(D \left(C^{(1)}, S^{(1)} \right) \right) = 1 \right) - Pr \left(A \left(D \left(C^{(2)}, S^{(2)} \right) \right) = 1 \right) \right| \\ & < 2 \max \left\{ \frac{1}{p_0^{n-k}} - \frac{1}{2^k \cdot p_0^k}, \frac{1}{p_0^k} - \frac{1}{2^{n-k} p_0^{n-k}} \right\}. \quad (2.47) \end{aligned}$$

Следовательно, схема AC-RRNS удовлетворяет формальному определению вычислительной безопасности.

Теорема доказана. \square

Теорема 2.5.1 имеет важное практическое значение. В частности, она доказывает, что злоумышленник не получит никакой дополнительной информации из неполного набора долей.

Пусть $(S^{(1)}, C^{(1)})$ и $(S^{(2)}, C^{(2)})$ определяются следующими соотношениями для всех $i \in [1, \dots, n]$:

$$c_i^{(1)} = \left| S^{(1)} + p_0 \cdot rand_1 \right|_{p_i}, c_i^{(2)} = \left| S^{(2)} + p_0 \cdot rand_2 \right|_{p_i}. \quad (2.48)$$

Поскольку $\forall i = \overline{1, n}: \gcd(p_0, p_i) = 1$, существуют $rand_1^*, rand_2^*, p_0^*$, такие, что выполняются следующие выражения

$$c_i^{(1)} = \left| S^{(2)} + p_0^* \cdot rand_2^* \right|_{p_i}, c_i^{(2)} = \left| S^{(1)} + p_0^* \cdot rand_1^* \right|_{p_i}. \quad (2.49)$$

Из (2.48) и (2.49) следует, что для однозначного отображения $(S^{(1)}, C^{(1)})$ и $(S^{(2)}, C^{(2)})$, требуется p_0 . Поскольку p_0 неизвестно, схема AC-RRNS вычислительно безопасна.

Продемонстрируем данный факт на примере.

Пример 2.5.1. Пусть структура доступа характеризуется параметрами $k = 2$, $n = 4$, и набором модулей RRNS $p_1 = 3221225473$, $p_2 = 3221225479$, $p_3 = 3221225533$, $p_4 = 3221225549$. Секретным ключом является $p_0 = 2635968733367020$. Пусть $S^{(1)} = 6323947392560$ и $S^{(2)} = 51771684174750$ – два сообщения, $rand_1 = 15$, и $rand_2 = 6$.

Если мы используем в качестве секретного ключа $p_0^* = 9089547356438$, а в качестве случайных значений $rand_1^* = 1745$, $rand_2^* = 4345$, то значения $S^{(1)}$ и $S^{(2)}$ обмениваются значениями, и, следовательно, невозможно однозначно отобразить $(S^{(1)}, C^{(1)})$ и $(S^{(2)}, C^{(2)})$.

Пример 2.5.1 демонстрирует, что схема вычислительно безопасна.

2.5.2 Свойства AC-RRNS

В данном разделе представлен сравнительный анализ предложенной схемы AC-RRNS со схемами Asmuth-Bloom [151] и Mignotte [280].

2.5.2.1 Избыточность данных

При использовании схемы AC-RRNS доли не превышают величин $(p_1 - 1, p_2 - 1, \dots, p_n - 1)$, поэтому суммарный размер хранимых данных ограничен сверху величиной $\sum_{i=1}^n \lceil \log_2 p_i \rceil$. Таким образом, длина входных данных примерно равна $\lceil \log_2 p_0 \rceil$. Избыточность вычисляется как отношение сохраненных закодированных данных к исходному размеру данных

$$\frac{\sum_{i=1}^n \lceil \log_2 p_i \rceil}{\lceil \log_2 p_0 \rceil} = \frac{n \cdot l}{\lceil \log_2 p_0 \rceil}, \quad (2.50)$$

где l – размер модуля RRNS (в битах).

Поскольку $\lceil \log_2 p_0 \rceil$ удовлетворяет неравенству $(k - 1)(l - 1) < \lceil \log_2 p_0 \rceil \leq k \cdot l$, избыточность данных удовлетворяет неравенству

$$\frac{n}{k} \leq \frac{\sum_{i=1}^n \lceil \log_2 p_i \rceil}{\lceil \log_2 p_0 \rceil} < \frac{nl}{(k - 1)(l - 1)}. \quad (2.51)$$

На рисунке 2.11 представлено сравнение избыточности предложенной схемы со схемами Asmuth-Bloom и Mignotte при размере модулей RRNS $l = 32$ бита. Из графиков можно сделать вывод о том, что предложенная схема имеет большую избыточность данных, чем схема Mignotte [280], и меньшую, чем асимптотически идеальная схема Asmuth-Bloom [151]. Однако, как было показано выше, предложенная схема AC-RRNS является вычислительно безопасной, чего нельзя сказать о схеме Mignotte [318].

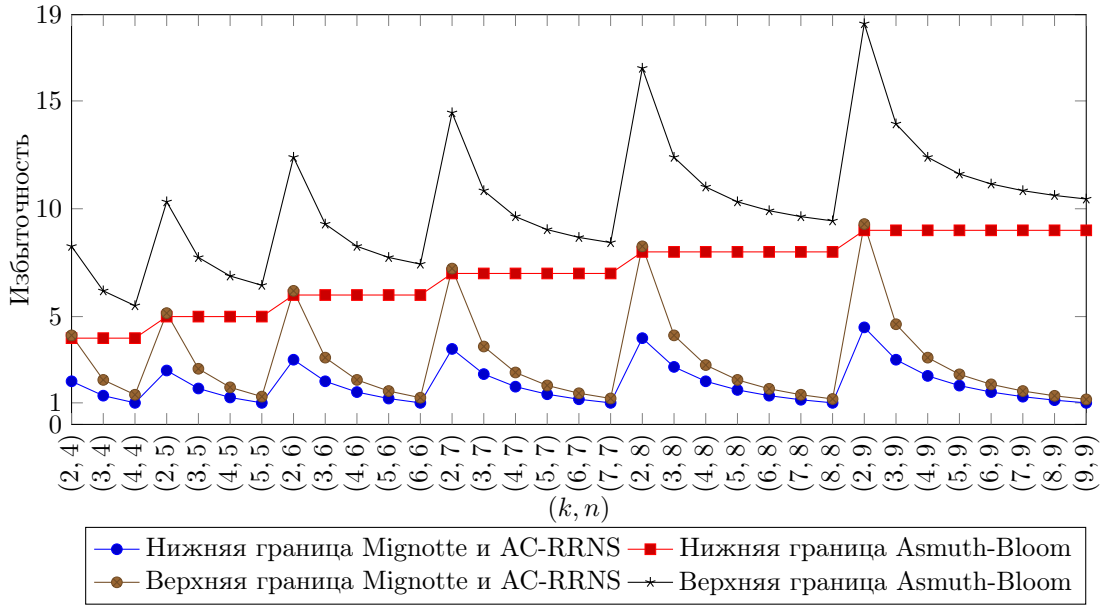


Рисунок 2.11 — Избыточность схем AC-RNNS, Asmuth-Bloom и Mignotte при длине модулей $l = 32$ бита

2.5.2.2 Вероятность получения несанкционированного доступа к данным посредством облачного сговора

Оценим вероятность получения данных в результате сговора при использовании схемы AC-RNNS. Пусть вероятность вступления злоумышленника в сговор равна pr . Вероятность создания коалиции из k или более членов может быть вычислена по формуле

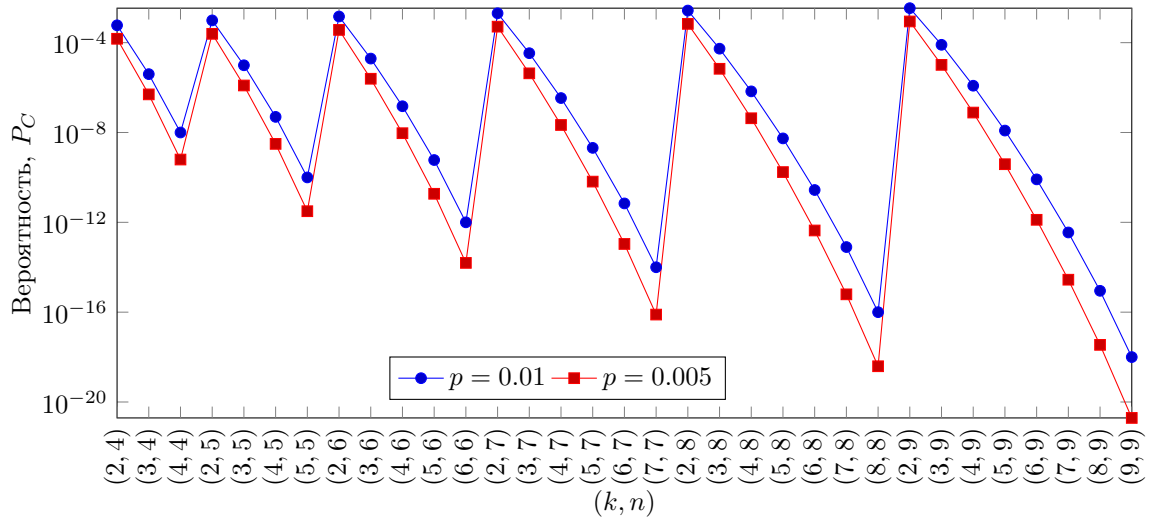
$$\sum_{i=k}^n \binom{n}{i} \cdot pr^i \cdot (1 - pr)^{n-i}, \quad (2.52)$$

где $\binom{n}{i} = C_n^i = \frac{n!}{i!(n-i)!}$ — количество сочетаний из n по i .

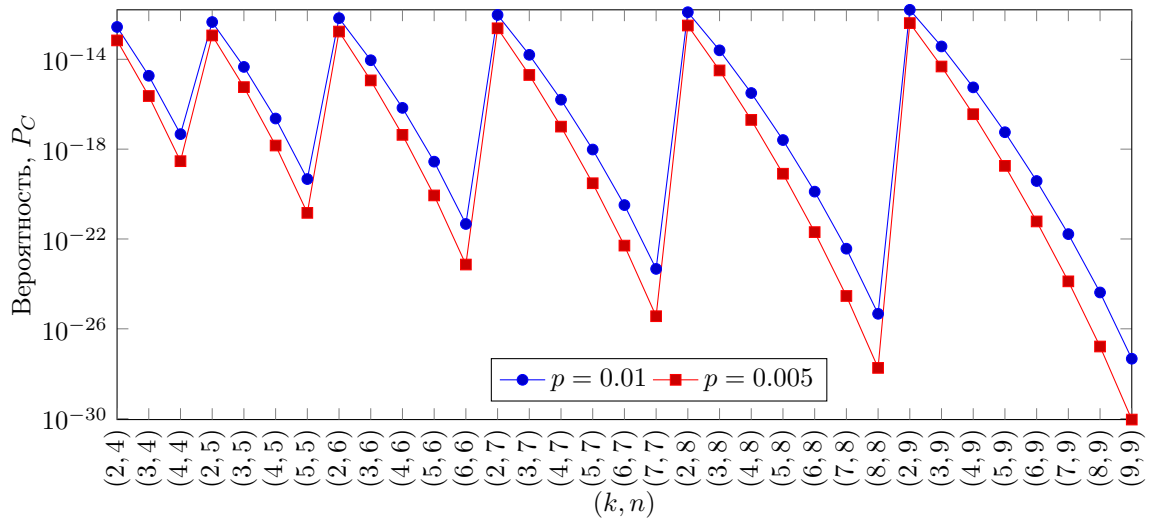
Вероятность получения доступа к данным составляет $\frac{1}{2^{l \cdot (k-1)}(2^{l-k}-1)}$ (утв. 2.5.2). Следовательно, вероятность создания коалиции P_C , которая может получить несанкционированный доступ к данным, может быть вычислена с использованием теоремы об умножении вероятностей независимых событий

$$P_C = \frac{1}{2^{l \cdot (k-1)}(2^{l-k}-1)} \cdot \sum_{i=k}^n \binom{n}{i} \cdot pr^i \cdot (1 - pr)^{n-i}. \quad (2.53)$$

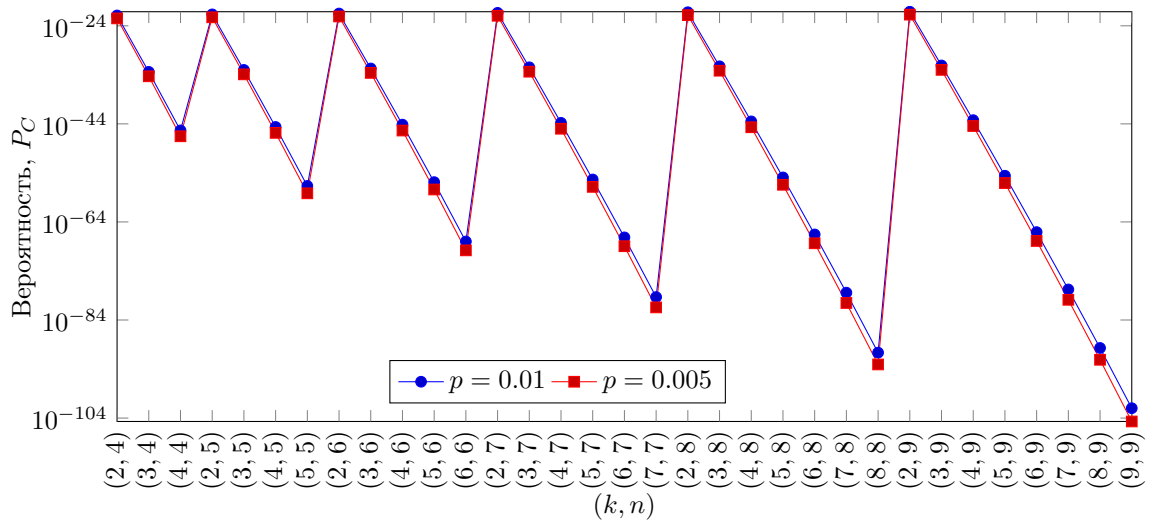
На рисунке 2.12 представлены вероятности P_C получения данных противоборствующей коалицией из k участников для схем Mignotte, Asmuth-Bloom



а) Схема Mignotte



б) Схема Asmuth-Bloom



в) Схема AC-RRNS

Рисунок 2.12 — Вероятность неавторизованного доступа коалиции из k злоумышленников для схем Mignotte, Asmuth-Bloom и AC-RRNS при длине модулей $l = 32$ бита

Таблица 15 — Основные характеристики сравниваемых структур доступа

| Схема | $< k$ долей | $\geq k$ долей | Секретный ключ | Вероятность получения несанкционированного доступа к данным (верхняя граница) | Избыточность (нижняя граница) | Диапазон (верхняя граница) | Совершенная | Асимптотически идеальная | Вычислительно безопасная | Противодействие сговору |
|--------------|-------------|----------------|----------------|---|-------------------------------|----------------------------|-------------|--------------------------|--------------------------|-------------------------|
| Asmuth-Bloom | • | | • | $\frac{1}{2^{l-1}}$ | n | p_0 | • | • | | • |
| | | • | | $\frac{1}{2^{l-1}}$ | | | | | | |
| | • | | | $\frac{1}{2^{l-1}}$ | | | | | | |
| Mignotte | | • | | 1 | $\frac{n}{k}$ | $\prod_{i=1}^k p_i$ | | | | |
| | • | | | $\frac{1}{2^{l-1}}$ | | | | | | |
| AC-RRNS | • | | • | $\frac{1}{2^{l-1}}$ | $\frac{n}{k}$ | $\prod_{i=1}^k p_i$ | • | | • | • |
| | | • | | $\frac{1}{2^{l(k-1)}(2^{l-k}-1)}$ | | | | | | |
| | • | | | $\frac{1}{2^{l \cdot k}}$ | | | | | | |

и AC-RRNS. При использовании AC-RRNS вероятность несанкционированного доступа к данным значительно меньше, чем в схемах Mignotte и Asmuth-Bloom. Наиболее высокие вероятности составляют $3.4 \cdot 10^{-3}$, $1.6 \cdot 10^{-12}$, $7.5 \cdot 10^{-22}$ для схем Mignotte, Asmuth-Bloom и AC-RRNS соответственно. Наименьшие вероятности равны $2.0 \cdot 10^{-21}$, $9.1 \cdot 10^{-31}$, $2.0 \cdot 10^{-105}$ соответственно в схемах Mignotte, Asmuth-Bloom и AC-RRNS. В таблице 15 показаны основные характеристики сравниваемых схем. Схемы Asmuth-Bloom и AC-RRNS способны решить проблему сговора в облаке. Однако, AC-RRNS вычислительно безопасна и вероятность получения несанкционированного доступа к данным в сценарии, когда противоборствующая коалиция не знает секретный ключ, намного ниже. Более того, AC-RRNS имеет значительно меньшую избыточность. Вышеперечисленные факты указывают на то, что схема AC-RRNS является наиболее оптимальной при построении реальных распределенных систем.

2.6 Выводы по второй главе

Вторая глава посвящена построению высокопроизводительной вычислительно стойкой структуры доступа, обладающей свойствами гомоморфизма колец, и обеспечивающей высокий уровень безопасности и надежности в нестационарной облачной среде. Предложена адаптивная распределенная служба хранения под названием WA-MRC-RRNS, которая реализует гомоморфное отображение и сочетает в себе функционал взвешенной пороговой структуры доступа и системы контроля корректности результатов обработки данных.

Использование взвешенной пороговой структуры доступа обусловлено доказанной теоремой о том, что вероятность потери данных при использовании взвешенной пороговой структуры доступа не превышает вероятности потери данных при использовании соответствующей классической пороговой структуры доступа. Показано, что в пессимистическом сценарии при настройке (3,4), вероятность потери данных при использовании WA-MRC-RRNS в 777.02 раза ниже, чем при использовании соответствующей классической пороговой структуры доступа MRC-RRNS. В среднем же вероятность потери данных при использовании WA-MRC-RRNS ниже в $9.23 \cdot 10^{17}$ раза.

Выбор избыточной системы остаточных классов (RRNS) в качестве основы для предложенной взвешенной пороговой структуры доступа обусловлен возможностью построения вычислительно стойкой схемы и реализации механизмов обнаружения/восстановления множественных ошибок данных. Кроме того, RRNS позволяет динамически настраивать параметры, чтобы справиться с различными объективными предпочтениями, рабочими нагрузками и свойствами облака.

Высокая производительность предложенной схемы достигается за счет разработанных алгоритмов кодирования/декодирования, основанных на переходе к представлению в обобщенной позиционной системе счисления (MRC), нейронной сети конечного кольца и их эффективной программной реализации. Сравнение предложенной схемы WA-MRC-RRNS с другой известной взвешенной схемой WA-AR-RRNS с точки зрения производительности дало следующие результаты: при кодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 13.73 раза, при декодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 385.07 раза. Отметим, что предложенная схема WA-MRC-RRNS так же превосходит клас-

сическую пороговую схему AR-RRNS с точки зрения производительности (в 4.83 раза при кодировании и в 120.04 раза при декодировании), проигрывая лишь классической пороговой схеме MRC-RRNS в 2.42 раза при кодировании и в 1.16 раза при декодировании. Данные потери в производительности абсолютно оправданы многократным повышением надежности и безопасности, достигаемым за счет использования взвешенной схемы WA-MRC-RRNS вместо классической пороговой схемы MRC-RRNS.

WA-MRC-RRNS – адаптивная схема, позволяющая динамически регулировать настройки (n_v, K, N) , чтобы справиться с отключениями, сбоями и изменением характеристик и параметров облачных сервисов. Настройки должны определяться экспериментально на основе накопленных статистических показателей. Статистический интервал времени должен быть установлен в соответствии с динамикой нестационарной среды и конфигурациями системы. Решение данных задач выходит за рамки данной работы и является предметом будущих исследований.

Для анализа предложенной схемы WA-MRC-RRNS с точки зрения безопасности данных, доказано утверждение, дающее оценку вероятности получения несанкционированного доступа к данным. Приведены вероятности получения несанкционированного доступа к данным для каждого из трех основных сценариев сговора: когда противоборствующая коалиция знает секретный ключ и не знает необходимое количество долей, не знает ни секретного ключа ни необходимого количества долей, а также не знает секретного ключа и знает необходимое количество долей. Для обеспечения безопасности данных предложено интегрировать WA-MRC-RRNS в разработанную конфигурируемую схему хранения данных AC-RRNS. Доказана вычислительная безопасность AC-RRNS. Сравнительный анализ предложенной схемы с известными структурами доступа, использующими аппарат RRNS, такими как схема HORNS, основанная на схеме Mignotte, и схема Asmuth-Bloom, дал следующие результаты: HORNS обладает меньшей избыточностью, но в отличие от предложенной схемы, не является вычислительно безопасной, уязвима для атаки открытым текстом и не может быть использована для решения проблемы сговора; схема Asmuth-Bloom является асимптотически идеальной, подходит для обеспечения безопасности данных при сговоре, но вводит избыточность в k раз превышающую избыточность предложенной схемы (k – параметр схемы Asmuth-Bloom). Кроме того, использование AC-RRNS многократно снижает вероятность неавторизованного

доступа к данным коалиции из k злоумышленников по сравнению со схемами HORNS и Asmuth-Bloom.

Таким образом, предложенная схема превосходит ранее разработанные аналоги по многим параметрам и соответствует требованиям, предъявляемым к гомоморфным кодам, используемым в распределенных средах хранения и обработки данных.

**Глава 3. РАЗРАБОТКА МЕТОДОВ И АЛГОРИТМОВ
ОПРЕДЕЛЕНИЯ ЗНАКА И СРАВНЕНИЯ ГОМОМОРФНО
ЗАКОДИРОВАННЫХ ЧИСЕЛ НАД КОЛЬЦОМ ВЫЧЕТОВ С
ДЕЛИТЕЛЯМИ НУЛЯ**

Знак числа в RNS над полной системой абсолютно наименьших вычетов по модулю m ($x \in [-\frac{m}{2}, \frac{m}{2})$) определяется следующей формулой

$$\text{sign}_m(x) = \begin{cases} 1, & \text{если } 0 < x < \frac{m}{2}, \\ 0, & \text{если } x = 0, \\ -1, & \text{если } -\frac{m}{2} \leq x < 0. \end{cases} \quad (3.1)$$

Учитывая, что $\forall x \in \mathbb{N}$ и $x \leq \frac{m}{2}$: $m - x \equiv -x \pmod{m}$, в системе наименьших неотрицательных вычетов по модулю m знак числа в RNS определяется следующим образом

$$\text{sign}_m(x) = \begin{cases} 1, & \text{если } 0 < x < \frac{m}{2}, \\ 0, & \text{если } x = 0, \\ -1, & \text{если } \frac{m}{2} \leq x < m. \end{cases} \quad (3.2)$$

В дальнейшем будем использовать обозначение $\text{sign}_m(x)$ для определения функции знака числа в полной системе наименьших положительных вычетов.

Исследуем вопрос об интерполяции функции знака числа алгебраическим многочленом над кольцом вычетов \mathbb{Z}_m . Покажем, что над полем Галуа можно задать функции определения знака числа (Раздел 4.1), а над кольцом вычетов с делителями нуля такого представления не существует. Для этого докажем следующие теоремы.

Теорема 3.0.1. *Если m – составное число, то в кольце $\mathbb{Z}_m[x]$ не существует многочлена $s(x) \in \mathbb{Z}_m[x]$, такого что $\forall x \in \mathbb{Z}_m$: $\text{sign}_m(x) = s(x)$.*

Доказательство. Предположим противное, что существует многочлен $s(x)$ степени d , такой что $\forall x \in \mathbb{Z}_m$: $\text{sign}_m(x) = s(x)$. Так как $\text{sign}_m(0) \neq \text{sign}_m(1)$, то $s(x) \neq \text{const}$, следовательно, $d \geq 1$. Значит многочлен $s(x)$ может быть задан в следующем виде

$$s(x) = \sum_{i=0}^d a_i \cdot x^i, \quad (3.3)$$

где $a_i \in \mathbb{Z}_m$, $d \geq 1$.

Так как $\text{sign}_m(0) = 0$, то $s(0) = a_0 = \text{sign}_m(0) = 0$, следовательно, $s(x)$, может быть представлен в виде

$$s(x) = x \cdot \sum_{i=0}^{d-1} a_{i+1} \cdot x^i. \quad (3.4)$$

Учитывая, что по условию теоремы m – составное число, существует нетривиальный делитель $\eta \neq 1, \eta \neq m$ числа m . Тогда $s(\eta)$ делит η и, следовательно, $s(\eta) \neq \pm 1$.

Теорема доказана. □

Из выше сказанного следует, что функцию определения знака числа над кольцом вычетов с делителями нуля интерполировать нельзя, следовательно, необходима разработка новых методов и алгоритмов сравнения чисел. В следующем разделе рассмотрен вопрос определения знака чисел, представленных кодами, основанными на RNS, и их модификаций.

3.1 Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с делителями нуля

Из определения кольца вычетов \mathbb{Z}_m , содержащего делители нуля, напрямую следует, что m – составное число. Согласно следствию из основной теоремы арифметики, любое натуральное число может быть представлено в виде $m = m_1^{d_1} \cdot m_2^{d_2} \cdot \dots \cdot m_n^{d_n}$, где $m_1 < m_2 < \dots < m_n$ – простые числа, а $\forall i = \overline{1, n}$: $d_i \in \mathbb{N}$. Пусть $\forall i = \overline{1, n}$: $p_i = m_i^{d_i}$, тогда выбирая в качестве оснований p_i , получим RNS с попарно взаимно простыми числами (основаниями или модулями RNS). Так как m составное число, то $n \geq 2$. Исследуем вопрос определения знака числа в подобных RNS.

3.1.1 Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с четным диапазоном

Докажем лемму, которая позволит свести операцию определения знака числа к $n - 1$ делению на модуль RNS.

Лемма 3.1.1. *Если $X, a, b \in N$ и $a, b \geq 2$, то выполняется равенство*

$$\left\lfloor \left\lfloor \frac{X}{a} \right\rfloor / b \right\rfloor = \left\lfloor \frac{X}{a \cdot b} \right\rfloor. \quad (3.5)$$

Доказательство. Так как $\lfloor \frac{X}{a} \rfloor = \frac{X - |X|_a}{a}$, то левая часть равенства (3.5) примет вид

$$\left\lfloor \left\lfloor \frac{X}{a} \right\rfloor / b \right\rfloor = \left\lfloor \frac{X - |X|_a}{a \cdot b} \right\rfloor. \quad (3.6)$$

Представим X в виде $X = a \cdot b \lfloor \frac{X}{a \cdot b} \rfloor + |X|_{a \cdot b}$, подставим в (3.6), получим

$$\left\lfloor \frac{X - |X|_a}{a \cdot b} \right\rfloor = \left\lfloor \left\lfloor \frac{X}{a \cdot b} \right\rfloor + \frac{|X|_{a \cdot b} - |X|_a}{a \cdot b} \right\rfloor = \left\lfloor \frac{X}{a \cdot b} \right\rfloor + \left\lfloor \frac{|X|_{a \cdot b} - |X|_a}{a \cdot b} \right\rfloor. \quad (3.7)$$

Пусть $X = \omega \cdot a + \gamma$, где $0 \leq \gamma \leq a - 1$.

Тогда $|\omega \cdot a|_{a \cdot b} = \omega \cdot a - \lfloor \frac{\omega \cdot a}{a \cdot b} \rfloor \cdot a \cdot b = a \left(\omega - b \cdot \lfloor \frac{\omega}{b} \rfloor \right) = a \cdot |\omega|_b$.

Так как $0 \leq |\omega|_b \leq b - 1$, $0 \leq a \cdot |\omega|_b + \gamma \leq a \cdot (b - 1) + a - 1 = a \cdot b - 1$.

Значит $|X|_{a \cdot b} = |\omega \cdot a + \gamma|_{a \cdot b} = ||\omega \cdot a|_{a \cdot b} + \gamma|_{a \cdot b} = |a \cdot |\omega|_b + \gamma|_{a \cdot b} = a \cdot |\omega|_b + \gamma$.

Следовательно, $|X|_{a \cdot b} - |X|_a = a \cdot |\omega|_b + \gamma - \gamma = a \cdot |\omega|_b$ и $0 \leq |X|_{a \cdot b} - |X|_a \leq a \cdot (b - 1)$, значит $\left\lfloor \frac{|X|_{a \cdot b} - |X|_a}{a \cdot b} \right\rfloor = 0$.

Подставим (3.7) в (3.6). С учетом последнего полученного выражения, (3.6) примет вид $\left\lfloor \left\lfloor \frac{X}{a} \right\rfloor / b \right\rfloor = \left\lfloor \frac{X}{a \cdot b} \right\rfloor$.

Лемма доказана. \square

В пределах кольца вычетов \mathbb{Z}_m , m определяет динамический диапазон RNS. Перейдем к более удобному для доказательства последующих лемм и теорем обозначению динамического диапазона RNS $m = M_n = \prod_{i=1}^n p_i$. Для четного динамического диапазона знак числа в RNS определим согласно следующей формуле

$$S(X) = \begin{cases} 0, & \text{если } 0 \leq X \leq \frac{M_n}{2} - 1, \\ 1, & \text{если } \frac{M_n}{2} \leq X \leq M_n - 1, \end{cases} \quad (3.8)$$

Так как диапазон RNS – четное число, то, следовательно, один из модулей RNS является четным числом. Без потери общности будем считать, что модуль p_n четный, тогда используя свойство $\lfloor \lfloor \frac{X}{a} \rfloor / b \rfloor = \lfloor \frac{X}{a \cdot b} \rfloor$ (формула (3.5)), определить знак числа в RNS можно в два этапа: первый этап – деление на целое число $M_{n-1} = \frac{M_n}{p_n} = \prod_{i=1}^{n-1} p_i$, второй этап – деление на целое число $\frac{p_n}{2}$. Формально математически данный алгоритм определяется следующей формулой

$$S(X) = \left\lfloor \frac{2X}{M_n} \right\rfloor = \left\lfloor \left\lfloor \frac{X}{M_{n-1}} \right\rfloor \cdot \frac{2}{p_n} \right\rfloor. \quad (3.9)$$

Следовательно,

$$S(X) = \left\lfloor \left[\left[\left[\left[\left\lfloor \frac{X}{p_1} \right\rfloor / p_2 \right] \dots \right] / p_{n-1} \right] \cdot \frac{2}{p_n} \right] \right\rfloor,$$

Запишем процесс определения знака числа в RNS в виде алгоритма.

Алгоритм 1: Определение знака числа для случая, когда диапазон RNS – четное число

Input: $\{p_1, p_2, \dots, p_{n-1}, p_n\}$ – модули RNS,

$X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ – представление числа X в RNS,

$w_{i,j} = |p_i^{-1}|_{p_j}$ – синаптические веса

Output: 0 ($X \geq 0$) или 1 ($X < 0$)

1 **for** $i = 1, i < n, i++$ **do**
 2 **for** $j = i + 1, j \leq n, j++$ **do**
 3 $x_j = |(x_j - x_i) \cdot w_{i,j}|_{p_j}$

Result: $\left\lfloor \frac{2x_n}{p_n} \right\rfloor$.

Рассмотрим работу Алгоритма 1 на примере.

Пример 3.1.1. Пусть задана система остаточных классов с модулями $p_1 = 17, p_2 = 19, p_3 = 23, p_4 = 32$. Определим знак числа $X = 118863 \xrightarrow{RNS} (16, 18, 22, 15)$ и знак числа $Y = 118864 \xrightarrow{RNS} (0, 0, 0, 16)$.

Вычислим синаптические веса:

$$\begin{aligned} w_{1,2} &= |p_1^{-1}|_{p_2} = \left| \frac{1}{17} \right|_{19} = 9, & w_{1,3} &= |p_1^{-1}|_{p_3} = \left| \frac{1}{17} \right|_{23} = 19, \\ w_{1,4} &= |p_1^{-1}|_{p_4} = \left| \frac{1}{17} \right|_{32} = 17, & w_{2,3} &= |p_2^{-1}|_{p_3} = \left| \frac{1}{19} \right|_{23} = 17, \\ w_{2,4} &= |p_2^{-1}|_{p_4} = \left| \frac{1}{19} \right|_{32} = 27, & w_{3,4} &= |p_3^{-1}|_{p_4} = \left| \frac{1}{23} \right|_{32} = 7. \end{aligned}$$

Для удобства, промежуточные результаты вычислений $S(X)$ и $S(Y)$ запишем в таблицы 16 и 17 соответственно. Так как

Таблица 16 — Вычисление значения функции $S(X)$

| Операция | p | 17 | 19 | 23 | 32 |
|--|------------------|----|----|----|----|
| | X | 16 | 18 | 22 | 15 |
| $X^{(1)} = \left\lfloor \frac{X}{p_1} \right\rfloor$ | $-x_1$ | 0 | 2 | 6 | 31 |
| | $\times w_{1,j}$ | - | 18 | 22 | 15 |
| $X^{(2)} = \left\lfloor \frac{X^{(1)}}{p_2} \right\rfloor$ | $-x_2^{(1)}$ | - | 0 | 4 | 29 |
| | $\times w_{2,j}$ | - | - | 22 | 15 |
| $X^{(3)} = \left\lfloor \frac{X^{(2)}}{p_3} \right\rfloor$ | $-x_3^{(2)}$ | - | - | 0 | 25 |
| | $\times w_{3,j}$ | - | - | - | 15 |

Таблица 17 — Вычисление значения функции $S(Y)$

| Операция | p | 17 | 19 | 23 | 32 |
|--|------------------|----|----|----|----|
| | Y | 0 | 0 | 0 | 16 |
| $Y^{(1)} = \left\lfloor \frac{Y}{p_1} \right\rfloor$ | $-y_1$ | 0 | 0 | 0 | 16 |
| | $\times w_{1,j}$ | - | 0 | 0 | 16 |
| $Y^{(2)} = \left\lfloor \frac{Y^{(1)}}{p_2} \right\rfloor$ | $-y_2^{(1)}$ | - | 0 | 0 | 16 |
| | $\times w_{2,j}$ | - | - | 0 | 16 |
| $Y^{(3)} = \left\lfloor \frac{Y^{(2)}}{p_3} \right\rfloor$ | $-y_3^{(2)}$ | - | - | 0 | 16 |
| | $\times w_{3,j}$ | - | - | - | 16 |

$S(X) = \left\lfloor \frac{2x_4^{(3)}}{p_4} \right\rfloor = \left\lfloor \frac{2 \cdot 15}{32} \right\rfloor = 0 < 1$ то, в RNS с модулями $p_1 = 17, p_2 = 19, p_3 = 23, p_4 = 32$ число $X = 118863 \xrightarrow{RNS} (16, 18, 22, 15)$ положительное. Так как $S(Y) = \left\lfloor \frac{2y_4^{(3)}}{p_4} \right\rfloor = \left\lfloor \frac{2 \cdot 16}{32} \right\rfloor = 1$ то, в RNS с модулями $p_1 = 17, p_2 = 19, p_3 = 23, p_4 = 32$ число $Y = 118864 \xrightarrow{RNS} (0, 0, 0, 16)$ отрицательное.

3.1.2 Методы определения знака числа над кольцом вычетов \mathbb{Z}_m с нечетным диапазоном

Аналогично тому, как это было сделано для кольца вычетов \mathbb{Z}_m , где m – четное число, перейдем к более удобному обозначению динамического диапазона RNS $m = M_n = \prod_{i=1}^n p_i$. Тогда M_i обозначает динамический диапазон,

обеспечиваемый первыми i модулями, равный их произведению, а $S_i(X)$ – промежуточный знак редуцированного числа X в RNS с модулями $\{p_1, p_2, \dots, p_i\}$. Таким образом, в n -модульной RNS, $S(X) = S_n(X)$.

Для нечетного динамического диапазона знак числа в RNS определим согласно следующей формуле

$$S(X) = \begin{cases} 0, & \text{если } 0 \leq X \leq \frac{M_n-1}{2}, \\ 1, & \text{если } \frac{M_n+1}{2} \leq X \leq M_n - 1, \end{cases} \quad (3.10)$$

т.е. 0 – если число неотрицательное, 1 – если число отрицательное.

Лемма 3.1.2. *Если $M_n = \prod_{i=1}^n p_i$ нечетное число и $n \geq 2$, то для любого целого числа $X \in [0, M_n - 1]$ выполняется равенство*

$$S_n(X) = \bar{S}_n(X), \quad (3.11)$$

где $S_n(X) = \left\lfloor \frac{2X}{M_n} \right\rfloor$, а $\bar{S}_n(X) = \left\lfloor \left\lfloor \frac{X}{M_{n-1}} + \frac{M_{n-1}-1}{2M_{n-1}} \right\rfloor \frac{2}{p_n+1} \right\rfloor$.

Доказательство. Так как $S_n(X)$ и $\bar{S}_n(X)$ являются возрастающими функциями и

$$S_n(X) = \left\lfloor \frac{2X}{M_n} \right\rfloor = \begin{cases} 0, & \text{если } 0 \leq X \leq \frac{M_n-1}{2}, \\ 1, & \text{если } \frac{M_n+1}{2} \leq X \leq M_n - 1, \end{cases} \quad (3.12)$$

то для доказательства равенства (3.11) необходимо и достаточно показать, что выполняются четыре равенства: $\bar{S}_n(0) = S_n(0)$, $\bar{S}_n\left(\frac{M_n-1}{2}\right) = S_n\left(\frac{M_n-1}{2}\right)$, $\bar{S}_n\left(\frac{M_n+1}{2}\right) = S_n\left(\frac{M_n+1}{2}\right)$, $\bar{S}_n(M_n - 1) = S_n(M_n - 1)$. Для проверки равенств вычислим значение функции $\bar{S}_n(X)$ в четырех точках

$$X \in \left\{ 0, \frac{M_n - 1}{2}, \frac{M_n + 1}{2}, M_n - 1 \right\}. \quad (3.13)$$

$$\begin{aligned} \bar{S}_n(0) &= \left\lfloor \left\lfloor \frac{M_{n-1} - 1}{2M_{n-1}} \right\rfloor \frac{2}{p_n + 1} \right\rfloor \\ &= \left\lfloor \left\lfloor \frac{1}{2} - \frac{1}{2M_{n-1}} \right\rfloor \frac{2}{p_n + 1} \right\rfloor. \end{aligned} \quad (3.14)$$

Так как $M_{n-1} \geq 3$, то $\frac{1}{2M_{n-1}} \leq \frac{1}{6}$, следовательно, $0 < \frac{1}{2} - \frac{1}{2M_{n-1}} < \frac{1}{2}$ и $\left\lfloor \frac{1}{2} - \frac{1}{2M_{n-1}} \right\rfloor = 0$. Подставляя $\left\lfloor \frac{1}{2} - \frac{1}{2M_{n-1}} \right\rfloor = 0$ в (3.14), получим $\bar{S}_n(0) = 0$.

$$\begin{aligned} \bar{S}_n\left(\frac{M_n - 1}{2}\right) &= \left\lfloor \left\lfloor \frac{M_n - 1}{2M_{n-1}} + \frac{M_{n-1} - 1}{2M_{n-1}} \right\rfloor \frac{2}{p_n + 1} \right\rfloor \\ &= \left\lfloor \left\lfloor \frac{p_n + 1}{2} - \frac{1}{M_{n-1}} \right\rfloor \frac{2}{p_n + 1} \right\rfloor. \end{aligned} \quad (3.15)$$

Так как p_n нечетное число, то $\frac{p_n+1}{2}$ является целым числом, учитывая что $0 < \frac{1}{M_{n-1}} \leq \frac{1}{3}$, получим $\left\lfloor \frac{p_n+1}{2} - \frac{1}{M_{n-1}} \right\rfloor = \frac{p_n+1}{2} - 1 = \frac{p_n-1}{2}$. С учетом последнего полученного равенства, выражение (3.15) примет вид $\bar{S}_n\left(\frac{M_n-1}{2}\right) = \left\lfloor \frac{p_n-1}{2} \cdot \frac{2}{p_n+1} \right\rfloor = 0$.

$$\begin{aligned} \bar{S}_n\left(\frac{M_n+1}{2}\right) &= \left\lfloor \left\lfloor \frac{M_n+1}{2M_{n-1}} + \frac{M_{n-1}-1}{2M_{n-1}} \right\rfloor \frac{2}{p_n+1} \right\rfloor \\ &= \left\lfloor \left\lfloor \frac{p_n+1}{2} \right\rfloor \frac{2}{p_n+1} \right\rfloor \\ &= \left\lfloor \frac{p_n+1}{2} \cdot \frac{2}{p_n+1} \right\rfloor = 1. \end{aligned} \quad (3.16)$$

$$\begin{aligned} \bar{S}_n(M_n-1) &= \left\lfloor \left\lfloor \frac{M_n-1}{M_{n-1}} + \frac{M_{n-1}-1}{2M_{n-1}} \right\rfloor \frac{2}{p_n+1} \right\rfloor \\ &= \left\lfloor \left\lfloor p_n + \frac{1}{2} - \frac{3}{2M_{n-1}} \right\rfloor \frac{2}{p_n+1} \right\rfloor. \end{aligned} \quad (3.17)$$

Так как $p_n \in \mathbb{Z}$, то $\left\lfloor p_n + \frac{1}{2} - \frac{3}{2M_{n-1}} \right\rfloor = p_n + \left\lfloor \frac{1}{2} - \frac{3}{2M_{n-1}} \right\rfloor$. Учитывая что $M_{n-1} \geq 3$, получим $0 \leq \frac{1}{2} - \frac{3}{2M_{n-1}} < \frac{1}{2}$. Следовательно, $\left\lfloor \frac{1}{2} - \frac{3}{2M_{n-1}} \right\rfloor = 0$. Подставив $\left\lfloor p_n + \frac{1}{2} - \frac{3}{2M_{n-1}} \right\rfloor = p_n$ в (3.17), получим $\bar{S}_n(M_n-1) = \left\lfloor \frac{2p_n}{p_n+1} \right\rfloor = 2 + \left\lfloor -\frac{2}{p_n+1} \right\rfloor$.

Так как $\lfloor -X \rfloor = -\lceil X \rceil$ и $0 < \frac{2}{p_n+1} \leq \frac{1}{2}$, то $\bar{S}_n(M_n-1) = 2 - \left\lceil \frac{2}{p_n+1} \right\rceil = 2 - 1 = 1$.

Равенства $\bar{S}_n(0) = S_n(0)$, $\bar{S}_n\left(\frac{M_n-1}{2}\right) = S_n\left(\frac{M_n-1}{2}\right)$, $\bar{S}_n\left(\frac{M_n+1}{2}\right) = S_n\left(\frac{M_n+1}{2}\right)$, $\bar{S}_n(M_n-1) = S_n(M_n-1)$ выполняются, следовательно, $S_n(X) = \bar{S}_n(X)$ для любого целого числа $X \in [0, M_n-1]$.

Лемма доказана. \square

Лемма 3.1.3. Если $M_n = \prod_{i=1}^n p_i$ нечетное число и $n \geq 2$, то для любого целого числа $X \in [0, M_n-1]$ выполняется равенство

$$S_n(X) = \tilde{S}_n(X), \quad (3.18)$$

где $S_n(X) = \left\lfloor \frac{2X}{M_n} \right\rfloor$, а $\tilde{S}_n(X) = \left\lfloor \frac{X}{M_n} + \frac{1}{2} - \frac{1}{2M_n} \right\rfloor$.

Доказательство. По аналогии с доказательством Леммы 3.1.2, вычислим значения функции $\tilde{S}_n(X)$ в четырех точках $X \in \left\{0, \frac{M_n-1}{2}, \frac{M_n+1}{2}, M_n-1\right\}$.

$$\tilde{S}_n(0) = \left\lfloor \frac{1}{2} - \frac{1}{2M_n} \right\rfloor = 0,$$

доказано ранее в Лемме 3.1.2.

$$\begin{aligned}\tilde{S}_n\left(\frac{M_n-1}{2}\right) &= \left\lfloor \frac{M_n-1}{2M_n} + \frac{1}{2} - \frac{1}{2M_n} \right\rfloor = \left\lfloor 1 - \frac{1}{M_n} \right\rfloor = 0. \\ \tilde{S}_n\left(\frac{M_n+1}{2}\right) &= \left\lfloor \frac{M_n+1}{2M_n} + \frac{1}{2} - \frac{1}{2M_n} \right\rfloor = 1. \\ \tilde{S}_n(M_n-1) &= \left\lfloor \frac{M_n-1}{M_n} + \frac{1}{2} - \frac{1}{2M_n} \right\rfloor = 1 + \left\lfloor \frac{1}{2} - \frac{3}{2M_n} \right\rfloor = 1,\end{aligned}$$

также доказано ранее в Лемме 3.1.2.

Так как $S_n(X)$ и $\tilde{S}_n(X)$ возрастающие функции, и их значения в четырех точках $X \in \{0, \frac{M_n-1}{2}, \frac{M_n+1}{2}, M_n-1\}$ равны, то по аналогии с Леммой 3.1.2, $S_n(X) = \tilde{S}_n(X)$ для любого целого числа $X \in [0, M_n-1]$.

Лемма доказана. \square

Обобщая утверждения, доказанные в Леммах 3.1.1, 3.1.2 и 3.1.3, предложим алгоритм определения знака числа в RNS для случая, когда диапазон RNS M_n – нечетное число. Докажем корректность работы Алгоритма 2.

Алгоритм 2: Определение знака числа для случая, когда диапазон RNS – нечетное число

Input: $\{p_1, p_2, \dots, p_{n-1}, p_n\}$ – модули RNS,

$X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ – представление числа X в RNS,

$w_{i,j} = |p_i^{-1}|_{p_j}$ – синаптические веса

Output: 0 ($X \geq 0$) или 1 ($X < 0$)

1 $S_1 = (2x_1) \operatorname{div} p_1;$

2 **for** $i = 1, i < n, i++$ **do**

3 **for** $j = i + 1, j \leq n, j++$ **do**

4 $x_j = |(x_j - x_i) \cdot w_{i,j}|_{p_j};$

5 $S_{i+1} = (2 \cdot (x_{i+1} + S_i)) \operatorname{div} (p_{i+1} + 1)$

Result: S_n

Теорема 3.1.1. Алгоритм 2 корректен.

Доказательство. Из Леммы 3.1.2, следует что

$$S_n(X) = \bar{S}_n(X) = \left\lfloor \left\lfloor \frac{X}{M_{n-1}} + \frac{M_{n-1}-1}{2M_{n-1}} \right\rfloor \frac{2}{p_n+1} \right\rfloor. \quad (3.19)$$

Представим $\frac{X}{M_{n-1}}$ в виде суммы целой части и дробной $\frac{X}{M_{n-1}} = \left\lfloor \frac{X}{M_{n-1}} \right\rfloor + \frac{|X|_{M_{n-1}}}{M_{n-1}}$. Подставляя данное выражение в (3.19), получим

$$S_n(X) = \left\lfloor \left(\left\lfloor \frac{X}{M_{n-1}} \right\rfloor + \left\lfloor \frac{|X|_{M_{n-1}}}{M_{n-1}} + \frac{M_{n-1} - 1}{2M_{n-1}} \right\rfloor \right) \frac{2}{p_n + 1} \right\rfloor. \quad (3.20)$$

Учитывая что $\frac{M_{n-1}-1}{2M_{n-1}} = \frac{1}{2} - \frac{1}{2M_{n-1}}$, выражение (3.20) примет вид

$$S_n(X) = \left\lfloor \left(\left\lfloor \frac{X}{M_{n-1}} \right\rfloor + \left\lfloor \frac{|X|_{M_{n-1}}}{M_{n-1}} + \frac{1}{2} - \frac{1}{2M_{n-1}} \right\rfloor \right) \frac{2}{p_n + 1} \right\rfloor. \quad (3.21)$$

Учитывая что $\tilde{S}_{n-1}(|X|_{M_{n-1}}) = \left\lfloor \frac{|X|_{M_{n-1}}}{M_{n-1}} + \frac{1}{2} - \frac{1}{2M_{n-1}} \right\rfloor$, выражение (3.21) примет вид

$$S_n(X) = \left\lfloor \left(\left\lfloor \frac{X}{M_{n-1}} \right\rfloor + \tilde{S}_{n-1}(|X|_{M_{n-1}}) \right) \frac{2}{p_n + 1} \right\rfloor. \quad (3.22)$$

Из Леммы 3.1.3 следует, что $\tilde{S}_{n-1}(|X|_{M_{n-1}}) = S_{n-1}(|X|_{M_{n-1}})$, поэтому выражение (3.22) можно записать в следующем виде

$$S_n(|X|_{M_n}) = \left\lfloor \left(\left\lfloor \frac{|X|_{M_n}}{M_{n-1}} \right\rfloor + S_{n-1}(|X|_{M_{n-1}}) \right) \frac{2}{p_n + 1} \right\rfloor. \quad (3.23)$$

Использование рекурсивного соотношения (3.23) позволяет вычислить $S_n(|X|_{M_n})$. Отметим, что $S_n(|X|_{M_n}) = S_n(X)$, т.к. $|X|_{M_n} = X$. Помещая значение $\left\lfloor \frac{|X|_{M_i}}{M_{i-1}} \right\rfloor$ в переменную $x_i^{(i-1)}$, получим рекуррентное соотношение для определения знака числа $S_{i+1} = \left(2 \cdot \left(x_{i+1}^{(i)} + S_i \right) \right) \operatorname{div} (p_{i+1} + 1)$. При реализации алгоритма значение $x_i^{(i-1)}$ заменяется значением $x_{i+1}^{(i)}$ на каждой итерации, поэтому верхняя индексация не нужна. Окончательно, получим рекуррентное соотношение для определения знака числа $S_{i+1} = (2 \cdot (x_{i+1} + S_i)) \operatorname{div} (p_{i+1} + 1)$. Следовательно, Алгоритм 2 корректен.

Теорема доказана. \square

Рассмотрим работу Алгоритма 2 на примере.

Пример 3.1.2. Пусть задана система остаточных классов с модулями $p_1 = 17$, $p_2 = 19$, $p_3 = 23$ и $p_4 = 31$.

1. Определим знак числа $X = 115149 \xrightarrow{RNS} (8, 9, 11, 15)$.

Таблица 18 — Вычисление значения функции $S(X)$

| Операция | p | 17 | 19 | 23 | 31 |
|--|------------------|----|----|----|----|
| | X | 8 | 9 | 11 | 15 |
| $X^{(1)} = \left\lfloor \frac{X}{p_1} \right\rfloor$ | $-x_1$ | 0 | 1 | 3 | 7 |
| | $\times w_{1,j}$ | - | 9 | 11 | 15 |
| $X^{(2)} = \left\lfloor \frac{X^{(1)}}{p_2} \right\rfloor$ | $-x_2^{(1)}$ | - | 0 | 2 | 6 |
| | $\times w_{2,j}$ | - | - | 11 | 15 |
| $X^{(3)} = \left\lfloor \frac{X^{(2)}}{p_3} \right\rfloor$ | $-x_3^{(2)}$ | - | - | 0 | 4 |
| | $\times w_{3,j}$ | - | - | - | 15 |

Таблица 19 — Вычисление значения функции $S(Y)$

| Операция | p | 17 | 19 | 23 | 31 |
|--|------------------|----|----|----|----|
| | Y | 9 | 10 | 12 | 16 |
| $Y^{(1)} = \left\lfloor \frac{Y}{p_1} \right\rfloor$ | $-y_1$ | 0 | 1 | 3 | 7 |
| | $\times w_{1,j}$ | - | 9 | 11 | 15 |
| $Y^{(2)} = \left\lfloor \frac{Y^{(1)}}{p_2} \right\rfloor$ | $-y_2^{(1)}$ | - | 0 | 2 | 6 |
| | $\times w_{2,j}$ | - | - | 11 | 15 |
| $Y^{(3)} = \left\lfloor \frac{Y^{(2)}}{p_3} \right\rfloor$ | $-y_3^{(2)}$ | - | - | 0 | 4 |
| | $\times w_{3,j}$ | - | - | - | 15 |

2. Определим знак числа $Y = 115150 \xrightarrow{RNS} (9, 10, 12, 16)$.

Вычислим синаттические веса $w_{i,j}$, получим $w_{1,2} = |p_1^{-1}|_{p_2} = \left| \frac{1}{17} \right|_{19} = 9$,
 $w_{1,3} = |p_1^{-1}|_{p_3} = \left| \frac{1}{17} \right|_{23} = 19$, $w_{1,4} = |p_1^{-1}|_{p_4} = \left| \frac{1}{17} \right|_{31} = 11$, $w_{2,3} = |p_2^{-1}|_{p_3} = \left| \frac{1}{19} \right|_{23} = 17$,
 $w_{2,4} = |p_2^{-1}|_{p_4} = \left| \frac{1}{19} \right|_{31} = 18$, $w_{3,4} = |p_3^{-1}|_{p_4} = \left| \frac{1}{23} \right|_{31} = 27$.

1. Вычислим знак числа $X = (8, 9, 11, 15)$. Результаты промежуточных вычислений занесем в таблицу 18.

$$S_1(X) = \left\lfloor \frac{2x_1}{p_1} \right\rfloor = \left\lfloor \frac{2 \cdot 8}{17} \right\rfloor = 0,$$

$$S_2(X) = \left\lfloor \left(x_2^{(1)} + S_1(X) \right) \frac{2}{p_2+1} \right\rfloor = \left\lfloor (9 + 0) \frac{2}{19+1} \right\rfloor = 0,$$

$$S_3(X) = \left\lfloor \left(x_3^{(2)} + S_2(X) \right) \frac{2}{p_3+1} \right\rfloor = \left\lfloor (11 + 0) \frac{2}{23+1} \right\rfloor = 0,$$

$$S_4(X) = \left\lfloor \left(x_4^{(3)} + S_3(X) \right) \frac{2}{p_4+1} \right\rfloor = \left\lfloor (15 + 0) \frac{2}{31+1} \right\rfloor = 0.$$

Так как $S(X) = S_4(X) = 0$, то $X \geq 0$.

2. Вычислим знак числа $Y = (9, 10, 12, 16)$. Результаты промежуточных вычислений занесем в таблицу 19.

$$\begin{aligned}
S_1(Y) &= \left\lfloor \frac{2y_1}{p_1} \right\rfloor = \left\lfloor \frac{2 \cdot 9}{17} \right\rfloor = 1 \\
S_2(Y) &= \left\lfloor \left(y_2^{(1)} + S_1(Y) \right) \frac{2}{p_2+1} \right\rfloor = \left\lfloor (9+1) \frac{2}{19+1} \right\rfloor = 1 \\
S_3(Y) &= \left\lfloor \left(y_3^{(2)} + S_2(Y) \right) \frac{2}{p_3+1} \right\rfloor = \left\lfloor (11+1) \frac{2}{23+1} \right\rfloor = 1 \\
S_4(Y) &= \left\lfloor \left(y_4^{(3)} + S_3(Y) \right) \frac{2}{p_4+1} \right\rfloor = \left\lfloor (15+1) \frac{2}{31+1} \right\rfloor = 1 \\
\text{Так как } S(Y) &= S_4(Y) = 1, \text{ то } Y < 0.
\end{aligned}$$

3.2 Подходы к сравнению чисел в кольце вычетов \mathbb{Z}_m

В двоичной системе счисления, как известно, существует эффективный алгоритм сравнения чисел. Основная идея этого алгоритма заключается в последовательном сравнении цифр соответствующих разрядов. В непозиционных системах по понятным причинам данный подход не работает. В связи с этим простых алгоритмов сравнения чисел в RNS попросту нет [351].

Обратим внимание на тот факт, что функцию сравнения чисел заданную над полем, можно реализовать с помощью интерполяционной функции Лагранжа от двух переменных [247]. Рассмотрим вопрос о возможности построения многочлена, реализующего сравнение чисел над кольцом вычетов Z_P (т.е. в RNS). Отметим, что P в данном случае обозначает диапазон, обеспечиваемый n -модульной RNS.

Теорема 3.2.1. *Если P – составное число, то не существует многочлена $f(X, Y) \in Z_P[x, y]$ такого, что*

$$f(X, Y) = \begin{cases} 2, & \text{если } X > Y, \\ 1, & \text{если } X = Y, \\ 0, & \text{если } X < Y. \end{cases} \quad (3.24)$$

где $X, Y \in Z_P$

Доказательство. Предположим, что такой многочлен существует, тогда его можно представить в виде

$$f(X, Y) = c + g(X) + h(Y) + O(X, Y), \quad (3.25)$$

где многочлен $g(X)$ делится на X без остатка, многочлен $h(Y)$ делится на Y без остатка, многочлен $O(X, Y)$ делится на $X \cdot Y$ без остатка и c – свободный член.

С учетом того, что $X, Y \in \mathbb{Z}_P$, а $X, Y \geq 0$, рассмотрим два случая.

Случай 1. Если $X = Y = 0$, то, используя формулу (3.24), получим $f(0, 0) = 1$. С другой стороны, используя формулу (3.25), получим значение $f(0, 0) = c$. Следовательно, $c = 1$.

Случай 2. Если $X > 0, Y = 0$, то, используя формулу (3.24), получим $f(X, 0) = 2$. С другой стороны, используя формулу (3.25), получим значение $f(X, 0) = c + g(X)$. Следовательно, $g(X) = 1$ для любого $X > 0$. Над кольцом вычетов \mathbb{Z}_P многочлены, делящиеся на X без остатка и удовлетворяющие условию $g(X) = 1$ для всех $X \neq 0$, существуют при условии, что $\gcd(X, P) = 1$. Например, $g(X) = X^{\phi(P)} \equiv 1 \pmod{P}$. Рассмотрим случай, когда $\gcd(X, P) = d$ и $d > 1$, тогда учитывая что $g(X)$ делится на X без остатка, получим $g(X) \pmod{d} \equiv 0$, следовательно,

$$f(X, 0) = c + g(X) \equiv c = 1 \pmod{d}. \quad (3.26)$$

Так как для любого $d > 1$ $2 \not\equiv 1 \pmod{d}$, следовательно, пришли к противоречию, и не существует многочлена от двух переменных, заданного над кольцом вычетов \mathbb{Z}_P , реализующего функцию сравнения чисел, если P – составное число.

Теорема доказана. □

Лемма 3.2.1. *Если P – простое число, то существует многочлен $f(X, Y) \in \mathbb{Z}_P[x, y]$, удовлетворяющий (3.24).*

Доказательство. P – простое число, то \mathbb{Z}_P – поле. Следовательно, $f(X, Y) \in \mathbb{Z}_P[x, y]$ может быть вычислена с помощью интерполяционной функции Лагранжа от двух переменных.

Следствие доказано. □

Таким образом, из Теоремы 3.2.1 можно сделать вывод о том, что функцию сравнения чисел в RNS нельзя представить в виде многочлена. Следовательно, реализация алгоритма сравнения чисел в RNS состоит из двух этапов. Первый этап – вычисление позиционной характеристики (ПХ) модулярных чисел $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$. Второй этап – сравнение

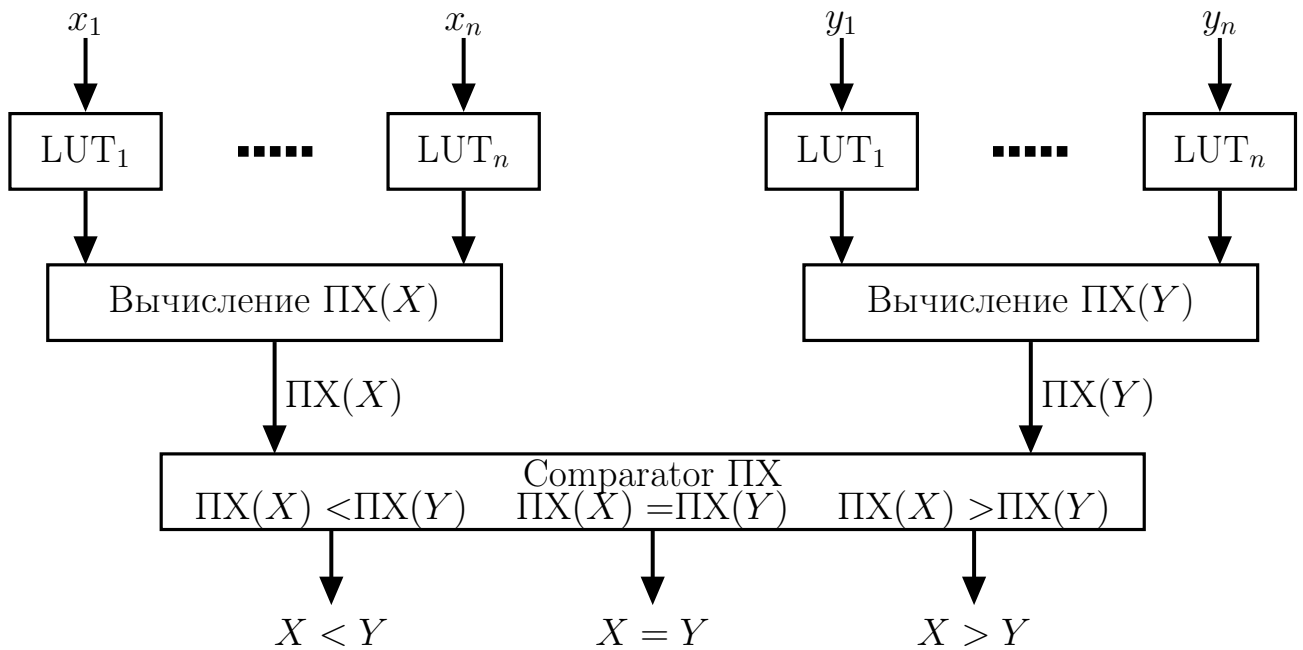


Рисунок 3.1 — Общая схема сравнения чисел с использованием позиционной характеристики

позиционных характеристик $PX(X)$ и $PX(Y)$ модулярных чисел в позиционной системе счисления (ПСС). Вышесказанное позволяет сформулировать научную проблему уменьшения вычислительной сложности алгоритма сравнения чисел в RNS, решение которой позволит расширить область применимости RNS при реализации практических приложений.

В качестве PX модулярного числа может выступать его представление в ПСС. Для перевода числа из RNS в ПСС можно использовать один из алгоритмов: Китайскую теорему об остатках (КТО), переход к обобщенной позиционной системе счисления [160], рекурсивный алгоритм сдваивания чисел (nCRT) [364] и их модификации.

Большая вычислительная сложность алгоритмов перехода от представления числа в RNS к представлению в двоичной системе счисления обусловила возникновение многочисленных исследований по аппроксимации двоичного представления. С целью уменьшения вычислительной сложности операции сравнения чисел в RNS исследователи предложили в качестве PX модулярного числа использовать следующие функции: диагональная функция (Dimauro и др., 1993) [195], функция ядра (Burgess, 2003) [170], фактор-функция (Dimauro и др., 2003) [326], монотонная функция Pirlo (Pirlo и Impedovo, 2013) [312], модифицированная диагональная функция (Babenko и др., 2020) [26] и др. Пред-

лагаемые алгоритмы вычисления ПХ позволяют уменьшить вычислительную сложность за счет уменьшения размерности операндов при выполнении операции деления с остатком.

Самыми эффективными являются подходы из работ [7, 24, 26], основанные на приближенном методе [28], позволяющем заменить операцию деления с остатком на операцию взятия старших бит числа. В работе [24] предложена оптимизация приближенного метода для реализации операции сравнения чисел, заключающаяся в уменьшении количества операций деления с остатком и улучшении точности вычислений для обеспечения корректной работы алгоритма.

3.3 Методы сравнения чисел, основанные на переводе чисел из RNS в двоичную систему счисления

В большинстве методов задача сравнения чисел решается посредством перевода числа из RNS в ПСС и их последующего сравнения.

3.3.1 Китайская теорема об остатках

Согласно (Omondi и Premkumar, 2007) [297], для перевода числа $X = (x_1, x_2, \dots, x_n)$ из RNS в ПСС можно использовать стандартное восстановление с помощью КТО, описываемое следующей формулой

$$X = \left| \sum_{i=1}^n \cdot P_i \cdot x_i \cdot \left| P_i^{-1} \right|_{p_i} \right|_P, \quad (3.27)$$

где $P_i = \frac{P}{p_i}$ и $\left| P_i^{-1} \right|_{p_i}$ – мультипликативная инверсия P_i по модулю p_i .

Рассмотрим примеры восстановления числа по формуле (3.27) с последующим сравнением чисел.

Пример 3.3.1. Пусть RNS задана набором модулей $(3, 5, 7)$, $X = (2, 2, 3)$, $Y = (1, 3, 4)$ – числа, представленные в данной RNS. Динамический диапазон данной системы остаточных классов равен $P = 3 \cdot 5 \cdot 7 = 105$.

Вычислим P_i

$$P_1 = \frac{P}{p_1} = \frac{105}{3} = 35, P_2 = \frac{P}{p_2} = \frac{105}{5} = 21 \text{ и } P_3 = \frac{P}{p_3} = \frac{105}{7} = 15.$$

Чтобы вычислить мультипликативную инверсию P_i , нужно найти число x , удовлетворяющее сравнению $x \cdot P_i \equiv 1 \pmod{p_i}$. Таким образом, $|P_1^{-1}|_3 = 2$, $|P_2^{-1}|_5 = 1$ и $|P_3^{-1}|_7 = 1$. Все константы, необходимые для вычисления позиционного представления согласно (3.27), получены. Вычислим значение первого числа

$$X = |35 \cdot 2 \cdot 2 + 21 \cdot 2 \cdot 1 + 15 \cdot 3 \cdot 1|_{105} = |227|_{105} = 17.$$

Вычислим значение второго числа

$$Y = |35 \cdot 1 \cdot 2 + 21 \cdot 3 \cdot 1 + 15 \cdot 4 \cdot 1|_{105} = |193|_{105} = 88.$$

Так как $17 < 88$, $X < Y$.

Учитывая вычислительную сложность вычисления остатка от деления на большое число P , исследователи предложили альтернативный подход, основанный на переходе к представлению в обобщенной позиционной системе счисления.

3.3.2 Обобщенная позиционная система счисления

Обобщенная позиционная система счисления (ОПСС) за счет своих свойств позволяет сравнивать числа без их прямого восстановления. Число в ОПСС задается кортежем $[a_1, a_2, \dots, a_n]$, а основаниями системы являются $p_1, p_1 \cdot p_2, p_1 \cdot p_2 \cdot p_3, \dots, p_1 \cdot p_2 \cdot \dots \cdot p_{n-1}$, где p_1, p_2, \dots, p_n — модули RNS. Связь между двоичной системой счисления и ОПСС определяется следующей формулой

$$X = a_1 + a_2 \cdot p_1 + a_3 \cdot p_1 \cdot p_2 + \dots + a_n \cdot p_1 \cdot p_2 \cdot \dots \cdot p_{n-1}.$$

Так как ОПСС является позиционной системой счисления, сравнение чисел равносильно поразрядному сравнению двух кортежей $[a_1, a_2, \dots, a_n]$ и $[b_1, b_2, \dots, b_n]$, начиная со старшего разряда. Для перевода числа $X = (x_1, x_2, \dots, x_n)$ из RNS в $[a_1, a_2, \dots, a_n]$ ОПСС используется схема Гар-

нера

$$a_1 = x_1,$$

$$a_2 = \left| (x_2 - a_1) \cdot p_1^{-1} \right|_{p_2},$$

.....,

$$a_n = \left| (x_n - a_1 - a_2 \cdot p_1 - \dots - a_{n-1} \cdot p_1 \cdot p_2 \cdot \dots \cdot p_{n-2}) \cdot p_1^{-1} \cdot p_2^{-1} \cdot \dots \cdot p_{n-1}^{-1} \right|_{p_n}.$$

Эффективная реализация алгоритма сравнения чисел с использованием ОПСС представлена в работе [254].

Использование ОПСС позволяет уйти от вычисления остатка от деления на большое число P , но порождает большое количество операций по модулям RNS.

3.3.3 Приближенный метод

Для исключения операции деления с остатком на большое число в работе [358] был предложен приближенный метод, основанный на отображении, переводящем динамический диапазон $[0, P)$ в диапазон $[0, 2)$. Перепишем (3.27) в виде

$$X = \sum_{i=1}^n P_i \cdot x_i \cdot |P_i^{-1}|_{p_i} - P \cdot r_X, \quad (3.28)$$

для некоторого неотрицательного целого числа X , где r_X – ранга числа X . Разделив (3.28) на $\frac{P}{2}$, получим

$$X_s = \left(\frac{2}{P} \right) \cdot X = \sum_{i=1}^n \frac{2}{p_i} \cdot x_i \cdot |P_i^{-1}|_{p_i} - 2 \cdot r_X. \quad (3.29)$$

Таким образом, согласно (3.29), X_s может быть вычислен как сумма дробных чисел с отбрасыванием кратной двум целой части результата. Заметим, что при таком подходе вычисления в двоичном виде могут быть реализованы довольно тривиально. Проиллюстрируем рассмотренный метод на примере.

Пример 3.3.2. Пусть в RNS с модулями $(3, 5, 7)$ задано число $X = (2, 2, 3)$, тогда по формуле (3.29) получим

$$X_s = \left| \frac{2}{3} \cdot 2 \cdot 2 + \frac{2}{5} \cdot 2 \cdot 1 + \frac{2}{7} \cdot 3 \cdot 1 \right|_2 = \left| 4 \frac{34}{105} \right|_2 = \frac{34}{105}.$$

Заметим, что в данном методе слагаемые редко могут быть представлены в виде конечной дроби. Для представления в виде десятичной (двоичной) дроби каждое слагаемое должно быть определенным образом округлено.

Если на каждое слагаемое суммы в формуле (3.29) выделить $N + 1$ бит, 1 – на целую часть и N – на дробную, и усекать оставшиеся биты, то ошибка в каждом слагаемом будет удовлетворять неравенству $0 \leq e_i < 2^{-N}$. Так как слагаемых n , то максимальная ошибка при усечении (3.29) будет $e = n \cdot 2^{-N}$.

Поскольку числа X_s распределены равномерно на интервале $[0, 2)$, то расстояние между двумя соседними числами равно $\frac{2}{P}$.

Обратим внимание, что если $X_s \in [0, 1)$, то $X \geq 0$, иначе $X < 0$.

Таким образом, для того, чтобы усеченное значение X_s соотносилось с точным значением X_s , ошибка должна удовлетворять следующим соотношениям

$$n \cdot 2^{-N} \leq \frac{2}{P}, \text{ если } |P|_2 = 0, \quad (3.30)$$

$$n \cdot 2^{-N} \leq \frac{1}{P}, \text{ иначе,} \quad (3.31)$$

или

$$N \geq \lceil \log_2 P \cdot n \rceil - 1, \text{ если } |P|_2 = 0,$$

$$N \geq \lceil \log_2 P \cdot n \rceil, \text{ иначе.}$$

Несмотря на то, что дробное представление требует примерно $\lceil \log_2 n \rceil$ бит дополнительной памяти, простота и скорость выполнения операций компенсируют эту избыточность. Данный способ вычисления X_s может быть относительно просто реализован с использованием памяти, хранящей предвычисленные значения: на вход подается остаток $|X|_{p_i}$, а на выход поступает умноженное на константы усеченное слагаемое, далее усеченные значения складываются по модулю 2, что легко реализуется аппаратно.

Рассмотрим численный пример.

Пример 3.3.3. Пусть в RNS с модулями $p_1 = 3$, $p_2 = 5$, $p_3 = 7$ заданы два числа $1 = (1, 1, 1)$ и $104 = (2, 4, 6)$. Для данной системы $N \geq \lceil \log_2(105 \cdot 3) \rceil = 9$.

Рассмотрим первое число по слагаемым

$$\begin{aligned} \left| \frac{2}{3} \cdot 1 \cdot 2 \right|_2 &= \frac{4}{3} = 1.010101010101011 \dots \approx 1.010101011, \\ \left| \frac{2}{5} \cdot 1 \cdot 1 \right|_2 &= \frac{2}{5} = 0.011001100110011 \dots \approx 0.011001101, \\ \left| \frac{2}{7} \cdot 1 \cdot 1 \right|_2 &= \frac{2}{7} = 0.010010010010010 \dots \approx 0.010010010. \end{aligned}$$

Просуммируем по модулю 2 слагаемые и получим 0.000001010. Рассмотрим второе число

$$\begin{aligned} \left| \frac{2}{3} \cdot 2 \cdot 2 \right|_2 &= \left| \frac{8}{3} \right|_2 = \frac{2}{3} = 0.101010101010101 \dots \approx 0.101010101, \\ \left| \frac{2}{5} \cdot 1 \cdot 4 \right|_2 &= \frac{8}{5} = 1.100110011001101 \dots \approx 1.100110011, \\ \left| \frac{2}{7} \cdot 1 \cdot 6 \right|_2 &= \frac{12}{7} = 1.101101101101110 \dots \approx 1.101101110. \end{aligned}$$

Просуммируем по модулю 2 слагаемые и получим 1.111110110. Сравнивая полученные значения, увидим что $(1, 1, 1) < (2, 4, 6)$.

Данный метод эффективнее, чем восстановление числа с помощью классической Китайской теоремы об остатках, однако, возникает вопрос о достаточности или избыточности точности согласно формулам (3.30)-(3.31).

Стоит заметить, что использование данного подхода позволяет вычислять позиционную характеристику по следующей формуле

$$V(X) = \left| \sum_{i=1}^n \left[\frac{2}{p_i} \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right]_{2^{-N}} \right|_2, \quad (3.32)$$

где $[x]_{2^{-N}} = [2^N x] / 2^N$.

С целью уменьшения количества операций в приближенном методе в работе (Chervyakov и др., 2017) [28] было предложено использовать следующую формулу

$$C(X) = \left| \sum_{i=1}^n W_i \cdot x_i \right|_1, \quad (3.33)$$

где $W_i = \left[\frac{2^N |P_i^{-1}|_{p_i}}{p_i} \right] / 2^N$, $|x|_1$ – дробная часть числа x , $N = \lceil \log_2(P\rho) \rceil$ и $\rho = -n + \sum_{i=1}^n p_i$.

Преимущество данного метода состоит в том, что он не требует дополнительных операции округления вверх, однако, размеры операндов при этом увеличиваются.

3.4 Методы сравнения чисел в RNS с использованием диагональной функции

С целью уменьшения вычислительной сложности алгоритма сравнения чисел в RNS в работе (Dimauro и др., 1993) [195] было предложено использовать монотонную диагональную функцию.

Метод на основе специальной диагональной функции отличается от вышеизложенных методов сравнения чисел и использует сумму соответствующих коэффициентов P_i (Sum of Quotients Technique – SQT). Описание данной функции можно найти, например, в работе (Dimauro и др., 1993) [195]. Диагональная функция представляет собой монотонно возрастающую функцию, на основе которой реализуется сравнение чисел.

Диагональная функция имеет вид

$$D(X) = \left\lfloor \frac{X}{p_1} \right\rfloor + \left\lfloor \frac{X}{p_2} \right\rfloor + \dots + \left\lfloor \frac{X}{p_n} \right\rfloor. \quad (3.34)$$

Формула (3.34) мало пригодна для использования в практических приложениях. В связи с этим в работе (Dimauro и др., 1993) [195] была предложена аналитическая функция для вычисления диагональной функции

$$D(X) = \left\lfloor \sum_{i=1}^n k_i^* \cdot x_i \right\rfloor_{SQ}, \quad (3.35)$$

где $k_i^* = \lfloor -p_i^{-1} \rfloor_{SQ}$, $i = 1, \dots, n$, $SQ = P_1 + P_2 + \dots + P_n$.

Так как диагональная функция (3.35) является монотонно возрастающей, то она может быть использована для сравнения чисел, т.е. если $D(X) < D(Y)$, то $X < Y$. Однако, возможны случаи, когда $D(X) = D(Y)$ при $X < Y$. В таких случаях $x_i < y_i$, $i = 1, \dots, n$. Схема сравнения чисел с использованием диагональной функции представлена на рисунке 3.2. Рассмотрим пример сравнения чисел на основе диагональной функции.

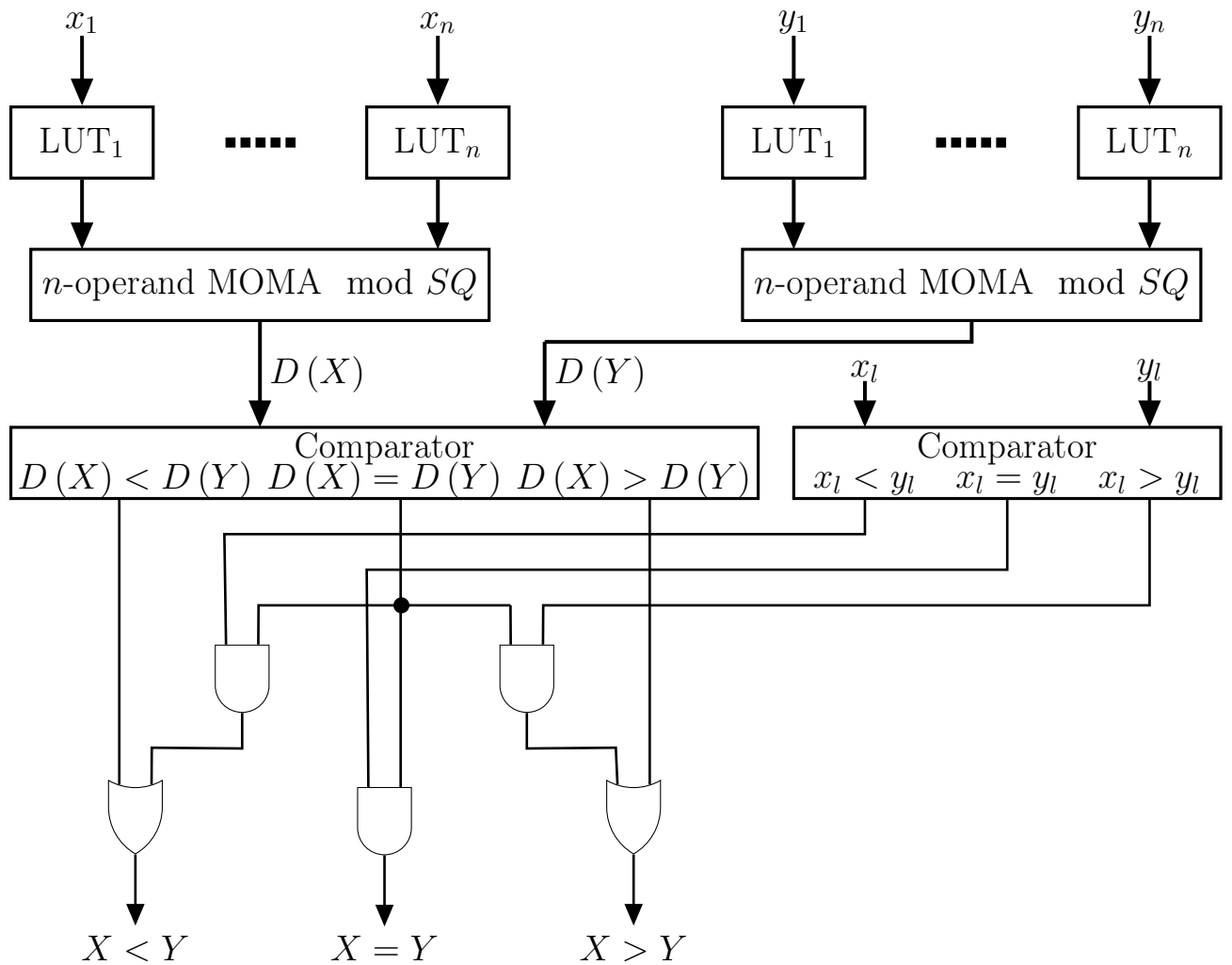


Рисунок 3.2 — Аппаратная реализация алгоритма сравнения чисел в RNS на основе диагональной функции

Пример 3.4.1. Возьмем ранее использовавшиеся числа $X = (2, 2, 3)$ и $Y = (1, 3, 4)$, представленные в RNS с основаниями $(3, 5, 7)$. Для начала вычислим значения констант

$$\begin{aligned}
 SQ &= 35 + 21 + 15 = 71, \\
 k_1^* &= |-p_1^{-1}|_{71} = |-3^{-1}|_{71} = 47, \\
 k_2^* &= |-p_2^{-1}|_{71} = |-5^{-1}|_{71} = 14, \\
 k_3^* &= |-p_3^{-1}|_{71} = |-7^{-1}|_{71} = 10.
 \end{aligned}$$

Найдем значение диагональной функции

$$\begin{aligned}
 D(X) &= |2 \cdot 47 + 2 \cdot 14 + 3 \cdot 10|_{71} = 10, \\
 D(Y) &= |1 \cdot 47 + 2 \cdot 14 + 3 \cdot 10|_{71} = 34.
 \end{aligned}$$

Т.к. $D(X) < D(Y)$, то $X < Y$.

Исследуем свойства диагональной функции. Определим верхнюю и нижнюю границы для значения SQ относительно P .

Лемма 3.4.1 (нижняя граница). $SQ > n \cdot P^{\frac{n-1}{n}}$.

Доказательство. Используя неравенство Коши, согласно которому среднее арифметическое нескольких неотрицательных чисел не меньше их среднего геометрического, и учитывая, что равенство в неравенстве Коши достигается только при равенстве чисел, а в RNS $\forall i \neq j: p_i \neq p_j$, получим строгое неравенство

$$\sum_{i=1}^n \frac{1}{p_i} > n \cdot \sqrt[n]{\prod_{i=1}^n \frac{1}{p_i}}. \quad (3.36)$$

Следовательно,

$$SQ = P \cdot \sum_{i=1}^n \frac{1}{p_i} > n \cdot P \cdot \sqrt[n]{\frac{1}{P}} = n \cdot P^{\frac{n-1}{n}}. \quad (3.37)$$

Лемма доказана. \square

Лемма 3.4.2 (верхняя граница). $SQ \leq P \cdot \sum_{i=1}^n \frac{1}{p_i}$.

Доказательство. Так как $SQ = \sum_{i=1}^n P_i = P \sum_{i=1}^n \frac{1}{p_i} \leq P \cdot \sum_{i=1}^n \frac{1}{p_i}$.

Лемма доказана. \square

Из Леммы 3.4.2 следует, что наилучший результат для диагональной функции возникает, если в качестве модулей берутся первые n простых чисел ряда. Таким образом, полученная оценка является точной верхней гранью.

Учитывая, что неравенство Коши обращается в равенство только в том случае, если числа равны между собой, можно сделать вывод, что наилучший результат будет получаться при $p_1 < p_2 < \dots < p_n = 2^{1-\epsilon} \cdot p_1$, где ϵ стремится к 1 снизу, но никогда ее не достигает. Рассмотрим пример двухмодульной RNS $\{p_1, p_2 = p_1 + 2\}$, где p_1 – нечетное число. Покажем, что при $p_1 \rightarrow \infty$ значение SQ приближается к нижней границе сверху, для этого вычислим значение $\frac{SQ}{n \cdot P^{\frac{n-1}{n}}}$, получим $\frac{SQ}{n \cdot P^{\frac{n-1}{n}}} = \frac{p_1 + p_2}{2 \cdot \sqrt{p_1 \cdot p_2}} = \frac{2 \cdot p_1 + 2}{2 \cdot \sqrt{p_1^2 + 2 \cdot p_1}} = \frac{p_1 + 1}{\sqrt{p_1^2 + 2 \cdot p_1}}$. Вычислим предел $\lim_{p_1 \rightarrow \infty} \frac{SQ}{n \cdot P^{\frac{n-1}{n}}}$, получим, $\lim_{p_1 \rightarrow \infty} \frac{SQ}{n \cdot P^{\frac{n-1}{n}}} = 1$. Так как $p_1 \neq p_2$ и $2 \leq p_1 < p_2$, то $p_1 + p_2 - 2 \cdot \sqrt{p_1 \cdot p_2} = (\sqrt{p_1} - \sqrt{p_2})^2 > 0$, следовательно, $\frac{p_1 + p_2}{2 \cdot \sqrt{p_1 \cdot p_2}} > 1$. Таким образом, нижняя граница в Лемме 3.4.1 является точной нижней границей.

Исследуем асимптотическое поведение ϵ в зависимости от p_1 , получим $1 - \epsilon = \log_2 \lim_{p_1 \rightarrow \infty} \frac{p_2}{p_1} = \log_2 \lim_{p_1 \rightarrow \infty} \frac{p_1+2}{p_1} = 0$. Учитывая, что $\forall p_1 \geq 2: \frac{p_1+2}{p_1} > 1$, следовательно, $1 - \epsilon$ стремится к нулю сверху, значит чем больше значение p_1 , тем ближе ϵ к единице, но при этом всегда меньше ее.

3.5 Функция ядра Акушского и ее свойства

Поиск позиционных характеристик, которые позволили бы уменьшить вычислительную сложность алгоритма сравнения чисел в RNS привели исследователей Акушского И.Я., Бурцева В.М. и Пака И.Т. к построению новой конструкции, названной функцией ядра Акушского. Задается функция ядра Акушского следующей формулой

$$C(X) = \sum_{i=1}^n \bar{w}_i \cdot \left\lfloor \frac{X}{p_i} \right\rfloor \quad (3.38)$$

или

$$C(X) = X \cdot \sum_{i=1}^n \frac{\bar{w}_i}{p_i} - \sum_{i=1}^n \frac{x_i \cdot \bar{w}_i}{p_i}, \quad (3.39)$$

где целые числа \bar{w}_i – постоянные определяемые выбором точки интерполяции. Константы \bar{w}_i задают вес каждого из частных $\left\lfloor \frac{X}{p_i} \right\rfloor$ в формуле (3.39), тем самым задавая функцию ядра и придавая ей различные свойства.

Числа \bar{w}_i в формуле (3.38) могут быть в определенном смысле произвольными. Именно они определяют каждую конкретную функцию ядра и могут меняться в зависимости от решаемой задачи.

Базовым свойством функции ядра является то, что ее максимальный диапазон может меняться и может быть значительно меньше числа P в зависимости от выбора весов. Например, в качестве C_P можно использовать некоторое произвольное значение $C(P)$, обладающее необходимыми для решения конкретной задачи свойствами. Это значение называется диапазоном функции ядра и определяется выражением

$$C(P) = \sum_{i=1}^n \bar{w}_i \cdot P_i. \quad (3.40)$$

Учитывая, что $P_j \equiv 0 \pmod{p_i}$ для любого $i \neq j$, константы \bar{w}_i этой функции могут быть определены из соотношения

$$\bar{w}_i \equiv C(P) \cdot P_i^{-1} \pmod{p_i}. \quad (3.41)$$

При этом необходимо учитывать, что (3.41) задает некий класс вычетов для каждого i , и числа \bar{w}_i могут оказаться отрицательными или положительными в каждом конкретном случае.

Рассмотрим так же еще одно важное выражение, связывающее диапазон RNS с диапазоном функции ядра. Так как число P в RNS представляется в форме $(0, 0, \dots, 0)$, то для него (3.38) примет вид

$$C(P) = C_P = P \cdot \sum_{i=1}^n \frac{\bar{w}_i}{p_i}. \quad (3.42)$$

Стоит отметить, что (3.38) плохо подходит для вычисления значений функции ядра в практических приложениях. Используя КТО, можно изменить способ вычисления значения функции ядра величины X на основе ее остатков.

Свойство 3.5.1. *Значение функции ядра $C(X)$, заданной весами $\bar{w}_1, \bar{w}_2, \dots, \bar{w}_n$, при условии $0 \leq C(X) < C_P$, $X \in [0, P)$, можно вычислить с использованием формулы*

$$C(X) \equiv \left| \sum_{i=1}^n c_i \cdot x_i \right|_{C_P}, \quad (3.43)$$

где $c_i = C(B_i)$ и $B_i = P_i \cdot |P_i^{-1}|_{p_i}$.

Доказательство. Доказательство Свойства 3.5.1 основано на следствиях Китайской теоремы об остатках, свойствах чисел B_i и использовании формулы (3.39).

Свойство доказано. □

Отметим, что вычисление функции ядра по этой формуле возможно только при отсутствии критических ядер: значений $C(X)$, не попадающих в диапазон $[0, C_P)$ при $X \in [0, P)$. При наличии критических ядер функция ядра не может быть вычислена точно с использованием (3.43). Существует два подхода к решению проблемы критических ядер: уточнять функцию ядра, полученную по формуле (3.43) или строить функцию ядра, заведомо не имеющую критических ядер. Сосредоточимся на разработке способов построения функции ядра

без критических ядер для эффективной реализации различных немодульных операций в RNS.

Числа $c_i = C(B_i)$ характеризуются следующим важным свойством.

Свойство 3.5.2. Для произвольной функции ядра $C(X)$, заданной весами $\bar{w}_1, \bar{w}_2, \dots, \bar{w}_n$,

$$\left| \sum_{i=1}^n c_i \right|_{C_P} = 0.$$

Доказательство. Вычислим значение функции ядра $C(1)$, используя формулу (3.38), получим

$$C(1) = \sum_{i=1}^n \bar{w}_i \cdot \left[\frac{1}{p_i} \right] = 0. \quad (3.44)$$

С другой стороны, согласно Свойству 3.5.1, значение $C(1)$ можно вычислить по формуле (3.43)

$$C(1) = \left| \sum_{i=1}^n c_i \right|_{C_P}. \quad (3.45)$$

Так как левые части равенств (3.44) и (3.45) равны, то равны и правые части, то есть $\left| \sum_{i=1}^n c_i \right|_{C_P} = 0$.

Свойство доказано. \square

Свойство 3.5.2 оказывается полезным при доказательстве некоторых утверждений, приведенных в данной работе далее. Это же касается Свойства 3.5.3, связывающего функцию ядра от суммы двух чисел с суммой функций ядра этих чисел.

Свойство 3.5.3. Для произвольной функции ядра $C(X)$, заданной весами $\bar{w}_1, \bar{w}_2, \dots, \bar{w}_n$, и двух чисел $X \geq 0$ и $Y \geq 0$, таких что $X + Y < P$, выполняется

$$C(X + Y) = C(X) + C(Y) + \sum_{i=1}^n \bar{w}_i \cdot \left[\frac{x_i + y_i}{p_i} \right],$$

где $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$, $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$.

Доказательство. Учитывая, что X и Y можно представить в виде $X = x_i + p_i \cdot \left[\frac{X}{p_i} \right]$ и $Y = y_i + p_i \cdot \left[\frac{Y}{p_i} \right]$, вычислим значение $C(X + Y)$, используя форму-

лу (3.38), получим

$$\begin{aligned}
C(X + Y) &= \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{X + Y}{p_i} \right\rfloor \\
&= \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{x_i + p_i \cdot \left\lfloor \frac{X}{p_i} \right\rfloor + y_i + p_i \cdot \left\lfloor \frac{Y}{p_i} \right\rfloor}{p_i} \right\rfloor \\
&= \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{x_i + y_i}{p_i} + \left\lfloor \frac{X}{p_i} \right\rfloor + \left\lfloor \frac{Y}{p_i} \right\rfloor \right\rfloor \\
&= \sum_{i=1}^n \bar{w}_i \left(\left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor + \left\lfloor \frac{X}{p_i} \right\rfloor + \left\lfloor \frac{Y}{p_i} \right\rfloor \right). \tag{3.46}
\end{aligned}$$

Так как $\lfloor \lfloor a \rfloor + \lfloor b \rfloor \rfloor = \lfloor a \rfloor + \lfloor b \rfloor$, то формула (3.46) примет вид

$$\begin{aligned}
C(X + Y) &= \sum_{i=1}^n \bar{w}_i \left(\left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor + \left\lfloor \frac{X}{p_i} \right\rfloor + \left\lfloor \frac{Y}{p_i} \right\rfloor \right) \\
&= \sum_{i=1}^n \bar{w}_i \cdot \left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor + \sum_{i=1}^n \bar{w}_i \cdot \left\lfloor \frac{X}{p_i} \right\rfloor + \sum_{i=1}^n \bar{w}_i \cdot \left\lfloor \frac{Y}{p_i} \right\rfloor \\
&= \sum_{i=1}^n \bar{w}_i \cdot \left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor + C(X) + C(Y). \tag{3.47}
\end{aligned}$$

Свойство доказано. □

Заметим, что, согласно Свойству 3.5.3, величина функции ядра суммы двух чисел отличается от суммы функций ядра этих чисел на сумму весов по тем остаткам, для которых сумма остатков превосходит величину модуля. Иными словами, если $x_i + y_i \geq p_i$, то частное в формуле выше будет равно 1 (это максимальное значение, так как $x_i < p_i$ и $y_i < p_i$), и к сумме $C(X) + C(Y)$ добавляется величина \bar{w}_i . Данное свойство позволяет анализировать поведение функции ядра и оценивать ее прирост в зависимости от значений X и Y .

Главная цель использования функции ядра – сократить диапазон, в котором вычисляется остаток от деления при реализации методов, аналогичных (3.27), или сделать его более удобным для вычислений. Использование КТО сопряжено с вычислениями по большому модулю, равному P , требующими больших затрат на выполнение данной операции. С другой стороны, для выполнения немодульных операций не обязательно восстанавливать число в позиционной системе счисления, достаточно сделать вывод о его расположении на числовой прямой.

3.6 Сравнение чисел с помощью функции ядра Акушского

Функция ядра является универсальной позиционной характеристикой. Однако, ее практическое использование затруднено в связи с необходимостью решения двух важных проблем. Первая проблема связана с точным вычислением значений функции ядра. Так как по определению веса \bar{w}_i функции ядра могут быть отрицательными, то и значения этой функции потенциально могут лежать вне диапазона $[0, C_P)$. Проблема выхода функции ядра за пределы этого полуинтервала называется проблемой критических ядер. Она существенно ограничивает применимость каждой конкретной функции ядра, так как требует уточнения ее значений. С другой стороны, алгоритмы сравнения чисел на основе функции ядра требуют решения проблемы выбора монотонных функций ядра. Данный раздел посвящен анализу путей решения этих проблем для функций ядра общего вида.

3.6.1 Проблема монотонности функции ядра

Монотонная функция ядра Акушского может быть использована для реализации операции сравнения чисел. Исследуем условия монотонности функции ядра Акушского. Для этого докажем следующую теорему.

Теорема 3.6.1. *Функция ядра Акушского монотонно возрастает тогда и только тогда, когда все ее коэффициенты \bar{w}_i неотрицательны.*

Доказательство. Вычислим значение $C(X - 1)$, используя формулу (3.39), получим

$$\begin{aligned} C(X - 1) &= (X - 1) \sum_{i=1}^n \frac{\bar{w}_i}{p_i} - \sum_{i=1}^n \frac{|x_i - 1|_{p_i} \cdot \bar{w}_i}{p_i} \\ &= X \cdot \sum_{i=1}^n \frac{\bar{w}_i}{p_i} - \sum_{i=1}^n \frac{\bar{w}_i}{p_i} - \sum_{i=1}^n \frac{|x_i - 1|_{p_i} \cdot \bar{w}_i}{p_i}. \end{aligned} \quad (3.48)$$

Используя формулы (3.39) и (3.48), вычислим значение $C(X) - C(X - 1)$, получим

$$\begin{aligned} C(X) - C(X - 1) &= \sum_{i=1}^n \frac{\bar{w}_i}{p_i} + \sum_{i=1}^n \frac{|x_i - 1|_{p_i} \cdot \bar{w}_i}{p_i} - \sum_{i=1}^n \frac{x_i \cdot \bar{w}_i}{p_i} \\ &= \sum_{i=1}^n \left(1 + |x_i - 1|_{p_i} - x_i\right) \cdot \frac{\bar{w}_i}{p_i}. \end{aligned} \quad (3.49)$$

Вычислим значение выражения $1 + |x_i - 1|_{p_i} - x_i$ в зависимости от x_i и p_i , получим

$$1 + |x_i - 1|_{p_i} - x_i = \begin{cases} p_i, & \text{если } x_i = 0, \\ 0, & \text{если } x_i \neq 0. \end{cases} \quad (3.50)$$

Подставляя (3.50) в (3.49), получим

$$C(X) - C(X - 1) = \sum_{i=1, x_i=0}^n \bar{w}_i. \quad (3.51)$$

Для доказательства теоремы воспользуемся принципом от противного. Предположим, что существует монотонно возрастающая функция ядра Акушского с отрицательными коэффициентами \bar{w}_i . Пусть отрицательных коэффициентов $s > 0$ штук, и они имеют соответственно индексы $\{i_1, i_2, \dots, i_s\}$. Рассмотрим число $X = \prod_{j=1}^s p_{i_j}$. Используя формулу (3.51), вычислим значение $C(X) - C(X - 1)$, получим

$$C(X) - C(X - 1) = \sum_{j=1}^s \bar{w}_{i_j}. \quad (3.52)$$

Так как в (3.52) правая часть $\sum_{j=1}^s \bar{w}_{i_j} < 0$, то, следовательно, $C(X) < C(X - 1)$, значит функция не является монотонно возрастающей. Пришли к противоречию, следовательно, для того, чтобы функция ядра Акушского была монотонно возрастающей, необходимо и достаточно чтобы все коэффициенты \bar{w}_i были неотрицательными.

Теорема доказана. □

Обратим внимание на то, что функция ядра не может быть строго монотонной, так как изменение ее значения для соседних чисел возможно только при наличии нулевых остатков в большем из них. Таким образом, функция ядра

с неотрицательными коэффициентами является позиционной характеристикой числа в RNS.

Выражение (3.51) позволяет определить структуру монотонной функции ядра. Слагаемое \bar{w}_i в (3.51) входит в сумму только при $x_i = 0$, тем самым функция ядра делится на «уровни», на которых значение функции не изменяется. На этих уровнях при увеличении X увеличиваются все x_i , $i = 1, 2, \dots, n$. Это свойство используется в алгоритмах сравнения на основе монотонной функции ядра. На рисунке 3.3а) изображен пример функции ядра с неотрицательными коэффициентами. Функция ядра в таком случае монотонна и разделена на «уровни», оценить величину числа на которых можно на основе величины значения остатка по одному из модулей, что видно из линий уровня функции ядра на рисунке 3.3б). Функция ядра Акушского, для которой $\bar{w}_i > 0$, переходит на следующий уровень каждый раз, когда $x_i = 0$ по любому из модулей p_i , $i = 1, 2, \dots, n$. Существует несколько примеров использования функции ядра в алгоритмах сравнения. Например, широко исследуемая в литературе диагональная функция [195, 283, 326], является вариантом функции ядра с $\bar{w}_i = 1$ для всех $i = 1, 2, \dots, n$.

С другой стороны, если взять $\bar{w}_i = 0$ для всех $i = 1, 2, \dots, n - 1$ и $\bar{w}_n = 1$, получим функцию ядра с диапазоном $C_P = P_n$. Таким образом, выражение (3.53) так же является частным случаем (3.38).

По сути, результатом выполнения такой функции является частное от деления на p_n . P_n является наименьшим из P_i , $i = 1, 2, \dots, n$. С учетом того, что коэффициенты функции ядра, пригодной для сравнения чисел, должны быть неотрицательными, согласно Теореме 3.6.1, функция ядра с модулем $C_P = P_n$ является функцией ядра с минимальным возможным модулем. Так как $C_{P_n}(X)$ есть частное от деления X на p_n , то сравнение чисел с помощью данной функции сводится к сравнению их частных. В случае равенства частных, далее сравниваются остатки по модулю p_n . Можно показать, что в случае использования в качестве диапазона C_P чисел, меньших P_n , сравнение становится невозможным.

Общая схема выполнения сравнения чисел с помощью функции ядра Акушского представлена на рисунке 3.4.

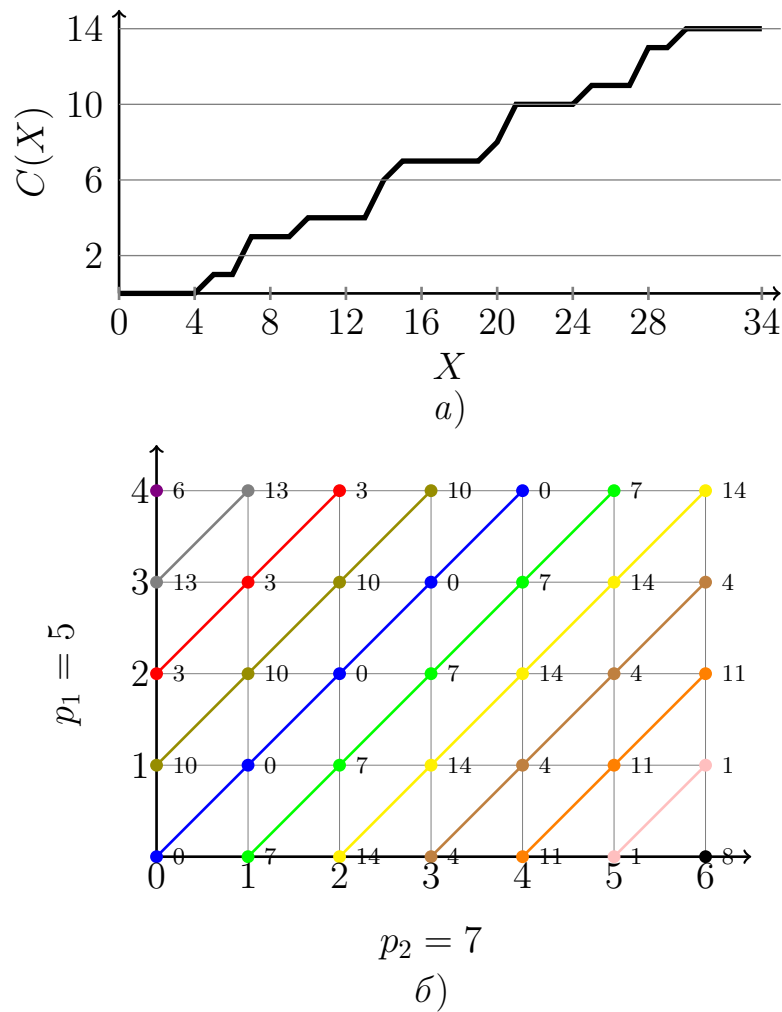


Рисунок 3.3 — График *a)* и линии уровня *б)* функции ядра с $\bar{w}_1 = 1$, $\bar{w}_2 = 2$ для RNS с основаниями $p_1 = 5$, $p_2 = 7$

3.6.2 Метод построения функций ядра Акушского, не содержащих критических ядер

Если анализировать поведение функции ядра на различных значениях чисел, то можно заметить, что в общем случае данная функция не является монотонной и не может быть использована в качестве позиционной характеристики. Это хорошо видно из графика на рисунке 3.5, на котором отражены значения функции ядра с коэффициентами $\bar{w}_1 = -3$ и $\bar{w}_2 = 5$ для RNS с основаниями $p_1 = 5$ и $p_2 = 6$. Видно, что такая функция не монотонна и, более того, может принимать отрицательные значения, что делает ее неприменимой, например, для сравнения чисел в RNS. Рисунок 3.5 также демонстрирует важную проблему – проблему критических ядер, которая наблюдается при использовании

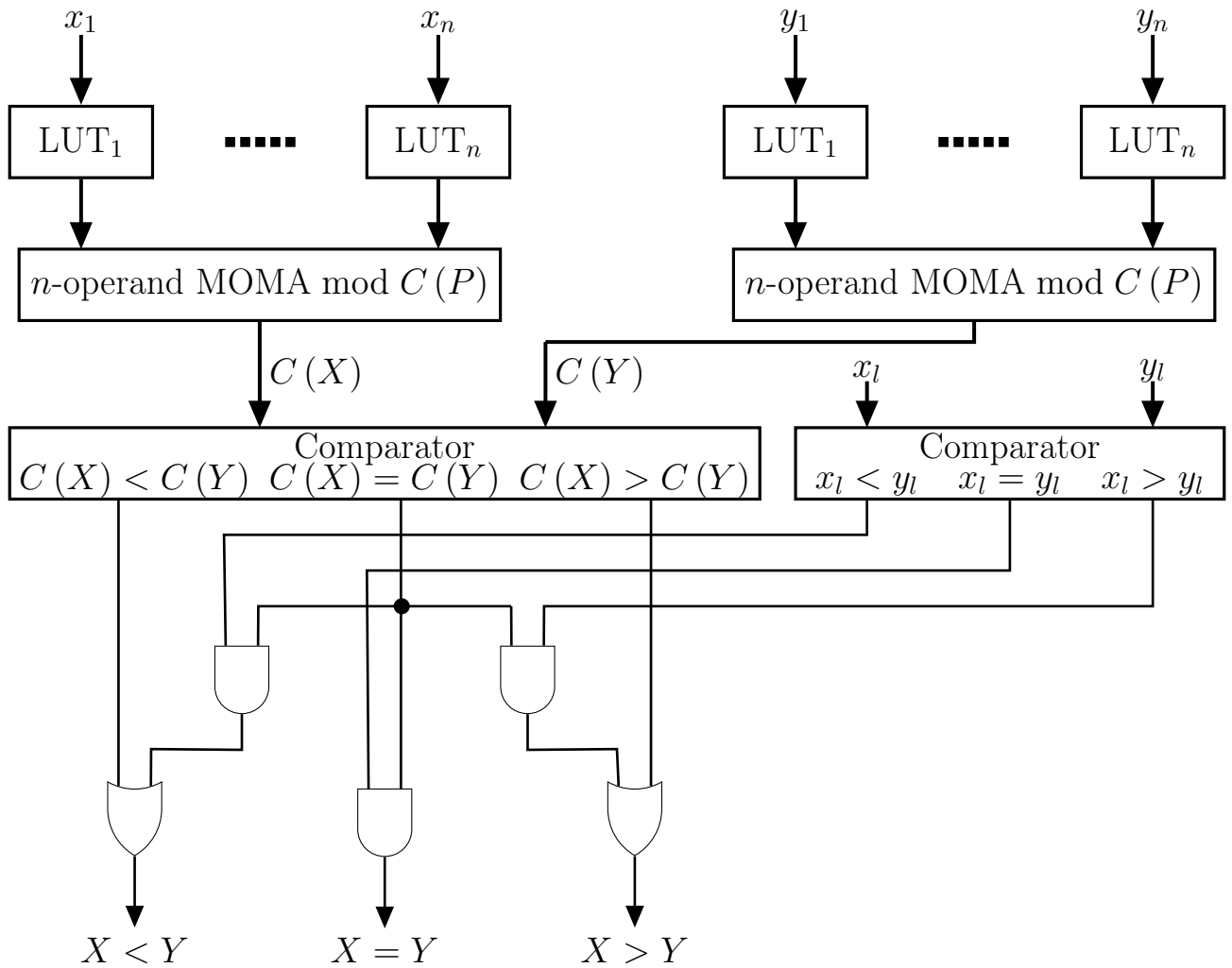


Рисунок 3.4 — Сравнение чисел в RNS с помощью монотонной функции ядра Акушского, где $\bar{w}_l > 0$

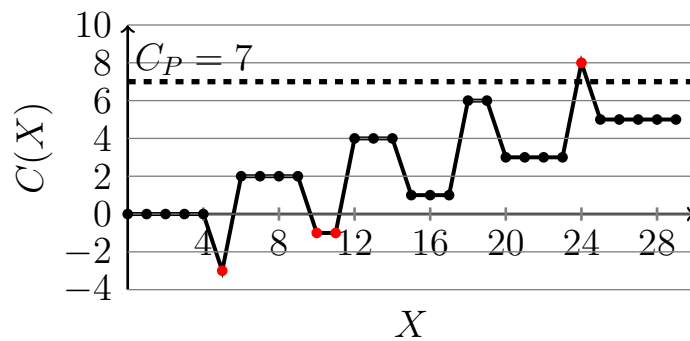


Рисунок 3.5 — График функции ядра с коэффициентами $\bar{w}_1 = -3$ и $\bar{w}_2 = 5$ для RNS с основаниями $p_1 = 5$ и $p_2 = 6$

некоторых вариантов функции ядра. В общем случае, для некоторых функций ядра $C(X)$ существуют такие X , для которых $C(X) < 0$ или $C(X) \geq C_P$. В приведенном примере, критические ядра будут в точках X , равных 5, 10, 11, в которых функция ядра будет отрицательной, и в точке X , равной 24, в которой значение функции ядра превосходит величину C_P . При этом значения функции ядра будут располагаться в некотором промежутке $[-\gamma_1, C_P + \gamma_2]$. Значения $C(X)$ для подобных X называются критическими значениями функции ядра или критическими ядрами. Критические значения делают неприменимой формулу (3.43), позволяющую эффективно вычислять значения функции ядра. Будем называть критические ядра из диапазона $[-\gamma_1, 0)$ нижними критическими ядрами, ядра из диапазона $[C_P, C_P + \gamma_2]$ – верхними критическими ядрами.

Существуют подходы, позволяющие контролировать критические ядра и исправлять значения функции ядра в них, однако, на практике это оказывается неэффективным. Более актуальным вопросом является построение функций ядра, не содержащих критических ядер. Сформулируем условия, которые позволят определить имеет ли функция ядра критические ядра, исходя из значений \bar{w}_i , $i = 1, 2, \dots, n$ для RNS с модулями p_1, p_2, \dots, p_n , упорядоченными по возрастанию.

Теорема 3.6.2. *Для того, чтобы функция ядра $C(X)$, определяемая коэффициентами $\bar{w}_1, \bar{w}_2, \dots, \bar{w}_n$, не содержала критических ядер, необходимо выполнение следующих условий для всех $k = 1, 2, \dots, n$:*

1. $\sum_{i=1}^n \bar{w}_i > 0$.
2. *Отсутствие нижних критических ядер:* $C(p_k) = \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{p_k}{p_i} \right\rfloor \geq 0$.
3. *Отсутствие верхних критических ядер:* $\sum_{i=1}^n \left(\left\lfloor \frac{p_k}{p_i} \right\rfloor + 1 \right) \cdot \bar{w}_i - \bar{w}_k > 0$.

Доказательство. Начнем с условия отсутствия нижних критических ядер.

Основываясь на формуле (3.51), заметим, что значение функции ядра изменяется только при возникновении нулей в коде числа в RNS. Нули в записи числа X в RNS возникают последовательно при возрастании его значения. Первый такой случай будет при нуле в остатке по модулю p_1 . Это означает, что для того, чтобы не было нижнего критического ядра нужно, как минимум, чтобы $\bar{w}_1 \geq 0$. При соблюдении этого условия, следующее изменение произойдет лишь при нуле в остатке по модулю p_2 . Возможное изменение функции ядра при этом будет равно $\left(\left\lfloor \frac{p_2}{p_1} \right\rfloor + 1 \right) \cdot \bar{w}_1 + \bar{w}_2$. Первое слагаемое отражает накопленное количество значений \bar{w}_1 до наступления условия $|X|_{p_2} = 0$. Для того, чтобы отсут-

ствовало нижнее критическое ядро, необходимо, чтобы $\left(\left\lfloor \frac{p_2}{p_1} \right\rfloor + 1\right) \cdot \bar{w}_1 + \bar{w}_2 \geq 0$. Продолжая рассуждения далее, получим, что для того, чтобы функция ядра не имела критических ядер снизу, достаточно, чтобы

$$\sum_{i=1}^k \left(\left\lfloor \frac{p_k}{p_i} \right\rfloor + 1 \right) \cdot \bar{w}_i + \bar{w}_k \geq 0,$$

для $k = 1, 2, \dots, n$. Аналогичное условие получается при требовании $C(p_k) \geq 0$, которое трансформируется в

$$C(p_k) = \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{p_k}{p_i} \right\rfloor = \sum_{i=1}^k \bar{w}_i \left\lfloor \frac{p_k}{p_i} \right\rfloor \geq 0.$$

Рассмотрим условие отсутствия верхних критических ядер.

Принимая во внимание условие $\sum_{i=1}^n \bar{w}_i \cdot P_i = C_P$, заметим, что при переходе от величины $X = P - 1$ к величине P разница между ядрами $C(P)$ и $C(P - 1)$ составит сумму $\bar{w}_1 + \bar{w}_2 + \dots + \bar{w}_n$. Если $C(P - 1) \geq C(P)$ (т.е. $C(P - 1)$ – критическое ядро), то эта сумма не превосходит 0. Следовательно, минимальным условием отсутствия критических ядер сверху является

$$\sum_{i=1}^n \bar{w}_i > 0.$$

Продолжая рассуждения, аналогичные рассуждениям для критических ядер снизу, заметим, что для отсутствия критических ядер сверху нужно проверить, не превосходят ли значения функции ядра от диапазона C_P значения функций ядра от чисел, меньших P , содержащих нулевые остатки от деления на модули RNS. Следовательно, необходимо потребовать, чтобы $C(P - p_k) < C_P$ для всех $k = 1, 2, \dots, n$.

Вычислим $C(P - p_k)$.

$$C(P - p_k) = (P - p_k) \frac{C_P}{P} - \frac{1}{P} \sum_{i=1}^n \bar{w}_i \cdot P_i |p_i - p_k|_{p_i}.$$

Если $i = k$, то $|p_i - p_k|_{p_i} = 0$. Заметим, что для любых i и k , $i \neq k$, из множества натуральных чисел $1, 2, \dots, n$

$$|p_i - p_k|_{p_i} = \left\lfloor \frac{p_k}{p_i} \right\rfloor \cdot p_i + p_i - p_k.$$

Отсюда

$$C(P - p_k) = C_P - p_k \cdot \frac{C_P}{P} - \frac{1}{P} \cdot \sum_{i=1, i \neq k}^n \bar{w}_i \cdot P_i \cdot \left(\left\lfloor \frac{p_k}{p_i} \right\rfloor \cdot p_i + p_i - p_k \right).$$

Далее, так как $p_k - \left\lfloor \frac{p_k}{p_k} \right\rfloor p_k = 0$, то

$$C(P - p_k) = C_P - p_k \cdot \frac{C_P}{P} - \sum_{i=1, i \neq k}^n \bar{w}_i + \frac{1}{P} \cdot \sum_{i=1}^n \bar{w}_i \cdot P_i \left(p_k - \left\lfloor \frac{p_k}{p_i} \right\rfloor \right).$$

Учитывая, что

$$C(p_k) = p_k \cdot \frac{C_P}{P} - \frac{1}{P} \cdot \sum_{i=1}^n \bar{w}_i \cdot P_i \left(p_k - \left\lfloor \frac{p_k}{p_i} \right\rfloor \right),$$

получим

$$C(P - p_k) = C_P - C(p_k) - \sum_{i=1, i \neq k}^n \bar{w}_i.$$

Потребуем, чтобы $C(P - p_k) < C_P$, тогда

$$C_P - C(p_k) - \sum_{i=1, i \neq k}^n \bar{w}_i < C_P.$$

Откуда

$$C(p_k) + \sum_{i=1, i \neq k}^n \bar{w}_i > 0$$

или

$$\sum_{i=1}^n \bar{w}_i \left\lfloor \frac{p_k}{p_i} \right\rfloor + \sum_{i=1, i \neq k}^n \bar{w}_i = \sum_{i=1}^n \bar{w}_i \cdot \left(\left\lfloor \frac{p_k}{p_i} \right\rfloor + 1 \right) + \bar{w}_k > 0.$$

Теорема доказана. □

3.6.3 Функция Pirlo и Impedovo

Обобщив результат, полученный (Dimauro и др., 1993) в работе [195], исследовательская группа в составе Pirlo и Impedovo в 2013 году [312] предложила использовать минимальную функцию ядра Акушского без критических ядер.

Данный подход является аналогичным методу сравнения с использованием диагональной функции. Функция Pirlo имеет следующий вид

$$Pi(X) = \left\lfloor \frac{X}{p_n} \right\rfloor. \quad (3.53)$$

Однако, формула (3.53) малоприспособлена для практических вычислений, в связи с чем была предложена аналитическая функция для вычисления функции Pirlo

$$Pi(X) = \left\lfloor \sum_{i=1}^n k_i^{**} \cdot x_i \right\rfloor_{P_n}, \quad (3.54)$$

где $k_i^{**} = \left\lfloor \frac{|P_i^{-1}|_{p_i} P_i}{p_n} \right\rfloor$.

Так как функция Pirlo (3.54) является монотонно возрастающей, то она может быть использована для сравнения чисел, т.е. если $Pi(X) < Pi(Y)$, то $X < Y$. Однако, возможны случаи, когда $Pi(X) = Pi(Y)$, и в этом случае необходимо сравнивать остатки от деления: $X < Y$, когда $x_n < y_n$; $X = Y$, когда $x_n = y_n$; иначе $X > Y$.

Рассмотрим пример сравнения чисел.

Пример 3.6.1. Сравним ранее использовавшиеся числа $X = (2, 2, 3)$ и $Y = (1, 3, 4)$, представленные в RNS с модулями $(3, 5, 7)$. Для начала вычислим необходимые константы

$$\begin{aligned} P_3 &= 15, \\ k_1^{**} &= \left\lfloor \frac{|P_1^{-1}|_{p_1} P_1}{p_3} \right\rfloor = \left\lfloor \frac{2 \cdot 35}{7} \right\rfloor = 10, \\ k_2^{**} &= \left\lfloor \frac{|P_2^{-1}|_{p_2} P_2}{p_3} \right\rfloor = \left\lfloor \frac{1 \cdot 21}{7} \right\rfloor = 3, \\ k_3^{**} &= \left\lfloor \frac{|P_3^{-1}|_{p_3} P_3}{p_3} \right\rfloor = \left\lfloor \frac{1 \cdot 15}{7} \right\rfloor = 2. \end{aligned}$$

Найдем значение функции Pirlo

$$\begin{aligned} Pi(X) &= |2 \cdot 10 + 2 \cdot 3 + 3 \cdot 2|_{15} = 2, \\ Pi(Y) &= |1 \cdot 10 + 3 \cdot 3 + 4 \cdot 2|_{15} = 12. \end{aligned}$$

Поскольку $Pi(X) < Pi(Y)$, то $X < Y$.

В работе (Mohan, 2016) [283] показано, что функция Pirlo проигрывает Китайской теореме об остатках с точки зрения ее эффективности для сравнения в RNS, так как требует дополнительных сравнений чисел.

В данном разделе установлено, что подбор весов функции ядра при заданном C_P является комбинаторной задачей. Решение этой задачи приведено далее.

3.7 Сравнение чисел на основе алгоритма определения знака числа

С целью оптимизации алгоритма сравнения чисел иногда целесообразно использовать алгоритм определения знака числа.

Некоторые приложения в RNS требуют использования отрицательных чисел. Для определения знака числа в RNS с отрицательными числами необходимо сравнить это число с серединой диапазона. Следует также обратить внимание, что в данном случае отрицательные числа представляются положительными значениями, и для сравнения чисел сначала нужно определить их знак.

RNS с модулями $\{p_1, p_2, \dots, p_n\}$ и динамическим диапазоном $P = \prod_{i=1}^n p_i$ подходит для представления чисел X , удовлетворяющих следующим соотношениям

$$\begin{aligned} -\frac{P-1}{2} \leq X \leq \frac{P-1}{2}, \text{ если } |P|_2 = 1, \\ -\frac{P}{2} \leq X \leq \frac{P}{2} - 1, \text{ иначе.} \end{aligned}$$

Согласно (Omondi и Premkumar, 2007) [297], если $X = (x_1, x_2, \dots, x_n)$ – положительное число, то отрицательным будет число $-X = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$, где \bar{x}_i является дополнением x_i до модуля p_i . Например, для RNS с модулями (3, 5, 7) и числа $X = 17 = (2, 2, 3)$ получим, $-X = (3 - 2, 5 - 2, 7 - 3) = (1, 3, 4)$. Очевидно, что для перехода от восстановленного числа к отрицательной форме необходимо отнять значение динамического диапазона, т.е. $(1, 3, 4) = 88 = 88 - 105 = -17$. Если рассматривать весь динамический диапазон, то числа распределяются следующим образом: $\underline{0, 1, \dots, 52}, \overline{-52, -51, \dots, -1}$.

Теперь, когда заданы отрицательные числа, возникает необходимость определения знака числа. Существует ряд подходов к определению знака чисел в RNS: восстановление числа с помощью Китайской теоремы об остатках

(КТО), использование обобщенной позиционной системы счисления (ОПСС) и другие.

Проблемой КТО является необходимость нахождения остатка по большому модулю P , что является довольно трудоемкой задачей, и последующего сравнения с константой.

Введем функцию знака $S(X)$ для системы с нечетным динамическим диапазоном P (в случае четного диапазона границей служит $\frac{P}{2}$)

$$S(X) = \begin{cases} 0, & \text{если } 0 \leq X < \frac{P-1}{2}, \\ 1, & \text{если } \frac{P-1}{2} \leq X < P. \end{cases} \quad (3.55)$$

Рассмотрим на примере сравнение чисел в RNS, кодирующей не только положительные, но и отрицательные значения.

Пример 3.7.1. Сравним числа $X = 17 = (2, 2, 3)$ и $Y = -8 = (1, 2, 6)$ в RNS с модулями $(3, 5, 7)$. Если $X > Y$, то $(X - Y) > 0$. Найдём разность

$$X - Y = (2 - 1, 2 - 2, 3 - 6) = (1, 0, 4).$$

Применим приближенную формулу на основе КТО и сравним результат с серединой диапазона, т.е. с $\frac{1}{2}$. Все необходимые константы предварительно вычислены в предыдущих примерах.

$$\frac{X}{P} = \left| \sum_{i=1}^3 \frac{x_i \cdot |P_i^{-1}|_{p_i}}{p_i} \right|_1 = \left| \frac{2}{3} + \frac{4}{7} \right|_1 = \frac{5}{21} < \frac{1}{2}.$$

Поскольку полученное значение меньше середины диапазона, то оно положительное и значит $X > Y$.

Стоит отметить, что для корректного сравнения чисел на основе алгоритма определения знака числа требуется удвоение диапазонов RNS, что ведет к дополнительным вычислительным нагрузкам при обработке данных, но при таком подходе необходимо вычисление лишь одной позиционной характеристики числа (разности сравниваемых чисел).

3.8 Модифицированная диагональная функция

Для уменьшения вычислительной сложности операции сравнения чисел в RNS в работе (Babenko и др., 2019) [24] была предложена новая позиционная характеристика, основанная на *модифицированной диагональной функции* (Modified Diagonal Function – MDF), использующей диагональную функцию из Раздела 3.4 и приближенный метод из Раздела 3.3.3. Суть метода, основанного на MDF, состоит в том, чтобы вычислять относительную величину диагональной функции к SQ . Такой подход позволяет заменить вычислительно сложную операцию нахождения остатка от деления на SQ на взятие дробной части числа, коэффициенты k_i^* при этом заменяются на относительные величины k_i^* на SQ в формуле (3.35).

$$\tilde{D}(X) = \frac{D(X)}{SQ} = \left\lfloor \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right\rfloor_1. \quad (3.56)$$

Использование формулы (3.56) позволяет перейти от целочисленных вычислений в полуинтервале $[0, SQ)$ к вычислениям с дробями в полуинтервале $[0, 1)$. Для реализации данного подхода в целых числах используем масштабирование, которое позволит заменить операцию нахождения остатка от деления на SQ на операцию взятия младших N_M бит числа. Корректный переход от вычислений с дробными числами к вычислениям в целых числах может быть усовершенствован за счет следующих шагов:

1. Умножение каждой константы на 2^{N_M} , где N_M – это количество битов дробной части.
2. Округление вверх вещественного числа, скажем для Z , данная операция равносильна вычислению $\lceil Z \rceil$, т.е. наименьшего целого числа большего или равного Z .
3. Выполнение всех вычислений по модулю 2^{N_M} (значения разрядов, начиная с N_M бита и старше можно не вычислять).

Математически вышеизложенные рекомендации по вычислению позиционной характеристики с использованием формулы (3.56) можно выразить следующей формулой

$$\hat{D}(X) = \left\lfloor \sum_{i=1}^n \hat{k}_i \cdot x_i \right\rfloor_{2^{N_M}}, \quad (3.57)$$

где $\hat{k}_i = \left\lceil \frac{k_i^* \cdot 2^{N_M}}{SQ} \right\rceil$, $i \in \overline{1, n}$.

Основной проблемой при таком подходе является точность, так как в большинстве случаев дробные числа представляют собой бесконечные двоичные последовательности и не могут быть точно представлены конечным числом бит. Для ответа на вопрос о количестве бит N_M , необходимом для корректного вычисления позиционной характеристики, докажем следующую теорему.

Теорема 3.8.1. *Если SQ не является степенью двойки и $N_M \geq \lceil \log_2(SQ \cdot (m - 1)) \rceil$, то функция $\hat{D}(X)$ на промежутке $[0, P)$ строго возрастающая, где $m = \max_{1 \leq i \leq n} p_i$.*

Доказательство. Во-первых, вычислим $D(X - 1)$ для любого $0 < X < P$. Для этого воспользуемся следующим равенством

$$\left\lfloor \frac{X - 1}{p_i} \right\rfloor = \begin{cases} \left\lfloor \frac{X}{p_i} \right\rfloor, & \text{если } x_i \neq 0, \\ \left\lfloor \frac{X}{p_i} \right\rfloor - 1, & \text{если } x_i = 0. \end{cases} \quad (3.58)$$

Подставляя (3.58) в (3.34), получим

$$D(X - 1) = \sum_{i=1}^n \left\lfloor \frac{X - 1}{p_i} \right\rfloor = \sum_{i=1}^n \left\lfloor \frac{X}{p_i} \right\rfloor - \sum_{i=1}^n z(x_i) = D(X) - \sum_{i=1}^n z(x_i), \quad (3.59)$$

где

$$z(x_i) = \begin{cases} 0, & \text{если } x_i \neq 0, \\ 1, & \text{если } x_i = 0. \end{cases}$$

Учитывая, что $\tilde{D}(X - 1) = \frac{D(X-1)}{SQ}$ и формулу (3.59), получим

$$\tilde{D}(X - 1) = \frac{D(X)}{SQ} - \frac{1}{SQ} \cdot \sum_{i=1}^n z(x_i) = \tilde{D}(X) - \frac{1}{SQ} \cdot \sum_{i=1}^n z(x_i). \quad (3.60)$$

Заметим, что \hat{k}_i можно представить в виде

$$\hat{k}_i = \left\lceil \frac{k_i^* \cdot 2^{N_M}}{SQ} \right\rceil = \frac{k_i^* \cdot 2^{N_M}}{SQ} + R_i,$$

где $R_i = \left\lfloor \frac{k_i^* \cdot 2^{N_M}}{SQ} \right\rfloor - \frac{k_i^* \cdot 2^{N_M}}{SQ}$. Так как SQ не делит 2^{N_M} нацело, то $R_i \in (0, 1)$. Вычислим $R = \sum_{i=1}^n R_i$, получим

$$\begin{aligned} R &= \sum_{i=1}^n R_i = \sum_{i=1}^n \left(\left\lfloor \frac{k_i^* \cdot 2^{N_M}}{SQ} \right\rfloor - \frac{k_i^* \cdot 2^{N_M}}{SQ} \right) \\ &= \sum_{i=1}^n \left\lfloor \frac{k_i^* \cdot 2^{N_M}}{SQ} \right\rfloor - \frac{2^{N_M}}{SQ} \cdot \sum_{i=1}^n k_i^*. \end{aligned} \quad (3.61)$$

Так как SQ делит $\sum_{i=1}^n k_i^*$, то из формулы (3.61) следует, что R является целым числом и удовлетворяет неравенству $1 \leq R < n$.

Определим значения функций $\hat{D}(X)$ и $\hat{D}(X-1)$, используя ранее введенные обозначения, получим

$$\begin{aligned} \hat{D}(X) &= \left| \sum_{i=1}^n \left(\frac{k_i^* \cdot 2^{N_M}}{SQ} + R_i \right) \cdot x_i \right|_{2^{N_M}} \\ &= \left| \sum_{i=1}^n \frac{k_i^* \cdot 2^{N_M}}{SQ} \cdot x_i + \sum_{i=1}^n R_i \cdot x_i \right|_{2^{N_M}} \\ &= \left| 2^{N_M} \cdot \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i + \sum_{i=1}^n R_i \cdot x_i \right|_{2^{N_M}}. \end{aligned} \quad (3.62)$$

Учитывая, что вещественное число x можно представить в виде $x = \lfloor x \rfloor + |x|_1$, то $\sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i$ можно представить в виде $\sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i = \left\lfloor \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right\rfloor + \left| \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right|_1$, формула (3.62) примет вид

$$\hat{D}(X) = \left| 2^{N_M} \cdot \left\lfloor \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right\rfloor + 2^{N_M} \cdot \left| \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right|_1 + \sum_{i=1}^n R_i \cdot x_i \right|_{2^{N_M}}. \quad (3.63)$$

Подставляя $\tilde{D}(X) = \left\lfloor \sum_{i=1}^n \frac{k_i^*}{SQ} \cdot x_i \right\rfloor$ в формулу (3.63), получим

$$\hat{D}(X) = \left| 2^{N_M} \cdot \tilde{D}(X) + \sum_{i=1}^n R_i \cdot x_i \right|_{2^{N_M}}. \quad (3.64)$$

Используя формулу (3.64) и учитывая, что $X-1 \xrightarrow{RNS} (|x_1-1|_{p_1}, |x_2-1|_{p_2}, \dots, |x_n-1|_{p_n})$, имеем

$$\hat{D}(X-1) = \left| 2^{N_M} \cdot \tilde{D}(X-1) + \sum_{i=1}^n R_i \cdot |x_i-1|_{p_i} \right|_{2^{N_M}}. \quad (3.65)$$

Так как для любого $1 \leq i \leq n$ выполняется равенство

$$|x_i - 1|_{p_i} = \begin{cases} x_i - 1, & \text{если } x_i \neq 0, \\ p_i - 1, & \text{если } x_i = 0, \end{cases}$$

выражение $\sum_{i=1}^n R_i \cdot |x_i - 1|_{p_i}$ преобразуется к виду

$$\sum_{i=1}^n R_i |x_i - 1|_{p_i} = \sum_{i=1}^n R_i \cdot x_i - R + \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i. \quad (3.66)$$

Подставляя (3.66) и (3.60) в (3.65), получим

$$\hat{D}(X - 1) = \left| 2^{N_M} \cdot \tilde{D}(X) + \sum_{i=1}^n R_i \cdot x_i - \frac{2^{N_M}}{SQ} \sum_{i=1}^n z(x_i) - R + \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i \right|_{2^{N_M}}. \quad (3.67)$$

С учетом формулы (3.64), формула (3.67) переписется в виде

$$\hat{D}(X - 1) = \left| \hat{D}(X) - \frac{2^{N_M}}{SQ} \sum_{i=1}^n z(x_i) - R + \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i \right|_{2^{N_M}}. \quad (3.68)$$

Из (3.68) следует, что $\hat{D}(X) - \hat{D}(X - 1)$ равно

$$\hat{D}(X) - \hat{D}(X - 1) = \left| \frac{2^{N_M}}{SQ} \sum_{i=1}^n z(x_i) + R - \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i \right|_{2^{N_M}}. \quad (3.69)$$

Таким образом, для корректного сравнения чисел в RNS необходимо и достаточно, чтобы выполнялись два условия

Условие 1. $2^{N_M} \tilde{D}(X) + \sum_{i=1}^n R_i \cdot x_i < 2^{N_M}$.

Условие 2. $\frac{2^{N_M}}{SQ} \sum_{i=1}^n z(x_i) + R - \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i > 0$.

Выполнение условия 1 позволяет перейти от вычисления остатка от деления к вычислению $\text{mod} 2^{N_M}$ для $\hat{D}(X)$ в формуле (3.64) и для $\hat{D}(X - 1)$ в формуле (3.68). Если выполняются условия 1 и 2, то функция $\hat{D}(X)$ строго возрастающая.

Покажем, что если $N_M \geq \lceil \log_2(SQ \cdot (m - 1)) \rceil$, то выполняются оба условия, и, следовательно, $\hat{D}(X)$ является строго возрастающей функцией. Так как

функция $\tilde{D}(X)$ монотонно возрастающая, то $\tilde{D}(X-1) \leq \tilde{D}(X)$. Следовательно, максимального значения функция $\tilde{D}(X)$ достигает в точке $P-1$. Вычислим $\tilde{D}(P-1)$, получим

$$\begin{aligned}\tilde{D}(P-1) &= \frac{D(P-1)}{SQ} = \frac{\sum_{i=1}^n \left\lfloor \frac{P-1}{p_i} \right\rfloor}{SQ} \\ &= \frac{\sum_{i=1}^n (P_i - 1)}{SQ} = 1 - \frac{n}{SQ}.\end{aligned}\quad (3.70)$$

Так как $0 < R_i < 1$ и $m = \max_{1 \leq i \leq n} p_i$, то

$$\sum_{i=1}^n z(x_i) \cdot R_i \cdot x_i < n \cdot (m-1).\quad (3.71)$$

Подставим в условие 1 значения (3.70) и (3.71), получим

$$2^{N_M} \left(1 - \frac{n}{SQ}\right) + n \cdot (m-1) < 2^{N_M}.$$

Следовательно, $2^{N_M} > SQ \cdot (m-1)$.

Покажем, что если выполняется неравенство $2^{N_M} > SQ \cdot (m-1)$, то выполняется условие 2.

$$\begin{aligned}\frac{2^{N_M}}{SQ} \sum_{i=1}^n z(x_i) + R - \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i &> \\ \frac{SQ \cdot (m-1)}{SQ} \sum_{i=1}^n z(x_i) + R - \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i &= \\ (m-1) \sum_{i=1}^n z(x_i) + R - \sum_{i=1}^n z(x_i) \cdot R_i \cdot p_i &= \\ \sum_{i=1}^n z(x_i) \cdot (m-1 - R_i \cdot p_i) + R.\end{aligned}$$

Так как для любого $p_i \neq m$ выполняется неравенство $m - p_i \geq 1$ и $R_i < 1$, то $m - 1 \geq p_i$. Следовательно, $z(x_i) \cdot (m - 1 - R_i \cdot p_i) \geq (m - 1) \cdot (1 - R_i) \geq 0$. Рассмотрим случай когда $p_i = m$, тогда $z(x_i) \cdot (m - 1 - R_i \cdot p_i) = z(x_i) \cdot (m \cdot (1 - R_i) - 1) > -1$. Следовательно, $\sum_{i=1}^n z(x_i) \cdot (m - 1 - R_i \cdot p_i) > -1$. Учитывая что $R \geq 1$, получим, что $\sum_{i=1}^n z(x_i) \cdot (m - 1 - R_i \cdot p_i) + R > 0$. Следовательно, условие 2 выполняется.

Теорема доказана. □

3.9 Модификация алгоритма сравнения чисел в RNS

Рассмотрим следующую функцию в качестве позиционной характеристики

$$f(X) = \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i \cdot x_i}{2^N} \right\rfloor,$$

где $\bar{k}_i = \left\lfloor \frac{2^N |P_i^{-1}|_{p_i}}{p_i} \right\rfloor$.

3.9.1 Сравнение чисел в RNS с нечетным диапазоном

Докажем вспомогательную теорему, которая позволит уточнить оценку значения N для операции сравнения чисел из работы [28].

Теорема 3.9.1. *Если $N = \lceil \log_2(n \cdot P - n) \rceil$, то справедливо следующее равенство [7, 24]*

$$\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^n k_i x_i}{P} \right\rfloor, \quad (3.72)$$

где $\bar{k}_i = \left\lfloor \frac{2^N |P^{-1}|_{p_i}}{p_i} \right\rfloor$, $k_i = |P^{-1}|_{p_i} P_i$.

Доказательство. Так как k_i и \bar{k}_i связаны равенством $\bar{k}_i = \frac{2^N k_i}{P} - \frac{|2^N k_i|_P}{P}$, то выражение $\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor$ примет вид

$$\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor = \left\lfloor \frac{1}{P} \sum_{i=1}^n k_i x_i - \frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P \cdot x_i \right\rfloor. \quad (3.73)$$

Подставив $\frac{1}{P} \sum_{i=1}^n k_i x_i = \left\lfloor \frac{1}{P} \sum_{i=1}^n k_i x_i \right\rfloor + \frac{X}{P}$ в (3.73), получим

$$\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^n k_i x_i}{P} \right\rfloor + \left\lfloor \frac{X}{P} - \frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P \cdot x_i \right\rfloor. \quad (3.74)$$

Из (3.74) следует, что условие Теоремы 3.9.1 эквивалентно следующему неравенству

$$0 \leq \frac{X}{P} - \frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P x_i < 1. \quad (3.75)$$

Согласно Китайской теореме об остатках, X удовлетворяет условию $0 \leq X < P$. Следовательно, $0 \leq \frac{X}{P} < 1$. Принимая во внимание, что $\frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P x_i \geq 0$, получим, что правая часть двойного неравенства (3.75) верна для всех $N \geq 0$.

Рассмотрим левую часть двойного неравенства (3.75). При $X = 0$ она выполняется для любого $N \geq 0$. Пусть X удовлетворяет неравенству $1 \leq X < P$, тогда левую часть неравенства (3.75) можно представить в следующем виде

$$2^N \geq \frac{1}{X} \sum_{i=1}^n |2^N k_i|_P x_i. \quad (3.76)$$

Так как $|2^N k_i|_P \leq P - 1$, то $\sum_{i=1}^n |2^N k_i|_P x_i \leq (P - 1) \sum_{i=1}^n x_i$. Следовательно, для всех $1 \leq X < P$ справедливо следующее неравенство

$$\frac{1}{X} \sum_{i=1}^n |2^N k_i|_P x_i \leq n \cdot (P - 1). \quad (3.77)$$

Из (3.76) и (3.77) следует, что если $N = \lceil \log_2(-n + n \cdot P) \rceil$, то левая часть неравенства (3.75) выполняется, следовательно, выполняется равенство (3.72).

Теорема доказана. \square

Исследуем при каком N функция $f(X)$ является строго возрастающей на промежутке $[0, P)$, для этого докажем следующую теорему.

Теорема 3.9.2. *Если $N = \lceil \log_2(-n + n \cdot P) \rceil$, то функция $f(X)$ на промежутке $[0, P)$ строго возрастающая [7, 24].*

Доказательство. Для того чтобы $f(X)$ являлась строго возрастающей функцией необходимо и достаточно, чтобы для всех целых чисел $1 \leq X \leq P - 1$ выполнялось следующее условие

$$f(X) - f(X - 1) > 0. \quad (3.78)$$

Так как $|X|_{2^N} = X - \lfloor \frac{X}{2^N} \rfloor \cdot 2^N$, то функцию $f(X)$ можно представить в следующем виде

$$f(X) = \sum_{i=1}^n \bar{k}_i x_i - \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor \cdot 2^N. \quad (3.79)$$

Принимая во внимание, что $\sum_{i=1}^n \bar{k}_i (x_i - |x_i - 1|_{p_i}) = \sum_{i=1}^n \bar{k}_i - \sum_{x_i=0} \bar{k}_i p_i$, получим, что

$$f(X) - f(X - 1) = \sum_{i=1}^n \bar{k}_i - \sum_{x_i=0} \bar{k}_i p_i - \left(\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i |x_i - 1|_{p_i}}{2^N} \right\rfloor \right) \cdot 2^N. \quad (3.80)$$

Так как условие Теоремы 3.9.1 выполнено, то

$$\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i x_i}{2^N} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i |x_i - 1|_{p_i}}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^n k_i x_i}{P} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^n k_i |x_i - 1|_{p_i}}{P} \right\rfloor. \quad (3.81)$$

С учетом теоремы из работы (Chervyakov и др., 2017) [11] и Теоремы 3.9.1, формула (3.81) примет вид

$$\begin{aligned} \left\lfloor \frac{\sum_{i=1}^n k_i x_i}{P} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^n k_i |x_i - 1|_{p_i}}{P} \right\rfloor &= \left\lfloor \frac{\sum_{i=1}^n k_i}{P} \right\rfloor - \sum_{x_i=0} |P_i^{-1}|_{p_i} \\ &= \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i}{2^N} \right\rfloor - \sum_{x_i=0} |P_i^{-1}|_{p_i}. \end{aligned} \quad (3.82)$$

Подставив (3.82) в (3.80), получим

$$f(X) - f(X - 1) = \sum_{i=1}^n \bar{k}_i - \sum_{x_i=0} \bar{k}_i p_i - \left(\left\lfloor \frac{\sum_{i=1}^n \bar{k}_i}{2^N} \right\rfloor - \sum_{x_i=0} |P_i^{-1}|_{p_i} \right) \cdot 2^N. \quad (3.83)$$

Так как $\sum_{i=1}^n \bar{k}_i - \left\lfloor \frac{\sum_{i=1}^n \bar{k}_i}{2^N} \right\rfloor \cdot 2^N = |\sum_{i=1}^n \bar{k}_i|_{2^N}$ и $|P_i^{-1}|_{p_i} \cdot 2^N - \bar{k}_i p_i = \frac{|2^N k_i|_P}{P_i}$, то для всех $i = \overline{1, n}$ формула (3.83) примет вид

$$f(X) - f(X - 1) = \left\lfloor \sum_{i=1}^n \bar{k}_i \right\rfloor_{2^N} + \sum_{x_i=0} \frac{|2^N \cdot k_i|_P}{P_i}. \quad (3.84)$$

Так как $|\sum_{i=1}^n \bar{k}_i|_{2^N} > 0$, то из (3.84) следует, что $f(X) - f(X - 1) > 0$, и, следовательно, функция $f(X)$ строго возрастает на промежутке $[0, P)$.

Теорема доказана. \square

Из Теоремы 3.9.2 следует, что введенная функция является строго монотонной, следовательно, ее можно использовать в качестве позиционной характеристики для сравнения чисел в RNS.

Предложенный подход позволяет уменьшить вычислительную сложность алгоритма сравнения чисел в RNS. Эффективная аппаратная реализация операции $|x \cdot y|_{2^N}$ позволяет уменьшить логическую схему по сравнению с классическим умножением двух чисел $x \cdot y$.

3.9.2 Сравнение чисел в RNS, содержащий модуль, равный степени двойки

Так как модули RNS являются взаимно простыми числами, следовательно, возможно наличие только одного четного модуля. Без потери общности будем считать, что n -ый модуль имеет следующий вид $p_n = 2^t$. Так как $p_n = 2^t$, то, согласно свойствам RNS, числа X, Y могут быть представлены в следующем виде

$$X = A \cdot 2^t + x_n, \quad Y = B \cdot 2^t + y_n. \quad (3.85)$$

Для сравнения чисел, сравним A и B . Если $A < B$, то $X < Y$. Аналогично, если $A > B$, то $X > Y$. В случае когда $A = B$ необходимо дополнительное сравнение остатков: $X < Y$ при условии, что $x_n < y_n$; $X = Y$ при условии, что $x_n = y_n$; иначе $X > Y$.

Так как n -ый модуль RNS четный, следовательно, модули p_1, p_2, \dots, p_{n-1} являются нечетными числами, тогда P_n – нечетное число. Коэффициенты A и B удовлетворяют неравенствам: $0 \leq A < P_n$ и $0 \leq B < P_n$. Вычислив значения A и B в RNS по модулям p_1, p_2, \dots, p_{n-1} , можно сравнить их, используя введенную функцию $f(X)$.

Таким образом, алгоритм сравнения чисел X и Y будет иметь вид Алгоритма 3. Количество операций, необходимых для получения результата с использованием данного алгоритма равно: умножений – $(4n - 4)$, вычитаний – $(2n - 2)$, сложений – n .

Алгоритм 3: Алгоритм сравнения чисел X и Y

Input: $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n),$

$Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n),$

$p_1, p_2, \dots, p_{n-1}, p_n,$

$I_i = \left\lfloor \frac{1}{p_n} \right\rfloor_{p_i}, \forall i = \overline{1, n-1},$

$\bar{k}_i = \left\lfloor 2^N \cdot |1/P_i^*|_{p_i} / p_i \right\rfloor, \forall i = \overline{1, n-1},$

где $N = \lceil \log_2(-n + nP_n) \rceil, P_i^* = P_n/p_i, \forall i = \overline{1, n-1}$

Output: $X > Y - '10', X < Y - '01', X = Y - '00'$

```

1 for  $i = 1, i < n, i ++$  do
2    $a_i = |x_i - x_n|_{p_i}$ ; Parallel processing
3    $a_i = |a_i \cdot I_i|_{p_i}$ ; Parallel processing
4    $b_i = |y_i - y_n|_{p_i}$ ; Parallel processing
5    $b_i = |b_i \cdot I_i|_{p_i}$ ; Parallel processing
6  $S_X = 0; S_Y = 0;$ 
7 for  $i = 1, i < n, i ++$  do
8    $S_X = |S_X + \bar{k}_i \cdot a_i|$ ; Sum tree
9    $S_Y = |S_Y + \bar{k}_i \cdot b_i|$ ; Sum tree
10 if  $S_X > S_Y$  then
     $\lfloor$  Result: '10'
11 if  $S_X < S_Y$  then
     $\lfloor$  Result: '01'
12 if  $a_n > b_n$  then
     $\lfloor$  Result: '10'
13 if  $a_n < b_n$  then
     $\lfloor$  Result: '01'
Result: '00'

```

3.10 Оценка производительности алгоритмов сравнения чисел в RNS

Более широкому использованию RNS препятствует необходимость выполнения немодульных арифметических операций, таких как обратное преобразование, определение знака и сравнение чисел. Они характеризуются высокой сложностью и используют при вычислении значения всех остатков. Функции ядра для реализации немодульных операций, предложенные в ранних работах по RNS, отличаются высокой степенью обобщенности и на первый взгляд непригодны для использования в практических приложениях. Данная точка зрения долгое время препятствовала разработке конкурентоспособных реализаций на их основе и не позволяла в полной мере использовать их высокий потенциал. Представленное исследование показало, что функции, используемые для сравнения чисел в RNS, должны быть монотонными и иметь только неотрицательные коэффициенты. Предложенная монотонная функция ядра минимального диапазона (Minimum-Range Monotonic Core Function – MMCF) позволяет сравнивать числа в RNS с произвольным общим набором модулей. Сравнение чисел в RNS, основанное на монотонной функции ядра минимального диапазона, требует меньших аппаратных затрат и, в некоторых случаях, вносит меньшую задержку, по сравнению с методом, основанным на диагональной функции. Установлено, что диагональная функция – это не что иное, как частный случай функции ядра, при котором все ее коэффициенты равны 1.

3.10.1 Анализ алгоритмов сравнения чисел в RNS

В данном разделе представлено сравнение разработанного устройства, реализующего операцию сравнения чисел в RNS, с его наиболее эффективными известными аналогами, основанными на диагональной функции [322], модифицированной диагональной функции [26] и устройствами сравнения на основе CRT [351]. В работах [311, 322] была установлена самая быстрая версия устройства сравнения чисел в RNS на основе CRT (рис. 3.1). Именно это устройство

Таблица 20 — Параметры МОМА, использующихся в устройствах сравнения чисел в RNS

| Тип устройства сравнения | Модули | Размер операнда, бит |
|--|-------------------------|---|
| На основе CRT | $P = \prod_{i=1}^n p_i$ | $a_P = \lceil \log_2 P \rceil$ |
| Диагональная функция | $SQ = \sum_{i=1}^n P_i$ | $a_{SQ} = \lceil \log_2 SQ \rceil$ |
| Модифицированная диагональная функция | 2^{N_M} | $N_M = \lceil \log_2 (SQ \cdot (m - 1)) \rceil$ |
| Предложенная ММСФ | $P_m = P/m$ | $a_{P_m} = \lceil \log_2 P_m \rceil$ |

использовалось при сравнении с другими устройствами, реализующими операцию сравнения в RNS, включая предложенное.

Отметим, что устройства сравнения, основанные на диагональной функции и на CRT, имеют схожие структуры (рис. 3.2) и отличаются лишь несколькими моментами:

1. Устройства сравнения, основанные на диагональной функции и на CRT, используют n -операндные модулярные сумматоры (Multi-Operand Modular Adder – МОМА) по модулям $\text{mod}SQ$ и $\text{mod}P$, соответственно.
2. В устройстве сравнения на основе CRT позиционное сравнение состоит только из простого a -битового устройства сравнения, тогда как в устройстве на основе диагональной функции, для позиционного сравнения требуется дополнительная логика, применяемая в случае равенства диагональных функций.

Поскольку во всех трех случаях (диагональная функция, CRT, модифицированная диагональная функция) n -операндный МОМА с переменным модулем является основным составным блоком, проанализируем влияние размера данного модуля на сложность оборудования, используя характеристики, приведенные в таблице 20. Сравним размеры операндов, обрабатываемых устройствами сравнения чисел в RNS, построенными с использованием стандартной реализации на основе CRT и ММСФ. Полагая $P_m = P/p_n$ ($m = p_n$) и логарифмируя обе части, получаем $\log_2 P_m = \log_2 P - \log_2 m$, что приводит к следующему неравенству

$$a_P - \lceil \log_2 m \rceil \leq a_{P_m} \leq a_P - \lfloor \log_2 m \rfloor. \quad (3.86)$$

В частности, если $m = 2^q$, то $a_{P_m} = a_P - q$. Очевидно, что чем больше наибольший модуль m , тем относительно короче операнды МОМА (a_{P_m} по сравнению с a_P) и тем большая аппаратная экономия (на $(n-2)(a_P - a_{P_m})$ полных сумматора (Full Adder – FA) меньше только в дереве сохраняющих перенос сумматоров (Carry-Save Adder – CSA) МОМА) наблюдаются по сравнению с реализацией на основе CRT. При позиционном сравнении не наблюдается экономии, поскольку размеры a -битного устройства сравнения для CRT и общий размер двух устройств сравнения для MMCF аналогичны, хотя для последнего требуется несколько дополнительных конечных шлюзов.

Чтобы сравнить размеры операндов, обрабатываемых устройствами сравнения, построенными с использованием диагональной функции и MMCF, обратим внимание на следующее выражение

$$\frac{SQ}{P_m} = \sum_{i=1}^n \frac{m \cdot P_i}{P} = \sum_{i=1}^n \frac{m}{p_i} = 1 + m \cdot \sum_{i=1, p_i \neq m}^n \frac{1}{p_i}, \quad (3.87)$$

откуда получаем

$$\log_2 \frac{SQ}{P_m} = \log_2 SQ - \log_2 P_m = \log_2 \left(1 + m \cdot \sum_{i=1, p_i \neq m}^n \frac{1}{p_i} \right). \quad (3.88)$$

Из последнего уравнения получаем нижнюю и верхнюю границы количества бит

$$\left[\log_2 \left(1 + m \cdot \sum_{i=1, p_i \neq m}^n \frac{1}{p_i} \right) \right] \leq a_{SQ} - a_{P_m} \leq \left\lceil \log_2 \left(1 + m \cdot \sum_{i=1, p_i \neq m}^n \frac{1}{p_i} \right) \right\rceil. \quad (3.89)$$

Таким образом, поскольку $n \geq 2$, то $SQ = \sum_{i=1}^n P_i > P_m$. Из полученного неравенства $a_{P_m} < a_{SQ}$ вытекают следующие общие наблюдения.

1. МОМА $\text{mod} P_m$ работает с более короткими операндами, чем МОМА $\text{mod} SQ$, поэтому как внутренние сумматоры с сохранением переноса (CSA), так и конечные сумматоры с быстрым распространением переноса (Carry-Propagate Adder – CPA), используемые в МОМА $\text{mod} P_m$, короче на $a_{SQ} - a_{P_m}$ бит. Аппаратная экономия в сумматорах составляет примерно $n \cdot (a_{SQ} - a_{P_m})$ FA, т.е. аппаратная экономия растет как с числом модулей n , так и с размером наибольшего модуля m (3.89).

2. МОМА $\text{mod } P_m$ использует меньшее количество выходов каждой из n поисковых таблиц (Look Up Table – LUT), что подразумевает использование меньшей площади аппаратуры.
3. Выбор наибольшего модуля m для дополнительного сравнения при устранении неоднозначности, возникающей если $P_i(X) = P_i(Y)$, не влияет на задержку всего устройства сравнения чисел в RNS, потому что дополнительное сравнение может быть выполнено параллельно со сравнением $P_i(X)$ и $P_i(Y)$ (рис. 3.2).
4. Некоторое снижение задержки может наблюдаться в случае использования наборов модулей, для которых $\lceil \log_2 a_{P_m} \rceil < \lceil \log_2 a_{SQ} \rceil$. Это связано с тем, что все сумматоры с быстрым распространением переноса (CRA), используемые модулем МОМА, имеют на несколько уровней шлюза меньше, чем их аналоги, используемые модулем МОМА $\text{mod } SQ$. Примеры таких наборов модулей RNS будут приведены ниже.

Что касается модифицированной диагональной функции, далее будет показано, что N_M всегда значительно больше чем a_{P_m} . Данное свойство делает алгоритм сравнения чисел в RNS, основанный на модифицированной диагональной функции, перспективным для использования в некоторых криптографических приложениях.

3.10.2 Анализ и оценка сложности алгоритмов сравнения чисел в RNS

Чтобы выявить различия между размерами МОМА, используемыми в различных устройствах сравнения чисел в RNS в зависимости от количества модулей n и динамического диапазона, рассмотрим различные наборы модулей RNS $SP_{n,i}$, представленные в таблице 21, где n – количество модулей, i – номер набора из n модулей.

Таблица 21 — Размеры операндов МОМА, используемых при реализации устройств сравнения для различных наборов модулей RNS

| n | Модули RNS | P | SQ | P_m | a_P | a_{SQ} | a_{P_m} | N_M |
|-----|---------------------------------------|----------------------|-------------------|-------------------|-------|----------|-----------|-------|
| 3 | $SP_{3,1} = \{63, 65, 256\}$ | $1.04 \cdot 10^6$ | $3.68 \cdot 10^4$ | $4.09 \cdot 10^3$ | 20 | 16 | 12 | 24 |
| | $SP_{3,2} = \{127, 129, 512\}$ | $8.38 \cdot 10^6$ | $1.47 \cdot 10^5$ | $1.64 \cdot 10^4$ | 23 | 18 | 14 | 27 |
| 4 | $SP_{4,1} = \{7, 15, 17, 64\}$ | $1.14 \cdot 10^5$ | $3.24 \cdot 10^4$ | $1.79 \cdot 10^3$ | 17 | 15 | 11 | 21 |
| | $SP_{4,2} = \{15, 17, 31, 64\}$ | $5.05 \cdot 10^5$ | $8.77 \cdot 10^4$ | $7.91 \cdot 10^3$ | 19 | 17 | 13 | 23 |
| | $SP_{4,3} = \{63, 65, 127, 512\}$ | $2.66 \cdot 10^8$ | $1.09 \cdot 10^7$ | $5.20 \cdot 10^5$ | 28 | 24 | 19 | 33 |
| | $SP_{4,4} = \{251, 253, 255, 256\}$ | $4.15 \cdot 10^9$ | $6.53 \cdot 10^7$ | $1.62 \cdot 10^7$ | 32 | 26 | 24 | 34 |
| | $SP_{4,5} = \{507, 509, 511, 512\}$ | $6.75 \cdot 10^{10}$ | $5.30 \cdot 10^8$ | $1.32 \cdot 10^8$ | 36 | 29 | 27 | 38 |
| 5 | $SP_{5,1} = \{5, 7, 9, 11, 13\}$ | $4.50 \cdot 10^4$ | $2.80 \cdot 10^4$ | $3.47 \cdot 10^3$ | 16 | 15 | 12 | 19 |
| | $SP_{5,2} = \{5, 7, 9, 11, 16\}$ | $5.54 \cdot 10^4$ | $3.37 \cdot 10^4$ | $3.47 \cdot 10^3$ | 16 | 16 | 12 | 19 |
| | $SP_{5,3} = \{5, 7, 9, 11, 31\}$ | $1.07 \cdot 10^5$ | $6.20 \cdot 10^4$ | $3.47 \cdot 10^3$ | 17 | 16 | 12 | 21 |
| | $SP_{5,4} = \{7, 15, 17, 31, 32\}$ | $1.77 \cdot 10^6$ | $5.88 \cdot 10^5$ | $5.53 \cdot 10^4$ | 21 | 20 | 16 | 25 |
| | $SP_{5,5} = \{7, 15, 17, 31, 64\}$ | $3.54 \cdot 10^6$ | $1.12 \cdot 10^6$ | $5.53 \cdot 10^4$ | 22 | 21 | 16 | 27 |
| | $SP_{5,6} = \{7, 15, 17, 31, 128\}$ | $7.08 \cdot 10^6$ | $2.18 \cdot 10^6$ | $5.53 \cdot 10^4$ | 23 | 22 | 16 | 29 |
| | $SP_{5,7} = \{31, 63, 65, 127, 256\}$ | $4.13 \cdot 10^9$ | $3.11 \cdot 10^8$ | $1.61 \cdot 10^7$ | 32 | 29 | 24 | 37 |
| 6 | $SP_{6,1} = \{5, 7, 9, 11, 13, 16\}$ | $7.21 \cdot 10^5$ | $4.93 \cdot 10^5$ | $4.50 \cdot 10^4$ | 20 | 19 | 16 | 23 |
| | $SP_{6,2} = \{5, 7, 9, 11, 13, 32\}$ | $1.44 \cdot 10^6$ | $9.41 \cdot 10^5$ | $4.50 \cdot 10^4$ | 21 | 20 | 16 | 25 |
| | $SP_{6,3} = \{7, 9, 11, 13, 16, 17\}$ | $2.45 \cdot 10^6$ | $1.33 \cdot 10^6$ | $1.44 \cdot 10^5$ | 22 | 21 | 18 | 25 |
| | $SP_{6,4} = \{7, 9, 11, 13, 31, 32\}$ | $8.94 \cdot 10^6$ | $434 \cdot 10^6$ | $2.79 \cdot 10^5$ | 24 | 23 | 19 | 28 |
| | $SP_{6,5} = \{5, 7, 9, 11, 13, 256\}$ | $1.15 \cdot 10^7$ | $7.21 \cdot 10^6$ | $4.50 \cdot 10^4$ | 24 | 23 | 16 | 31 |

Таблица 22 — Размеры операндов МОМА, используемых при реализации устройств сравнения для различных наборов модулей RNS (продолжение таблицы 21)

| n | Модули RNS | P | SQ | P_m | a_P | a_{SQ} | a_{P_m} | N_M |
|-----|---|----------------------|----------------------|----------------------|-------|----------|-----------|-------|
| 7 | $SP_{7,1} = \{5, 7, 9, 11, 13, 17, 32\}$ | $2.45 \cdot 10^7$ | $1.74 \cdot 10^7$ | $7.66 \cdot 10^5$ | 25 | 25 | 20 | 30 |
| | $SP_{7,2} = \{19, 23, 25, 27, 29, 31, 64\}$ | $1.70 \cdot 10^{10}$ | $4.34 \cdot 10^9$ | $2.65 \cdot 10^8$ | 34 | 33 | 28 | 38 |
| 8 | $SP_{8,1} = \{5, 7, 9, 11, 13, 17, 19, 32\}$ | $4.66 \cdot 10^8$ | $3.56 \cdot 10^8$ | $1.45 \cdot 10^7$ | 29 | 29 | 24 | 34 |
| 11 | $SP_{11,1} = \{7, 11, 13, 17, 19, 23, 25, 27, 29, 31, 32\}$ | $1.55 \cdot 10^{14}$ | $8.49 \cdot 10^{13}$ | $4.85 \cdot 10^{12}$ | 48 | 47 | 43 | 52 |

Проанализировав экспериментальные данные из таблицы 21, можно выделить следующие основные преимущества устройств сравнения чисел в RNS, основанных на диагональной функции, по сравнению с аналогами на основе CRT:

1. Какие-либо существенные преимущества диагональной функции ($a_P - a_{SQ} \geq 4$) наблюдаются только для нескольких самых маленьких наборов модулей, составленных из $n = 3$ или 4 модулей: $SP_{3,1}$, $SP_{3,2}$, $SP_{4,3}$, $SP_{4,4}$ и $SP_{4,5}$.
2. Для $n \geq 6$ разница (если таковая имеется) между a_P и a_{SQ} становится несущественной, это означает, что диагональная функция фактически не дает каких-либо значимых преимуществ по сравнению со стандартной реализацией устройства сравнения чисел в RNS на основе CRT.

Так же, основываясь на экспериментальных данных из таблицы 21, можно отметить преимущества алгоритмов сравнения чисел в RNS на базе MMCF по сравнению с аналогами, использующими диагональную функцию:

1. Если использовать четный модуль $m = 2^q$ в предлагаемой конструкции, для $n \geq 7$ сохраняется аппаратное преимущество как минимум на $(q - 1) \cdot n$ FA по сравнению с аналогом, основанным на диагональной функции. Анализ столбцов SQ P_m таблицы 21 показывает, что верхняя граница редукции операнда MOA (3.89) является точной для большинства перечисленных выборочных наборов модулей RNS.
2. При использовании наборов модулей, для которых $\lceil \log_2 a_{SQ} \rceil \leq \lceil \log_2 a_{P_m} \rceil$ (выделено жирным шрифтом в столбце a_{P_m}), предложенное устройство сравнения чисел в RNS также работает быстрее, поскольку для него подходит уменьшенная на одну ступень по сравнению с диагональной функцией схема CPA. Например, для $SP_{6,1}$ устройство сравнения, основанное на диагональной функции, использует сумматор по модулю 493189, работающий с 19-битными операндами, при этом задержка используемого CPA составляет 12 задержек шлюза; с другой стороны, устройство сравнения на основе MMCF использует сумматор по модулю 45045, работающий с 16-битными операндами, при этом задержка используемого CPA составляет 8 задержек шлюза.
3. Данные таблицы 21 подтверждают, что полученная оценка верхней границы $a_{SQ} - a_{P_m}$ (3.89) не может быть уточнена.

Последний столбец N_M включен в таблицу 21 чтобы обозначить разницу в размерах операндов устройств на основе модифицированной диагональной функции [26] и предложенных устройств сравнения на основе ММСФ. Размер четного модуля N_M , используемого модифицированной диагональной функцией [26], для всех рассмотренных случаев значительно больше чем a_{P_m} , на 7-15 бит (например, для $SP_{6,5}$).

Все рассмотренные выше устройства сравнения чисел в RNS имеют общую структуру, представленную на рисунке 3.2. Основными блоками устройств сравнения чисел в RNS являются:

- $L(l, a)$ – справочная таблица из 2^l ячеек с длиной слова a -бит (временная задержка – $t_L(l, a)$);
- $МОМА(n, a)$ – модульный n -операндный сумматор с длиной слова a бит (временная задержка – $t_{МОМА}(n, a)$);
- $C(a)$ – двоичное устройство сравнения a -битовых целых чисел (с временной задержкой, обозначенной $t_C(a)$).

Для оценки задержки будем использовать те же обозначения, что и в работе [311]. Так же, как в работах [322] и [311], сложность различных реализаций оценивается посредством количества бит, используемых справочными таблицами (L), количества полных сумматоров (FA) и временной задержки (TD). В качестве единицы измерения временной задержки используется Δ – задержка логического элемента И-НЕ. Предполагается, что $t_{FA} = t_{MUX} = 2\Delta$, а $t_{XOR} = \Delta$.

Задержка справочной таблицы с l -битовым входом и a -битовым выходом (выраженная в Δ) может быть оценена с помощью формулы из работы [365]

$$t_L(l, a) = 2 + \left\lceil \log_f \frac{l}{2} \right\rceil + \left\lceil \log_f 2^{l/2} \right\rceil, \quad (3.90)$$

где f – максимальное количество входов, которое может обрабатывать логический вентиль. В частности, при $f = 3$ имеем $t_L(l, a) = 5\Delta$ для $4 \leq a \leq 6$, $t_L(l, a) = 7\Delta$ для $7 \leq a \leq 9$ и $t_L(l, a) = 8\Delta$ для $10 \leq a \leq 12$.

Задержки наиболее быстрых реализаций устройств сравнения позиционных чисел, подобных тем, что приведены в [246, с. 45-47], составляют $t_C(a) = 4\Delta$ для $2 \leq a \leq 4$, $t_C(a) = 8\Delta$ для $5 \leq a \leq 24$ и $t_C(a) = 12\Delta$ для $25 \leq a \leq 120$.

Таблица 23 — Минимальное количество ступеней CSA $\theta(n)$, достаточное для обработки n операндов МОМА

| | | | | | | | |
|-------------|---|---|-----|-----|-------|-------|-------|
| n | 3 | 4 | 5-6 | 7-9 | 10-13 | 14-19 | 20-28 |
| $\theta(n)$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

Задержка сумматора с упреждающим переносом (Carry-Look-Ahead – CLA) [246] составляет $t_{CLA}(a) = 6\Delta$ для $a \leq 8$, $t_{CLA}(a) = 8\Delta$ для $9 \leq a \leq 16$ и $t_{CLA}(a) = 12\Delta$ для $17 \leq a \leq 64$.

Следует отметить, что несмотря на небольшие дополнительные затраты на оборудование для всех оценок сложности далее будет использоваться МОМА из работы [309], который быстрее чем МОМА из работы [310], использовавшийся также в работе [311]. Блочная схема данного МОМА представлена на рисунке 3.6 с указанием оценок задержек. Согласно схеме, дерево CSA создает пару векторов S и C , которые разбиваются на две пары: наиболее значащие биты (Most Significant Bits – MSB) и наименее значащие биты $S = \{S_H, S_L\}$ и $C = \{C_H, C_L\}$, так что $\max\{S_L, C_L\} < P$. Фактическое точное общее количество бит в S и C , как и верхнюю границу количества MSB, которые могут входить в преобразователь MSB ($\max\{h_s, h_c\}$), можно найти в таблице 21. Конвертер MSB представляет собой не что иное как $L(h_s + h_c, 2a)$ – справочную таблицу, которая генерирует $|S_H + C_H|_P$. Общая задержка МОМА, в котором CLA используются для реализации CPA, равна

$$t_{МОМА}(n, a) = (\theta(n) + 1)t_{FA} + t_{L(h_c+h_s, a)} + t_{XOR} + t_{CPA}(a) + t_{MUX}, \quad (3.91)$$

где $\theta(n)$ обозначает минимальное количество ступеней в дереве CSA, достаточное для обработки n входных операндов. Некоторые примерные значения $\theta(n)$ перечислены в таблице 23.

Пример 3.10.1. Рассмотрим 6-модульную RNS $SP_{6,1} = \{5, 7, 9, 11, 13, 16\}$, все основные параметры которой можно найти в таблице 21. Ее динамический диапазон $P > 2^{20}$ достаточен для многих приложений цифровой обработки сигналов (Digital Signal Processing – DSP). Оценим производительность двух разных версий устройства сравнения. В таблице 25 подробно описаны характеристики всех базовых блоков, используемых для создания этих устройств сравнения, включая задержки. Отметим, что задержки обеих справочных таблиц (LUT) и МОМА суммируются дважды, потому что предполагает-

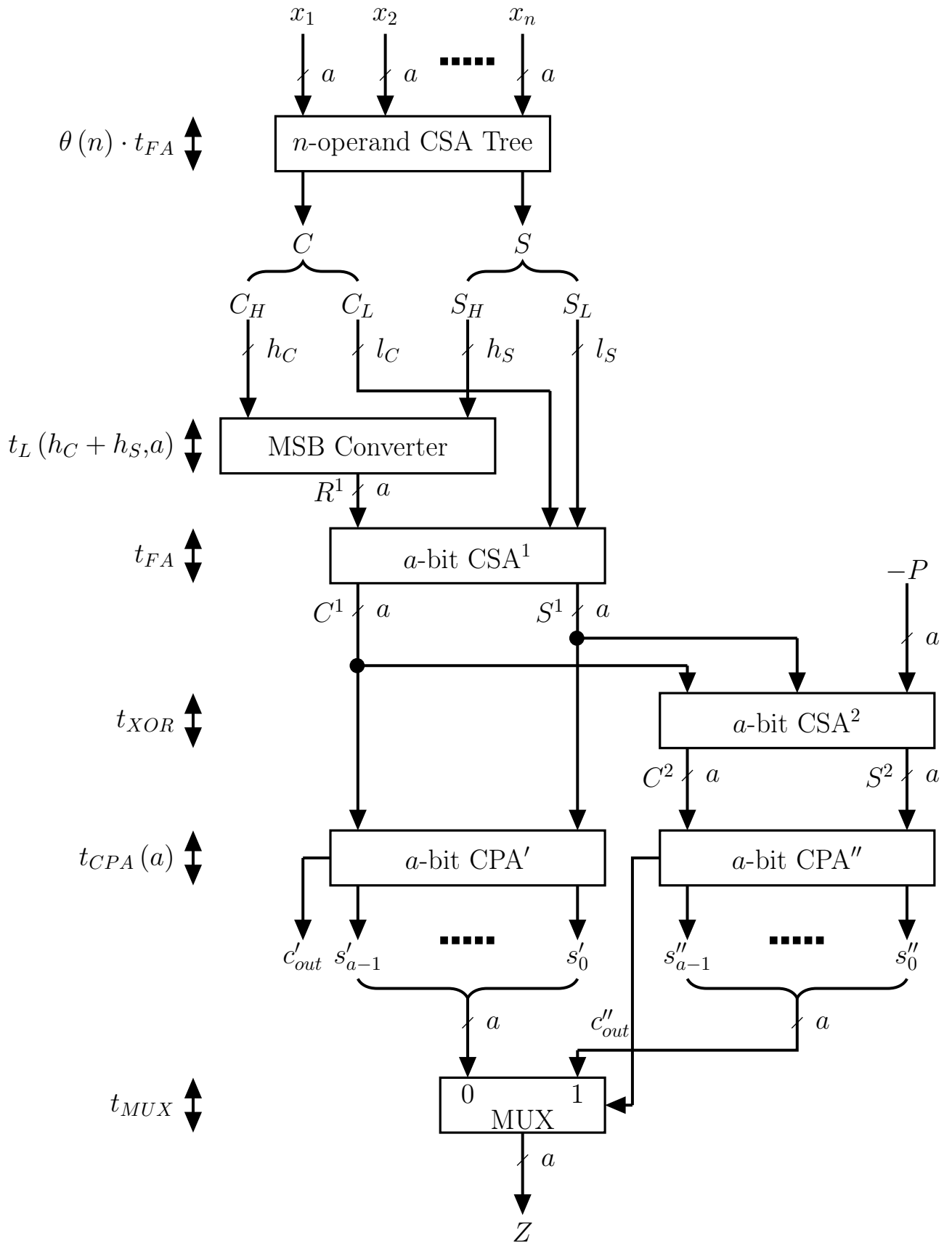


Рисунок 3.6 — Схема n -операндного MOMA $\text{mod } P$

Таблица 24 — Выходные данные n -операндного дерева CSA [309]

| r | S | C | $\max\{h_s, h_c\}$ |
|---------|---------------------|---------------------|--------------------|
| 4 | $s_a \dots s_0$ | $c_{a-1} \dots c_0$ | 6 |
| 5, 6 | $s_a \dots s_0$ | $c_a \dots c_1$ | 7 |
| 7, 8 | $s_{a+1} \dots s_0$ | $c_a \dots c_1$ | 8 |
| 9 – 12 | $s_{a+1} \dots s_0$ | $c_{a+1} \dots c_2$ | 9 |
| 13 | $s_{a+2} \dots s_0$ | $c_{a+1} \dots c_2$ | 10 |
| 14 – 18 | $s_{a+2} \dots s_0$ | $c_{a+2} \dots c_3$ | 11 |

ся, что для каждой пары сравниваемых чисел их позиционные значения и их основные функции вычисляются последовательно одной и той же схемой.

Вычислим задержки МОМА для устройств сравнения, использующих диагональную функцию и ММСF, согласно уравнению (3.91), соответственно

$$\begin{aligned} t_{\text{МОМА-DF}} &= (\theta(6) + 1) \cdot t_{FA} + t_L(7, 19) + t_{XOR} + t_{CPA}(19) + t_{MUX} \\ &= 3 \cdot 2 + 7 + 1 + 12 + 2 = 28, \end{aligned}$$

$$\begin{aligned} T_{\text{МОМА-ММСF}} &= (\theta(6) + 1) \cdot t_{FA} + t_L(7, 19) + t_{XOR} + t_{CPA}(16) + t_{MUX} \\ &= 3 \cdot 2 + 7 + 1 + 8 + 2 = 24. \end{aligned}$$

По данным, представленным в таблице 25, можно сделать вывод, что предложенное устройство сравнения чисел в RNS работает быстрее, поскольку имеет меньшую задержку (на 8Δ меньше по сравнению с устройством, использующим диагональную функцию). Кроме того, оно имеет меньшую сложность, поскольку использует меньше FA (на 27), HA (на 3), MUX 2:1 (на 3), а конечное устройство сравнения короче на 4 бита.

Таким образом, снижение задержки наблюдается при использовании наборов модулей, для которых выполняется хотя бы одно из следующих условий.

1. Каждая пара a -битных CPA МОМА работает быстрее, если $\lceil \log_2 a_{P_m} \rceil < \lceil \log_2 a_{SQ} \rceil$. Помимо набора модулей $SP_{6,1}$, рассмотренного выше в примере 3.10.1, проверка столбцов a_{SQ} и a_{P_m} таблицы 21 показывает, что этому условию также удовлетворяют несколько других наборов модулей.
2. Относительно редкие случаи, когда последнее a -битное устройство сравнения работает быстрее, представляют большой практический интерес.

Таблица 25 — Оценка сложности устройства сравнения для 6-модульной RNS $SP_{6,1} = \{5, 7, 9, 11, 13, 16\}$

| | | Диагональная функция | Функция ядра |
|----------------------|---------------|--|---|
| LUT | | $2L(3, 10), 4L(4, 19)$ | $2L(3, 16), 4L(4, 16)$ |
| МОМА | Дерево CSA | $6 \cdot 19 = 114$ FA | $6 \cdot 16 = 96$ FA |
| | Конвертер MSB | $L(7, 19)$ | $L(7, 16)$ |
| | 2 CSA | 19 FA + 19 HA | 16 FA + 16 HA |
| | 2 CPA | $2 \cdot 19 = 38$ FA | $2 \cdot 16 = 32$ FA |
| | MUX | 19 2:1 MUX | 16 2:1 MUX |
| Устройство сравнения | | $C(19 + 3) + 2$ | $C(16) + 4$ |
| Временные задержки | | $(t_L(4, 19) + t_{МОМА-DF})$ $+10 = 2(5 + 28) + 10$ $= 76$ | $2(t_L(4, 16)$ $+t_{МОМА-ММCF}) + 10 =$ $2(5 + 24) + 10 = 68$ |

Например, если $a_{P_m} \leq 24$ и $a_{SQ} > 24$, задержка уменьшается на 4Δ . В таблице 21 этому условию удовлетворяет только набор $SP_{8,1}$.

Чтобы получить максимальную выгоду от параллельной обработки данных в RNS, организованной путем использования независимых вычислительных каналов по каждому модулю p_i ($1 \leq i \leq n$), нужно чтобы модули были максимально сбалансированы, т.е. вводили аналогичную задержку и потребляли аналогичное количество ресурсов оборудования. Особенно предпочтительны наборы модулей, в которых наибольший модуль m является четным числом типа 2^q . Это связано с тем, что несмотря на то, что модуль 2^q на несколько бит больше оставшихся нечетных модулей, задержка и сложность оборудования вычислительного канала обработки данных по модулю 2^q сопоставимы с аналогичными характеристиками вычислительных каналов по нечетным модулям. Подобная ситуация наблюдается и при использовании специальных малозатратных наборов модулей вида $2^q \pm 1$ и 2^q [26]. Таким образом, выбор в качестве наибольшего модуля четного модуля 2^q выгоден для реализации эффективных устройств сравнения чисел в RNS в независимости от того используются ли специальные наборы модулей или наборы модулей общего вида. Для предлагаемых в данной работе устройств, использующих произвольные наборы модулей RNS, включающие модули, отличные от $2^q \pm 1$, экономия ресурсов оборудования составляет не менее $q(n + 1)$ FA по сравнению с известными аналогами.

3.11 Выводы по третьей главе

В третьей главе исследованы различные подходы к выполнению операций определения знака и сравнения чисел, разработаны методы и алгоритмы реализации указанных операций, позволяющие повысить производительность гомоморфных вычислений над кольцом вычетов с делителями нуля.

Установлено, что результаты операций определения знака числа и сравнения чисел, заданных над кольцом вычетов с делителями нуля, невозможно вычислить с помощью многочленов.

Разработаны два алгоритма, реализующие определение знака числа для гомоморфных вычислений над кольцом вычетов с делителями нуля, основанных на RNS с четным и нечетным диапазоном.

Представлен обзор существующих и предложены новые методы сравнения чисел для гомоморфных вычислений над кольцом вычетов с делителями нуля на основе RNS.

Методы, основанные на переводе чисел из RNS в позиционную систему счисления, являются наиболее очевидными и наименее производительными способами сравнения чисел. С целью повышения производительности разработаны методы сравнения, основанные на вычислении позиционных характеристик, таких как диагональная функция, функция ядра Акушского, функция Pirlo и Impedovo. Все вышеперечисленные позиционные характеристики, кроме функции ядра Акушского, являются монотонными, и возможна ситуация, когда разные числа имеют одинаковую позиционную характеристику. В этом случае требуется выполнение дополнительных действий для сравнения чисел, представленных в RNS. Кроме того, для получения значений указанных позиционных характеристик используется ресурсозатратная операция вычисления остатка от деления на большой модуль. Для устранения этих недостатков введено понятие модифицированной диагональной функции, которое служит теоретической основой для разработки значительно более быстрого алгоритма сравнения. Модифицированная диагональная функция (MDF) представляет собой строго возрастающую позиционную характеристику чисел, представленных в RNS, сочетающую преимущества диагональной функции и приближенного метода. Строгая монотонность MDF обеспечивает взаимнооднозначное соответствие числа и его позиционной характеристики, поэтому не возникает ситуаций, когда требу-

ется выполнение дополнительных действий для сравнения чисел. Кроме того, вместо операции нахождения остатка от деления на большое число, при вычислении MDF используются значительно более простые в реализации вычисления по модулю, равному степени числа 2.

Разработанное устройство сравнения на основе MDF и его наиболее эффективные известные аналоги, применяемые для сравнения чисел в RNS с модулями общего вида, были синтезированы для технологии 65 нм с использованием нескольких образцов наборов модулей. Согласно полученным оценкам производительности, предложенный подход обеспечивает снижение задержки на 11 – 75% (в зависимости от набора модулей) по сравнению с самыми быстрыми существующими реализациями известных методов сравнения чисел в RNS. Более того, наблюдается снижение аппаратных затрат (более чем на 41%) и значительное снижение энергопотребления, которое в ряде случаев превышает 100%. Таким образом, предложенный метод на основе MDF позволяет реализовывать наиболее эффективные на сегодняшний день устройства сравнения чисел, представленных в RNS с наборами модулей общего вида.

Особого внимания заслуживает функции ядра Акушского, свойства которой зависят от используемых при ее построении коэффициентов. Доказано, что для достижения монотонности при построении функции ядра необходимо использовать только неотрицательные коэффициенты. Показано, что уже известная диагональная функция, ранее предложенная для реализации сравнения чисел в RNS, есть не что иное, как частный случай функции ядра со всеми коэффициентами равными единице. Сформулированы условия, при которых обеспечивается минимальный диапазон функции ядра (необходимый для получения наилучших характеристик устройства сравнения чисел в RNS). Установлено, что монотонная функция ядра минимального диапазона (ММCF) имеет только один коэффициент, равный единице (соответствующий наибольшему модулю), все остальные коэффициенты равны нулю. Сформулирована и доказана теорема об условиях отсутствия критических ядер функции ядра Акушского, имеющая важное практическое значение для построения эффективных позиционных характеристик чисел, представленных в RNS. Представленное исследование позволяет сделать вывод, что функция ядра Акушского является обобщением позиционных характеристик чисел, представленных в RNS, ее изучение углубит понимание свойств позиционных характеристик и, следовательно

но, позволит разрабатывать более высокопроизводительные подходы и методы реализации операций над закодированными данными.

Глава 4. РАЗРАБОТКА И ОПТИМИЗАЦИЯ МЕТОДОВ И АЛГОРИТМОВ ОПРЕДЕЛЕНИЯ ЗНАКА И СРАВНЕНИЯ ГОМОМОРФНО ЗАКОДИРОВАННЫХ ЧИСЕЛ НАД ПОЛЕМ

Гомоморфные вычисления над полем, описанные в проекте стандарта 2018 года по гомоморфным вычислениям [238], принято делить на два класса: целочисленные и вещественные, по формату обрабатываемых цифровых данных. Соответственно задачи определения знака числа и сравнения чисел следует рассматривать над полем \mathbb{Z}_m и над полем \mathbb{R} .

4.1 Интерполяция функции знака числа над полем \mathbb{Z}_m

Теорема 4.1.1. *Если m — простое число и $m \geq 3$, то в поле $\mathbb{Z}_m[x]$ существует единственный многочлен $s(x) \in \mathbb{Z}_m[x]$, такой что $\forall x \in \mathbb{Z}_m : \text{sign}_m(x) = s(x)$. При этом $\deg s(x) = m - 2$ и*

$$s(x) = \sum_{i=1}^{m-2} a_i \cdot x^i,$$

$$\text{где } a_{2j} = 0 \text{ и } a_{2j-1} = - \sum_{i=1}^{\frac{m-1}{2}} \left(\frac{1}{i}\right)^{2j-1} + \sum_{i=\frac{m-1}{2}}^{m-1} \left(\frac{1}{i}\right)^{2j-1}.$$

Доказательство. Вычислим многочлен $s(x)$, используя интерполяционную формулу Лагранжа. Решим задачу интерполяции для функции $\text{sign}(x)$ с m узлами — всеми элементами \mathbb{Z}_m . Получим многочлен $s(x) = \sum_{i=0}^{m-1} a_i x^i$ степени $m - 1$. Заметим, что функция $\text{sign}(x)$ нечетная. Поэтому $a_{2i} = 0$, в частности, поскольку m нечетно, степень многочлена не превосходит $m - 2$.

Вычислим функции Лагранжа для выбранных узлов.

Положим $g_i(x) = \prod_{j=0, j \neq i}^{m-1} (x - j)$. Поскольку $\prod_{i=0}^{m-1} (x - j) = x^m - x$, при $i \neq 0$ имеем

$$g_i(x) = \frac{x^m - x}{x - i} = \sum_{j=1}^{m-1} \left(\frac{x}{i}\right)^j.$$

При $i = 0$ получаем равенство $g_0(x) = x^{m-1} - 1$. Поскольку многочлен $l_0(x)$ четный, его вклад в интерполяцию нулевой и он исключается из дальнейшего рассмотрения. Далее из равенства $(m-1)! = -1$ следует, что $g_i(i) = -1$. Поэтому

$$l_i(x) = \frac{g_i(x)}{g_i(i)} = - \sum_{j=1}^{m-1} \left(\frac{x}{i}\right)^j.$$

Поскольку интерполяционный многочлен содержит только нечетные степени x , то рассмотрим многочлены

$$h_i(x) = - \sum_{j=1}^{\frac{m-1}{2}} \left(\frac{x}{i}\right)^{2j-1}.$$

Тогда

$$\begin{aligned} s(x) &= \sum_{i=1}^{m-1} \text{sign}_m(i) \cdot h_i(x) = \sum_{i=1}^{\frac{m-1}{2}} h_i(x) - \sum_{i=\frac{m-1}{2}}^{m-1} h_i(x) \\ &= - \sum_{i=1}^{\frac{m-1}{2}} \sum_{j=1}^{\frac{m-1}{2}} \left(\frac{x}{i}\right)^{2j-1} + \sum_{i=\frac{m-1}{2}}^{m-1} \sum_{j=1}^{\frac{m-1}{2}} \left(\frac{x}{i}\right)^{2j-1} \\ &= - \sum_{j=1}^{\frac{m-1}{2}} \sum_{i=1}^{\frac{m-1}{2}} \left(\frac{x}{i}\right)^{2j-1} + \sum_{j=1}^{\frac{m-1}{2}} \sum_{i=\frac{m-1}{2}}^{m-1} \left(\frac{x}{i}\right)^{2j-1} \\ &= \sum_{j=1}^{\frac{m-1}{2}} x^{2j-1} \left(- \sum_{i=1}^{\frac{m-1}{2}} \left(\frac{1}{i}\right)^{2j-1} + \sum_{i=\frac{m-1}{2}}^{m-1} \left(\frac{1}{i}\right)^{2j-1} \right) \end{aligned}$$

□

4.2 Сравнение чисел над полями характеристики m

Пусть $\mathbb{Z}_{m^d} = \mathbb{Z}_m[x] / (f(x))$, где $f(x)$ – неприводимый многочлен над \mathbb{Z}_m и $\deg f(x) = d$. Зададим отображение \mathbb{Z}_{m^d} в \mathbb{Z}_m используя абсолютный след α

$$\text{Tr}_{\mathbb{Z}_{m^d}/\mathbb{Z}_m}(\alpha) = \alpha + \alpha^m + \dots + \alpha^{m^{d-1}}, \quad (4.1)$$

где $\alpha \in \mathbb{Z}_{m^d}$.

Тогда, используя Теорему 2.24 из работы [139, с. 75], определим линейное отображение $L_\beta(\alpha)$ формулой $L_\beta(\alpha) = \text{Tr}_{\mathbb{Z}_{m^d}/\mathbb{Z}_m}(\beta\alpha)$.

Теорема 4.2.1. Пусть \mathbb{Z}_{m^d} – конечное расширение конечного поля \mathbb{Z}_m . Тогда линейными отображениями из \mathbb{Z}_{m^d} в \mathbb{Z}_m являются отображения $L_\beta, \beta \in \mathbb{Z}_{m^d}$, определяемые условием $L_\beta(\alpha) = \text{Tr}_{\mathbb{Z}_{m^d}/\mathbb{Z}_m}(\beta\alpha)$ для всех $\alpha \in \mathbb{Z}_{m^d}$, и только они. При этом если β и γ – различные элементы поля \mathbb{Z}_{m^d} , то $L_\beta \neq L_\gamma$ [139, с. 75, Теорема 2.24].

Из Теоремы 4.2.1 можно сделать вывод о том, что отображения L_α могут быть использованы для сравнения чисел. Таким образом, задача сравнения чисел в целочисленных гомоморфных вычислениях сводится к задаче сравнения чисел, заданных над простым полем.

Функцию сравнения чисел над простым полем \mathbb{Z}_m определим следующим образом

$$\text{comp}_m(x, y) = \begin{cases} 1, & \text{если } x > y, \\ 0, & \text{если } x = y, \\ -1, & \text{если } x < y, \end{cases} \quad (4.2)$$

где m – простое число, а $x, y \in \mathbb{Z}_m$. В Разделе 3.2 доказано, что если m -составное число, то не существует многочлена от двух переменных над \mathbb{Z}_m , определяющего функцию сравнения чисел. В работе [199] используется интерполяционный многочлен Лагранжа от двух переменных, аппроксимирующий функцию сравнения чисел над \mathbb{Z}_m .

Метод, предложенный в работе [199], позволяет сравнивать целые числа в закодированном виде. Вычислительная сложность метода зависит от степени многочлена $s(x, y)$, меньшей либо равной $2m - 2$. В данном разделе уточнена оценка степени интерполяционного многочлена $s(x, y)$ из работы [199] и показано, что $\deg f(x, y) = m$.

4.3 Матрицы специального вида над \mathbb{Z}_m и их свойства

Рассмотрим матрицы D_x вида $D_x = x \cdot E$ над \mathbf{Z}_m , где E – единичная матрица порядка $m - 1$, m – простое число. Обратим внимание на следующие свойства подобных матриц.

Свойство 4.3.1. *Для матриц вида D_x имеют место следующие соотношения:*

1. $D_{x \cdot y} = D_x \cdot D_y$.
2. $D_{x+y} = D_x + D_y$.
3. $\forall x \in \mathbb{Z}_m^*: D_x^{-1} = \frac{1}{x} \cdot E$.
4. $\forall x \in \mathbb{Z}_m^*: \det D_x \equiv 1 \pmod{m}$.

Доказательство. Для матриц вида D_x имеют место следующие соотношения:

1. $D_{x \cdot y} = x \cdot y \cdot E = x \cdot E \cdot y \cdot E = D_x \cdot D_y$.
2. $D_{x+y} = (x + y) \cdot E = x \cdot E + y \cdot E = D_x + D_y$.
3. $\forall x \in \mathbb{Z}_m^*: D_x^{-1} = D_{\frac{1}{x}} = \frac{1}{x} \cdot E$.
4. $\forall x \in \mathbb{Z}_m^*: \det D_x = x^{m-1} \equiv 1 \pmod{m}$.

Свойство доказано. □

Лемма 4.3.1. *Если m – простое число, то $\det \bar{D} \in \{-1, 0, 1\}$ над \mathbb{Z}_m , где*

$$\bar{D} = \begin{pmatrix} D_{d_{1,1}} & D_{d_{1,2}} & \cdots & D_{d_{1,n}} \\ D_{d_{2,1}} & D_{d_{2,2}} & \cdots & D_{d_{2,n}} \\ \vdots & \vdots & \ddots & \vdots \\ D_{d_{n,1}} & D_{d_{n,2}} & \cdots & D_{d_{n,n}} \end{pmatrix}. \quad (4.3)$$

Доказательство. Используя преобразования строк матрицы 1-го типа, сохраняющих значение определителя [134, с. 80, Предложение 2] и Свойства 4.3 матрицы D_x , приведем матрицу \bar{D} к квазитреугольному виду

$$\bar{D}'_{\oplus} = \begin{pmatrix} D_{d'_{1,1}} & D_{d'_{1,2}} & \cdots & D_{d'_{1,n}} \\ D_{d'_{2,1}} & D_{d'_{2,2}} & \cdots & D_{d'_{2,n}} \\ \vdots & \vdots & \ddots & \vdots \\ D_{d'_{n,1}} & D_{d'_{n,2}} & \cdots & D_{d'_{n,n}} \end{pmatrix} \quad (4.4)$$

или

$$\bar{D}'_{\ominus} = - \begin{pmatrix} D_{d'_{1,1}} & D_{d'_{1,2}} & \cdots & D_{d'_{1,n}} \\ D_{d'_{2,1}} & D_{d'_{2,2}} & \cdots & D_{d'_{2,n}} \\ \vdots & \vdots & \ddots & \vdots \\ D_{d'_{n,1}} & D_{d'_{n,2}} & \cdots & D_{d'_{n,n}} \end{pmatrix}. \quad (4.5)$$

Используя Теорему 4 из работы [134, с. 82], вычислим значения определителей квазитреугольных матриц \bar{D}'_{\oplus} и \bar{D}'_{\ominus} , получим

$$\det \bar{D}'_{\oplus} = - \det \bar{D}'_{\ominus} = \prod_{i=1}^n \det D_{d'_{i,i}}. \quad (4.6)$$

Учитывая, что $\forall i \in \overline{1, n}$:

$$\det D_{d'_{i,i}} = \begin{cases} 1, & \text{если } d'_{i,i} \in \mathbf{Z}_m^*, \\ 0, & \text{если } d'_{i,i} \equiv 0 \pmod{m}, \end{cases} \quad (4.7)$$

$\prod_{i=1}^n \det D_{d'_{i,i}}$ может принимать два значения $\prod_{i=1}^n \det D_{d'_{i,i}} = 1$, если $\forall i \in \overline{1, n}$: $d'_{i,i} \in \mathbf{Z}_m^*$, и $\prod_{i=1}^n \det D_{d'_{i,i}} = 0$, если $\exists i \in \overline{1, n}$: $d'_{i,i} = 0$. Значит, $\det \bar{D} \in \{-1, 0, 1\}$.

Лемма доказана. \square

Лемма 4.3.2. *Если m – простое число, то $\det \bar{D}$ над \mathbb{Z}_m равен нулю тогда и только тогда, когда $\det \tilde{D}$ над \mathbb{Z}_m равен нулю, где $\tilde{D} = (d_{i,j})_{n \times n}$.*

Доказательство. Покажем, что если $\det \tilde{D} = 0$, то $\det \bar{D} = 0$. Предположим, что $\det \tilde{D} = 0$. В этом случае строки матрицы \tilde{D} линейно зависимы, следовательно, существуют такие числа α_i , одновременно не равные нулю, что выполняется условие $\sum_{i=1}^n \alpha_i \cdot \vec{d}_i = 0$. Значит будет существовать номер $j \in \overline{1, n}$: $\alpha_j \neq 0$ и $\vec{d}_j = \sum_{i=1, i \neq j}^n \alpha'_i \vec{d}_i$, где $\forall i \in \overline{1, n}$: $\alpha'_i = -\frac{\alpha_i}{\alpha_j}$ и $\forall i \in \overline{1, n}$: $\vec{d}_i = (d_{i,1}, d_{i,2}, \dots, d_{i,n})$. Обозначим через \vec{D}_i вектор, элементами которого являются блочные матрицы $D_{d_{i,j}}$: $\vec{D}_i = (D_{d_{i,1}}, D_{d_{i,2}}, \dots, D_{d_{i,n}})$.

Вычислим значение $\sum_{i=1, i \neq j}^n \alpha'_i \vec{D}_i$, используя Свойства 4.3, получим

$$\begin{aligned} \sum_{i=1, i \neq j}^n \alpha'_i \vec{D}_i &= \left(\sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,1}}, \sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,2}}, \dots, \sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,n}} \right) \\ &= \left(\sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,1}}, \sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,2}}, \dots, \sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,n}} \right) \\ &= \left(D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,1}}, D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,2}}, \dots, D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,n}} \right) \\ &= \vec{D}_j. \end{aligned} \quad (4.8)$$

Из (4.8) следует, что строки матриц \bar{D} линейно зависимы, следовательно, $\det \bar{D} = 0$.

Покажем, что если $\det \bar{D} = 0$, то $\det \tilde{D} = 0$. Предположим, что $\det \bar{D} = 0$, тогда строки матрицы \bar{D} линейно зависимы и существуют такие числа α_i , одновременно не равные нулю, что выполняется условие $\sum_{i=1}^n \alpha_i \vec{D}_i = 0$. Значит существует такое $j \neq 0$, что выполняется условие $\vec{D}_j = \sum_{i=1, i \neq j}^n \alpha'_i \vec{D}_i$.

Вычислим значение $\vec{D}_j = \sum_{i=1, i \neq j}^n \alpha'_i \vec{D}_i$, используя Свойства 4.3, получим

$$\begin{aligned}
\vec{D}_j &= \sum_{i=1, i \neq j}^n \alpha'_i \vec{D}_i \\
&= \left(\sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,1}}, \sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,2}}, \dots, \sum_{i=1, i \neq j}^n \alpha'_i \cdot D_{d_{i,n}} \right) \\
&= \left(\sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,1}}, \sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,2}}, \dots, \sum_{i=1, i \neq j}^n D_{\alpha'_i \cdot d_{i,n}} \right) \\
&= \left(D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,1}}, D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,2}}, \dots, D_{\sum_{i=1, i \neq j}^n \alpha'_i \cdot d_{i,n}} \right). \tag{4.9}
\end{aligned}$$

Из (4.9) следует, что $\vec{d}_j = \sum_{i=1, i \neq j}^n \alpha'_i \vec{d}_i$, значит строки матрицы \tilde{D} линейно зависимы и $\det \tilde{D} = 0$. Равенство выполняется в обе стороны.

Лемма доказана. □

Лемма 4.3.3. *Если m – простое число, то $\det D_V \neq 0$ над \mathbb{Z}_m , где*

$$D_V = \begin{pmatrix} D_1 & D_1^2 & \cdots & D_1^{m-1} \\ D_2 & D_2^2 & \cdots & D_2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m-1} & D_{(m-1)^2} & \cdots & D_{(m-1)^{m-1}} \end{pmatrix}. \tag{4.10}$$

Доказательство. Рассмотрим матрицу Вандермонда V , равную

$$V = \begin{pmatrix} 1 & 1^2 & \cdots & 1^{m-1} \\ 2 & 2^2 & \cdots & 2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ m-1 & (m-1)^2 & \cdots & (m-1)^{m-1} \end{pmatrix}. \tag{4.11}$$

Определитель матрицы V равен

$$\det V = \prod_{1 \leq j < i \leq m} (i - j) = \prod_{i=1}^{m-1} i^{m-i} \not\equiv 0 \pmod{m}. \tag{4.12}$$

Из Леммы 4.3.2 следует, что $\det D_V \neq 0$.

Лемма доказана. □

4.4 Полиномиальная интерполяция функции сравнения чисел над простым полем

Теорема 4.4.1. *Если m – простое число и $m \geq 3$, то в поле $\mathbb{Z}_m[x]$ существует многочлен $c(x, y) \in \mathbb{Z}_m[x, y]$ степени $\deg c(x, y) \leq 2m - 2$, такой что $\forall x, y \in \mathbb{Z}_m$: $\text{compr}_m(x, y) \equiv c(x, y) \pmod{m}$. $c(x, y)$ имеет вид*

$$c(x, y) = x^{m-1} - y^{m-1} + \sum_{t=1}^{m-1} \sum_{k=1}^{m-1} b_{t,k} \cdot x^t \cdot y^k, \quad (4.13)$$

где $\forall 1 \leq t < k \leq m - 1$:

$$\begin{aligned} b_{t,k} &= \sum_{1 \leq i < j \leq m-1} (i^{m-1-k} \cdot j^{m-1-t} - i^{m-1-t} \cdot j^{m-1-k}), \\ b_{k,t} &= -b_{t,k}, \\ b(x, y) &= \sum_{t=1}^{m-1} \sum_{k=1}^{m-1} b_{t,k} \cdot x^t \cdot y^k, \end{aligned} \quad (4.14)$$

причем для $c(x, y) \in \mathbb{Z}_m[x, y]$ многочлен $b(x, y) \in \mathbb{Z}_m[x, y]$ определяется единственным образом.

Доказательство. Пусть существует многочлен $c(x, y) \in \mathbb{Z}_m[x, y]$ такой, что $\forall x, y \in \mathbb{Z}_m$: $\text{compr}_m(x, y) = c(x, y)$, тогда его можно представить в виде

$$c(x, y) = \alpha + \eta(x) + \xi(y) + b(x, y), \quad (4.15)$$

где $\alpha \in \mathbb{Z}_m$, $\eta(x) \in \mathbb{Z}_m[x]$ и $x|\eta(x)$, $\xi(y) \in \mathbb{Z}_m[y]$ и $y|\xi(y)$, $b(x, y) \in \mathbb{Z}_m[x, y]$ и $(x \cdot y) | b(x, y)$.

Так как $\text{compr}_m(0, 0) = 0$, то $c(0, 0) = \alpha = 0$. Учитывая, что $\forall x \in \mathbb{Z}_m^*$: $\text{compr}_m(x, 0) = 1$, $c(x, 0) = \eta(x) \equiv 1 \pmod{m}$, следовательно, $\forall a \in \mathbb{Z}_m$: $(x - a) | (\eta(x) - 1)$, значит, $(x^{m-1} - 1) | (\eta(x) - 1)$. Выбирая в качестве $\eta(x)$ многочлен наименьшей степени, удовлетворяющий условиям $x|\eta(x)$ и $(x^{m-1} - 1) | (\eta(x) - 1)$, получим $\eta(x) = x^{m-1}$.

Учитывая, что $\forall y \in \mathbb{Z}_m^*$: $\text{comp}_m(0, y) = -1$, $c(0, y) = \xi(y) \equiv -1 \pmod{m}$. Рассуждая аналогично $\eta(x)$, получим $\xi(y) = -y^{m-1}$. Таким образом, $c(x, y)$ примет следующий вид

$$\begin{aligned} c(x, y) &= x^{m-1} - y^{m-1} + b(x, y) \\ &= x^{m-1} - y^{m-1} + \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} b_{i,j} x^i \cdot y^j. \end{aligned} \quad (4.16)$$

Покажем, что $\forall x, y \in \mathbb{Z}_m^*$ существует единственный многочлен $b(x, y) = \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} b_{i,j} x^i \cdot y^j$, удовлетворяющий условию (4.16).

$$H \times \begin{pmatrix} b_{1,1} \\ b_{1,2} \\ \vdots \\ b_{m-1,m-1} \end{pmatrix} = \begin{pmatrix} \text{comp}_m(1, 1) \\ \text{comp}_m(1, 2) \\ \vdots \\ \text{comp}_m(m-1, m-1) \end{pmatrix}, \quad (4.17)$$

где

$$H = \begin{pmatrix} 1^1 \cdot 1^1 & 1^1 \cdot 1^2 & \dots & 1^{m-1} \cdot 1^{m-1} \\ 1^1 \cdot 2^1 & 1 \cdot 2^2 & \dots & 1^{m-1} \cdot 2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ (m-1)^1 (m-1)^1 & (m-1)^1 (m-1)^2 & \dots & (m-1)^{m-1} (m-1)^{m-1} \end{pmatrix}. \quad (4.18)$$

Матрицу H можно представить в следующем виде

$$H = D_V \times \begin{pmatrix} V & D_0 & \dots & D_0 \\ D_0 & V & \dots & D_0 \\ \vdots & \vdots & \ddots & \vdots \\ D_0 & D_0 & \dots & V \end{pmatrix}. \quad (4.19)$$

Используя Теорему 6 из работы [134, с. 85], вычислим значение определителя матрицы H , получим

$$\begin{aligned} \det H &= \det D_V \cdot \det \begin{pmatrix} V & D_0 & \dots & D_0 \\ D_0 & V & \dots & D_0 \\ \vdots & \vdots & \ddots & \vdots \\ D_0 & D_0 & \dots & V \end{pmatrix} \\ &= \det D_V \cdot (\det V)^{m-1} \\ &\equiv \det D_V \pmod{m}. \end{aligned} \quad (4.20)$$

Из Леммы 4.3.3 следует, что $\det H \not\equiv 0 \pmod{m}$, значит существует единственное решение соответствующего матричного уравнения и многочлен $b(x, y) \in \mathbb{Z}_m^*[x, y]$ определяется единственным образом.

Вычислим многочлен $b(x, y)$, используя интерполяционный многочлен Лагранжа от двух переменных. Вычислим базисные многочлены.

$$L_{i,j}(x, y) = \frac{g_i(x)}{g_i(i)} \cdot \frac{g_j(y)}{g_j(j)}, \quad (4.21)$$

где $g_i(x) = \sum_{t=1}^{m-1} i^{m-1-t} \cdot x^t$. Вычислим значение $g_i(i)$, получим

$$g_i(i) = \sum_{t=1}^{m-1} i^{m-1-t} \cdot i^t = \sum_{t=1}^{m-1} i^{m-1} \equiv m-1 \pmod{m}. \quad (4.22)$$

Следовательно,

$$\begin{aligned} L_{i,j}(x, y) &\equiv g_i(x) \cdot g_j(y) \\ &\equiv \left(\sum_{t=1}^{m-1} i^{m-1-t} \cdot x^t \right) \left(\sum_{k=1}^{m-1} i^{m-1-k} \cdot y^k \right) \pmod{m}. \end{aligned} \quad (4.23)$$

Используя равенство из работы [137, с. 57, формула (2.34)],

$$\left(\sum_{k=1}^{m-1} a_k \right) \left(\sum_{t=1}^{m-1} b_t \right) = (m-1) \sum_{k=1}^{m-1} a_k \cdot b_k - \sum_{1 \leq t < k \leq m-1} (a_k - a_t)(b_k - b_t), \quad (4.24)$$

вычислим значение $L_{i,j}(x, y)$. Учитывая, что $a_k = i^{m-1-k} \cdot x^k$, а $b_k = j^{m-1-k} \cdot y^k$, получим

$$\begin{aligned} L_{i,j}(x, y) &= (m-1) \sum_{k=1}^{m-1} i^{m-1-k} \cdot x^k \cdot j^{m-1-k} \cdot y^k \\ &\quad - \sum_{1 \leq t < k \leq m-1} (i^{m-1-k} \cdot x^k - i^{m-1-t} \cdot x^t) (j^{m-1-k} \cdot y^k - j^{m-1-t} \cdot y^t). \end{aligned} \quad (4.25)$$

Так как $\forall i, j, x, y \in \mathbb{Z}_m^*$ выполняются следующие равенства

$$i^{m-1-k} \cdot x^k \cdot j^{m-1-k} \cdot y^k = \left(\frac{x \cdot y}{i \cdot j} \right)^k, \quad (4.26)$$

$$i^{m-1-k} \cdot x^k - i^{m-1-t} \cdot x^t = x^t \cdot i^{-k} (x^{k-t} - i^{k-t}), \quad (4.27)$$

$$j^{m-1-k} \cdot y^k - j^{m-1-t} \cdot y^t = y^t \cdot j^{-k} (y^{k-t} - j^{k-t}), \quad (4.28)$$

то $L_{i,j}(x, y)$ преобразуется к следующему виду

$$L_{i,j}(x, y) = (m-1) \sum_{k=1}^{m-1} \left(\frac{xy}{ij} \right)^k - \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^{-k} (x^{k-t} - i^{k-t}) (y^{k-t} - j^{k-t}). \quad (4.29)$$

Вычислим разность $L_{i,j}(x, y) - L_{j,i}(x, y)$

$$L_{i,j}(x, y) - L_{j,i}(x, y) = \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^{-k} (x^{k-t} - j^{k-t}) (y^{k-t} - i^{k-t}) - \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^{-k} (x^{k-t} - i^{k-t}) (y^{k-t} - j^{k-t}).$$

Используя сочетательный закон [137, с. 48, формула (2.16)], преобразуем $L_{i,j}(x, y) - L_{j,i}(x, y)$ к следующему виду

$$L_{i,j}(x, y) - L_{j,i}(x, y) = \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^k ((x^{k-t} - j^{k-t}) (y^{k-t} - i^{k-t}) - (x^{k-t} - i^{k-t}) (y^{k-t} - j^{k-t})). \quad (4.30)$$

Учитывая, что

$$(x^{k-t} - j^{k-t}) (y^{k-t} - i^{k-t}) - (x^{k-t} - i^{k-t}) (y^{k-t} - j^{k-t}) = (x^{k-t} - y^{k-t}) \cdot (j^{k-t} - i^{k-t}),$$

получим

$$L_{i,j}(x, y) - L_{j,i}(x, y) = \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^{-k} (x^{k-t} - y^{k-t}) (j^{k-t} - i^{k-t}).$$

Используя интерполяционную функцию Лагранжа, найдем многочлен $b(x, y)$

$$b(x, y) = - \sum_{1 \leq i < j \leq m-1} (L_{i,j}(x, y) - L_{j,i}(x, y)) = - \sum_{1 \leq i < j \leq m-1} \sum_{1 \leq t < k \leq m-1} (xy)^t (ij)^{-k} (x^{k-t} - y^{k-t}) (j^{k-t} - i^{k-t}).$$

Используя правило изменения порядка суммирования и обобщающий сочетательный закон [137, с. 53, формула (2.27)], получим

$$b(x, y) = - \sum_{1 \leq t < k \leq m-1} (xy)^t (x^{k-t} - y^{k-t}) \sum_{1 \leq i < j \leq m-1} (ij)^{-k} (j^{k-t} - i^{k-t}).$$

Таким образом, можно сделать вывод, что если $1 \leq t < k \leq m - 1$, то $b_{t,k} = \sum_{1 \leq i < j \leq m-1} (ij)^{-k} (j^{k-t} - i^{k-t})$ и $b_{k,t} = -b_{t,k}$.

Вычислим $\forall 1 \leq t < k \leq m - 1$ значение $b_{t,k}$, получим

$$\begin{aligned} b_{t,k} &= \sum_{1 \leq i < j \leq m-1} (ij)^{-k} (j^{k-t} - i^{k-t}) \\ &= \sum_{1 \leq i < j \leq m-1} (i^{m-1-k} j^{m-1-t} - i^{m-1-t} j^{m-1-k}). \end{aligned} \quad (4.31)$$

Теорема доказана. □

4.5 Коэффициенты многочлена, определенного для интерполяционной функции сравнения чисел над простым полем

В предыдущем разделе с помощью интерполяционной формулы Лагранжа было получено представление интерполяционной функции сравнения чисел. В данном разделе рассмотрены свойства коэффициентов многочлена $b(x, y)$.

Свойство 4.5.1. *Для коэффициентов многочлена $b(x, y)$, определенного для интерполяционной функции сравнения чисел $s(x, y)$, имеют место следующие соотношения.*

1. Если $t + k$ – четное число, то $b_{t,k} = 0$.
2. Если t – четное, а k – нечетное, то

$$b_{t,k} = 2 \cdot \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i} j^{m-1-t}. \quad (4.32)$$

3. Если t – нечетное, а k – четное, то

$$b_{t,k} = -2 \cdot \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t}. \quad (4.33)$$

Доказательство. Используя Теорему 4.4.1, вычислим значения коэффициентов $b_{t,k}$

$$\begin{aligned} b_{t,k} &= \sum_{1 \leq i < j \leq m-1} (i^{m-1-k} \cdot j^{m-1-t} - i^{m-1-t} \cdot j^{m-1-k}) \\ &= \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} (i^{m-1-k} \cdot j^{m-1-t} - i^{m-1-t} \cdot j^{m-1-k}). \end{aligned} \quad (4.34)$$

Используя сочетательный закон [137, с. 48, формула (2.16)], получим

$$b_{t,k} = \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} i^{m-1-k} \cdot j^{m-1-t} - \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} i^{m-1-t} \cdot j^{m-1-k}. \quad (4.35)$$

Заменим j на $m - j$, получим

$$b_{t,k} = \sum_{i=1}^{m-2} \sum_{j=1}^{m-i-1} i^{m-1-k} \cdot (m-j)^{m-1-t} - \sum_{i=1}^{m-2} \sum_{j=1}^{m-i-1} i^{m-1-t} \cdot (m-j)^{m-1-k}. \quad (4.36)$$

Так как

$$(m-j)^{m-1-t} \equiv (-1)^{m-1-t} j^{m-1-t} \pmod{m}$$

и

$$(m-j)^{m-1-k} \equiv (-1)^{m-1-k} j^{m-1-k} \pmod{m},$$

то

$$b_{t,k} = (-1)^{m-1-t} \sum_{i=1}^{m-2} \sum_{j=1}^{m-i-1} i^{m-1-k} \cdot j^{m-1-t} - (-1)^{m-1-k} \sum_{i=1}^{m-2} \sum_{j=1}^{m-i-1} i^{m-1-t} \cdot j^{m-1-k}.$$

Используя распределительный закон [137, с. 48, формула (2.16)], преобразуем суммы к следующему виду

$$\begin{aligned} b_{t,k} &= (-1)^{m-1-t} \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t} \\ &\quad - (-1)^{m-1-k} \sum_{i=1}^{m-2} i^{m-1-t} \cdot \sum_{j=1}^{m-i-1} j^{m-1-k}. \end{aligned} \quad (4.37)$$

Учитывая, что

$$\sum_{i=1}^{m-2} i^{m-1-t} \sum_{j=1}^{m-i-1} j^{m-1-k} = \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t}, \quad (4.38)$$

получим

$$b_{t,k} = \left((-1)^{m-1-t} - (-1)^{m-1-k} \right) \cdot \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t}. \quad (4.39)$$

Рассмотрим четыре случая.

Случай 1. Если t – четное и k – четное, то $m - 1 - t$ – четное и $m - 1 - k$ – четное, следовательно,

$$b_{t,k} = 0. \quad (4.40)$$

Случай 2. Если t – четное и k – нечетное, то $m - 1 - t$ – четное и $m - 1 - k$ – нечетное, следовательно,

$$b_{t,k} = 2 \cdot \sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t}. \quad (4.41)$$

Случай 3. Если t – нечетное и k – нечетное, то $m - 1 - t$ – нечетное и $m - 1 - k$ – нечетное, следовательно,

$$b_{t,k} = 0. \quad (4.42)$$

Случай 4. Если t – нечетное и k – четное, то $m - 1 - t$ – нечетное и $m - 1 - k$ – четное, следовательно,

$$b_{t,k} = -2 \cdot \sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t}. \quad (4.43)$$

Объединяя случаи 1 и 3, можно сделать вывод о том, что если $t + k$ – четное число, то $b_{t,k} = 0$.

Свойство доказано. □

Свойство 4.5.2. Для коэффициентов многочлена $b(x, y)$ выполняется следующее соотношение

$$b_{m-1,2k+1} = \begin{cases} 2, & \text{если } k = 0, \\ 0, & \text{иначе.} \end{cases} \quad (4.44)$$

Доказательство. Так как по условию Теоремы 4.4.1 m – простое число и $m \geq 3$, то m – нечетное число, следовательно, $m - 1$ – четное число. Число вида $2k + 1$ всегда нечетное, значит значение $b_{m-1,2k+1}$ вычисляется согласно Свойству 4.5.1 (Случай 2)

$$b_{m-1,2k+1} = 2 \cdot \sum_{i=1}^{m-2} i^{m-2-2k} \cdot \sum_{j=1}^{m-i-1} j^0. \quad (4.45)$$

Учитывая, что $\sum_{j=1}^{m-i-1} j^0 = \sum_{j=1}^{m-i-1} 1 = m - i - 1$, получим

$$\begin{aligned} b_{m-1,2k+1} &= 2 \sum_{i=1}^{m-2} i^{m-2-2k} (m - i - 1) \\ &\equiv -2 \sum_{i=1}^{m-2} i^{m-1-2k} - 2 \sum_{i=1}^{m-2} i^{m-2-2k} \pmod{m}. \end{aligned} \quad (4.46)$$

Представим

$$\begin{aligned} \sum_{i=1}^{m-2} i^{m-1-2k} &= -(m-1)^{m-1-2k} + \sum_{i=1}^{m-1} i^{m-1-2k} \\ &\equiv -(-1)^{m-1-2k} + \sum_{i=1}^{m-1} i^{m-1-2k} \pmod{m}, \end{aligned} \quad (4.47)$$

$$\begin{aligned} \sum_{i=1}^{m-2} i^{m-2-2k} &= -(m-1)^{m-2-2k} + \sum_{i=1}^{m-1} i^{m-2-2k} \\ &\equiv -(-1)^{m-2-2k} + \sum_{i=1}^{m-1} i^{m-2-2k} \pmod{m}, \end{aligned} \quad (4.48)$$

получим

$$\begin{aligned} b_{m-1,2k+1} &= 2 \left((-1)^{m-1-2k} + (-1)^{m-2-2k} \right) \\ &\quad - 2 \sum_{i=1}^{m-1} i^{m-1-2k} - 2 \sum_{i=1}^{m-1} i^{m-2-2k} \\ &= -2 \sum_{i=1}^{m-1} i^{m-1-2k} - 2 \sum_{i=1}^{m-1} i^{m-2-2k}. \end{aligned} \quad (4.49)$$

Так как $\forall k = \overline{0, \frac{m-3}{2}}$: $\gcd(m-2k-1, m) = 1$ и $\forall k = \overline{1, \frac{m-1}{2}}$: $\gcd(m-2k, m) = 1$, то, используя формулу Бернулли из работы [137, с. 314, формула (6.78)], вычислим значение $\left| \sum_{i=1}^{m-1} i^{m-1-2k} \right|_m$, получим

$$\begin{aligned} \left| \sum_{i=1}^{m-1} i^{m-1-2k} \right|_m &= \left| \frac{1}{m-2k} \sum_{s=0}^{m-1-2k} \binom{m-2k}{s} B_s \cdot m^{m-2k-s} \right|_m \\ &= \begin{cases} \left| \binom{m}{m-1} B_{m-1} \right|_m, & \text{если } k = 0, \\ 0, & \text{иначе,} \end{cases} \end{aligned} \quad (4.50)$$

где B_s – числа Бернулли.

Вычислим значение $\left| \sum_{i=1}^{m-1} i^{m-1-2k} \right|_m$ в случае, если $k = 0$, получим

$$\left| \sum_{i=1}^{m-1} i^{m-1} \right|_m \equiv \left| \sum_{i=1}^{m-1} 1 \right|_m \equiv -1 \pmod{m}, \quad (4.51)$$

следовательно,

$$b_{m-1,2k+1} = \begin{cases} 2, & \text{если } k = 0, \\ 0, & \text{иначе.} \end{cases} \quad (4.52)$$

Свойство доказано. \square

Свойство 4.5.3. Если $t + k > m$, то $b_{t,k} = 0$.

Доказательство. Воспользуемся формулой Бернулли [137, с. 314, формула (6.78)] и вычислим значение выражения

$$\sum_{i=1}^{m-2} i^{m-1-k} \cdot \sum_{j=1}^{m-i-1} j^{m-1-t}, \quad (4.53)$$

получим

$$\sum_{j=1}^{m-i-1} j^{m-1-t} = \frac{1}{m-t} \sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s (m-i)^{m-t-s}. \quad (4.54)$$

Так как $\forall t \in \mathbf{Z}_m^*$ и $t \neq m-1$: $\gcd(m-t, m) = 1$, то

$$\frac{1}{m-t} \sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s (m-i)^{m-t-s} \equiv -\frac{1}{t} \sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s (-i)^{m-t-s} \pmod{m}. \quad (4.55)$$

Следовательно,

$$\sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t} \equiv \sum_{i=1}^{m-2} i^{m-1-k} \left(-\frac{1}{t} \sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s (-i)^{m-t-s} \right). \quad (4.56)$$

Используя распределительный закон [137, с. 48, формула (2.15)], получим

$$\begin{aligned} \sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t} &= \frac{(-1)^{m-1-t}}{t} \sum_{i=1}^{m-2} i^{m-1-k} \sum_{s=0}^{m-1-t} (-1)^s \binom{m-t}{s} B_s i^{m-t-s} \\ &= \frac{(-1)^{m-1-t}}{t} \sum_{i=1}^{m-2} i^{m-1-k} \sum_{s=0}^{m-1-t} (-1)^s \binom{m-t}{s} B_s t^{m-t-s} \\ &= \frac{(-1)^{m-1-t}}{t} \sum_{s=0}^{m-1-t} (-1)^s \binom{m-t}{s} B_s \sum_{i=1}^{m-2} i^{2m-t-k-s-1}. \end{aligned}$$

Представим $\sum_{i=1}^{m-2} i^{2m-t-k-s-1}$ в следующем виде

$$\sum_{i=1}^{m-2} i^{2m-t-k-s-1} = -(m-1)^{2m-t-k-s-1} \sum_{i=1}^{m-1} i^{2m-t-k-s-1}. \quad (4.57)$$

Так как по условию $t+k > m$, то $1 \leq 2m-t-k-s < m$, следовательно, $\gcd(2m-t-k-s, m) = 1$ и

$$\begin{aligned} \left| \sum_{i=1}^{m-2} i^{2m-t-k-s-1} \right|_m &\equiv -(-1)^{2m-t-k-s-1} \\ &+ \frac{1}{2m-t-k-s} \sum_{j=0}^{2m-t-k-s-1} \binom{2m-t-k-s}{j} B_j m^{2m-t-k-s-j} \\ &\equiv (-1)^{2m-t-k-s} \pmod{m}. \end{aligned} \quad (4.58)$$

Следовательно,

$$\begin{aligned} \sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t} &\equiv \frac{(-1)^{m-1-t}}{t} \sum_{s=0}^{m-1-t} (-1)^s \binom{m-t}{s} B_s (-1)^{2m-t-k-s} \\ &\equiv \frac{(-1)^{m-1-t} (-1)^{2m-t-k}}{t} \sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s. \end{aligned} \quad (4.59)$$

Учитывая, что $\forall t \in \mathbb{Z}_m^*$ и $t \neq m-1$: $m-1-t > 0$, используя формулу (2.79) из работы [137, с. 314], получим

$$\sum_{s=0}^{m-1-t} \binom{m-t}{s} B_s = 0. \quad (4.60)$$

Таким образом, согласно Свойству 4.5.1, если $t+k$ – четное число, то $b_{t,k} = 0$. Если t – четное, k – нечетное, $t+k > m$ и $t \neq m-1$, то

$$b_{t,k} = 2 \sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t} = 2 \cdot 0 = 0. \quad (4.61)$$

Если t – нечетное, k – четное, $t+k > m$ и $t \neq m-1$, то

$$b_{t,k} = -2 \sum_{i=1}^{m-2} i^{m-1-k} \sum_{j=1}^{m-i-1} j^{m-1-t} = -2 \cdot 0 = 0. \quad (4.62)$$

Из Свойства 4.5.2 следует, что если $t = m-1$, $t+k$ – нечетное число и $t+k > m$, то $b_{t,k} = 0$. Следовательно, если $t+k > m$, то $b_{t,k} = 0$.

Свойство доказано. \square

Из Свойства 4.5.3 следует, что степень многочлена $c(x, y) \leq m$. С другой стороны, согласно Свойству 4.5.2, $b_{m-1,1} = 2$, следовательно, коэффициент при $x^{m-1}y$ не равен нулю, значит $\deg c(x, y) = m$. Таким образом, получена уточненная оценка степени интерполяционного многочлена из работы [199]. Степень многочлена в вышеупомянутой работе не превышает значения $2m - 2$, тогда как полученная оценка дает точное значение степени многочлена, равное m .

Пример 4.5.1. Вычислим значение многочлена $c(x, y)$ над \mathbb{Z}_5 .

Используя Свойство 4.5.1, вычислим значения коэффициентов $b_{t,k}$, получим

$$\begin{aligned} b_{2,1} &= \left| 2 \cdot \sum_{i=1}^3 i^3 \cdot \sum_{j=1}^{4-i} j^2 \right|_m \\ &= |2 \cdot (1 + 4 + 9 + 8 \cdot 1 + 8 \cdot 4 + 27 \cdot 1)|_5 = 2, \end{aligned} \quad (4.63)$$

$$\begin{aligned} b_{3,2} &= \left| -2 \cdot \sum_{i=1}^3 i^2 \cdot \sum_{j=1}^{4-i} j^2 \right|_m \\ &= |-2 \cdot (1 + 2 + 3 + 4 \cdot 1 + 4 \cdot 2 + 9 \cdot 1)|_5 = 1. \end{aligned} \quad (4.64)$$

Согласно Свойству 4.5.2, $b_{4,1} = 2$. Используя Теорему 4.4.1, вычислим значения коэффициентов $b_{1,2}$, $b_{2,3}$ и $b_{1,4}$, получим

$$b_{1,2} = -b_{2,1} = 3 \pmod{5}, \quad (4.65)$$

$$b_{2,3} = -b_{3,2} = 4 \pmod{5}, \quad (4.66)$$

$$b_{1,4} = -b_{4,1} = 3 \pmod{5}. \quad (4.67)$$

Таким образом, многочлен $c(x, y)$ над \mathbb{Z}_5 имеет следующий вид

$$c(x, y) = x^4 - y^4 + 3 \cdot x \cdot y^2 + 3 \cdot x \cdot y^4 + 4 \cdot x^2 \cdot y^3 + x^3 \cdot y^2 + 2 \cdot x^4 \cdot y + 2 \cdot x^2 \cdot y. \quad (4.68)$$

Рассмотрим значения, которые принимает многочлен $c(x, y)$ над \mathbb{Z}_5 . Вычисленные значения приведены в таблице 26. Значения в таблице 26 демонстрируют, что многочлен $c(x, y)$ построен корректно.

Таблица 26 — Значения многочлена $c(x, y)$ над \mathbb{Z}_5

| $c(x, y)$ | | y | | | | |
|-----------|---|-----|----|----|----|----|
| | | 0 | 1 | 2 | 3 | 4 |
| x | 0 | 0 | -1 | -1 | -1 | -1 |
| | 1 | 1 | 0 | -1 | -1 | -1 |
| | 2 | 1 | 1 | 0 | -1 | -1 |
| | 3 | 1 | 1 | 1 | 0 | -1 |
| | 4 | 1 | 1 | 1 | 1 | 0 |

4.6 Аппроксимация функции определения знака закодированного числа над полем \mathbb{R}

Рассмотрим функцию определения знака числа, заданную над множеством действительных чисел

$$\text{sign}(x) = \begin{cases} 1, & \text{если } x > 0, \\ 0, & \text{если } x = 0, \\ -1, & \text{если } x < 0. \end{cases} \quad (4.69)$$

Так как функция $\text{sign}(x)$ непрерывна на множестве $x \in (-\infty, 0) \cup (0, +\infty)$, то задача аппроксимации рассматривается на двух интервалах $[-1, -\epsilon] \cup [\epsilon, 1]$.

Учитывая, что функцию $\text{sign}(x)$ невозможно аппроксимировать с помощью многочлена в окрестности точки $x = 0$ с точностью меньше 0.5, Вонга и др. [165] вводят численно устойчивый метод аппроксимации $\text{sign}(x)$ в терминах ряда Фурье по целевому интервалу. По сравнению с классическими приближениями функций многочленами, аналитический метод имеет три основных преимущества: во-первых, он не выходит за пределы интервала к ∞ (лучшая численная устойчивость); во-вторых, коэффициенты Фурье малы и, наконец, в-третьих, ряд сходится равномерно к функции на любом интервале, не содержащем точек разрыва. В случае функции $\text{sign}(x)$ наличие разрыва на обоих графиках означает, что ряд Фурье плохо сходится в окрестности точки $x = 0$. Предлагаемая последовательность Фурье содержит только синусоидальные члены нечетных степеней, аппроксимирующие функцию $\text{sign}(x)$, и обозначается

следующим образом

$$\text{sign}(x) = K_1 \cdot \sum_{k=0}^{\infty} \frac{\sin(2\pi \cdot (2k+1) \cdot x)}{2k+1} + K_2. \quad (4.70)$$

В работе [165] авторы строят гомоморфную модель нейронной сети, используя предложенное тригонометрическое приближение, т.е. последовательность Фурье. Показано, что гомоморфные когнитивные модели способны поглощать не менее десяти процентов относительных ошибок без какого-либо влияния на общую точность. Другими словами, можно немного увеличить ошибку аппроксимации не жертвуя точностью модели и, следовательно, уменьшить вычислительную сложность.

Chialva и Dooms в работе 2018 года [180] предложили аппроксимировать функцию знака числа, используя $\text{th}(k \cdot x) = \frac{e^{k \cdot x} - e^{-k \cdot x}}{e^{k \cdot x} + e^{-k \cdot x}} \cong \text{sign}(x)$ при достаточно больших $k > 0$. Чтобы эффективно вычислить $\text{th}(k \cdot x)$, авторы неоднократно применяют формулу двойного угла $\text{th}(2x) = \frac{2 \text{th} x}{1 + \text{th}^2 x} \approx \frac{2x}{1+x^2}$, где обратная операция заменяется минимаксным приближенным многочленом маленькой степени. Этот подход можно интерпретировать как композицию отображений f , которая является полиномиальной аппроксимацией $\frac{2x}{1+x^2}$. Однако, из-за особенностей метода ядра, ошибка между полиномом f и $\text{sign}(x)$ не может быть ниже определенного предела, даже если увеличивать k до ∞ .

Cheon и др. [178] предложили новый метод сравнения (определения знака), основанный на аппроксимации композицией полиномов. Отметим, что операции сравнения и определения знака тесно связаны, т.к. сравнить числа можно вычитанием одного из другого и определением знака результата вычитания. Суть подхода состоит в том, чтобы аппроксимировать $\text{sign}(x)$ композицией многочленов $f \circ \dots \circ f \circ g \circ \dots \circ g$, где f, g систематически проектируются так, что требуется небольшое количество композиций для достижения приемлемой точности.

Структурирование аппроксимирующих многочленов по принципу композиции некоторых многочленов постоянной степени дает существенное преимущество в вычислительной сложности. Если многочлен f степени $\Theta(2^\alpha)$ выражается через композицию $f_0 \circ f_0 \circ \dots \circ f_0$ некоторого многочлена постоянной степени f_0 , то f может быть вычислен с линейной сложностью $\Theta(\alpha)$.

Пусть $f(x) = \frac{x^2}{x^2 + (1-x)^2}$, тогда каждую итерацию [294] можно интерпретировать как оценку $f(a)$ и $f(b) = 1 - f(a)$ для $0 \leq a, b \leq 1$ соответственно.

Действительно, d итераций соответствуют композиции из d полиномов f , обозначенных $f^{(d)}$. Другими словами, составной полином по существу соответствует итерационному алгоритму, который многократно вычисляет f , т.е.

$$\frac{f_n^{(d)} + 1}{2} \approx \frac{\text{sign}(a - b) + 1}{2}. \quad (4.71)$$

Согласно [178], для построения многочлена (одной переменной), обеспечивающего хорошее приближение функции знака числа, вместо функции сравнения (двух переменных) используются композиции двух семейств многочленов f и g . Первое семейство многочленов задается выражением

$$f_n(x) = x \cdot \sum_{i=0}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot (1 - x^2)^i. \quad (4.72)$$

Функция $f_n(x)$ характеризуется тремя основными свойствами:

Свойство 1. Так как $\text{sign}(x)$ – нечетная функция, то $f_n(x)$ – нечетная функция.

Свойство 2. $f(1) = 1$ и $f(-1) = -1$ гарантируют сходимость к ± 1 .

Свойство 3. $f'(x) = c \cdot (1 - x)^n \cdot (1 + x)^n$ для некоторой константы $c > 0$.

Строго говоря, для каждого $n \geq 1$ многочлен степени $2n + 1$, удовлетворяющий трем указанным свойствам, однозначно определяет f_n . На рисунке 4.1 показаны многочлены $f_n(x)$, сгенерированные для $n = 1, 2, 3, 4$

$$f_1(x) = -\frac{1}{2} \cdot x^3 + \frac{3}{2} \cdot x \quad (4.73)$$

$$f_2(x) = \frac{3}{8} \cdot x^5 - \frac{5}{4} \cdot x^3 + \frac{15}{8} \cdot x \quad (4.74)$$

$$f_3(x) = -\frac{5}{16} \cdot x^7 + \frac{21}{16} \cdot x^5 - \frac{35}{16} \cdot x^3 + \frac{35}{16} \cdot x \quad (4.75)$$

$$f_4(x) = \frac{35}{128} \cdot x^9 - \frac{45}{32} \cdot x^7 + \frac{189}{64} \cdot x^5 - \frac{105}{32} \cdot x^3 + \frac{315}{128} \cdot x \quad (4.76)$$

Вместе с тем, многочлены $g_n(x)$ используются для ускорения сходимости и удовлетворяют следующим свойствам: поскольку $\text{sign}(x)$ и $f_n(x)$ – нечетные функции, $g_n(x)$ также должна быть нечетной функцией. Более того, для $g_n(x)$, $\exists 0 < \delta < 1$ такая что $x < g(x) \leq 1$ для всех $x \in (0, \delta]$, и $g([\delta, 1]) \subseteq [1 - \tau, 1]$, где τ обозначает ошибку аппроксимации. На рисунке 4.2 показаны многочлены

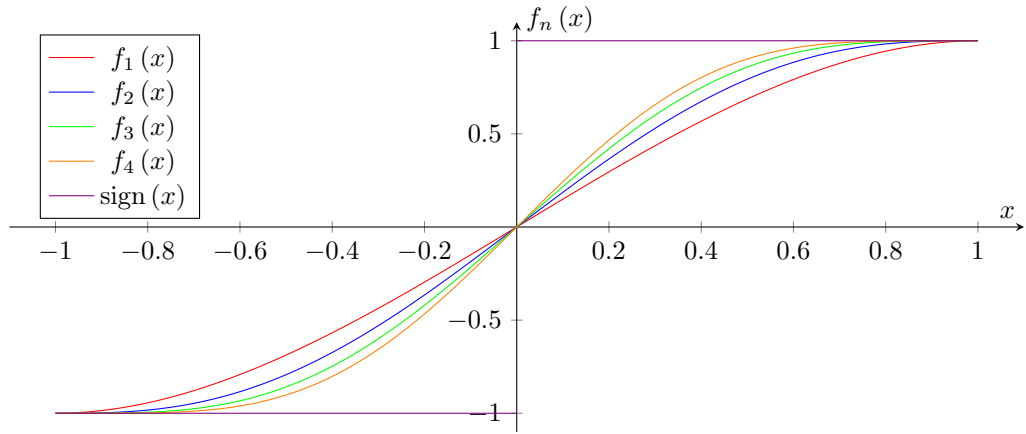


Рисунок 4.1 — Аппроксимация функции знака числа с использованием многочленов $f_n(x)$, где $n = 1, 2, 3, 4$

$g_n(x)$, сгенерированные для $n = 1, 2, 3, 4$.

$$g_1(x) = \frac{-1359}{2^{10}} \cdot x^3 + \frac{2126}{2^{10}} \cdot x, \quad (4.77)$$

$$g_2(x) = \frac{3796}{2^{10}} \cdot x^5 - \frac{6108}{2^{10}} \cdot x^3 + \frac{3334}{2^{10}} \cdot x, \quad (4.78)$$

$$g_3(x) = -\frac{12860}{2^{10}} \cdot x^7 + \frac{25614}{2^{10}} \cdot x^5 - \frac{16577}{2^{10}} \cdot x^3 + \frac{4589}{2^{10}} \cdot x, \quad (4.79)$$

$$g_4(x) = \frac{46623}{2^{10}} \cdot x^9 - \frac{113492}{2^{10}} \cdot x^7 + \frac{97015}{2^{10}} \cdot x^5 - \frac{34974}{2^{10}} \cdot x^3 + \frac{5850}{2^{10}} \cdot x. \quad (4.80)$$

Аппроксимация композицией многочленов имеет множество преимуществ перед использованием независимых многочленов. Например, согласно рисунку 4.2, погрешность в окрестности нуля близка к единице. За счет композиции многочленов погрешность в окрестности нуля и τ значительно уменьшаются. С алгоритмической точки зрения, для $\alpha > 0$ и $0 \leq \epsilon \leq 1$, при аппроксимации композицией многочленов, в отличие от ранних и итерационных подходов, генерируются многочлены $f(x)$, которые (α, ϵ) -близки к $\text{sign}(x)$ над $[-1, 1]$, т.е.

$$\|f(x) - \text{sign}(x)\|_{\infty, [-1, -\epsilon] \cup [\epsilon, 1]} \leq 2^{-\alpha}, \quad (4.81)$$

где $\|\bullet\|_{\infty, D}$ обозначает норму бесконечности над областью определения $D = [-1, -\epsilon] \cup [\epsilon, 1]$. Это означает, что ошибка аппроксимации гарантировано ниже $\tau = 2^{-\alpha}$ для $\epsilon \leq |x| \leq 1$.

Кроме того, применение композиций уменьшает промежуток $[-\epsilon, \epsilon]$. При использовании многочлена g , этот промежуток короче, чем при использовании композиции двух идентичных многочленов f . Скорость вычислений при этом

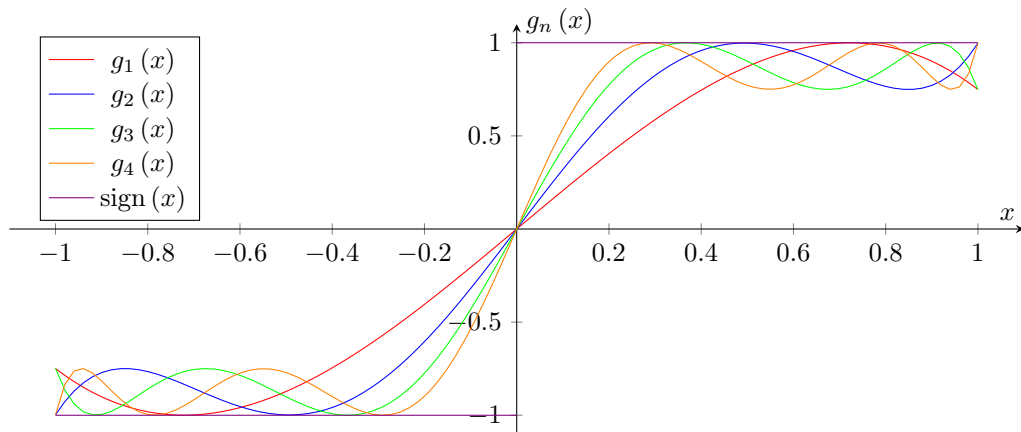


Рисунок 4.2 — Аппроксимация функции знака числа с использованием многочленов $g_n(x)$, где $n = 1, 2, 3, 4$

остается той же [59], но при использовании композиции двух идентичных многочленов увеличивается ошибка на промежутке $[-1, -\epsilon] \cup [\epsilon, 1]$. Т.е. композиция многочленов $f_1^{d_f} \circ g_1^{d_g}$ обеспечивает лучшее приближение, чем многочлен $f_1^{d_f+d_g}$. Наконец, композиция двух многочленов степени n требует меньше времени для вычисления по сравнению с многочленом степени $n \cdot n$. На рисунке 4.3 показаны композиции многочленов $f_n(g_n(x))$, сгенерированные для $n = 1, 2, 3, 4$. После

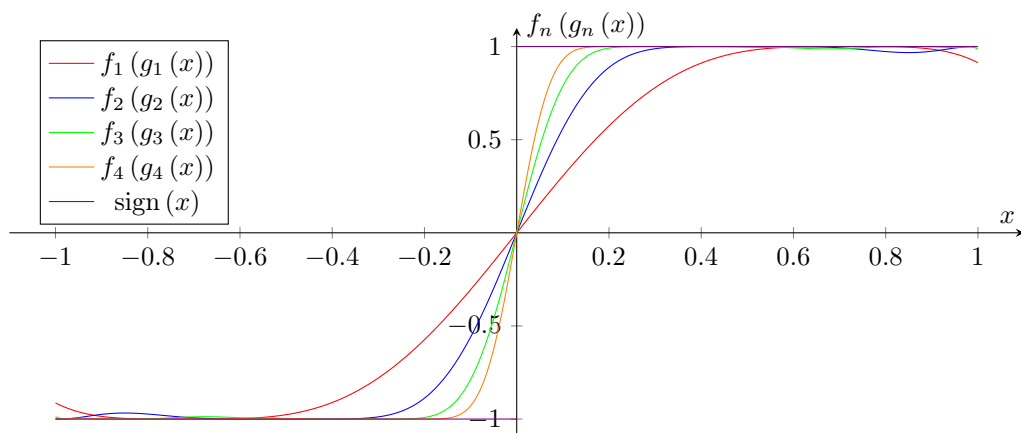


Рисунок 4.3 — Аппроксимация функции знака числа с использованием композиции многочленов $f_n(g_n(x))$ для $n = 1, 2, 3, 4$

того, как Cheon и др. [178] представили первое сравнение закодированных чисел с помощью аппроксимации композицией многочленов, появилось несколько методов приближения по этой схеме. Несмотря на некоторые улучшения, операция гомоморфного сравнения по-прежнему требует большого количества времени, поэтому необходимы дополнительные исследования для повышения производительности при практическом использовании.

Lee и др. [282] предложили реализовать операцию гомоморфного сравнения композицией многочленов, состоящей из минимаксных аппроксимирующих многочленов. Данный метод использует модифицированный алгоритм Ремеза для нахождения точных минимаксных аппроксимирующих многочленов [323]. В общем, алгоритм вычисляет многочлены в зависимости от входных параметров. Вычисленная композиция многочленов минимизирует не скалярные умножения и глубину всех композиций, составленных из минимаксных аппроксимирующих многочленов. Для решения этой задачи авторы используют алгоритмы динамического программирования с полиномиальным временем, поскольку поиск методом перебора подразумевает экспоненциальное время.

В работе [59] Бабенко и др. исследуют методы ускорения выполнения операции гомоморфного сравнения закодированных текстов. Как и в работе [178], авторы используют вычитание чисел и композицию двух семейств многочленов $f_n(x)$ и $g_n(x)$ для аппроксимации функции знака числа. Композиция обеспечивает точность приближения функции знака числа в 1.98 раза лучше, чем современные теоретические оценки для многочленов $f_n(x)$. Так же в работе показано, что теоретические результаты для $g_n(x)$ неприменимы для $n \leq 4$. Кроме того установлено, что если приложению не нужно учитывать знак отклонения аппроксимации по сравнению с истинным значением, то следует использовать композицию $f_n(g_n(x))$; в противном случае следует использовать $f_n(f_n(x))$.

Vajard и др. [251] указывают на то, что гомоморфная реализация функции знака числа является основным затруднением при построении сохраняющих конфиденциальность моделей векторных машин (как в искусственных нейронных сетях). Здесь авторы предлагают новый подход для улучшения аппроксимации функции знака числа путем введения метода, основанного на алгоритме поиска корня Newton-Raphson для функции $r(x) = 1 - \frac{1}{x^2}$, то есть многочлена $f_1(x)$ из [178]. Функция f выражается как $f(x) = \frac{x}{2} \cdot (3 - x^2)$, что позволяет получить аппроксимацию функции знака числа путем итеративного вычисления $f(x)$, которое будет близко к ± 1 в зависимости от знака x . Авторы предлагают способ рандомизации итераций, который увеличивает как безопасность, так и скорость сходимости, а также снижает вычислительную сложность.

В таблице 27 представлены основные характеристики современных методов гомоморфного сравнения чисел с фиксированной точностью. На рисунке 4.4 представлены аппроксимации многочленами девятой степени, полученными и

Таблица 27 — Основные характеристики методов гомоморфного сравнения чисел. Ошибка аппроксимации $\epsilon = 2^{-\alpha}$

| Метод сравнения | Требуемая степень | Сложность | (α, ϵ) -близость | $[a, b]$ |
|---------------------------|---------------------------------|---|--------------------------------|-------------------------------|
| [164, 184, 273, 316, 339] | $\Theta(2^\alpha)$ | $\Theta(2^{\frac{\alpha}{2}})$ | Нет | — |
| [290, 295, 346] | $\Theta(2^\alpha)$ | $\Theta(\sqrt{\alpha} \cdot 2^{\frac{\alpha}{2}})$ | Нет | $[0, 1]$ |
| [294] | $\Theta(\alpha \cdot 2^\alpha)$ | $\Theta(\alpha \cdot \log \alpha)$ | Нет | $[\frac{1}{2}, \frac{3}{2}]$ |
| [165] | $\Theta(2^\alpha)$ | $\Theta(2^{\frac{\alpha}{2}})$ | Нет | $[-\frac{1}{2}, \frac{1}{2}]$ |
| [178] | $\Theta(2^\alpha)$ | $\Theta(\log \frac{1}{\epsilon}) + \Theta(\log \alpha)$ | Да | $[0, 1]$ |

выполненными для реализации гомоморфного сравнения чисел с фиксированной точностью. В случае итеративного подхода [294] не генерируется многочлен как таковой; вместо этого метод аппроксимирует функцию $\text{sign}(x)$ путем итеративного вычисления следующего тождества

$$\text{comp}(a, b) = \lim_{k \rightarrow \infty} \frac{a^k}{a^k + b^k} = \begin{cases} 1, & \text{если } a > b, \\ \frac{1}{2}, & \text{если } a = b, \\ 0, & \text{если } a < b, \end{cases} \quad (4.82)$$

где $a, b \in [\frac{1}{2}, \frac{3}{2}]$. Поскольку для получения результата сравнения с небольшой погрешностью требуется $k = 2^d$, авторы предложили итеративно вычислять $a \leftarrow \frac{a^2}{a^2+b^2}$ и $b \leftarrow \frac{b^2}{a^2+b^2}$. После d итераций вычисляется приближенное значение $\frac{a^{2^d}}{a^{2^d}+b^{2^d}} \cong \text{comp}(a, b)$. Аппроксимации композицией многочленов становятся

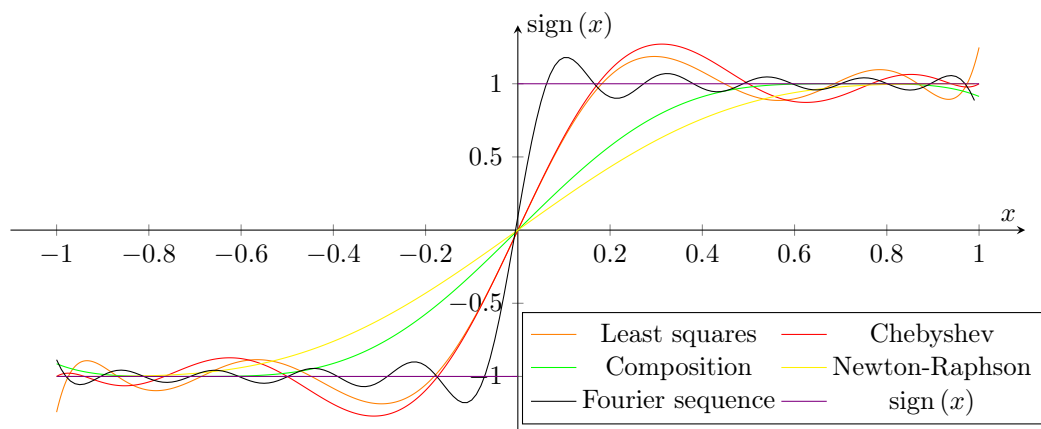


Рисунок 4.4 — Полиномиальная аппроксимация функции знака гомоморфно закодированных чисел с фиксированной точностью современными методами лучшей альтернативой определению знака закодированных чисел. Подобный

подход способствует построению эффективной и точной модели машинного обучения, сохраняющей конфиденциальность. Тщательно структурированные аппроксимирующие многочлены обеспечивают значительные преимущества в вычислительной сложности по сравнению со стандартными современными подходами. Композиции многочленов требуют меньшего времени выполнения, обеспечивают лучшую аппроксимацию, и уменьшают ошибку аппроксимации в окрестности точки $x = 0$, т.е. промежуток $[-\epsilon, \epsilon]$ уменьшается.

4.7 Об оценке точности полиномиальной аппроксимации функции определения знака закодированного числа над полем \mathbb{R}

Для оценки точности полиномиальной аппроксимации функции знака числа над действительным полем предварительно докажем связь между $f_n(x)$ и многочленами Бернштейна.

Лемма 4.7.1. *Если $1 \leq d \leq 2n + 1$, то*

$$\sum_{j=0}^d \cdot (-1)^j \binom{2n+1}{d-j} \cdot \binom{2n+j+1-d}{j} = 0. \quad (4.83)$$

Доказательство. Используя свойство симметрии $\binom{2n+1}{d-j} = \binom{2n+1}{2n+1+j-d}$, получим

$$\binom{2n+1}{d-j} \cdot \binom{2n+j+1-d}{j} = \binom{2n+1}{2n+1+j-d} \cdot \binom{2n+j+1-d}{j}. \quad (4.84)$$

Согласно триномиальному варианту свойства $\binom{r}{m} \cdot \binom{m}{k} = \binom{r}{k} \cdot \binom{r-k}{m-k}$, тогда

$$\binom{2n+1}{2n+1+j-d} \cdot \binom{2n+j+1-d}{j} = \binom{2n+1}{j} \cdot \binom{2n+1-j}{2n+1-d}. \quad (4.85)$$

Учитывая, что

$$\begin{aligned} \binom{2n+1}{j} \cdot \binom{2n+1-j}{2n+1-d} &= \frac{(2n+1)!}{(2n+1-j)! \cdot j!} \cdot \frac{(2n+1-j)!}{(2n+1-d)! \cdot (d-j)!} \\ &= \frac{(2n+1)!}{(2n+1-d)! \cdot d!} \cdot \frac{d!}{j! \cdot (d-j)!} \\ &= \binom{2n+1}{d} \cdot \binom{d}{j}. \end{aligned} \quad (4.86)$$

Подставляя (4.86) в формулу (4.83), получим

$$\begin{aligned} \sum_{j=0}^d (-1)^j \cdot \binom{2n+1}{d-j} \cdot \binom{2n+j+1-d}{j} &= \\ \sum_{j=0}^d (-1)^j \cdot \binom{2n+1}{d} \cdot \binom{d}{j} &= \binom{2n+1}{d} \cdot \sum_{j=0}^d (-1)^j \cdot \binom{d}{j} \end{aligned} \quad (4.87)$$

Так как $d \geq 1$ и $\sum_{j=0}^d (-1)^j \cdot \binom{d}{j} = (1 + (-1))^d = 0^d = 0$, то

$$\sum_{j=0}^d (-1)^j \cdot \binom{2n+1}{d-j} \cdot \binom{2n+j+1-d}{j} = 0. \quad (4.88)$$

Лемма доказана. □

Лемма 4.7.2. Если $0 \leq d \leq n$, то

$$\sum_{k=d}^n \binom{n+k}{k} \cdot \binom{k}{d} = \binom{n+d}{d} \cdot \binom{2n+1}{n-d}. \quad (4.89)$$

Доказательство. Преобразуем значение $\binom{n+k}{k} \cdot \binom{k}{d}$, получим

$$\begin{aligned} \binom{n+k}{k} \cdot \binom{k}{d} &= \frac{(n+k)!}{n! \cdot k!} \cdot \frac{k!}{(k-d)! \cdot d!} \\ &= \frac{(n+d)!}{n! \cdot d!} \cdot \frac{(n+k)!}{(k-d)! \cdot (n+d)!} \\ &= \binom{n+d}{d} \cdot \binom{n+k}{n+d}. \end{aligned} \quad (4.90)$$

Следовательно,

$$\begin{aligned} \sum_{k=d}^n \binom{n+d}{d} \cdot \binom{n+k}{n+d} &= \binom{n+d}{d} \cdot \sum_{k=d}^n \binom{n+k}{n+d} \\ &= \binom{n+d}{d} \cdot \sum_{k=0}^{n-d} \binom{n+k+d}{n+d}. \end{aligned} \quad (4.91)$$

Используем свойство суммирования по обоим индексам, получим $\sum_{k=0}^{n-d} \binom{n+k+d}{n+d} = \binom{2n+1}{n-d}$, тогда

$$\sum_{k=d}^n \binom{n+k}{k} \cdot \binom{k}{d} = \binom{n+d}{d} \cdot \binom{2n+1}{n-d}. \quad (4.92)$$

Лемма доказана. □

Лемма 4.7.3. Если $n + 1 \leq d \leq 2n$, то

$$\sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} = (-1)^n \cdot \binom{2n+1}{d} \cdot \binom{d-1}{n}.$$

Доказательство. Заменяем $t + d - n = k$, получим

$$\begin{aligned} & \sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} \\ &= \sum_{t=0}^n (-1)^{t-n} \cdot \binom{2n+1}{n-t} \cdot \binom{n+t+1}{t+d-n}. \end{aligned} \quad (4.93)$$

Так как

$$\begin{aligned} \binom{2n+1}{n-t} \cdot \binom{n+t+1}{t+d-n} &= \frac{(2n+1)!}{(n-t)! \cdot (n+t+1)!} \cdot \frac{(n+t+1)!}{(t+d-n)! \cdot (2n-d+1)!} \\ &= \frac{(2n+1)!}{(2n-d+1)! \cdot d!} \cdot \frac{d!}{(t+d-n)! (n-t)!} = \binom{2n+1}{d} \cdot \binom{d}{n-t}, \end{aligned}$$

то

$$\begin{aligned} & \sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} \\ &= \sum_{t=0}^n (-1)^{t-n} \cdot \binom{2n+1}{d} \cdot \binom{d}{n-t} \\ &= (-1)^n \cdot \binom{2n+1}{d} \cdot \sum_{t=0}^n (-1)^t \cdot \binom{d}{n-t}. \end{aligned} \quad (4.94)$$

Используя свойство верхнего обращения $\binom{d}{n-t} = (-1)^{n-t} \cdot \binom{n-t-d-1}{n-t}$, получим

$$\begin{aligned} & \sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} \\ &= \binom{2n+1}{d} \cdot \sum_{t=0}^n \binom{n-d-1-t}{n-t}. \end{aligned} \quad (4.95)$$

Заменяя $i = n - t$, получим

$$\begin{aligned} & \sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} \\ &= \binom{2n+1}{d} \cdot \sum_{t=0}^n \binom{i-d-1}{i}. \end{aligned} \quad (4.96)$$

Используя формулу суммирования по обоим индексам, получим $\sum_{t=0}^n \binom{i-d-1}{i} = \binom{n-d}{n}$. Согласно свойству верхнего обращения $\binom{n-d}{n} = (-1)^n \cdot \binom{d-1}{n}$, тогда

$$\sum_{k=d-n}^d (-1)^{k-d} \cdot \binom{2n+1}{d-k} \cdot \binom{2n+k+1-d}{k} = (-1)^n \cdot \binom{2n+1}{d} \cdot \binom{d-1}{n}.$$

Лемма доказана. □

Лемма 4.7.4. Если $n+1 \leq d \leq 2n$, то

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} \\ = (-1)^n \cdot \binom{2n+1}{d} \cdot \binom{d-1}{n}. \end{aligned} \quad (4.97)$$

Доказательство. Используя триномиальный вариант, получим

$$\binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} = \binom{2n+1}{k} \cdot \binom{2n+1-k}{2n+1-d}. \quad (4.98)$$

Обращая внимание на то, что

$$\begin{aligned} \binom{2n+1}{k} \cdot \binom{2n+1-k}{2n+1-d} &= \frac{(2n+1)!}{k! \cdot (2n+1-k)!} \cdot \frac{(2n+1-k)!}{(d-k)! \cdot (2n+1-d)!} \\ &= \frac{(2n+1)!}{(2n+1-d)! \cdot d!} \cdot \frac{d!}{(d-k)! \cdot k!} = \binom{2n+1}{d} \cdot \binom{d}{k}, \end{aligned}$$

получим

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} \\ = \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{d} \cdot \binom{d}{k} \\ = \binom{2n+1}{d} \cdot \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{d}{k}. \end{aligned} \quad (4.99)$$

Согласно свойству верхнего обращения $\binom{d}{k} = (-1)^k \cdot \binom{k-d-1}{k}$, тогда

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} \\ = (-1)^{d+1} \cdot \binom{2n+1}{d} \cdot \sum_{k=0}^{d-n-1} \binom{k-d-1}{k}. \end{aligned} \quad (4.100)$$

Согласно свойству суммирования по обоим индексам $\sum_{k=0}^{d-n-1} \binom{k-d-1}{k} = \binom{-1-n}{d-n-1}$.
Используя свойство верхнего обращения, получим $\binom{-1-n}{d-n-1} = (-1)^{d-n-1} \cdot \binom{d-1}{d-n-1}$.
Применяя свойство симметрии, получим $\binom{d-1}{d-n-1} = \binom{d-1}{n}$, тогда

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} \\ = (-1)^{2d-n} \cdot \binom{2n+1}{d} \cdot \binom{d-1}{n}. \end{aligned} \quad (4.101)$$

Так как $(-1)^{2d-n} = (-1)^n$, то

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{k+1-d} \cdot \binom{2n+1}{2n+k+1-d} \cdot \binom{2n+k+1-d}{k} \\ = (-1)^n \cdot \binom{2n+1}{d} \cdot \binom{d-1}{n}. \end{aligned} \quad (4.102)$$

Лемма доказана. \square

Теорема 4.7.1. $\forall n \in \mathbb{N}: f_n(x) = B_{2n+1}\left(\frac{x+1}{2}\right)$, где $B_n(x)$ – многочлен Бернштейна,

$$B_n(x) = \sum_{i=0}^n \phi\left(\frac{i}{n}\right) \cdot \binom{n}{i} \cdot x^i (1-x)^{n-i}, \quad (4.103)$$

где

$$\phi(x) = \begin{cases} -1, & \text{если } x < \frac{1}{2}, \\ 0, & \text{если } x = 0, \\ 1, & \text{если } x > \frac{1}{2}, \end{cases}$$

$$\text{а } f_n(x) = \sum_{i=0}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot x (1-x^2)^i.$$

Доказательство. Вычислим $B_{2n+1}\left(\frac{x+1}{2}\right)$. Учитывая, что $\forall i \in \overline{0, n}: \phi\left(\frac{i}{2n+1}\right) = -1$, а $\forall i \in \overline{n+1, 2n+1}: \phi\left(\frac{i}{2n+1}\right) = 1$, получим

$$\begin{aligned} B_{2n+1}\left(\frac{x+1}{2}\right) &= \sum_{i=n+1}^{2n+1} \binom{2n+1}{i} \left(\frac{x+1}{2}\right)^i \left(\frac{1-x}{2}\right)^{2n+1-i} \\ &\quad - \sum_{i=0}^n \binom{2n+1}{i} \left(\frac{x+1}{2}\right)^i \left(\frac{1-x}{2}\right)^{2n+1-i}. \end{aligned} \quad (4.104)$$

Сделаем замену $t = i - n - 1$, получим

$$\begin{aligned} & \sum_{i=n+1}^{2n+1} \binom{2n+1}{i} \left(\frac{x+1}{2}\right)^i \left(\frac{1-x}{2}\right)^{2n+1-i} \\ &= \sum_{t=0}^n \binom{2n+1}{t+n+1} \left(\frac{x+1}{2}\right)^{n+t+1} \left(\frac{1-x}{2}\right)^{n-t}. \end{aligned} \quad (4.105)$$

Если $x = 1 + 2y$, то $\frac{x+1}{2} = 1 + y$, а $\frac{1-x}{2} = -y$, и выражение (4.104) преобразуется к виду

$$\begin{aligned} B(y) &= B_{2n+1}(1+2y) \\ &= \sum_{i=0}^n \binom{2n+1}{n+i+1} (1+y)^{n+i+1} (-y)^{n-i} \\ &\quad - \sum_{i=0}^n \binom{2n+1}{i} (1+y)^i (-y)^{2n+1-i}. \end{aligned} \quad (4.106)$$

Учитывая, что $(-1)^{n-i} = (-1)^{n+i}$ и $(-1)^{2n+1-i} = -(-1)^i$, то

$$\begin{aligned} B(y) &= \sum_{i=0}^n \binom{2n+1}{i+n+1} y^{n-i} (1+y)^{n+i+1} (-1)^{n+i} \\ &\quad + \sum_{i=0}^n \binom{2n+1}{i} y^{2n+1-i} (1+y)^i (-1)^i. \end{aligned} \quad (4.107)$$

Заметим, что $(1+y)^{n+i+1} = \sum_{k=0}^{n+i+1} \binom{n+i+1}{k} y^k$ и $(1+y)^i = \sum_{k=0}^i \binom{i}{k} y^k$,

тогда

$$\begin{aligned} B(y) &= \sum_{i=0}^n (-1)^{n+i} \binom{2n+1}{n+i+1} \sum_{k=0}^{n+i+1} \binom{n+i+1}{k} y^{n+k-i} \\ &\quad + \sum_{i=0}^n (-1)^i \binom{2n+1}{i} \sum_{k=0}^i \binom{i}{k} y^{2n+1+k-i} \\ &= \sum_{d=0}^{2n+1} b_d \cdot y^d, \end{aligned}$$

где $b_d \in \mathbb{R}$ – коэффициенты многочлена $B(y)$.

Рассмотри четыре случая.

Случай 1. Если $d = 0$, то

$$b_0 = (-1)^{2n} \binom{2n+1}{2n+1} \binom{2n+1}{0} = 1. \quad (4.108)$$

Случай 2. Если $1 \leq d \leq n$, то

$$\begin{aligned} b_d &= \sum_{k=0}^d (-1)^{2n+k-d} \binom{2n+k+1-d}{k} \binom{2n+1}{2n+k+1-d} \\ &= (-1)^{2n-d} \sum_{k=0}^d (-1)^k \binom{2n+1}{2n+k+1-d} \binom{2n+k+1-d}{k}. \end{aligned} \quad (4.109)$$

Используя свойство симметрии $\binom{2n+1}{2n+k+1-d} = \binom{2n+1}{d-k}$, следовательно

$$b_d = (-1)^{2n-d} \sum_{k=0}^d (-1)^k \binom{2n+1}{d-k} \binom{2n+k+1-d}{k}. \quad (4.110)$$

Согласно Леммы 4.7.1 $\sum_{k=0}^d (-1)^k \binom{2n+1}{d-k} \binom{2n+k+1-d}{k} = 0$, значит $b_d = 0$.

Случай 3. Если $n+1 \leq d \leq 2n$, то

$$\begin{aligned} b_d &= \sum_{k=d-n}^d (-1)^{2n+k-d} \binom{2n+1}{2n+k+1-d} \binom{2n+k+1-d}{k} \\ &+ \sum_{k=0}^{d-n-1} (-1)^{2n+k+1-d} \binom{2n+1}{2n+k+1-d} \binom{2n+k+1-d}{k}. \end{aligned} \quad (4.111)$$

Из Леммы 4.7.3 и свойства симметрии следует, что

$$\begin{aligned} \sum_{k=d-n}^d (-1)^{2n+k-d} \binom{2n+1}{2n+k+1-d} \binom{2n+k+1-d}{k} \\ = (-1)^n \binom{2n+1}{d} \binom{d-1}{n}. \end{aligned} \quad (4.112)$$

Согласно Лемме 4.7.4,

$$\begin{aligned} \sum_{k=0}^{d-n-1} (-1)^{2n+k+1-d} \binom{2n+1}{2n+k+1-d} \binom{2n+k+1-d}{k} \\ = (-1)^n \binom{2n+1}{d} \binom{d-1}{n}. \end{aligned} \quad (4.113)$$

Подставляя (4.112) и (4.113) в (4.111), получим

$$b_d = 2(-1)^n \binom{2n+1}{d} \binom{d-1}{n}. \quad (4.114)$$

Случай 4. Если $d = 2n + 1$, то

$$b_{2n+1} = \sum_{i=0}^n (-1)^{n+i} \binom{2n+1}{n+i+1} \binom{n+i+1}{n+i+1} + \sum_{i=0}^n (-1)^i \binom{2n+1}{i} \binom{i}{i}. \quad (4.115)$$

Учитывая, что $\binom{n+i+1}{n+i+1} = \binom{i}{i} = 1$, получим

$$b_{2n+1} = \sum_{i=0}^n (-1)^{n+i} \binom{2n+1}{n+i+1} + \sum_{i=0}^n (-1)^i \binom{2n+1}{i}. \quad (4.116)$$

Согласно свойству симметрии $\binom{2n+1}{n+i+1} = \binom{2n+1}{n-i}$. Используя формулу верхнего обращения, получим $\binom{2n+1}{n-i} = (-1)^{n-i} \binom{-n-i-2}{n-i}$ и $\binom{2n+1}{i} = (-1)^i \binom{i-2n-2}{i}$, тогда

$$b_{2n+1} = \sum_{i=0}^n \binom{-n-i-2}{n-i} + \sum_{i=0}^n \binom{i-2n-2}{i}. \quad (4.117)$$

Заменяя $j = n - i$, получим

$$\begin{aligned} b_{2n+1} &= \sum_{j=0}^n \binom{j-2n-2}{j} + \sum_{i=0}^n \binom{i-2n-2}{i} \\ &= 2 \sum_{i=0}^n \binom{i-2n-2}{i}. \end{aligned} \quad (4.118)$$

Согласно свойству суммирования по обоим индексам $\sum_{i=0}^n \binom{i-2n-2}{i} = \binom{-n-1}{n}$. Применяя формулу верхнего обращения, получим $\binom{-n-1}{n} = (-1)^n \binom{2n}{n}$.

Так как при $d = 2n + 1$ выполняется равенство

$$\binom{2n+1}{d} \binom{d-1}{n} = \binom{2n+1}{2n+1} \binom{2n}{n} = \binom{2n}{n}, \quad (4.119)$$

случаи 3 и 4 можно объединить, то есть, если $n + 1 \leq d \leq 2n + 1$, то $b_d = 2(-1)^n \binom{2n+1}{d} \binom{d-1}{n}$. Значит коэффициенты $B(y)$ задаются следующим образом

$$b_d = \begin{cases} 1, & \text{если } d = 0, \\ 0, & \text{если } 1 \leq d \leq n, \\ (-1)^n 2 \binom{d-1}{n} \binom{2n+1}{d}, & \text{если } n + 1 \leq d \leq 2n + 1. \end{cases} \quad (4.120)$$

Рассмотрим многочлен $f_n(x)$. Подставим $z = \frac{1+x}{2}$ в тождество [137, Eq. 5.138]

$$(1-z)^n \sum_{k=0}^n \binom{n+k}{k} z^k = 1 + \frac{1-2z}{2-2z} \sum_{k=1}^n \binom{2k}{k} (z(1-z))^k, \quad (4.121)$$

получим

$$\left(\frac{1-x}{2}\right)^n \sum_{k=0}^n \binom{n+k}{k} \left(\frac{x+1}{2}\right)^k = 1 - \frac{x}{1-x} \sum_{k=1}^n \frac{1}{4^k} \binom{2k}{k} (1-x^2)^k. \quad (4.122)$$

Следовательно,

$$f_n(x) = 1 - \frac{1}{2^n} (1-x)^{n+1} \sum_{k=0}^n \binom{n+k}{k} \left(\frac{1+x}{2}\right)^k. \quad (4.123)$$

Сделаем замену $x = 1 + 2y$, получим

$$f_n(1+2y) = 1 + (-1)^n 2y^{n+1} \sum_{k=0}^n \binom{n+k}{k} (1+y)^k. \quad (4.124)$$

Так как $(1+y)^k = \sum_{i=0}^k \binom{k}{i} y^i$, то

$$\begin{aligned} f_n(1+2y) &= 1 + (-1)^n 2y^{n+1} \sum_{k=0}^n \binom{n+k}{k} \sum_{i=0}^k \binom{k}{i} y^i \\ &= 1 + (-1)^n 2y^{n+1} \sum_{d=0}^n y^d \sum_{k=d}^n \binom{n+k}{k} \binom{k}{d}. \end{aligned} \quad (4.125)$$

Воспользуемся Леммой 4.7.2, тогда выражение $f_n(1+2y)$ преобразуется к виду

$$\begin{aligned} f_n(1+2y) &= 1 + (-1)^n 2y^{n+1} \sum_{d=0}^n y^d \binom{n+d}{d} \binom{2n+1}{n-d} \\ &= 1 + \sum_{d=0}^n y^{d+n+1} (-1)^n 2 \binom{n+d}{d} \binom{2n+1}{n-d}. \end{aligned} \quad (4.126)$$

Заменив $k = d + n + 1$, получим

$$f_n(1+2y) = 1 + \sum_{k=n+1}^{2n+1} y^k (-1)^n 2 \binom{k-1}{k-n-1} \binom{2n+1}{2n+1-k}.$$

Согласно свойству симметрии $\binom{k-1}{k-n-1} = \binom{k-1}{n}$ и $\binom{2n+1}{2n+1-k} = \binom{2n+1}{k}$, тогда

$$f_n(1+2y) = 1 + \sum_{k=n+1}^{2n+1} y^k (-1)^n 2 \binom{k-1}{n} \binom{2n+1}{k} = \sum_{d=0}^{2n+1} a_d y^d, \quad (4.127)$$

где

$$a_d = \begin{cases} 1, & \text{если } d = 0, \\ 0, & \text{если } 1 \leq d \leq n, \\ (-1)^n 2 \binom{d-1}{n} \binom{2n+1}{d}, & \text{если } n+1 \leq d \leq 2n+1. \end{cases} \quad (4.128)$$

Из (4.120) и (4.128), следует, что $\forall 0 \leq d \leq 2n+1$: $a_d = b_d$
 $\deg B(y) = \deg f_n(1+2y) = 2n+1$, значит $B(y) = f_n(1+2y)$ и
 $f_n(x) = B_{2n+1}\left(\frac{x+1}{2}\right)$.

Теорема доказана. \square

Лемма 4.7.5. $\forall k, n \in \mathbb{N}$ и $k \leq n$:

$$\sum_{i=k}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot \binom{i}{k} = \frac{1}{4^n} \cdot \frac{2n+1}{2k+1} \cdot \binom{2n}{n} \cdot \binom{n}{k}. \quad (4.129)$$

Доказательство. Используем для доказательства метод математической индукции. Проверим доказываемое равенство при $n = k$

$$\frac{1}{4^k} \cdot \binom{2k}{k} \cdot \binom{k}{k} = \frac{1}{4^k} \cdot \frac{2k+1}{2k+1} \cdot \binom{2k}{k} \cdot \binom{k}{k}. \quad (4.130)$$

Предположим, что

$$\sum_{i=k}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot \binom{i}{k} = \frac{1}{4^n} \cdot \frac{2n+1}{2k+1} \cdot \binom{2n}{n} \cdot \binom{n}{k}, \quad (4.131)$$

тогда

$$\begin{aligned} \sum_{i=k}^{n+1} \frac{1}{4^i} \cdot \binom{2i}{i} \cdot \binom{i}{k} &= \sum_{i=k}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot \binom{i}{k} + \frac{1}{4^{n+1}} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k} \\ &= \frac{1}{4^n} \cdot \frac{2n+1}{2k+1} \cdot \binom{2n}{n} \cdot \binom{n}{k} + \frac{1}{4^{n+1}} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k}. \end{aligned}$$

Обратим внимание на то, что

$$\frac{1}{4^n} \cdot \frac{2n+1}{2k+1} \cdot \binom{2n}{n} \cdot \binom{n}{k} = \frac{1}{4^{n+1}} \cdot \frac{2n-2k+2}{2k+1} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k}, \quad (4.132)$$

следовательно,

$$\begin{aligned}
\sum_{i=k}^{n+1} \frac{1}{4^i} \cdot \binom{2i}{i} \cdot \binom{i}{k} &= \frac{1}{4^{n+1}} \cdot \frac{2n-2k+2}{2k+1} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k} \\
&+ \frac{1}{4^{n+1}} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k} \\
&= \frac{1}{4^{n+1}} \cdot \frac{2n+3}{2k+1} \cdot \binom{2n+2}{n+1} \cdot \binom{n+1}{k}. \quad (4.133)
\end{aligned}$$

Лемма доказана. \square

Рассмотрим соотношение

$$\begin{aligned}
f_n(x) &= x \cdot \sum_{i=0}^n \frac{1}{4^i} \cdot \binom{2i}{i} \cdot (1-x^2)^i \\
&= \sum_{i=0}^n (-1)^i \cdot f_{n,2i+1} \cdot x^{2i+1}, \quad (4.134)
\end{aligned}$$

где $f_{n,2i+1} = \sum_{j=i}^n \frac{1}{4^j} \cdot \binom{2j}{j} \cdot \binom{j}{i}$. Так как согласно Леммы 4.7.5 $\sum_{j=i}^n \frac{1}{4^j} \cdot \binom{2j}{j} \cdot \binom{j}{i} = \frac{1}{4^n} \cdot \frac{2n+1}{2i+1} \cdot \binom{2n}{n} \cdot \binom{n}{i}$, то

$$f_{n,2i+1} = \frac{1}{4^n} \cdot \frac{2n+1}{2i+1} \cdot \binom{2n}{n} \cdot \binom{n}{i}. \quad (4.135)$$

Исследуем зависимость $f_{n,2i+1}$ от параметров n и i . Для этого докажем ряд свойств.

Свойство 4.7.1. Если $n \in \mathbb{N}$ и $0 \leq i \leq n$, то $f_{n+1,2i+1} > f_{n,2i+1}$.

Доказательство. Для любых $n \in \mathbb{N}$ и $0 \leq i \leq n$ выполняется неравенство $f_{n,2i+1} = \frac{1}{4^n} \cdot \frac{2n+1}{2i+1} \cdot \binom{2n}{n} \cdot \binom{n}{i} > 0$.

Вычислим $\frac{f_{n+1,2i+1}}{f_{n,2i+1}}$, получим

$$\frac{f_{n+1,2i+1}}{f_{n,2i+1}} = \frac{1}{4} \cdot \frac{2n+3}{2n+1} \cdot \frac{\binom{2n+2}{n+1} \cdot \binom{n+1}{i}}{\binom{2n}{n} \cdot \binom{n}{i}}. \quad (4.136)$$

Заметим, что

$$\begin{aligned}
\frac{\binom{2n+2}{n+1}}{\binom{2n}{n}} &= \frac{(2n+2)!}{(n+1)! \cdot (n+1)!} \cdot \frac{n! \cdot n!}{(2n)!} \\
&= \frac{2(2n+1)}{n+1}, \quad (4.137)
\end{aligned}$$

$$\begin{aligned}\frac{\binom{n+1}{i}}{\binom{n}{i}} &= \frac{(n+1)!}{(n+1-i)! \cdot i!} \cdot \frac{i! \cdot (n-i)!}{n!} \\ &= \frac{n+1}{n+1-i}.\end{aligned}\quad (4.138)$$

Подставляя (4.137) и (4.138) в (4.136), получим

$$\begin{aligned}\frac{f_{n+1,2i+1}}{f_{n,2i+1}} &= \frac{1}{4} \cdot \frac{2n+3}{2n+1} \cdot \frac{2(2n+1)}{n+1} \cdot \frac{n+1}{n+1-i} \\ &= 1 + \frac{2i+1}{2n+2-2i}.\end{aligned}\quad (4.139)$$

Так как $n \in \mathbb{N}$ и $0 \leq i \leq n$, то выполняется неравенство $\frac{2i+1}{2n+2-2i} > 0$, следовательно $\frac{f_{n+1,2i+1}}{f_{n,2i+1}} > 1$. Учитывая что $\frac{f_{n+1,2i+1}}{f_{n,2i+1}} > 1$ и $f_{n,2i+1} > 0$, $f_{n+1,2i+1} > f_{n,2i+1}$.

Свойство доказано. \square

Свойство 4.7.2.

$$\lim_{n \rightarrow \infty} f_{n,2n+1} = 0, \quad (4.140)$$

$$0 \leq i < n : \lim_{n \rightarrow \infty} f_{n,2i+1} = \infty. \quad (4.141)$$

Доказательство. Так как $\lim_{n \rightarrow \infty} \frac{\sqrt{\pi \cdot n} \cdot \binom{2n}{n}}{4^n} = 1$ [196, Eq. (3')], то

$$\begin{aligned}\lim_{n \rightarrow \infty} f_{n,2n+1} &= \lim_{n \rightarrow \infty} \frac{1}{4^n} \cdot \frac{2n+1}{2n+1} \cdot \binom{2n}{n} \cdot \binom{n}{n} \\ &= \lim_{n \rightarrow \infty} \frac{1}{4^n} \cdot \binom{2n}{n} = \lim_{n \rightarrow \infty} \frac{1}{\sqrt{\pi \cdot n}} \\ &= 0,\end{aligned}\quad (4.142)$$

$$\begin{aligned}\lim_{n \rightarrow \infty} f_{n,2 \cdot 0+1} &= \lim_{n \rightarrow \infty} \frac{2n+1}{4^n} \cdot \binom{2n}{n} \cdot \binom{n}{0} \\ &= \lim_{n \rightarrow \infty} \frac{2n+1}{4^n} \cdot \binom{2n}{n} = \lim_{n \rightarrow \infty} \frac{2n+1}{\sqrt{\pi \cdot n}} \\ &= \infty,\end{aligned}\quad (4.143)$$

$$\begin{aligned}\lim_{n \rightarrow \infty} f_{n,2i+1} &= \lim_{n \rightarrow \infty} \frac{1}{4^n} \cdot \frac{2n+1}{2i+1} \cdot \binom{2n}{n} \cdot \binom{n}{i} \\ &= \lim_{n \rightarrow \infty} \frac{2n+1}{2i+1} \cdot \frac{1}{\sqrt{\pi \cdot n}} \cdot \binom{n}{i} \\ &\geq \lim_{n \rightarrow \infty} \frac{1}{\sqrt{\pi \cdot n}} \cdot \binom{n}{i}.\end{aligned}\quad (4.144)$$

Учитывая что для всех $1 \leq i < n$ выполняется неравенство $\binom{n}{1} \leq \binom{n}{i}$,

$$\lim_{n \rightarrow \infty} f_{n,2i+1} \geq \lim_{n \rightarrow \infty} \frac{1}{\sqrt{\pi \cdot n}} \cdot \binom{n}{1} = \lim_{n \rightarrow \infty} \frac{n}{\sqrt{\pi \cdot n}} = \infty. \quad (4.145)$$

Следовательно,

$$\lim_{n \rightarrow \infty} f_{n,2n+1} = 0, \quad (4.146)$$

$$\forall 0 \leq i < n : \lim_{n \rightarrow \infty} f_{n,2i+1} = \infty. \quad (4.147)$$

Свойство доказано. \square

Свойство 4.7.3. Если $n \geq 4$, то в зависимости от i выполняется одно из двух утверждений:

1. Если $0 \leq i \leq \lfloor \frac{2n-5}{4} \rfloor$, то $f_{n,2(i+1)+1} > f_{n,2i+1}$.
2. Если $\lfloor \frac{2n-5}{4} \rfloor < i < n$, то $f_{n,2(i+1)+1} < f_{n,2i+1}$.

Доказательство. Вычислим $\frac{f_{n,2(i+1)+1}}{f_{n,2i+1}}$, получим

$$\frac{f_{n,2(i+1)+1}}{f_{n,2i+1}} = \frac{2i+1}{2i+3} \cdot \frac{\binom{n}{i+1}}{\binom{n}{i}} = \frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1}. \quad (4.148)$$

Так как $\forall i, n: f_{n,2i+1} > 0$, то для того чтобы $f_{n,2(i+1)+1} > f_{n,2i+1}$ необходимо и достаточно, чтобы $\frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1} > 1$. Решая неравенство получим, что $i \in \left(\frac{n-3-\sqrt{n^2-2n-3}}{4}, \frac{n-3+\sqrt{n^2-2n-3}}{4} \right)$. Так как $i \in \mathbb{Z}$, то левая граница определяется следующим выражением

$$\begin{aligned} \left\lceil \frac{n-3-\sqrt{n^2-2n-3}}{4} \right\rceil &= \left\lceil \frac{n-3 + \lceil -\sqrt{n^2-2n-3} \rceil}{4} \right\rceil \\ &= \left\lceil \frac{n-3 - \lfloor \sqrt{n^2-2n-3} \rfloor}{4} \right\rceil. \end{aligned} \quad (4.149)$$

Вычислим $\lfloor \sqrt{n^2-2n-3} \rfloor$. Учитывая что $n \in \mathbb{N}$ и $n \geq 4$, имеет место следующее неравенство $0 < n-1-\sqrt{n^2-2n-3} < 1$, тогда $\lfloor \sqrt{n^2-2n-3} \rfloor = n-2$, следовательно,

$$\left\lceil \frac{n-3-\sqrt{n^2-2n-3}}{4} \right\rceil = \left\lceil \frac{n-3-(n-2)}{4} \right\rceil = \left\lceil \frac{-1}{4} \right\rceil = 0. \quad (4.150)$$

Рассмотрим правую границу

$$\begin{aligned} \left\lfloor \frac{n-3+\sqrt{n^2-2n-3}}{4} \right\rfloor &= \left\lfloor \frac{n-3+\lfloor \sqrt{n^2-2n-3} \rfloor}{4} \right\rfloor \\ &= \left\lfloor \frac{2n-5}{4} \right\rfloor. \end{aligned} \quad (4.151)$$

Вычислим значение $\frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1}$ при $i = \lfloor \frac{2n-5}{4} \rfloor + 1 = \lfloor \frac{2n-1}{4} \rfloor$, получим

$$\frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1} = \frac{2 \cdot \lfloor \frac{2n-1}{4} \rfloor + 1}{2 \cdot \lfloor \frac{2n-1}{4} \rfloor + 3} \cdot \frac{n - \lfloor \frac{2n-1}{4} \rfloor}{\lfloor \frac{2n-1}{4} \rfloor + 1}. \quad (4.152)$$

Рассмотрим два случая.

Случай 1. Если $n = 2a$, то $i = \lfloor \frac{2n-1}{4} \rfloor = a - 1$:

$$\frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1} = \frac{2a-1}{2a+1} \cdot \frac{a+1}{a} = 1 - \frac{1}{2a^2+a} < 1. \quad (4.153)$$

Случай 2. Если $n = 2a + 1$, то $i = \lfloor \frac{2n-1}{4} \rfloor = a$:

$$\frac{2i+1}{2i+3} \cdot \frac{n-i}{i+1} = \frac{2a+1}{2a+3} = 1 - \frac{2}{2a+3} < 1. \quad (4.154)$$

Следовательно, если $\lfloor \frac{2n-5}{4} \rfloor < i < n$, то $f_{n,2(i+1)+1} < f_{n,2i+1}$.

Свойство доказано. \square

Из Свойства 4.7.3, следует что при фиксированном значении $n \geq 4$, наибольший коэффициент $f_{n,2\lfloor \frac{2n-1}{4} \rfloor + 1}$, а наименьший коэффициент $f_{n,2n+1}$.

Случай 1. Если $n = 2a$, то $i = \lfloor \frac{2n-1}{4} \rfloor = a - 1$:

$$\begin{aligned} f_{n,2\lfloor \frac{2n-1}{4} \rfloor + 1} &= \frac{1}{16^a} \cdot \frac{4a+1}{2a-1} \cdot \binom{4a}{2a} \cdot \binom{2a}{a-1} \\ &= \frac{1}{16^a} \cdot \frac{4a+1}{2a-1} \cdot \binom{4a}{2a} \cdot \binom{2a}{a} \cdot \frac{a}{a+1} \\ &\sim \frac{2}{\sqrt{2\pi \cdot a}} \cdot \frac{4^a}{\sqrt{\pi \cdot a}} = \frac{\sqrt{2} \cdot 4^a}{\pi \cdot a} \\ &= \frac{\sqrt{2} \cdot 2^{n+1}}{\pi \cdot n}. \end{aligned} \quad (4.155)$$

Случай 2. Если $n = 2a + 1$, то $i = \lfloor \frac{2n-1}{4} \rfloor = a$:

$$\begin{aligned}
f_{n,2\lfloor \frac{2n-1}{4} \rfloor+1} &= \frac{1}{4 \cdot 16^a} \cdot \frac{4a+3}{2a+1} \cdot \binom{4a+2}{2a+1} \cdot \binom{2a+1}{a} \\
&= \frac{1}{4 \cdot 16^a} \cdot \frac{4a+3}{a+1} \cdot \binom{4a+2}{2a+1} \cdot \binom{2a}{a} \\
&\sim \frac{4 \cdot 4^a}{\sqrt{\pi \cdot (2a+1)} \cdot \sqrt{\pi \cdot a}} = \frac{2^{n+1}}{\sqrt{\pi \cdot n} \cdot \sqrt{\pi \cdot \frac{n-1}{2}}} \\
&\sim \frac{\sqrt{2} \cdot 2^{n+1}}{\pi \cdot n}.
\end{aligned} \tag{4.156}$$

Из (4.155) и (4.156) следует, что

$$f_{n,2\lfloor \frac{2n-1}{4} \rfloor+1} \sim \frac{\sqrt{2} \cdot 2^{n+1}}{\pi \cdot n}. \tag{4.157}$$

Так как рабочий диапазон в схеме СККС равен $(-p/2, p/2)$, то используя параметры схемы СККС из работы [239], получим следующие ограничения.

1. Если $\log_2 p = 30$, то при реализации схемы СККС можно использовать многочлен не выше $34 \cdot 2 + 1 = 69$ степени, так как $f_{35,2 \cdot 17+1} > 2^{29}$ и $f_{34,2 \cdot 16+1} < 2^{29}$.
2. Если $\log_2 p = 56$, то при реализации схемы СККС можно использовать многочлен не выше $61 \cdot 2 + 1 = 123$ степени, так как $f_{62,2 \cdot 30+1} > 2^{55}$ и $f_{61,2 \cdot 30+1} < 2^{55}$.

4.8 О наилучшем приближении функции определения знака закодированного числа многочленом над полем \mathbb{R}

Для аппроксимации функции знака числа многочленами используют различные подходы, основанные на использовании рациональных функций [294], многочленов Бернштейна [178], многочленов Чебышева первого рода [175, 232], разложения в ряд Фурье и искусственных нейронных сетей [165], Least-squares [164, 184, 273, 316, 339], Newton-Raphson [251]. В качестве меры рассматривают наименьшее отклонение многочлена от функции знака числа. Однако, использование данной меры в окрестности нуля обладает максимальной ошибкой близкой к 0.5 в независимости от степени многочлена, что делает ее неприменимой

для оценки аппроксимации многочленом на множестве $[-1, 1]$. В связи с этим задачу аппроксимации рассматривают на множестве $[-1, -\epsilon] \cup [\epsilon, 1]$, и, соответственно, функцию знака числа заменяют на непрерывную функцию $s(x)$ равную

$$s(x) = \begin{cases} -1, & \text{если } x < -\epsilon, \\ \frac{x}{\epsilon}, & \text{если } -\epsilon \leq x \leq \epsilon, \\ 1, & \text{если } x > \epsilon, \end{cases} \quad (4.158)$$

для которой, согласно теории Чебышева, можно построить единственный многочлен минимального отклонения. Вид минимаксного многочлена для аппроксимации функции знака числа зависит от ϵ . Рассматриваются различные стратегии выбора ϵ для наилучшего приближения, но при этом вопрос построения наилучшего приближения функции знака числа многочленами остается открытым.

Для построения наилучшего приближения функции знака числа многочленом в качестве нормы используем площадь между функцией знака числа и многочленом, вычисляемую согласно следующей формулы

$$\|f(x)\| = \int_{-1}^0 |1 + f(x)| dx + \int_0^1 |1 - f(x)| dx. \quad (4.159)$$

Использование данной нормы позволяет избежать неопределенности, возникающей в окрестности нуля при использовании нормы наименьшего отклонения многочлена от функции знака числа.

Сформулируем задачу построения многочлена наилучшего приближения. Требуется найти наилучшее приближение функции знака числа многочленом $Q_n(x) = \sum_{i=0}^n a_i x^i$, где $\deg Q_n(x) \leq n$. Формально задается, следующим образом

$$\left\| \sum_{i=0}^n a_i^{(0)} x^i \right\| = \Delta = \inf_{a_0, a_1, \dots, a_n} \left\| \sum_{i=0}^n a_i x^i \right\|. \quad (4.160)$$

Если $Q_n(x)$ существует, то он называется многочленом наилучшего приближения функции знака числа. В работе [132, с. 160] доказана теорема о том, что элемент наилучшего приближения существует, остается открытым вопрос о количестве элементов и их виде.

4.8.1 Норма и ее свойства

В разделе рассмотрены основные свойства введенной ранее нормы, используемые в дальнейшем для доказательств.

Свойство 4.8.1. Если $f(x)$ – четная функция, то $\|f(x)\| \geq 2$.

Доказательство. Так как $f(x)$ – четная функция, то

$$\int_{-1}^0 |1 + f(x)| dx = \int_0^1 |1 + f(x)| dx, \quad (4.161)$$

следовательно,

$$\begin{aligned} \|f(x)\| &= \int_{-1}^0 |1 + f(x)| dx + \int_0^1 |1 - f(x)| dx \\ &= \int_0^1 |1 + f(x)| + |1 - f(x)| dx. \end{aligned} \quad (4.162)$$

Учитывая, что $\forall x \in [0, 1]: |1 + f(x)| + |1 - f(x)| \geq 2$,

$$\int_0^1 |1 + f(x)| + |1 - f(x)| dx \geq \int_0^1 2 dx = 2, \quad (4.163)$$

т.е. $\|f(x)\| \geq 2$.

Свойство доказано. □

Рассмотрим пример вычисления нормы при $n = 0$.

Пример 4.8.1. 1. Вычислим $\|a_0\|$, если $|a_0| \leq 1$. Тогда $\int_0^1 |1 + a_0| + |1 - a_0| dx = 2$.
2. Вычислим $\|a_0\|$, если $|a_0| > 1$. Тогда $\int_0^1 |1 + a_0| + |1 - a_0| dx = 2|a_0| > 2$.

Из данных, представленных в примере 4.8.1, можно сделать вывод, что при $n = 0$, существует бесконечное количество многочленов наилучшего приближения функции знака числа, и они имеют вид $f(x) = a_0$, где $|a_0| \leq 1$.

Свойство 4.8.2. Если $f(x)$ – нечетная функция, то

$$\|f(x)\| = 2 \int_0^1 |1 - f(x)| dx.$$

Доказательство. Так как $f(x)$ – нечетная функция, то $\int_{-1}^0 |1 + f(x)| dx = \int_0^1 |1 - f(x)| dx$, следовательно,

$$\|f(x)\| = 2 \int_0^1 |1 - f(x)| dx. \quad (4.164)$$

Свойство доказано. \square

Свойство 4.8.3. Если $f(x)$ – функция общего вида, то $\|f(x)\| \geq \|o(x)\|$, где $f(x) = e(x) + o(x)$, $e(x)$ – четная функция, $o(x)$ – нечетная функция.

Доказательство.

$$\|f(x)\| = \int_{-1}^0 |1 + e(x) + o(x)| dx + \int_0^1 |1 - e(x) - o(x)| dx. \quad (4.165)$$

Пусть $x = -t$, тогда

$$\begin{aligned} \int_{-1}^0 |1 + e(x) + o(x)| dx &= - \int_1^0 |1 + e(-t) + o(-t)| dt \\ &= \int_0^1 |1 + e(t) - o(t)| dt. \end{aligned} \quad (4.166)$$

Следовательно,

$$\begin{aligned} \|f(x)\| &= \int_0^1 |1 + e(x) - o(x)| + |1 - e(x) - o(x)| dx \\ &\geq \int_0^1 |2 - 2o(x)| dx = 2 \int_0^1 |1 - o(x)| dx. \end{aligned} \quad (4.167)$$

Согласно Свойству 4.8.2 $\|o(x)\| = 2 \int_0^1 |1 - o(x)| dx$, тогда $\|f(x)\| \geq \|o(x)\|$.

Свойство доказано. \square

Свойство 4.8.4. Для всех $\phi \in (0, \frac{\pi}{2})$:

$$\|f(x) + g(x)\| \leq \sin^2 \phi \left\| \frac{1}{\sin^2 \phi} \cdot f(x) \right\| + \cos^2 \phi \left\| \frac{1}{\cos^2 \phi} \cdot g(x) \right\|. \quad (4.168)$$

Доказательство. По определению,

$$\|f(x) + g(x)\| = \int_{-1}^0 |1 + f(x) + g(x)| dx + \int_0^1 |1 - f(x) - g(x)| dx. \quad (4.169)$$

Так как, согласно основному тригонометрическому тождеству, $\sin^2 \phi + \cos^2 \phi = 1$, то

$$\begin{aligned}
 |1 + f(x) + g(x)| &= |\sin^2 \phi + f(x) + \cos^2 \phi + g(x)| \\
 &\leq |\sin^2 \phi + f(x)| + |\cos^2 \phi + g(x)| \\
 &= \sin^2 \phi \left| 1 + \frac{1}{\sin^2 \phi} \cdot f(x) \right| \\
 &\quad + \cos^2 \phi \left| 1 + \frac{1}{\cos^2 \phi} \cdot g(x) \right|. \tag{4.170}
 \end{aligned}$$

$$\begin{aligned}
 |1 - f(x) - g(x)| &= |\sin^2 \phi - f(x) + \cos^2 \phi - g(x)| \\
 &\leq |\sin^2 \phi - f(x)| + |\cos^2 \phi - g(x)| \\
 &= \sin^2 \phi \left| 1 - \frac{1}{\sin^2 \phi} \cdot f(x) \right| \\
 &\quad + \cos^2 \phi \left| 1 - \frac{1}{\cos^2 \phi} \cdot g(x) \right|. \tag{4.171}
 \end{aligned}$$

Следовательно,

$$\begin{aligned}
 \|f(x) + g(x)\| &\leq \int_{-1}^0 \sin^2 \phi \left| 1 + \frac{1}{\sin^2 \phi} \cdot f(x) \right| + \cos^2 \phi \left| 1 + \frac{1}{\cos^2 \phi} \cdot g(x) \right| dx \\
 &\quad + \int_0^1 \sin^2 \phi \left| 1 - \frac{1}{\sin^2 \phi} \cdot f(x) \right| + \cos^2 \phi \left| 1 - \frac{1}{\cos^2 \phi} \cdot g(x) \right| dx \\
 &= \sin^2 \phi \left\| \frac{1}{\sin^2 \phi} \cdot f(x) \right\| + \cos^2 \phi \left\| \frac{1}{\cos^2 \phi} \cdot g(x) \right\|. \tag{4.172}
 \end{aligned}$$

Свойство доказано. □

Следствие 4.8.1. Для всех $\phi \in [0, \frac{\pi}{2}]$:

$$\| \sin^2 \phi \cdot f(x) + \cos^2 \phi \cdot g(x) \| \leq \sin^2 \phi \|f(x)\| + \cos^2 \phi \|g(x)\|. \tag{4.173}$$

Доказательство. Согласно Свойству 4.8.4 $\forall \phi \in (0, \frac{\pi}{2})$:

$$\| \sin^2 \phi \cdot f(x) + \cos^2 \phi \cdot g(x) \| \leq \sin^2 \phi \|f(x)\| + \cos^2 \phi \|g(x)\|. \tag{4.174}$$

Покажем, что неравенство выполняется и в случае $\phi = 0$, и в случае $\phi = \frac{\pi}{2}$.

Если $\phi = 0$, то $\|g(x)\| \leq \|g(x)\|$.

Если $\phi = \frac{\pi}{2}$, то $\|f(x)\| \leq \|f(x)\|$.

В обоих случаях неравенство (4.174) выполняется.

Следствие доказано. □

Следствие 4.8.2. Если $\|f(x)\| = \|g(x)\| = a$, то $\forall \phi \in [0, \frac{\pi}{2}]$:

$$\|\sin^2 \phi \cdot f(x) + \cos^2 \phi \cdot g(x)\| \leq a. \quad (4.175)$$

Доказательство. Согласно Следствию 4.8.1, получим

$$\begin{aligned} \|\sin^2 \phi \cdot f(x) + \cos^2 \phi \cdot g(x)\| &\leq \sin^2 \phi \|f(x)\| + \cos^2 \phi \|g(x)\| \\ &= a \cdot \sin^2 \phi + a \cdot \cos^2 \phi = a. \end{aligned} \quad (4.176)$$

Следствие доказано. □

Из примера 4.8.1 следует, что если $n = 0$, то многочленов наилучшего приближения нулевой степени функции знака числа бесконечно много. Если $f(x) = -1$ и $g(x) = 1$, то $Q_0(x) = \sin^2 \phi \cdot f(x) + \cos^2 \phi \cdot g(x) = \cos 2\phi$ задает все многочлены наилучшего приближения нулевой степени для функции знака числа. Исследуем вопрос о количестве многочленов наилучшего приближения функции знака числа степени больше либо равной одному.

4.8.2 Приближения функции определения знака закодированного числа над полем \mathbb{R} многочленами Бернштейна

Рассмотрим вопрос применимости многочленов Бернштейна

$$f_n(x) = \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+1} \cdot \binom{n}{i} x^{2i+1} \quad (4.177)$$

для аппроксимации функции знака числа.

Обратим внимание на то, что $\deg f_n(x) = 2n+1$ и функция $f_n(x)$ является нечетной. Согласно Свойству 4.8.2, $\|f_n(x)\| = 2 \int_0^1 |1 - f_n(x)| dx$. Вычислим значение $\int_0^1 |1 - f_n(x)| dx$, для этого докажем следующее утверждение.

Утверждение 4.8.1. Для всех $n \in \mathbb{Z}_+$:

$$\int_0^1 |1 - f_n(x)| dx = \frac{2n+1}{2n+2} \cdot 4^{-n} \binom{2n}{n}. \quad (4.178)$$

Доказательство. Так как на отрезке $[-1, 1]$ многочлены Бернштейна обладают следующим свойством $\forall n \in \mathbb{Z}_+, x \in [-1, 1]: |f_n(x)| \leq 1$ [178], то

$$\int_0^1 |1 - f_n(x)| dx = \int_0^1 1 - f_n(x) dx. \quad (4.179)$$

Подставляя вместо $f_n(x)$ выражение (4.177), получим

$$\begin{aligned}
\int_0^1 1 - f_n(x) dx &= \int_0^1 1 - \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+1} \cdot \binom{n}{i} x^{2i+1}(x) dx \\
&= \left(x - \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{(2i+1)(2i+2)} \cdot \binom{n}{i} x^{2i+2} \right) \Big|_0^1 \\
&= 1 - \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{(2i+1)(2i+2)} \cdot \binom{n}{i}. \quad (4.180)
\end{aligned}$$

Представим $\frac{1}{(2i+1)(2i+2)}$ в виде $\frac{1}{(2i+1)(2i+2)} = \frac{1}{2i+1} - \frac{1}{2i+2}$, получим

$$\begin{aligned}
\int_0^1 1 - f_n(x) dx &= 1 - \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+1} \cdot \binom{n}{i} \\
&\quad + \frac{2n+1}{4^n} \binom{2n}{n} \sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+2} \cdot \binom{n}{i}. \quad (4.181)
\end{aligned}$$

Учитывая, что

$$\sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+1} \cdot \binom{n}{i} = \frac{4^n}{(2n+1) \binom{2n}{n}}, \quad (4.182)$$

$$\sum_{i=0}^n (-1)^i \cdot \frac{1}{2i+2} \cdot \binom{n}{i} = \frac{1}{2n+2}, \quad (4.183)$$

получим

$$\int_0^1 |1 - f_n(x)| dx = \frac{2n+1}{2n+2} \cdot 4^{-n} \binom{2n}{n}. \quad (4.184)$$

Утверждение доказано. \square

Следствие 4.8.3. Для всех $n \in \mathbb{Z}_+$:

$$\|f_n(x)\| \leq \min \left(1, \frac{2}{\sqrt{3n+1}} \right). \quad (4.185)$$

Доказательство. Так как $f_n(x)$ – нечетная функция, то согласно Свойству 4.8.2 $\|f_n(x)\| = 2 \int_0^1 |1 - f_n(x)| dx$.

Пусть $n \geq 1$, тогда учитывая Утверждение 4.8.1,

$$\|f_n(x)\| = 2 \cdot \frac{2n+1}{2n+2} \cdot 4^{-n} \binom{2n}{n} < \frac{2}{4^n} \binom{2n}{n} \leq \frac{2}{\sqrt{3n+1}} \leq 1. \quad (4.186)$$

Если $n = 0$, то $\|f_n(x)\| = 1$, следовательно $\|f_n(x)\| \leq \min \left(1, \frac{2}{\sqrt{3n+1}} \right)$.

Следствие доказано. \square

Из Свойства 4.8.1 и Следствия 4.8.3, можно сделать вывод о том, что если $n \geq 1$, то многочлен наилучшего приближения для функции знака числа $Q_n(x)$ не является четной функцией.

4.8.3 Свойства многочлена наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R}

Так как многочлен $Q_n(x)$ является непрерывной функцией на отрезке $[-1, 1]$, то, согласно теореме Вейерштрасса, он ограничен на нем и притом достигает минимальных и максимальных значений. Т.е. существуют $x_m, x_M \in [-1, 1]$ такие, что $\forall x \in [-1, 1]: Q_n(x_m) \leq Q_n(x) \leq Q_n(x_M)$. Обозначим $m_Q = Q_n(x_m)$ и $M_Q = Q_n(x_M)$, тогда $m_Q \leq M_Q$. Исследуем возможные значения m_Q и M_Q многочлена наилучшего приближения функции знака числа $Q_n(x)$, результат представим в виде следующей леммы.

Лемма 4.8.1. *Если $n \geq 1$ и $Q_n(x)$ – многочлен наилучшего приближения функции знака числа, то $m_Q \leq -1$, а $M_Q \geq 1$.*

Доказательство. Разобьем двумерное пространство \mathbb{R}^2 на множества с помощью кривых $m_Q = \pm 1$ и $M_Q = \pm 1$ (рис. 4.5). Рассмотрим каждое из множеств

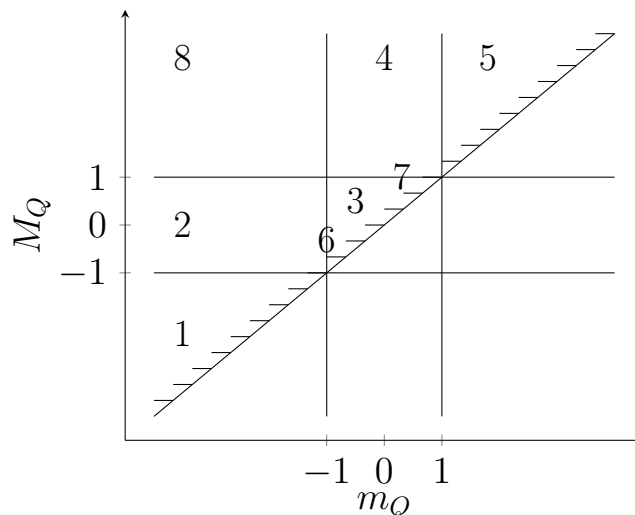


Рисунок 4.5 — Множество возможных значений m_Q и M_Q

в отдельности.

Случай 1 (множество 1, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет усло-

вию $M_Q \leq -1$, тогда $\forall x \in [-1, 1]: Q_n(x) \leq -1, 1 + Q_n(x) \leq 0, 1 - Q_n(x) \geq 0$, следовательно,

$$\|Q_n(x)\| = \int_{-1}^0 -1 - Q_n(x) dx + \int_0^1 1 - Q_n(x) dx = \int_{-1}^1 -Q_n(x) dx \geq \int_{-1}^1 1 dx = 2.$$

Согласно Следствию 4.8.3, для $n \geq 1$ многочлен наилучшего приближения функции знака числа характеризуется свойством $\|Q_n(x)\| \leq 1$, пришли к противоречию, значит исходное предположение не верно, т.е. $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 2 (множество 2, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $m_Q < -1, -1 < M_Q < 1$, тогда $\forall x \in [-1, 1]: 1 - Q_n(x) > 0$. Следовательно,

$$\|Q_n(x)\| = \int_{-1}^0 |1 + Q_n(x)| dx + \int_0^1 1 - Q_n(x) dx. \quad (4.187)$$

Вычислим $\|Q_n(x) + 1 - M_Q\|$, получим

$$\|Q_n(x) + 1 - M_Q\| = \int_{-1}^0 |2 + Q_n(x) - M_Q| dx + \int_0^1 M_Q - Q_n(x) dx. \quad (4.188)$$

Вычислим $\Delta_n = \|Q_n(x)\| - \|Q_n(x) + 1 - M_Q\|$, получим

$$\begin{aligned} \Delta_n &= 1 - M_Q + \int_{-1}^0 |1 + Q_n(x)| - |2 + Q_n(x) - M_Q| dx \\ &= \int_{-1}^0 |1 + Q_n(x)| - |2 + Q_n(x) - M_Q| + 1 - M_Q dx. \end{aligned} \quad (4.189)$$

Учитывая, что $\forall x \in [-1, 0]$:

$$|2 + Q_n(x) - M_Q| \leq |1 + Q_n(x)| + |1 - M_Q| = |1 + Q_n(x)| + 1 - M_Q,$$

то

$$|1 + Q_n(x)| - |2 + Q_n(x) - M_Q| + 1 - M_Q \geq 0. \quad (4.190)$$

Следовательно, $\Delta_n \geq 0$. Если $\Delta_n > 0$, то $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа, значит $\Delta_n = 0$ и $\forall x \in [-1, 0]: 1 + Q_n(x) \geq 0$, получим

$$\|Q_n(x)\| = \int_{-1}^0 |1 + Q_n(x)| dx + \int_0^1 1 - Q_n(x) dx = 2 + \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx.$$

Пусть $\lambda = \frac{2}{1+M_Q} > 1$, тогда

$$\forall x \in [-1, 0] : \lambda Q_n(x) + \frac{1 - M_Q}{1 + M_Q} + 1 = \lambda(Q_n(x) + 1) \geq 0, \quad (4.191)$$

$$\forall x \in [0, 1] : 1 - \lambda Q_n(x) - \frac{1 - M_Q}{1 + M_Q} = \lambda(M_Q - Q_n(x)) \geq 0. \quad (4.192)$$

Следовательно,

$$\left\| \lambda Q_n(x) + \frac{1 - M_Q}{1 + M_Q} \right\| = 2 + \lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right). \quad (4.193)$$

Так как, согласно условию леммы, $n \geq 1$, то с учетом Следствия 4.8.3 $\forall n \geq 1: \|Q_n(x)\| \leq 1$. Так как $\lambda > 1$ и $\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \leq -1$, получим

$$\lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.194)$$

Следовательно, $\left\| \lambda Q_n(x) + \frac{1 - M_Q}{1 + M_Q} \right\| < \|Q_n(x)\|$. Значит $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 3 (множество 3, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $-1 < m_Q \leq M_Q < 1$, тогда $\forall x \in [-1, 1]: -1 < Q_n(x) < 1, 1 + Q_n(x) > 0, 1 - Q_n(x) > 0$. Пусть $M = \max(|m_Q|, |M_Q|) < 1$. Если $M = 0$, то $Q_n(x) = 0$, и $\|Q_n(x)\| = 2$, что противоречит свойству многочлена наилучшего приближения $\forall n \geq 1: \|Q_n(x)\| \leq 1$, следовательно, $M \neq 0$.

Пусть $\lambda = \frac{1}{M} > 1$, тогда $\forall x \in [-1, 1]: -1 \leq \lambda \cdot Q_n(x) \leq 1, 1 + \lambda \cdot Q_n(x) \geq 0, 1 - \lambda \cdot Q_n(x) \geq 0$. Вычислим значение $\|\lambda Q_n(x)\|$, получим

$$\|\lambda Q_n(x)\| = \int_{-1}^0 1 + \lambda Q_n(x) dx - \int_0^1 1 - \lambda Q_n(x) dx = 2 + \int_{-1}^0 \lambda Q_n(x) dx - \int_0^1 \lambda Q_n(x) dx.$$

Так как, согласно условию леммы, $n \geq 1$, и с учетом Следствия 4.8.3 $\forall n \geq 1: \|Q_n(x)\| \leq 1$, получим

$$\int_{-1}^0 \lambda Q_n(x) dx - \int_0^1 \lambda Q_n(x) dx \leq -1. \quad (4.195)$$

Вычислим $\|\lambda \cdot Q_n(x)\|$, получим

$$\begin{aligned} \|\lambda \cdot Q_n(x)\| &= \int_{-1}^0 1 + \lambda \cdot Q_n(x) dx - \int_0^1 1 - \lambda \cdot Q_n(x) dx \\ &= 2 + \lambda \cdot \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right). \end{aligned} \quad (4.196)$$

Так как $\lambda > 1$ и $\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \leq -1$, то

$$\lambda \cdot \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.197)$$

Следовательно,

$$2 + \lambda \cdot \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < 2 + \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \quad (4.198)$$

и

$$\|\lambda \cdot Q_n(x)\| < \|Q_n(x)\|. \quad (4.199)$$

Пришли к противоречию, значит исходное предположение не верно, т.е. $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 4 (множество 4, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $M_Q > 1$, $-1 < m_Q < 1$, тогда $\|Q_n(x)\|$ равна

$$\|Q_n(x)\| = \int_{-1}^0 1 + Q_n(x) dx + \int_0^1 |1 - Q_n(x)| dx. \quad (4.200)$$

Вычислим $\|Q_n(x) - 1 - m_Q\|$, получим

$$\|Q_n(x) - 1 - m_Q\| = \int_{-1}^0 Q_n(x) - m_Q dx + \int_0^1 |2 - Q_n(x) + m_Q| dx. \quad (4.201)$$

Вычислим $\Delta_n = \|Q_n(x)\| - \|Q_n(x) - 1 - m_Q\|$, получим

$$\begin{aligned} \Delta_n &= 1 + m_Q + \int_0^1 |1 - Q_n(x)| - |2 - Q_n(x) + m_Q| dx \\ &= \int_0^1 1 + m_Q + |1 - Q_n(x)| - |2 - Q_n(x) + m_Q| dx. \end{aligned} \quad (4.202)$$

Учитывая, что $\forall x \in [0, 1]: |2 - Q_n(x) + m_Q| \leq |1 - Q_n(x)| + |1 + m_Q| = |1 - Q_n(x)| + 1 + m_Q$, то $\forall x \in [0, 1]: |1 - Q_n(x)| - |2 - Q_n(x) + m_Q| + 1 + m_Q \geq 0$, следовательно, $\Delta_n \geq 0$. Если $\Delta_n > 0$, то $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа, значит $\Delta_n = 0$ и $\forall x \in [0, 1]: 1 - Q_n(x) \geq 0$, тогда

$$\|Q_n(x)\| = 2 + \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.203)$$

Пусть $\lambda = \frac{2}{1-m_Q} > 1$, тогда

$$\forall x \in [-1, 0] : \lambda Q_n(x) - \frac{1+m_Q}{1-m_Q} + 1 = \lambda(Q_n(x) - m_Q) \geq 0, \quad (4.204)$$

$$\forall x \in [0, 1] : 1 - \lambda Q_n(x) + \frac{1+m_Q}{1-m_Q} = \lambda(1 - Q_n(x)) \geq 0. \quad (4.205)$$

Следовательно,

$$\left\| \lambda Q_n(x) - \frac{1+m_Q}{1-m_Q} \right\| = 2 + \lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right). \quad (4.206)$$

Так как $\lambda > 1$ и $\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \leq -1$, то

$$\lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.207)$$

Следовательно, $\left\| \lambda Q_n(x) - \frac{1+m_Q}{1-m_Q} \right\| < \|Q_n(x)\|$. Значит $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 5 (множество 5, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $m_Q \geq 1$. Следовательно, $\forall x \in [-1, 1] : Q_n(x) \geq 1, 1 + Q_n(x) \geq 0, 1 - Q_n(x) \leq 0$, значит

$$\|Q_n(x)\| = \int_{-1}^0 1 + Q_n(x) dx + \int_0^1 Q_n(x) - 1 dx = \int_{-1}^1 Q_n(x) dx \geq \int_{-1}^1 1 dx = 2.$$

Согласно Следствию 4.8.3, для $n \geq 1$ многочлен наилучшего приближения функции знака числа характеризуется свойством $\|Q_n(x)\| \leq 1$. Таким образом, получаем противоречие, значит и исходное предположение не верно, т.е. $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 6 (множество 6, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $m_Q = -1, -1 < M_Q < 1$, тогда

$$\begin{aligned} \|Q_n(x)\| &= \int_{-1}^0 1 + Q_n(x) dx + \int_0^1 1 - Q_n(x) dx \\ &= 2 + \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \end{aligned} \quad (4.208)$$

Пусть $\lambda = \frac{2}{1+M_Q} > 1$, тогда

$$\forall x \in [-1, 0] : \lambda Q_n(x) + \frac{1-M_Q}{1+M_Q} + 1 = \lambda(1 + Q_n(x)) \geq 0, \quad (4.209)$$

$$\forall x \in [0, 1] : 1 - \lambda Q_n(x) - \frac{1-M_Q}{1+M_Q} = \lambda(M_Q - Q_n(x)) \geq 0. \quad (4.210)$$

Следовательно,

$$\left\| \lambda Q_n(x) + \frac{1 - M_Q}{1 + M_Q} \right\| = 2 + \lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right). \quad (4.211)$$

Так как $\lambda > 1$ и $\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \leq -1$, то

$$\lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.212)$$

Следовательно, $\left\| \lambda Q_n(x) + \frac{1 - M_Q}{1 + M_Q} \right\| \leq \|Q_n(x)\|$. Значит $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Случай 7 (множество 7, рис. 4.5). Предположим, что $Q_n(x)$ является многочленом наилучшего приближения функции знака числа и удовлетворяет условию $M_Q = 1$, $-1 < m_Q < 1$, тогда

$$\begin{aligned} \|Q_n(x)\| &= \int_{-1}^0 1 + Q_n(x) dx + \int_0^1 1 - Q_n(x) dx \\ &= 2 + \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \end{aligned} \quad (4.213)$$

Пусть $\lambda = \frac{2}{1 - m_Q} > 1$, тогда

$$\forall x \in [-1, 0] : \lambda Q_n(x) - \frac{1 + m_Q}{1 - m_Q} + 1 = \lambda(Q_n(x) - m_Q) \geq 0, \quad (4.214)$$

$$\forall x \in [0, 1] : 1 - \lambda Q_n(x) + \frac{1 + m_Q}{1 - m_Q} = \lambda(1 - Q_n(x)) \geq 0. \quad (4.215)$$

Следовательно,

$$\left\| \lambda Q_n(x) - \frac{1 + m_Q}{1 - m_Q} \right\| = 2 + \lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right). \quad (4.216)$$

Так как $\lambda > 1$ и $\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \leq -1$, то

$$\lambda \left(\int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx \right) < \int_{-1}^0 Q_n(x) dx - \int_0^1 Q_n(x) dx. \quad (4.217)$$

Следовательно, $\left\| \lambda Q_n(x) - \frac{1 + m_Q}{1 - m_Q} \right\| \leq \|Q_n(x)\|$. Значит $Q_n(x)$ не является многочленом наилучшего приближения функции знака числа.

Так как во всех семи случаях получили противоречие то, следовательно, если $Q_n(x)$ многочлен наилучшего приближения функции знака числа, то он удовлетворяет граничным условиям, задающим множество 8 (рис. 4.5).

Лемма доказана. □

Лемма 4.8.2. При $n \geq 1$ существует нечетная функция $Q_n^1(x)$ являющаяся многочленом наилучшего приближения функции знака числа.

Доказательство. Существование многочлена $Q_n(x)$ наилучшего приближения функции знака числа следует из теоремы [132, с. 160]. Так как при $n \geq 1$ многочлен наилучшего приближения функции знака числа $Q_n(x)$ не является четной функцией, то, следовательно, $Q_n(x)$ является либо функцией общего вида либо нечетной функцией.

Предположим, что $Q_n(x)$ функция общего вида, тогда ее можно представить в виде $Q_n(x) = Q_n^0(x) + Q_n^1(x)$, где $Q_n^0(x)$ – четная функция и $Q_n^1(x)$ – нечетная функция. Из Свойства 4.8.3 следует, что $\|Q_n(x)\| \geq \|Q_n^1(x)\|$. Учитывая, что $Q_n(x)$ многочлен наилучшего приближения функции знака числа, $\|Q_n(x)\| = \|Q_n^1(x)\|$. Значит нечетная функция $Q_n^1(x)$ – многочлен наилучшего приближения функции знака числа. Следовательно, при любом $n \geq 1$ существует многочлен наилучшего приближения $Q_n(x)$ являющийся нечетной функцией.

Лемма доказана. □

Следствие 4.8.4. Пусть $n \geq 1$ и нечетная функция $Q_n^1(x)$ является многочленом наилучшего приближения функции знака числа, тогда $M_Q > 1$, а $m_Q < -1$.

Доказательство. Из Леммы 4.8.1, следует, что если $n \geq 1$ и $Q_n^1(x)$ многочлен наилучшего приближения, то $M_Q \geq 1$ и $m_Q \leq 1$. Учитывая что $Q_n^1(x)$ – нечетная функция, $M_Q = -m_Q$. Предположим, что нечетная функция $Q_n^1(x)$ является многочленом наилучшего приближения функции знака числа и $M_Q = -m_Q = 1$.

Рассмотрим функцию $R(x) = \lambda Q_n^1(x)$, где $\lambda \in \mathbb{R}$. Так как $Q_n^1(x)$ нечетная функция, то и $R(x)$ нечетная функция. Вычислим $\|Q_n^1(x)\|$ и $\|R(x)\|$, используя Свойство 4.8.2, получим

$$\|Q_n^1(x)\| = 2 \int_0^1 1 - Q_n^1(x) dx, \quad (4.218)$$

$$\|R(x)\| = 2 \int_0^1 |1 - \lambda Q_n^1(x)| dx. \quad (4.219)$$

Покажем, что существует такое $\lambda > 1$, для которого выполняется неравенство $\|Q_n^1(x)\| > \|R(x)\|$, равносильное следующему неравенству

$$\int_0^1 1 - Q_n^1(x) - |1 - \lambda Q_n^1(x)| dx > 0. \quad (4.220)$$

Введем следующие обозначения

$$I = \int_0^1 1 - Q_n^1(x) - |1 - \lambda Q_n^1(x)| dx, \quad (4.221)$$

$$G^+ = \{x | x \in [0, 1] \& 1 - \lambda Q_n^1(x) \geq 0\}, \quad (4.222)$$

$$G^- = \{x | x \in [0, 1] \& 1 - \lambda Q_n^1(x) \leq 0\}, \quad (4.223)$$

тогда

$$\begin{aligned} I &= \int_{G^+} \lambda Q_n^1(x) - Q_n^1(x) dx + \int_{G^-} 2 - Q_n^1(x) - \lambda Q_n^1(x) dx \\ &= \lambda \int_{G^+} Q_n^1(x) dx - \int_{G^+} Q_n^1(x) dx + 2D_{G^-} - \int_{G^-} Q_n^1(x) dx - \lambda \int_{G^-} Q_n^1(x) dx, \end{aligned}$$

где D_{G^-} – длина множества G^- .

Учитывая, что $\int_{G^+} Q_n^1(x) dx + \int_{G^-} Q_n^1(x) dx = \int_0^1 Q_n^1(x) dx$, получим

$$\begin{aligned} I &= \lambda \left(\int_{G^+} Q_n^1(x) dx - \int_{G^-} Q_n^1(x) dx \right) + 2D_{G^-} - \int_0^1 Q_n^1(x) dx \\ &= \lambda \left(\int_0^1 Q_n^1(x) dx - 2 \int_{G^-} Q_n^1(x) dx \right) + 2D_{G^-} - \int_0^1 Q_n^1(x) dx. \end{aligned} \quad (4.224)$$

Так как $\int_{G^-} Q_n^1(x) dx \leq \int_{G^-} 1 dx = D_{G^-}$, то

$$I \geq (\lambda - 1) \left(\int_0^1 Q_n^1(x) dx - 2D_{G^-} \right). \quad (4.225)$$

Обозначим $g(\lambda) = \{D_{G^-} | G^- = \{x | 1 - \lambda Q_n^1(x) \leq 0 \& 0 \leq x \leq 1\}\}$. Заметим, что при $n \geq 1 \forall x \in [0, 1]: Q_n^1(x) \leq 1$, тогда $g(1) = 0$, а $\forall \lambda > 1: g(\lambda) < 1$.

Рассмотрим два случая.

Случай 1: если $\forall x \in [0, 1]: Q_n^1(x) < 1$, тогда существует такое число $x_a \in [0, 1]$, для которого выполняется $\forall x \in [0, 1]: Q_n^1(x) \leq Q_n^1(x_a) = M_Q^a$. Если $M_Q^a \leq 0$, то $\int_0^1 1 - Q_n^1(x) dx \geq 1$, следовательно, $\|Q_n^1(x)\| \geq 2 > 1$. В этом случае $Q_n^1(x)$ – не является многочленом наилучшего приближения функции знака числа. Если $M_Q^a > 0$, выберем в качестве λ значение $\lambda = \frac{1}{M_Q^a} > 1$, для которого будет соответственно выполняться неравенство $\|Q_n^1(x)\| > \|R(x)\|$. В этом

случае $Q_n^1(x)$ также не будет является многочленом наилучшего приближения функции знака числа.

Случай 2: если $M_Q^a = 1$, то $g(\lambda)$ является возрастающей функцией, т.е. будет существовать такое $\epsilon > 1$, для которого выполняется равенство $\int_0^1 Q_n^1(x) dx - 2g(\epsilon) = 0$. Следовательно, для любого $\lambda \in (1, \epsilon)$, выполняется неравенство

$$\int_0^1 1 - Q_n^1(x) - |1 - \lambda Q_n^1(x)| dx > 0. \quad (4.226)$$

Следовательно, пришли к противоречию. Если $M_Q = 1$ и $m_Q = -1$, то $\forall n \geq 1$: $Q_n^1(x)$ не является многочленом наилучшего приближения функции знака числа.

Следствие доказано. □

4.8.4 Количество многочленов наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R} являющихся нечетными функциями

В Лемме 4.8.2 доказано, что при $n \geq 1$ существует многочлен наилучшего приближения функции знака числа, являющийся нечетной функцией, но вопрос о количестве таких многочленов является открытым. Рассмотрим следующую теорему.

Теорема 4.8.1. *Если $n \geq 1$, то существует единственная нечетная функция $Q_n^1(x)$, являющаяся многочленом наилучшего приближения функции знака числа. В зависимости от n функция $Q_n^1(x)$ определяется следующим образом:*

1. Если n – нечетное число, то

$$Q_n^1(x) = x \sum_{i=1}^{\frac{n+1}{2}} \frac{1}{\sin \frac{i\pi}{n+3}} \prod_{j=1, j \neq i}^{\frac{n+1}{2}} \frac{x^2 - \sin^2 \frac{j\pi}{n+3}}{\sin^2 \frac{i\pi}{n+3} - \sin^2 \frac{j\pi}{n+3}} \quad (4.227)$$

и

$$\|Q_n^1(x)\| = 2 \operatorname{tg} \frac{\pi}{2n+6}. \quad (4.228)$$

2. Если n – четное число, то

$$Q_n^1(x) = x \sum_{i=1}^{\frac{n}{2}} \frac{1}{\sin \frac{i\pi}{n+2}} \prod_{j=1, j \neq i}^{\frac{n}{2}} \frac{x^2 - \sin^2 \frac{j\pi}{n+2}}{\sin^2 \frac{i\pi}{n+2} - \sin^2 \frac{j\pi}{n+2}} \quad (4.229)$$

и

$$\|Q_n^1(x)\| = 2 \operatorname{tg} \frac{\pi}{2n+4}. \quad (4.230)$$

Доказательство. Рассмотрим точки $0 < x_1 < x_2 < \dots < x_u \leq 1$ такие, что $\forall i = \overline{1, u}: Q_n^1(x_i) = 1$. Согласно Следствию 4.8.4, значение M_Q^a удовлетворяет условию $M_Q^a > 1$. Учитывая, что $Q_n^1(x)$ – нечетная непрерывная функция, существует хотя бы одна точка $x_1 \in [0, 1]$ такая, что $Q_n^1(x) = 1$.

Рассмотрим точки $0 = y_0 < y_1 < y_2 < \dots < y_v < y_{v+1} = 1$. В каждой из точек y_1, y_2, \dots, y_v значение функции $f(x) = 1 - Q_{2v-1}^1(x) = 1 - \sum_{i=0}^{v-1} a_{2i+1} \cdot x^{2i+1}$ меняет свой знак.

$$\begin{aligned} I_v &= \frac{\|Q_{2v-1}^1(x)\|}{2} = \int_0^1 |1 - Q_n^1(x)| dx \\ &= \sum_{i=0}^v (-1)^i \int_{y_i}^{y_{i+1}} f(x) dx \\ &= 2 \cdot \sum_{i=1}^v (-1)^{i+1} F(y_i) + (-1)^v F(y_{v+1}), \end{aligned} \quad (4.231)$$

где $F(x) = x - \sum_{i=0}^{v-1} \frac{a_{2i+1}}{2i+2} x^{2i+2}$.

Вычислим значения частных производных $\forall i = \overline{1, v}$:

$$\frac{\partial F(y_i)}{\partial y_i} = 1 - \sum_{i=0}^{v-1} a_{2i+1} y_i^{2i+1} - \sum_{i=0}^{v-1} \frac{\partial a_{2i+1}}{\partial y_i} \cdot \frac{y_i^{2i+2}}{2i+2}. \quad (4.232)$$

Так как $1 - \sum_{i=0}^{v-1} a_{2i+1} \cdot y_i^{2i+1} = 0$ по определению, то

$$\frac{\partial F(y_i)}{\partial y_i} = - \sum_{i=0}^{v-1} \frac{\partial a_{2i+1}}{\partial y_i} \cdot \frac{y_i^{2i+2}}{2i+2}. \quad (4.233)$$

Вычислим значение частных производных $\forall i \neq j$

$$\frac{\partial F(y_i)}{\partial y_j} = - \sum_{i=0}^{v-1} \frac{\partial a_{2i+1}}{\partial y_j} \cdot \frac{y_i^{2i+2}}{2i+2}. \quad (4.234)$$

Необходимым условием, чтобы значение $\|Q_{2v+1}^1(x)\|$ было минимальным, является $\frac{\partial I_v}{\partial y_i} = 0, \forall i = \overline{1, v}$, следовательно,

$$\begin{aligned} \frac{\partial I_v}{\partial y_i} &= -2 \sum_{j=1}^v (-1)^{j+1} \sum_{k=0}^{v-1} \frac{\partial a_{2k+1}}{\partial y_i} \cdot \frac{y_j^{2k+2}}{2k+2} \\ &- (-1)^v \sum_{k=0}^{v-1} \frac{\partial a_{2k+1}}{\partial y_i} \cdot \frac{1}{2k+2} \\ &= - \sum_{k=0}^{v-1} \frac{\partial a_{2k+1}}{\partial y_i} \cdot \frac{1}{2k+2} \left(2 \sum_{j=1}^v (-1)^{j+1} y_j^{2k+2} + (-1)^v \right). \end{aligned} \quad (4.235)$$

Решая систему $\frac{\partial I_v}{\partial y_i} = 0$ [261], получим

$$\forall k = \overline{1, v-1} : 2 \sum_{j=1}^v (-1)^{j+1} y_j^{2k+2} + (-1)^v = 0. \quad (4.236)$$

Вычислим значение I_v используя формулы (4.231) и (4.236), получим

$$I_v = 2 \sum_{i=1}^v (-1)^{i+1} \sin \frac{i \cdot \pi}{2v+2} + (-1)^v. \quad (4.237)$$

Если v – чётно, то

$$I_v = 2 \sum_{i=1}^{v/2} \sin \frac{(2i-1)\pi}{2v+2} - 2 \sum_{i=1}^{v/2} \sin \frac{i \cdot \pi}{v+1} + 1. \quad (4.238)$$

Используя формулу $\sin(\alpha - \beta) = \sin \alpha \cos \beta - \sin \beta \cos \alpha$, где $\alpha = \frac{2i\pi}{2v+2}$ и $\beta = \frac{\pi}{2v+2}$, получим

$$\begin{aligned} \sum_{i=1}^{v/2} \sin \frac{(2i-1)\pi}{2v+2} &= \sum_{i=1}^{v/2} \left(\sin \frac{i\pi}{v+1} \cos \frac{\pi}{2v+2} - \sin \frac{\pi}{2v+2} \cos \frac{i\pi}{v+1} \right) \\ &= \cos \frac{\pi}{2v+2} \sum_{i=1}^{v/2} \sin \frac{i\pi}{v+1} \\ &- \sin \frac{\pi}{2v+2} \sum_{i=1}^{v/2} \cos \frac{i\pi}{v+1}. \end{aligned} \quad (4.239)$$

Так как $\frac{1}{2} + \sum_{i=1}^n \cos ix = \frac{\sin((n+\frac{1}{2})x)}{2\sin\frac{x}{2}}$ и $\sum_{i=1}^n \sin ix = \frac{\cos\frac{x}{2} - \cos((n+\frac{1}{2})x)}{2\sin\frac{x}{2}}$ [377, с. 2], где $n = \frac{v}{2}$, а $x = \frac{\pi}{v+1}$, получим

$$\sum_{i=1}^{v/2} \cos \frac{i\pi}{v+1} = \frac{1}{2\sin\frac{\pi}{2v+2}} - \frac{1}{2}, \quad (4.240)$$

$$\sum_{i=1}^{v/2} \sin \frac{i\pi}{v+1} = \frac{\cos\frac{\pi}{2v+2}}{2\sin\frac{\pi}{2v+2}}. \quad (4.241)$$

Следовательно,

$$\begin{aligned} I_v &= 2 \left(\cos \frac{\pi}{2v+2} \cdot \frac{\cos\frac{\pi}{2v+2}}{2\sin\frac{\pi}{2v+2}} - \sin \frac{\pi}{2v+2} \left(\frac{1}{2\sin\frac{\pi}{2v+2}} - \frac{1}{2} \right) \right) - \frac{\cos\frac{\pi}{2v+2}}{\sin\frac{\pi}{2v+2}} + 1 \\ &= \frac{\cos^2\frac{\pi}{2v+2} + \sin^2\frac{\pi}{2v+2}}{\sin\frac{\pi}{2v+2}} - \frac{\cos\frac{\pi}{2v+2}}{\sin\frac{\pi}{2v+2}}. \end{aligned} \quad (4.242)$$

Используя основное тригонометрическое тождество $\cos^2 2\alpha + \sin^2 2\alpha = 1$ и формулы двойного угла $1 - \cos 2\alpha = 2\sin^2 \alpha$, $\sin 2\alpha = 2\sin \alpha \cos \alpha$, где $\alpha = \frac{\pi}{4v+4}$, получим

$$I_v = \operatorname{tg} \frac{\pi}{4v+4}. \quad (4.243)$$

Если v – нечетно, то

$$I_v = 2 \sum_{i=1}^{\frac{v+1}{2}} \sin \frac{(2i-1)\pi}{2v+2} - 2 \sum_{i=1}^{\frac{v-1}{2}} \sin \frac{i\pi}{v+1} - 1. \quad (4.244)$$

Используя формулу $\sin(\alpha - \beta) = \sin \alpha \cos \beta - \sin \beta \cos \alpha$, где $\alpha = \frac{2i\pi}{2v+2}$, а $\beta = \frac{\pi}{2v+2}$, получим

$$\begin{aligned} \sum_{i=1}^{\frac{v+1}{2}} \sin \frac{(2i-1)\pi}{2v+2} &= \sum_{i=1}^{\frac{v+1}{2}} \left(\sin \frac{i\pi}{v+1} \cos \frac{\pi}{2v+2} - \sin \frac{\pi}{2v+2} \cos \frac{i\pi}{v+1} \right) \\ &= \cos \frac{\pi}{2v+2} \sum_{i=1}^{\frac{v+1}{2}} \sin \frac{i\pi}{v+1} \\ &\quad - \sin \frac{\pi}{2v+2} \sum_{i=1}^{\frac{v+1}{2}} \cos \frac{i\pi}{v+1}. \end{aligned} \quad (4.245)$$

Так как $\sum_{i=1}^n \sin ix = \frac{\cos \frac{x}{2} - \cos((n+\frac{1}{2})x)}{2 \sin \frac{x}{2}}$ [377, с. 2], где $n = \frac{v+1}{2}$, а $x = \frac{\pi}{v+1}$, получим

$$\begin{aligned} \sum_{i=1}^{\frac{v+1}{2}} \sin \frac{i\pi}{v+1} &= \frac{\cos \frac{\pi}{2v+2} - \cos\left(\left(\frac{v+1}{2} + \frac{1}{2}\right) \frac{\pi}{v+1}\right)}{2 \sin \frac{\pi}{2v+2}} \\ &= \frac{\cos \frac{\pi}{2v+2} - \cos\left(\frac{\pi}{2} + \frac{\pi}{2v+2}\right)}{2 \sin \frac{\pi}{2v+2}}. \end{aligned} \quad (4.246)$$

Согласно формуле приведения $\cos\left(\frac{\pi}{2} + \frac{\pi}{2v+2}\right) = -\sin \frac{\pi}{2v+2}$, тогда

$$\sum_{i=1}^{\frac{v+1}{2}} \sin \frac{i\pi}{v+1} = \frac{\cos \frac{\pi}{2v+2} + \sin \frac{\pi}{2v+2}}{2 \sin \frac{\pi}{2v+2}}. \quad (4.247)$$

Используя формулу $\frac{1}{2} + \sum_{i=1}^n \cos ix = \frac{\sin((n+\frac{1}{2})x)}{2 \sin \frac{x}{2}}$ [377, с. 2], где $n = \frac{v+1}{2}$, а $x = \frac{\pi}{v+1}$, получим

$$\sum_{i=1}^{\frac{v+1}{2}} \cos \frac{i\pi}{v+1} = \frac{\sin\left(\frac{\pi}{2} + \frac{\pi}{2v+2}\right)}{2 \sin \frac{\pi}{2v+2}} - \frac{1}{2}. \quad (4.248)$$

Согласно формуле приведения $\sin\left(\frac{\pi}{2} + \frac{\pi}{2v+2}\right) = \cos \frac{\pi}{2v+2}$, тогда

$$\sum_{i=1}^{\frac{v+1}{2}} \cos \frac{i\pi}{v+1} = \frac{\cos \frac{\pi}{2v+2}}{2 \sin \frac{\pi}{2v+2}} - \frac{1}{2}. \quad (4.249)$$

Так как $\sum_{i=1}^n \sin ix = \frac{\cos \frac{x}{2} - \cos((n+\frac{1}{2})x)}{2 \sin \frac{x}{2}}$ [377, с. 2], где $n = \frac{v-1}{2}$, а $x = \frac{\pi}{v+1}$, получим

$$\begin{aligned} \sum_{i=1}^{\frac{v-1}{2}} \sin \frac{i\pi}{v+1} &= \frac{\cos \frac{\pi}{2v+2} - \cos\left(\left(\frac{v-1}{2} + \frac{1}{2}\right) \frac{\pi}{v+1}\right)}{2 \sin \frac{\pi}{2v+2}} \\ &= \frac{\cos \frac{\pi}{2v+2} - \cos\left(\frac{\pi}{2} - \frac{\pi}{2v+2}\right)}{2 \sin \frac{\pi}{2v+2}}. \end{aligned} \quad (4.250)$$

Согласно формуле приведения $\cos\left(\frac{\pi}{2} - \frac{\pi}{2v+2}\right) = \sin \frac{\pi}{2v+2}$, тогда

$$\sum_{i=1}^{\frac{v-1}{2}} \sin \frac{i\pi}{v+1} = \frac{\cos \frac{\pi}{2v+2} - \sin \frac{\pi}{2v+2}}{2 \sin \frac{\pi}{2v+2}}. \quad (4.251)$$

Следовательно,

$$I_v = \operatorname{tg} \frac{\pi}{4v+4}. \quad (4.252)$$

Из (4.243) и (4.252) следует, что $\forall v \in \mathbb{N}: \|Q_{2v-1}^1(x)\| = 2I_v = 2 \operatorname{tg} \frac{\pi}{4v+4}$.

Так как $\forall v \in \mathbb{N}: I_{v-1} > I_v$, то наименьшее значение $\|Q_{2v-1}^1(x)\|$ при максимальном v . Рассмотрим вопрос о количестве нулей функции $U(x) = \frac{dQ_n^1(x)}{dx}$. Так как функция $Q_n^1(x)$ – нечетная непрерывная функция, то $U(x)$ четная функция. Количество нулей $U(x)$ меньше либо равно $n-1$ из них неотрицательных чисел меньше либо равно $\frac{n-1}{2}$ штук. Следовательно, количество решений уравнения $Q_n^1(x) = 1$, удовлетворяющих условию $x \in (0, 1]$, меньше либо равно $\frac{n-1}{2} + 1 = \frac{n+1}{2}$, т.е. $v \leq \frac{n+1}{2}$. Следовательно, если n – нечетное число, то максимальное значение $v = \frac{n+1}{2}$, если n – четное число, то максимальное значение $v = \frac{n}{2}$.

Используем интерполяционную формулу Лагранжа для вычисления значения $Q_{2v-1}^1(x)$, получим $Q_{2v-1}^1(x) = \sum_{i=1}^v l_i(x) - \sum_{i=1}^v \bar{l}_i(x)$, где

$$l_i(x) = \prod_{j=1}^v \frac{x + y_i}{y_i + y_j} \cdot \prod_{j=1, j \neq i}^v \frac{x - y_j}{y_i - y_j}, \quad (4.253)$$

$$\bar{l}_i(x) = - \prod_{j=1, j \neq i}^v \frac{x + y_j}{y_i - y_j} \cdot \prod_{j=1}^v \frac{x - y_j}{y_i + y_j}. \quad (4.254)$$

Тогда

$$\begin{aligned} Q_{2v-1}^1(x) &= \sum_{i=1}^v \left(\prod_{j=1}^v \frac{x + y_j}{y_i + y_j} \cdot \prod_{j=1, j \neq i}^v \frac{x - y_j}{y_i - y_j} + \prod_{j=1, j \neq i}^v \frac{x + y_j}{y_i - y_j} \cdot \prod_{j=1}^v \frac{x - y_j}{y_i + y_j} \right) \\ &= \sum_{i=1}^v \prod_{j=1, j \neq i}^v \frac{x + y_j}{y_i + y_j} \cdot \prod_{j=1, j \neq i}^v \frac{x - y_j}{y_i - y_j} \left(\frac{x + y_i}{2y_i} + \frac{x - y_i}{2y_i} \right) \\ &= x \sum_{i=1}^v \frac{1}{y_i} \prod_{j=1, j \neq i}^v \frac{x + y_j}{y_i + y_j} \cdot \prod_{j=1, j \neq i}^v \frac{x - y_j}{y_i - y_j} \\ &= x \sum_{i=1}^v \frac{1}{y_i} \prod_{j=1, j \neq i}^v \frac{x^2 - y_j^2}{y_i^2 - y_j^2}. \end{aligned} \quad (4.255)$$

Теорема доказана. \square

Из Теоремы 4.8.1 следует, что при $n \geq 1$ существует единственная нечетная функция наилучшего приближения функции знака числа, которая строится с помощью интерполяционной формулы Лагранжа, а узлы интерполяции являются нулями многочлена Чебышева второго рода.

Пример 4.8.2. Построим многочлены наилучшего приближения функции знака числа при $n = 3$ и $n = 4$, являющиеся нечетными функциями.

Если $n = 3$, то согласно Теореме 4.8.1 многочлен наилучшего приближения функции знака числа задается следующей формулой

$$\begin{aligned} Q_3^1(x) &= x \sum_{i=1}^2 \frac{1}{\sin \frac{i \cdot \pi}{6}} \prod_{j=1, j \neq i}^2 \frac{x^2 - \sin^2 \frac{j \cdot \pi}{6}}{\sin^2 \frac{i \cdot \pi}{6} - \sin^2 \frac{j \cdot \pi}{6}} \\ &= \frac{4\sqrt{3} - 12}{3} x^3 + \frac{9 - \sqrt{3}}{3} x. \end{aligned} \quad (4.256)$$

Если $n = 4$, то согласно Теореме 4.8.1 многочлен наилучшего приближения функции знака числа задается следующей формулой

$$\begin{aligned} Q_4^1(x) &= x \sum_{i=1}^2 \frac{1}{\sin \frac{i \cdot \pi}{6}} \prod_{j=1, j \neq i}^2 \frac{x^2 - \sin^2 \frac{j \cdot \pi}{6}}{\sin^2 \frac{i \cdot \pi}{6} - \sin^2 \frac{j \cdot \pi}{6}} \\ &= \frac{4\sqrt{3} - 12}{3} x^3 + \frac{9 - \sqrt{3}}{3} x. \end{aligned} \quad (4.257)$$

Обратим внимание на тот факт, что $Q_3^1(x) = Q_4^1(x)$. Этот факт можно выразить следующим обобщающим утверждением: если n – четное число и $n \geq 2$, то $Q_n^1(x) = Q_{n-1}^1(x)$.

4.8.5 Количество многочленов наилучшего приближения функции определения знака закодированного числа над полем \mathbb{R} являющихся функциями общего вида

Исследуем вопрос о существовании многочленов наилучшего приближения функции знака числа $Q_n(x)$, являющихся функциями общего вида при различных значениях n .

Теорема 4.8.2.

1. Если n – нечетное число, то не существует функций общего вида $Q_n(x)$, являющихся многочленами наилучшего приближения функции знака числа.
2. Если n – четное число, то существует несчетное множество функций общего вида $Q_n(x)$, являющихся многочленами наилучшего приближения функции знака числа.

Доказательство. Из Теоремы 4.8.1 следует, что существует единственная нечетная функция $Q_n^1(x)$ являющаяся многочленом наилучшего приближения функции знака числа. Покажем, что существует четная функция $Q_n^0(x) \neq 0$, такая, что $\|Q_n(x)\| = \|Q_n^1(x)\|$. Для этого вычислим $\|Q_n(x)\| - \|Q_n^1(x)\|$:

$$\begin{aligned} & \|Q_n(x)\| - \|Q_n^1(x)\| \\ &= \int_0^1 |1 - Q_n^0(x) - Q_n^1(x)| + |1 + Q_n^0(x) - Q_n^1(x)| - 2|1 - Q_n^1(x)| dx, \end{aligned} \quad (4.258)$$

где $Q_n(x) = Q_n^1(x) + Q_n^0(x)$, $Q_n^0(x)$ – четная функция, $Q_n^1(x)$ – нечетная функция.

Так как $\forall x \in \mathbb{R}$: $|1 - Q_n^0(x) - Q_n^1(x)| + |1 + Q_n^0(x) - Q_n^1(x)| - 2|1 - Q_n^1(x)| \geq 0$, то $\|Q_n(x)\| - \|Q_n^1(x)\|$ равна нулю тогда и только тогда, когда выполняется условие $\forall x \in [0, 1]$: $|1 - Q_n^0(x) - Q_n^1(x)| + |1 + Q_n^0(x) - Q_n^1(x)| - 2|1 - Q_n^1(x)| = 0$, что равносильно совокупности

$$\forall x \in [0, 1] \& Q_n^1(x) \leq 1 : \begin{cases} 1 - Q_n^0(x) - Q_n^1(x) \geq 0, \\ 1 + Q_n^0(x) - Q_n^1(x) \geq 0; \end{cases} \quad (4.259)$$

и

$$\forall x \in [0, 1] \& Q_n^1(x) \geq 1 : \begin{cases} 1 - Q_n^0(x) - Q_n^1(x) \leq 0, \\ 1 + Q_n^0(x) - Q_n^1(x) \leq 0; \end{cases} \quad (4.260)$$

следовательно, $\forall x \in [0, 1]$: $-|1 - Q_n^1(x)| \leq Q_n^0(x) \leq |1 - Q_n^1(x)|$.

Так как $Q_n^1(x)$ – нечетная функция наилучшего приближения функции знака числа, то из Теоремы 4.8.1 следует, что существуют точки $x_1, x_2, \dots, x_v \in (0, 1]$, такие что $\forall i = \overline{1, v}$: $Q_n^1(x_i) = 1$. Так как $Q_n^1(x)$ – нечетная функция наилучшего приближения функции знака числа, то из доказательства Теоремы 4.8.1 следует, что если n – нечетное число, то $v = \frac{n+1}{2}$, иначе $v = \frac{n}{2}$.

Подставляя x_1, x_2, \dots, x_v в неравенства $-|1 - Q_n^1(x)| \leq Q_n^0(x) \leq |1 - Q_n^1(x)|$, получим $\forall i = \overline{1, v}$: $0 \leq Q_n^0(x_i) \leq 0$, следовательно, сформулируем необходимое условие: $\forall i = \overline{1, v}$: $Q_n^0(x_i) = 0$. Так как $Q_n^0(x)$ – четная функция, то $\forall i = \overline{1, v}$: $Q_n^0(-x_i) = 0$, следовательно, $Q_n^0(x)$ делится на многочлен $\prod_{i=1}^v (x^2 - x_i^2)$ и $\deg Q_n^0(x) \geq 2v$. Рассмотрим два случая.

Случай 1. Если n – нечетное число, то $\deg Q_n^0(x) \geq 2v = n + 1$, следовательно, не существует многочлена, являющегося четной функцией, удовлетворяющего условию $\deg Q_n^0(x) \leq n$. Значит, если n – нечетное число, не существу-

ет многочлена наилучшего приближения функции знака числа, являющегося функцией общего вида.

Случай 2. Если n – четное число, то $\deg Q_n^0(x) \geq 2v = n$. С другой стороны, $\deg Q_n^0(x) \leq n$, следовательно, $\deg Q_n^0(x) = n$. Для построения многочлена $Q_n^0(x)$ рассмотрим многочлен вида

$$Z_n(x) = \frac{Q_n^1(x) - 1}{\prod_{i=1}^{n/2} (x - x_i)}, \quad (4.261)$$

где $\forall i = \overline{1, \frac{n}{2}}: x_i = \sin \frac{i\pi}{n+2}$.

Рассмотрим уравнение $Q_n^1(x) - 1 = 0$. $\forall i = \overline{1, \frac{n}{2}}: Q_n^1(x_i) - 1 = 0$, следовательно, согласно теореме Роля, в каждом из интервалов (x_i, x_{i+1}) существует хотя бы одна точка $\epsilon_i \in (x_i, x_{i+1})$, для которой $F(\epsilon_i) = 0$, где $F(x) = \frac{d(Q_n^1(x)-1)}{dx} = \frac{dQ_n^1(x)}{dx}$, $i \in \overline{1, \frac{n}{2} - 1}$.

Так как $Q_n^1(x)$ – нечетная функция, то $F(x)$ – четная функция, следовательно, $\forall i \in \overline{1, \frac{n}{2} - 1}: F(-\epsilon_i) = 0$. Учитывая, что $\deg F(x) = n - 2$, то, согласно основной теореме алгебры, уравнение $F(x) = 0$ над полем действительных чисел может иметь не больше $n - 2$ корней с учетом их кратности, значит $\pm\epsilon_i$ являются корнями кратности один.

Учитывая, что $\pm\epsilon_i$ – корни кратности один, функция $F(x)$ проходя через $\pm\epsilon_i$ меняет свой знак. Следовательно, $(-\infty, -\epsilon_{\frac{n}{2}-1})$, $(-\epsilon_{\frac{n}{2}-1}, -\epsilon_{\frac{n}{2}-2}), \dots, (-\epsilon_2, -\epsilon_1)$, $(\epsilon_1, \epsilon_2), \dots, (\epsilon_{\frac{n}{2}-2}, \epsilon_{\frac{n}{2}-1})$, $(\epsilon_{\frac{n}{2}-1}, +\infty)$ являются интервалами возрастания или убывания функции $Q_n^1(x)$. Значит, уравнение $Q_n^1(x) - 1 = 0$ на каждом из интервалов имеет не больше одного решения. Учитывая, что на каждом из интервалов $(\epsilon_{-1}, \epsilon_1)$, $(\epsilon_1, \epsilon_2), \dots, (\epsilon_{\frac{n}{2}-2}, \epsilon_{\frac{n}{2}-1})$, $(\epsilon_{\frac{n}{2}-1}, +\infty)$, согласно теореме Коши, уравнение имеет не больше чем одно решение, и решениями уравнения $Q_n^1(x) - 1 = 0$ являются соответственно $x_1, x_2, \dots, x_{\frac{n}{2}}$, то, следовательно, не существует $\phi \geq 0$ и $i = \overline{1, \frac{n}{2}}: \phi \neq x_i$ и $Q_n^1(\phi) - 1 = 0$.

Покажем, что x_i является корнем кратности один уравнения $Q_n^1(x) - 1 = 0$. Предположим, что существует k , для которого x_k – является корнем уравнения $Q_n^1(x) - 1 = 0$ кратности больше единицы, тогда x_k является также корнем уравнения $F(x) = 0$. Следовательно, уравнение $F(x) = 0$ имеет как минимум $n - 1$ корней $\forall i = \overline{1, \frac{n}{2}}: \pm\epsilon_i$ и x_k , при условии, что $\deg F(x) = n - 2$. Пришли к противоречию, следовательно, x_i являются корнями кратности один уравнения $Q_n^1(x) - 1 = 0$. Следовательно, если существует $\gamma \in \mathbb{R}$, для которой

выполняется условие $Z_n(\gamma) = 0$, то $\gamma < 0$ и выполняется одно из двух условий $\forall x \geq 0: Z_n(x) > 0$ или $Z_n(x) < 0$.

Так как $Z_n(0) = \frac{Q_n^1(0)-1}{\prod_{i=1}^{\frac{n}{2}}(-x_i)} = \frac{(-1)^{\frac{n}{2}+1}}{\prod_{i=1}^{\frac{n}{2}} x_i}$, то, если $\frac{n}{2}$ – четное число, то $\forall x \geq 0: Z_n(x) < 0$, иначе $\forall x \geq 0: Z_n(x) > 0$.

Рассмотрим функцию $R_n(x)$, заданную следующей формулой

$$R_n(x) = \frac{Z_n(x)}{\prod_{j=1}^{\frac{n}{2}}(x+x_j)}. \quad (4.262)$$

Функция $R_n(x)$ непрерывна на отрезке $[0, 1]$, согласно теореме Вейерштрасса, она ограничена, т.е. существуют $x_m^R, x_M^R \in [0, 1]$ такие, что $\forall x \in [0, 1]: R_n(x_m^R) \leq R(x) \leq R_n(x_M^R)$. Учитывая, что $\forall x \in [0, 1]: \prod_{j=1}^{\frac{n}{2}}(x+x_j) > 0$, получим, если $\frac{n}{2}$ – четное число, то $R_n(x_m^R) < R_n(x_M^R) < 0$, иначе $0 < R_n(x_m^R) < R_n(x_M^R)$. Выбирая в качестве τ значение по следующему правилу: если $\frac{n}{2}$ – четное число, $\tau = -R_n(x_M^R)$, иначе, $\tau = R_n(x_m^R)$, получим функцию $Q_n^0(x) = \tau \prod_{i=1}^{\frac{n}{2}}(x^2 - x_i^2)$, удовлетворяющую неравенству $\forall x \in [0, 1]: -|1 - Q_n^1(x)| \leq Q_n^0(x) \leq |1 - Q_n^1(x)|$.

Так как $Q_n(x) = Q_n^0(x) + Q_n^1(x)$ – многочлен наилучшего приближения функции знака числа, то из Следствия 4.8.2 следует, что $\forall \phi \in [0, \frac{\pi}{2}]: \|\sin^2 \phi \cdot Q_n(x) + \cos^2 \phi \cdot Q_n^1(x)\| \leq \|Q_n^1(x)\|$, значит $Q_{\phi,n}(x) = \sin^2 \phi \cdot Q_n(x) + \cos^2 \phi \cdot Q_n^1(x)$ является многочленом наилучшего приближения функции знака числа и $Q_{\phi,n}(x) = \sin^2 \phi \cdot Q_n(x) + \cos^2 \phi \cdot Q_n^1(x) = \sin^2 \phi \cdot Q_n^0(x) + Q_n^1(x)$. Стоит отметить, что и $\bar{Q}_n(x) = -Q_n^0(x) + Q_n^1(x)$ также является многочленом наилучшего приближения функции знака числа, значит $\bar{Q}_{\phi,n}(x) = \sin^2 \phi \cdot \bar{Q}_n(x) + \cos^2 \phi \cdot Q_n^1(x) = -\sin^2 \phi \cdot Q_n^0(x) + Q_n^1(x)$ является многочленом наилучшего приближения функции знака числа.

Теорема доказана. □

Пример 4.8.3. Построим многочлены наилучшего приближения функции знака числа при $n = 4$ общего вида.

Из примера 4.8.2 следует, что $Q_4^1(x) = \frac{4\sqrt{3}-12}{3}x^3 + \frac{9-\sqrt{3}}{3}x$, вычислим многочлен $Z_4(x)$

$$Z_4(x) = \frac{Q_4^1(x) - 1}{(x - \frac{1}{2})(x - \frac{\sqrt{3}}{2})} = \frac{4\sqrt{3} - 12}{3}x - \frac{4\sqrt{3}}{3}. \quad (4.263)$$

Вычислим функцию $R_4(x)$, получим

$$R_4(x) = \frac{Z_4(x)}{\left(x + \frac{1}{2}\right) \left(x + \frac{\sqrt{3}}{2}\right)} = \frac{\frac{4\sqrt{3}}{3}}{x + \frac{\sqrt{3}}{2}} - \frac{4}{x + \frac{1}{2}}. \quad (4.264)$$

Вычислим производную функции $R_4(x)$, получим

$$\frac{dR_4(x)}{dx} = -\frac{\frac{4\sqrt{3}}{3}}{\left(x + \frac{\sqrt{3}}{2}\right)^2} + \frac{4}{\left(x + \frac{1}{2}\right)^2}. \quad (4.265)$$

Так как на отрезке $[0, 1]$ критических точек нет, то, следовательно, максимальное и минимальное значение функция $R_4(x)$ принимает на концах отрезка. Вычислим $R_4(0)$ и $R_4(1)$, получим, соответственно, $R_4(0) = -\frac{16}{3}$ и

$$R_4(1) = -\frac{16}{6 + 3\sqrt{3}}. \quad (4.266)$$

Следовательно, $\tau = \frac{16}{6+3\sqrt{3}}$ и в качестве

$$Q_{4,\mu}^0 = \mu \left(x^4 - x^2 + \frac{3}{16} \right) = \mu \cdot \frac{U_5(x)}{32 \cdot x}, \quad (4.267)$$

где μ – любое число, удовлетворяющее условию $\mu \in [-\tau, \tau]$, а $U_5(x)$ – многочлен Чебышева второго рода. Таким образом, многочлен наилучшего приближения имеет вид $Q_{4,\mu}(x) = \mu \cdot x^4 + \frac{4\sqrt{3}-12}{3} \cdot x^3 - \mu \cdot x^2 + \frac{9-\sqrt{3}}{3} \cdot x + \frac{3}{16} \cdot \mu$.

Лемма 4.8.3. Если n – четное число, то $\forall i = \overline{1, \frac{n}{2}}$:

$$\alpha_i = \prod_{j=1, j \neq i}^{\frac{n}{2}} \left(\sin^2 \frac{i\pi}{n+1} - \sin^2 \frac{j\pi}{n+2} \right) = \frac{(-1)^{\frac{n}{2}-i}}{2^n \cdot \sin^2 \frac{2i\pi}{n+2}}. \quad (4.268)$$

Доказательство. Вычислим $\alpha_i = \prod_{j=1, j \neq i}^{\frac{n}{2}} \left(\sin^2 \frac{i\pi}{n+2} - \sin^2 \frac{j\pi}{n+2} \right)$, используя выражение $\sin^2 x - \sin^2 y = \sin(x-y) \cdot \sin(x+y)$, получим

$$\alpha_i = \prod_{j=1, j \neq i}^{\frac{n}{2}} \sin \frac{(i+j)\pi}{n+2} \sin \frac{(i-j)\pi}{n+2}. \quad (4.269)$$

Рассмотрим два случая.

Случай 1: если $i = \frac{n}{2}$, то

$$\alpha_{\frac{n}{2}} = \prod_{j=1}^{\frac{n}{2}-1} \sin \frac{\left(\frac{n}{2} + j\right)\pi}{n+2} \sin \frac{\left(\frac{n}{2} - j\right)\pi}{n+2} = \frac{\prod_{j=1}^{n-1} \sin \frac{j\pi}{n+2}}{\sin \frac{n\pi}{2n+4}}. \quad (4.270)$$

Учитывая, что $\prod_{j=1}^{n+1} \sin \frac{j\pi}{n+2} = \frac{n+2}{2^{n+1}}$,

$$\alpha_{\frac{n}{2}} = \frac{n+2}{2^n \sin^2 \frac{2\pi}{n+2}}. \quad (4.271)$$

Случай 2: если $i \neq \frac{n}{2}$, то

$$\alpha_i = \frac{1}{\sin \frac{2i\pi}{n+2} \sin \frac{i\pi}{n+2}} \prod_{j=i-\frac{n}{2}}^{-1} \sin \frac{j\pi}{n+2} \prod_{j=1}^{i+\frac{n}{2}} \sin \frac{j\pi}{n+2}. \quad (4.272)$$

Учитывая, что $\sin \frac{j\pi}{n+2} = -\sin \frac{(n+2+j)\pi}{n+2}$, то

$$\begin{aligned} \alpha_i &= \frac{(-1)^{\frac{n}{2}-i}}{\sin \frac{2i\pi}{n+2} \sin \frac{i\pi}{n+2}} \prod_{j=i-\frac{n}{2}}^{-1} \sin \frac{(n+2+j)\pi}{n+2} \prod_{j=1}^{i+\frac{n}{2}} \sin \frac{j\pi}{n+2} \\ &= \frac{(-1)^{\frac{n}{2}-i}}{\sin \frac{2i\pi}{n+2} \sin \frac{i\pi}{n+2}} \prod_{j=\frac{n}{2}+i+2}^{n+1} \sin \frac{j\pi}{n+2} \prod_{j=1}^{i+\frac{n}{2}} \sin \frac{j\pi}{n+2} \\ &= \frac{(-1)^{\frac{n}{2}-i}}{\sin \frac{2i\pi}{n+2} \sin \frac{i\pi}{n+2} \sin \frac{(\frac{n}{2}+i+1)\pi}{n+2}} \prod_{j=1}^{n+1} \sin \frac{j\pi}{n+2}. \end{aligned} \quad (4.273)$$

Так как $\prod_{j=1}^{n+1} \sin \frac{j\pi}{n+2} = \frac{n+2}{2^{n+1}}$ и $\sin \frac{(\frac{n}{2}+i+1)\pi}{n+2} = \cos \frac{i\pi}{n+2}$, то

$$\alpha_i = \frac{(-1)^{\frac{n}{2}-i}}{\sin^2 \frac{2i\pi}{n+2}} \cdot \frac{n+2}{2^n}. \quad (4.274)$$

□

Лемма доказана.

Теорема 4.8.3. Если n – четное число, то многочлены наилучшего приближения функции определения знака числа, являющиеся функциями общего вида, задаются следующей формулой

$$Q_{\mu,n}(x) = \mu \cdot \prod_{i=1}^{\frac{n}{2}} (x^2 - x_i^2) + Q_n^1(x), \quad (4.275)$$

где $Q_n^1(x)$ – многочлен наилучшего приближения функции определения знака числа, являющийся нечетной функцией, $\mu \in [-\tau, 0) \cup (0, \tau]$, $\forall i = \overline{1, \frac{n}{2}}$: $x_i = \sin \frac{i\pi}{n+2}$ и $0 < \tau \leq \frac{2^{n+1}}{n+2} \operatorname{tg} \frac{\pi}{2n+4}$.

Доказательство. Используя теорему о разложении рациональных функций, имеющих различные корни [225], представим $R_n(x)$ в виде элементарных дробей первого типа

$$R_n(x) = \frac{Z_n(x)}{\prod_{j=1}^{\frac{n}{2}} (x + x_j)} = \sum_{j=1}^{\frac{n}{2}} \frac{b_j}{x + x_j}, \quad (4.276)$$

где $\forall j = \overline{1, \frac{n}{2}}: b_j \in \mathbb{R}$.

Из (4.276) следует, что

$$Z_n(x) = \sum_{j=1}^{\frac{n}{2}} b_j \prod_{i=1, i \neq j}^{\frac{n}{2}} (x + x_i). \quad (4.277)$$

Вычислим $\forall j = \overline{1, \frac{n}{2}}$ значения $Z_n(-x_j)$

$$Z_n(-x_j) = b_j \prod_{i=1, i \neq j}^{\frac{n}{2}} (x_i - x_j). \quad (4.278)$$

С другой стороны, вычислим $Z_n(-x_j)$, используя формулу (4.261), получим

$$Z_n(-x_j) = \frac{-2}{\prod_{i=1}^{\frac{n}{2}} (-x_j - x_i)} = (-1)^{\frac{n}{2}+1} \cdot \frac{2}{\prod_{i=1}^{\frac{n}{2}} (x_i + x_j)}. \quad (4.279)$$

Выразим b_j , используя (4.278) и (4.279)

$$b_j = (-1)^{\frac{n}{2}+1} \cdot \frac{1}{x_j \prod_{i=1, i \neq j}^{\frac{n}{2}} (x_i^2 - x_j^2)} = \frac{1}{x_j \prod_{i=1, i \neq j}^{\frac{n}{2}} (x_i^2 - x_j^2)}. \quad (4.280)$$

Используя Лемму 4.8.3, вычислим значение b_j

$$b_j = (-1)^{\frac{n}{2}+j} \cdot \frac{2^n}{n+2} \cdot \frac{x_{2j}^2}{x_j}. \quad (4.281)$$

Следовательно, $R_n(x)$ можно представить в следующем виде

$$\begin{aligned} R_n(x) &= (-1)^{\frac{n}{2}} \frac{2^n}{n+2} \cdot \sum_{j=1}^{\frac{n}{2}} (-1)^j \frac{x_{2j}^2}{x_j} \cdot \frac{1}{x + x_j} \\ &= (-1)^{\frac{n}{2}} \frac{2^{n+2}}{n+2} \cdot \sum_{j=1}^{\frac{n}{2}} (-1)^j \frac{x_j - x_j^3}{x + x_j}. \end{aligned} \quad (4.282)$$

Вычислим $R(1)$ и $R_n(0)$, используя (4.282), получим

$$R_n(0) = (-1)^{\frac{n}{2}} \frac{2^{n+2}}{n+2} \sum_{j=1}^{\frac{n}{2}} (-1)^j (1 - x_j^2), \quad (4.283)$$

$$\begin{aligned} R_n(1) &= (-1)^{\frac{n}{2}} \frac{2^{n+2}}{n+2} \cdot \sum_{j=1}^{\frac{n}{2}} (-1)^j (x_j - x_j^2) \\ &= (-1)^{\frac{n}{2}} \frac{2^{n+2}}{n+2} \cdot \left(\sum_{j=1}^{\frac{n}{2}} (-1)^j x_j - \sum_{j=1}^{\frac{n}{2}} (-1)^j x_j^2 \right). \end{aligned} \quad (4.284)$$

Учитывая, что

$$\sum_j^{\frac{n}{2}} (-1)^j = \frac{-1 + (-1)^{\frac{n}{2}}}{2}, \quad (4.285)$$

$$\sum_{j=1}^{\frac{n}{2}} (-1)^j x_j^2 = \frac{(-1)^{\frac{n}{2}}}{2}, \quad (4.286)$$

$$\sum_{j=1}^{\frac{n}{2}} (-1)^j x_j = \frac{(-1)^{\frac{n}{2}}}{2} - \frac{1}{2} \operatorname{tg} \frac{\pi}{2n+4}, \quad (4.287)$$

получим

$$R_n(0) = (-1)^{\frac{n}{2}+1} \frac{2^{n+1}}{n+2}, \quad (4.288)$$

$$R_n(1) = (-1)^{\frac{n}{2}+1} \frac{2^{n+1}}{n+2} \operatorname{tg} \frac{\pi}{2n+4}. \quad (4.289)$$

Так как $\forall n \geq 2$ и $n \bmod 2 \equiv 0$: $|R_n(0)| > |R_n(1)|$, то $0 < \tau \leq |R_n(1)| = \frac{2^{n+1}}{n+2} \operatorname{tg} \frac{\pi}{2n+4}$.

Теорема доказана. \square

4.9 Нейросетевой метод определения знака закодированного числа над полем \mathbb{R}

Нейронные Сети Прямого Распространения (НСПР) с одним скрытым слоем, математически описываются следующим выражением

$$y_n(x) = \sum_{j=0}^n c_j \cdot \sigma(\langle a_j \cdot x \rangle + b_j), \quad (4.290)$$

где $x \in \mathbb{R}^s$, $s \in \mathbb{N}$, $\forall j = \overline{1, n}$: $b_j \in \mathbb{R}$ – пороги, $a_j \in \mathbb{R}^s$ – веса связи, $c_j \in \mathbb{R}$ – коэффициенты, $\langle a_j \cdot x \rangle$ – скалярное произведение a_j и x , а σ – функция активации [136]. Во многих фундаментальных моделях НСПР в качестве функции активации используется сигмоидальная функция.

НСПР являются универсальными аппроксиматорами. Теоретически любую непрерывную функцию, определенную на компактном множестве, можно аппроксимировать с любой желаемой степенью точности путем увеличения количества скрытых нейронов. Cybenko [186] и Funahashi [210] доказали, что любую непрерывную функцию можно аппроксимировать на компактном множестве с равномерной топологией сетью вида (4.290) с использованием непрерывной сигмоидальной функции активации. Hornik и др. [240] показали, что с помощью такой сети можно приблизиться к любой измеримой функции. Кроме того, в работе [241] авторы доказали, что к любой функции из пространств Соболева можно приблизиться со всеми производными. Различные результаты плотности для НСПР-аппроксимаций функций от многих переменных были получены позже многими авторами с использованием методов для различных частных случаев: Leshno и др. в работе [287] 1993 года, Chen и Chen в работе [176] 1995 года, Nahm и Hong в работе [229] 2004 года и т.д.

Важное значение имеет проблема сложности: определение количества нейронов, необходимых для гарантии того, что все функции (принадлежащие определенному классу) могут быть аппроксимированы с заданной степенью точности ϵ . Например, классический результат Barron [156] показывает, что если предполагается, что функция удовлетворяет определенным условиям, выраженным в терминах ее преобразования Фурье, и если каждый из нейронов имеет сигмоидальную функцию активации, то не более $O(\epsilon^{-2})$ нейронов необходимо для достижения порядка приближения ϵ . Ранее некоторые авторы опубликовали аналогичные результаты о сложности аппроксимаций НСПР: Suzuki в работе [350] 1998 года, Maiorov и Meir в работе [275] 1998 года, Ferrari и Stengel в работе [208] 2005 года, Chen и Cao [177], Makovoz в работе [276] 1998 года т.д.

Если использовать (4.290) для аппроксимации функции знака числа, то неопределенность результата в окрестности точки $x = 0$ сохраняется (рис. 4.6б)). Из результатов, представленных на рисунке 4.6, можно сделать вывод о том, что НСПР, используемые для аппроксимации функции знака числа, имеют тот же недостаток, что и аппроксимирующие многочлены: определе-

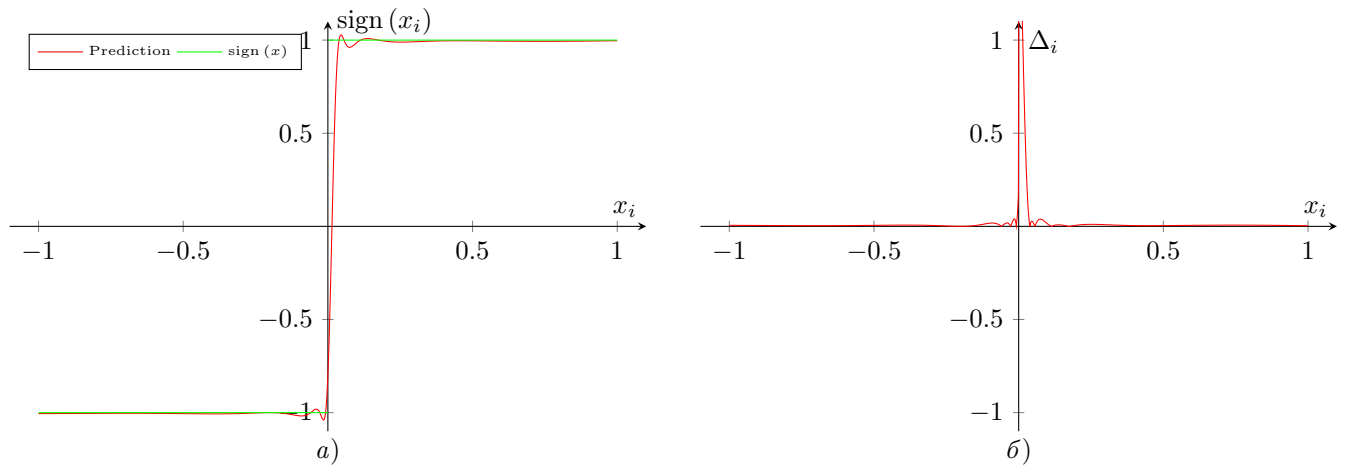


Рисунок 4.6 — Аппроксимация функции знака числа с помощью НСПР

ние знака в окрестности точки $x = 0$ реализуется с ошибкой, что затрудняет использование НСПР на практике.

Для устранения этого недостатка предложим нейросетевой метод определения знака числа.

Заметим, что $\forall a > 0$: $\lim_{x \rightarrow +\infty} \frac{2 \operatorname{arctg}(a \cdot x)}{\pi} = 1$ и $\lim_{x \rightarrow -\infty} \frac{2 \operatorname{arctg}(a \cdot x)}{\pi} = -1$, следовательно, функция знака числа является горизонтальной асимптотой функции $\frac{2 \operatorname{arctg}(a \cdot x)}{\pi}$, значит для аппроксимации функции знака числа можно использовать функции $\frac{2 \operatorname{arctg}(a \cdot x)}{\pi}$, т.е.

$$\operatorname{sign}(x) \approx \frac{2 \operatorname{arctg}(a \cdot x)}{\pi}, \quad (4.291)$$

где $a > 0$ – константа, которая позволяет регулировать точность аппроксимации функции знака числа, заданной над дискретным множеством.

Рассмотрим вопрос выбора константы a . Так как мы рассматриваем задачу аппроксимации функции знака числа на дискретном множестве, то можно поставить в соответствие каждому из элементов множества определения функции знака числа элемент множества натуральных чисел.

Разобьем множество определения функции знака числа на три части.

1. Если $x > 0$, то через t_1 обозначим минимальное x из области определения функции знака числа, удовлетворяющее условию $x > 0$. Учитывая, что точки в области определения являются равноотстоящими, $\forall i \in \mathbb{N}$: $t_i = t_1 + h \cdot (i - 1)$, где h – расстояние между двумя ближайшими точками области определения. Так как существует конечный предел $\lim_{n \rightarrow +\infty} \frac{2 \operatorname{arctg}(t_n)}{\pi} = 1$, то, согласно определению предела, какое бы ни было число $\epsilon > 0$, найдется такой номер $N_t(\epsilon)$, что для любого $n > N_t$

- всегда имеет место неравенство $\left| \frac{2 \operatorname{arctg}(t_n)}{\pi} - 1 \right| < \epsilon$. Таким образом, для обеспечения точности аппроксимации функции знака числа с помощью функции $\frac{2 \operatorname{arctg}(t_n)}{\pi}$, необходимо и достаточно в качестве константы a выбрать число равное $a = \frac{t_{N_t}}{t_1}$.
2. Если $x = 0$, то $\frac{2 \operatorname{arctg}(t_n)}{\pi} = 0$.
 3. Если $x < 0$, то через k_1 обозначим наибольшее x из области определения функции знака числа, удовлетворяющее условию $x < 0$. Учитывая, что точки в области определения являются равноотстоящими, то $\forall i \in \mathbb{N}: k_i = k_1 - h \cdot (i - 1)$. Так как существует конечный предел $\lim_{n \rightarrow +\infty} \frac{2 \operatorname{arctg}(k_n)}{\pi} = -1$, то, согласно определению предела, какое бы ни было число $\epsilon > 0$, найдется такой номер $N_k(\epsilon)$, что для любого $n > N_k$ всегда имеет место неравенство $\left| \frac{2 \operatorname{arctg}(k_n)}{\pi} + 1 \right| < \epsilon$.

Так как $\forall i \in \mathbb{N}: k_i = -t_i$, то $\left| \frac{2 \operatorname{arctg}(k_n)}{\pi} + 1 \right| < \epsilon$ равносильно неравенству $\left| \frac{2 \operatorname{arctg}(t_n)}{\pi} - 1 \right| < \epsilon$, следовательно, $N_t = N_k$. Таким образом, для обеспечения точности аппроксимации функции знака числа с помощью функции $\frac{2 \operatorname{arctg}(a \cdot x)}{\pi}$, необходимо и достаточно в качестве константы a выбрать число равное $a = \frac{k_{N_k}}{k_1} = \frac{t_{N_t}}{t_1}$.

Рассмотрим на примере аппроксимацию функции знака числа, заданную в точках $\forall i = \overline{1, 2001}: x_i = -1 + (i - 1)/1000$, с точностью $\epsilon = 10^{-3}$. Из заданной области определения функции знака числа следует, что $t_1 = 1/1000$, $h = 1/1000$ и $N_t = 636620$ (так как $1 - \frac{2 \operatorname{arctg}(t_{636620})}{\pi} \approx 0.0009999988 < \epsilon$ и $1 - \frac{2 \operatorname{arctg}(t_{636619})}{\pi} \approx 0.0010000004 > \epsilon$), значит $a = \frac{t_{N_t}}{t_1} = 636620$.

Обратим внимание на то, что $t_1 = h$, следовательно, $t_i = h \cdot i$. Подставляя в формулу $a = \frac{k_{N_k}}{k_1} = \frac{t_{N_t}}{t_1}$, получим $a = N_k = N_t$. Таким образом, можно сделать вывод о том, что $\left| \operatorname{sign}(t_i) - \frac{2 \operatorname{arctg}(a \cdot h)}{\pi} \right| \leq \epsilon$, если a удовлетворяет неравенству

$$1 - \frac{2 \operatorname{arctg}(a \cdot h)}{\pi} \leq \epsilon, \quad (4.292)$$

что равносильно

$$\frac{\pi}{2} \cdot (1 - \epsilon) \leq \operatorname{arctg}(a \cdot h). \quad (4.293)$$

Так как $h > 0$ и $\operatorname{arctg}(a \cdot h)$ – возрастающая функция, то

$$a \geq \frac{1}{h} \cdot \operatorname{tg} \left(\frac{\pi}{2} \cdot (1 - \epsilon) \right). \quad (4.294)$$

Таким образом, если в качестве a рассматривать действительные числа, то можно выбрать значение $a = \frac{1}{h} \cdot \operatorname{tg} \left(\frac{\pi}{2} \cdot (1 - \epsilon) \right)$. Если рассматривать целые числа, то $a = \left\lceil \frac{1}{h} \cdot \operatorname{tg} \left(\frac{\pi}{2} \cdot (1 - \epsilon) \right) \right\rceil$.

На рисунках 4.7а) и 4.7б) представлены результат аппроксимации функции знака числа, заданной на множестве точек $\forall i = \overline{1, 2001}$: $x_i = -1 + \frac{i-1}{1000}$ с помощью функции $\frac{2 \operatorname{arctg}(636620 \cdot x)}{\pi}$ и погрешность аппроксимации соответственно. Из результатов, представленных на рисунке 4.7б) можно сделать вывод о

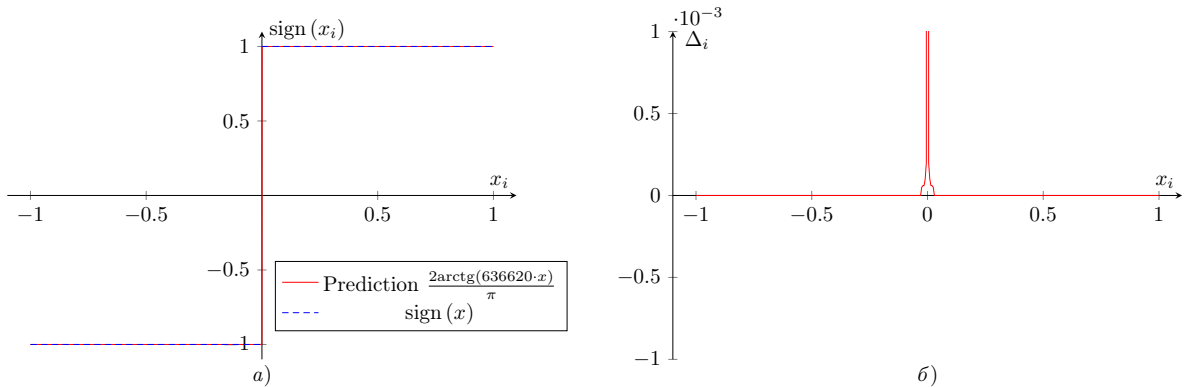


Рисунок 4.7 — Аппроксимация функции знака числа, заданной на множестве точек $\forall i = \overline{1, 2001}$: $x_i = -1 + \frac{i-1}{1000}$ с помощью функции $\frac{2 \operatorname{arctg}(636620 \cdot x)}{\pi}$

том, что максимальная погрешность аппроксимации функции знака числа с помощью функции $\frac{2 \operatorname{arctg}(636620 \cdot x)}{\pi}$ не превышает 10^{-3} . Причем, стоит заметить, что предлагаемый метод позволяет контролировать погрешность не только в окрестности, но и на всей области определения функции знака числа.

Таким образом, аппроксимацию функции знака числа удалось свести к аппроксимации функции $\operatorname{arctg} x$. Рассмотрим вопрос аппроксимации функции $\operatorname{arctg} x$ с помощью НСПР заданной формулой (4.290). Результаты аппроксимации представлены на рисунке 4.8. Результаты, представленные на рисунке 4.8б), указывают на то, что максимальная погрешность аппроксимации $\operatorname{arctg} x$ с областью определения $[-636620, 636620]$ равна 0.05 и достигается в окрестности точки $x = 0$. Следовательно, согласно теории погрешностей, максимальная погрешность аппроксимации функции знака числа с помощью функции $\operatorname{arctg} x$ и НСПР, заданной (4.290), для данного примера не превосходит значения $0.001 + \frac{2}{\pi} \cdot 0.05 \approx 0.033$ в окрестности точки $x = 0$.

Данный результат показывает, что предлагаемый метод позволяет повысить точность определения знака числа в окрестности точки $x = 0$ более чем в 15.1 раза.

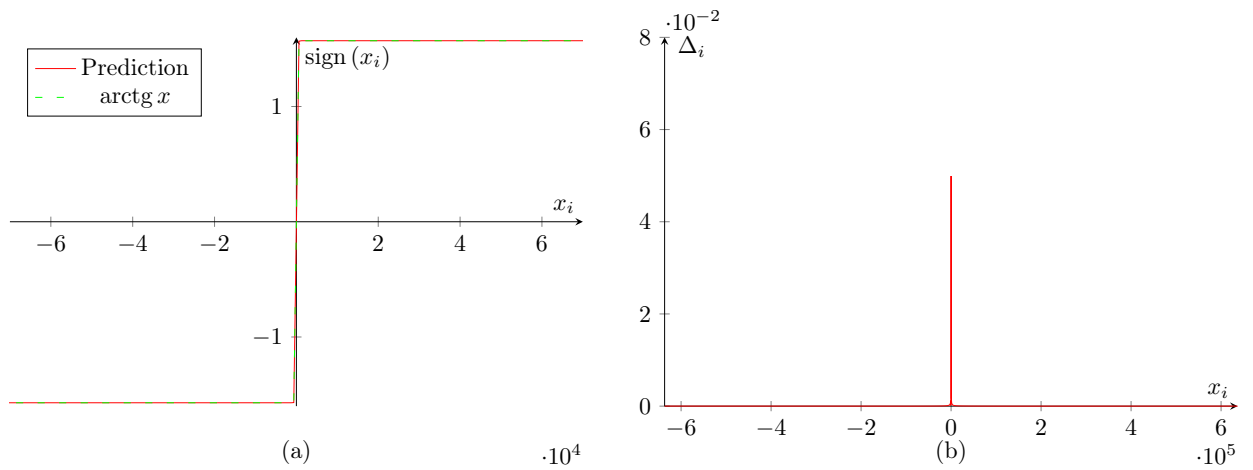


Рисунок 4.8 — Аппроксимация функции $\text{arctg } x$ с помощью НСПР заданной формулой (4.290)

Для использования данного метода в гомоморфных вычислениях предлагается при его реализации заменить сигмоидальную функцию активации на многочлен $f(x) = x - \frac{1}{3} \cdot x^3 + \frac{2}{15} \cdot x^5 - \frac{17}{315} \cdot x^7$. Таким образом, реализация операции определения знака числа над полем действительных чисел с заданной точностью возможна при глубине умножения равной 5.

4.10 Выводы по четвертой главе

В четвертой главе исследованы различные подходы к выполнению операций определения знака и сравнения чисел, оптимизированы известные и разработаны новые методы и алгоритмы реализации указанных операций, позволяющие повысить их эффективность в контексте гомоморфных вычислений над полем.

Гомоморфные вычисления над полем принято делить на два класса: целочисленные и вещественные, по формату обрабатываемых цифровых данных. Соответственно задачи определения знака числа и сравнения чисел следует рассматривать над полем \mathbb{Z}_m и над полем \mathbb{R} .

Построены многочлены, использующие интерполяционные многочлены Лагранжа, позволяющие определять знак числа и сравнивать числа над полем \mathbb{Z}_m . Вычислительная сложность алгоритмов определения знака и сравнения чисел при использовании целочисленных гомоморфных вычислений зависит от количества арифметических операций сложения и умножения, которые

необходимо выполнить для вычисления интерполяционного многочлена. Целочисленные гомоморфные вычисления поддерживают ограниченное количество умножений, поэтому от мультипликативной глубины алгоритма, реализующего ту или иную операцию, во многом зависит производительность системы. Мультипликативная глубина вычисления многочлена зависит от его степени, что было показано в работе [203] на примере алгоритма Paterson–Stockmeyer. Доказана теорема, дающая оценку степени интерполяционного многочлена функции определения знака числа: показано, что степень многочлена равна $m - 2$. Доказана теорема, уточняющая оценку степени интерполяционного многочлена функции сравнения чисел: показано, что степень многочлена из работы [199], равная $2m - 2$, может быть уточнена до m .

Для аппроксимации функции определения знака числа над полем \mathbb{R} исследована проблема построения многочлена наилучшего приближения указанной функции. Показано, что если степень аппроксимирующего многочлена $n = 0$, то многочленами наилучшего приближения являются $Q_n(x) = a_0$, где $|a_0| \leq 1$. Доказано, что если степень аппроксимирующего многочлена $n \geq 1$, то не существует многочленов наилучшего приближения, являющихся четными функциями. Если степень аппроксимирующего многочлена $n \geq 1$, то существует единственный многочлен наилучшего приближения, являющийся нечетной функцией, который строится с помощью интерполяционной формулы Лагранжа, где в качестве узлов интерполяции используются нули многочлена Чебышева второго рода. Доказано, что если $n \geq 1$ и n – нечетное число, то не существует многочленов наилучшего приближения, являющихся функциями общего вида. Если $n \geq 1$ и n – четное число, то существует несчетное множество многочленов наилучшего приближения, являющихся функциями общего вида. Для каждого рассмотренного случая построены аппроксимирующие многочлены и доказано, что каждый из них является многочленом наилучшего приближения. Для случаев, когда многочлена наилучшего приближения не существует, так же доказаны соответствующие теоремы.

Предложен модифицированный нейросетевой метод определения знака числа над полем \mathbb{R} , позволяющий более чем в 15.1 раза повысить точность указанной операции в окрестности проблемной точки $x = 0$.

Глава 5. ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ АЛГОРИТМОВ ОБНАРУЖЕНИЯ И ИСПРАВЛЕНИЯ АРИФМЕТИЧЕСКИХ ОШИБОК ОБРАБОТКИ ЗАКОДИРОВАННЫХ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ СВОЙСТВ РАНГА ЧИСЛА

Проблема ускорения вычислительно сложных операций в RNS, таких как определение знака числа, сравнение чисел, расширение оснований и перевод чисел из RNS в позиционную систему счисления, может решаться за счет уменьшения вычислительной сложности алгоритма определения ранга числа. Однако, основным приложением функции ранга числа, представленного в RNS, являются алгоритмы обнаружения и исправления ошибок арифметических вычислений, и от эффективности его вычисления во многом зависит производительность указанных алгоритмов. Чтобы добиться приемлемой задержки, требуется разработать такие алгоритмы вычисления ранга числа, представленного в RNS $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$, которые бы позволили вычислять значение ранга за время модулярного суммирования не более чем n вычетов по модулю q . При этом подразумевается, что величина модуля q приблизительно одного порядка с величиной модулей RNS p_i [130].

Учитывая, что в RNS можно эффективно реализовать операции сложения и умножения чисел в \mathbb{Z}_p , одним из способов решения проблемы эффективного вычисления ранга могло бы стать представление функции ядра в виде алгебраического многочлена над \mathbb{Z}_p . Ниже показано, что функцию ядра, нормализованную функцию ядра и функцию ядра Акушского нельзя вычислить с помощью алгебраического многочлена над \mathbb{Z}_p .

Исследуем вопрос аппроксимации функции ранга числа с помощью приближенного метода [26, 28] и докажем ряд арифметических свойств функции ранга числа.

5.1 Ранг числа и его свойства

Различают три формы представления Китайской теоремы об остатках, каждой из которых соответствует позиционная характеристика числа, представленного в RNS.

Первая форма

$$X = \left| \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \cdot x_i \right|_P = \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \cdot x_i - r(X) \cdot P, \quad (5.1)$$

где $r(X) = \left| \sum_{i=1}^n \frac{1}{p_i} \cdot |P_i^{-1}|_{p_i} \cdot x_i \right|$ – ранг числа.

Вторая форма

$$X = \left| \sum_{i=1}^n P_i \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right|_P = \sum_{i=1}^n P_i \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \hat{r}(X) \cdot P, \quad (5.2)$$

где $\hat{r}(X) = \left| \sum_{i=1}^n \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right|$ – нормализованный ранг числа.

Третья форма

$$C(X) \equiv \left| \sum_{i=1}^n c_i \cdot x_i \right|_{C_P} = \sum_{i=1}^n c_i \cdot x_i - \check{r}(X) \cdot C_P, \quad (5.3)$$

где $\check{r} = \left| \frac{\sum_{i=1}^n c_i \cdot x_i}{C_P} \right|$ – ранг числа функции ядра Акушского.

Свойство 5.1.1. $\hat{r}(X) = -\frac{X}{P} + \sum_{i=1}^n \frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i}}{p_i}$.

Доказательство. По определению

$$\hat{r}(X) = \left| \sum_{i=1}^n \frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i}}{p_i} \right| = \left[\frac{1}{P} \sum_{i=1}^n \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \cdot P_i \right].$$

Так как $\left[\frac{X}{P} \right] = \frac{X}{P} - \frac{|X|_P}{P}$, то

$$\hat{r}(X) = \frac{1}{P} \sum_{i=1}^n \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \cdot P_i - \frac{1}{P} \cdot \left| \sum_{i=1}^n \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \cdot P_i \right|_P.$$

Согласно Китайской теоремы об остатках, $\left| \sum_{i=1}^n \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \cdot P_i \right|_P = X$, следовательно, $\hat{r}(X) = \frac{1}{P} \sum_{i=1}^n \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \cdot P_i - \frac{X}{P}$.

Свойство доказано. \square

Свойство 5.1.2. $\hat{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i}$.

Доказательство. Из Свойства 5.1.1 напрямую следует, что $\hat{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i}$.

Свойство доказано. \square

Третья форма является обобщением позиционных характеристик: диагональной функции и функции Pirlo и Impedovo.

Теорема 5.1.1. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$, число $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и веса функции ядра Акушского $\bar{w}_1, \bar{w}_2, \dots, \bar{w}_n$ удовлетворяющие условию $0 \leq X < P$, тогда

$$\check{r}(X) = r(X) + \left\lfloor \frac{C(X)}{C_P} \right\rfloor. \quad (5.4)$$

Доказательство. Вычислим c_i , получим

$$c_i = C(B_i) = \sum_{j=1}^n \bar{w}_j \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot P_i}{p_j} \right\rfloor. \quad (5.5)$$

Так как $\forall i \neq j: |P_i^{-1}|_{p_i} \cdot P_i \equiv 0 \pmod{p_j}$ и $\forall i: |P_i^{-1}|_{p_i} \cdot P_i \equiv 1 \pmod{p_i}$, то для $i \neq j: \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot P_i}{p_j} \right\rfloor = \frac{|P_i^{-1}|_{p_i} \cdot P_i}{p_j}$, а для $i = j: \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot P_i}{p_i} \right\rfloor = \frac{|P_i^{-1}|_{p_i} \cdot P_i - 1}{p_i}$, следовательно, коэффициент c_i можно представить в следующем виде

$$c_i = |P_i^{-1}|_{p_i} \cdot P_i \cdot \sum_{j=1}^n \frac{\bar{w}_j}{p_j} - \frac{\bar{w}_i}{p_i}. \quad (5.6)$$

Учитывая, что $\sum_{j=1}^n \frac{\bar{w}_j}{p_j} = \frac{C_P}{P}$, то (5.6) преобразуется к виду

$$c_i = |P_i^{-1}|_{p_i} \cdot P_i \cdot \frac{C_P}{P} - \frac{\bar{w}_i}{p_i}. \quad (5.7)$$

Подставим (5.7) в (5.3), получим

$$\check{r}(X) = \left\lfloor \frac{\sum_{i=1}^n c_i \cdot x_i}{C_P} \right\rfloor = \left\lfloor \frac{1}{P} \cdot \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \cdot x_i - \frac{1}{C_P} \cdot \sum_{i=1}^n \frac{x_i \cdot \bar{w}_i}{p_i} \right\rfloor. \quad (5.8)$$

Подставляя (5.1) в (5.8), получим

$$\check{r}(X) = \left[r(X) + \frac{X}{P} - \frac{1}{C_P} \cdot \sum_{i=1}^n \frac{x_i \cdot \bar{w}_i}{p_i} \right]. \quad (5.9)$$

Учитывая, что

$$\begin{aligned} \sum_{i=1}^n \frac{x_i \cdot \bar{w}_i}{p_i} &= \sum_{i=1}^n \frac{\left(X - p_i \cdot \left\lfloor \frac{X}{p_i} \right\rfloor \right) \cdot \bar{w}_i}{p_i} = X \cdot \sum_{i=1}^n \frac{\bar{w}_i}{p_i} - \sum_{i=1}^n \left\lfloor \frac{X}{p_i} \right\rfloor \cdot \bar{w}_i \\ &= X \cdot \frac{C_P}{P} - C(X). \end{aligned} \quad (5.10)$$

Подставляя (5.10) в (5.9), получим

$$\check{r}(X) = \left[r(X) + \frac{C(X)}{C_P} \right]. \quad (5.11)$$

Так как $r(X) \in \mathbb{Z}$, а $\forall a \in \mathbb{R}, n \in \mathbb{Z}: [a + n] = [a] + n$, то

$$\check{r}(X) = r(X) + \left\lfloor \frac{C(X)}{C_P} \right\rfloor. \quad (5.12)$$

Теорема доказана. □

Следствие 5.1.1. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$, число $X \in \mathbb{Z}_P$ и функция ядра Акушского, не содержащая критических ядер, тогда $\check{r}(X) = r(X)$.

Доказательство. Согласно Теореме 5.1.1, $\check{r}(X) = r(X) + \left\lfloor \frac{C(X)}{C_P} \right\rfloor$. Учитывая, что функция ядра Акушского не содержит критических ядер, $\forall X \in [0, P): 0 \leq C(X) < C_P$. Отсюда $\left\lfloor \frac{C(X)}{C_P} \right\rfloor = 0$, и значит $\check{r}(X) = r(X)$.

Следствие доказано. □

5.2 Представление ранга числа в виде алгебраического многочлена над \mathbb{Z}_P

Возникает проблема вычисления алгебраического многочлена, интерполирующего функцию ранга числа.

Теорема 5.2.1. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$, число $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и определена функция ранга числа $r(X) = \left\lfloor \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor$, тогда функцию $r(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Доказательство. Предположим противное, пусть функцию ранга числа $r(X)$ можно представить в виде многочлена степени $\phi(P) - 2$ (так как $X^{\phi(P)} \equiv X \pmod{P}$, $\phi(P)$ – функция Эйлера), получим

$$r(X) = \sum_{i=0}^{\phi(P)-2} a_i \cdot X^i. \quad (5.13)$$

Учитывая, что $r(0) = 0$, $a_0 = 0$ и формула (5.13) примет следующий вид

$$r(X) = \sum_{i=1}^{\phi(P)-2} a_i \cdot X^i. \quad (5.14)$$

Вычислим значение $|r(p_1)|_{p_1}$, используя формулу (5.14), получим

$$|r(p_1)|_{p_1} \equiv \left| \sum_{i=1}^{\phi(P)-2} a_i \cdot p_1^i \right|_{p_1} \equiv \sum_{i=1}^{\phi(P)-2} |a_i \cdot p_1^i|_{p_1} \equiv 0.$$

Вычислим значения $r(p_1 - 1)$ и $r(p_1)$, используя формулу $r(X + Y) = r(X) + r(Y) - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i}$ из работы [11], получим

$$\begin{aligned} r(p_1 - 1) &= (p_1 - 1) \cdot r(1), \\ r(p_1) &= r(p_1 - 1) + r(1) - |P_1^{-1}|_{p_1} \\ &= (p_1 - 1) \cdot r(1) + r(1) - |P_1^{-1}|_{p_1} \\ &= p_1 \cdot r(1) - |P_1^{-1}|_{p_1}. \end{aligned} \quad (5.15)$$

Используя формулу (5.15), вычислим значение $|r(p_1)|_{p_1}$, получим

$$|r(p_1)|_{p_1} \equiv -|P_1^{-1}|_{p_1}. \quad (5.16)$$

Так как $\gcd(P_1, p_1) = 1$, то $-|P_1^{-1}|_{p_1} \not\equiv 0 \pmod{p_1}$. Учитывая, что согласно формуле (5.15) $|r(p_1)|_{p_1} \equiv 0$, а согласно формуле (5.16) $|r(p_1)|_{p_1} \not\equiv 0$, имеет место противоречие, следовательно, функцию $r(X)$ нельзя выразить в виде алгебраического многочлена над \mathbb{Z}_P .

Теорема доказана. □

Следствие 5.2.1. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$, функция ядра Акушского, не содержащая критических ядер, и число $X \in \mathbb{Z}_P$, тогда функцию $\check{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Доказательство. Согласно Следствию 5.1.1 $\check{r} = r(X)$. Обратим внимание, что согласно Теореме 5.2.1 $r(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Следствие доказано. □

Докажем теорему о том, что функцию $\hat{r}(X)$ нельзя представить в виде алгебраического многочлена.

Теорема 5.2.2. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$ и число $X \in \mathbb{Z}_P$, тогда функцию $\hat{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Доказательство. Предположим противное, что функцию $\hat{r}(X)$ можно представить в виде алгебраического многочлена степени $\phi(P) - 2$, получим

$$\hat{r}(X) = \sum_{i=0}^{\phi(P)-2} a_i \cdot X^i. \quad (5.17)$$

Согласно формуле (5.2), ранг числа $\hat{r}(0) = 0$, следовательно, $a_0 = 0$, и формула (5.17) примет вид

$$\hat{r}(X) = \sum_{i=1}^{\phi(P)-2} a_i \cdot X^i. \quad (5.18)$$

Рассмотрим два случая.

Случай 1. Если $n = 2$. Предположим, что $\hat{r}(1) = 0$, тогда согласно Китайской теореме об остатках, заданной уравнением во второй форме (5.2), получим $\left\lfloor \frac{1}{p_2} \right\rfloor_{p_1} \cdot p_2 + \left\lfloor \frac{1}{p_1} \right\rfloor_{p_2} \cdot p_1 = 1$. Так как $\gcd(p_1, p_2) = 1$, то $\left\lfloor \frac{1}{p_2} \right\rfloor_{p_1} \geq 1$ и $\left\lfloor \frac{1}{p_1} \right\rfloor_{p_2} \geq 1$, следовательно, $\left\lfloor \frac{1}{p_2} \right\rfloor_{p_1} \cdot p_2 + \left\lfloor \frac{1}{p_1} \right\rfloor_{p_2} \cdot p_1 \geq p_1 + p_2$. Учитывая, что $p_1 \geq 2$ и $p_2 \geq 3$, $\left\lfloor \frac{1}{p_2} \right\rfloor_{p_1} \cdot p_2 + \left\lfloor \frac{1}{p_1} \right\rfloor_{p_2} \cdot p_1 \geq 5$ и $\left\lfloor \frac{1}{p_2} \right\rfloor_{p_1} \cdot p_2 + \left\lfloor \frac{1}{p_1} \right\rfloor_{p_2} \cdot p_1 \neq 1$. Таким образом, имеет место противоречие и $\hat{r}(1) \neq 0$. Учитывая, что $\forall X \in [0, P): \hat{r}(X) \in \{0, 1\}$ и $\hat{r}(1) \neq 0$, $\hat{r}(1) = 1$. Следовательно, если функцию ядра $\hat{r}(X)$ можно представить в виде

алгебраического многочлена над \mathbb{Z}_P , то его коэффициенты a_i удовлетворяют сравнению

$$\hat{r}(1) = \sum_{i=1}^{\phi(P)-2} a_i \equiv 1 \pmod{P}. \quad (5.19)$$

Вычислим значение $\hat{r}\left(\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1\right)$. Учитывая, что $\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1 \xrightarrow{RNS} (0, 1)$, и, используя Свойство 5.1.1, получим

$$\hat{r}\left(\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1\right) = -\frac{\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1}{P} + \frac{\left|\frac{1}{p_1}\right|_{p_2}}{p_2} = -\frac{\left|\frac{1}{p_1}\right|_{p_2}}{p_2} + \frac{\left|\frac{1}{p_1}\right|_{p_2}}{p_2} = 0. \quad (5.20)$$

Обратим внимание, что $\left(\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1\right)^2 \equiv \left|\frac{1}{p_1}\right|_{p_2} \cdot p_1 \pmod{P}$, следовательно, $\forall i \in \mathbb{N}$: $\left(\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1\right)^i \equiv \left|\frac{1}{p_1}\right|_{p_2} \cdot p_1 \pmod{P}$. Из (5.20) следует, что если функцию ядра $\hat{r}(X)$ можно представить в виде алгебраического многочлена над \mathbb{Z}_P , то его коэффициенты a_i удовлетворяют сравнению

$$\hat{r}\left(\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1\right) = \left|\frac{1}{p_1}\right|_{p_2} \cdot p_1 \cdot \sum_{i=1}^{\phi(P)-2} a_i \equiv 0 \pmod{P}. \quad (5.21)$$

Умножим (5.19) на $\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1$ и вычтем из результата (5.21), получим

$$\left|\frac{1}{p_1}\right|_{p_2} \cdot p_1 \equiv 0 \pmod{P}. \quad (5.22)$$

Таким образом, имеет место противоречие, следовательно, при $n = 2$ функцию ранга числа $\hat{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Случай 2. Если $n \geq 3$. Используя формулу (5.18), вычислим значение $|\hat{r}(p_n)|_{p_n}$, получим

$$|\hat{r}(p_n)|_{p_n} = \left| \sum_{i=1}^{\phi(P)-2} a_i \cdot p_n^i \right|_{p_n} \equiv \sum_{i=1}^{\phi(P)-2} |a_i \cdot p_n^i|_{p_n} \equiv 0. \quad (5.23)$$

Учитывая, что $p_n \xrightarrow{RNS} \left(|p_n|_{p_1}, |p_n|_{p_2}, \dots, |p_n|_{p_{n-1}}, 0\right)$, вычислим значение $\hat{r}(p_n)$, используя Свойство 5.1.1, получим

$$\hat{r}(p_n) = -\frac{p_n}{P} + \sum_{i=1}^{n-1} \frac{\left|P_i^{-1}\right|_{p_i} \cdot |p_n|_{p_i}}{p_i}. \quad (5.24)$$

Пусть $\forall i = \overline{1, n-1}$: $\hat{P}_i = \frac{P}{p_n \cdot p_i}$, тогда $\left| |P_i^{-1}|_{p_i} \cdot |p_n|_{p_i} \right|_{p_i} = \left| \hat{P}_i^{-1} \right|_{p_i}$ и $\frac{p_n}{P} = \frac{1}{P_n}$.
С учетом последних соотношений получим

$$\hat{r}(p_n) = -\frac{1}{P_n} + \sum_{i=1}^{n-1} \frac{\left| \hat{P}_i^{-1} \right|_{p_i}}{p_i}. \quad (5.25)$$

Оценим правую часть формулы (5.25). Учитывая, что $\forall i = \overline{1, n-1}$: $\gcd(\hat{P}_i, p_i) = 1$, $1 \leq \left| \hat{P}_i^{-1} \right|_{p_i} \leq p_i - 1$, тогда значение $\hat{r}(p_n)$ удовлетворяет неравенству

$$-\frac{1}{P_n} + \sum_{i=1}^{n-1} \frac{1}{p_i} \leq -\frac{1}{P_n} + \sum_{i=1}^{n-1} \frac{\left| \hat{P}_i^{-1} \right|_{p_i}}{p_i} \leq -\frac{1}{P_n} + \sum_{i=1}^{n-1} \frac{p_i - 1}{p_i}. \quad (5.26)$$

Учитывая, что

$$\sum_{i=1}^{n-1} \frac{1}{p_i} = \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i, \quad (5.27)$$

$$\sum_{i=1}^{n-1} \frac{p_i - 1}{p_i} = n - 1 - \sum_{i=1}^{n-1} \frac{1}{p_i} = n - 1 - \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i, \quad (5.28)$$

неравенство (5.26) примет вид

$$-\frac{1}{P_n} + \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i \leq -\frac{1}{P_n} + \sum_{i=1}^{n-1} \frac{\left| \hat{P}_i^{-1} \right|_{p_i}}{p_i} \leq -\frac{1}{P_n} + n - 1 - \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i. \quad (5.29)$$

Так как $n \geq 3$, то $\hat{P}_1 \geq 3$, $-\frac{1}{P_n} + \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i > 0$, $-\frac{1}{P_n} + n - 1 - \frac{1}{P_n} \sum_{i=1}^{n-1} \hat{P}_i < n - 1$, следовательно, $\hat{r}(p_n) \in (0, n - 1)$. Учитывая, что $\hat{r}(p_n) \in \mathbb{Z}$, $\hat{r}(p_n) \in \{1, 2, \dots, n - 2\}$. Учитывая, что $\forall i \neq j$: $\gcd(p_i, p_j) = 1$, может существовать только две пары чисел (оснований RNS), удовлетворяющих условию $p_{i+1} - p_i = 1$, для всех остальных пар $p_{i+1} - p_i \geq 2$, значит $p_n \geq p_1 + 2 + (n - 3) \cdot 2 = p_1 + 2n - 4$. Учитывая, что $p_1 \geq 2$, получим неравенство $p_n \geq 2n - 2$. Так как согласно условию теоремы $n \geq 3$, $2n - 2 > n - 2$ и $p_n > n - 2$, следовательно, $0 < \hat{r}(p_n) < p_n$, значит $|\hat{r}(p_n)|_{p_n} \not\equiv 0$. Таким образом, имеет место противоречие: с одной стороны, $|\hat{r}(p_n)|_{p_n} \equiv 0$, с другой стороны, $|\hat{r}(p_n)|_{p_n} \not\equiv 0$. Следовательно, функцию ранга $\hat{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Объединяя результаты, полученные в первом и втором случае, делаем вывод, что функцию ранга $\hat{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Теорема доказана. \square

Свойство 5.2.1. $\check{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i}$.

Доказательство. Согласно Теореме 5.1.1, значение $\check{r}(1)$ равно $\check{r}(1) = r(1) + \left\lfloor \frac{C(1)}{C_P} \right\rfloor$. Учитывая, что $\forall i: p_i \geq 2$, $\left\lfloor \frac{1}{p_i} \right\rfloor = 0$, следовательно, $C(1) = \sum_{i=1}^n \bar{w}_i \left\lfloor \frac{1}{p_i} \right\rfloor = 0$, значит $\check{r}(1) = r(1)$. Вычислим значение $r(1)$, используя формулу (5.1), получим

$$\check{r}(1) = r(1) = \left\lfloor \frac{\sum_{i=1}^n P_i \cdot |P_i^{-1}|_{p_i}}{P} \right\rfloor. \quad (5.30)$$

Учитывая, что $\left\lfloor \frac{X}{P} \right\rfloor = \frac{X}{P} - \frac{|X|_P}{P}$, и согласно Китайской теореме об остатках $\left\lfloor \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \right\rfloor_P = 1$, формула (5.30) примет вид

$$\check{r}(1) = -\frac{1}{P} + \frac{\sum_{i=1}^n P_i \cdot |P_i^{-1}|_{p_i}}{P} = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i}. \quad (5.31)$$

Свойство доказано. \square

Свойство 5.2.2. Если модули RNS удовлетворяют условию $p_1 < p_2 < \dots < p_n$, то

$$\check{r}(p_1) = p_1 \cdot r(1) - |P_1^{-1}|_{p_1} + \left\lfloor \frac{\bar{w}_1}{C_P} \right\rfloor. \quad (5.32)$$

Доказательство. Используя Теорему 5.1.1, получим

$$\check{r}(p_1) = r(p_1) + \left\lfloor \frac{C(p_1)}{C_P} \right\rfloor. \quad (5.33)$$

Учитывая, что $r(p_1) = p_1 \cdot r(1) - |P_1^{-1}|_{p_1}$ и $C(p_1) = \bar{w}_1$, получим

$$\check{r}(p_1) = p_1 \cdot r(1) - |P_1^{-1}|_{p_1} + \left\lfloor \frac{\bar{w}_1}{C_P} \right\rfloor.$$

Свойство доказано. \square

Свойство 5.2.3. Если модули RNS удовлетворяют условию $p_1 < p_2 < \dots < p_n$ и $1 \leq t \leq n$, то

$$\check{r} \left(|P_t^{-1}|_{p_t} \cdot P_t \right) = 0. \quad (5.34)$$

Доказательство. Учитывая, что $B_t = |P_t^{-1}|_{p_t} \cdot P_t \xrightarrow{RNS} (b_1, b_2, \dots, b_n)$, где $\forall i \neq t: b_i = 0$, а $b_t = 1$, вычислим $C(B_t)$ по определению

$$C(B_t) = \sum_{i=1}^n c_i \cdot b_i - \check{r}(B_t) \cdot C_P = c_t - \check{r}(B_t) \cdot C_P = C(B_t) - \check{r}(B_t) \cdot C_P. \quad (5.35)$$

Из (5.35) следует, что $\check{r}(B_t) \cdot C_P = 0$. Учитывая, что $C_P \neq 0$, получим, что $\check{r}(B_t) = 0$.

Свойство доказано. \square

Теорема 5.2.3. Пусть заданы попарно взаимно простые модули RNS $p_1 < p_2 < \dots < p_n$ и число $X \in \mathbb{Z}_P$, тогда функцию $\check{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Доказательство. Предположим, что над \mathbb{Z}_P существует алгебраический многочлен

$$\check{r}(X) = \sum_{i=0}^{\phi(P)-2} a_i \cdot X^i. \quad (5.36)$$

Вычислим $\check{r}(0)$ с использованием формулы (5.36), получим, что $\check{r}(0) = a_0 = 0$, следовательно, (5.36) преобразуется к виду

$$\check{r}(X) = \sum_{i=1}^{\phi(P)-2} a_i \cdot X^i. \quad (5.37)$$

Используя формулу (5.37) вычислим значения $\check{r}(1)$ и $\check{r}(B_n)$, получим

$$\check{r}(1) = \sum_{i=1}^{\phi(P)-2} a_i, \quad (5.38)$$

$$\check{r}(B_n) = \sum_{i=1}^{\phi(P)-2} a_i \cdot B_n^i. \quad (5.39)$$

Учитывая, что $\forall i \in \mathbb{N}: B_n^i \equiv B_n \pmod{P}$, формула (5.39) преобразуется к виду

$$\check{r}(B_n) \equiv B_n \cdot \sum_{i=1}^{\phi(P)-2} a_i \pmod{P}. \quad (5.40)$$

С другой стороны, согласно Свойству 5.2.3

$$\check{r}(B_n) = 0.$$

Составим систему сравнений над \mathbb{Z}_P

$$\begin{cases} \sum_{i=1}^{\phi(P)-2} a_i & \equiv \check{r}(1) \pmod{P}, \\ B_n \cdot \sum_{i=1}^{\phi(P)-2} a_i & \equiv 0 \pmod{P}. \end{cases} \quad (5.41)$$

Система сравнений (5.41) равносильна сравнению

$$B_n \cdot \check{r}(1) \equiv 0 \pmod{P}. \quad (5.42)$$

Так как $B_n = P_n \cdot |P_n^{-1}|_{p_n}$, то $\gcd(B_n, P) = P_n$, тогда (5.42) равносильно сравнению

$$|P_n^{-1}|_{p_n} \cdot \check{r}(1) \equiv 0 \pmod{p_n}. \quad (5.43)$$

Так как $\gcd(|P_n^{-1}|_{p_n}, p_n) = 1$, то (5.43) равносильно сравнению

$$\check{r}(1) \equiv 0 \pmod{p_n}. \quad (5.44)$$

Из (5.44) следует, что чтобы прийти к противоречию необходимо и достаточно показать, что сравнение не имеет решений.

Для начала покажем, что $\check{r}(1)$ не равно нулю. Согласно Свойству 5.2.1, значение $\check{r}(1)$ вычисляется по формуле $\check{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i}$ и $\check{r}(1) \in \mathbb{Z}$.

Так как $n \geq 2$, то $P_1 \geq 2$, $P_2 \geq 3$, следовательно, $\sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \geq |P_1^{-1}|_{p_1} \cdot P_1 + |P_2^{-1}|_{p_2} \cdot P_2 \geq 2 \cdot |P_1^{-1}|_{p_1} + 3 \cdot |P_2^{-1}|_{p_2}$. Учитывая, что $\gcd(P_1, p_1) = 1$ и $\gcd(P_2, p_2) = 1$, $|P_1^{-1}|_{p_1} \geq 1$ и $|P_2^{-1}|_{p_2} \geq 1$. Значит, $\sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \geq 5$. Подставим полученный результат в $\check{r}(1)$

$$\check{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i}}{p_i} = -\frac{1}{P} + \frac{1}{P} \cdot \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \geq -\frac{1}{P} + \frac{5}{P} = \frac{4}{P} \quad (5.45)$$

Учитывая, что $\check{r}(1) \in \mathbb{Z}$ и $\check{r}(1) \geq \frac{4}{P} > 0$, то $\check{r}(1) \geq 1$. Таким образом, $\check{r}(1) \neq 1$.

Покажем, что $\check{r}(1) < p_n$. Для этого найдем верхнюю границу значения $\check{r}(1)$. Учитывая, что $\forall i \in \overline{1, n}$: $|P_i^{-1}|_{p_i} \leq p_i - 1$, получим

$$\check{r}(1) = -\frac{1}{P} + \sum_{i=1}^n \frac{|P_i^{-1}|}{p_i} \leq -\frac{1}{P} + \sum_{i=1}^n \frac{p_i - 1}{p_i} = n - \frac{1}{P} - \frac{1}{P} \sum_{i=1}^n P_i = n - \frac{1}{P} - \frac{SQ}{P} \leq n.$$

Учитывая, что $\check{r}(1) < n$ и $\check{r}(1) \in \mathbb{Z}$, $\check{r}(1) \leq n - 1$. Ранее было показано, что $p_n \geq 2n - 2$, следовательно, $p_n > n - 1$ и $\check{r}(1) < p_n$.

Таким образом, $0 < \check{r}(1) < p_n$, следовательно, сравнение (5.44) решений не имеет. Значит функцию ранга числа ядра Акушского $\check{r}(X)$ нельзя представить в виде алгебраического многочлена над \mathbb{Z}_P .

Теорема доказана. □

5.3 Разработка методов обнаружения и исправления ошибок арифметических операций обработки закодированных данных с использованием свойств ранга числа

5.3.1 Разработка методов обнаружения и исправления ошибок арифметических операций с использованием свойств ранга числа $r(X)$

Теорема 5.3.1. [11, 129] Если $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$, заданные в RNS с основаниями p_1, p_2, \dots, p_n , удовлетворяют следующим условиям $0 \leq X < P$, $0 \leq Y < P$ и $X + Y < P$, то для них выполняется следующее соотношение

$$r(X + Y) = r(X) + r(Y) - \sum_{x_i + y_i \geq 0} |P_i^{-1}|_{p_i}. \quad (5.46)$$

Доказательство. По определению ранг числа $r(X + Y)$ вычисляется согласно следующей формуле

$$r(X + Y) = \left\lfloor \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot |x_i + y_i|_{p_i}}{p_i} \right\rfloor. \quad (5.47)$$

Учитывая, что $\forall i \in \overline{1, n}$ выполняется следующее равенство

$$|x_i + y_i|_{p_i} = \begin{cases} x_i + y_i - p_i, & \text{если } x_i + y_i \geq p_i, \\ x_i + y_i, & \text{иначе,} \end{cases}$$

тогда (5.47) можно записать в следующем виде

$$r(X + Y) = \left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} + \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} \right]. \quad (5.48)$$

Поскольку число $\forall a \in \mathbb{R}$ можно представить в виде суммы целой и дробной частей, т.е. $a = [a] + \{a\}$, где $\{a\}$ – дробная часть числа a , то $\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i}$ и $\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i}$ можно представить в виде

$$\begin{aligned} \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} &= \left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right] + \left\{ \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\} \\ &= \left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right] + \frac{1}{P} \cdot \left| \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot x_i \cdot P_i \right|_P. \end{aligned} \quad (5.49)$$

$$\begin{aligned} \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} &= \left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} \right] + \left\{ \sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} \right\} \\ &= \left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} \right] + \frac{1}{P} \cdot \left| \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot y_i \cdot P_i \right|_P. \end{aligned} \quad (5.50)$$

Обратим внимание на то, что по определению ранга числа $\left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right] = r(X)$, а $\left[\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} \right] = r(Y)$. Согласно Китайской теореме об остатках $\left| \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot x_i \cdot P_i \right|_P = X$, а $\left| \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot y_i \cdot P_i \right|_P = Y$, следовательно, формулы (5.49) и (5.50) преобразуются к виду

$$\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} = r(X) + \frac{X}{P}, \quad (5.51)$$

$$\sum_{i=1}^n \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} = r(Y) + \frac{Y}{P}. \quad (5.52)$$

Подставляя (5.51) и (5.52) в (5.48), получим

$$r(X + Y) = \left[r(X) + \frac{X}{P} + r(Y) + \frac{Y}{P} - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} \right]. \quad (5.53)$$

Учитывая, что $\forall a \in \mathbb{R}, b \in \mathbb{Z}$ выполняется равенство $[a + b] = [a] + b$, и $r(X), r(Y), \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} \in \mathbb{Z}$, то (5.53) преобразуется к виду

$$r(X + Y) = r(X) + r(Y) + \left[\frac{X}{P} + \frac{Y}{P} \right] - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i}. \quad (5.54)$$

По условию теоремы $X + Y$ удовлетворяет неравенству $0 \leq X + Y < P$, следовательно, слагаемое $\left\lfloor \frac{X}{P} + \frac{Y}{P} \right\rfloor = 0$, и (5.54) примет вид

$$r(X + Y) = r(X) + r(Y) - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i}. \quad (5.55)$$

Теорема доказана. \square

Теорема 5.3.2. Если $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$, заданные в RNS с основаниями p_1, p_2, \dots, p_n , удовлетворяют следующим условиям $0 \leq X < P$, $0 \leq Y < P$ и $0 \leq X - Y < P$, то для них выполняется следующее соотношение

$$r(X - Y) = r(X) - r(Y) + \sum_{x_i < y_i} |P_i^{-1}|_{p_i}. \quad (5.56)$$

Доказательство. Вычислим $r(X - Y)$, получим

$$r(X - Y) = \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} |x_i - y_i|_{p_i}}{P} \right\rfloor. \quad (5.57)$$

Так как $|x_i - y_i|_{p_i}$ можно вычислить по формуле

$$|x_i - y_i|_{p_i} = \begin{cases} x_i - y_i + p_i, & \text{если } x_i < y_i, \\ x_i - y_i, & \text{иначе,} \end{cases}$$

то (5.57) преобразуется к виду

$$r(X - Y) = \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} (x_i - y_i) + P \cdot \sum_{x_i < y_i} |P_i^{-1}|_{p_i}}{P} \right\rfloor. \quad (5.58)$$

Учитывая, что для любых $m \in \mathbb{Z}$ и $a \in \mathbb{R}$ выполняется равенство $\lfloor m + a \rfloor = m + \lfloor a \rfloor$, и $\sum_{x_i < y_i} |P_i^{-1}|_{p_i} \in \mathbb{Z}$, формула (5.58) примет вид

$$r(X - Y) = \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} - \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right\rfloor + \sum_{x_i < y_i} |P_i^{-1}|_{p_i}. \quad (5.59)$$

Так как $P \in \mathbb{Z}$, то

$$\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i = \left\lfloor \sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i \right\rfloor_P + P \cdot \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} \right\rfloor \quad (5.60)$$

$$\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i = \left\lfloor \sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i \right\rfloor_P + P \cdot \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right\rfloor. \quad (5.61)$$

Подставляя (5.60) и (5.61) в (5.59), получим

$$r(X - Y) = \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} \right] - \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right] + \left[\frac{|\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i|_P}{P} - \frac{|\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i|_P}{P} \right] + \sum_{x_i < y_i} |P_i^{-1}|_{p_i}. \quad (5.62)$$

Согласно Китайской теореме об остатках $|\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i|_P = X$, $|\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i|_P = Y$, следовательно, (5.62) примет вид

$$r(X - Y) = \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} \right] - \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right] + \left[\frac{X}{P} - \frac{Y}{P} \right] + \sum_{x_i < y_i} |P_i^{-1}|_{p_i}. \quad (5.63)$$

Так как по условию теоремы $0 \leq X - Y < P$, то слагаемое $[\frac{X}{P} - \frac{Y}{P}]$ в (5.63) равно нулю. Учитывая, что

$$r(X) = \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} \right], \quad (5.64)$$

$$r(Y) = \left[\frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right], \quad (5.65)$$

получим

$$r(X - Y) = r(X) - r(Y) + \sum_{x_i < y_i} |P_i^{-1}|_{p_i}. \quad (5.66)$$

Теорема доказана. \square

Теорема 5.3.3. Пусть заданы модули p_1, p_2, \dots, p_n и два целых числа $X, Y \in \mathbb{Z}_P$ в соответствующей RNS: $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$ и $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$. Если существует такое $j \in \overline{1, n}$, для которого выполняется равенство $X = p_j \cdot Y$, то

$$r(X) = p_j \cdot r(Y) - \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left[\frac{p_j \cdot y_i}{p_i} \right]. \quad (5.67)$$

Доказательство. Вычислим значение $r(X)$, используя формулу (5.64), получим

$$r(X) = \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot x_i}{P} \right\rfloor. \quad (5.68)$$

Так как для любого i выполняется равенство $x_i = |p_j \cdot y_i|_{p_i} = p_j \cdot y_i - p_i \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor$, то (5.68) преобразуется к виду

$$\begin{aligned} r(X) &= \left\lfloor \frac{\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot \left(p_j \cdot y_i - p_i \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor \right)}{P} \right\rfloor \\ &= \left\lfloor \frac{p_j \cdot \sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i - P \cdot \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor}{P} \right\rfloor \\ &= \left\lfloor \frac{p_j \cdot \sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i}{P} \right\rfloor - \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor. \end{aligned} \quad (5.69)$$

Согласно Китайской теореме об остатках $\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i$ можно представить в виде $\sum_{i=1}^n P_i |P_i^{-1}|_{p_i} \cdot y_i = P \cdot r(Y) + Y$, следовательно, формула (5.69) преобразуется к виду

$$\begin{aligned} r(X) &= \left\lfloor \frac{p_j \cdot P \cdot r(Y) + p_j \cdot Y}{P} \right\rfloor - \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor \\ &= p_j \cdot r(Y) + \left\lfloor \frac{p_j \cdot Y}{P} \right\rfloor - \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor. \end{aligned} \quad (5.70)$$

Из условия теоремы следует, что $X = p_j \cdot Y$ и $X \in \mathbb{Z}_P$, следовательно, X удовлетворяет неравенству $0 \leq X < P$, значит слагаемое $\left\lfloor \frac{p_j \cdot Y}{P} \right\rfloor$ в формуле (5.70) равно нулю, и формула (5.70) преобразуется к виду

$$r(X) = p_j \cdot r(Y) - \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{p_j \cdot y_i}{p_i} \right\rfloor.$$

Теорема доказана. □

Используя Теоремы 5.3.2, 5.3.3 и формулу (5.46), предложим алгоритм вычисления ранга числа (Алгоритм 4). Рассмотрим пример вычисления ранга числа с помощью формул (5.46), (5.56) и (5.67).

Пример 5.3.1. Пусть заданы модули RNS $p_1 = 2, p_2 = 3, p_3 = 5$. Вычислим ранг числа $X \xrightarrow{RNS} (1, 2, 3)$.

Алгоритм 4: Алгоритм вычисления ранга числа $r(X)$

Input: $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n), p_1, p_2, \dots, p_{n-1}, p_n, w_{i,j} = \left| p_i^{-1} \right|_{p_j}$
 $\forall i, j = \overline{1, n} \ i \neq j, B_i = \left| P_i^{-1} \right|_{p_i} \ \forall i = \overline{1, n}, r_i = r(i) \ \forall i = \overline{1, p_n}$, где
 $P_i = P/p_i \ \forall i = \overline{1, n}$

Output: $r(X)$

1 $x_1^{(1)} = 0;$

2 **for** $j = 2, j \leq n, j++$ **do**

3 $\left[x_j^{(1)} = \left| x_j - x_1 \right|_{p_j}; y_j^{(1)} = \left| w_{1,j} \cdot x_j^{(1)} \right|_{p_j} \right];$ Parallel processing

4 **for** $i = 2, i < n, i++$ **do**

5 $x_i^{(i)} = 0;$

6 **for** $j = i + 1, j \leq n, j++$ **do**

7 $\left[x_j^{(i)} = \left| y_j^{(i-1)} - y_i^{(i-1)} \right|_{p_j}; y_j^{(i)} = \left| w_{i,j} \cdot x_j^{(i)} \right|_{p_j} \right];$ Parallel processing

8 $r = p_{n-1} \cdot r \left(y_n^{(n-1)} \right) = p_{n-1} \cdot r_{y_n^{(n-1)}};$

9 **for** $j = 1, j < n, j++$ **do**

10 $\left[y_j^{(n-1)} = \left| y_n^{(n-1)} \right|_{p_j} \right];$ Parallel processing

11 $\left[x_j^{(n-1)} = \left| p_{n-1} \cdot y_j^{(n-1)} \right|_{p_j}; r = r - B_j \cdot \left[\frac{p_{n-1} \cdot y_j^{(n-1)}}{p_j} \right]; \right]$ Parallel processing

12 **for** $i = n - 2, i \geq 1, i--$ **do**

13 $r_{mult} = 0;$

14 $r_{add} = 0;$

15 **for** $j = 1, j \leq n, j++$ **do**

16 $\left[y_j^{(i)} = \left| x_j^{(i)} + y_{i+1}^{(i)} \right|_{p_j} \right];$ Parallel processing

17 **if** $x_j^{(i)} + y_{i+1}^{(i)} \geq p_j$ **then**

18 $\left[r_{add} = r_{add} + B_j; \right]$

19 $\left[x_j^{(i)} = \left| p_i \cdot y_j^{(i)} \right|_{p_j}; r_{mult} = r_{mult} + B_j \cdot \left[\frac{p_i \cdot y_j^{(i)}}{p_j} \right]; \right]$ Parallel processing

20 $r = r + r_{y_{i+1}^{(i)}} - r_{add}; r = p_i \cdot r - r_{mult};$

21 **for** $j = 1, j \leq n, j++$ **do**

22 **if** $x_j^{(1)} + x_1 \geq p_j$ **then**

23 $\left[r = r - B_j; \right]$

24 $r = r + r_{x_1}$

Result: r

1. Диапазон RNS равен $P = \prod_{i=1}^n p_i = 30$.
2. Вычислим значения констант P_i и $|P_i^{-1}|_{p_i}$, получим:
 $P_1 = P/p_1 = 15$, $|P_1^{-1}|_{p_1} = |15^{-1}|_2 = 1$, $P_2 = P/p_2 = 10$,
 $|P_2^{-1}|_{p_2} = |10^{-1}|_3 = 1$, $P_3 = P/p_3 = 6$ и $|P_3^{-1}|_{p_3} = |6^{-1}|_5 = 1$.
3. Вычислим значения констант $|p_i^{-1}|_{p_j}$, получим:
 $|p_1^{-1}|_{p_2} = |2^{-1}|_3 = 2$, $|p_1^{-1}|_{p_3} = |2^{-1}|_5 = 3$, $|p_2^{-1}|_{p_3} = |3^{-1}|_5 = 2$,
4. Вычислим ранги чисел $0, \dots, (p_n - 1)$, получим:
 $r(0) = 0$,
 $r(1) = \left\lfloor \frac{\sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot P_i \cdot x_i}{P} \right\rfloor = \left\lfloor \frac{1 \cdot 15 \cdot 1 + 1 \cdot 10 \cdot 1 + 1 \cdot 6 \cdot 1}{30} \right\rfloor = 1$,
 $r(2) = r(1) + r(1) - |P_1^{-1}|_{p_1} = 1 + 1 - 1 = 1$,
 $r(3) = r(2) + r(1) - |P_2^{-1}|_{p_2} = 1 + 1 - 1 = 1$,
 $r(4) = r(3) + r(1) - |P_1^{-1}|_{p_1} = 1 + 1 - 1 = 1$.
5. Для удобства результаты вычислений занесем в таблицы 28, 29.

Таблица 28 — Вычисление $X = \left\lfloor \frac{X}{p_1 \cdot p_2} \right\rfloor$

| | | $p_1 = 2$ | $p_2 = 3$ | $p_3 = 5$ |
|--|---------------------------|-----------|-----------|-----------|
| X | | $x_1 = 1$ | $x_2 = 2$ | $x_3 = 3$ |
| $X^{(1)} = X - x_1$ | $-x_1$ | 0 | 1 | 2 |
| $Y^{(1)} = \left\lfloor \frac{X^{(1)}}{p_1} \right\rfloor$ | $\times p_1^{-1} _{p_i}$ | — | 2 | 1 |
| $X^{(2)} = Y^{(1)} - y_2^{(1)}$ | $-y_2^{(1)}$ | — | 0 | 4 |
| $Y^{(2)} = \left\lfloor \frac{X^{(2)}}{p_2} \right\rfloor$ | $\times p_2^{-1} _{p_i}$ | — | — | 3 |

Из таблицы 28 следует, что $X = \left\lfloor \frac{X}{p_1 \cdot p_2} \right\rfloor = Y^{(2)} = 3$. Для удобства промежуточные результаты при реализации обратного хода занесем в таблицу 29. Из вычислений, представленных в таблице 29, следует,

Таблица 29 — Вычисление ранга $r(X)$

| | | $p_1 = 2$ | $p_2 = 3$ | $p_3 = 5$ | $r(X)$ |
|---------------------------------|--------------|-----------------|-----------------|-----------------|---|
| $Y^{(2)}$ | | $y_1^{(2)} = 1$ | $y_2^{(2)} = 0$ | $y_3^{(2)} = 3$ | $r(Y^{(2)}) = r(3) = 1$ |
| $X^{(2)} = p_2 \cdot Y^{(2)}$ | $\times p_2$ | 1 | 0 | 4 | $r(X^{(2)}) = p_2 \cdot r(Y^{(2)}) - \sum_{i=1}^n P_i^{-1} _{p_i} \left\lfloor \frac{p_2 \cdot y_i^{(2)}}{p_i} \right\rfloor =$ $3 \cdot 1 - 1 \cdot 1 - 1 \cdot 0 - 1 \cdot 1 = 1$ |
| $Y^{(1)} = X^{(2)} + y_2^{(1)}$ | $+y_2^{(1)}$ | 1 | 2 | 1 | $r(Y^{(1)}) = r(X^{(2)}) + r(y_2^{(1)}) - \sum_{x_i^{(2)} + y_2^{(1)} \geq p_i} P_i^{-1} _{p_i} =$ $1 + 1 - 1 = 1$ |
| $X^{(1)} = p_1 \cdot Y^{(1)}$ | $\times p_1$ | 0 | 1 | 2 | $r(X^{(1)}) = p_1 \cdot r(Y^{(1)}) - \sum_{i=1}^n P_i^{-1} _{p_i} \left\lfloor \frac{p_1 \cdot y_i^{(1)}}{p_i} \right\rfloor =$ $2 \cdot 1 - 1 \cdot 1 - 1 \cdot 1 - 1 \cdot 0 = 0$ |
| $X = X^{(1)} + x_1$ | $+x_1$ | 1 | 2 | 3 | $r(X) = r(X^{(1)}) + r(x_1) - \sum_{x_i^{(1)} + x_1 \geq p_i} P_i^{-1} _{p_i} =$ $0 + 1 = 1$ |

что $r(X) = 1$.

5.3.2 Разработка методов обнаружения и исправления ошибок арифметических операций с использованием свойств нормализованного ранга числа $\hat{r}(X)$

Рассмотрим лемму, позволяющую связать две функции ранга числа $r(X)$ и $\hat{r}(X)$ между собой.

Лемма 5.3.1. Пусть заданы основания RNS p_1, p_2, \dots, p_n и целое число X , удовлетворяющее условию $0 \leq X < P$, тогда справедливо следующее выражение

$$\hat{r}(X) = r(X) - \sum_{i=1}^n \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor. \quad (5.71)$$

Доказательство. Заметим, что для любых $a \in \mathbb{Z}$ и $p \in \mathbb{N}$ выполняется равенство $|a|_p = a - p \cdot \left\lfloor \frac{a}{p} \right\rfloor$ подставляя его в формулу (5.2) вычислим значение $\hat{r}(X)$

$$\begin{aligned} \hat{r}(X) &= \left\lfloor \sum_{i=1}^n \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right\rfloor \\ &= \left\lfloor \sum_{i=1}^n \frac{1}{p_i} \cdot \left(\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - p_i \cdot \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor \right) \right\rfloor \\ &= \left\lfloor \sum_{i=1}^n \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \sum_{i=1}^n \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor \right\rfloor. \end{aligned} \quad (5.72)$$

Учитывая, что $\forall a \in \mathbb{R}$ и $n \in \mathbb{Z}$ выполняется равенство $\lfloor a + n \rfloor = \lfloor a \rfloor + n$, (5.72) примет вид

$$\hat{r}(X) = \left\lfloor \sum_{i=1}^n \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right\rfloor - \sum_{i=1}^n \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor. \quad (5.73)$$

Подставив (5.1) в (5.73), получим

$$\hat{r}(X) = r(X) - \sum_{i=1}^n \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor. \quad (5.74)$$

Лемма доказана. □

Лемма 5.3.2. Пусть заданы основания RNS p_1, p_2, \dots, p_n и целые числа $X, Y \in \mathbb{Z}_P$: $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$, $Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$, удовлетворяющее условию $0 \leq X + Y < P$, тогда справедливо следующее выражение

$$\hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} + \left| |P_i^{-1}|_{p_i} \cdot y_i \right|_{p_i}}{p_i} \right]. \quad (5.75)$$

Доказательство. Из Леммы 5.3.1 следует, что

$$\hat{r}(X + Y) = r(X + Y) - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot |x_i + y_i|_{p_i} \right|}{p_i} \right].$$

Используя формулу $r(X + Y) = r(X) + r(Y) - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i}$, получим

$$\hat{r}(X + Y) = r(X) + r(Y) - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot |x_i + y_i|_{p_i} \right|}{p_i} \right]. \quad (5.76)$$

Выражая $r(X)$ и $r(Y)$ через $\hat{r}(X)$ и $\hat{r}(Y)$ соответственно, с использованием Леммы 5.3.1, получим

$$\begin{aligned} \hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|}{p_i} \right] + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot y_i \right|}{p_i} \right] \\ - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot |x_i + y_i|_{p_i} \right|}{p_i} \right]. \end{aligned} \quad (5.77)$$

Так как $|x_i + y_i|_{p_i} = x_i + y_i - p_i \left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor$ и $\forall a \in \mathbb{Z}, b \in \mathbb{R}: \lfloor a + b \rfloor = a + \lfloor b \rfloor$, то (5.75) примет следующий вид

$$\begin{aligned} \hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|}{p_i} \right] + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot y_i \right|}{p_i} \right] \\ - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot (x_i + y_i) \right|}{p_i} \right] + \sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor. \end{aligned} \quad (5.78)$$

Учитывая, что $\sum_{i=1}^n |P_i^{-1}|_{p_i} \cdot \left\lfloor \frac{x_i + y_i}{p_i} \right\rfloor - \sum_{x_i + y_i \geq p_i} |P_i^{-1}|_{p_i} = 0$, (5.78) упрощается и примет следующий вид

$$\begin{aligned} \hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot x_i \right|}{p_i} \right] + \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot y_i \right|}{p_i} \right] \\ - \sum_{i=1}^n \left[\frac{\left| |P_i^{-1}|_{p_i} \cdot (x_i + y_i) \right|}{p_i} \right]. \end{aligned} \quad (5.79)$$

Так как $\forall a \in \mathbb{Z}, b \in \mathbb{R}: [a + b] = a + [b]$, то

$$\hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) - \sum_{i=1}^n \left[\frac{|P_i^{-1}|_{p_i} \cdot (x_i + y_i)}{p_i} - \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor - \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot y_i}{p_i} \right\rfloor \right]. \quad (5.80)$$

Учитывая, что $\frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} - \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor = \frac{||P_i^{-1}|_{p_i} \cdot x_i|_{p_i}}{p_i}$,

$$\hat{r}(X + Y) = \hat{r}(X) + \hat{r}(Y) - \sum_{i=1}^n \left[\frac{||P_i^{-1}|_{p_i} \cdot x_i|_{p_i} + ||P_i^{-1}|_{p_i} \cdot y_i|_{p_i}}{p_i} \right]. \quad (5.81)$$

Лемма доказана. □

5.4 Разработка методов вычисления ранга числа с использованием приближенного метода

Теорема 5.4.1. Если для фиксированного N , $\forall i \in \overline{1, n}$ и $x_i \in [1, p_i - 1]$ выполняется условие

$$2^N \geq \frac{x_i \cdot ||P_i^{-1}|_{p_i} \cdot 2^N|_{p_i}}{|x_i \cdot |P_i^{-1}|_{p_i}|_{p_i}}, \quad (5.82)$$

то

$$\sum_{i=1}^n \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor = \sum_{i=1}^n \left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor, \quad (5.83)$$

где $k_i = \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot 2^N}{p_i} \right\rfloor$.

Доказательство. Подставим $k_i = \left\lfloor \frac{1}{p_i} \cdot |P_i^{-1}|_{p_i} \cdot 2^N \right\rfloor$ в $\left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor$, получим

$$\left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor = \left\lfloor \left\lfloor \frac{1}{p_i} \cdot |P_i^{-1}|_{p_i} \cdot 2^N \right\rfloor \cdot \frac{x_i}{2^N} \right\rfloor. \quad (5.84)$$

Так как $\left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot 2^N}{p_i} \right\rfloor = \frac{1}{p_i} \cdot \left(|P_i^{-1}|_{p_i} \cdot 2^N - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \right)$, то (5.84) примет вид

$$\begin{aligned} \left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor &= \left\lfloor \frac{1}{p_i} \cdot \left(|P_i^{-1}|_{p_i} \cdot 2^N - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \right) \cdot \frac{x_i}{2^N} \right\rfloor \\ &= \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \cdot \frac{x_i}{2^N \cdot p_i} \right\rfloor. \end{aligned} \quad (5.85)$$

Учитывая, что $\forall a \in \mathbb{Z}, b \in \mathbb{N}: \frac{a}{b} = \left\lfloor \frac{a}{b} \right\rfloor + \left\{ \frac{a}{b} \right\} = \left\lfloor \frac{a}{b} \right\rfloor + \frac{|a|_b}{b}$, и $\forall c \in \mathbb{R}, d \in \mathbb{Z}: \lfloor c + d \rfloor = \lfloor c \rfloor + d$, (5.85) преобразуется к виду

$$\begin{aligned} \left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor &= \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor \\ &\quad + \left\lfloor \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \frac{x_i}{2^N \cdot p_i} \right\rfloor. \end{aligned} \quad (5.86)$$

Из формулы (5.86) следует, что $\left\lfloor \frac{k_i \cdot x_i}{2^N} \right\rfloor = \left\lfloor \frac{|P_i^{-1}|_{p_i} \cdot x_i}{p_i} \right\rfloor$, тогда и только тогда, когда $\left\lfloor \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \frac{x_i}{2^N \cdot p_i} \right\rfloor = 0$. Условие $\left\lfloor \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \frac{x_i}{2^N \cdot p_i} \right\rfloor = 0$ выполняется, если

$$0 \leq \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \frac{x_i}{2^N \cdot p_i} < 1. \quad (5.87)$$

Умножим (5.87) на $p_i \cdot 2^N > 0$, получим

$$0 \leq 2^N \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \cdot x_i < 2^N \cdot p_i. \quad (5.88)$$

Так как $\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \leq p_i - 1$ и $\left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \geq 0$, то $\forall N \in \mathbb{N}: 2^N \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \cdot x_i < 2^N \cdot (p_i - 1)$. Следовательно, правая часть неравенства (5.88) выполняется при любом N . Таким образом, $\left\lfloor \frac{1}{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \frac{x_i}{2^N \cdot p_i} \right\rfloor = 0$, если выполняется неравенство

$$0 \leq 2^N \cdot \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \cdot x_i. \quad (5.89)$$

Рассмотрим два случая.

Случай 1. Если $x_i = 0$, то $\left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} - \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \cdot x_i = 0$, и неравенство (5.89) выполняется при любом N .

Случай 2. Если $x_i \neq 0$, то неравенство (5.89) примет вид

$$2^N \geq \frac{x_i \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}}{\left| x_i \cdot |P_i^{-1}|_{p_i} \right|_{p_i}}.$$

Теорема доказана. □

Следствие 5.4.1 (верхняя граница). Пусть модули RNS – попарно взаимно простые числа $p_1 < p_2 < \dots < p_n$, тогда существует хотя бы одно значение $N \leq \lceil 2 \log_2(p_n - 1) \rceil$, удовлетворяющее условию Теоремы 5.4.1.

Доказательство. Так как $1 \leq x_i \leq p_i - 1$, $\left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i} \leq p_i - 1$ и $\left| x_i \cdot |P_i^{-1}|_{p_i} \right|_{p_i} \geq 1$, то

$$\frac{x_i \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}}{\left| x_i \cdot |P_i^{-1}|_{p_i} \right|_{p_i}} \leq \frac{(p_i - 1)^2}{1} \leq (p_n - 1)^2.$$

Следовательно, если $2^N \geq (p_n - 1)^2$, то условие Теоремы 5.4.1 выполняется. Таким образом, при выборе $N = \lceil 2 \log_2(p_n - 1) \rceil$, условие Теоремы 5.4.1 выполняется. Значит существует хотя бы одно $N \leq \lceil 2 \log_2(p_n - 1) \rceil$, удовлетворяющее условию Теоремы 5.4.1.

Следствие доказано. □

Следствие 5.4.2 (нижняя граница). Для любого $N < \log_2 U$ условие Теоремы 5.4.1 не выполняется, где $U = \max_{i=1, \dots, n, \gcd(p_i, 2)=1} |P_i|_{p_i}$.

Доказательство. Так как $x_i \neq 0$, то его можно представить в виде $x_i = |a \cdot P_i|_{p_i}$, где $a \in [1, p_i - 1]$ и $a = \left| x_i \cdot P_i^{-1} \right|_{p_i}$. Вычислим значения правой части неравенства (5.82) в точках x_i , получим

$$\frac{x_i \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}}{\left| x_i \cdot |P_i^{-1}|_{p_i} \right|_{p_i}} = \frac{|a \cdot P_i|_{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}}{\left| |a \cdot P_i|_{p_i} \cdot |P_i^{-1}|_{p_i} \right|_{p_i}}. \quad (5.90)$$

Так как $\left| \left| a \cdot P_i \right|_{p_i} \cdot \left| P_i^{-1} \right|_{p_i} \right|_{p_i} = a$, то формула (5.90) примет вид

$$\frac{x_i \cdot \left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i}}{\left| x_i \cdot \left| P_i^{-1} \right|_{p_i} \right|_{p_i}} = \frac{\left| a \cdot P_i \right|_{p_i}}{a} \cdot \left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i}. \quad (5.91)$$

Покажем, что $\frac{\left| a \cdot P_i \right|_{p_i}}{a} \leq \left| P_i \right|_{p_i}$, для этого представим $\left| a \cdot P_i \right|_{p_i}$ в виде $\left| a \cdot P_i \right|_{p_i} = \left| a \cdot \left| P_i \right|_{p_i} \right|_{p_i} = a \cdot \left| P_i \right|_{p_i} - p_i \cdot \left\lfloor \frac{a \cdot \left| P_i \right|_{p_i}}{p_i} \right\rfloor$. Так как $a \cdot \left| P_i \right|_{p_i} \geq 1$, то $\left\lfloor \frac{a \cdot \left| P_i \right|_{p_i}}{p_i} \right\rfloor \geq 0$, следовательно, $\left| a \cdot P_i \right|_{p_i} \leq a \cdot \left| P_i \right|_{p_i}$, значит $\frac{\left| a \cdot P_i \right|_{p_i}}{a} \leq \frac{a \cdot \left| P_i \right|_{p_i}}{a} \leq \left| P_i \right|_{p_i}$. Обратим внимание на тот факт, что в неравенстве $\frac{\left| a \cdot P_i \right|_{p_i}}{a} \leq \left| P_i \right|_{p_i}$ равенство достигается при $a = 1$. Подставляя полученный результат в (5.91), получим

$$\frac{x_i \cdot \left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i}}{\left| x_i \cdot \left| P_i^{-1} \right|_{p_i} \right|_{p_i}} \leq \left| P_i \right|_{p_i} \cdot \left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i}. \quad (5.92)$$

Равенство в неравенстве (5.92) достигается в точке $x_i = \left| P_i \right|_{p_i}$.

Если $\gcd(p_i, 2) = 1$, то $\left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i} \geq 1$, следовательно, если $2^N < U$, то условие Теоремы 5.4.1 не будет выполнено хотя бы в одной точке $x_t = U$, где t – индекс, при котором $U = \left| P_t \right|_{p_t}$.

Следствие доказано. \square

Из формулы (5.92) следует, что для того, чтобы найти минимальное N , для которого бы выполнялось условие Теоремы 5.4.1, необходимо и достаточно проверить условие $\forall i = \overline{1, n}: 2^N \geq \left| P_i \right|_{p_i} \cdot \left| \left| P_i^{-1} \right|_{p_i} \cdot 2^N \right|_{p_i}$.

Пример 5.4.1. Пусть задана RNS с модулями $p_1 = 23, p_2 = 25, p_3 = 27, p_4 = 29$. Вычислим наименьшее N , удовлетворяющее условиям Теоремы 5.4.1.

Вычислим диапазон RNS: $P = \prod_{i=1}^n p_i = 23 \cdot 25 \cdot 27 \cdot 29 = 450225$.

Вычислим P_i : $P_1 = \frac{P}{p_1} = \frac{450225}{23} = 19575$, $P_2 = \frac{P}{p_2} = \frac{450225}{25} = 18009$, $P_3 = \frac{P}{p_3} = \frac{450225}{27} = 16675$, $P_4 = \frac{P}{p_4} = \frac{450225}{29} = 15525$.

Вычислим $\left| P_i \right|_{p_i}$: $\left| P_1 \right|_{p_1} = \left| 19575 \right|_{23} = 2$, $\left| P_2 \right|_{p_2} = \left| 18009 \right|_{25} = 9$, $\left| P_3 \right|_{p_3} = \left| 16675 \right|_{27} = 16$, $\left| P_4 \right|_{p_4} = \left| 15525 \right|_{29} = 10$.

Следовательно, $U = \max_{i=\overline{1, n}, \gcd(p_i, 2)=1} \left| P_i \right|_{p_i} = \max(2, 9, 16, 10) = 16$, значит $N \geq \log_2 U = 4$. С другой стороны, $N \leq \lceil 2 \log_2 (p_n - 1) \rceil = \lceil 2 \log_2 28 \rceil = 10$.

Таким образом, минимальное N , удовлетворяющее условиям Теоремы 5.4.1, принадлежит отрезку $4 \leq N \leq 10$.

Таблица 30 — Значения $|P_i|_{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}$ для $i = \overline{1, 4}$ и $N = \overline{4, 10}$

| N | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|-----|-----|-----|-----|------------|-----|------|
| $b_{1,N} = P_1 _{p_1} \cdot \left P_1^{-1} _{p_1} \cdot 2^N \right _{p_1}$ | 16 | 32 | 18 | 36 | 26 | 6 | 12 |
| $b_{2,N} = P_2 _{p_2} \cdot \left P_2^{-1} _{p_2} \cdot 2^N \right _{p_2}$ | 216 | 207 | 189 | 153 | 81 | 162 | 99 |
| $b_{3,N} = P_3 _{p_3} \cdot \left P_3^{-1} _{p_3} \cdot 2^N \right _{p_3}$ | 16 | 32 | 64 | 128 | 256 | 80 | 160 |
| $b_{4,N} = P_4 _{p_4} \cdot \left P_4^{-1} _{p_4} \cdot 2^N \right _{p_4}$ | 190 | 90 | 180 | 70 | 140 | 280 | 270 |
| $b_{max,N} = \max(b_{1,N}, b_{2,N}, b_{3,N}, b_{4,N})$ | 216 | 207 | 189 | 153 | 256 | 280 | 270 |
| 2^N | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |

Найдем минимальное N , удовлетворяющее условию Теоремы 5.4.1. Для этого вычислим значения $|P_i|_{p_i} \cdot \left| |P_i^{-1}|_{p_i} \cdot 2^N \right|_{p_i}$, результаты вычислений занесем в таблицу 30. Исходя из полученных результатов (табл. 30) делаем вывод, что минимальное N , при котором выполняются условия Теоремы 5.4.1, равно $N = 8$.

Из результатов, полученных в примере 5.4.1, можно сделать вывод, что при выборе 5-битных модулей RNS необходимо использовать $N = 8$ бит, что в 1.6 раза превышает размер модулей. Возникает задача: выбрать модули RNS так, чтобы N максимально приблизилось к нижней границе. Одним из путей решения данной задачи является наложение дополнительных условий на модули RNS, например, если $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, тогда $N \leq \lceil \log_2 p_n \rceil$. Исследуем вопрос о количестве наборов модулей, удовлетворяющих указанному условию, для $n \leq 10$. Для этого докажем следующие утверждения.

Лемма 5.4.1. Если модули RNS $\{p_1, p_2, \dots, p_n\}$ удовлетворяют условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, то $n \geq 3$.

Доказательство. Покажем, что не существует двухмодульной RNS, удовлетворяющей условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$. Без потери общности будем считать, что выполняется неравенство $p_1 < p_2$. Предположим, что существует двухмодульная RNS, удовлетворяющая условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$. Следовательно, $|p_1|_{p_2} = 1$ и $|p_2|_{p_1} = 1$. Из условия $|p_1|_{p_2} = 1$ следует, что p_1 можно представить в виде $p_1 = b \cdot p_2 + 1$, где $b \in \mathbb{Z}$. Учитывая, что $p_1 < p_2$, $b \cdot p_2 + 1 < p_2$, сле-

довательно, $b < 1 - \frac{1}{p_2}$, значит $b \leq 0$. Обращая внимание на то, что, с одной стороны, $p_1 \geq 2$, с другой стороны, $p_1 \leq 1$, обнаружим противоречие. Следовательно, двухмодульной RNS, удовлетворяющей условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, не существует.

Лемма доказана. \square

Пример 5.4.2. Пусть RNS задана модулями $\{2, 3, 5\}$, тогда $P = p_1 \cdot p_2 \cdot p_3 = 30$, $P_1 = \frac{P}{p_1} = 15$, $P_2 = \frac{P}{p_2} = 10$, $P_3 = \frac{P}{p_3} = 6$, следовательно, $|P_1|_{p_1} = |15|_2 = 1$, $|P_2|_{p_2} = |10|_3 = 1$ и $|P_3|_{p_3} = |6|_5 = 1$. Значит для данного набора модулей выполняется условие $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.

Лемма 5.4.2. Модули RNS $\{p_1, p_2, \dots, p_n\}$ удовлетворяют условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$ тогда и только тогда, когда

$$P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1. \quad (5.93)$$

Доказательство. Так как $|P_n|_{p_n} = 1$, P_n можно представить в виде $P_n = a \cdot p_n + 1$, где $a \in \mathbb{Z}_{P_n}$. Учитывая, что $\forall i = \overline{1, n-1}: |P_n|_{p_i} = 1$, то $\forall i = \overline{1, n-1}: |a \cdot p_n + 1|_{p_i} = 0$, следовательно, $a \equiv \left| -\frac{1}{p_n} \right|_{p_i}$. Вычислим значение a , используя

Китайскую теорему об остатках, получим $a = \left| \sum_{i=1}^{n-1} \left| \hat{P}_i^{-1} \right|_{p_i} \cdot \left| -\frac{1}{p_n} \right|_{p_i} \cdot \hat{P}_i \right|_{P_n}$.

Заметим, что $\forall i = \overline{1, n-1}: \left| \hat{P}_i^{-1} \right|_{p_i} \cdot \left| -\frac{1}{p_n} \right|_{p_i} = \left| -\frac{1}{P_i} \right|_{p_i}$. Из условия теоремы следует, что $|P_i|_{p_i} = 1$, значит $\left| -\frac{1}{P_i} \right|_{p_i} = p_i - 1$. Таким образом,

$a = \left| \sum_{i=1}^{n-1} (p_i - 1) \cdot \hat{P}_i \right|_{P_n}$. Так как $\hat{P}_i \cdot p_i = P_n$, то

$$a = \left| \sum_{i=1}^{n-1} (p_i - 1) \cdot \hat{P}_i \right|_{P_n} = \left| \sum_{i=1}^{n-1} P_n - \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n} = P_n - \left| \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n}. \quad (5.94)$$

Учитывая, что $a = P_n - \left| \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n}$, получим

$$P_n = a \cdot p_n + 1 = p_n \cdot \left(P_n - \left| \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n} \right) + 1 = P - p_n \cdot \left| \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n} + 1. \quad (5.95)$$

Так как

$$\begin{aligned} p_n \cdot \left| \sum_{i=1}^{n-1} \hat{P}_i \right|_{P_n} &= p_n \cdot \left(\sum_{i=1}^{n-1} \hat{P}_i - P_n \cdot \left[\frac{\sum_{i=1}^{n-1} \hat{P}_i}{P_n} \right] \right) \\ &= \sum_{i=1}^{n-1} P_i - P \cdot \left[\frac{\sum_{i=1}^{n-1} P_i}{P} \right] = \left| \sum_{i=1}^{n-1} P_i \right|_P. \end{aligned} \quad (5.96)$$

Следовательно, $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1$.

Докажем, что если $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1$, то $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$. Так как $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1$, то, следовательно, выполняется сравнение

$$P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1 \pmod{p_i}.$$

Учитывая, что $\forall i = \overline{1, n}: P \equiv 0 \pmod{p_i}$ и $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P \equiv P_i \pmod{p_i}$, получим $\forall i = \overline{1, n}: P_i \equiv 1 \pmod{p_i}$.

Лемма доказана. \square

Лемма 5.4.3. *Если модули RNS $p_1 < p_2 < \dots < p_n$ удовлетворяют условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$ и $2 \leq n \leq 58$, то $\sum_{i=1}^n P_i = P + 1$.*

Доказательство. Заметим, что $\forall i \in \overline{1, n}: |\sum_{i=1}^n P_i|_{p_i} = |P_i|_{p_i}$. Согласно условию леммы, $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, следовательно, $\forall i \in \overline{1, n}: |\sum_{i=1}^n P_i|_{p_i} = 1$, значит, согласно Китайской теореме об остатках, $|\sum_{i=1}^n P_i|_P = 1$. Из равенства $|\sum_{i=1}^n P_i|_P = 1$ следует, что $\sum_{i=1}^n P_i = 1 + a \cdot P$, где $a \in \mathbb{Z}$. Так как $n \geq 2$, то $P_{n-1} \geq 2$, $P_n \geq 3$ и $\sum_{i=1}^n P_i \geq 5$. Следовательно, $a \geq \frac{4}{P}$. Учитывая, что $a \in \mathbb{Z}$, получим $a \geq 1$. С другой стороны, $\sum_{i=1}^n P_i = P \sum_{i=1}^n \frac{1}{p_i}$. Так как $p_1 \geq 2$, $p_2 \geq 3$, и так далее, $p_n \geq pr_n$, где pr_i – последовательность простых чисел, то $\sum_{i=1}^n \frac{1}{p_i} \leq \sum_{i=1}^n \frac{1}{pr_i}$. Учитывая, что $\sum_{i=1}^{58} \frac{1}{pr_i} \approx 1.998$ и $\sum_{i=1}^{59} \frac{1}{pr_i} \approx 2.002$, получим, если $n \leq 58$, то справедливо неравенство $1 + a \cdot P < 2 \cdot P$, следовательно, $a < 2 - \frac{1}{P} < 2$. Так как $a \in \mathbb{Z}$, $a \geq 1$ и $a < 2$, получим $a = 1$, следовательно, если $n \leq 58$, то $\sum_{i=1}^n P_i = 1 + P$.

Лемма доказана. \square

Из Леммы 5.4.3 следует, что если $2 \leq n \leq 58$ и выполнены условия Леммы 5.4.2, то $SQ = P + 1$.

Исследуем подробнее вопрос существования трех-, четырех- и пяти-модульных RNS, удовлетворяющих условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.

Теорема 5.4.2. Пусть модули RNS – попарно взаимно простые числа $p_1 < p_2 < \dots < p_n$, удовлетворяющие условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, тогда справедливы следующие утверждения

1. Если $n = 3$, то модули RNS $\{2, 3, 5\}$.
2. Если $n = 4$, то модули RNS $\{2, 3, 7, 41\}$ или $\{2, 3, 11, 13\}$.
3. Если $n = 5$, то модули RNS $\{2, 3, 7, 43, 1805\}$, $\{2, 3, 7, 83, 85\}$ или $\{2, 3, 11, 17, 59\}$.

Доказательство. Рассмотрим первое утверждение, когда $n = 3$. Для начала покажем, что условие теоремы выполняется только тогда, когда $p_1 = 2$, для этого предположим противное. Пусть существует трехмодульная RNS, удовлетворяющая условию теоремы, для которой $p_1 \geq 3$. Так как $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$ и $n = 3 \leq 58$, то согласно Лемме 5.4.3, выполняется равенство $SQ = P + 1$. Разделим $SQ = P + 1$ на P , получим $\sum_{i=1}^3 \frac{1}{p_i} = 1 + \frac{1}{P}$, т.е. $\sum_{i=1}^3 \frac{1}{p_i} > 1$. С другой стороны, так как $p_1 \geq 3$, то $\sum_{i=1}^3 \frac{1}{p_i} \leq \frac{1}{3} + \frac{1}{4} + \frac{1}{5} = \frac{47}{60} < 1$, следовательно, имеет место противоречие, значит $p_1 = 2$.

Подставляя $p_1 = 2$ в $SQ = P + 1$, получим $2 \cdot p_2 + 2 \cdot p_3 + p_2 \cdot p_3 = 2 \cdot p_2 \cdot p_3 + 1$, следовательно,

$$p_3 = \frac{2 \cdot p_2 - 1}{p_2 - 2} = 2 + \frac{3}{p_2 - 2}. \quad (5.97)$$

Учитывая, что $p_3 \in \mathbb{N}$, из формулы (5.97) следует, что $\frac{3}{p_2 - 2} \in \mathbb{N}$, значит $p_2 - 2 = 1$ или $p_2 - 2 = 3$. Решая уравнения, получим $p_2 = 3$ или $p_2 = 5$. Подставляя в формулу (5.97) $p_2 = 3$, получим $p_3 = 5$, а подставив $p_2 = 5$, получим $p_3 = 3$. Учитывая, что $p_1 < p_2 < p_3$, существует единственный набор трехмодульной RNS, удовлетворяющий условию теоремы, – $\{2, 3, 5\}$.

Докажем второе утверждение, для $n = 4$. Если $p_1 \geq 3$, то $\sum_{i=1}^4 \frac{1}{p_i} \leq \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{7} = \frac{389}{420} < 1$, следовательно, аналогично случаю при $n = 3$, получаем, что $p_1 = 2$. Предположим, что существует набор модулей RNS, удовлетворяющий условию теоремы, для которого $p_2 > 3$, следовательно, $p_2 \geq 5$. Тогда $\frac{1}{2} + \frac{1}{p_2} + \frac{1}{p_3} + \frac{1}{p_4} > 1$, откуда $\frac{1}{p_2} + \frac{1}{p_3} + \frac{1}{p_4} > \frac{1}{2}$. Так как $5 \leq p_2 < p_3 < p_4$ и $\forall i = \overline{2, 4}: \gcd(2, p_i) = 1$, $\frac{1}{p_2} + \frac{1}{p_3} + \frac{1}{p_4} \leq \frac{1}{5} + \frac{1}{7} + \frac{1}{9} = \frac{143}{315} < \frac{1}{2}$. Следовательно, имеет место противоречие и $p_2 = 3$. Подставляя $p_1 = 2$ и $p_2 = 3$ в уравнение $SQ = P + 1$, и

выражая p_4 через p_3 , получим

$$p_4 = \frac{6 \cdot p_3 - 1}{p_3 - 6} = 6 + \frac{35}{p_3 - 6}. \quad (5.98)$$

Так как $p_3, p_4 \in \mathbb{N}$ и $3 < p_3 < p_4$, то $p_3 - 6 = 1$ или $p_3 - 6 = 5$. Следовательно, $p_3 = 7$ или 11 , а $p_4 = 41$ или 13 соответственно. Значит, существует два набора четырехмодульных RNS, удовлетворяющих условию теоремы, — $\{2, 3, 7, 41\}$ и $\{2, 3, 11, 13\}$.

Рассмотрим третье утверждение, для $n = 5$. Так как если $p_1 \geq 4$, то $\sum_{i=1}^5 \frac{1}{p_i} \leq \frac{1}{4} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \frac{1}{11} = \frac{11017}{13860} < 1$, $p_1 = 2$ или $p_1 = 3$.

Пусть $p_1 = 2$. Предположим, что $p_2 \geq 7$, тогда $\sum_{i=2}^5 \frac{1}{p_i} \leq \frac{1}{7} + \frac{1}{9} + \frac{1}{11} + \frac{1}{13} = \frac{3800}{9009} < \frac{1}{2}$. С другой стороны, $\sum_{i=2}^5 \frac{1}{p_i} = \frac{1}{2} + \frac{1}{P}$, имеет место противоречие, следовательно, $3 \leq p_2 < 7$. Значит, если $p_1 = 2$, то $p_2 = 3$ или $p_2 = 5$.

Рассмотрим случай, когда $p_1 = 2$ и $p_2 = 3$, тогда

$$\sum_{i=3}^5 \frac{1}{p_i} = 1 - \frac{1}{2} - \frac{1}{6} + \frac{1}{P} = \frac{1}{6} + \frac{1}{P}. \quad (5.99)$$

Если $p_3 \geq 17$, то левая часть равенства (5.99) удовлетворяет неравенству $\sum_{i=3}^5 \frac{1}{p_i} \leq \frac{1}{17} + \frac{1}{19} + \frac{1}{23} = \frac{1151}{7429} < \frac{1}{6}$, при этом правая часть равенства (5.99) больше $\frac{1}{6}$. Следовательно, $p_3 \leq 16$ и $\gcd(p_3, 6) = 1$, отсюда $p_3 = 7, 11$ или 13 .

Если $p_1 = 2, p_2 = 3$ и $p_3 = 7$, то

$$p_5 = \frac{42 \cdot p_4 - 1}{p_4 - 42} = 42 + \frac{1763}{p_4 - 42}. \quad (5.100)$$

Учитывая, что $1763 = 41 \cdot 43$, получим, $p_4 - 42 = 1$ или $p_4 - 42 = 41$. Следовательно, $p_4 = 43$ или 83 , а $p_5 = 1805$ или 85 соответственно. Таким образом, основания RNS, удовлетворяющие условию теоремы, — $\{2, 3, 7, 43, 1805\}$ и $\{2, 3, 7, 83, 85\}$.

Если $p_1 = 2, p_2 = 3$ и $p_3 = 11$, то

$$p_5 = \frac{66 \cdot p_4 - 1}{5 \cdot p_4 - 66} = 13 + \frac{p_4 + 857}{5 \cdot p_4 - 66}. \quad (5.101)$$

Так как $\frac{p_4 + 857}{5 \cdot p_4 - 66} \in \mathbb{N}$, то $\frac{p_4 + 857}{5 \cdot p_4 - 66} \geq 1$, следовательно, $p_4 + 857 \geq 5 \cdot p_4 - 66$, значит $p_4 \leq \frac{923}{4}$. Проверив все возможные варианты, получим что для $p_4 \in [14, 230]$: $\frac{p_4 + 857}{5 \cdot p_4 - 66} \in \mathbb{N}$, $p_4 = 17$ или 59 , следовательно, $p_5 = 59$ или 4 , соответственно. Учитывая, что $p_1 < p_2 < p_3 < p_4 < p_5$, получим, что условию теоремы в этом случае удовлетворяют основания RNS $\{2, 3, 11, 17, 59\}$.

Если $p_1 = 2$, $p_2 = 3$ и $p_3 = 13$, то

$$p_5 = \frac{78 \cdot p_4 - 1}{7 \cdot p_4 - 78} = 11 + \frac{p_4 + 857}{7 \cdot p_4 - 78}. \quad (5.102)$$

Учитывая, что $\frac{p_4+857}{7 \cdot p_4 - 78} \in \mathbb{N}$, $\frac{p_4+857}{7 \cdot p_4 - 78} \geq 1$, следовательно, $p_4 + 857 \geq 7 \cdot p_4 - 78$, значит $p_4 \in [14, 155]$. Проверяем, все возможные значения получаем, что при указанных условиях уравнение в целых числах решения не имеет.

Рассмотрим случай, когда $p_1 = 2$ и $p_2 = 5$. Для таких RNS $\sum_{i=3}^5 \frac{1}{p_i} = 1 - \frac{1}{2} - \frac{1}{5} - \frac{1}{P} = \frac{3}{10} + \frac{1}{P}$. Если $p_3 \geq 9$, то $\sum_{i=3}^5 \frac{1}{p_i} \leq \frac{1}{9} + \frac{1}{11} + \frac{1}{13} = \frac{359}{1287} < \frac{3}{10}$. С другой стороны, $\sum_{i=3}^5 \frac{1}{p_i} = \frac{3}{10} + \frac{1}{P} > \frac{3}{10}$, следовательно, $5 < p_3 < 9$ и $\gcd(p_3, 10) = 1$, значит $p_3 = 7$.

Подставим $p_1 = 2$, $p_2 = 5$ и $p_3 = 7$ в равенство $SQ = P + 1$, получим

$$p_5 = \frac{70 \cdot p_4 - 1}{11 \cdot p_4 - 70} = 6 + \frac{4 \cdot p_4 + 419}{11 \cdot p_4 - 70}. \quad (5.103)$$

Учитывая, что $7 < p_4 < p_5$ и $p_5 \in \mathbb{N}$, $\frac{4 \cdot p_4 + 419}{11 \cdot p_4 - 70} \in \mathbb{N}$, следовательно,

$$\begin{cases} p_4 & > 7, \\ 11 \cdot p_4 - 70 & > 0, \\ \frac{4 \cdot p_4 + 419}{11 \cdot p_4 - 70} & \geq 3. \end{cases} \quad (5.104)$$

Решая систему неравенств получим, что $7 < p_4 \leq \frac{629}{29}$. Учитывая, что $p_4 \in \mathbb{N}$ и $\gcd(p_4, 70) = 1$, возможными решениями являются $p_4 \in \{9, 11, 13, 17, 19\}$. Проверим какие из чисел $\{9, 11, 13, 17, 19\}$ при подстановке в $f(p_4) = \frac{4 \cdot p_4 + 419}{11 \cdot p_4 - 70}$ удовлетворяют условию $f(p_4) \in \mathbb{N}$, получим $f(9) = \frac{455}{29} \notin \mathbb{N}$, $f(11) = \frac{463}{51} \notin \mathbb{N}$, $f(13) = \frac{471}{73} \notin \mathbb{N}$, $f(17) = \frac{487}{117} \notin \mathbb{N}$ и $f(19) = \frac{495}{139} \notin \mathbb{N}$. Значит не существует пятимодульных RNS с модулями $p_1 = 2$ и $p_2 = 5$, удовлетворяющих условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.

Рассмотрим случай, когда $p_1 = 3$, тогда $\sum_{i=2}^5 \frac{1}{p_i} = 1 - \frac{1}{3} + \frac{1}{P} = \frac{2}{3} + \frac{1}{P} > \frac{2}{3}$. Если $p_2 \geq 5$, то $\sum_{i=2}^5 \frac{1}{p_i} \leq \frac{1}{5} + \frac{1}{7} + \frac{1}{8} + \frac{1}{11} = \frac{1721}{3080} < \frac{2}{3}$, таким образом, имеет место противоречие, следовательно, $3 < p_2 < 5$, значит $p_2 = 4$.

Оценим значения, которые может принимать p_3 . $\sum_{i=3}^5 \frac{1}{p_i} = 1 - \frac{1}{3} - \frac{1}{4} + \frac{1}{P} = \frac{5}{12} + \frac{1}{P} > \frac{5}{12}$. Если $p_3 \geq 7$, то $\sum_{i=3}^5 \frac{1}{p_i} \leq \frac{1}{7} + \frac{1}{11} + \frac{1}{13} = \frac{311}{1001} < \frac{5}{12}$, следовательно, имеет место противоречие, и p_3 удовлетворяет условиям $4 < p_3 < 7$ и $\gcd(p_3, 12) = 1$, значит $p_3 = 5$.

Подставим $p_1 = 3$, $p_2 = 4$ и $p_3 = 5$ в $SQ = P + 1$, получим

$$p_5 = \frac{60 \cdot p_4 - 1}{13 \cdot p_4 - 60} = 4 + \frac{8 \cdot p_4 + 239}{13 \cdot p_4 - 60}. \quad (5.105)$$

Учитывая, что $5 < p_4 < p_5$, $\gcd(p_5, 60) = 1$ и $p_5 \in \mathbb{N}$, $p_4 \geq 7$, $p_5 \geq 11$, $\frac{8 \cdot p_4 + 239}{13 \cdot p_4 - 60} \geq 7$ и $\frac{8 \cdot p_4 + 239}{13 \cdot p_4 - 60} \in \mathbb{N}$, следовательно

$$\begin{cases} p_4 & \geq 7, \\ 13 \cdot p_4 - 60 & > 0, \\ \frac{8 \cdot p_4 + 239}{13 \cdot p_4 - 60} & \geq 7. \end{cases} \quad (5.106)$$

Учитывая, что $p_4 \in \mathbb{N}$, $p_4 = 7$ – единственное решение, удовлетворяющее системе неравенств (5.106). Проверим является ли натуральным числом $g(p_4) = \frac{8 \cdot p_4 + 239}{13 \cdot p_4 - 60}$ в точке $p_4 = 7$, получим $g(7) = \frac{295}{31} \notin \mathbb{N}$. Следовательно, не существует пятимодульных RNS, для которых выполнялись бы условия $p_1 = 3$ и $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.

Теорема доказана. \square

Теорема 5.4.3. *Не существует попарно взаимно простых чисел (оснований RNS) $p_1 < p_2 < \dots < p_n < 2 \cdot p_1$, удовлетворяющих условиям $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.*

Доказательство. Покажем, что если $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, то $p_1 < n$. Предположим противное, что существует набор модулей RNS, для которых выполняются условия $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$ и $p_1 \geq n$. Так как выполняются условия $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, то, следовательно, выполняются условия Леммы 5.4.2, и $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1$. Разделим равенство $P_n + \left| \sum_{i=1}^{n-1} P_i \right|_P = P + 1$ на P , получим

$$\frac{1}{p_n} + \left\{ \sum_{i=1}^{n-1} \frac{1}{p_i} \right\} = 1 + \frac{1}{P}, \quad (5.107)$$

где $\{x\}$ – дробная часть вещественного числа x .

Так как $n \leq p_1 < p_2 < \dots < p_n$, то $\sum_{i=1}^{n-1} \frac{1}{p_i} \leq \sum_{i=1}^{n-1} \frac{1}{n} = \frac{n-1}{n} < 1$, следовательно, $\left\{ \sum_{i=1}^{n-1} \frac{1}{p_i} \right\} = \sum_{i=1}^{n-1} \frac{1}{p_i}$, тогда формула (5.107) примет вид

$$\sum_{i=1}^n \frac{1}{p_i} = 1 + \frac{1}{P}. \quad (5.108)$$

Учитывая, что $n \leq p_1 < p_2 < \dots < p_n$, левая часть равенства (5.108) удовлетворяет условию $\sum_{i=1}^n \frac{1}{p_i} \leq \sum_{i=1}^n \frac{1}{n} = 1$, а правая часть $1 + \frac{1}{P} > 1$, следовательно, если $p_1 \geq n$, то имеет место противоречие. Таким образом, не существует RNS с модулями, удовлетворяющими условиям $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$ и $p_1 \geq n$. Значит условием $p_1 < n$ является необходимым.

С другой стороны, ранее было показано, что $p_n \geq p_1 + 2 \cdot n - 4$. Учитывая, что согласно условию теоремы $p_n < 2 \cdot p_1$, $p_n < 2 \cdot n$. Решая неравенство $p_1 + 2 \cdot n - 4 < 2 \cdot n$, получим, что условию $p_1 < p_2 < \dots < p_n < 2 \cdot p_1$ могут удовлетворять значения $p_1 < 4$, т.е. $p_1 = 2$ и $p_1 = 3$. Если $p_1 = 2$, то $p_n < 4$, следовательно, если и существуют подходящие RNS, то только двухмодульная RNS $\{2, 3\}$, что противоречит Лемме 5.4.1, согласно которой $n \geq 3$ при указанных в формулировке теоремы условиях. Рассмотрим случай, если $p_1 = 3$. Тогда $p_n < 6$, следовательно, возможен только один вариант, при условии что $n \geq 3$, равный $\{3, 4, 5\}$, но он не удовлетворяет условию $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$. Следовательно, не существует модулей RNS, удовлетворяющих условиям: $p_1 < p_2 < \dots < p_n < 2 \cdot p_1$ и $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$.

Теорема доказана. □

Из Теоремы 5.4.3 следует, что если модули RNS удовлетворяют условиям $\forall i = \overline{1, n}: |P_i|_{p_i} = 1$, то они не являются компактной или сбалансированной последовательностью.

5.5 Выводы по пятой главе

Пятая глава посвящена разработке высокопроизводительных методов и алгоритмов вычисления ранга чисел, представленных в RNS. Основным приложением функции ранга числа, представленного в RNS, являются алгоритмы обнаружения и исправления ошибок арифметических вычислений, и от эффективности его вычисления во многом зависит производительность указанных алгоритмов.

Рассмотрены три формы ранга числа: классическая форма ранга, следующая из Китайской теоремы об остатках, нормализованный ранг числа и ранг числа, построенный с использованием функции ядра Акушского. Исследован вопрос об интерполяции функции ранга числа с помощью алгебраических многочленов. Доказан ряд теорем, позволяющих утверждать, что не существует многочлена, заданного над \mathbb{Z}_P , позволяющего вычислить ранг числа, представленного в RNS, вне зависимости от его формы.

Предложен эффективный метод вычисления ранга числа, основанный на использовании функции ядра Акушского, не содержащей критических ядер. Доказаны теоремы, дающие оценку верхней и нижней границ разрядности констант при использовании приближенного метода для вычисления ранга числа. Показано, что наборы модулей, удовлетворяющие полученной оценке, не являются компактной последовательностью. Предложенный метод позволяет сократить объем необходимых вычислений и увеличить скорость вычисления ранга числа по сравнению с приближенным методом: для нахождения ранга числа с использованием приближенного метода необходимо выполнить n операций с числами, превышающими значение модуля, тогда как в предлагаемом методе необходимо выполнить $\frac{n(n-1)}{2}$ операций с числами, не превышающими значение модуля.

Разработаны алгоритмы вычисления ранга числа, представленного в RNS, и доказаны теоремы, позволяющие осуществлять контроль результатов обработки закодированных чисел с использованием арифметических свойств классического и нормализованного рангов.

Глава 6. РАЗРАБОТКА МЕТОДОВ ПОВЫШЕНИЯ НАДЕЖНОСТИ СИСТЕМ ОБРАБОТКИ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ДВУХУРОВНЕВОЙ RRNS

Безопасное и отказоустойчивое облачное хранилище должно предотвращать несанкционированный доступ, использование, раскрытие информации, нарушение, изменение и т.д. Конфиденциальность, целостность и доступность должны сохраняться даже при наличии сбоев, угроз, как преднамеренных, так и случайных.

С этой целью широко используются системы шифрования данных, гомоморфные вычисления, коды исправления ошибок, защищенные структуры доступа и т.д.

Мультиоблачная среда имеет динамический характер с рисками потери информации, отказа в доступе, утечки информации, сговора и нарушений безопасности данных, которые трудно прогнозировать и предвидеть заранее. Эти типы нестационарности являются одной из основных проблем при проектировании надежного хранилища, способного снизить влияние на систему связанных с ними негативных последствий.

Многие потенциальные клиенты предпочитают не использовать облачные хранилища из-за риска раскрытия данных. Механизмов шифрования данных недостаточно для обеспечения безопасности и конфиденциальности. Для реализации удаленной обработки данные должны быть расшифрованы, что является исходной проблемой уязвимости данных.

В последнее десятилетие наблюдается значительный интерес к использованию широко известной и глубоко изученной избыточной системы остаточных классов (Redundant Residue Number System – RRNS) в качестве варианта реализации полностью гомоморфной схемы кодирования.

Применение кодов исправления ошибок на основе RRNS связано с тремя основными проблемами: низкая скорость кодирования/декодирования в/из RRNS, высокая вычислительная сложность алгоритма исправления ошибок, сравнительно небольшое количество исправляемых ошибок относительно вносимой избыточности. Таким образом, первая задача – увеличить скорость кодирования и декодирования данных из позиционной системы счисления в RRNS

и обратно [11, 145, 283, 309]. Вторая задача связана с уменьшением вычислительной сложности алгоритма исправления ошибок [34, 157, 220]. Третья задача заключается в увеличении количества исправляемых с помощью RRNS ошибок [157, 220, 250].

В данной главе предложена двухуровневая схема RRNS(2 Level back propagation RRNS – 2Lbp-RRNS) для обеспечения надежного и безопасного хранения данных в нескольких облаках. Схема имеет масштабируемую структуру доступа к данным. В отличие от классических решений, она способна восстанавливать данные при меньшем количестве доступных долей по сравнению с современными подходами, обеспечивая конфиденциальность и безопасность хранимых данных.

6.1 Подходы к повышению надежности и безопасности обрабатываемых конфиденциальных данных

В основе кодов исправления ошибок лежит идея Хэмминга о добавлении дополнительных данных, которые помогают обнаруживать и исправлять ошибки [231]. В зависимости от области применения подходы к построению систем исправления ошибок различаются.

Для систем хранения важен баланс между надежностью и избыточностью данных, поскольку избыточность влияет на объем хранимых данных и, следовательно, на затраты. Самый затратный механизм обеспечения надежности – это репликация данных.

С другой стороны, коды исправления ошибок и их модификации, такие как коды стирания и регенерационные коды, могут обеспечить большую надежность с такой же избыточностью, что и при репликации [285]. Важным параметром кода исправления является максимальное количество ошибок, которое он может обнаружить и исправить при фиксированной избыточности данных [11]. Данный параметр является одним из определяющих при выборе того или иного алгоритма.

Другим не менее важным параметром при выборе кода, исправляющего ошибки, является набор моделей ошибок, характерных для конкретной системы. Можно выделить три большие группы, объединяющие в себе схожие модели

ошибок: перманентные ошибки, блоковые (пакетные) ошибки и автономные (битовые) ошибки. Блочные коды [285] оптимизированы для исправления ошибок, находящихся в одном блоке, однако, исправляют меньшее количество автономных ошибок по сравнению со сверхточными кодами. Негативное влияние перманентных ошибок, как правило, снижается посредством встроенных аппаратных модулей.

Альтернативным решением являются модулярные корректирующие коды, построенные с использованием RRNS [145]. Дополнительным преимуществом RRNS при проектировании распределенных систем хранения является то, что RRNS реализует структуру доступа, обеспечивающую безопасность данных [29]. К основным недостаткам можно отнести большую вычислительную сложность и объем памяти, необходимой для эффективной реализации алгоритма декодирования данных.

На этапе декодирования широко используются два метода: метод модулярных проекций и метод синдромного декодирования. Метод модулярных проекций – универсальный метод, позволяющий обнаруживать и исправлять ошибки, используя модули RRNS общего вида [34]. Его недостаток – экспоненциально возрастающая вычислительная сложность, имеющая место при возрастании количества исправляемых ошибок [11]. Для снижения вычислительной сложности метода модулярных проекций используют вспомогательные функции, например, расстояние Хэмминга и ранг числа в RNS [11].

Метод синдромного декодирования снижает вычислительную сложность до квадратичной, но требует хранения больших таблиц констант в памяти [206]. Снизить объем требуемой памяти можно, наложив дополнительные условия на остатки по некоторым модулям RRNS. В частности, предполагая, что некоторые остатки абсолютно надежны (не содержат ошибок), можно многократно снизить объемы хранимых в памяти констант. В свою очередь, надежность доверенных остатков обеспечивается другими дополнительными методами отказоустойчивого кодирования, что отрицательно сказывается на быстродействии всей системы. Так же метод синдромного декодирования имеет опцию увеличения корректирующей способности без изменения параметров схемы RRNS [206]. Для реализации данной возможности требуется разбалансировать модули RRNS, используя один или несколько модулей превосходящих в несколько раз остальные по размеру [23]. Недостаток такого подхода становится очевиден при

возникновении ошибок в этих больших модулях RRNS: количество некорректных данных становится значительным и может превышать пороговое значение.

Альтернативным подходом является использование 2L-RRNS. Учитывая, что модули второго уровня действуют как независимый код коррекции ошибок, этот подход позволяет исправить большее количество ошибок [207,345], а также снизить вычислительную сложность декодирования [250,353].

Одна из основных задач облачных технологий – предоставление бесперебойного доступа к данным посредством сервиса с достаточно простыми и понятными интерфейсами, не требующими привлечения специалистов. Классические подходы к обеспечению целостности данных основаны на методах идентичного или неидентичного резервирования (хранение копий или хранение историй соответственно). Стратегии организации надежного хранения данных обычно формируются для каждой конкретной системы индивидуально на основе решения задачи многофакторного анализа [334].

Задача обеспечения целостности данных сложна. Она включает в себя не только контроль целостности, но также поддержание и восстановление данных в случае их потери или искажения по какой-либо причине.

Существуют различные способы решения проблемы мониторинга и обеспечения целостности данных. Один из них – вычисление контрольных сумм и сравнение их со справочными контрольными суммами [279,284]. Другие способы основаны на использовании криптографических методов, ключевого и безключевого хеширования, а также электронной подписи [152,194,363]. Недостатком этих методов является невозможность обеспечить целостность без дополнительных данных для механизмов восстановления.

Введение избыточности – широко распространенное решение для обеспечения целостности данных. Примерами являются аппаратные и программные реализации избыточного массива независимых дисков (Redundant Array of Independent Disks – RAID) [187,319], методов дублирования, отказоустойчивого кодирования [33] и т.д. Недостатком этих методов является невозможность управления данными как теоретически безопасной информацией и высокая избыточность.

Некоторые методы позволяют контролировать целостность путем сравнения контрольных значений и вычисленных хэш-кодов (контрольных сумм) при запросе данных. Однако, отсутствие механизмов их восстановления не позволяет обеспечить целостность.

Напротив, другие методы обеспечивают целостность данных, восстанавливая их, например, из резервной копии. Однако, их практическая реализация без возможности предварительного контроля целостности данных малоэффективна.

Некоторые методы позволяют контролировать и обеспечивать целостность данных, но за счет высокой избыточности. Одно из таких решений – последовательное использование криптографических преобразований и технологий резервного копирования.

Альтернативный способ – использование RRNS, которая, с одной стороны, представляет собой корректирующий код, позволяющий восстановить результат при возникновении ошибки, а с другой стороны, является структурой доступа, обеспечивающей безопасность данных [11]. Tchernykh и др. в работе [10] показали зависимость безопасности хранимых данных от параметров RRNS. Использование 2L-RRNS позволяет увеличить количество обнаруживаемых и исправляемых ошибок по сравнению с 1L-RRNS (классической RRNS) [250]. Следовательно, 2L-RRNS обеспечивает более высокий уровень надежности и целостности хранимых данных по сравнению с 1L-RRNS.

6.2 Обеспечение надежности и конфиденциальности данных с использованием двухуровневой RRNS

RNS как версия гомоморфного кодирования может использоваться для сохранения конфиденциальности при организации облачных вычислений. RNS позволяет вычислять функции от закодированных данных (выполнять операции над остатками), не зная набора модулей. Владелец может восстановить реальный результат по результатам вычислений на соответствующих закодированных данных. Данная характеристика делает RNS многообещающим решением для безопасного делегирования обработки конфиденциальных данных клиентов облачным вычислительным серверам.

После того, как Rivest и др. в работе [299] представили концепцию гомоморфных вычислений, криптографы предложили и проанализировали множество различных гомоморфных криптосистем.

Brickell и Yacobi в работе [169], а также Paillier в работе [303] предложили частично гомоморфное кодирование с одной арифметической операцией (сложение или умножение).

Gentry [213] разработал первую полностью гомоморфную схему кодирования на основе идеальных решеток. Данная схема реализует вычисления над закодированными данными с неограниченным количеством гомоморфных умножений и сложений.

Все полностью гомоморфные криптосистемы можно разделить на две категории.

Первая категория содержит полностью гомоморфные схемы кодирования с открытым ключом, основанные на вводе шума [158,167,205,213]. Эти криптосистемы основаны на технике Gentry, улучшающей их производительность [213].

Криптосистема Gentry безопасна и устойчива к различным атакам, но имеет высокую вычислительную сложность, что не позволяет использовать ее в практических приложениях. Чтобы уменьшить вычислительную сложность, в работе [23] был предложен аналог схемы Gentry на основе RNS. Tchernykh и др. в работе [29] показали, что RNS обеспечивает необходимый уровень безопасности и снижает вычислительную сложность кодирования и декодирования.

Ко второй категории относятся полностью гомоморфные криптосистемы, построенные над кольцом вычетов с делителями нуля и не использующие шум. Примеры таких криптографических схем представлены в работах [205, 259] и др. Полностью гомоморфные вычисления над кольцом вычетов с делителями нуля основаны на диагональных матрицах. Использование диагональных матриц снижает вычислительную сложность алгоритмов кодирования и декодирования, но при этом снижается безопасность данных. В работах [368, 369] представлена полностью гомоморфная схема кодирования, основанная на алгебре октонионов; однако, в работе [362] было показано, что данная схема небезопасна. Альтернативный подход к построению полностью гомоморфных схем кодирования заключается в использовании RNS. В работах [11, 29] показано, что за счет RNS можно снизить вычислительную сложность алгоритмов кодирования и декодирования и обеспечить необходимый уровень безопасности.

В следующих разделах показано, как улучшить технические характеристики вычислительной системы посредством использования двухуровневой RRNS (2Lbp-RRNS).

В таблицах 31 и 32 представлены и описаны необходимые для введения 2Lbp-RRNS обозначения.

Таблица 31 — Обозначения для схемы 2Lbp-RRNS

| Обозначение | Определение |
|---|---|
| D, S | Оригинал, исходные данные (двоичные числа в позиционной системе счисления) |
| \tilde{S} | Представление S в 2L-RRNS |
| $\bar{S} \stackrel{RRNS}{\leftarrow} S + E$ | Представление S с ошибкой E в 2L-RRNS |
| $\text{size}(D)$ | Размер оригинальных данных D |
| D_n | n младших битов числа D |
| T_E, T_D | Время кодирования и декодирования данных |
| t_{dow}, t_{up} | Время скачивания и загрузки из/в облака |
| V_u, V_d | $V_u = \frac{\text{size}(D)}{T_E + t_{up}}, V_d = \frac{\text{size}(D)}{T_D + t_{dow}}$ |
| \bar{I} | Кортеж остатков с ошибкой на 1-ом уровне |
| I_D | Подмножество $\{1, \dots, n_1\}$, $ I_D = k_1$ |
| I_E | Подмножество $\{1, \dots, n_1\}$, $ I_E = \lfloor \frac{k_1 + n_1}{2} \rfloor$ |
| \bar{I}_i | Кортеж остатков с ошибкой на 2-ом уровне |
| N_D^{2L} | Количество обнаруживаемых ошибок в 2L-RRNS |
| N_E^{2L} | Количество исправляемых ошибок в 2L-RRNS |
| N_D^{2Lbp} | Количество обнаруживаемых ошибок в 2Lbp-RRNS |
| N_E^{2Lbp} | Количество исправляемых ошибок в 2Lbp-RRNS |
| N_{DI} | Количество обнаруживаемых ошибок при заранее известной локализации |
| N_{EI} | Количество исправляемых ошибок при заранее известной локализации |

6.2.1 Одноуровневая RRNS

Пусть $p_{1,1}, p_{1,2}, \dots, p_{1,n_1}$ – попарно взаимно простые числа, используемые в качестве набора модулей 1L-RRNS, $n_1 = k_1 + r_1$. Допустимый динамический диапазон 1L-RRNS определяется как $P = \prod_{i=1}^{k_1} p_{1,i}$.

Таблица 32 — Обозначения параметров первого и второго уровня схемы 2Lbp-RRNS

| Обозначение | Определение |
|---|--|
| Первый уровень | |
| n_1 | Количество модулей первого уровня |
| $k_1 \leq n_1$ | Значение порога на первом уровне структуры доступа |
| $r_1 = n_1 - k_1$ | Количество контрольных (избыточных) модулей |
| $p_{1,i}$ | i -ые RRNS модули на первом уровне |
| $P = \prod_{i=1}^{k_1} p_{1,i}$ | Динамический диапазон RRNS на первом уровне, $S \in [0, P)$ |
| $\bar{P} = \prod_{i=1}^{n_1} p_{1,i}$ | $[0, \bar{P})$ – полный диапазон RRNS |
| $S_i = S _{p_{1,i}}$ | Остаток от деления S на модуль $p_{1,i}$ |
| Второй уровень | |
| $n_{2,i}$ | Количество модулей $p_{2,i,1}, p_{2,i,2}, \dots, p_{2,i,n_{2,i}}$, используемых для представления $S_i, i = \overline{1, n_1}$ |
| $k_{2,i} \leq n_{2,i}$ | Пороговое значение структуры доступа, используемой для представления S_i |
| $r_{2,i} = n_{2,i} - k_{2,i}$ | Количество контрольных (избыточных) модулей, используемых в представлении S_i |
| $p_{2,i,j}$ | j -ый RRNS модуль, используемый для представления $S_i, i = \overline{1, n_1}, j = \overline{1, n_{2,i}}$ |
| $M_{2,i} = \prod_{j=1}^{k_{2,i}} p_{2,i,j}$ | $M_i = \prod_{j=1}^{k_{2,i}} p_{2,i,j}$ Динамический диапазон RRNS, используемой для представления $S_i \in [0, M_{2,i}), i = \overline{1, n_1}$ |
| $S_{i,j} = S_i _{p_{2,i,j}}$ | Остаток от деления S_i на модуль $p_{2,i,j}, i = \overline{1, n_1}, j = \overline{1, n_{2,i}}$ |

S – число в двоично-взвешенной системе счисления, где $S \in [0, P)$. S представлено в RRNS кортежем

$$S \xrightarrow{RRNS} (S_1, S_2, \dots, S_n), \quad (6.1)$$

где S_i – остаток от деления S на $p_{1,i}$.

Согласно свойствам 1L-RRNS, система может обнаруживать $r_1 = n_1 - k_1$ и исправлять $\lfloor \frac{r_1}{2} \rfloor$ ошибок.

При классическом подходе к локализации и исправлению ошибок в коде RRNS используются проекционные методы, где количество рассчитываемых

модулярных проекций растёт экспоненциально в зависимости от r_1 . Как следствие, 1L-RRNS неприменима на практике без значительной оптимизации.

Celesti и др. в работе [145] предложили использовать 1L-RRNS для построения надежных и масштабируемых облачных систем хранения. Операции с остатками можно выполнять независимо и параллельно, что упрощает и ускоряет вычисления. Избыточные остатки позволяют системе обнаруживать и исправлять множественные ошибки.

Поскольку представление чисел в 1L-RRNS можно рассматривать как структуру доступа, появляется возможность построения вычислительно безопасного хранилища данных.

Gomathisankaran и др. в работе [223], изучив полностью гомоморфные системы кодирования, основанные на RRNS ориентированных структурах доступа, отмечают, что использовать модули RRNS в качестве секретных ключей нецелесообразно. Это приводит к высокой избыточности и ресурсоемкому декодированию, сравнимому по сложности с прямыми методами решения исходной проблемы.

Cheon и др. в работе [174] предложили альтернативный способ построения гомоморфной системы кодирования на основе RRNS. Авторы предложили обобщение алгоритма DGHV (Dijk, Gentry, Halevi и Vaikuntanathan), способствующее улучшению основных характеристик: снижению вычислительной сложности и избыточности. Предложенная схема основана на структуре доступа с использованием 1L-RRNS из работы [151] и имеет высокую избыточность по сравнению со схемами, использующими классическую 1L-RRNS.

Проблемы использования 1L-RRNS для обнаружения и исправления ошибок хранения и обработки данных подробно рассмотрены в работе [157]. В работах [11, 220] предложены модификации процедур обнаружения и исправления ошибок с помощью 1L-RRNS, направленные на снижение их вычислительной сложности. Отметим, что большинство работ, связанных с разработкой и оптимизацией алгоритмов обнаружения и исправления ошибок с использованием 1L-RRNS, рассматривают задачу обнаружения и исправления однократной ошибки. Данный факт объясняется соотношением вероятностей возникновения однократной и многократной ошибок: вероятность возникновения многократной ошибки в разы меньше вероятности возникновения однократной ошибки, поэтому однократным ошибкам уделяется гораздо большее внимание. Данный тезис не работает, когда речь идет об обработке больших данных, при которой

необходимы эффективные алгоритмы обнаружения и исправления нескольких ошибок.

Схема 1L-RRNS способна обеспечить безопасность, надежность и масштабируемость при хранении и обработке больших данных. Она объединяет в себе свойства кодов исправления ошибок и функционал двух криптографических примитивов: структуры доступа и гомоморфного кода, что делает ее полезной для обработки данных в закодированном виде.

6.2.2 Кодирование и декодирование данных в двухуровневой RRNS

Конструктивная версия Китайской теоремы об остатках содержит метод восстановления S из представления в RRNS. При использовании схемы RRNS с параметрами (k_1, n_1) , значение S может быть восстановлено по любым k_1 остаткам из n_1 (при условии их корректности).

Чтобы гарантировать требуемый динамический диапазон, можно использовать либо большое количество маленьких модулей, либо несколько больших модулей. При использовании большого количества малых модулей преобразование чисел из RRNS в двоичную систему счисления является более сложным в вычислительном отношении. Кроме того, должны быть разработаны эффективные программные и аппаратные реализации основных модульных операций.

2L-RRNS является рекурсивным расширением классической 1L-RRNS. На первом уровне модули $p_{1,1}, p_{1,2}, \dots, p_{1,n_1}$ используются для расчета долей S_1, S_2, \dots, S_{n_1} . На втором уровне каждая доля S_i преобразуется в набор остатков $S_{i,j} = |S_i|_{p_{2,i,j}}$ своим собственным набором модулей $p_{2,i,1}, p_{2,i,2}, \dots, p_{2,i,n_{2,i}}$ (рис. 6.1). При этом S_i удовлетворяет условию $S_i < p_{1,i}$ для всех $i = \overline{1, n_1}$. Из CRT следует, что для взаимно однозначного отображения между $S_i \in [0, p_{1,i})$ и $\tilde{S}_i = (S_{i,1}, S_{i,2}, \dots, S_{i,n_{2,i}})$ необходимо и достаточно, чтобы $M_{2,i} \geq p_{1,i}$ для каждого $i = \overline{1, n_1}$

$$M_{2,i} = \prod_{j=1}^{k_{2,i}} p_{2,i,j} \geq p_{1,i}. \quad (6.2)$$

На рисунке 6.2 показана схема декодирования данных. Каждая из долей S_i , $i = \overline{1, n_1}$, восстанавливается по соответствующим остаткам $S_{i,j}$, затем S

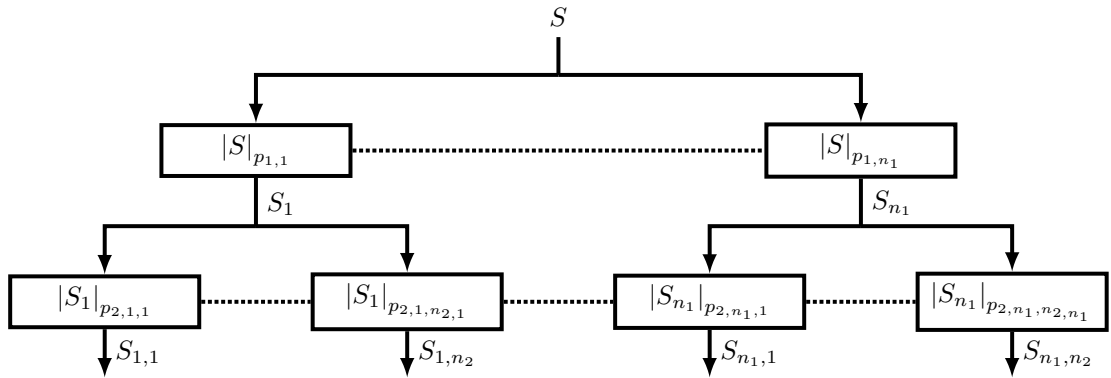


Рисунок 6.1 — 2L-RRNS кодирование

восстанавливается по S_1, S_2, \dots, S_{n_1} . Многооперандный сумматор по модулю

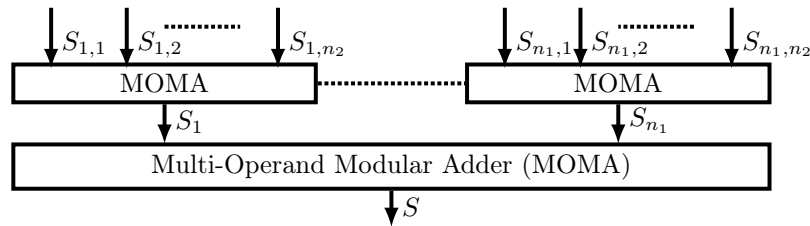


Рисунок 6.2 — 2L-RRNS декодирование

(MOMA) – алгоритмический примитив, который принимает на вход n_1 операндов S_1, S_2, \dots, S_{n_1} , где $0 \leq S_i < p_{1,i}$ для каждого $i = \overline{1, n_1}$, и вычисляет остаток от деления их суммы на модуль P . То есть реализует следующую формулу

$$S = |w_i \cdot S_1 + w_2 \cdot S_2 + \dots + w_{n_1} \cdot S_{n_1}|_P, \quad (6.3)$$

где $w_i = P_i \cdot |P_i^{-1}|_{p_{1,i}}$ и $P_i = \frac{P}{p_{1,i}}$, для всех $i = \overline{1, n_1}$.

6.3 Двухуровневая RRNS

В данном разделе представлено решение 2Lbp-RRNS, являющееся расширением классической 2L-RRNS. Приведен теоретический анализ, формулировка и доказательства основных ее свойств. Также показана зависимость надежности и производительности системы от (k, n) параметров RRNS на каждом уровне.

Особое внимание уделено механизмам, использующим и расстояние Хэмминга, которые позволяют увеличить количество обнаруживаемых и исправляемых ошибок. Кроме того, представлен сравнительный анализ верхних границ

количества обнаруживаемых и исправляемых ошибок для предлагаемой и классической схем, позволяющий оценить преимущества предлагаемого решения.

6.3.1 Алгоритм коррекции ошибок с использованием двухуровневой RRNS

2L-RRNS использует код коррекции ошибок, основанный на классической 1L-RRNS, на каждом уровне. Он может исправить значение S_i тогда и только тогда, когда количество ошибок меньше или равно $\lfloor \frac{r_{2,i}}{2} \rfloor$. Во всех остальных случаях, аналогично 1L-RRNS, значение S_i будет восстановлено неверно.

Рассмотрим Алгоритм 5, «Коррекция ошибок в 2L-RRNS», реализующий общий метод модулярных проекций 1L-RRNS на каждом уровне. Использование синдромного метода не рассматривается, т.к. объем памяти, требуемый для его реализации в 1L-RRNS, увеличивается экспоненциально в зависимости от количества исправляемых ошибок [11]. При использовании двухуровневого алгоритма, объем памяти, требуемой для хранения таблиц констант второго уровня, увеличивается в n_1 раз, что приводит к увеличению ресурсопотребления до уровня, неприемлемого для современных корректирующих алгоритмов. Функция CRTtoBin преобразует числа из RRNS в двоичную систему счисления с помощью CRT. Функция ProRRNS реализует метод модулярных проекций, возвращает значение '0', если ошибки были исправлены, и '1', в противном случае, а также сохраняет значение корректной проекции в переменную val .

Количество обнаруживаемых и исправляемых 2L-RRNS ошибок, N_D^{2L} и N_E^{2L} соответственно, может быть вычислено по следующим формулам [250]

$$N_D^{2L} = \sum_{i=1}^{n_1} (n_{2,i} - k_{2,i}), \quad (6.4)$$

$$N_E^{2L} = \sum_{i=1}^{n_1} \left\lfloor \frac{n_{2,i} - k_{2,i}}{2} \right\rfloor. \quad (6.5)$$

Для лучшего понимания свойств 2Lbp-RRNS рассмотрим частные случаи 1L-RRNS и 2L-RRNS, когда номера ошибочных остатков известны (предварительно локализованы). Используем специальные индексы для обозначения обнаружения Dl и исправления El заранее локализованных ошибок.

Алгоритм 5: Коррекция ошибок в 2L-RRNS

Input: $(k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$
 $S_1 \xrightarrow{RRNS} (S_{1,1}, S_{1,2}, \dots, S_{1,n_{2,1}}), S_2 \xrightarrow{RRNS} (S_{2,1}, S_{2,2}, \dots, S_{2,n_{2,2}}), \dots,$
 $S_{n_1} \xrightarrow{RRNS} (S_{n_1,1}, S_{n_1,2}, \dots, S_{n_1,n_{2,n_1}}),$
 $(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}})$

Output: $S, flag, (S_1, S_2, \dots, S_{n_1})$

1. $flag = 0$, если нет ошибок.
2. $flag = 1$, если ошибки обнаружены и исправлены.
3. $flag = -1$, если ошибки не исправлены.

```

1   $flag = 0;$ 
2  for  $i = 1, n_1, i++$  do
3       $S'_i = \text{CRTtoBin}((S_{i,1}, S_{i,2}, \dots, S_{i,n_{2,i}}), (p_{2,i,1}, p_{2,i,2}, \dots, p_{2,i,n_{2,i}}));$ 
4      if  $S'_i \geq M_{2,i}$  then
5           $temp = \text{ProRRNS}(S'_i, (p_{2,i,1}, p_{2,i,2}, \dots, p_{2,i,n_{2,i}}), k_{2,i}, val);$ 
6           $flag = flag + temp;$ 
7          if  $temp == 0$  then
8               $S_i = val;$ 
9          else
10              $S_i = -1;$ 
11         else
12              $S_i = S'_i;$ 
13 if  $flag == 0$  then
14      $S = \text{CRTtoBin}((S_1, S_2, \dots, S_{n_1}), (p_{1,1}, p_{1,2}, \dots, p_{1,n_1}));$ 
15 else
16     if  $flag \leq \lfloor \frac{n_1 - k_1}{2} \rfloor$  then
17          $flag = 1;$ 
18          $S = \text{CRTtoBin}((S_1, S_2, \dots, S_{n_1}), (p_{1,1}, p_{1,2}, \dots, p_{1,n_1}));$ 
19     else
20          $flag = -1$ 

```

Result: $S, flag, (S_1, S_2, \dots, S_{n_1})$

Лемма 6.3.1. Для (k_1, n_1) 1L-RRNS, если заранее известна локализация k_1 правильных остатков S_i , исходные данные S могут быть восстановлены.

Доказательство. Без ограничения общности пусть правильными значениями будут $S_{i_1}, S_{i_2}, \dots, S_{i_{k_1}}$. С помощью CRT можно восстановить S по формуле

$$S = \left| w_1 S_{i_1} + w_2 S_{i_2} + \dots + w_{k_1} S_{i_{k_1}} \right|_{P_I}, \quad (6.6)$$

где $P_I = \prod_{j=1}^{k_1} p_{i_j}$, $w_j = \frac{P_I}{p_{i_j}} \cdot \left| \frac{p_{i_j}}{P_I} \right|_{p_{i_j}}$.

Лемма доказана. \square

Свойство 6.3.1. Для (k_1, n_1) 1L-RRNS, если заранее известна локализация k_1 правильных остатков S_i , 1L-RRNS может исправить $N_{El}^{1L} \leq r_1 = n_1 - k_1$ ошибок.

Доказательство. Если известны номера корректных остатков $S_{i_1}, S_{i_2}, \dots, S_{i_{k_1}}$, то выполняется условие Леммы 6.3.1. Следовательно, исходные данные могут быть восстановлены, а по ним вычислены корректные значения оставшихся $r_1 = n_1 - k_1$ ошибочных остатков.

Если количество корректных остатков не превышает значения $k_1 - 1$, учитывая, что (k_1, n_1) является пороговой структурой доступа, истинное значение S не может быть восстановлено.

Следовательно, если существует алгоритм, способный определить, какие из остатков S_i правильные, можно исправить не более $r_1 = n_1 - k_1$ ошибок.

Свойство доказано. \square

Лемма 6.3.2. Если существует алгоритм, способный определить, какие из остатков $S_{i,j}$ правильные, то 2L-RRNS может исправить

$$N_{El}^{2L} \leq \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i} \quad (6.7)$$

ошибок.

Доказательство. Без потери общности, предположим, что $k_{2,1} \leq k_{2,2} \leq \dots \leq k_{2,n_1}$.

Предположим, что в остатках \tilde{S}_i возникло $r_{2,i}$ ошибок для всех $i \in \overline{1, k_1}$ и $n_{2,i}$ ошибок для всех $i \in \overline{k_1 + 1, n_1}$. Суммарное количество ошибок для первой группы остатков равно $\sum_{i=1}^{k_1} r_{2,i} = \sum_{i=1}^{k_1} (n_{2,i} - k_{2,i})$, а для второй $\sum_{i=k_1+1}^{n_1} n_{2,i}$. Следуя Лемме 6.3.1, можно восстановить истинные значения S_i

для всех $i \in \overline{1, k_1}$. Следовательно, условие Леммы 6.3.1 выполнено, и можно восстановить истинное значение S , исправив N_{El}^{2L} ошибок, где

$$\begin{aligned}
N_{El}^{2L} &\leq \sum_{i=1}^{k_1} (n_{2,i} - k_{2,i}) + \sum_{i=k_1+1}^{n_1} n_{2,i} \\
&= \sum_{i=1}^{k_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i} + \sum_{i=k_1+1}^{n_1} n_{2,i} \\
&= \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i}.
\end{aligned} \tag{6.8}$$

Если мы добавим к остаткам \tilde{S}_i еще одну ошибку (без ограничения общности будем считать, что ошибка добавлена в представление S_j), $i \in \overline{1, k_1}$, то, согласно Лемме 6.3.1, невозможно восстановить фактические значения S_i для всех $i \in \{k_1 + 1, k_1 + 2, \dots, n_1\} \cup \{j\}$. Т.е. возможно восстановить истинные значения S_i только для $i \in \{1, 2, \dots, k_1\} \setminus \{j\}$, следовательно, невозможно восстановить реальное значение S .

Таким образом, количество исправляемых 2L-RRNS ошибок, при условии, что известна их локализация, равно

$$N_{El}^{2L} \leq \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i}. \tag{6.9}$$

Лемма доказана. □

6.3.2 Разработка методов и алгоритмов коррекции ошибок с использованием двухуровневой RRNS

Чтобы увеличить количество обнаруживаемых и исправляемых ошибок, в 2Lbp-RRNS используются расстояние Хэмминга (Hamming Distance – HD).

Пример 6.3.1. Рассмотрим пример, когда на первом и втором уровнях используется схема (2, 3). Следовательно, $k_1 = k_{2,1} = k_{2,2} = k_{2,3} = 2$, $n_1 = n_{2,1} = n_{2,2} = n_{2,3} = 3$.

Алгоритм 6: Коррекция ошибок в 2Lbp-RRNS

Input: $(k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$
 $S_1 \xrightarrow{RRNS} (S_{1,1}, S_{1,2}, \dots, S_{1,n_{2,1}}), S_2 \xrightarrow{RRNS} (S_{2,1}, S_{2,2}, \dots, S_{2,n_{2,2}}), \dots,$
 $S_{n_1} \xrightarrow{RRNS} (S_{n_1,1}, S_{n_1,2}, \dots, S_{n_1,n_{2,n_1}}),$
 $(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}})$

Output: $S, flag$

- 1 2L-RRNS вычисляет S и $flag$. Если флаг $flag \neq -1$, S восстанавливается, иначе S не восстанавливается.
 - 2 Выбирая из $S_{i,1}, S_{i,2}, \dots, S_{i,n_{2,i}}$ подмножества $k_{2,i}$ элементов из фиксированного набора из $n_{2,i}$ элементов, вычисляем возможные значения S_i^l для каждого из S_i .
 - 3 Выбирая подмножества k_1 элементов из полученных S_i^l , восстанавливаем возможные значения S^j функцией CRTtoBin.
 - 4 Используя концепцию обратного распространения ошибки, кодируем каждое S^j в 2L-RRNS-представление \tilde{S}^j и вычисляем HD между \tilde{S}^j и \bar{S} .
 - 5 Выберем \tilde{S}^j , для которого HD минимально. Если минимальное HD между \tilde{S}^j и \bar{S} больше $N_E^{2Lbp} = \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i} - 1$, то возвращается $flag = -1$; в противном случае $S = S^j$ и $flag = 1$.
-

На первом уровне имеем кортеж из трех элементов (S_1, S_2, S_3) . На втором уровне имеем три кортежа $(S_{1,1}, S_{1,2}, S_{1,3})$, $(S_{2,1}, S_{2,2}, S_{2,3})$ и $(S_{3,1}, S_{3,2}, S_{3,3})$.

Допустим, что возникли ошибки в $S_{1,1}, S_{1,2}, S_{1,3}$. Тогда традиционная 2L-RRNS обнаруживает ошибки, но не способна их исправить.

На первом шаге 2Lbp-RRNS, пытаясь восстановить S_1 , используем три возможных кортежа для восстановления S_1^1, S_1^2 и S_1^3 .

$$S_1^1 \xrightarrow{RNS} \tilde{S}_1^1 = (S_{1,1}, S_{1,2}), S_1^2 \xrightarrow{RNS} \tilde{S}_1^2 = (S_{1,1}, S_{1,3}),$$

$$S_1^3 \xrightarrow{RNS} \tilde{S}_1^3 = (S_{1,2}, S_{1,3}). \quad (6.10)$$

Выполним аналогичные действия для восстановления S_2 , обозначив возможные кортежи

$$S_2^1 \xrightarrow{RNS} \tilde{S}_2^1 = (S_{2,1}, S_{2,2}), S_2^2 \xrightarrow{RNS} \tilde{S}_2^2 = (S_{2,1}, S_{2,3}),$$

$$S_2^3 \xrightarrow{RNS} \tilde{S}_2^3 = (S_{2,2}, S_{2,3}). \quad (6.11)$$

Чтобы восстановить S , проанализируем девять возможных кортежей-кандидатов первого уровня, обозначенных как

$$\begin{aligned} S^1 \xrightarrow{RNS} \tilde{S}^1 &= (S_1^1, S_2^1), S^2 \xrightarrow{RNS} \tilde{S}^2 = (S_1^1, S_2^2), S^3 \xrightarrow{RNS} \tilde{S}^3 = (S_1^1, S_2^3), \\ S^4 \xrightarrow{RNS} \tilde{S}^4 &= (S_1^2, S_2^1), S^5 \xrightarrow{RNS} \tilde{S}^5 = (S_1^2, S_2^2), S^6 \xrightarrow{RNS} \tilde{S}^6 = (S_1^2, S_2^3), \\ S^7 \xrightarrow{RNS} \tilde{S}^7 &= (S_1^3, S_2^1), S^8 \xrightarrow{RNS} \tilde{S}^8 = (S_1^3, S_2^2), S^9 \xrightarrow{RNS} \tilde{S}^9 = (S_1^3, S_2^3). \end{aligned} \quad (6.12)$$

Согласно концепции, каждое из девяти значений S^j преобразуем обратно в 2L-RRNS и обозначается \tilde{S}^j .

Для каждого \tilde{S}^j вычисляем HD между \bar{S} и \tilde{S}^j , $j = \overline{1, 9}$. В рассматриваемом примере HD равно трем для всех $j = \overline{2, 9}$ и равно двум между \bar{S} и \tilde{S}^1 . Заметим, что $k = 2$, следовательно, $S = S^1$. Восстанавливаем данные.

Теперь посчитаем количество обнаруживаемых и исправляемых 2Lbp-RRNS ошибок в общем случае.

Основная идея предложенного метода заключается в обратном распространении восстановленного варианта, который не может быть признан правильным или неправильным. Восстановленный, затем повторно закодированный вариант \tilde{S}^j сравнивается с первоначально закодированным значением \bar{S} посредством вычисления HD. Если HD меньше заданного порога, \tilde{S}^j принимается в качестве корректно восстановленного S .

Таким образом, исправление ошибок с использованием 2Lbp-RRNS включает в себя два дополнительных процесса: кодирование в 2L-RRNS и вычисление HD.

Количество возможных вариантов \tilde{S}^j зависит от количества ошибок, исправляемых используемой 2Lbp-RRNS схемой. Из-за большого количества возможных комбинаций время восстановления может значительно вырасти.

Обратное распространение восстановленного варианта, с одной стороны, увеличивает вычислительную сложность алгоритма обнаружения и исправления ошибок, с другой стороны, позволяет восстанавливать данные, содержащие большее количество ошибок.

Рассмотрим более подробно свойства HD, используемые при локализации ошибок в 2L-RRNS. Пусть имеется два 2L-RRNS представления S : без ошибок \tilde{S} и с ошибками \bar{S} .

Свойство 6.3.2. Если $HD(\tilde{S}, \bar{S}) = 0$, то \bar{S} не содержит ошибок, т.е. $\tilde{S} = \bar{S}$.

Доказательство. Для доказательства воспользуемся принципом от обратного. Предположим, что \bar{S} содержит ошибки и $HD(\tilde{S}, \bar{S}) = 0$. Поскольку \bar{S} содержит ошибки, существует представление $\bar{S} \stackrel{2L-RRNS}{\leftarrow} S + E$, где $0 < E < P$, поэтому

$$\bar{S} = \left((S'_{1,1}, \dots, S'_{1,n_{2,1}}), \dots, (S'_{n_{1,1}}, \dots, S'_{n_{1,n_{2,n_1}}}) \right), \quad (6.13)$$

$$\tilde{E} = \left((E_{1,1}, \dots, E_{1,n_{2,1}}), \dots, (E_{n_{1,1}}, \dots, E_{n_{1,n_{2,n_1}}}) \right), \quad (6.14)$$

$$\tilde{S} = \left((S_{1,1}, \dots, S_{1,n_{2,1}}), \dots, (S_{n_{1,1}}, \dots, S_{n_{1,n_{2,n_1}}}) \right), \quad (6.15)$$

где для всех i, j : $S'_{i,j} = S_{i,j} + E_{i,j}$.

Учитывая, что $0 < E < P$, то существует хотя бы одна пара (i, j) такая, что $E_{i,j} \neq 0$, поэтому существует хотя бы одно значение $S_{i,j} \neq S'_{i,j}$, тогда $HD(\tilde{S}, \bar{S}) > 0$. Получили противоречие. Следовательно, если $HD(\tilde{S}, \bar{S}) = 0$, то \bar{S} не содержит ошибок.

Свойство доказано. \square

Свойство 6.3.3. *Количество ошибок, содержащееся в 2L-RRNS-представлении \bar{S} , равно $HD(\tilde{S}, \bar{S})$.*

Доказательство. Доказательство напрямую следует из определения расстояния Хэмминга для чисел, представленных в RNS.

Свойство доказано. \square

Свойство 6.3.4. *Пусть для $S^1 \neq S^2 \neq \dots \neq S^t$ выполняется условие*

$$HD(\tilde{S}^1, \bar{S}) = HD(\tilde{S}^2, \bar{S}) = \dots = HD(\tilde{S}^t, \bar{S}), \quad (6.16)$$

тогда количество ошибок в каждом из представлений \tilde{S}^j одинаково.

Доказательство. Поскольку количество ошибок в 2L-RRNS представлении S^j , по Свойству 6.3.3, определяется $HD(\tilde{S}^j, \bar{S})$, из условия $HD(\tilde{S}^1, \bar{S}) = HD(\tilde{S}^2, \bar{S}) = \dots = HD(\tilde{S}^t, \bar{S})$ следует, что количество ошибок в каждом S^j одинаково для всех $j = \overline{1, t}$.

Свойство доказано. \square

Свойство 6.3.5. *Пусть для $S^1 \neq S^2 \neq \dots \neq S^t$ выполняется условие $HD(\tilde{S}^1, \bar{S}) < HD(\tilde{S}^2, \bar{S}) < \dots < HD(\tilde{S}^t, \bar{S})$, тогда представление S^1 в 2L-RRNS содержит наименьшее количество ошибок.*

Доказательство. Доказательство напрямую следует из Свойства 6.3.3.

Отметим, что если среди S^j найдутся не менее двух значений, без ограничения общности пусть это будут значения S^1 и S^2 , что $S^1 \neq S^2$, $HD(\tilde{S}^1, \bar{S}) = HD(\tilde{S}^2, \bar{S}) = d$ и $d \leq HD(\tilde{S}^j, \bar{S})$ для всех $j = \overline{1, t}$, то исправить \bar{S} невозможно, так как в этом случае невозможно определить, какое из двух значений S^1, S^2 является истинным.

Свойство доказано. \square

Теорема 6.3.1. *2Lbp-RRNS способна обнаруживать N_D^{2Lbp} и исправлять N_E^{2Lbp} ошибок, где*

$$N_D^{2Lbp} = \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i}, \quad (6.17)$$

$$N_E^{2Lbp} \leq \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i} - 1. \quad (6.18)$$

Доказательство. Из Свойства 6.3.4 следует, что максимальное количество ошибок, определяемое с помощью HD, равно максимальному количеству исправляемых ошибок когда известна их локализация, поэтому

$$N_D^{2Lbp} = \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i}. \quad (6.19)$$

Оценим сверху количество исправляемых 2Lbp-RRNS ошибок.

Аналогично традиционной 2L-RRNS, в каждом \tilde{S}^i локализовано не более $\lfloor \frac{r_{2,i}}{2} \rfloor$ ошибок. Следовательно, существуют $k_1 + \lfloor \frac{r_{2,i}}{2} \rfloor$ значений S_i , правильность которых может быть подтверждена.

Без ограничения общности будем предполагать, что каждое из представлений в 1L-RRNS $S_{i_q} \in \{S_{i_1}, S_{i_2}, \dots, S_{i_l}\}$ содержит не более r_{2,i_q} ошибок, где $l \geq k_1$. Обозначим $I = \{i_1, \dots, i_l\}$. Если существует $u \notin I$, для которого представление в 1L-RRNS S_u содержит меньше чем $n_{2,u}$ ошибок, то, используя Свойство 6.3.5, возможно восстановить S Согласно обратному распространению восстановленного варианта, каждый кандидат S^j для S кодируется обратно в представление RRNS и обозначается \tilde{S}^j .

Максимальное количество исправляемых ошибок в этом случае не превышает величины

$$N_E^{2Lbp} \leq \sum_{i=1}^{k_1} (n_{2,i} - k_{2,i}) + \sum_{i=k_1+1}^{n_1} n_{2,i} - 1 = \sum_{i=1}^{n_1} n_{2,i} - \sum_{i=1}^{k_1} k_{2,i} - 1. \quad (6.20)$$

Теорема доказана. □

6.3.3 Корректирующие свойства двухуровневой RRNS с обратным распространением ошибки

В данном разделе представлено сравнение свойств схем 2L-RRNS и 2Lbp-RRNS с различными параметрами, представленными в таблице 33. Здесь $(k_1, n_1) = (k_{2,i}, n_{2,i})$, следовательно, каждое хранилище имеет одинаковое количество долей с одинаковым порогом. На рисунках 6.3 и 6.4 показано количество обнаруживаемых и исправляемых ошибок. Отметим, что 2Lbp-RRNS способна обнаруживать и исправлять большее количество ошибок, чем 2L-RRNS для всех тестовых случаев. Согласно экспериментальным данным 2Lbp-RRNS может обнаруживать в среднем в 1.58 раза (рис. 6.3) и исправлять в 3.37 раза (рис. 6.4) больше ошибок, чем 2L-RRNS. В таблице 34 представлены средние

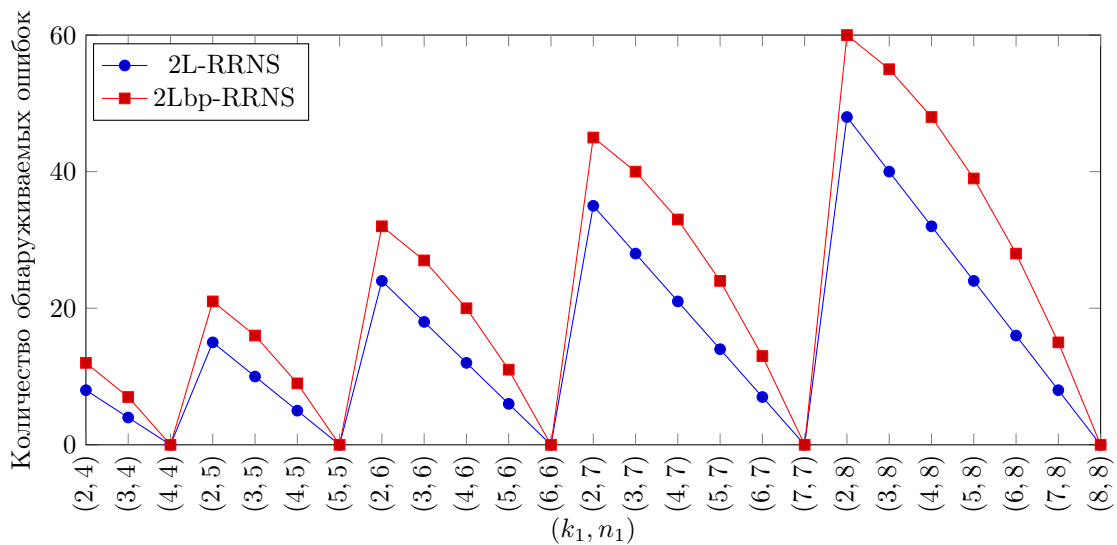


Рисунок 6.3 — Обнаружение ошибок в 2L-RRNS и 2Lbp-RRNS

скорости кодирования/декодирования для схем с параметрами, описанными в таблице 33, при разном размере данных.

Таблица 33 — Параметры схем 2L-RRNS (2Lbp-RRNS)

| id | k_1 | n_1 | $(k_{2,1}, n_{2,1})-\dots-(k_{2,n_1}, n_{2,n_1})$ |
|------|-------|-------|---|
| 1 | 2 | 4 | (2, 4)-(2, 4)-(2, 4)-(2, 4) |
| 2 | 3 | 4 | (3, 4)-(3, 4)-(3, 4)-(3, 4) |
| 3 | 4 | 4 | (4, 4)-(4, 4)-(4, 4)-(4, 4) |
| 4 | 2 | 5 | (2, 5)-(2, 5)-(2, 5)-(2, 5)-(2, 5) |
| 5 | 3 | 5 | (3, 5)-(3, 5)-(3, 5)-(3, 5)-(3, 5) |
| 6 | 4 | 5 | (4, 5)-(4, 5)-(4, 5)-(4, 5)-(4, 5) |
| 7 | 5 | 5 | (5, 5)-(5, 5)-(5, 5)-(5, 5)-(5, 5) |
| 8 | 2 | 6 | (2, 6)-(2, 6)-(2, 6)-(2, 6)-(2, 6)-(2, 6) |
| 9 | 3 | 6 | (3, 6)-(3, 6)-(3, 6)-(3, 6)-(3, 6)-(3, 6) |
| 10 | 4 | 6 | (4, 6)-(4, 6)-(4, 6)-(4, 6)-(4, 6)-(4, 6) |
| 11 | 5 | 6 | (5, 6)-(5, 6)-(5, 6)-(5, 6)-(5, 6)-(5, 6) |
| 12 | 6 | 6 | (6, 6)-(6, 6)-(6, 6)-(6, 6)-(6, 6)-(6, 6) |
| 13 | 2 | 7 | (2, 7)-(2, 7)-(2, 7)-(2, 7)-(2, 7)-(2, 7)-(2, 7) |
| 14 | 3 | 7 | (3, 7)-(3, 7)-(3, 7)-(3, 7)-(3, 7)-(3, 7)-(3, 7) |
| 15 | 4 | 7 | (4, 7)-(4, 7)-(4, 7)-(4, 7)-(4, 7)-(4, 7)-(4, 7) |
| 16 | 5 | 7 | (5, 7)-(5, 7)-(5, 7)-(5, 7)-(5, 7)-(5, 7)-(5, 7) |
| 17 | 6 | 7 | (6, 7)-(6, 7)-(6, 7)-(6, 7)-(6, 7)-(6, 7)-(6, 7) |
| 18 | 7 | 7 | (7, 7)-(7, 7)-(7, 7)-(7, 7)-(7, 7)-(7, 7)-(7, 7) |
| 19 | 2 | 8 | (2, 8)-(2, 8)-(2, 8)-(2, 8)-(2, 8)-(2, 8)-(2, 8)-(2, 8) |
| 20 | 3 | 8 | (3, 8)-(3, 8)-(3, 8)-(3, 8)-(3, 8)-(3, 8)-(3, 8)-(3, 8) |
| 21 | 4 | 8 | (4, 8)-(4, 8)-(4, 8)-(4, 8)-(4, 8)-(4, 8)-(4, 8)-(4, 8) |
| 22 | 5 | 8 | (5, 8)-(5, 8)-(5, 8)-(5, 8)-(5, 8)-(5, 8)-(5, 8)-(5, 8) |
| 23 | 6 | 8 | (6, 8)-(6, 8)-(6, 8)-(6, 8)-(6, 8)-(6, 8)-(6, 8)-(6, 8) |
| 24 | 7 | 8 | (7, 8)-(7, 8)-(7, 8)-(7, 8)-(7, 8)-(7, 8)-(7, 8)-(7, 8) |
| 25 | 8 | 8 | (8, 8)-(8, 8)-(8, 8)-(8, 8)-(8, 8)-(8, 8)-(8, 8)-(8, 8) |

6.4 Разработка алгоритмов кодирования и декодирования в двухуровневой RRNS

В данном разделе представлен сравнительный анализ трех алгоритмов кодирования/декодирования: Mignotte, MRC8 и MRC16. Последние два алгорит-

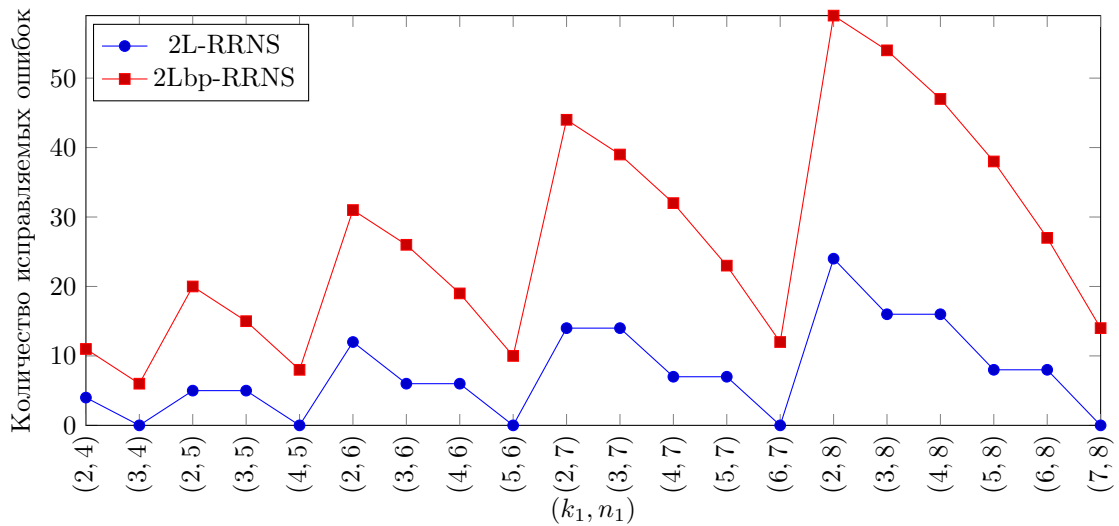


Рисунок 6.4 — Исправление ошибок в 2L-RRNS и 2Lbp-RRNS

Таблица 34 — Средняя скорость кодирования и декодирования данных (Мб/с)

| | Размер данных | Кодирование | | | Декодирование | | |
|----------|---------------|-------------|---------|-------|---------------|---------|-------|
| | | Min | Среднее | Max | Min | Среднее | Max |
| Mignotte | 1 Мб | 0.505 | 0.975 | 1.379 | 0.529 | 0.813 | 0.783 |
| | 10 Мб | 0.507 | 0.995 | 1.399 | 0.647 | 0.923 | 1.137 |
| | 100 Мб | 0.510 | 1.015 | 1.419 | 0.764 | 1.047 | 1.491 |
| MRC8 | 1 Мб | 0.784 | 2.174 | 3.145 | 5.921 | 7.371 | 9.314 |
| | 10 Мб | 0.786 | 2.194 | 3.166 | 6.038 | 7.926 | 9.667 |
| | 100 Мб | 0.788 | 2.221 | 3.186 | 6.156 | 8.483 | 10.02 |
| MRC16 | 1 Мб | 2.405 | 4.983 | 6.645 | 12.50 | 13.63 | 15.60 |
| | 10 Мб | 2.408 | 5.003 | 6.665 | 12.62 | 14.19 | 15.96 |
| | 100 Мб | 2.410 | 5.023 | 6.685 | 12.74 | 14.74 | 16.30 |

ма основаны на переходе к представлению в обобщенной позиционной (смешанной) системе счисления (Mixed Radix Conversion – MRC). Все три алгоритма кодирования на 1-ом уровне преобразуют S в n_1 долей, затем каждую из долей S_i преобразуют в $n_{2,i}$ долей согласно комбинации настроек уровня 2.

При кодировании алгоритмом Mignotte используется модификация операции $S_i = |S|_{p_i}$ для прямого преобразования входных данных. При кодировании алгоритмами MRC8 и MRC16 остатки от деления входных данных вычисляются с использованием FRNN.

Для восстановления S , применяется алгоритм декодирования 2Lbp-RRNS, в котором вариативно используются либо функции декодирования Mignotte, ос-

Алгоритм 7: Кодирование 2L-RRNS

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

S – входные данные,

$\hat{p} = (p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}}),$

$W_{1,i} = (w_{1,i,0}, \dots, w_{1,i,l}), W_{2,i,j} = (w_{2,i,j,0}, \dots, w_{2,i,j,l})$ – синаптические веса FRNN,

$algo$ – *Mignotte, MRC8, MRC16*

Output: \tilde{S}

1 **if** $algo == Mignotte$ **then**

2 $\tilde{S} = \text{Mignotte}(settings, S, \hat{p}, encoding);$

3 **if** $algo == MRC8$ **then**

4 $\tilde{S} = \text{MRC8}(settings, S, \hat{p}, W_{1,i}, W_{2,i,j}, encoding);$

5 **if** $algo == MRC16$ **then**

6 $\tilde{S} = \text{MRC16}(settings, S, \hat{p}, W_{1,i}, W_{2,i,j}, encoding)$

Result: \tilde{S}

нованные на классической CRT; либо декодирование с помощью MRC [29] на основе FRNN [373]. Все перечисленные алгоритмы используют доли $k_{2,i}$ уровня 2 для получения S_i долей уровня 1 при различных комбинациях настроек уровня 2. Затем выбираются k_1 долей S_i и, наконец, восстанавливается S .

6.4.1 Алгоритмы кодирования 2Lbp-RRNS

Схема Mignotte [281] — классическая структура доступа с механизмом проецирования на основе CRT для обнаружения и исправления ошибок. Согласно схеме Mignotte, целое число S представляется в виде кортежа (S_1, \dots, S_{n_1}) , где $S_i = |S|_{p_{1,i}}$, $P_i = \frac{P}{p_{1,i}}$. Восстанавливается S с помощью классической CRT, в рамках используемого корректирующего алгоритма

$$S = \left| \sum_{i=1}^{k_1} S_i P_i |P_i^{-1}|_{p_{1,i}} \right|_P. \quad (6.21)$$

MRC8 и MRC16 [29, 243] основаны на использовании взвешенной системы под названием обобщенная (смешанная) позиционная система счисления (Mixed-Radix

Алгоритм 8: Декодирование 2L-RRNS

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

\tilde{S} – представление S в 2L-RRNS,

$\hat{p} = (p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}}),$

I_D – структура доступа,

$\hat{W}_{1,i} = (\hat{w}_{1,i,0}, \dots, \hat{w}_{1,i,l}), \hat{W}_{2,i,j} = (\hat{w}_{2,i,j,0}, \dots, \hat{w}_{2,i,j,l})$ – синаптические веса DNN,

$algo$ – *Mignotte, MRC8, MRC16*

Output: S

1 **if** $algo == Mignotte$ **then**

2 $S = Mignotte(settings, I_D, \tilde{S}, \hat{p}, decoding);$

3 **if** $algo == MRC8$ **then**

4 $S = MRC8(settings, I_D, \tilde{S}, \hat{p}, \hat{W}_1, \hat{W}_{2,i}, decoding);$

5 **if** $algo == MRC16$ **then**

6 $S = MRC16(settings, I_D, \tilde{S}, \hat{p}, \hat{W}_1, \hat{W}_{2,i}, decoding)$

Result: S

Systems – MRS) – нестандартной позиционной системы счисления, в которой основание системы счисления меняется от позиции к позиции. Базовые реализации данных алгоритмов, использующие архитектуру нейронной сети под названием нейронная сеть конечного кольца (FRNN), представлены в [373].

На рисунке 6.5 показана обобщенная интерпретация этой архитектуры – параллельная взаимосвязанная сеть простых элементов, состоящая из двух компонентов:

1. Нейронной сети повторяющихся элементов, способных выполнять основные арифметические операции.
2. Весов, представляющих знания системы.

Арифметические операции в классах вычетов, такие как сложение, умножение на константу и их комбинации, имеют схожую реализацию, сводящуюся к этой архитектуре. Простая архитектура FRNN основана на методе Паскаля с использованием метода окон [262] для нахождения остатка от деления.

Исходные данные S представлены как $S = s_l | s_{l-1} | \dots | s_0$, где $|$ – конкатенация L -битовых строк s_i . L определяет размер окна $L \in \{8, 16\}$, $w_{i,j} = |2^{L \cdot j}|_{p_{1,i}}$ – синаптические веса FRNN, $j = \overline{0, l}$.

Алгоритм 9: Кодирование Mignotte

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

S – входные данные,

$(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}})$

Output: $S_{i,j}$ – проекции входных данных S

```

1 for  $i = 1, i \leq n_1, i++$  do
2    $S_i = |S|_{p_{1,i}}$ ;
3 for  $i = 1, i \leq n_1, i++$  do
4   for  $j = 1, j \leq n_{2,i}, j++$  do
5      $S_{i,j} = |S_i|_{p_{2,i,j}}$ ;

```

Result: \tilde{S}

Алгоритм 10: Декодирование Mignotte

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

\tilde{S} – представление S в 2L-RRNS,

$(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}}),$

I_D – структура доступа

Output: S

```

1  $S_{list} = []; p_{list} = [];$  // вспомогательные списки
2 for  $i = 1, i \leq k_1, i++$  do
3    $j = I_D[i];$ 
4    $S_i = \text{CRTtoBin}((S_{j,1}, \dots, S_{j,n_{2,j}}), (p_{2,j,1}, \dots, p_{2,j,n_{2,j}}));$ 
5    $S_{list}.append(S_i);$ 
6    $p_{list}.append(p_{1,j});$ 
7  $S = \text{CRTtoBin}(S_{list}, p_{list})$ 

```

Result: S

FRNN состоит из двух слоев: первый – предвычисленный слой, на котором произведение s_i на синаптический вес $w_{i,j}$ вычисляется по значению s_i , являющемуся адресом заранее сохраненного в LUT-таблице соответствующего произведения по модулю $p_{1,i}$; на втором вычислительном слое полученные произведения суммируются по модулю $p_{1,i}$. Таким образом, FRNN описывается

Алгоритм 11: Кодирование MRC

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

S – входные данные,

$(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}}),$

$W_{1,i} = (w_{1,i,0}, \dots, w_{1,i,l}), W_{2,i,j} = (w_{2,i,j,0}, \dots, w_{2,i,j,l})$ – синаптические веса
FRNN

Output: $S_{i,j}$

```

1 for  $i = 1, i \leq n_1, i++$  do
2    $S_i = \text{FRNN}(S, p_{1,i}, W_{1,i});$ 
3 for  $i = 1, i \leq n_1, i++$  do
4   for  $j = 1, j \leq n_{2,i}, j++$  do
5      $S_{i,j} = \text{FRNN}(S_i, p_{2,i,j}, W_{2,i,j})$ 

```

Result: $S_{i,j}$

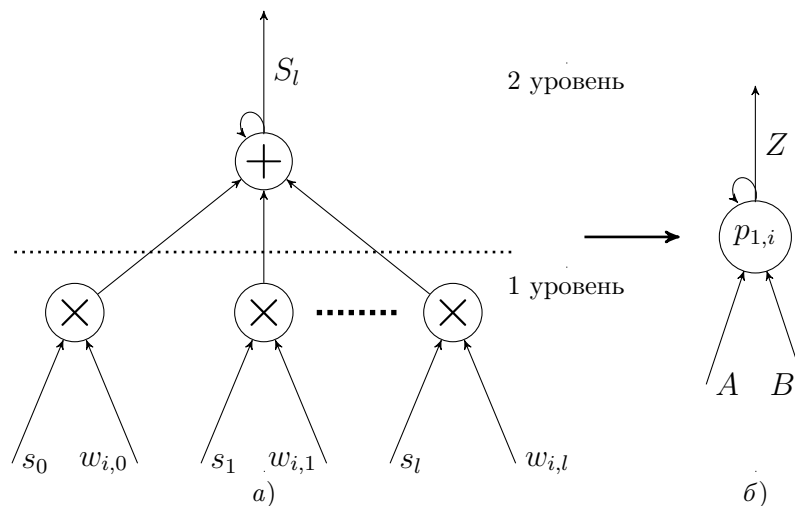


Рисунок 6.5 — Архитектура FRNN *a)* и ее символическое отображение *б)*

следующей формулой

$$S_i = \left| \sum_{j=0}^l s_j \cdot w_{i,j} \right|_{p_{1,i}} . \quad (6.22)$$

Алгоритм 12: Декодирование MRC

Input: $settings = (k_1, n_1), (k_{2,1}, n_{2,1}), \dots, (k_{2,n_1}, n_{2,n_1}),$

\tilde{S} – представление S в 2L-RRNS,

$(p_{1,1}, \dots, p_{1,n_1}), (p_{2,1,1}, \dots, p_{2,1,n_{2,1}}), \dots, (p_{2,n_1,1}, \dots, p_{2,n_1,n_{2,n_1}}),$

I_D – структура доступа,

$\hat{W}_1 = (\hat{w}_{1,1}, \dots, \hat{w}_{1,n_1}), \hat{W}_{2,i} = (\hat{w}_{2,i,1}, \dots, \hat{w}_{2,i,n_{2,i}})$ – синаптические веса
DNN

Output: S

```

1  $S_{list} = []$ ;  $p_{list} = []$ ;  $\hat{W}_{list} = []$ ; //вспомогательные списки
2 for  $i = 1, i \leq k_1, i ++$  do
3    $j = I_D[i]$ ;
4    $S_i = \text{DNN} \left( (S_{j,1}, \dots, S_{j,n_{2,j}}), (p_{2,j,1}, \dots, p_{2,j,n_{2,j}}), \hat{W}_{2,j} \right)$ ;
5    $S_{list}.append(S_i)$ ;
6    $p_{list}.append(p_{1,j})$ ;
7    $\hat{W}_{list}.append(\hat{w}_{1,j})$ ;
8  $S = \text{DNN} \left( S_{list}, p_{list}, \hat{W}_{list} \right)$ 

```

Result: S

6.4.2 Алгоритмы декодирования 2Lbp-RRNS

Для декодирования чисел из RNS в двоичное представление используются различные алгоритмы: CRT [351], метод Wang [364], MRS [243], диагональная функция [283], функция ядра [128] и приближенный метод [12]. В работе [283] было показано, что диагональную функцию и функцию ядра не рекомендуется использовать для подобных преобразований.

Классическая CRT содержит вычислительно сложную арифметическую операцию нахождения остатка от деления на диапазон RNS, поэтому не является эффективной для обратного преобразования чисел.

Приближенный метод позволяет снизить вычислительную сложность декодирования по сравнению с CRT. Он основан на замене абсолютных величин относительными и замене операции деления с остатком общей формы на бинарный сдвиг вправо. Однако, для получения правильного значения с помощью приближенного метода необходимо увеличить размер (точность) коэффициен-

тов с $\lceil \log_2 P \rceil$ до $\lceil \log_2 P \cdot \rho \rceil$, где $\rho = -n_1 + \sum_{i=1}^{n_1} p_{1,i}$, что исключает результирующий прирост эффективности декодирования.

Рекурсивный метод удвоения Wang так же предназначен для уменьшения вычислительной сложности. Согласно данному методу размер делителя уменьшается с P до \sqrt{P} . Но при этом количество остатков от деления увеличивается с единицы до $\lceil \log_2 n_1 \rceil$.

Альтернативным решением для декодирования чисел из RNS в двоичную систему счисления являются алгоритмы на основе MRS. Декодирование состоит из двух этапов: на первом этапе число конвертируется из RNS в MRS, на втором – из MRS в двоичное представление.

Для уменьшения вычислительной сложности первого этапа, предложена модификация преобразования из RNS в двоичную систему счисления с использованием CRT и нейронной сети конечного кольца (FRNN). Второй этап перехода от MRS к бинарному представлению осуществляется с помощью сверточной нейронной сети (Convolutional Neural Network – CNN) (рис. 6.6). Остатки RRNS

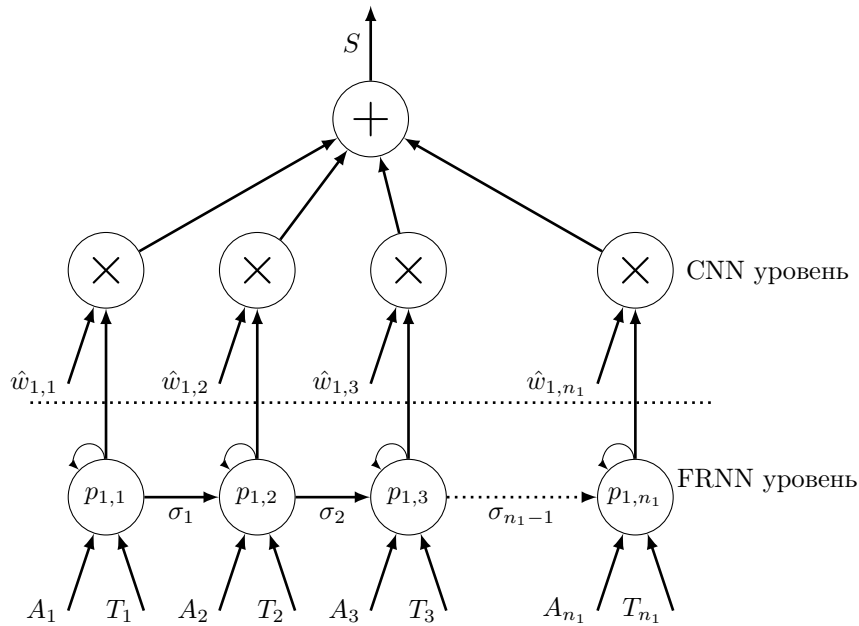


Рисунок 6.6 — Архитектура DNN для декодирования из 1L-RRNS в двоичную систему счисления

преобразуются в цифры MRS, а затем в двоичную систему счисления по следующей формуле

$$S = \sum_{i=1}^{n_1} \hat{s}_i \cdot \hat{w}_{1,i}, \quad (6.23)$$

где $\hat{w}_{1,i}$ – синаптические веса CNN, $\hat{w}_{1,i} = \prod_{j=1}^{i-1} p_{1,j}$, \hat{s}_i – цифры MRS ($S \xrightarrow{MRS} = [\hat{s}_1, \dots, \hat{s}_{n_1}]$), $0 \leq \hat{s}_i < p_{1,i}$.

Эквивалентная формула для восстановления S , применяемая при использовании классического MRC, формулируется следующим образом

$$S = \hat{s}_1 + \hat{s}_2 \cdot p_{1,1} + \hat{s}_3 \cdot p_{1,1} \cdot p_{1,2} + \dots + \hat{s}_n \cdot p_{1,1} \cdot p_{1,2} \cdot \dots \cdot p_{1,n_1-1}. \quad (6.24)$$

Цифры MRS \hat{s}_i вычисляются с использованием следующих формул

$$\begin{aligned} \hat{s}_1 &= S_1 \\ \hat{s}_2 &= \left| (S_2 - \hat{s}_1) \left| p_{1,1}^{-1} \right|_{p_{1,2}} \right|_{p_{1,2}} \\ \hat{s}_3 &= \left| \left((S_3 - \hat{s}_1) \left| p_{1,1}^{-1} \right|_{p_{1,3}} - \hat{s}_2 \right) \left| p_{1,2}^{-1} \right|_{p_{1,3}} \right|_{p_{1,3}} \\ \dots & \\ \hat{s}_n &= \left| \left(\dots \left((S_n - \hat{s}_1) \left| p_{1,1}^{-1} \right|_{p_{1,n_1}} - \hat{s}_2 \right) \left| p_{1,2}^{-1} \right|_{p_{1,n_1}} - \dots - \hat{s}_{n_1-1} \right) \left| p_{1,n_1-1}^{-1} \right|_{p_{1,n_1}} \right|_{p_{1,n_1}}. \end{aligned}$$

Положительное число в интервале $[0, P - 1]$ может быть однозначно представлено цифрами MRS. Использование MRS снижает вычислительную сложность преобразования из 1L-RRNS в двоичную систему счисления за счет замены операции нахождения остатка от деления на P вычислением \hat{s}_i . Вычислительная сложность расчета \hat{s}_i квадратична по n_1 . Одновременное использование идей MRS и CRT для декодирования из 1L-RRNS в двоичную систему счисления позволяет ускорить алгоритм.

Пусть $B_i = \left| P_i^{-1} \right|_{p_{1,i}} \cdot P_i$ – ортогональные базисы 1L-RRNS, где для всех $i = \overline{1, n_1}$: $P_i = \frac{P}{p_{1,i}}$. Для любого $i = \overline{1, n_1}$: B_i представляется в MRS как $B_i \xrightarrow{MRS} = [\hat{b}_{i,1}, \dots, \hat{b}_{i,n_1}]$, при этом в вычислениях используются $T_i = (\hat{b}_{1,i}, \dots, \hat{b}_{i,i})$ – укороченные кортежи представлений коэффициентов в MRS, $A_i = (S_i, S_i, \dots, S_i)$ и $\sigma_i = \left\lfloor \frac{1}{p_{1,i}} \cdot \sum_{j=1}^i S_j \cdot \hat{b}_{j,i} \right\rfloor$ – смещение.

На рисунке 6.6 показана двухуровневая архитектура декодирующей нейронной сети (Decoding Neural Network – DNN) для преобразования из 1L-RRNS в двоичную систему счисления, включающая уровни FRNN и CNN.

6.5 Анализ производительности двухуровневой RRNS

В данном разделе представлена оценка производительности 2Lbp-RRNS с точки зрения скорости кодирования/декодирования с использованием трех алгоритмов: Mignotte, MRC8 и MRC16, а также скорости загрузки и выгрузки в реальные облачные хранилища.

Разработанная программная платформа основана на JMetal 5.6 и JDK 11.0.1 (64-бит). Аппаратная платформа включает Dell Precision T3610, Intel Xeon CPU E5-1606 @ 2, 80 ГГц, 16 ГБ оперативной памяти DDR3 с 64-разрядной версией Windows 10 Enterprise.

Экспериментальный сценарий включает семь облачных хранилищ: DropBox, OneDrive, Box, Salesforce, GoogleDrive, Sharefile и Egnyte. Чтобы получить доступ к общедоступному REST API CSP, использовались оболочка Java для Google Drive, Dropbox, Box и Sharefile, а так же библиотека Apache HttpClient для OneDrive, Egnyte и Salesforce [242].

В таблице 35 показаны минимальная, максимальная и средняя скорости доступа к семи CSP. Отметим, что в большинстве случаев скорость доступа меньше, чем скорости кодирования/декодирования (табл. 34). Для MRC16 сред-

Таблица 35 — Скорость доступа к семи облачным сервисам (Мб/с)

| Облачное хранилище | Скорость загрузки | | | Скорость выгрузки | | |
|--------------------|-------------------|------|---------|-------------------|------|---------|
| | Min | Max | Средняя | Min | Max | Средняя |
| GoogleDrive | 1.79 | 3.24 | 2.98 | 2.15 | 3.26 | 3.06 |
| OneDrive | 0.91 | 1.70 | 1.46 | 1.21 | 2.41 | 2.18 |
| Dropbox | 2.59 | 3.05 | 2.93 | 3.07 | 3.32 | 3.25 |
| Box | 1.91 | 3.26 | 2.55 | 2.01 | 3.20 | 2.62 |
| Egnyte | 1.24 | 1.93 | 1.70 | 2.17 | 2.36 | 2.30 |
| Sharefile | 0.11 | 0.65 | 0.51 | 0.72 | 0.76 | 0.75 |
| Salesforce | 0.52 | 0.73 | 0.64 | 0.68 | 0.72 | 0.71 |

няя скорость загрузки в $\frac{5.023}{2.98} = 1.67$ раза меньше средней скорости кодирования, а средняя скорость выгрузки в $\frac{14.74}{3.25} = 4.53$ раза меньше скорости декодирования (табл. 34).

6.5.1 Скорость кодирования и декодирования в двухуровневой RRNS

Чтобы всесторонне протестировать предлагаемую систему, исследуем скорость кодирования/декодирования в/из 2Lbp-RRNS при различных настройках (k_1, n_1) уровня 1 и различных настройках $(k_{2,i}, n_{2,i})$ уровня 2, где $1 \leq i \leq n_1$.

На рисунках 6.7 и 6.8 представлены графики изменения скоростей кодирования/декодирования, соответственно. На первом уровне структура доступа ограничена схемой $(k_1, n_1) = (3, 4)$. На втором уровне рассмотрены 27 вариантов $(k_{2,i}, n_{2,i})$, от $n_{2,i} = 3$ до $n_{2,i} = 8$. Поскольку размер доли уменьшается при

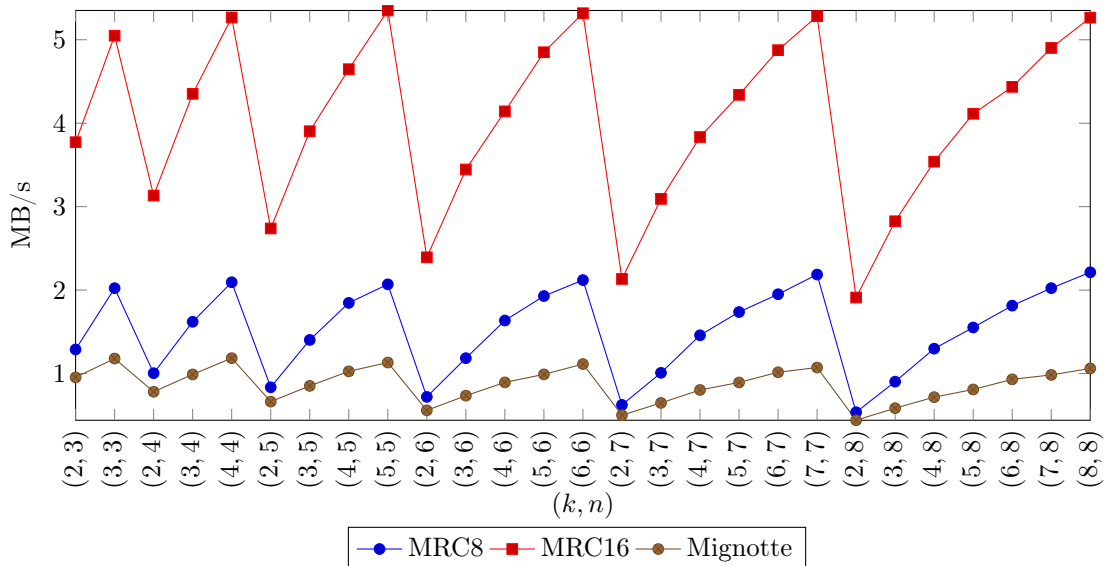


Рисунок 6.7 — Скорость кодирования для настроек $(3, 4)$ на уровне 1

увеличении k_2 , самые высокие скорости кодирования достигаются при $k_2 = n_2$ (рис. 6.7).

С другой стороны, самые низкие скорости декодирования наблюдаются так же при $k_2 = n_2$ (рис. 6.8). Это объясняется необходимостью декодирования всех n_2 долей для восстановления исходных данных.

Результаты экспериментов показывают, что скорость кодирования находится в диапазоне 0.505 – 6.685 МБ/с, скорость декодирования находится в диапазоне 0.529 – 16.3 МБ/с. Скорость доступа к облачным сервисам находится в пределах 0.11 – 3.32 МБ/с.

Алгоритм Mignotte – самый медленный из рассмотренных алгоритмов со скоростью кодирования не выше 1.23 МБ/с и декодирования не выше 1.3 МБ/с.

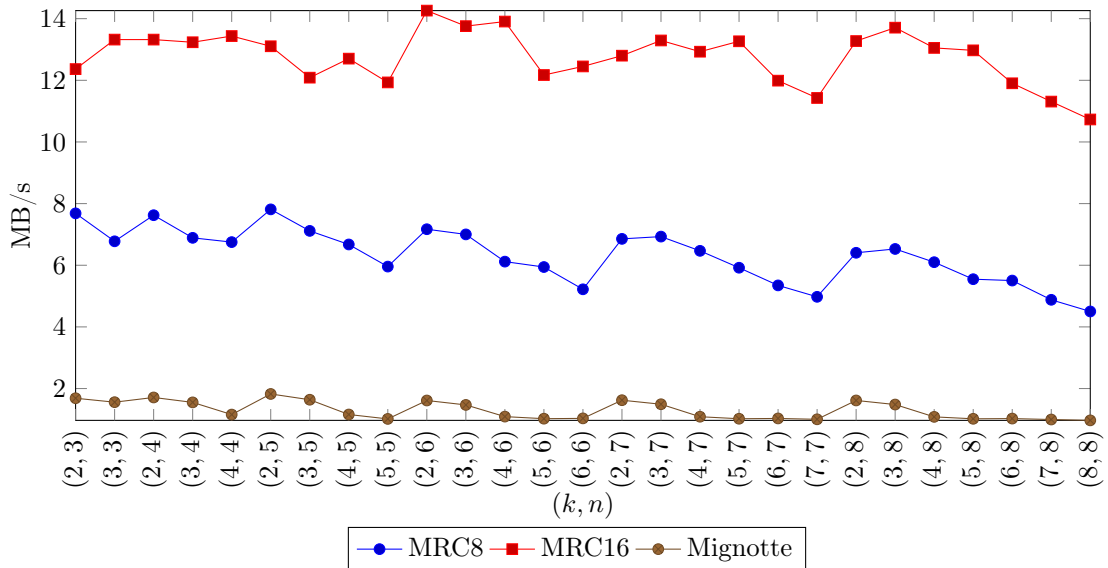


Рисунок 6.8 — Скорость декодирования для настроек (3, 4) на уровне 1

MRC8 имеет максимальную скорость кодирования 3 МБ/с при использовании настроек уровня 2, для которых $k_2 = n_2$, и максимальную скорость декодирования 10.02 МБ/с при использовании настроек (2, 5) уровня 2.

Из полученных графиков можно сделать вывод, что MRC16 превосходит два других алгоритма во всех экспериментах. Например, MRC16 обеспечивает максимальную скорость кодирования 6.68 МБ/с при использовании настроек (3, 4) на уровне 1 и (6, 6) на уровне 2 (рис. 6.7).

MRC16 в среднем в 2.53 раза быстрее MRC8 и в 4.83 раза быстрее чем алгоритм Mignotte на этапе кодирования. На этапе декодирования MRC16 в среднем в 1.78 раза быстрее чем MRC8 и в 11.43 раза быстрее, чем алгоритм Mignotte.

По умолчанию все экспериментальные данные были получены для параллельной реализации 2Lbp-RRNS, т.к. именно возможность параллельного выполнения операций по различным модулям и возможность распараллеливания самих операций делает RRNS столь эффективным инструментом для специализированных приложений, в частности, реализующих распределенное хранение и обработку данных. Дополним проведенное исследование сравнением производительности при последовательной и параллельной реализации 2Lbp-RRNS. На рисунках 6.9 и 6.10 показаны диаграммы изменения скоростей кодирования/декодирования соответственно, для обеих версий. Коробчатая диаграмма удобно отображает медианный, нижний и верхний квартили, минимальные и максимальные значения выборки и выбросы.

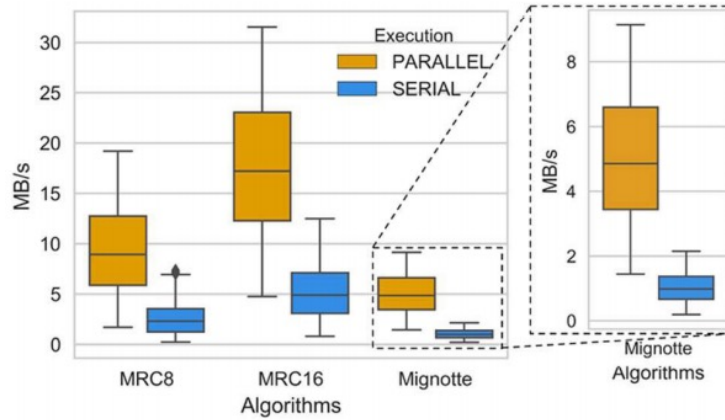


Рисунок 6.9 — Диаграмма изменения скорости кодирования для различных комбинаций настроек

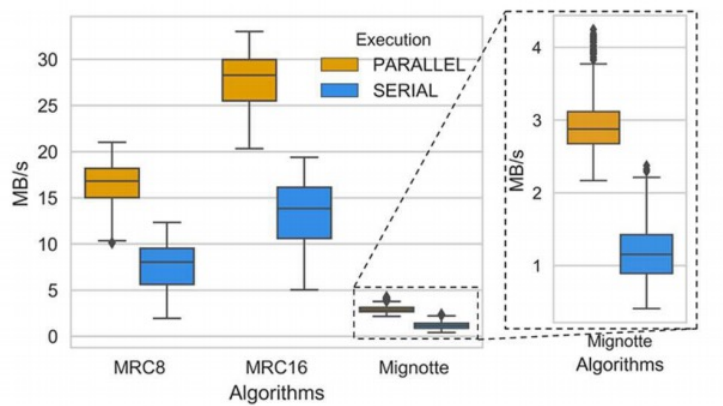


Рисунок 6.10 — Диаграмма изменения скорости декодирования для различных комбинаций настроек

С целью получения статистически значимых результатов было выполнено по 30 экспериментов для каждой настройки уровня 1, от $n_1 = 3$ до $n_1 = 8$, при всевозможных комбинациях настроек уровня 2.

Алгоритм MRC8 имеет выбросы, которые указывают на то, что существует большой разрыв между минимальными и максимальными скоростями кодирования и декодирования.

Алгоритм MRC16 более сбалансирован и превосходит по скорости алгоритмы MRC8 и Mignotte, достигая максимальной скорости декодирования более 30 МБ/с при параллельной реализации (рис. 6.10).

Так же отметим, что MRC16 не имеет выбросов, и можно сделать вывод, что скоростей, выделяющихся из общей выборки, нет.

В целом MRC16 показывает лучшую производительность для всех комбинаций настроек 2Lbp-RRNS.

6.5.2 Скорость обработки данных в двухуровневой RRNS

Поскольку скорости кодирования и декодирования сравнимы со скоростями загрузки и выгрузки, следует рассмотреть полный цикл обработки данных: кодирование плюс загрузка и выгрузка плюс декодирование. Далее под скоростью загрузки и выгрузки будут подразумеваться скорости выполнения указанных пар операций.

Скорость загрузки $V_u = \frac{\text{size}(D)}{T_E + t_{up}}$ зависит от времени кодирования из двоичного кода в RRNS и времени загрузки. Скорость выгрузки $V_d = \frac{\text{size}(D)}{T_D + t_{dow}}$ зависит от времени выгрузки (скачивания) из облачного хранилища и декодирования из RRNS в двоичный код.

Рассмотренные в работе методы и алгоритмы были разработаны с учетом особенностей мультиоблачной среды. Мультиоблачная среда имеет динамический характер, параметры меняются с течением времени, и их изменение трудно спрогнозировать и предвидеть заранее. Эти нестационарности являются одной из основных проблем при разработке эффективных алгоритмов, способных смягчить или полностью устранить их последствия.

Для оценки практической применимости предложенной схемы и изучения ее свойств рассмотрим наилучший и наихудший сценарии. При наилучшем сценарии для хранения данных выбираются облака с наилучшими скоростями доступа, при наихудшем – выбираются самые медленные облака.

Следующие формулы позволяют вычислить время загрузки

$$t_{up} = \sum_{i=1}^{n_1} \sum_{j=1}^{n_{2,i}} \frac{\text{size}(S_{i,j})}{\text{up}(j)} \quad (6.25)$$

и время выгрузки

$$t_{dow} = \sum_{i=1}^{n_1} \sum_{j=1}^{n_{2,i}} \frac{\text{size}(S_{i,j})}{\text{down}(j)}, \quad (6.26)$$

где $\text{up}(j)$ и $\text{down}(j)$ – скорости загрузки и выгрузки в/из j -го облака.

Для примера рассмотрим настройку уровня 1 (3, 4) с настройкой уровня 2 (5, 5) при скоростях доступа из таблицы 35.

На рисунке 6.11 показано, что MRC16 имеет скорость загрузки $V_u = 0.837$ МБ/с в лучшем случае и $V_u = 0.406$ МБ/с в худшем случае. Алгоритм Mignotte имеет скорость загрузки $V_u = 0.257$ МБ/с в лучшем случае

и $V_u = 0.13$ МБ/с в худшем случае. Отметим, что при тех же настройках

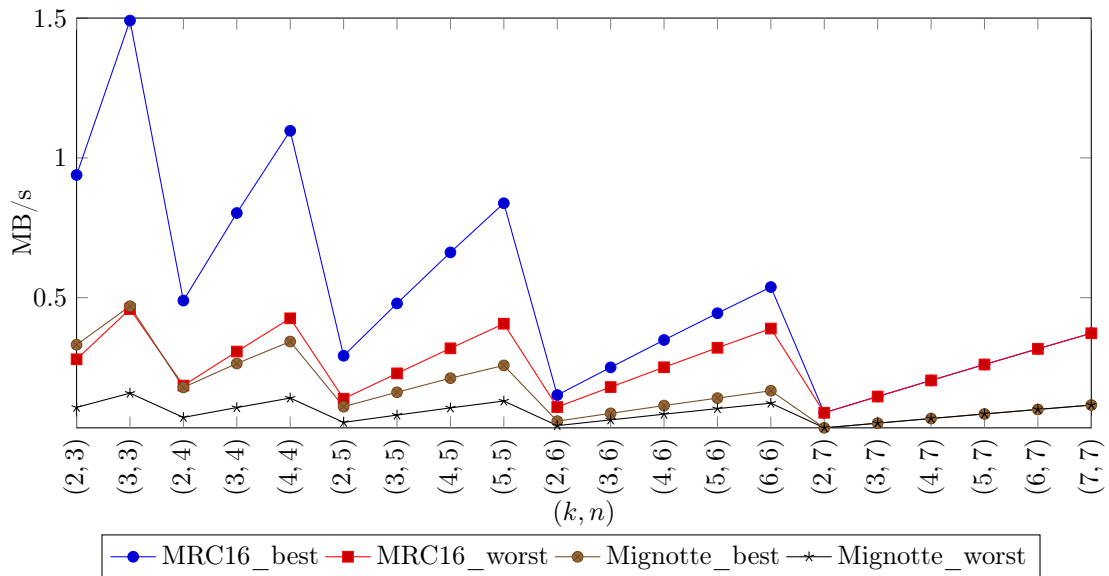


Рисунок 6.11 — Скорость загрузки при настройках (3, 4) уровня 1

скорость кодирования MRC16 составляет 6.583 МБ/с (рис. 6.7), а скорость кодирования алгоритмом Mignotte составляет 1.317 МБ/с.

На рисунке 6.12 показано, что MRC16 имеет скорость выгрузки $V_d = 1.093$ МБ/с в лучшем случае и $V_d = 0.54$ МБ/с в худшем случае. Алгоритм Mignotte имеет скорость $V_d = 0.292$ МБ/с в лучшем случае и $V_d = 0.16$ МБ/с в худшем случае.

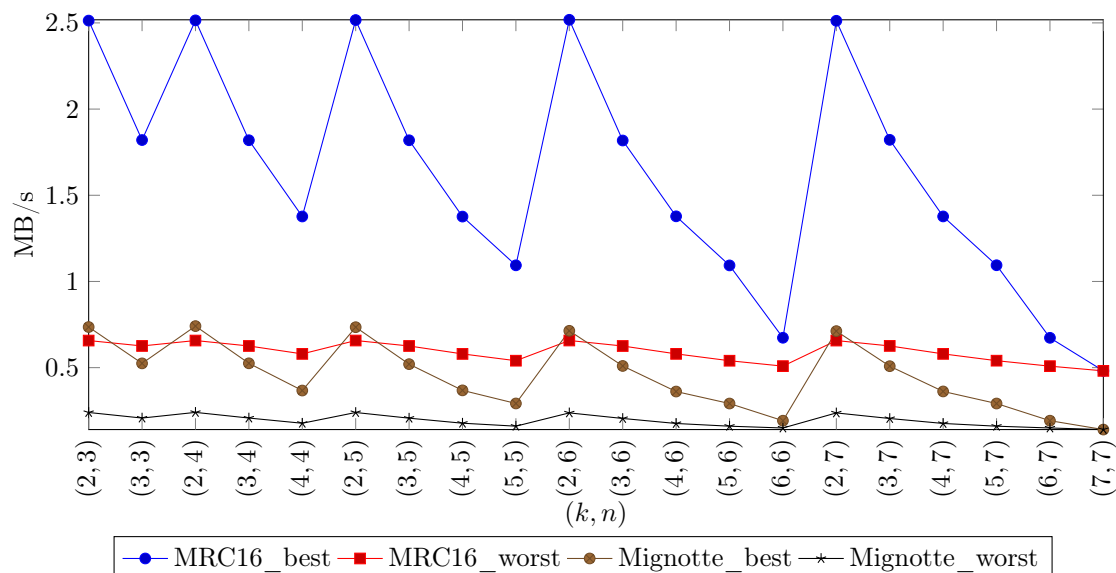


Рисунок 6.12 — Скорость выгрузки при настройках (3, 4) уровня 1

6.6 Выводы по шестой главе

В шестой главе представлена конфигурируемая масштабируемая двухуровневая структура доступа на основе RRNS (2Lbp-RRNS), разработанная для надежного и безопасного хранения данных в мультиоблачных системах, допускающая реализацию гомоморфных вычислений и позволяющая осуществлять параллельную обработку данных с сохранением их конфиденциальности. Разработанная схема 2Lbp-RRNS является расширением классической схемы 2L-RRNS. Высокая эффективность и производительность 2Lbp-RRNS достигается за счет использования расстояния Хэмминга.

Получена верхняя граница для количества обнаруживаемых и исправляемых ошибок при использовании традиционных пороговых двухуровневых схем 2L-RRNS и предложенных пороговых двухуровневых схем 2Lbp-RRNS. Показано, что предложенная схема 2Lbp-RRNS обладает лучшими корректирующими свойствами по сравнению с традиционной схемой 2L-RRNS, позволяет обнаруживать в среднем в 1.58 раза и исправлять в среднем в 3.37 раза больше ошибок.

Предложены эффективные реализации алгоритмов кодирования и декодирования данных в 2Lbp-RRNS: эффективность при кодировании достигается за счет использования метода Паскаля и нейронной сети конечного кольца (FRNN), эффективность декодирования обусловлена использованием перехода к представлению в обобщенной позиционной системе счисления (MRC), FRNN и сверточной нейронной сети (CNN).

Отметим, что при кодировании/декодировании данных в 2Lbp-RRNS, для реализации обратного преобразования вариативно может быть использован один из алгоритмов: Mignotte (основанный на Китайской теореме об остатках), MRC8 или MRC16 (основанные на переходе к обобщенной позиционной системе счисления и отличающиеся лишь размером окна, 8 или 16 бит, при реализации FRNN). Для определения наиболее оптимального из перечисленных алгоритмов, выполнен сравнительный анализ производительности схем 2Lbp-RRNS, использующих указанные алгоритмы, учитывающий полный цикл хранения данных: кодирование-загрузка и выгрузка-декодирование для различных параметров облачных хранилищ. Результаты анализа показали, что производительность MRC16 при кодировании-загрузке колеблется в диапазоне

0.406-0.837 МБ/с, а при выгрузке-декодировании в диапазоне 0.54-1.093 МБ/с в зависимости от параметров хранилищ. Для сравнения, показатели наиболее близкого по производительности алгоритма Mignotte при кодировании-загрузке колеблются в диапазоне 0.13-0.257 МБ/с, а при выгрузке-декодировании в диапазоне 0.16-0.292 МБ/с. Таким образом, MRC16 является более сбалансированным и быстрым алгоритмом, превосходящим MRC8 и Mignotte. Также показано преимущество в производительности, достигаемое за счет параллельной реализации предложенной схемы. Все экспериментальные данные получены для восьми реальных облачных хранилищ.

ЗАКЛЮЧЕНИЕ

Работа посвящена исследованию методов и алгоритмов, необходимых для построения надежной и безопасной системы распределенного хранения и обработки конфиденциальных данных с использованием гомоморфных вычислений. Основным сдерживающим фактором для широкого практического использования гомоморфных вычислений является их высокая вычислительная сложность, складывающаяся из сложностей проблемных операций, таких как определение знака закодированного числа, сравнение закодированных чисел и контроль результатов обработки закодированных данных без их декодирования. Основные полученные и представленные в работе результаты исследования можно сформулировать следующим образом:

- Построена структурная модель обработки данных в распределенных средах, объединяющая в себе краевые, туманные и облачные вычисления. Выделено пять уровней передачи, хранения и обработки данных. Для каждого из уровней выявлены основные угрозы безопасности данных и проанализированы современные методы уменьшения вероятности кражи, потери или искажения данных. Установлено, что в распределенных средах в условиях повышенной неопределенности фундаментальные подходы к снижению рисков конфиденциальности, целостности и доступности, использующие механизмы репликации данных, резервного копирования, структуры доступа, избыточную систему остаточных классов, коды стирания, регенерационные коды недостаточно эффективны и должны быть усовершенствованы. Предложено использование вышеперечисленных механизмов, адаптированных, оптимизированных и интегрированных в концепцию мультиоблачного хранения и обработки данных. Показано, что использование мультиоблачного подхода позволяет существенно повысить надежность распределенных систем и снизить вероятности потери, утечки информации, отказа в доступе в течение длительного времени.
- Показано, что модель, наиболее адекватную мультиоблачному подходу с точки зрения организации распределенного хранения и обработки данных, реализуют пороговые структуры доступа. Приведено обоснование выбора алгоритмов реализации пороговой структуры доступа с

точки зрения обеспечения конфиденциальности, надежности хранения, возможности осуществления контроля корректности операций с данными и вводимой избыточности.

- Предложена адаптивная распределенная служба хранения WA-MRC-RRNS, которая сочетает в себе взвешенную пороговую структуру доступа, систему контроля корректности результатов обработки данных и допускает реализацию гомоморфных вычислений.

Использование взвешенной пороговой структуры доступа обусловлено доказанной теоремой о том, что вероятность потери данных при использовании взвешенной пороговой структуры доступа не превышает вероятности потери данных при использовании соответствующей классической пороговой структуры доступа. Показано, что в пессимистическом сценарии при настройке (3,4) вероятность потери данных при использовании WA-MRC-RRNS в 777.02 раза ниже, чем при использовании классической пороговой структуры доступа MRC-RRNS. В среднем же вероятность потери данных при использовании WA-MRC-RRNS ниже в $9.23 \cdot 10^{17}$ раза.

Выбор RRNS в качестве основы для предложенной взвешенной пороговой структуры доступа обусловлен возможностью построения вычислительно стойкой схемы и реализации механизмов обнаружения/восстановления множественных ошибок данных. Кроме того, RRNS позволяет динамически настраивать параметры, чтобы справиться с различными объективными предпочтениями, рабочими нагрузками и свойствами облака.

Высокая производительность предложенной схемы достигается за счет разработанных алгоритмов кодирования/декодирования, основанных на MRC, FRNN и их эффективной программной реализации. Сравнение предложенной схемы WA-MRC-RRNS с другой известной взвешенной схемой WA-AR-RRNS с точки зрения производительности дало следующие результаты: при кодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 13.73 раза, при декодировании WA-MRC-RRNS быстрее WA-AR-RRNS в 385.07 раза. Отметим, что предложенная схема WA-MRC-RRNS также превосходит классическую пороговую схему AR-RRNS с точки зрения производительности (в 4.83 раза при кодировании и в 120.04 раза при декодировании), проигрывая лишь классической по-

роговой схеме MRC-RRNS в 2.42 раза при кодировании и в 1.16 раза при декодировании. Данные потери в производительности абсолютно оправданны многократным повышением надежности и безопасности, достигаемым за счет использования взвешенной схемы WA-MRC-RRNS вместо классической пороговой схемы MRC-RRNS.

Для анализа предложенной схемы WA-MRC-RRNS с точки зрения безопасности данных доказано утверждение, дающее оценку вероятности получения несанкционированного доступа к данным. Приведены вероятности получения несанкционированного доступа к данным для каждого из трех основных сценариев сговора: когда противоборствующая коалиция знает секретный ключ и не знает необходимое количество долей; не знает ни секретного ключа, ни необходимого количества долей; не знает секретного ключа и знает необходимое количество долей. Для обеспечения безопасности данных предложено интегрировать WA-MRC-RRNS в разработанную конфигурируемую схему хранения данных AC-RRNS. Доказана вычислительная безопасность AC-RRNS. Сравнительный анализ предложенной схемы с известными структурами доступа, использующими аппарат RRNS, такими как схема HORNS, основанная на схеме Mignotte, и схема Asmuth-Bloom дал следующие результаты: HORNS обладает меньшей избыточностью, но в отличие от предложенной схемы не является вычислительно безопасной, а также уязвима для атаки открытым текстом и не может быть использована для решения проблемы сговора; схема Asmuth-Bloom является асимптотически идеальной, подходит для обеспечения безопасности данных при сговоре, но вводит избыточность, в k раз превышающую избыточность предложенной схемы (k – параметр схемы Asmuth-Bloom).

- Установлено, что результаты операций определения знака числа и сравнения чисел, заданных над кольцом вычетов с делителями нуля, невозможно вычислить с помощью многочленов.
- Разработаны два алгоритма, реализующие функции определения знака числа для гомоморфных вычислений над кольцом вычетов с делителями нуля, основанных на RNS с четным и нечетным диапазоном.
- Предложен метод гомоморфного сравнения чисел над кольцом вычетов с делителями нуля, использующий введенное понятие модифицированной диагональной функции (MDF). MDF представляет собой строго

- возрастающую позиционную характеристику чисел, представленных в RNS, сочетающую преимущества диагональной функции и приближенного метода. Строгая монотонность MDF обеспечивает взаимоднозначное соответствие числа и его позиционной характеристики, поэтому не возникает ситуаций, когда требуется выполнение дополнительных действий для сравнения чисел. Кроме того, вместо операции нахождения остатка от деления на большое число при вычислении MDF используются значительно более простые в реализации вычисления по модулю, равному степени числа 2.
- Разработано устройство сравнения на основе MDF, которое, вместе с его наиболее эффективными известными аналогами, применяемыми для сравнения чисел в RNS с модулями общего вида, было синтезировано для технологии 65 нм с использованием нескольких образцов наборов модулей. Согласно полученным оценкам производительности, предложенный подход обеспечивает снижение задержки на 11 – 75% (в зависимости от набора модулей) по сравнению с самыми быстрыми существующими реализациями известных методов сравнения чисел в RNS. Более того, наблюдается снижение аппаратных затрат (более чем на 41%) и значительное снижение энергопотребления, которое в ряде случаев превышает 100%. Таким образом, предложенный метод на основе MDF позволяет реализовывать наиболее эффективные на сегодняшний день устройства сравнения чисел, представленных в RNS с наборами модулей общего вида.
 - Исследованы свойства функции ядра Акушского. Доказано, что для достижения монотонности при построении функции ядра необходимо использовать только неотрицательные коэффициенты. Показано, что уже известная диагональная функция, ранее предложенная для реализации сравнения чисел в RNS, есть не что иное, как частный случай функции ядра со всеми коэффициентами равными единице. Сформулированы условия, при которых обеспечивается минимальный диапазон функции ядра (необходимый для получения наилучших характеристик устройства сравнения чисел в RNS). Установлено, что монотонная функция ядра минимального диапазона (ММСФ) имеет только один коэффициент, равный единице (соответствующий наибольшему модулю), все остальные коэффициенты равны нулю. Сформулирована и дока-

зана теорема об условиях отсутствия критических ядер функции ядра Акушского, имеющая важное практическое значение для построения эффективных позиционных характеристик чисел, представленных в RNS.

- Построены многочлены, использующие интерполяционные многочлены Лагранжа, позволяющие определять знак числа и сравнивать числа над полем \mathbb{Z}_m . Доказана теорема, дающая оценку степени интерполяционного многочлена функции определения знака числа: показано, что степень многочлена равна $m - 2$. Доказана теорема, уточняющая оценку степени интерполяционного многочлена функции сравнения чисел: степень многочлена, равная $2m - 2$, уточнена до m .
- Исследована проблема построения многочлена наилучшего приближения для аппроксимации функции определения знака числа над полем \mathbb{R} . Показано, что если степень аппроксимирующего многочлена $n = 0$, то многочленами наилучшего приближения являются $Q_n(x) = a_0$, где $|a_0| \leq 1$. Доказано, что если степень аппроксимирующего многочлена $n \geq 1$, то не существует многочленов наилучшего приближения, являющихся четными функциями. Если степень аппроксимирующего многочлена $n \geq 1$, то существует единственный многочлен наилучшего приближения, являющийся нечетной функцией, который строится с помощью интерполяционной формулы Лагранжа, где в качестве узлов интерполяции используются нули многочлена Чебышева второго рода. Доказано, что если $n \geq 1$ и n – нечетное число, то не существует многочленов наилучшего приближения, являющихся функциями общего вида. Если $n \geq 1$ и n – четное число, то существует несчетное множество многочленов наилучшего приближения, являющихся функциями общего вида. Для каждого рассмотренного случая построены аппроксимирующие многочлены и доказано, что каждый из них является многочленом наилучшего приближения. Для случаев, когда многочлена наилучшего приближения не существует, также доказаны соответствующие теоремы.
- Предложен модифицированный нейросетевой метод определения знака числа над полем \mathbb{R} , позволяющий более, чем в 15.1 раза, повысить точность указанной операции в окрестности проблемной точки $x = 0$.

- Исследован вопрос об интерполяции функции ранга числа с помощью алгебраических многочленов. Доказан ряд теорем, позволяющих утверждать, что не существует многочлена, заданного над \mathbb{Z}_p , позволяющего вычислить ранг числа, представленного в RNS, вне зависимости от его формы: классической форма ранга, следующей из Китайской теоремы об остатках, нормализованной формы ранга числа или ранга числа, построенного с использованием функции ядра Акушского.
- Предложен эффективный метод вычисления ранга числа, основанный на использовании функции ядра Акушского, не содержащей критических ядер. Доказаны теоремы, дающие оценку верхней и нижней границ разрядности констант при использовании приближенного метода для вычисления ранга числа. Показано, что наборы модулей, удовлетворяющие полученной оценке, не являются компактной последовательностью. Предложенный метод позволяет сократить объем необходимых вычислений и увеличить скорость вычисления ранга числа по сравнению с приближенным методом: для нахождения ранга числа с использованием приближенного метода необходимо выполнить n операций с числами, превышающими значение модуля, тогда как в предлагаемом методе необходимо выполнить $\frac{n(n-1)}{2}$ операций с числами, не превышающими значение модуля.
- Разработаны алгоритмы вычисления ранга числа, представленного в RNS, и доказаны теоремы, позволяющие осуществлять контроль результатов обработки закодированных чисел с использованием арифметических свойств классического и нормализованного рангов.
- Представлена конфигурируемая масштабируемая двухуровневая структура доступа на основе RRNS (2Lbp-RRNS), разработанная для надежного и безопасного хранения данных в мультиоблачных системах, допускающая реализацию гомоморфных вычислений и позволяющая осуществлять параллельную обработку данных с сохранением их конфиденциальности. Разработанная схема 2Lbp-RRNS является расширением классической схемы 2L-RRNS. Высокая эффективность и производительность 2Lbp-RRNS достигается за счет использования расстояния Хэмминга.
- Получена верхняя граница для количества обнаруживаемых и исправляемых ошибок при использовании традиционных пороговых двухуров-

невых схем 2L-RRNS и предложенных пороговых двухуровневых схем 2Lbp-RRNS.

- Предложены эффективные реализации алгоритмов кодирования и декодирования данных в 2Lbp-RRNS: эффективность при кодировании достигается за счет использования метода Паскаля и FRNN, эффективность декодирования обусловлена использованием MRC, FRNN и CNN.
- Установлено, что MRC16 является более сбалансированным и быстрым алгоритмом кодирования/декодирования данных в 2Lbp-RRNS, превосходящим MRC8 и Mignotte. Производительность MRC16 при кодировании-загрузке колеблется в диапазоне 0.406-0.837 МБ/с, а при выгрузке-декодировании в диапазоне 0.54-1.093 МБ/с в зависимости от параметров хранилищ. Для сравнения, показатели наиболее близкого по производительности алгоритма Mignotte при кодировании-загрузке колеблются в диапазоне 0.13-0.257 МБ/с, а при выгрузке-декодировании в диапазоне 0.16-0.292 МБ/с. Все экспериментальные данные получены для восьми реальных облачных хранилищ.

Полученные результаты позволили:

Снизить:

- время выполнения операции в 1.75 раз
- аппаратные затраты в 1.41 раз
- энергопотребление в 2 раза

для устройства сравнения зашифрованных чисел по сравнению с самыми быстрыми существующими реализациями.

Увеличить скорость гомоморфных шифров:

- кодирования в 13.73 раз
- декодирования в 120.04 раз

Повысить надежность и отказоустойчивость:

- увеличить в 1.58 раз количество обнаруживаемых ошибок
- увеличить в 3.37 раз количество исправляемых ошибок

для двухуровневой RRNS.

Увеличение скорости доступа к данным:

- загрузки в 1.67 раз
- скачивания в 4.53 раза

СПИСОК ЛИТЕРАТУРЫ

Статьи автора в журналах, рекомендованных ВАК РФ, Scopus, Web of Science

1. Безопасная и надежная передача данных в MANET на основе принципов вычислительно стойкого разделения секрета / Н.И. Червяков, М.А. Дерябин, А.С. Назаров, М. Г. Бабенко, Н. Н. Кучеров, А. В. Гладков, Г. И. Радченко // *Труды Института системного программирования РАН*. — 2019. — Т. 31. — № 2. — С. 153–169.
2. Новая схема хранения информации в облачной среде на основе системы остаточных классов и схем разделения секрета / Н.И. Червяков, М.Г. Бабенко, Н.Н. Кучеров и др. // *Современная наука и инновации*. — 2017. — Т. 4. — № 20. — С. 21–25.
3. Обучение многослойного перцептрона с учителем в задаче распознавания с помощью корреляционного показателя / Н. А. Вершков, М. Г. Бабенко, В. А. Кучуков, Н. Н. Кучукова // *Труды Института системного программирования РАН*. — 2021. — Т. 33. — № 1. — С. 33–46.
4. Разработка нового нейросетевого метода вычисления модульного умножения в системе остаточных классов / Н.И. Червяков, М.Г. Бабенко, А.Н. Черных и др. // *Нейрокомпьютеры: разработка, применение*. — 2016. — № 10. — С. 41–48.
5. Червяков Н. И., Бабенко М. Г. Разработка схемы разделения секрета видеоизображения на основе матрицы Адамара в нейросетевом модулярном базисе // *Нейрокомпьютеры: разработка, применение*. — 2014. — №. 9. — С. 25–29.
6. Червяков Н. И., Бабенко М. Г., Кучеров Н. Н. Алгебраические аспекты эффективной реализации методов защиты информации в облачных вычислениях с использованием системы остаточных классов // *Инфокоммуникационные технологии*. — 2016. — Т. 14. — № 4. — С. 343–349.

7. Эффективное сравнение чисел в системе остаточных классов на основе позиционной характеристики / М. Г. Бабенко, А. Н. Черных, Н. И. Червяков и др. // *Труды Института системного программирования РАН*. — 2019. — Т. 31. — № 2. — С. 187–201.
8. 2Lbp-RRNS: Two-Levels RRNS With Backpropagation for Increased Reliability and Privacy-Preserving of Secure Multi-Clouds Data Storage / V. Miranda-López, A. Tchernykh, M. Babenko et al. // *IEEE Access*. — 2020. — Vol. 8. — P. 199424–199439.
9. An Efficient Method for Comparing Numbers and Determining the Sign of a Number in RNS for Even Ranges / A. Tchernykh, M. Babenko, E. Shiriaev, et al. // *Computation*. — 2022. — Vol. 10. — P. 17. <https://www.mdpi.com/2079-3197/10/2/17>.
10. AC-RRNS: Anti-collusion secured data sharing scheme for cloud storage / A. Tchernykh, M. Babenko, N. Chervyakov et al. // *International Journal of Approximate Reasoning*. — 2018. — Vol. 102. — P. 60–73. <https://www.sciencedirect.com/science/article/pii/S0888613X18300562>.
11. AR-RRNS: Configurable reliable distributed data storage systems for Internet of Things to ensure security / N. Chervyakov, M. Babenko, A. Tchernykh et al. // *Future Generation Computer Systems*. — 2019. — Vol. 92. — P. 1080–1092. <https://www.sciencedirect.com/science/article/pii/S0167739X17306015>.
12. An Approximate Method for Comparing Modular Numbers and its Application to the Division of Numbers in Residue Number Systems / N. I. Chervyakov, M. G. Babenko, P. A. Lyakhov, I. N. Lavrinenko // *Cybernetics and Systems Analysis*. — 2014. — Vol. 50. — № 6. — P. 977–984. <https://doi.org/10.1007/s10559-014-9689-2>.
13. Babenko M. G., Golimblevskaia E. I., Shiriaev E. M. Comparative analysis of homomorphic encryption algorithms based on learning with errors // *Proceedings of the Institute for System Programming of the RAS*. — 2020. — Vol. 32. — № 2. — P. 37–52.

14. Babenko M., Tchernykh A., Kuchukov V. Improved Modular Division Implementation with the Akushsky Core Function // *Computation*. — 2022. — Vol. 10. — P. 9.
15. Chervyakov N. I., Babenko M. G., Kucherov N. N. Development of Homomorphic Encryption Scheme Based on Polynomial Residue Number System // *Siberian Electronic Mathematical Reports-Sibirskie Elektronnye Matematicheskie Izvestiya*. — 2015. — Vol. 12. — P. C33–C41.
16. Comparison of modular numbers based on the chinese remainder theorem with fractional values / N. I. Chervyakov, A. S. Molahosseini, P. A. Lyakhov, M. G. Babenko, I. N. Lavrinenko, A. V. Lavrinenko // *Automatic Control and Computer Sciences*. — 2015. — Vol. 49. — № 6. — P. 354–365. URL: <https://doi.org/10.3103/S0146411615060048> – (дата обращения: 17.10.2021).
17. A Division Algorithm in a Redundant Residue Number System Using Fractions / N. Chervyakov, P. Lyakhov, M. Babenko et al. // *Applied Sciences*. — 2020. — Vol. 10. — № 2. — P. 695. URL: <https://www.mdpi.com/2076-3417/10/2/695> – (дата обращения: 17.10.2021).
18. Dynamic performance–Energy tradeoff consolidation with contention-aware resource provisioning in containerized clouds / R. M. Canosa-Reyes, A. Tchernykh, J. M. Cortes-Mendoza et al. // *PLoS ONE*. — 2022. — Vol. 17. — № 1. — P. e0261856. URL: <https://doi.org/10.1371/journal.pone.0261856> – (дата обращения: 15.02.2022).
19. A High-Speed Division Algorithm for Modular Numbers Based on the Chinese Remainder Theorem with Fractions and Its Hardware Implementation / N. Chervyakov, P. Lyakhov, M. Babenko et al. // *Electronics*. — 2019. — Vol. 8. — № 3. — P. 261. URL: <https://www.mdpi.com/2079-9292/8/3/261> – (дата обращения: 17.10.2021).
20. High performance parallel computing in residue number system / M. Deryabin, N. Chervyakov, A. Tchernykh, M. Babenko, M. Shabalina // *International Journal of Combinatorial Optimization Problems and Informatics*. — 2018. — Vol. 9. — № 1. — P. 62. URL: <https://ijcopi.org/ojs/article/view/80/73> – (дата обращения: 17.10.2021).

21. En-AR-PRNS: Entropy-Based Reliability for Configurable and Scalable Distributed Storage Systems / A. Tchernykh, M. Babenko, A. Avetisyan, A. Yu. Drozdov // *Mathematics*. — 2022. — Vol. 10. — № 1. — P. 84. URL: <https://www.mdpi.com/2227-7390/10/1/84> — (дата обращения: 27.12.2021).
22. Multiple Error Correction in Redundant Residue Number Systems: A Modified Modular Projection Method with Maximum Likelihood Decoding / M. Babenko, A. Nazarov, M. Deryabin, N. Kucherov, A. Tchernykh, N. V. Hung, A. Avetisyan, V. Toporkov // *Applied Sciences*. — 2022. — Vol. 12. — № 1. — P. 463. URL: <https://www.mdpi.com/2076-3417/12/1/463> — (дата обращения: 04.01.2022).
23. Performance evaluation of secret sharing schemes with data recovery in secured and reliable heterogeneous multi-cloud storage / A. Tchernykh, V. Miranda-López, M. Babenko et al. // *Cluster Computing*. — 2019. — Vol. 22. — № 4. — P. 1173–1185. URL: <https://doi.org/10.1007/s10586-018-02896-9> — (дата обращения: 17.10.2021).
24. Positional characteristics for efficient number comparison over the homomorphic encryption / M. Babenko, A. Tchernykh, N. Chervyakov et al. // *Programming and Computer Software*. — 2019. — Vol. 45. — № 8. — P. 532–543.
25. Privacy-preserving neural networks with Homomorphic encryption: Challenges and opportunities / B. Pulido-Gaytan, A. Tchernykh, J. M. Cortés-Mendoza, M. Babenko, G. Radchenko, A. Avetisyan, A. Y. Drozdov // *Peer-to-Peer Networking and Applications*. — 2021. — Vol. 14. — № 3. — P. 1666–1691. URL: <https://doi.org/10.1007/s12083-021-01076-8> — (дата обращения: 17.10.2021).
26. RNS Number Comparator Based on a Modified Diagonal Function / M. Babenko, M. Deryabin, S. J Piestrak et al. // *Electronics*. — 2020. — Vol. 9. — № 11. — P. 1784. URL: <https://doi.org/10.3390/electronics9111784> — (дата обращения: 17.10.2021).
27. Reliability improvement of information systems by residue number system code / A. Nazarov, N. Chervyakov, A. Tchernykh, M. Babenko // *International Journal of Combinatorial Optimization Problems and Informatics*. —

2018. — Vol. 9. — № 1. — P. 81. URL: <https://ijcopi.org/ojs/article/view/82/75> – (дата обращения: 17.10.2021).
28. Residue-to-binary conversion for general moduli sets based on approximate Chinese remainder theorem / N. I. Chervyakov, A. S. Molahosseini, P. A. Lyakhov, M. G. Babenko, M. A. Deryabin // *International Journal of Computer Mathematics*. — 2017. — Vol. 94. — № 9. — P. 1833–1849.
29. Scalable Data Storage Design for Nonstationary IoT Environment With Adaptive Security and Reliability / A. Tchernykh, M. Babenko, N. Chervyakov et al. // *IEEE Internet of Things Journal*. — 2020. — Vol. 7. — № 10. — P. 10171–10188.
30. Search for the Global Extremum Using the Correlation Indicator for Neural Networks Supervised Learning / N. Vershkov, M. Babenko, V. Kuchukov, N. Kuchukova // *Programming and Computer Software*. — 2020. — Vol. 46. — № 8. — P. 609–618. URL: <https://doi.org/10.1134/S0361768820080265> – (дата обращения: 17.10.2021).
31. The Study of Monotonic Core Functions and Their Use to Build RNS Number Comparators / M. Babenko, S. J. Piestrak, N. Chervyakov, M. Deryabin // *Electronics*. — 2021. — Vol. 10. — № 9. — P. 1041. URL: <https://www.mdpi.com/2079-9292/10/9/1041> – (дата обращения: 17.10.2021).
32. Towards the Sign Function Best Approximation for Secure Outsourced Computations and Control / M. Babenko, A. Tchernykh, B. Pulido-Gaytan, A. Avetisyan, S. Nesmachnow, X. Wang, and F. Granelli // *Mathematics*. — 2022. — Vol. 10. — №. - 12. P. 2006. URL: <https://www.mdpi.com/2227-7390/10/12/2006> – (дата обращения: 11.06.2022).
33. Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability / A. Tchernykh, U. Schwiegelsohn, E. G. Talbi, M. Babenko // *Journal of Computational Science*. — 2019. — Vol. 36. — P. 100581. URL: <https://www.sciencedirect.com/science/article/pii/S1877750316303878> – (дата обращения: 17.10.2021).
34. The architecture of a fault-tolerant modular neurocomputer based on modular number projections / N. I. Chervyakov, P. A. Lyakhov, M. G. Babenko et al. //

- Neurocomputing*. — 2018. — Vol. 272. — P. 96–107. URL: <https://www.sciencedirect.com/science/article/pii/S0925231217311918> — (дата обращения: 17.10.2021).
35. An efficient method of error correction in fault-tolerant modular neurocomputers / N. I. Chervyakov, P. A. Lyakhov, M. G. Babenko et al. // *Neurocomputing*. — 2016. — Vol. 205. — P. 32–44. URL: <https://www.sciencedirect.com/science/article/pii/S0925231216302259> — (дата обращения: 17.10.2021).
36. A new model to optimize the architecture of a fault-tolerant modular neurocomputer / N. I. Chervyakov, P. A. Lyakhov, M. G. Babenko et al. // *Neurocomputing*. — 2018. — Vol. 303. — P. 37–46. URL: <https://www.sciencedirect.com/science/article/pii/S0925231218304569> — (дата обращения: 17.10.2021).

Другие публикации автора по теме диссертации

37. A Survey on Privacy-Preserving Machine Learning with Fully Homomorphic Encryption / L. B. Pulido-Gaytan, A. Tchernykh, J. M. Cortés-Mendoza, M. Babenko, G. Radchenko // Latin American High Performance Computing Conference (CARLA) / Ed. by S. Nesmachnow, H. Castro, A. Tchernykh; Springer. Cuenca, Ecuador, 2020. — Vol. 1327 of *Communications in Computer and Information Science*. — 2020. — P. 115–129.
38. About Cloud Storage Systems Survivability / N. Kucherov, I. Dvoryaninova, M. Babenko et al. // 2020 International Workshop on Data Mining and Knowledge Engineering (YRID) / Ed. by M. Mecella, A. Fensel, M. Lapina. Stavropol, 2020. — Vol. 2842 of *CEUR Workshop Proceedings*. — 2021. — P. 43–50. URL: http://ceur-ws.org/Vol-2842/paper_5.pdf — (дата обращения: 17.10.2021).
39. Adaptive encrypted cloud storage model / E. Lopez-Falcon, A. Tchernykh, N. Chervyakov, M. Babenko, E. Nepretimova, V. Miranda-López, A. Yu. Drozdov, G. Radchenko, A. Avetisyan // 2018 IEEE Conference of Russian Young

- Researchers in Electrical and Electronic Engineering (EIconRus). Moscow and St. Petersburg, Russia. — 2018. — P. 329–334.
40. Analysis of secured distributed cloud data storage based on multilevel RNS / A. Tchernykh, M. Babenko, N. Chervyakov et al. // 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). Moscow and St. Petersburg, Russia. — 2018. — P. 382–386.
 41. Architecture Development of Cloud-Based Fail-Safe Privacy Data Storage and Processing System / N. N. Kucherov, M. G. Babenko, A. S. Nazarov, I. S. Vashchenko // 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). Moscow and St. Petersburg, Russia. — 2020. — P. 366–369. URL: <https://ieeexplore.ieee.org/abstract/document/9039130> – (дата обращения: 17.10.2021).
 42. *Babenko M., Deryabin M., Tchernykh A.* The Accuracy Estimation of the Interval-Positional Characteristic in Residue Number System // 2019 International Conference on Engineering and Telecommunication (En&T). Dolgoprudny, Russia. — 2019. — P. 1–5.
 43. *Babenko M., Golimblevskaia E.* About One Property of Number Rank in RNS // 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus) / IEEE. — IEEE Computer Society, Moscow and St. Petersburg, Russia. 2021. — P. 212–216. URL: <https://ieeexplore.ieee.org/document/9396072> – (дата обращения: 17.10.2021).
 44. *Babenko M., Golimblevskaia E.* Euclidean Division Method for the Homomorphic Scheme CKKS // 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus) / IEEE. — IEEE Computer Society, Moscow and St. Petersburg, Russia. 2021. — P. 217–220. URL: <https://ieeexplore.ieee.org/document/9396347> – (дата обращения: 17.10.2021).
 45. Bi-objective analysis of an adaptive secure data storage in a multi-cloud / E. C. Lopez-Falcon, V. Miranda-López, A. Tchernykh, M. Babenko, A. Avetisyan // Latin American High Performance Computing Conference (CARLA) / Ed. by E. Menese, H. Castro, C. Barrios Hernández, R. Ramos-Pollán; Springer. Bucaramanga, Colombia. 2018.— Vol. 979 of *Communications*

- in Computer and Information Science.* — 2019. — P. 307–321. URL: https://link.springer.com/chapter/10.1007/978-3-030-16205-4_23 – (дата обращения: 17.10.2021).
46. *Cherviakov N. I., Babenko M. G., Shabalina M. N.* Development of a secure system for distributed data storage and processing in the clouds based on the concept of active security in RNS // 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM). St. Petersburg, Russia. — 2017. — P. 558–560.
 47. *Cherviakov N. I., Babenko M. G., Shabalina M. N.* Effective implementation of Wang method using approximate method in RNS // 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM). St. Petersburg, Russia. — 2017. — P. 514–515.
 48. *Chervyakov N. I., Babenko M. G., Kuchukov V. A.* Research of effective methods of conversion from positional notation to RNS on FPGA // 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). St. Petersburg and Moscow, Russia. — 2017. — P. 277–281.
 49. Comparative Performance Analysis of Information Dispersal Methods / M. Deryabin, N. Chervyakov, A. Tchernykh, V. Berezhnoy, A. Djurabaev, A. Nazarov, M. Babenko // 2019 24th Conference of Open Innovations Association (FRUCT). Moscow, Russia. — 2019. — P. 67–74.
 50. Comparative analysis of the scalar point multiplication algorithms in the NIST FIPS 186 elliptic curve cryptography / M. Babenko, A. Tchernykh, A. Redvanov, A. Djurabaev // 3rd International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE) / Ed. by I. Bychkov, A. Tchernykh, A. Feoktistov. Irkutsk, Russia. 2021. — Vol. 2913 of *CEUR Workshop Proceedings*. — 2021. — P. 21–31. URL: <http://ceur-ws.org/Vol-2913/paper2.pdf> – (дата обращения: 17.10.2021).
 51. Computation of positional characteristics of numbers in RNS based on approximate method / N. I. Chervyakov, M. G. Babenko, M. A. Deryabin et al. // 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIconRusNW). St. Petersburg, Russia. — 2016. — P. 177–179.

52. Computationally secure threshold secret sharing scheme with minimal redundancy. / M. G Babenko, A. Tchernykh, E. Golimblevskaia et al. // 2nd International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE). / Ed. by I. Bychkov, A. Tchernykh, A. Feoktistov. Irkutsk, Russia. 2020. — Vol. 2638 of *CEUR Workshop Proceedings*. — 2020. — P. 23–32. URL: <http://ceur-ws.org/Vol-2638/paper2.pdf> — (дата обращения: 17.10.2021).
53. Cryptanalysis of secret sharing schemes based on spherical spaces / N. I. Chervyakov, M. G. Babenko, M. A. Deryabin, A. S. Nazarov // 2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT). Astana, Kazakhstan.— 2014. — P. 1–5.
54. Cryptographic Primitives Optimization Based on the Concepts of the Residue Number System and Finite Ring Neural Network / A. Tchernykh, M. Babenko, B. Pulido-Gaytan et al. // International Conference on Optimization and Learning (OLA) / Ed. by B. Dorronsoro, L. Amodeo, M. Pavone, P. Ruiz; Springer. Catania, Italy. 2021. — Vol. 1443 of *Communications in Computer and Information Science*. — 2021. — P. 241–253.
55. Data Reliability and Redundancy Optimization of a Secure Multi-cloud Storage Under Uncertainty of Errors and Falsifications / A. Tchernykh, M. Babenko, V. Kuchukov et al. // 2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). Rio de Janeiro, Brazil. — 2019. — P. 565–572.
56. Determining the rank of a number in the residue number system / M. Babenko, N. Kucherov, A. Tchernykh et al. // 3rd International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE) / Ed. by I. Bychkov, A. Tchernykh, A. Feoktistov. Irkutsk, Russia. 2021. — Vol. 2913 of *CEUR Workshop Proceedings*. — 2021. — P. 8–20. URL: <http://ceur-ws.org/Vol-2913/paper1.pdf> — (дата обращения: 17.10.2021).
57. Development of Information Security’s Theoretical Aspects in Cloud Technology with the Use of Threshold Structures / N. Chervyakov, M. Babenko, M. Deryabin, A. Garianina // 2014 International Conference on Engineering and Telecommunication (En&T). Moscow, Russia. — 2014. — P. 38–42.

58. Development of a control system for computations in BOINC with homomorphic encryption in residue number system / M. Babenko, N. Kucherov, A. Tchernykh et al. // 3rd International Conference BOINC-Based High Performance Computing: Fundamental Research and Development (BOINC:FAST) / Ed. by E. Ivahsko, A. Rumyantsev. Petrozavodsk, Russia. 2017. — Vol. 1973 of *CEUR Workshop Proceedings*. — 2017. — P. 77–84. URL: <http://ceur-ws.org/Vol-1973/paper10.pdf> – (дата обращения: 17.10.2021).
59. Experimental Evaluation of Homomorphic Comparison Methods / M. Babenko, A. Tchernykh, B. Pulido-Gaytan et al. // 2020 Ivannikov ISPRAS Open Conference (ISPRAS) / IEEE. Moscow, Russia. — 2020. — P. 69–74. URL: <https://ieeexplore.ieee.org/document/9394162> – (дата обращения: 17.10.2021).
60. Experimental analysis of large prime numbers generation in residue number system / N. I. Chervyakov, M. G. Babenko, D. S. Konyaeva et al. // 2017 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS). St. Petersburg, Russia. — 2017. — P. 315–318.
61. Experimental analysis of secret sharing schemes for cloud storage based on RNS / V. Miranda-López, A. Tchernykh, J. M. Cortés-Mendoza et al. // Latin American High Performance Computing Conference (CARLA) / Ed. by E. Mocskos, S. Nesmachnow; Springer. Buenos Aires, Argentina, and Colonia del Sacramento, Uruguay. 2017 — Vol. 796 of *Communications in Computer and Information Science*. — 2018. — P. 370–383. URL: https://link.springer.com/chapter/10.1007/978-3-319-73353-1_26 – (дата обращения: 17.10.2021).
62. Fast modular multiplication execution in residue number system / N. I. Chervyakov, M. G. Babenko, V. A. Kuchukov et al. // 2016 IEEE Conference on Quality Management, Transport and Information Security, Information Technologies (IT&MQ&IS). Nalchik, Russia. — 2016. — P. 30–32.
63. Homomorphic Comparison Methods: Technologies, Challenges, and Opportunities / M. Babenko, A. Tchernykh, E. Golimblevskaia et al. // 2020 International Conference Engineering and Telecommunication (En&T) / IEEE. —

- Dolgoprudny, Russia. 2020. — P. 1–5. URL: <https://ieeexplore.ieee.org/document/9431252> – (дата обращения: 17.10.2021).
64. Improvement of the Approximate Method for the Comparison Operation in the RNS / E. Shiryaev, E. Golimblevskaia, M. Babenko et al. // 2020 International Conference Engineering and Telecommunication (En&T). Dolgoprudny, Russia. — 2020. — P. 1–6.
65. Increasing reliability and fault tolerance of a secure distributed cloud storage / N. N. Kucherov, M. G. Babenko, A. Tchernykh et al. // 2nd International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE) / Ed. by I. Bychkov, A. Tchernykh, A. Feoktistov. Irkutsk, Russia. 2020. — Vol. 2638 of *CEUR Workshop Proceedings*. — 2020. — P. 166–180. URL: <http://ceur-ws.org/Vol-2638/paper16.pdf> – (дата обращения: 17.10.2021).
66. *Kucherov N. N., Deryabin M. A., Babenko M. G.* Homomorphic Encryption Methods Review // 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIconRus). St. Petersburg and Moscow, Russia. — 2020. — P. 370–373. URL: <https://ieeexplore.ieee.org/document/9039110> – (дата обращения: 17.10.2021).
67. *Kuchukov V., Babenko M.* The Application of Modular Arithmetic for Matrix Calculations // 2019 Ivannikov Ispras Open Conference (ISPRAS). Moscow, Russia. — 2019. — P. 49–54.
68. LR-GD-RNS: Enhanced Privacy-Preserving Logistic Regression Algorithms for Secure Deployment in Untrusted Environments / J. M. Cortés-Mendoza, G. Radchenko, A. Tchernykh, B. Pulido-Gaytan, M. Babenko, A. Avetisyan, P. Bouvry, A. Zomaya // 2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid) / IEEE. Melbourne, Australia. — 2021. — P. 770–775. URL: <https://ieeexplore.ieee.org/document/9499569> – (дата обращения: 17.10.2021).
69. Multi-objective Configuration of a Secured Distributed Cloud Data Storage / L. E. García-Hernández, A. Tchernykh, V. Miranda-López, M. Babenko, A. Avetisyan, R. Rivera-Rodriguez, G. Radchenko, C. J. Barrios-Hernandez, H. Castro, A. Yu. Drozdov // Latin America High Performance Computing

- Conference (CARLA) / Ed. by J. Luis Crespo-Mariño, E. Meneses-Rojas. Turrialba, Costa Rica. 2019. — Vol. 1087 of *Communications in Computer and Information Science*. — Cham: Springer International Publishing, 2020. — P. 78–93.
70. *Nazarov A., Babenko M., Golimblevskaia E.* Efficient Hardware Implementation of Forward Conversion WNS-RNS on FPGA // 2020 International Conference Engineering and Telecommunication (En&T). Dolgoprudny, Russia. — 2020. — P. 1–4.
71. *Nazarov A., Babenko M., Golimblevskaia E.* Hardware Implementation of the Reverse Conversion RNS-WNS on FPGA // 2020 International Conference Engineering and Telecommunication (En&T). Dolgoprudny, Russia. — 2020. — P. 1–5.
72. Neural network method for base extension in residue number system. / M. G. Babenko, E. Shiriaev, A. Tchernykh, E. Golimblevskaia // 2nd International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE) / Ed. by I. Bychkov, A. Tchernykh, A. Feoktistov. Irkutsk, Russia. 2020. — Vol. 2638 of *CEUR Workshop Proceedings*. — 2020. — P. 9–22. URL: <http://ceur-ws.org/Vol-2638/paper1.pdf> — (дата обращения: 17.10.2021).
73. Privacy-Preserving Logistic Regression as a Cloud Service Based on Residue Number System / J. M. Cortés-Mendoza, A. Tchernykh, M. Babenko et al. // International Conference Russian Supercomputing Days (RuSCDays) / Ed. by Vladimir Voevodin, Sergey Sobolev. 2020. — Vol. 1331 of *Communications in Computer and Information Science*. — Cham: Springer International Publishing, 2020. — P. 598–610.
74. Protocol for Secure and Reliable Data Transmission in MANET based on Modular Arithmetic / M. Deryabin, M. Babenko, A. Nazarov et al. // 2019 International Conference on Engineering and Telecommunication (En&T). Dolgoprudny, Russia. — 2019. — P. 1–5.
75. RRNS Base Extension Error-Correcting Code for Performance Optimization of Scalable Reliable Distributed Cloud Data Storage / M. Babenko, A. Tchernykh,

- B. Pulido-Gaytan et al. // 2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW) / IEEE. — Portland, OR, USA. 2021. — P. 548–553. URL: <https://ieeexplore.ieee.org/document/9460678> – (дата обращения: 17.10.2021).
76. Realization problems of cryptographic transformations by transfer of modular data in security systems / N. I. Chervyakov, M. G. Babenko, A. S. Nazarov, A. I. Garianina // 2015 International Siberian Conference on Control and Communications (SIBCON). Omsk, Russia. — 2015. — P. 1–5.
77. Secure Verifiable Secret Short Sharing Scheme for Multi-Cloud Storage / M. Deryabin, N. Chervyakov, A. Tchernykh, M. Babenko, N. Kucherov, V. Miranda-López, A. Avetisyan // 2018 International Conference on High Performance Computing Simulation (HPCS). Orleans, France. — 2018. — P. 700–706.
78. Security analysis of homomorphic encryption scheme for cloud computing: Known-plaintext attack / M. Babenko, N. Chervyakov, A. Tchernykh et al. // 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). Moscow and St. Petersburg, Russia. — 2018. — P. 270–274.
79. The Data Transfer Development in MANET Networks on the Base of Chinese Remainder Theorem / N. I. Chervyakov, M. G. Babenko, N. N. Kucherov et al. // Proceedings of the First International Scientific Conference “Intelligent Information Technologies for Industry” (IITI) / Ed. by A. Abraham, S. Kovalev, V. Tarassov, V. Snasel; Springer. Sochi, Russia. 2016. — Vol. 451 of *Advances in Intelligent Systems and Computing*. — 2016. — P. 3–13.
80. The Development of Probabilistic Algorithm of Monitoring a Result Correctness for Cloud Computing in Residue Number System / N. I. Chervyakov, A. S. Nazarov, M. G. Babenko et al. // 2015 International Conference on Engineering and Telecommunication (En&T). Moscow, Russia. — 2015. — P. 45–49.
81. The development of secure mobile computing network based on secret sharing schemes / N. I. Chervyakov, M. G. Babenko, M. A. Deryabin et al. // 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW). St. Petersburg, Russia. — 2016. — P. 180–184.

82. The Effective Neural Network Implementation of the Secret Sharing Scheme with the Use of Matrix Projections on FPGA / N. I. Chervyakov, M. G. Babenko, N. N. Kucherov, A. I. Garianina // ICSI 2015 held in conjunction with the 2nd BRICS Congress on Computational Intelligence, CCI 2015 / Ed. by Y. Tan, Y. Shi, F. Buarque et al. Beijing, China. 2015. — Vol. 9142 of *Lecture Notes in Computer Science*. — 2015. — P. 3–10. URL: https://link.springer.com/chapter/10.1007/978-3-319-20469-7_1 – (дата обращения: 17.10.2021).
83. The Fast Algorithm for Number Comparing in Three-Modular RNS / N. Chervyakov, M. Babenko, A. Tchernykh et al. // 2016 International Conference on Engineering and Telecommunication (En&T). Moscow, Russia. — 2016. — P. 26–28.
84. Toward digital twins' workload allocation on clouds with low-cost microservices streaming interaction / A. Tchernykh, A. Facio-Medina, B. Pulido-Gaytan, R. Rivera-Rodriguez, J. M. Cortés-Mendoza, G. Radchenko, M. Babenko, I. Chernykh, I. Kulikov, S. Nesmachnow // 2020 Ivannikov Ispras Open Conference (ISPRAS) / IEEE. Moscow, Russia. — 2020. — P. 115–121.
85. Towards Mitigating Uncertainty of Data Security Breaches and Collusion in Cloud Computing / A. Tchernykh, M. Babenko, N. Chervyakov et al. // 2017 28th International Workshop on Database and Expert Systems Applications / IEEE. Lyon, France. — 2017. — P. 137–141. URL: <https://ieeexplore.ieee.org/document/8049702> – (дата обращения: 17.10.2021).
86. Towards Optimizing Cloud Computing Using Residue Number System / N. Kucherov, E. Kuchukova, A. Tchernykh, V. Kuchukov, M. Babenko / International Conference «Marchuk Scientific Readings 2020» (MSR-2020), dedicated to the 95th anniversary of the birthday of RAS Academician Guri. I. Marchuk October 19 - 23, 2020, Akademgorodok, Novosibirsk, Russia. — 2021. — Vol. 1715 of *Journal of Physics: Conference*. 2021. — P. 012052. URL: <https://doi.org/10.1088/1742-6596/1715/1/012052> – (дата обращения: 17.10.2021).
87. Unfairness Correction in P2P Grids Based on Residue Number System of a Special Form / M. Babenko, N. Chervyakov, A. Tchernykh et al. // 2017

28th International Workshop on Database and Expert Systems Applications (DEXA). Lyon, France. — 2017. — P. 147–151.

88. WA-RRNS: Reliable Data Storage System Based on Multi-cloud / A. Tchernykh, M. Babenko, V. Miranda-López et al. // 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW) / IEEE. Vancouver, BC, Canada. — 2018. — P. 666–673. URL: <https://ieeexplore.ieee.org/abstract/document/8425474> – (дата обращения: 17.10.2021).
89. Weighted Two-Levels Secret Sharing Scheme for Multi-Clouds Data Storage with Increased Reliability / V. Miranda-López, A. Tchernykh, M. Babenko et al. // 2019 International Conference on High Performance Computing Simulation (HPCS) Dublin, Ireland.— 2019. — P. 915–922.

Свидетельства о государственной регистрации программ для ЭВМ

90. Свидетельство о государственной регистрации программы для ЭВМ № 2012610503 Российская Федерация. «Среда моделирования вычислений в системе остаточных классов на основе приближенных методов» («СМВСОКОПМ») / Н. И. Червяков, М. Г. Бабенко, П. А. Ляхов; заявитель и правообладатель ФГБОУ ВПО «Ставропольский государственный университет». № 2011618721; заявл. 17.11.2011; опубл. 10.01.2012– 1 с.
91. Свидетельство о государственной регистрации программы для ЭВМ № 2016612432 Российская Федерация. Ускоренный метод вычисления остатка от деления с использованием распределенной арифметики / Н. И. Червяков, М. Г. Бабенко, М. А. Дерябин, А. С. Назаров, А. В. Лавриненко; заявитель и правообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». № 2015663065; заявл. 29.12.2015; опубл. 26.02.2016– 1 с.

92. Свидетельство о государственной регистрации программы для ЭВМ № 2016618312 Российская Федерация. Программа управления устройством перевода чисел из системы остаточных классов в позиционную систему счисления на основе перевода в обобщенную позиционную систему счисления / Н. И. Червяков, М. Г. Бабенко, М. А. Дерябин, А. С. Назаров, А. В. Лавриненко; заявитель и правообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». № 2016615527; заявл. 30.05.2016; опубл. 26.07.2016– 1 с.
93. Свидетельство о государственной регистрации программы для ЭВМ № 2016618315 Российская Федерация. Программа управления устройством перевода чисел из системы остаточных классов в позиционную систему счисления на основе Китайской теоремы об остатках с дробными числами / Н. И. Червяков, М. Г. Бабенко, М. А. Дерябин, А. С. Назаров, А. В. Лавриненко; заявитель и правообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». № 2016615529; заявл. 30.05.2016; опубл. 26.07.2016– 1 с.
94. Свидетельство о государственной регистрации программы для ЭВМ № 2017615372 Российская Федерация. Программа моделирования гомоморфного шифрования в облачных вычислениях / Н. И. Червяков, М. Г. Бабенко, Н. Н. Кучеров, Ю. В. Черногорова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2017612222; заявл. 20.03.2017; опубл. 15.05.2017 – 1 с.
95. Свидетельство о государственной регистрации программы для ЭВМ № 2018612690 Российская Федерация. Среда моделирования алгоритмов цифровой фильтрации изображений / Н. И. Червяков, М. Г. Бабенко, В. А. Кучуков, Н. Г. Гудиева; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2017663489; заявл. 25.12.2017; опубл. 21.02.2018 – 1 с.
96. Свидетельство о государственной регистрации программы для ЭВМ № 2018612694 Российская Федерация. Модуль оценки рисков безопасности облачных, краевых и туманных вычислений в условиях вычислительной неопределенности / Н. И. Червяков, М. Г. Бабенко, А. Н. Черных,

- Н. Н. Кучукова, Н. Г. Гудиева; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2017663493; заявл. 25.12.2017; опубл. 21.02.2018 – 1 с.
97. Свидетельство о государственной регистрации программы для ЭВМ № 2019610808 Российская Федерация. Модуль кодирования и декодирования данных в системе остаточных классов / М. Г. Бабенко, Н. И. Червяков, А. Н. Черных, В. А. Кучуков, Н. Н. Кучеров, Е. А. Кучукова, С. Ч. Аль-Гальда; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2018665334; заявл. 28.12.2018; опубл. 18.01.2019 – 1 с.
98. Свидетельство о государственной регистрации программы для ЭВМ № 2019611375 Российская Федерация. Распределенная система надежного хранения и обработки данных в мультиоблачной среде / М. Г. Бабенко, Н. И. Червяков, А. Н. Черных, Н. Н. Кучеров, В. А. Кучуков, Е. А. Кучукова, С. Ч. Аль-Гальда; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2018665335; заявл. 28.12.2018; опубл. 24.01.2019 – 1 с.
99. Свидетельство о государственной регистрации программы для ЭВМ № 2019619806 Российская Федерация. Программный модуль деления с остатком чисел большой разрядности / В. В. Петров, М. Г. Бабенко, М. А. Дерябин, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019618843; заявл. 19.07.2019; опубл. 24.07.2019 – 1 с.
100. Свидетельство о государственной регистрации программы для ЭВМ № 2019619899 Российская Федерация. Программный комплекс моделирования методов распределения информации / Н. И. Червяков, М. А. Дерябин, А. Э. Джурабаев, М. Г. Бабенко, А. С. Редванов; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019618665; заявл. 17.07.2019; опубл. 26.07.2019 – 1 с.
101. Свидетельство о государственной регистрации программы для ЭВМ № 2019661394 Российская Федерация. Программа модулярного нейросетевого кодирования данных / М. Г. Бабенко, Е. А. Кучукова; заявитель

- и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019618837; заявл. 19.07.2019; опубл. 28.08.2019 – 1 с.
102. Свидетельство о государственной регистрации программы для ЭВМ № 2019661480 Российская Федерация. Программа модулярного нейросетевого декодирования данных / М. Г. Бабенко, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019618875; заявл. 19.07.2019; опубл. 02.09.2019– 1 с.
103. Свидетельство о государственной регистрации программы для ЭВМ № 2019663760 Российская Федерация. Программный модуль эффективной реализации арифметических операций в конечном поле / М. Г. Бабенко, И. С. Ващенко, А. С. Назаров, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019662609; заявл. 16.10.2019; опубл. 23.10.2019 – 1 с.
104. Свидетельство о государственной регистрации программы для ЭВМ № 2019663930 Российская Федерация. Программный модуль сравнения чисел в системе остаточных классов / М. Г. Бабенко, И. С. Ващенко, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019662611; заявл. 16.10.2019; опубл. 25.10.2019 – 1 с.
105. Свидетельство о государственной регистрации программы для ЭВМ № 2020610041 Российская Федерация. Программа избыточного кодирования и декодирования модулярного кодам / В. А. Кучуков, М. Г. Бабенко; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2019666586; заявл. 17.12.2019; опубл. 09.01.2020– 1 с.
106. Свидетельство о государственной регистрации программы для ЭВМ № 2020618967 Российская Федерация. Программа подготовки файлов для распределенного хранения данных в облаках / Н. Н. Кучеров, М. Г. Бабенко, В. А. Кучуков, И. С. Ващенко; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020618381; заявл. 03.08.2020; опубл. 10.08.2020– 1 с.

107. Свидетельство о государственной регистрации программы для ЭВМ № 2020619140 Российская Федерация. Программа восстановления полученных данных при распределенном хранении данных в облаках / Н. Н. Кучеров, М. Г. Бабенко, В. А. Кучуков, И. С. Ващенко; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020618376; заявл. 03.08.2020; опубл. 12.08.2020 – 1 с.
108. Свидетельство о государственной регистрации программы для ЭВМ № 2020660256 Российская Федерация. Программный модуль декодирования данных с использованием минимально избыточного модулярного кода / М. Г. Бабенко, Н. Н. Кучеров, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020619581; заявл. 28.08.2020; опубл. 01.09.2020 – 1 с.
109. Свидетельство о государственной регистрации программы для ЭВМ № 2020660257 Российская Федерация. Система моделирования исправления ошибок в модулярном коде / М. Г. Бабенко, В. А. Кучуков, И. С. Ващенко; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020619583; заявл. 28.08.2020; опубл. 01.09.2020 – 1 с.
110. Свидетельство о государственной регистрации программы для ЭВМ № 2020660392 Российская Федерация. Программный модуль сбора данных о технических характеристиках облачных провайдеров в реальном режиме времени / М. Г. Бабенко, Н. А. Сотникова, Е. А. Кучукова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020619579; заявл. 28.08.2020; опубл. 03.09.2020 – 1 с.
111. Свидетельство о государственной регистрации программы для ЭВМ № 2020660531 Российская Федерация. Модуль выбора оснований системы остаточных классов для оптимизации минимально избыточного кода / М. Г. Бабенко, В. А. Кучуков, И. С. Ващенко; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020619552; заявл. 28.08.2020; опубл. 04.09.2020– 1 с.
112. Свидетельство о государственной регистрации программы для ЭВМ № 2020660532 Российская Федерация. Программный модуль управления

- адаптивной безопасностью в мультиоблачной среде / М. Г. Бабенко, Н. Н. Кучеров, Н. А. Сотникова; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020619548; заявл. 28.08.2020; опубл. 04.09.2020– 1 с.
113. Свидетельство о государственной регистрации программы для ЭВМ № 2020665103 Российская Федерация. Программа управления устройством коррекции однократных ошибок на основе перехода к обобщенной позиционной системе счисления / А. С. Назаров, М. Г. Бабенко, В. А. Кучуков, Н. Н. Кучеров; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020663618; заявл. 09.11.2020; опубл. 23.11.2020– 1 с.
114. Свидетельство о государственной регистрации программы для ЭВМ № 2020665416 Российская Федерация. Программа управления устройством коррекции однократных ошибок на основе Китайской теоремы об остатках с дробными величинами / А. С. Назаров, М. Г. Бабенко, В. А. Кучуков, Н. Н. Кучеров; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2020663608; заявл. 09.11.2020; опубл. 26.11.2020– 1 с.
115. Свидетельство о государственной регистрации программы для ЭВМ № 2021616029 Российская Федерация. Программа для умножения зашифрованных матриц с использованием СККС схемы / М. Г. Бабенко, Е. И. Голимблевская, Е. М. Ширяев; заявитель и правообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». № 2021614751; заявл. 07.04.2021; опубл. 15.04.2021– 1 с.

Патенты на изобретения

116. Патент № 038389 Евразийский патент, МПК G06F 7/02, G06F 7/72. Устройство сравнения и определения знака чисел, представленных в системе остаточных классов: № 202090736; заявл. 14.04.2020; / Дерябин М.А., Бабенко М.Г., Кучуков В. А., Назаров А.С., Кучеров Н.Н.; заявитель и патен-

- тообладатель Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет".— 8 с.
117. Патент № 2747371 Российская Федерация, МПК G06F 7/38, G06F 7/72. Устройство определения знака числа, представленного в системе остаточных классов: № 2020134778; заявл. 22.10.2020; / Бабенко М.Г., Кучуков В. А.; заявитель и патентообладатель Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет".— 15 с.
118. Патент № 2751992 Российская Федерация, МПК G06F 7/38, G06F 7/72. Устройство сравнения чисел, представленных в системе остаточных классов: № 2020134772; заявл. 22.10.2020; / Бабенко М.Г., Кучуков В. А.; заявитель и патентообладатель Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет".— 18 с.
119. Патент № 2483346 Российская Федерация, МПК G06F 11/08, G06F 7/72. Устройство для обнаружения переполнения динамического диапазона, определения ошибки и локализации неисправности вычислительного канала в эвм, функционирующих в системе остаточных классов: № 2011145755/08; заявл. 10.11.2011; опубл. 27.05.2013 /Червяков Н. И., Бабенко М. Г., Ляхов П. А., Лавриненко И. Н., Лавриненко А. В.; заявитель и патентообладатель Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет". — 10 с.
120. Патент № 2503992 Российская Федерация, МПК G06F 7/02, G06F 7/72. Устройство для сравнения чисел, представленных в системе остаточных классов: № 2011139397; заявл. 27.09.2011; опубл. 10.04.2013 / Червяков Н. И., Бабенко М. Г., Ляхов П. А., Лавриненко И. Н., Лавриненко А. В.; заявитель и патентообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». — 11 с.
121. Патент № 2503995 Российская Федерация, МПК G06F 7/72. Устройство для определения знака модулярного числа: № 2011139278; заявл.

- 26.09.2011; опубл. 10.04.2013 / Червяков Н. И., Бабенко М. Г., Ляхов П. А., Лавриненко И. Н., Лавриненко А. В.; заявитель и патентообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». — 8 с.
122. Патент № 2628179 Российская Федерация, МПК G06F7/72. Устройство деления модулярных чисел : № 2016146626; заявл. 28.11.2016; опубл. 15.08.2017 / Червяков Н. И., Бабенко М. Г., Кучуков В. А., Дерябин М. А., Лавриненко И. Н., Лавриненко А. В.; заявитель и патентообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». — 6 с.
123. Патент № 2652450 Российская Федерация, МПК G06F7/72. Устройство вычисления модулярного произведения Монтгомери : № 2017129526; заявл. 2017129526; опубл. 26.04.2018 / Червяков Н. И., Коляда А. А., Кучуков В. А., Бабенко М. Г.; заявитель и патентообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». — 2 с.
124. Патент № 2653257 Российская Федерация, МПК G06F 11/08, G06F 7/72. Устройство обнаружения и коррекции ошибки модулярного кода : № 2017126350; заявл. 21.07.2017; опубл. 07.05.2018 / Червяков Н.И., Кучуков В. А., Бабенко М. Г., Кучукова Н. Н.; заявитель и патентообладатель ФГАОУ ВО «Северо-Кавказский федеральный университет». — 3 с.
125. Патент № 2744815 Российская Федерация, МПК G06F 7/72. Устройство для перевода чисел из системы остаточных классов и расширения оснований : № 2020120649; заявл. 22.06.2020; опубл. 16.03.2021 / Бабенко М. Г., Кучуков В. А., Черных А. Н., Кучеров Н. Н.; заявитель и патентообладатель Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет". — 13 с.
126. Патент № 2559771 Российская Федерация, МПК G06F7/72. Устройство для основного деления модулярных чисел / Червяков Н. И., Бабенко М. Г., Ляхов П. А., Лавриненко И. Н.; заявитель и патентообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». — № 2013148505/08; заявл. 30.10.2013; опубл. 10.08.2015, Бил. № 22.
127. Патент № 2559772 Российская Федерация, МПК G06F7/72. Устройство для основного деления модулярных чисел в формате системы остаточных клас-

сов / Червяков Н. И., Бабенко М. Г., Ляхов П. А., Лавриненко И. Н., Лавриненко А. В.; заявитель и патентообладатель ФГАОУ ВПО «Северо-Кавказский федеральный университет». — № 2013149446/08; заявл. 06.11.2013; опубл. 10.08.2015, Бил. № 22.

Цитируемая литература

128. *Акушский И. Я., Бурцев В. М., Пак И. Т.* О новой позиционной характеристике непозиционного кода и ее приложении // Теория кодирования и оптимизация сложных систем. — Алма-Ата, Наука, КазССР. 1977. — С. 8–16.
129. *Акушский И. Я., Юдицкий Д. И.* Машинная арифметика в остаточных классах. — М.: Советское радио, 1968. — 440 с.
130. *Амербаев В. М.* Теоретические основы машинной арифметики. — Алма-Ата: Наука. КазССР, 1976. — 324 с.
131. *Бабенко Л. К., Русаловский И. Д.* Библиотека полностью гомоморфного шифрования целых чисел // *Известия Южного федерального университета. Технические науки.* — 2020. — Т. 2020. №. 2. — С. 218–227.
132. *Бахвалов Н. С., Жидков Н. П., Кобельков Г. М.* Численные методы. — М.: Наука, 1987. — 600 с.
133. *Варновский Н.П., Шокуров А.В.* Гомоморфное шифрование // *Труды Института системного программирования РАН.* — 2007. — Т. 12. — С. 27–36.
134. *Винберг Э. Б.* Курс алгебры. — 2-е изд., испр. и доп. изд. — М.: Изд-во Факториал Пресса, 2001. — 544 с.
135. *Винберг Э. Б.* Курс алгебры. — М.: Изд-во МЦНМО, 2017. — 592 с.

136. *Галушкин А И, Судариков В А, Шабанов Е В.* Нейроматематика: методы решения задач на нейрокомпьютерах // *Математическое моделирование.* — 1991. — Т. 3, № 8. — С. 93–111.
137. *Грэхем Р. Л., Кнут Д., Паташник О.* Конкретная математика. Основание информатики. — М.: МИР, 1998. — 703 с.
138. *Корн Г., Корн Т.* Справочник по математике (для научных работников и инженеров). — М.: Наука. Главная редакция физико-математической литературы, 1973. — 832 с.
139. *Лидл Р., Нидеррайтер Г.* Конечные поля: В 2-х т. Т. 1. Пер. с англ. — М.: Мир, 1988. — 430 с.
140. Методы полностью гомоморфного шифрования на основе матричных полиномов / Л. К. Бабенко, Ф. Б. Буртыка, О. Б. Макаревич, А. В. Трепачева // *Вопросы кибербезопасности.* — 2015. — Т. 9. № 1. — С. 14–25.
141. *Abu-Libdeh H., Princehouse L., Weatherspoon H.* RACS: a case for cloud storage diversity // *Proceedings of the 1st ACM symposium on Cloud computing.* Indianapolis, Indiana, USA. — 2010. — P. 229–240.
142. Adaptive energy efficient scheduling in Peer-to-Peer desktop grids / A. Tchernykh, J. E Pecero, A. Barrondo, E. Schaeffer // *Future Generation Computer Systems.* — 2014. — Vol. 36. — P. 209–220.
143. Adaptive parallel job scheduling with resource admissible allocation on two-level hierarchical grids / A. Quezada-Pina, A. Tchernykh, J. L. González-García et al. // *Future Generation Computer Systems.* — 2012. — Vol. 28. — № 7. — P. 965–976.
144. Adaptive resource allocation with job runtime uncertainty / R. Ramírez-Velarde, A. Tchernykh, C. Barba-Jimenez et al. // *Journal of Grid Computing.* — 2017. — Vol. 15. — № 4. — P. 415–434.
145. Adding long-term availability, obfuscation, and encryption to multi-cloud storage systems / A. Celesti, M. Fazio, M. Villari, A. Puliafito // *Journal of Network and Computer Applications.* — 2016. — Vol. 59. — P. 208–218.

146. *Ahmed E., Rehmani M. H.* Introduction to the special section on social collaborative Internet of Things // *Computers & Electrical Engineering*. — 2017. — Vol. 100. — № 58. — P. 382–384.
147. *Aihara K.* Chaos engineering and its application to parallel distributed processing with chaotic neural networks // *Proceedings of the IEEE*. — 2002. — Vol. 90. — № 5. — P. 919–930.
148. *Alhazmi B., Gebali F.* Fast Large Integer Modular Addition in GF(p) Using Novel Attribute-Based Representation // *IEEE Access*. — 2019. — Vol. 7. — P. 58704–58719.
149. *Alperin-Sheriff J., Peikert C.* Faster Bootstrapping with Polynomial Error // *Advances in Cryptology – CRYPTO 2014* / Ed. by J. A. Garay, R. Gennaro. Santa Barbara, USA. 2014.— Vol. 8616 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. — P. 297–314.
150. *Alrimeih H., Rakhmatov D.* Fast and flexible hardware support for ECC over multiple standard prime fields // *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*. — 2014. — Vol. 22. — № 12. — P. 2661–2674.
151. *Asmuth C., Bloom J.* A modular approach to key safeguarding // *IEEE Transactions on Information Theory*. — 1983. — Vol. 29. — № 2. — P. 208–210.
152. *Attas D., Batrafi O.* Efficient integrity checking technique for securing client data in cloud computing // *International Journal of Electrical & Computer Sciences* . — 2011. — Vol. 11. — № 05. — P. 43–48.
153. *Attasena V., Darmont J., Harbi N.* Secret Sharing for Cloud Data Security // *The International Journal on Very Large Databases*. — 2017. — Vol. 26. — № 5. — P. 657–681. URL: <https://hal.archives-ouvertes.fr/hal-01529610> – (дата обращения: 24.10.2021).
154. *Bajard J-C, Imbert L.* A full RNS implementation of RSA // *IEEE Transactions on Computers*. — 2004. — Vol. 53. — № 6. — P. 769–774.
155. *Banik S., Bogdanov A., Regazzoni F.* Compact circuits for combined AES encryption/decryption // *Journal of Cryptographic Engineering*. — 2019. — Vol. 9. — № 1. — P. 69–83.

156. *Barron A. R.* Universal approximation bounds for superpositions of a sigmoidal function // *IEEE Transactions on Information theory*. — 1993. — Vol. 39. — № 3. — P. 930–945.
157. *Barsi F., Maestrini P.* Error Detection and Correction by Product Codes in Residue Number Systems // *IEEE Transactions on Computers*. — 1974. — Vol. C-23. — № 9. — P. 915–924.
158. Batch Fully Homomorphic Encryption over the Integers / J. H. Cheon, J.-S. Coron, J. Kim et al. // *Advances in Cryptology – EUROCRYPT 2013* / Ed. by Thomas Johansson, Phong Q. Nguyen. Athens, Greece.— Vol. 7881 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. — P. 315–335.
159. *Benaloh J.* Dense probabilistic encryption // *Proceedings of the Workshop on Selected Areas of Cryptography*. Kingston, Ontario, Canada. — 1994. — P. 120–128.
160. *Bi S., Gross W. J.* The mixed-radix Chinese remainder theorem and its applications to residue comparison // *IEEE Transactions on Computers*. — 2008. — Vol. 57. — № 12. — P. 1624–1632.
161. Bigtable: A distributed storage system for structured data / F. Chang, J. Dean, S. Ghemawat et al. // *ACM Transactions on Computer Systems (TOCS)*. — 2008. — Vol. 26. — № 2. — P. 1–26.
162. *Boneh D., Goh E.-J., Nissim K.* Evaluating 2-DNF Formulas on Ciphertexts // *Theory of Cryptography Conference (TCC)* / Ed. by J. Kilian. Cambridge, MA, USA.— Vol. 3378 of *Lecture Notes in Computer Science* — Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. — P. 325–341.
163. *Boneh D., Shaw J.* Collusion-secure fingerprinting for digital data // *IEEE Transactions on Information Theory*. — 1998. — Vol. 44. — № 5. — P. 1897–1905.
164. *Bonte C., Vercauteren F.* Privacy-preserving logistic regression training // *BMC medical genomics*. — 2018. — Vol. 11. — № 4. — P. 13–21.

165. *Boura Christina, Gama Nicolas, Georgieva Mariya.* Chimera: a unified framework for B/FV, TFHE and HEAAN fully homomorphic encryption and predictions for deep learning. // *IACR Cryptol. ePrint Arch.* — 2018. — Vol. 2018. — P. 758.
166. *Brakerski Z.* Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP // *Advances in Cryptology – CRYPTO 2012 / Ed. by R. Safavi-Naini, R. Canetti.* Santa Barbara, CA, USA.— Vol. 7417 of *Lecture Notes in Computer Science* — Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. — P. 868–886.
167. *Brakerski Z., Gentry C., Vaikuntanathan V.* (Leveled) fully homomorphic encryption without bootstrapping // *ACM Transactions on Computation Theory (TOCT).* — 2014. — Vol. 6. — № 3. — P. 1–36.
168. *Brakerski Z., Vaikuntanathan V.* Efficient fully homomorphic encryption from (standard) LWE // *SIAM Journal on Computing.* — 2014. — Vol. 43. — № 2. — P. 831–871.
169. *Brickell E. F., Yacobi Y.* On Privacy Homomorphisms (Extended Abstract) // *Advances in Cryptology – EUROCRYPT’ 87 / Ed. by D. Chaum, W. L. Price.* Amsterdam, The Netherlands.— Vol. 304 of *Lecture Notes in Computer Science.*— Berlin, Heidelberg: Springer Berlin Heidelberg, 1988. — P. 117–125.
170. *Burgess N.* Scaling an RNS number using the core function // *Proceedings 2003 16th IEEE Symposium on Computer Arithmetic / IEEE.* — 2003. — P. 262–269.
171. *Butler B.* No TitBrandon Butler and the Cloud Provider With the Best Uptime in 2015. [Электронный ресурс] –Режим доступа: — <https://www.networkworld.com/article/3020235/and-the-cloud-provider-with-the-best-uptime-in-2015-is.html>, свободный. – (дата обращения: 17.10.2021).
172. CA-DAG: Modeling communication-aware applications for scheduling in cloud computing / D. Kliazovich, J. E Pecero, A. Tchernykh et al. // *Journal of Grid Computing.* — 2016. — Vol. 14. — № 1. — P. 23–39.

173. *Spiess C., Frantz B., Fitzpatrick G. et al.* CIS Amazon Web Services Foundations. — [Электронный ресурс]. — Режим доступа: https://d1.awsstatic.com/whitepapers/compliance/AWS_CIS_Foundations_Benchmark.pdf. свободный. — (дата обращения: 17.10.2021).
174. CRT-based fully homomorphic encryption over the integers / J. H. Cheon, J. Kim, M. S. Lee, A. Yun // *Information Sciences*. — 2015. — Vol. 310. — P. 149–162. <https://www.sciencedirect.com/science/article/pii/S002002551500184X>.
175. *Chen H., Chillotti I., Song Y.* Improved bootstrapping for approximate homomorphic encryption // Annual International Conference on the Theory and Applications of Cryptographic Techniques. Darmstadt, Germany. 2019. / Springer. — Vol. 11477 of *Lecture Notes in Computer Science*. 2019. — P. 34–54.
176. *Chen T., Chen H.* Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems // *IEEE Transactions on Neural Networks*. — 1995. — Vol. 6. — № 4. — P. 911–917.
177. *Chen Z., Cao F.* The approximation operators with sigmoidal functions // *Computers & Mathematics with Applications*. — 2009. — Vol. 58. — № 4. — P. 758–765.
178. *Cheon J. H., Kim D., Kim D.* Efficient homomorphic comparison methods with optimal complexity // International Conference on the Theory and Application of Cryptology and Information Security. Daejeon, South Korea. 2020. / Springer. — Vol. 12492 of *Lecture Notes in Computer Science*. 2020. — P. 221–256.
179. *Chessa S., Di Pietro R., Maestrini P.* Dependable and Secure Data Storage in Wireless Ad Hoc Networks: An Assessment of DS2 // Wireless On-Demand Network Systems (WONS) / Ed. by R. Battiti, M. Conti, R. L. Cigno. Madonna di Campiglio, Italy. 2004. — Vol. 2928 of *Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. — P. 184–198.
180. *Chialva D., Dooms A.* Conditionals in homomorphic encryption and machine learning applications // *arXiv preprint arXiv:1810.12380*. — 2018. URL: <https://arxiv.org/abs/1810.12380> — (дата обращения: 24.10.2021).

181. Cloud service delivery across multiple cloud platforms / I. Houidi, M. Mechtri, W. Louati, D. Zeglache // 2011 IEEE International Conference on Services Computing / IEEE. Washington, DC, USA. — 2011. — P. 741–742.
182. Cloud storage reliability for big data applications: A state of the art survey / R. Nachiappan, B. Javadi, R. N. Calheiros, K. M. Matawie // *Journal of Network and Computer Applications*. — 2017. — Vol. 97. — P. 35–47.
183. *Collberg C., Thomborson C., Low D.* Tech. Rep.: : Citeseer, 1997. [Электронный ресурс]. — Режим доступа: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.2651&rep=rep1&type=pdf>, свободный. — (дата обращения: 17.10.2021).
184. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy / R. Gilad-Bachrach, N. Dowlin, K. Laine et al. // International Conference on Machine Learning / PMLR. New York, NY, USA. — Vol. 48. 2016. — P. 201–210.
185. Csbauditor: Proactive security risk analysis for cloud storage broker systems / K. A. Torkura, M. I. H. Sukmana, T. Strauss et al. // 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA) / IEEE. Cambridge, MA, USA. — 2018. — P. 1–10.
186. *Cybenko G.* Approximation by superpositions of a sigmoidal function // *Mathematics of Control, Signals and Systems*. — 1992. — Vol. 5. — № 4. — P. 455–455.
187. DPPDL: A Dynamic Partial-Parallel Data Layout for Green Video Surveillance Storage / Z. Sun, Q. Zhang, Y. Li, Y.-A. Tan // *IEEE Transactions on Circuits and Systems for Video Technology*. — 2018. — Vol. 28. — № 1. — P. 193–205.
188. DRINA: A lightweight and reliable routing approach for in-network aggregation in wireless sensor networks / L. A. Villas, A. Boukerche, H. S. Ramos et al. // *IEEE Transactions on Computers*. — 2012. — Vol. 62. — № 4. — P. 676–689.
189. DROPS: division and replication of data in cloud for optimal performance and security / M. Ali, K. Bilal, S. U. Khan et al. // *IEEE Transactions on Cloud computing*. — 2015. — Vol. 6. — № 2. — P. 303–315.

190. *Damgård I., Jurik M.* A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System // Public Key Cryptography (PKC) / Ed. by K. Kim. Cheju Island, Korea. 2001.— Vol.1992 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. — P. 119–136.
191. *Dean J., Ghemawat S.* MapReduce: simplified data processing on large clusters // *Communications of the ACM*. — 2008. — Vol. 51. — № 1. — P. 107–113.
192. DepSky: dependable and secure storage in a cloud-of-clouds / A. Bessani, M. Correia, B. Quaresma et al. // *ACM Transactions on Storage (ToS)*. — 2013. — Vol. 9. — № 4. — P. 1–33.
193. *Diffie W., Hellman M.* New directions in cryptography // *IEEE Transactions on Information Theory*. — 1976. — Vol. 22. — № 6. — P. 644–654.
194. Digital forensics for network, internet, and cloud computing / T. V. Lillard, C. P. Garrison, C. A. Schiller et al. // *Syngress Publication Elsevier Inc.* — 2010.
195. *Dimauro G., Impedovo S., Pirlo G.* A new technique for fast number comparison in the residue number system // *IEEE Transactions on Computers*. — 1993. — Vol. 42. — № 5. — P. 608–612.
196. *Dutka J.* The early history of the factorial function // *Archive for History of Exact Sciences*. — 1991. — Vol. 43. — № 3. — P. 225–249.
197. *Dworkin M. J.* Information Tech Laboratory National Institute of Standards and Technology, SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. — 2015. URL: <https://doi.org/10.6028/NIST.FIPS.202> – (дата обращения: 17.10.2021).
198. ESDR: an efficient and secure data repairing paradigm in cloud storage / S. Zhou, R. Du, J. Chen et al. // *Security and Communication Networks*. — 2016. — Vol. 9. — № 16. — P. 3646–3657.
199. Efficient Private Comparison Queries over Encrypted Databases using Fully Homomorphic Encryption with Finite Fields / B. H. M. Tan, H. T. Lee, H. Wang et al. // *IEEE Transactions on Dependable and Secure Computing*. — 2020. — P. 1–1.

200. *Ekodeck Stéphane G. R., Ndoundam R.* PDF steganography based on Chinese Remainder Theorem // *Journal of Information Security and Applications*. — 2016. — Vol. 29. — P. 1–15 URL: <https://www.sciencedirect.com/science/article/pii/S221421261500068X> – (дата обращения: 17.10.2021).
201. *Elgamal T.* A public key cryptosystem and a signature scheme based on discrete logarithms // *IEEE Transactions on Information Theory*. — 1985. — Vol. 31. — № 4. — P. 469–472.
202. *Fan J., Vercauteren F.* Somewhat practical fully homomorphic encryption. // *IACR Cryptol. ePrint Arch.* — 2012. — Vol. 2012. — P. 144. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.400.6346&rep=rep1&type=pdf> – (дата обращения: 17.10.2021).
203. *Fasi M.* Optimality of the Paterson–Stockmeyer method for evaluating matrix polynomials and rational matrix functions // *Linear Algebra and its Applications*. — 2019. — Vol. 574. — P. 182–200. URL: <https://www.sciencedirect.com/science/article/pii/S0024379519301454> – (дата обращения: 17.10.2021).
204. Fast Data Collection in Tree-Based Wireless Sensor Networks / O. Durmaz Incel, A. Ghosh, B. Krishnamachari, K. Chintalapudi // *IEEE Transactions on Mobile Computing*. — 2012. — Vol. 11. — № 1. — P. 86–99.
205. Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds / I. Chillotti, N. Gama, M. Georgieva, M. Izabachène // *Advances in Cryptology – ASIACRYPT 2016* / Ed. by J. H. Cheon, T. Takagi. Hanoi, Vietnam. 2016. — Vol. 10031 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2016. — P. 3–33.
206. *Fatt Tay T., Chang C.-H.* A non-iterative multiple residue digit error detection and correction algorithm in RRNS // *IEEE Transactions on Computers*. — 2016. — Vol. 65. — № 2. — P. 396–408.
207. Fault-tolerant and information security in networks using multi-level redundant residue number system / P. Ali, M. Kambiz, S. S. Mohammad et al. // *Research Journal of Recent Sciences*. — 2014. — Vol. 3. — № 3 — P. 89–92.

208. *Ferrari S., Stengel R. F.* Smooth function approximation using neural networks // *IEEE Transactions on Neural Networks*. — 2005. — Vol. 16. — № 1. — P. 24–38.
209. Fully Homomorphic Encryption over the Integers / M. van Dijk, C. Gentry, S. Halevi, V. Vaikuntanathan // *Advances in Cryptology – EUROCRYPT 2010* / Ed. by H. Gilbert. French Riviera. — Vol. 6110 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. — P. 24–43.
210. *Funahashi K.-I.* On the approximate realization of continuous mappings by neural networks // *Neural Networks*. — 1989. — Vol. 2. — № 3. — P. 183–192.
211. *Gage D.* Nirvanix Files for Chapter 11 Bankruptcy. — 2013. [Электронный ресурс]. — Режим доступа: <https://blogs.wsj.com/venturecapital/2013/10/01/nirvanix-files-for-chapter-11-bankruptcy/>, свободный. — (дата обращения: 17.10.2021).
212. *Galbraith S. D.* Elliptic Curve Paillier Schemes // *Journal of Cryptology*. — 2002. — Vol. 15. — № 2. — P. 129–138. URL: <https://doi.org/10.1007/s00145-001-0015-6> — (дата обращения: 17.10.2021).
213. *Gentry C.* A fully homomorphic encryption scheme. — Stanford University, 2009. [Электронный ресурс]. — Режим доступа: <https://www.proquest.com/openview/93369e65682e50979432340f1fdae44e/1?pq-origsite=gscholar&cbl=18750>, свободный. — (дата обращения: 17.10.2021).
214. *Gentry C.* Computing arbitrary functions of encrypted data // *Communications of the ACM*. — 2010. — Vol. 53. — № 3. — P. 97–105.
215. *Gentry C., Halevi S.* Implementing Gentry’s Fully-Homomorphic Encryption Scheme // *Advances in Cryptology – EUROCRYPT 2011* / Ed. by Kenneth G. Paterson. Tallinn, Estonia. 2011. — Vol. 6632 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. — P. 129–148.
216. *Gentry C., Sahai A., Waters B.* Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based // Ad-

- vances in Cryptology – CRYPTO 2013 / Ed. by R. Canetti, J. A. Garay. Santa Barbara, CA, USA. 2013. — Vol. 8042 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. — P. 75–92.
217. *Ghemawat S., Gbioff H., Leung S.-T.* The Google file system // Proceedings of the nineteenth ACM symposium on Operating systems principles (SOSP). Bolton Landing NY USA. — 2003. — P. 29–43.
218. *Gjøsteen K.* Subgroup membership problems and public key cryptosystems. — 2004.[Электронный ресурс]. — Режим доступа: <https://core.ac.uk/download/pdf/30862502.pdf>, свободный. — (дата обращения: 17.10.2021).
219. Globally distributed content delivery / J. Dilley, B. Maggs, J. Parikh et al. // *IEEE Internet Computing*. — 2002. — Vol. 6. — № 5. — P. 50–58.
220. *Goh V. T., Siddiqi M. U.* Multiple error detection and correction based on redundant residue number systems // *IEEE Transactions on Communications*. — 2008. — Vol. 56. — № 3. — P. 325–330.
221. *Goldreich O.* A uniform-complexity treatment of encryption and zero-knowledge // *Journal of Cryptology*. — 1993. — Vol. 6. — № 1. — P. 21–53. URL: <https://doi.org/10.1007/BF02620230> — (дата обращения: 17.10.2021).
222. *Goldwasser S., Micali S., Tong P.* Why and how to establish a private code on a public network // Probabilistic Encryption & How to Play Mental Poker Keeping Secret All Partial Information. — 23rd Annual Symposium on Foundations of Computer Science (STOC '82). — New York, NY, USA: Association for Computing Machinery, 1982. — P. 365–377. URL: <https://doi.org/10.1145/800070.802212> — (дата обращения: 17.10.2021).
223. *Gomathisankaran M., Tyagi A., Namuduri K.* HORNS: A homomorphic encryption scheme for Cloud Computing using Residue Number System // 2011 45th Annual Conference on Information Sciences and Systems / IEEE. Baltimore, MD, USA. — 2011. — P. 1–5.
224. *Goparaju S., Fazeli A., Vardy A.* Minimum Storage Regenerating Codes for All Parameters // *IEEE Transactions on Information Theory*. — 2017. — Vol. 63. — № 10. — P. 6318–6328.

225. *Graham R. L., Knuth D. E., Patashnik O.* Concrete Mathematics. Addison-Wesley Publishing Company, Reading, MA // A foundation for computer science. — 1994.
226. *Gregory R. T., Krishnamurthy E. V.* Methods and applications of error-free computation. — Springer Science & Business Media, 2012.
227. *Armknrecht F., Boyd C., Carr C. et al.* A Guide to Fully Homomorphic Encryption. — Cryptology ePrint Archive, Report 2015/1192. — 2015. URL: <https://ia.cr/2015/1192> – (дата обращения: 17.10.2021).
228. *Guillermín N.* A High Speed Coprocessor for Elliptic Curve Scalar Multiplications over F_p // International Workshop on Cryptographic Hardware and Embedded Systems (CHES) / Springer. Santa Barbara, USA. — Vol. 6225 of *Lecture Notes in Computer Science*. — 2010. — P. 48–64.
229. *Hahm N., Hong B. I.* An approximation by neural networks with a fixed weight // *Computers & Mathematics with Applications*. — 2004. — Vol. 47. — № 12. — P. 1897–1903.
230. *Halevi S., Polyakov Y., Shoup V.* An Improved RNS Variant of the BFV Homomorphic Encryption Scheme // Topics in Cryptology – CT-RSA 2019 / Ed. by M. Matsui. San Francisco, CA, USA. — Vol. 11405 of *Lecture Notes in Computer Science*. — Cham: Springer International Publishing, 2019. — P. 83–105.
231. *Hamming R. W.* Error detecting and error correcting codes // *The Bell System Technical Journal*. — 1950. — Vol. 29. — № 2. — P. 147–160.
232. *Han K., Ki D.* Better bootstrapping for approximate homomorphic encryption // Cryptographers' Track at the RSA Conference CT-RSA 2020 / Springer. San Francisco, CA, USA. 2020. — Vol. 12006 of *Lecture Notes in Computer Science*. 2020. — P. 364–390.
233. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds / T. Ristenpart, E. Tromer, H. Shacham, S. Savage // Proceedings of the 16th ACM conference on Computer and communications security (CCS). Chicago Illinois USA. — 2009. — P. 199–212.

234. *Hiasat A.* A Reverse Converter and Sign Detectors for an Extended RNS Five-Moduli Set // *IEEE Transactions on Circuits and Systems I: Regular Papers*. — 2017. — Vol. 64. — № 1. — P. 111–121.
235. *Hiromasa R., Abe M., Okamoto T.* Packing Messages and Optimizing Bootstrapping in GSW-FHE // *Public-Key Cryptography – PKC 2015* / Ed. by J. Katz. Gaithersburg, MD, USA. — Vol. 9020 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2015. — P. 699–715.
236. *Hodjat A., Verbaauwhede I.* Minimum area cost for a 30 to 70 Gbits/s AES processor // *IEEE Computer Society Annual Symposium on VLSI* / IEEE. Lafayette, LA, USA. — 2004. — P. 83–88.
237. *Hoffstein J., Pipher J., Silverman J. H.* NTRU: A ring-based public key cryptosystem (ANTS) // *Algorithmic Number Theory* / Ed. by Joe P. Buhler. Portland, Oregon, USA. 1998. — Vol. 1423 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. — P. 267–288.
238. Tech. Rep.: / M. Albrecht, M. Chase, H. Chen et al. — Toronto, Canada. [Электронный ресурс]. — Режим доступа: HomomorphicEncryption.org, свободный. — (дата обращения: 17.10.2021).
239. Homomorphic encryption for arithmetic of approximate numbers / J. H. Cheon, A. Kim, M. Kim, Y. Song // *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)* / Springer. Hong Kong, China. 2017. — Vol. 10624 of *Lecture Notes in Computer Science*. 2017. — P. 409–437.
240. *Hornik K., Stinchcombe M., White H.* Multilayer feedforward networks are universal approximators // *Neural networks*. — 1989. — Vol. 2. — № 5. — P. 359–366.
241. *Hornik K., Stinchcombe M., White H.* Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks // *Neural networks*. — 1990. — Vol. 3. — № 5. — P. 551–560.
242. HttpClient. — 2020. [Электронный ресурс]. — Режим доступа: <https://github.com/amcewen/HttpClient>, , свободный. — (дата обращения: 17.10.2021).

243. *Huang C. H.* A Fully Parallel Mixed-Radix Conversion Algorithm for Residue Number Applications // *IEEE Transactions on Computers*. — 1983. — Vol. C-32. — № 4. — P. 398–402.
244. *Hubbard D., Sutton M.* Top threats to cloud computing v 1.0. — 2010. [Электронный ресурс]. — Режим доступа: <https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf>, свободный. — (дата обращения: 17.10.2021).
245. *Hur J.* Improving Security and Efficiency in Attribute-Based Data Sharing // *IEEE Transactions on Knowledge and Data Engineering*. — 2013. — Vol. 25. — № 10. — P. 2271–2282.
246. *H. Kai.* Computer arithmetic principles, architecture, and design. — John Wiley & Sons New York, 1979.
247. Implementation and performance evaluation of RNS variants of the BFV homomorphic encryption scheme / A. Al Badawi, Y. Polyakov, K. M. M. Aung et al. // *IEEE Transactions on Emerging Topics in Computing*. — 2019. — Vol. 9. — № 2. — P. 941–956.
248. Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme / J. W. Bos, K. Lauter, J. Loftus, M. Naehrig // IMA International Conference on Cryptography and Coding (IMACC) / Ed. by M. Stam. Oxford, UK. 2013. — Vol. 8308 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. — P. 45–64.
249. Improved proxy re-encryption schemes with applications to secure distributed storage / G. Ateniese, K. Fu, M. Green, S. Hohenberger // *ACM Transactions on Information and System Security (TISSEC)*. — 2006. — Vol. 9. — № 1. — P. 1–30.
250. Improving fault tolerance in ad-hoc networks by using residue number system / A Barati, M Dehghan, A Movaghar, H Barati // *Journal of Applied Sciences*. — 2008. — Vol. 8. — № 18. — P. 3273–3278.
251. Improving the Efficiency of SVM Classification with FHE / J.-C. Bajard, P. Martins, L. Sousa, V. Zucca // *IEEE Transactions on Information Forensics and Security*. — 2019. — Vol. 15. — P. 1709–1722.

252. Is cloud storage ready? Performance comparison of representative IP-based storage systems / Z. Ou, M. Song, Z.-H. Hwang et al. // *Journal of Systems and Software*. — 2018. — Vol. 138. — P. 206–221.
253. *Ishai Y., Paskin A.* Evaluating Branching Programs on Encrypted Data // Theory of Cryptography Conference (TCC) / Ed. by S. P. Vadhan. Amsterdam, The Netherlands. 2007. — Vol. 4392 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. — P. 575–594.
254. *Isupov K.* An Algorithm for Magnitude Comparison in RNS based on Mixed-Radix Conversion II // *International Journal of Computer Applications*. — 2016. — Vol. 141. — № 5.
255. *Junghanns P., Fabian B., Ermakova T.* Engineering of secure multi-cloud storage // *Computers in Industry*. — 2016. — Vol. 83. — P. 108–120. URL: <https://www.sciencedirect.com/science/article/pii/S0166361516301749> – (дата обращения: 17.10.2021).
256. *Kawachi A., Tanaka K., Xagawa K.* Multi-bit Cryptosystems Based on Lattice Problems // Public Key Cryptography – PKC 2007 / Ed. by T. Okamoto, X. Wang. Beijing, China. 2007.— Vol. 4450 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. — P. 315–329.
257. *Khedr A., Gulak G., Vaikuntanathan V.* SHIELD: Scalable Homomorphic Implementation of Encrypted Data-Classifiers // *IEEE Transactions on Computers*. — 2016. — Vol. 65. — № 9. — P. 2848–2858.
258. *Kianpishch S., Jalili S., Charkari N. M.* Predicting job wait time in grid environment by applying machine learning methods on historical information // *International Journal of Grid and Distributed Computing*. — 2012. — Vol. 5. — № 3. — P. 11–22.
259. *Kipnis Aviad, Hibshoosh Eliphaz.* Efficient Methods for Practical Fully Homomorphic Symmetric-key Encryption, Randomization and Verification. // *IACR Cryptol. ePrint Arch.* — 2012. — Vol. 2012. — P. 637. URL: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.252.4135&rep=rep1&type=pdf> – (дата обращения: 17.10.2021).

260. Klein A. Backblaze hard drive stats for 2017. — 2018. [Электронный ресурс]. — Режим доступа: <https://www.backblaze.com/blog/hard-drive-stats-for-2017/>, свободный. — (дата обращения: 17.10.2021).
261. Korkine A., Zolotareff G. Sur un certain minimum // *Nouvelles annales de mathématiques: journal des candidats aux écoles polytechnique et normale*. — 1873. — Vol. 12. — P. 337–355.
262. Koyama K., Tsuruoka Y. Speeding up Elliptic Cryptosystems by Using a Signed Binary Window Method // 12th Annual International Cryptology Conference (CRYPTO) / Ed. by E. F. Brickell. Santa Barbara, California, USA. 1992. — Vol. 740 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1994. — P. 345–357.
263. Krawczyk H. Secret Sharing Made Short // 13th Annual International Cryptology Conference (CRYPTO) / Ed. by D. R. Stinson. Santa Barbara, California, USA. 1993. — Vol. 773 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1994. — P. 136–146.
264. Kupreev O., Strohschneider J., Khalimonenko A. Kaspersky DDOS Intelligence Report for Q3 SecureList. — 2016. [Электронный ресурс]. — Режим доступа: <https://securelist.com/kaspersky-ddos-intelligencereport-for-q3-2016/76464/>, свободный. — (дата обращения: 17.10.2021).
265. Lattigo: lattice-based cryptographic library in Go. — 2021. — Jul. [Электронный ресурс]. — Режим доступа: <http://github.com/ldsec/lattigo>, свободный. — (дата обращения: 17.10.2021).
266. Leavitt N. Will NoSQL databases live up to their promise? // *Computer*. — 2010. — Vol. 43. — № 2. — P. 12–14.
267. Lee B.-H., Dewi E. K., Wajdi M. F. Data security in cloud computing using AES under HEROKU cloud // 2018 27th Wireless and Optical Communication Conference (WOCC) / IEEE. Hualien, Taiwan. — 2018. — P. 1–5.
268. Li Yingjiu, Swarup V., Jajodia S. Fingerprinting relational databases: schemes and specialties // *IEEE Transactions on Dependable and Secure Computing*. — 2005. — Vol. 2. — № 1. — P. 34–45.

269. *Lidl R., Niederreiter H.* Finite fields. — № 20. — Cambridge university press, 1997.
270. *Lin H.-Y., Tzeng W.-G.* A secure erasure code-based cloud storage system with secure data forwarding // *IEEE Transactions on Parallel and Distributed Systems.* — 2012. — Vol. 23. — № 6. — P. 995–1003.
271. *Lin S.-J., Chung W.-H., Han Y. S.* Novel polynomial basis and its application to reed-solomon erasure codes // 2014 IEEE 55th annual symposium on foundations of computer science / IEEE. — 2014. — P. 316–325.
272. Locality and Availability in Distributed Storage / A. S. Rawat, D. S. Papailiopoulos, A. G. Dimakis, S. Vishwanath // *IEEE Transactions on Information Theory.* — 2016. — Vol. 62. — № 8. — P. 4481–4493.
273. Logistic regression model training based on the approximate homomorphic encryption / A. Kim, Y. Song, M. Kim et al. // *BMC Medical Genomics.* — 2018. — Vol. 11. — № 4. — P. 23–31.
274. *López-A. A., Tromer E., Vaikuntanathan V.* On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption // Proceedings of the forty-fourth annual ACM symposium on Theory of computing (STOC). New York, NY, USA. — 2012. — P. 1219–1234.
275. *Maiorov V., Meir R. S.* Approximation bounds for smooth functions in $C(\mathbb{R}/\text{sup } d/)$ by neural and mixture networks // *IEEE Transactions on Neural Networks.* — 1998. — Vol. 9. — № 5. — P. 969–978.
276. *Makovoz Y.* Uniform approximation by neural networks // *Journal of Approximation Theory.* — 1998. — Vol. 95. — № 2. — P. 215–228.
277. Managing multi-cloud systems with CloudMF / N. Ferry, F. Chauvel, A. Rossini et al. // Proceedings of the Second Nordic Symposium on Cloud Computing & Internet Technologies (NordiCloud). Oslo Norway.— 2013. — P. 38–45.
278. *Mather T., Kumaraswamy S., Latif S.* Cloud security and privacy: an enterprise perspective on risks and compliance. — "O'Reilly Media, Inc. 2009.

279. *Menezes A. J., Van Oorschot P. C., Vanstone S. A.* Handbook of applied cryptography. — CRC press, 2018. URL: <https://doi.org/10.1201/9780429466335> – (дата обращения: 17.10.2021).
280. *Mignotte M.* How to share a secret // Proceedings of the Workshop on Cryptography / Ed. by T. Beth. Santa Barbara, California, USA. 1993. —Springer. Burg Feuerstein, Germany. 1982. — Vol. 149 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1983. — 1982. — P. 371–375.
281. *Minelli M.* Fully homomorphic encryption for machine learning: Ph.D. thesis / PSL Research University. — 2018. [Электронный ресурс]. — Режим доступа: <https://tel.archives-ouvertes.fr/tel-02449018/>, свободный. — (дата обращения: 17.10.2021).
282. Minimax Approximation of Sign Function by Composite Polynomial for Homomorphic Comparison / E. Lee, J.-W. Lee, Y.-S. Kim, J.-S. No // *IEEE Transactions on Dependable and Secure Computing* — 2021. URL: <https://ieeexplore.ieee.org/abstract/document/9517029> – (дата обращения: 17.10.2021).
283. *Mohan PV A.* RNS to binary conversion using diagonal function and Pirlo and Impedovo monotonic function // *Circuits, Systems, and Signal Processing*. — 2016. — Vol. 35. — № 3. — P. 1063–1076.
284. *Mohite M. P., Ardhapurkar S. B.* Design and Implementation of a Cloud Based Computer Forensic Tool // 2015 Fifth International Conference on Communication Systems and Network Technologies. Gwalior, India.— 2015. — P. 1005–1009.
285. *Morelos-Zaragoza R. H.* The art of error correcting coding. — John Wiley & Sons, 2006. [Электронный ресурс]. — Режим доступа: <https://shinnytech.ezyro.com/wp-content/uploads/2020/09/art-oferrorcorrecting-coding-s.pdf?i=1>, свободный. — (дата обращения: 17.10.2021).
286. Multi cloud management for unified cloud services across cloud sites / T. Liu, Y. Katsuno, K. Sun et al. // 2011 IEEE International Conference on Cloud

- Computing and Intelligence Systems / IEEE. Beijing, China. — 2011. — P. 164–169.
287. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function / M. Leshno, V. Y. Lin, A. Pinkus, S. Schocken // *Neural networks*. — 1993. — Vol. 6. — № 6. — P. 861–867.
288. *Naccache D., Stern J.* A New Public-Key Cryptosystem // International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT). Konstanz, Germany. 1997. — Vol. 1233 of *Lecture Notes in Computer Science* — Springer, Berlin, Heidelberg, 1997. — P. 27–36. URL: https://doi.org/10.1007/3-540-69053-0_3 — (дата обращения: 17.10.2021).
289. *Naehrig M., Lauter K., Vaikuntanathan V.* Can Homomorphic Encryption Be Practical? // Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop. — (CCSW). — New York, NY, USA: Association for Computing Machinery, 2011. — P. 113–124. URL: <https://doi.org/10.1145/2046660.2046682> — (дата обращения: 17.10.2021).
290. *Nakatsukasa Y., Bai Z., Gygi F.* Optimizing Halley’s iteration for computing the matrix polar decomposition // *SIAM Journal on Matrix Analysis and Applications*. — 2010. — Vol. 31. — № 5. — P. 2700–2720.
291. *Neal C. M.* Petraeus: CIA Could Use Smart Household Appliances to Spy. — 2012. [Электронный ресурс]. — Режим доступа: <https://slate.com/technology/2012/03/smart-appliances-couldhelp-cia-spy-says-petraeus.html>, свободный. — (дата обращения: 17.10.2021).
292. Network coding for distributed storage systems / A. G. Dimakis, P. B. Godfrey, Y. Wu et al. // *IEEE Transactions on Information Theory*. — 2010. — Vol. 56. — № 9. — P. 4539–4551.
293. *Nishikawa N., Amano H., Iwai K.* Implementation of bitsliced AES encryption on CUDA-enabled GPU // International Conference on Network and System Security (NSS) / Ed. by Z. Yan, R. Molva, W. Mazurczyk, R. Kantola. Springer. Helsinki, Finland. 2017 — Vol. 10394 of *Lecture Notes in Computer Science*. 2017. — P. 273–287.

294. Numerical method for comparison on homomorphically encrypted numbers / J. H. Cheon, D. Kim, D. Kim et al. // International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT) / Ed. by S. Galbraith, S. Moriai. Springer. Kobe, Japan. 2019. — Vol. 11922 of *Lecture Notes in Computer Science*. 2019. — P. 415–445.
295. Numerically stable improved Chebyshev–Halley type schemes for matrix sign function / A. Cordero, F. Soleymani, J. R. Torregrosa, M. Z. Ullah // *Journal of Computational and Applied Mathematics*. — 2017. — Vol. 318. — P. 189–198.
296. Okamoto T., Uchiyama S. A new public-key cryptosystem as secure as factoring // Advances in Cryptology — EUROCRYPT'98 / Ed. by K. Nyberg. Espoo, Finland. 1998. — Vol. 1403 of *Lecture Notes in Computer Science*. 1998. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. — P. 308–318.
297. Omondi A. R., Premkumar A. B. Residue number systems: theory and implementation. — World Scientific, 2007. — Vol. 2. — P. 296.
298. On Technical Security Issues in Cloud Computing / M. Jensen, J. Schwenk, N. Gruschka, L. L. Iacono // 2009 IEEE International Conference on Cloud Computing. Bangalore, India. — 2009. — P. 109–116.
299. On data banks and privacy homomorphisms / R. L. Rivest, L. Adleman, M. L. Dertouzos et al. // *Foundations of secure computation*. — 1978. — Vol. 4. — № 11. — P. 169–180.
300. One secure data integrity verification scheme for cloud storage / Y. Fan, X. Lin, G. Tan et al. // *Future Generation Computer Systems*. — 2019. — Vol. 96. — P. 376–385.
301. Oram A. Peer-to-Peer: Harnessing the power of disruptive technologies. — "O'Reilly Media, Inc. 2001.
302. Ozsu M. T., Valduriez P. Distributed database systems: Where are we now? // *Computer*. — 1991. — Vol. 24. — № 8. — P. 68–78.
303. Paillier P. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes // Advances in Cryptology — EUROCRYPT '99 / Ed. by J. Stern. Prague, Czech Republic. 1999. — Vol. 1592 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1999. — P. 223–238.

304. *Park M., Oh H., Lee K.* Security risk measurement for information leakage in IoT-based smart homes from a situational awareness perspective // *Sensors*. — 2019. — Vol. 19. — № 9. — P. 2148.
305. *Patronik P., Piestrak S. J.* Design of Reverse Converters for General RNS Moduli Sets $\{2^k, 2^n - 1, 2^n + 1, 2^{n+1} - 1\}$ and $\{2^k, 2^n - 1, 2^n + 1, 2^{n-1} - 1\}$ (n even) // *IEEE Transactions on Circuits and Systems I: Regular Papers*. — 2014. — Vol. 61. — № 6. — P. 1687–1700.
306. *Patronik P., Piestrak S. J.* Design of Reverse Converters for the New RNS Moduli Set $\{2^n + 1, 2^n - 1, 2^n, 2^{n-1} + 1\}$ odd // *IEEE Transactions on Circuits and Systems I: Regular Papers*. — 2014. — Vol. 61. — № 12. — P. 3436–3449.
307. Performance and cost evaluation of an adaptive encryption architecture for cloud databases / L. Ferretti, F. Pierazzi, M. Colajanni, Mirco Marchetti // *IEEE Transactions on Cloud Computing*. — 2014. — Vol. 2. — № 2. — P. 143–155.
308. *Phatak D. S., Houston S. D.* New distributed algorithms for fast sign detection in residue number systems (RNS) // *Journal of Parallel and Distributed Computing*. — 2016. — Vol. 97. — P. 78–95. URL: <https://www.sciencedirect.com/science/article/pii/S0743731516300703> — (дата обращения: 17.10.2021).
309. *Piestrak S. J.* Design of high-speed residue-to-binary number system converter based on Chinese remainder theorem // *Proceedings 1994 IEEE International Conference on Computer Design: VLSI in Computers and Processors* / IEEE. Cambridge, MA, USA. — 1994. — P. 508–511.
310. *Piestrak S. J.* Design of residue generators and multioperand modular adders using carry-save adders // *IEEE Transactions on Computers*. — 1994. — Vol. 43. — № 1. — P. 68–77.
311. *Piestrak S. J.* A note on RNS architectures for the implementation of the diagonal function // *Information Processing Letters*. — 2015. — Vol. 115. — № 4. — P. 453–457.

312. *Pirlo G., Impedovo D.* A new class of monotone functions of the residue number system // *Int. J. Math. Models Methods Appl. Sci.* — 2013. — Vol. 7. — № 9. — P. 803–809.
313. *Player R.* Parameter selection in lattice-based cryptography: Ph.D. thesis / Information Security Group, Royal Holloway, University of London. — 2018. [Электронный ресурс]. — Режим доступа: <https://core.ac.uk/download/pdf/159157762.pdf>, свободный. — (дата обращения: 17.10.2021).
314. Predicting disk replacement towards reliable data centers / М. М. Botezatu, I. Giurgiu, J. Bogojeska, D. Wiesmann // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, California, USA. — 2016. — P. 39–48.
315. *Preneel B.* Perspectives on Lightweight Cryptography, ECRYPT. — 2010.
316. Privacy-Preserving Classification on Deep Neural Network. / H. Chabanne, A. de Wargny, J. Milgram et al. // *IACR Cryptol. ePrint Arch.* — 2017. — Vol. 2017. — P. 35. URL: <https://img.chainnews.com/paper/5ad2543f3bf1ca450efbdd94f2a0bd9a.pdf> — (дата обращения: 17.10.2021).
317. Privacy-Preserving Deep Learning via Additively Homomorphic Encryption / L. T. Phong, Y. Aono, T. Hayashi et al. // *IEEE Transactions on Information Forensics and Security.* — 2018. — Vol. 13. — № 5. — P. 1333–1345.
318. *Quisquater M., Preneel B., Vandewalle J.* On the Security of the Threshold Scheme Based on the Chinese Remainder Theorem // Public Key Cryptography (PKC) / Ed. by D. Naccache, P. Paillier. Paris, France. 2002. — Vol. 2274 of *Lecture Notes in Computer Science.* — Berlin, Heidelberg: Springer Berlin Heidelberg, 2002. — P. 199–210.
319. RAID: High-Performance, Reliable Secondary Storage / P. M. Chen, E. K. Lee, G. A. Gibson et al. // *ACM Computing Surveys* — 1994. — Vol. 26. — № 2. — P. 145–185. URL: <https://doi.org/10.1145/176979.176981> — (дата обращения: 17.10.2021).
320. RESIDENT: a reliable residue number system-based data transmission mechanism for wireless sensor networks / R. Ye, A. Boukerche, H. Wang et al. // *Wireless Networks.* — 2018. — Vol. 24. — № 2. — P. 597–610.

321. An RNS Implementation of an F_p Elliptic Curve Point Multiplier / D. M. Schinianakis, A. P. Fournaris, H. E. Michail et al. // *IEEE Transactions on Circuits and Systems I: Regular Papers*. — 2009. — Vol. 56. — № 6. — P. 1202–1213.
322. RNS architectures for the implementation of the diagonal function' / G. Dimauro, S. Impedovo, G. Pirlo, A. Salzo // *Information Processing Letters*. — 2000. — Vol. 73. — № 5-6. — P. 189–198.
323. *Remez E. Y.* Sur la détermination des polynômes d'approximation de degré donnée // *Comm. Soc. Math. Kharkov*. — 1934. — Vol. 10. — № 196. — P. 41–63.
324. Research and compare cloud providers and services. — 2015. [Электронный ресурс]. — Режим доступа: <https://cloudharmony.com>, свободный. — (дата обращения: 17.10.2021).
325. Residue Number Systems: A New Paradigm to Datapath Optimization for Low-Power and High-Performance Digital Signal Processing Applications / C.-H. Chang, A. S. Molahosseini, A. A. E. Zarandi, T. F. Tay // *IEEE Circuits and Systems Magazine*. — 2015. — Vol. 15. — № 4. — P. 26–44.
326. Residue-to-binary conversion by the "quotient function-/ G Dimauro, S. Impedovo, R. Modugno et al. // *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*. — 2003. — Vol. 50. — № 8. — P. 488–493.
327. Rethinking security in the era of cloud computing / J. Aikat, A. Akella, J. S. Chase et al. // *IEEE Security & Privacy*. — 2017. — Vol. 15. — № 3. — P. 60–69.
328. *Riteau P.* Building dynamic computing infrastructures over distributed clouds // 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum / IEEE. Anchorage, AK, USA. — 2011. — P. 2097–2100.
329. *Rivest R. L., Shamir A., Adleman L.* A Method for Obtaining Digital Signatures and Public-Key Cryptosystems // *Communications of the ACM*. — 1978. — Vol. 21. — № 2. — P. 120–126. URL: <https://doi.org/10.1145/359340.359342> — (дата обращения: 17.10.2021).

330. Rohloff K., Cousins D. B. A Scalable Implementation of Fully Homomorphic Encryption Built on NTRU // Financial Cryptography and Data Security (FC) / Ed. by Rainer Böhme, Michael Brenner, Tyler Moore, Matthew Smith. Christ Church, Barbados. 2014. — Vol. 8438 of *Lecture Notes in Computer Science*. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. — P. 221–234.
331. Sander T., Young A., Yung Moti. Non-interactive cryptocomputing for NC^1 // 40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039). New York, NY, USA— 1999. — P. 554–566.
332. Sarma S. E., Weis S. A., Engels D. W. RFID systems and security and privacy implications // International workshop on cryptographic hardware and embedded systems (CHES) / Ed. by B. S. Kaliski, K. Koç, C. Paar. Springer. Redwood Shores, CA, USA. 2002. — Vol. 2523 of *Lecture Notes in Computer Science*. 2002. — P. 454–469.
333. Schinianakis D., Stouraitis T. Multifunction residue architectures for cryptography // *IEEE Transactions on Circuits and Systems I: Regular Papers*. — 2014. — Vol. 61. — № 4. — P. 1156–1169.
334. S. Bruce. Applied cryptography: protocols, algorithms, and source code in C. — John Wiley & Sons, 2007. URL: <https://bib-pubdb1.desy.de/record/364233> – (дата обращения: 17.10.2021).
335. Secure Social Multimedia Big Data Sharing Using Scalable JFE in the TSHWT Domain / C. Ye, H. Ling, Z. Xiong et al. // *ACM Transactions on Multimedia Computing, Communications, and Applications*. — 2016. — Vol. 12. — № 4. — P. 1–23.
336. Secure clustered distributed storage against eavesdropping / B. Choi, J.-Y. Sohn, S. W. Yoon, J. Moon // *IEEE Transactions on Information Theory*. — 2019. — Vol. 65. — № 11. — P. 7646–7668.
337. Secure distributed adaptive bin packing algorithm for cloud storage / I. Mohiuddin, A. Almogren, M. Al. Qurishi et al. // *Future Generation Computer Systems*. — 2019. — Vol. 90. — P. 307–316. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X18304035> – (дата обращения: 17.10.2021).

338. Secure integration of IoT and Cloud Computing / C. Stergiou, K. E. Psannis, B.-G. Kim, B. Gupta // *Future Generation Computer Systems*. — 2018. — Vol. 78. — P. 964–975. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X1630694X> – (дата обращения: 17.10.2021).
339. Secure logistic regression based on homomorphic encryption: Design and evaluation / M. Kim, Y. Song, S. Wang et al. // *JMIR medical informatics*. — 2018. — Vol. 6. — № 2. — P. e19.
340. Security and privacy aspects of low-cost radio frequency identification systems / S. A. Weis, S. E. Sarma, R. L. Rivest, D. W. Engels // *Security in Pervasive Computing*. — Springer, 2004. — P. 201–212.
341. Security issues in cloud environments: a survey / D. A. B. Fernandes, L. F. B. Soares, J. V. Gomes et al. // *International Journal of Information Security*. — 2014. — Vol. 13. — № 2. — P. 113–170. URL: <https://doi.org/10.1007/s10207-013-0208-7> – (дата обращения: 17.10.2021).
342. Security issues in NoSQL databases / L. Okman, N. Gal-Oz, Y. Gonen et al. // 2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications / IEEE. Changsha, China.— 2011. — P. 541–547.
343. *Shamir A.* How to share a secret // *Communications of the ACM*. — 1979. — Vol. 22. — № 11. — P. 612–613.
344. *Sharwood S.* GitLab.com Melts Down After Wrong Directory Deleted, Backups Fail. — 2017.[Электронный ресурс]. — Режим доступа: https://www.theregister.co.uk/2017/02/01/gitlab_data_loss/, свободный. — (дата обращения: 17.10.2021).
345. *Skavantzios A., Abdallah M.* Implementation issues of the two-level residue number system with pairs of conjugate moduli // *IEEE Transactions on Signal Processing*. — 1999. — Vol. 47. — № 3. — P. 826–838.
346. *Soheili A. R., Toutounian F., Soleymani F.* A fast convergent numerical method for matrix sign function with application in SDEs // *Journal of Computational and Applied Mathematics*. — 2015. — Vol. 282. — P. 167–178.

347. SpyStorage: A highly reliable multi-cloud storage with secure and anonymous data sharing / P. Shen, W. Liu, Z. Wu et al. // 2017 International Conference on Networking, Architecture, and Storage (NAS) / IEEE. Shenzhen, China. — 2017. — P. 1–6.
348. Starfish: A Self-tuning System for Big Data Analytics. / H. Herodotou, H. Lim, G. Luo et al. // Biennial Conference on Innovative Data Systems Research (CIDR). Asilomar, California, USA. — Vol. 11. — 2011. — P. 261–272.
349. *Subashini S., Kavitha V.* A survey on security issues in service delivery models of cloud computing // *Journal of Network and Computer Applications*. — 2011. — Vol. 34. — № 1. — P. 1–11. URL: <https://www.sciencedirect.com/science/article/pii/S1084804510001281> – (дата обращения: 17.10.2021).
350. *Suzuki S.* Constructive function-approximation by three-layer artificial neural networks // *Neural Networks*. — 1998. — Vol. 11. — № 6. — P. 1049–1058.
351. *Szabo N. S., Tanaka R. I.* Residue arithmetic and its applications to computer technology. — New York: McGraw-Hill, 1967. — P. 236.
352. *Takabi H., Hesamifard E., Ghasemi M.* Privacy preserving multi-party machine learning with homomorphic encryption // 29th Annual Conference on Neural Information Processing Systems (NIPS). Barcelona, Spain.— 2016. URL: <https://inspire.cse.unt.edu/sites/default/files/17.pdf> – (дата обращения: 17.10.2021).
353. *Timarchi S., Navi K.* Efficient Class of Redundant Residue Number System // 2007 IEEE International Symposium on Intelligent Signal Processing. Alcalá de Henares, Spain. — 2007. — P. 1–6.
354. Top Threats to Cloud Computing: The Egregious Eleven / JM Brook, A Getsin, G Jensen et al. // *Cloud Security Alliance*. — 2019. [Электронный ресурс]. — Режим доступа: <https://cloudsecurityalliance.org/artifacts/topthreats-to-cloud-computing-egregious-eleven/>, свободный. — (дата обращения: 17.10.2021).
355. Top ten big data security and privacy challenges. /Mora A.C., Chen Y., Fuchs A. et al.//*Cloud Security Alliance* — 2012. [Электронный ресурс]. — Режим доступа: <https://www.isaca.org/Groups/>

- Professional-English/big-data/GroupDocuments/Big_Data_Top_Ten_v1.pdf, свободный. – (дата обращения: 17.10.2021).
356. Towards the AlexNet Moment for Homomorphic Encryption: HCNN, the First Homomorphic CNN on Encrypted Data With GPUs / A. A. Badawi, C. Jin, J. Lin et al. // *IEEE Transactions on Emerging Topics in Computing*. — 2021. — Vol. 9. — № 3. — P. 1330–1343.
357. Towards the AlexNet Moment for Homomorphic Encryption: HCNN, the First Homomorphic CNN on Encrypted Data with GPUs. / Badawi A. A., Chao J., Lin J. et al. // *Arxiv*. — 2020. [Электронный ресурс]. — Режим доступа: <https://arxiv.org/pdf/1811.00778.pdf>. свободный. — (дата обращения: 17.10.2021).
358. *Van Vu T.* Efficient implementations of the Chinese remainder theorem for sign detection and residue decoding // *IEEE Transactions on Computers*. — 1985. — Vol. 100. — № 7. — P. 646–651.
359. *Venugopal S., Buyya R., Ramamohanarao K.* A taxonomy of data grids for distributed data sharing, management, and processing // *ACM Computing Surveys (CSUR)*. — 2006. — Vol. 38. — № 1. — P. 3–es.
360. *Vouk M. A.* Cloud computing—issues, research and implementations // *Journal of computing and information technology*. — 2008. — Vol. 16. — № 4. — P. 235–246.
361. *Wagh S., Gupta D., Chandran N.* SecureNN: 3-Party Secure Computation for Neural Network Training // *Proceedings on Privacy Enhancing Technologies*. — 2019. — Vol. 2019. — № 3. — P. 26–49. URL: <https://doi.org/10.2478/popets-2019-0035> – (дата обращения: 17.10.2021).
362. *Wang F., Wang K., Li B.* LWE-Based FHE with Better Parameters // *Advances in Information and Computer Security (IWSEC)* / Ed. by Keisuke Tanaka, Yuji Suga. Nara, Japan. 2015. — Vol. 9241 of *Lecture Notes in Computer Science*. — Cham: Springer International Publishing, 2015. — P. 175–192.
363. *Wang X., Yu H.* How to Break MD5 and Other Hash Functions // 24th Annual International Conference on the Theory and Applications of Cryptographic

- Techniques (EUROCRYPT). Aarhus, Denmark. — Vol. 3494 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg. 2005. P. 19–35.
364. Wang Y. Residue-to-binary converters based on new Chinese remainder theorems // *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*. — 2000. — Vol. 47. — № 3. — P. 197–205.
365. Waser S., Flynn M. J. Introduction to arithmetic for digital systems designers. CBS College Publishing. — 1982. P. 215–222.
366. Xiao L., Bastani O., Yen I.-L. An Efficient Homomorphic Encryption Protocol for Multi-User Systems. // *IACR Cryptol. EPrint Arch.* — 2012. — Vol. 2012. — P. 193. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.231.1754&rep=rep1&type=pdf> — (дата обращения: 17.10.2021).
367. Xiao Z., Xiao Y. Security and Privacy in Cloud Computing // *IEEE Communications Surveys Tutorials*. — 2013. — Vol. 15. — № 2. — P. 843–859.
368. Yagisawa M. Fully homomorphic encryption without bootstrapping. // *IACR Cryptol. EPrint Arch.* — 2015. — Vol. 2015. — P. 474. URL: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.738.2089&rep=rep1&type=pdf> — (дата обращения: 17.10.2021).
369. Yagisawa M. Improved Fully Homomorphic Encryption with Composite Number Modulus. // *IACR Cryptol. EPrint Arch.* — 2016. — Vol. 2016. — P. 50. URL: <https://img.chainnews.com/paper/bd47354d44e84d8172e8f9688cfd9a.pdf> — (дата обращения: 17.10.2021).
370. Yang H., Lee J. Secure Distributed Computing With Straggling Servers Using Polynomial Codes // *IEEE Transactions on Information Forensics and Security*. — 2019. — Vol. 14. — № 1. — P. 141–150.
371. Yang H., Shin W., Lee J. Private Information Retrieval for Secure Distributed Storage Systems // *IEEE Transactions on Information Forensics and Security*. — 2018. — Vol. 13. — № 12. — P. 2953–2964.
372. Yao A. C. Protocols for secure computations // 23rd Annual Symposium on Foundations of Computer Science (SFCS). Chicago, IL, USA. — 1982. — P. 160–164.

373. Zhang D., Jullien G. A., Miller W. C. A neural-like network approach to finite ring computations // *IEEE Transactions on Circuits and Systems*. — 1990. — Vol. 37. — № 8. — P. 1048–1052.
374. Zhang Q., Yang L. T., Chen Z. Privacy Preserving Deep Computation Model on Cloud for Big Data Feature Learning // *IEEE Transactions on Computers*. — 2016. — Vol. 65. — № 5. — P. 1351–1362.
375. Zhu Z., Jiang R. A Secure Anti-Collusion Data Sharing Scheme for Dynamic Groups in the Cloud // *IEEE Transactions on Parallel and Distributed Systems*. — 2016. — Vol. 27. — № 1. — P. 40–50.
376. Z. Brakerski C. Gentry V. Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping // TCS '12 Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS). Cambridge, Massachusetts. — 2012. — P. 309–325. URL: <https://dl.acm.org/doi/abs/10.1145/2090236.2090262> – (дата обращения: 17.10.2021).
377. Zygmund A. Trigonometric series. — Cambridge university press, 2002. — Vol. 1. 781 p.
378. A symmetric cryptographic scheme for data integrity verification in cloud databases / L. Ferretti, M. Marchetti, M. Andreolini, M. Colajanni // *Information Sciences*. — 2018. — Vol. 422. — P. 497–515.

СПИСОК РИСУНКОВ

| | | |
|------|--|-----|
| 1.1 | Структурная схема работы Интернета вещей | 35 |
| 1.2 | Структуры доступа: а) пороговая структура доступа (одна доля на каждое хранилище); б) пороговая структура доступа (несколько коротких долей на каждое хранилище); в) взвешенная пороговая структура доступа (по одной доле разного размера на хранилище); г) взвешенная пороговая структура доступа (разное количество долей одинакового размера на хранилище) | 46 |
| 1.3 | Хронология изобретения гомоморфных кодов | 51 |
| 1.4 | Хронология разработки полностью гомоморфных кодов | 55 |
| 1.5 | Концепция гомоморфного кодирования | 56 |
| 2.1 | Вероятность безотказной работы хранилищ | 72 |
| 2.2 | а) Вероятность потери данных для схемы MRC-RRNS; б) Вероятность потери данных для схемы WA-MRC-RRNS; в) Отношение вероятностей потери данных при пессимистичном и оптимистичном сценариях для каждой из схем MRC-RRNS и WA-MRC-RRNS соответственно | 78 |
| 2.3 | Преимущество схемы WA-MRC-RRNS по сравнению с MRC-RRNS с точки зрения вероятности потери данных в пессимистичном сценарии | 80 |
| 2.4 | Скорость кодирования WA-AR-RRNS и AR-RRNS | 84 |
| 2.5 | Скорость декодирования WA-AR-RRNS и AR-RRNS | 84 |
| 2.6 | Скорость кодирования WA-MRC-RRNS и MRC-RRNS | 85 |
| 2.7 | Скорость декодирования WA-MRC-RRNS и MRC-RRNS | 85 |
| 2.8 | Отношение скорости кодирования WA-MRC-RRNS к скорости кодирования WA-AR-RRNS, AR-RRNS и MRC-RRNS | 86 |
| 2.9 | Отношение скорости декодирования WA-MRC-RRNS к скорости декодирования WA-AR-RRNS, AR-RRNS и MRC-RRNS | 86 |
| 2.10 | Зависимость вероятности потери данных/получения несанкционированного доступа от избыточности данных, при $W = 1$ и $L = 128$ бит | 91 |
| 2.11 | Избыточность схем AC-RNNS, Asmuth-Bloom и Mignotte при длине модулей $l = 32$ бита | 104 |

| | | |
|------|---|-----|
| 2.12 | Вероятность неавторизованного доступа коалиции из k злоумышленников для схем Mignotte, Asmuth-Bloom и AC-RNNS при длине модулей $l = 32$ бита | 105 |
| 3.1 | Общая схема сравнения чисел с использованием позиционной характеристики | 122 |
| 3.2 | Аппаратная реализация алгоритма сравнения чисел в RNS на основе диагональной функции | 129 |
| 3.3 | График $a)$ и линии уровня $b)$ функции ядра с $\bar{w}_1 = 1, \bar{w}_2 = 2$ для RNS с основаниями $p_1 = 5, p_2 = 7$ | 138 |
| 3.4 | Сравнение чисел в RNS с помощью монотонной функции ядра Акушского, где $\bar{w}_l > 0$ | 139 |
| 3.5 | График функции ядра с коэффициентами $\bar{w}_1 = -3$ и $\bar{w}_2 = 5$ для RNS с основаниями $p_1 = 5$ и $p_2 = 6$ | 139 |
| 3.6 | Схема n -операндного МОМА $\text{mod } P$ | 165 |
| 4.1 | Аппроксимация функции знака числа с использованием многочленов $f_n(x)$, где $n = 1, 2, 3, 4$ | 191 |
| 4.2 | Аппроксимация функции знака числа с использованием многочленов $g_n(x)$, где $n = 1, 2, 3, 4$ | 192 |
| 4.3 | Аппроксимация функции знака числа с использованием композиции многочленов $f_n(g_n(x))$ для $n = 1, 2, 3, 4$ | 192 |
| 4.4 | Полиномиальная аппроксимация функции знака гомоморфно закодированных чисел с фиксированной точностью современными методами | 194 |
| 4.5 | Множество возможных значений m_Q и M_Q | 216 |
| 4.6 | Аппроксимация функции знака числа с помощью НСПР | 239 |
| 4.7 | Аппроксимация функции знака числа, заданной на множестве точек $\forall i = \overline{1, 2001}: x_i = -1 + \frac{i-1}{1000}$ с помощью функции $\frac{2 \arctg(636620 \cdot x)}{\pi}$ | 241 |
| 4.8 | Аппроксимация функции $\arctg x$ с помощью НСПР заданной формулой (4.290) | 242 |
| 6.1 | 2L-RRNS кодирование | 288 |
| 6.2 | 2L-RRNS декодирование | 288 |
| 6.3 | Обнаружение ошибок в 2L-RRNS и 2Lbp-RRNS | 297 |
| 6.4 | Исправление ошибок в 2L-RRNS и 2Lbp-RRNS | 299 |

| | | |
|------|--|-----|
| 6.5 | Архитектура FRNN <i>a</i>) и ее символическое отображение <i>б</i>) | 303 |
| 6.6 | Архитектура DNN для декодирования из 1L-RRNS в двоичную систему счисления | 305 |
| 6.7 | Скорость кодирования для настроек (3, 4) на уровне 1 | 308 |
| 6.8 | Скорость декодирования для настроек (3, 4) на уровне 1 | 309 |
| 6.9 | Диаграмма изменения скорости кодирования для различных комбинаций настроек | 310 |
| 6.10 | Диаграмма изменения скорости декодирования для различных комбинаций настроек | 310 |
| 6.11 | Скорость загрузки при настройках (3, 4) уровня 1 | 312 |
| 6.12 | Скорость выгрузки при настройках (3, 4) уровня 1 | 312 |

СПИСОК ТАБЛИЦ

| | | |
|----|---|-----|
| 1 | Характеристики схем распределенного облачного хранения данных . | 41 |
| 2 | Обозначения используемые в схеме WA-RRNS | 68 |
| 3 | Вероятность отказа CSP (CloudHarmony [324]) | 71 |
| 4 | Вероятность потери данных при использовании схемы MRC-RRNS в течение года | 74 |
| 5 | Вероятность потери данных при использовании схемы WA-MRC-RRNS в течение года | 75 |
| 6 | Количество долей в схемах AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS | 77 |
| 7 | Вероятность потери данных при использовании схем MRC-RRNS и WA-MRC-RRNS (пессимистичный (песс.) и оптимистичный (оптим.) сценарии) в течение года | 79 |
| 8 | Простые 16-битные числа, используемые при моделировании структуры доступа на основе RRNS | 81 |
| 9 | Скорость кодирования (MB/s) AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS | 82 |
| 10 | Скорость декодирования (MB/s) AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS | 83 |
| 11 | Размеры долей AR-RRNS, MRC-RRNS, WA-AR-RRNS и WA-MRC-RRNS (в байтах) | 87 |
| 12 | Параметры MRC-RRNS и WA-MRC-RRNS | 91 |
| 13 | Вычисление полуинтервала для секретного ключа HORNS | 93 |
| 14 | Уточнение секретного ключа HORNS | 93 |
| 15 | Основные характеристики сравниваемых структур доступа | 106 |
| 16 | Вычисление значения функции $S(X)$ | 114 |
| 17 | Вычисление значения функции $S(Y)$ | 114 |
| 18 | Вычисление значения функции $S(X)$ | 119 |
| 19 | Вычисление значения функции $S(Y)$ | 119 |
| 20 | Параметры MOMA, использующихся в устройствах сравнения чисел в RNS | 157 |

| | | |
|----|--|-----|
| 21 | Размеры операндов МОМА, используемых при реализации устройств сравнения для различных наборов модулей RNS | 160 |
| 22 | Размеры операндов МОМА, используемых при реализации устройств сравнения для различных наборов модулей RNS (продолжение таблицы 21) | 161 |
| 23 | Минимальное количество ступеней CSA $\theta(n)$, достаточное для обработки n операндов МОМА | 164 |
| 24 | Выходные данные n -операндного дерева CSA [309] | 166 |
| 25 | Оценка сложности устройства сравнения для 6-модульной RNS $SP_{6,1} = \{5, 7, 9, 11, 13, 16\}$ | 167 |
| 26 | Значения многочлена $c(x, y)$ над \mathbb{Z}_5 | 188 |
| 27 | Основные характеристики методов гомоморфного сравнения чисел. Ошибка аппроксимации $\epsilon = 2^{-\alpha}$ | 194 |
| 28 | Вычисление $X = \left\lfloor \frac{X}{p_1 \cdot p_2} \right\rfloor$ | 261 |
| 29 | Вычисление ранга $r(\bar{X})$ | 262 |
| 30 | Значения $ P_i _{p_i} \cdot \left P_i^{-1} \cdot 2^N \right _{p_i}$ для $i = \overline{1, 4}$ и $N = \overline{4, 10}$ | 269 |
| 31 | Обозначения для схемы 2Lbp-RRNS | 284 |
| 32 | Обозначения параметров первого и второго уровня схемы 2Lbp-RRNS | 285 |
| 33 | Параметры схем 2L-RRNS (2Lbp-RRNS) | 298 |
| 34 | Средняя скорость кодирования и декодирования данных (Мб/с) | 299 |
| 35 | Скорость доступа к семи облачным сервисам (Мб/с) | 307 |

Приложение А

ПАТЕНТЫ



**ЕВРАЗИЙСКАЯ ПАТЕНТНАЯ ОРГАНИЗАЦИЯ
ЕВРАЗИЙСКОЕ ПАТЕНТНОЕ ВЕДОМСТВО**

ЕВРАЗИЙСКИЙ ПАТЕНТ



ЕВРАЗИЙСКИЙ ПАТЕНТ

№ 038389

Название изобретения:

**«УСТРОЙСТВО СРАВНЕНИЯ И ОПРЕДЕЛЕНИЯ ЗНАКА
ЧИСЕЛ, ПРЕДСТАВЛЕННЫХ В СИСТЕМЕ ОСТАТОЧНЫХ
КЛАССОВ»**

Патентовладелец (льцы):

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
"СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ" (RU)**

Изобретатель (и):

**Дерябин Максим Анатольевич, Бабенко Михаил Григорьевич,
Кучуков Виктор Андреевич, Назаров Антон Сергеевич,
Кучеров Николай Николаевич (RU)**

Заявка №: 202090736
Дата подачи заявки: 14 апреля 2020 г.
Дата выдачи патента: 20 августа 2021 г.

Настоящим удостоверяется, что евразийский патент выдан на изобретение с формулой, опубликованной в Бюллетене Евразийского патентного ведомства «Изобретения (евразийские заявки и патенты)» № 8 / 2021 год.

При уплате установленных годовых пошлин патент действует на территории государств - участников Евразийской патентной конвенции - Азербайджанской Республики, Кыргызской Республики, Республики Армения, Республики Беларусь, Республики Казахстан, Республики Таджикистан, Российской Федерации, Туркменистана.

ТЛЕВЛЕСОВА Сауле Январбековна
Президент Евразийского патентного ведомства



РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2483346

**УСТРОЙСТВО ДЛЯ ОБНАРУЖЕНИЯ ПЕРЕПОЛНЕНИЯ
ДИНАМИЧЕСКОГО ДИАПАЗОНА, ОПРЕДЕЛЕНИЯ
ОШИБКИ И ЛОКАЛИЗАЦИИ НЕИСПРАВНОСТИ
ВЫЧИСЛИТЕЛЬНОГО КАНАЛА В ЭВМ,
ФУНКЦИОНИРУЮЩИХ В СИСТЕМЕ ОСТАТОЧНЫХ
КЛАССОВ**

Патентообладатель(ли): *Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет" (RU)*

Автор(ы): *см. на обороте*

Заявка № 2011145755

Приоритет изобретения **10 ноября 2011 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **27 мая 2013 г.**

Срок действия патента истекает **10 ноября 2031 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Б.П. Симонов



Автор(ы): *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Ляхов Павел Алексеевич (RU), Лавриненко Ирина Николаевна (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2503992

**УСТРОЙСТВО ДЛЯ СРАВНЕНИЯ ЧИСЕЛ,
ПРЕДСТАВЛЕННЫХ В СИСТЕМЕ ОСТАТОЧНЫХ
КЛАССОВ**

Патентообладатель(ли): *Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет" (RU)*

Автор(ы): *см. на обороте*

Заявка № 2011139397

Приоритет изобретения **27 сентября 2011 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **10 января 2014 г.**

Срок действия патента истекает **27 сентября 2031 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Б.П. Симонов



Автор(ы): *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Ляхов Павел Алексеевич (RU), Лавриненко Ирина Николаевна (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2503995

УСТРОЙСТВО ДЛЯ ОПРЕДЕЛЕНИЯ ЗНАКА
МОДУЛЯРНОГО ЧИСЛА

Патентообладатель(и): *Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет" (RU)*

Автор(ы): *см. на обороте*

Заявка № 2011139278

Приоритет изобретения 26 сентября 2011 г.

Зарегистрировано в Государственном реестре изобретений Российской Федерации 10 января 2014 г.

Срок действия патента истекает 26 сентября 2031 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Б.И. Симонов



Автор(ы): *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Ляхов Павел Алексеевич (RU), Лавриненко Ирина Николаевна (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2559771

**УСТРОЙСТВО ДЛЯ ОСНОВНОГО ДЕЛЕНИЯ
МОДУЛЯРНЫХ ЧИСЕЛ**

Патентообладатель(ли): *Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет" (RU)*

Автор(ы): *см. на обороте*

Заявка № 2013148505

Приоритет изобретения **30 октября 2013 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **15 июля 2015 г.**

Срок действия патента истекает **30 октября 2033 г.**

Врио руководителя Федеральной службы
по интеллектуальной собственности

Л.Л. Кирий



Автор(ы): *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Ляхов Павел Алексеевич (RU), Лавриненко Ирина Николаевна (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2559772

**УСТРОЙСТВО ДЛЯ ОСНОВНОГО ДЕЛЕНИЯ
МОДУЛЯРНЫХ ЧИСЕЛ В ФОРМАТЕ СИСТЕМЫ
ОСТАТОЧНЫХ КЛАССОВ**

Патентообладатель(и): *Федеральное государственное автономное образовательное учреждение высшего профессионального образования "Северо-Кавказский федеральный университет" (RU)*

Автор(ы): *см. на обороте*

Заявка № 2013149446

Приоритет изобретения **06 ноября 2013 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **15 июля 2015 г.**

Срок действия патента истекает **06 ноября 2033 г.**

Врио руководителя Федеральной службы
по интеллектуальной собственности

Л.Л. Кирий



Автор(ы): *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Ляхов Павел Алексеевич (RU), Лавриненко Ирина Николаевна (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**ПАТЕНТ**

НА ИЗОБРЕТЕНИЕ

№ 2628179

Устройство деления модулярных чисел

Патентообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет" (RU)*

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Кучуков Виктор Андреевич (RU), Дерябин Максим Анатольевич (RU), Лавриненко Ирина Николаевна (RU), Лавриненко Антон Викторович (RU)*

Заявка № 2016146626

Приоритет изобретения 28 ноября 2016 г.

Дата государственной регистрации в

Государственном реестре изобретений

Российской Федерации 15 августа 2017 г.

Срок действия исключительного права

на изобретение истекает 28 ноября 2036 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2653257

Устройство обнаружения и коррекции ошибки модулярного кода

Патентообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет" (RU)*

Авторы: *Червяков Николай Иванович (RU), Кучуков Виктор Андреевич (RU), Бабенко Михаил Григорьевич (RU), Кучукова Наталья Николаевна (RU)*

Заявка № 2017126350

Приоритет изобретения 21 июля 2017 г.

Дата государственной регистрации в

Государственном реестре изобретений

Российской Федерации 07 мая 2018 г.

Срок действия исключительного права

на изобретение истекает 21 июля 2037 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**ПАТЕНТ**

НА ИЗОБРЕТЕНИЕ

№ 2652450

**Устройство вычисления модулярного произведения
Монтгомери**

Патентообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования "Северо-
Кавказский федеральный университет" (RU)*

Авторы: *Червяков Николай Иванович (RU), Коляда Андрей
Алексеевич (BY), Кучуков Виктор Андреевич (RU), Бабенко
Михаил Григорьевич (RU)*

Заявка № 2017129526

Приоритет изобретения 18 августа 2017 г.

Дата государственной регистрации в

Государственном реестре изобретений

Российской Федерации 26 апреля 2018 г.

Срок действия исключительного права

на изобретение истекает 18 августа 2037 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**ПАТЕНТ**

НА ИЗОБРЕТЕНИЕ

№ 2744815

Устройство для перевода чисел из системы остаточных классов и расширения оснований

Патентообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования "Северо-Кавказский федеральный университет" (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучуков Виктор Андреевич (RU), Черных Андрей Николаевич (RU), Кучеров Николай Николаевич (RU)*

Заявка № **2020120649**

Приоритет изобретения **22 июня 2020 г.**
 Дата государственной регистрации
 в Государственном реестре изобретений
 Российской Федерации **16 марта 2021 г.**
 Срок действия исключительного права
 на изобретение истекает **22 июня 2040 г.**

Руководитель Федеральной службы
 по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2751992

**Устройство сравнения чисел, представленных в системе
остаточных классов**

Патентообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
"Северо-Кавказский федеральный университет" (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучуков Виктор
Андреевич (RU)*

Заявка № 2020134772

Приоритет изобретения **22 октября 2020 г.**

Дата государственной регистрации
в Государственном реестре изобретений

Российской Федерации **21 июля 2021 г.**

Срок действия исключительного права
на изобретение истекает **22 октября 2040 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г. П. Иалиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2747371

**Устройство определения знака числа, представленного в
системе остаточных классов**

Патентообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
"Северо-Кавказский федеральный университет" (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучуков Виктор
Андреевич (RU)*

Заявка № 2020134778

Приоритет изобретения **22 октября 2020 г.**

Дата государственной регистрации
в Государственном реестре изобретений
Российской Федерации **04 мая 2021 г.**

Срок действия исключительного права
на изобретение истекает **22 октября 2040 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

Г.П. Ивлиев Г.П. Ивлиев



Приложение Б

СВИДЕТЕЛЬСТВА О ГОСУДАРСТВЕННОЙ РЕГИСТРАЦИИ
ПРОГРАММ ДЛЯ ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012610503

«Среда моделирования вычислений в системе остаточных классов на основе приближенных методов» («СМВСОКОПМ»)

Правообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ставропольский государственный университет» (RU)*

Автор(ы): *Червяков Николай Иванович, Бабенко Михаил Григорьевич, Ляхов Павел Алексеевич (RU)*

Заявка № 2011618721

Дата поступления 17 ноября 2011 г.

Зарегистрировано в Реестре программ для ЭВМ

10 января 2012 г.

Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам



Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016612432

Ускоренный метод вычисления остатка от деления с использованием распределенной арифметики

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Дерябин Максим Анатольевич (RU), Назаров Антон Сергеевич (RU), Лавриченко Антон Викторович (RU)*

Заявка № 2015663065

Дата поступления 29 декабря 2015 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 26 февраля 2016 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016618312

Программа управления устройством перевода чисел из системы остаточных классов в позиционную систему счисления на основе перевода в обобщенную позиционную систему счисления

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *см. на обороте*




Заявка № 2016615527

Дата поступления 30 мая 2016 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 26 июля 2016 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Иалиев

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Дерябин Максим Анатольевич (RU), Назаров Антон Сергеевич (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016618315

Программа управления устройством перевода чисел из системы остаточных классов в позиционную систему счисления на основе Китайской теоремы об остатках с дробными числами

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *см. на обороте*




Заявка № 2016615529

Дата поступления 30 мая 2016 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 26 июля 2016 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Ивлиев

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Дерябин Максим Анатольевич (RU), Назаров Антон Сергеевич (RU), Лавриненко Антон Викторович (RU)*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2017615372

Программа моделирования гомоморфного шифрования в
облачных вычислениях

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил
Григорьевич (RU), Кучеров Николай Николаевич (RU),
Черногорова Юлия Викторовна (RU)*



Заявка № 2017612222

Дата поступления 20 марта 2017 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 15 мая 2017 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018612690

**Среда моделирования алгоритмов цифровой фильтрации
изображений**

Правообладатель: **Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)**

Авторы: **Червяков Николай Иванович (RU), Бабенко Михаил
Григорьевич (RU), Кучуков Виктор Андреевич (RU), Гудиева
Наталья Григорьевна (RU)**



Заявка № **2017663489**

Дата поступления **25 декабря 2017 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **21 февраля 2018 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018612694

Модуль оценки рисков безопасности облачных, краевых и туманных вычислений в условиях вычислительной неопределенности

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *Червяков Николай Иванович (RU), Бабенко Михаил Григорьевич (RU), Черных Андрей Николаевич (RU), Кучукова Наталья Николаевна (RU), Гудиева Наталья Григорьевна (RU)*

Заявка № 2017663493

Дата поступления 25 декабря 2017 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 21 февраля 2018 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019610808

**Модуль кодирования и декодирования данных в системе
остаточных классов**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Червяков Николай
Иванович (RU), Черных Андрей Николаевич (RU), Кучуков Виктор
Андреевич (RU), Кучеров Николай Николаевич (RU), Кучукова
Екатерина Андреевна (RU), Аль-Гальда Сафват Чиад (IQ)*



Заявка № 2018665334

Дата поступления 28 декабря 2018 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 18 января 2019 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Иалиев Г.П. Иалиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019611375

Распределенная система надежного хранения и обработки
данных в мультиоблачной среде

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Червяков Николай
Иванович (RU), Черных Андрей Николаевич (RU), Кучеров Николай
Николаевич (RU), Кучуков Виктор Андреевич (RU), Кучукова
Екатерина Андреевна (RU), Аль-Гальда Сафват Чиад (IQ)*



Заявка № 2018665335

Дата поступления 28 декабря 2018 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 24 января 2019 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019619806

Программный модуль деления с остатком чисел большой
разрядности

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Петров Валентин Владимирович (RU), Бабенко Михаил
Григорьевич (RU), Дерябин Максим Анатольевич (RU), Кучукова
Екатерина Андреевна (RU)*



Заявка № 2019618843

Дата поступления 19 июля 2019 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 24 июля 2019 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019619899

**Программный комплекс моделирования методов
распределения информации**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Червяков Николай Иванович (RU), Дерябин Максим
Анатольевич (RU), Джурабаев Анвар Эркин угли (UZ), Бабенко
Михаил Григорьевич (RU), Редванов Азиз Салимович (RU)*

Заявка № 2019618665

Дата поступления 17 июля 2019 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 26 июля 2019 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019661394

**«Программа модулярного нейросетевого кодирования
данных»**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU),
Кучукова Екатерина Андреевна (RU)*

Заявка № 2019618837

Дата поступления 19 июля 2019 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 28 августа 2019 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019661480

**«Программа модульного нейросетевого декодирования
данных»**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU),
Кучукова Екатерина Андреевна (RU)*

Заявка № 2019618875

Дата поступления 19 июля 2019 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 02 сентября 2019 г.

*Руководитель Федеральной службы
по интеллектуальной собственности*

Г.П. Ивлиев Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019663760

**Программный модуль эффективной реализации
арифметических операций в конечном поле**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Ващенко Ирина
Сергеевна (RU), Назаров Антон Сергеевич (RU), Кучукова
Екатерина Андреевна (RU)*



Заявка № **2019662609**

Дата поступления **16 октября 2019 г.**

Дата государственной регистрации
в Реестре программ для ЭВМ **23 октября 2019 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Излиев Г.П. Излиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019663930

Программный модуль сравнения чисел в системе
остаточных классов

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Ващенко Ирина
Сергеевна (RU), Кучукова Екатерина Андреевна (RU)*

Заявка № 2019662611

Дата поступления 16 октября 2019 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 25 октября 2019 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Излиев Г.П. Излиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020610041

**Программа избыточного кодирования и декодирования
модулярного кода**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Кучуков Виктор Андреевич (RU),
Бабенко Михаил Григорьевич (RU)*



Заявка № **2019666586**

Дата поступления **17 декабря 2019 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **09 января 2020 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020618967

Программа подготовки файлов для распределенного хранения данных в облаках

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *Кучеров Николай Николаевич (RU), Бабенко Михаил Григорьевич (RU), Кучуков Виктор Андреевич (RU), Ващенко Ирина Сергеевна (RU)*



Заявка № 2020618381

Дата поступления 03 августа 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 10 августа 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020619140

**Программа восстановления полученных данных при
распределенном хранении данных в облаках**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Кучеров Николай Николаевич (RU), Бабенко Михаил
Григорьевич (RU), Кучуков Виктор Андреевич (RU), Ващенко
Ирина Сергеевна (RU)*



Заявка № 2020618376

Дата поступления 03 августа 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 12 августа 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020660256

Программный модуль декодирования данных с
использованием минимально избыточного
модулярного кода

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучеров Николай
Николаевич (RU), Кучукова Екатерина Андреевна (RU)*



Заявка № 2020619581

Дата поступления 28 августа 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 01 сентября 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020660257

Система моделирования исправления ошибок в
модульном коде

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучуков Виктор
Андреевич (RU), Ващенко Ирина Сергеевна (RU)*



Заявка № 2020619583

Дата поступления 28 августа 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 01 сентября 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Излиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**СВИДЕТЕЛЬСТВО**

о государственной регистрации программы для ЭВМ

№ 2020660392

Программный модуль сбора данных о технических характеристиках облачных провайдеров в реальном режиме времени

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Сотникова Наталья Алексеевна (RU), Кучукова Екатерина Андреевна (RU)*



Заявка № 2020619579

Дата поступления 28 августа 2020 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 03 сентября 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**СВИДЕТЕЛЬСТВО**

о государственной регистрации программы для ЭВМ

№ 2020660531**Модуль выбора оснований системы остаточных классов
для оптимизации минимально избыточного кода****Правообладатель: Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)****Авторы: Бабенко Михаил Григорьевич (RU), Кучуков Виктор
Андреевич (RU), Ващенко Ирина Сергеевна (RU)**Заявка № **2020619552**Дата поступления **28 августа 2020 г.**Дата государственной регистрации
в Реестре программ для ЭВМ **04 сентября 2020 г.***Руководитель Федеральной службы
по интеллектуальной собственности* **Г.П. Излиев**

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**СВИДЕТЕЛЬСТВО**

о государственной регистрации программы для ЭВМ

№ 2020660532**Программный модуль управления адаптивной
безопасностью в мультиоблачной среде**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Кучеров Николай
Николаевич (RU), Сотникова Наталья Алексеевна (RU)*

Заявка № **2020619548**Дата поступления **28 августа 2020 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **04 сентября 2020 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

Г.П. Излиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020665103

Программа управления устройством коррекции
однократных ошибок на основе перехода к обобщенной
позиционной системе счисления

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Назаров Антон Сергеевич (RU), Бабенко Михаил
Григорьевич (RU), Кучуков Виктор Андреевич (RU), Кучеров
Николай Николаевич (RU)*

Заявка № 2020663618

Дата поступления 09 ноября 2020 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 23 ноября 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020665416

Программа управления устройством коррекции
однократных ошибок на основе Китайской теоремы об
остатках с дробными величинами

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Назаров Антон Сергеевич (RU), Бабенко Михаил
Григорьевич (RU), Кучуков Виктор Андреевич (RU), Кучеров
Николай Николаевич (RU)*



Заявка № 2020663608

Дата поступления 09 ноября 2020 г.

Дата государственной регистрации
в Реестре программ для ЭВМ 26 ноября 2020 г.

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ
№ 2021616029

**Программа для умножения зашифрованных матриц с
использованием СККС схемы**

Правообладатель: *Федеральное государственное автономное
образовательное учреждение высшего образования
«Северо-Кавказский федеральный университет» (RU)*

Авторы: *Бабенко Михаил Григорьевич (RU), Голиблевская
Елена Игоревна (RU), Ширяев Егор Михайлович (RU)*

Заявка № **2021614751**

Дата поступления **07 апреля 2021 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **15 апреля 2021 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

Г.П. Ивлиев

