DOI: 10.15514/ISPRAS-2026-38(3)-2

Оптимизация алгоритма расширения оснований в модулярном коде для гомоморфных шифров

[1,2] *М.Г. Бабенко, ORCID: 0000-0001-7066-0061<mgbabenko@ncfu.ru>*
[1] *А.А. Синицын, ORCID: 0009-0002-7740-4217 <antony@email.su>*
[3] *М.А. Дерябин, ORCID: 0000-0002-6761-3667 <maxim.deryabin@gmail.com>*

[1] *Институт системного программирования им. В.П. Иванникова РАН, Россия, 109004, г. Москва, ул. А. Солженицына, д. 25.*

[2] *Северо-Кавказский федеральный университет, Россия, 355017, г. Ставрополь, ул. Пушкина, 1.*

[3] *Институт передовых технологий Samsung, Республика Корея, Сувон 16678.*

Аннотация. Гомоморфное шифрование позволяет обрабатывать данные без их расшифровки в удаленном пространстве (таком как облачной системы обработки данных). Несмотря на то, что это одна из ключевых перспективных технологий защиты персональных данных пользователей в современном мире, она сталкивается с проблемой низкой производительности. Для повышения скорости обработки данных большинство основных систем гомоморфного шифрования используют модулярный код через систему остаточных классов (СОК) как арифметическую основу для высокопроизводительных вычислений. Среди сложных операций, необходимых для гомоморфных шифров, особое место занимает операция расширения системы оснований СОК. Наибольшую популярность имеет ранее предложенный подход к этой операции, основанный на вычислении приближенного ранга числа в СОК использований числа с плавающей запятой. Для оптимизации алгоритма расширения оснований СОК ранее были получены оценки точности, с которой необходимо выполнять вычисления приближенного ранга числа, причем авторы использовали классическую теорию погрешности, не учитывающую свойства модулярного кода. Мы предлагаем теоретическое исследование позволявшее оценить точность вычисления приближенного ранга числа. Доказанная теорема позволяет уменьшить длину операндов более, чем в 3 раза по сравнению с оценками других авторов. Результаты моделирования показывают, что ранее предложенная оптимизация алгоритма позволяет повысить скорость алгоритма масштабирования чисел в системе остаточных классов в среднем на 46.15%.

Ключевые слова: система остаточных классов; расширения оснований в системы остаточных классов; Китайская теорема об остатках; ранг числа; аппроксимация ранга числа.

Для цитирования: Бабенко М.Г., Синицын А. А., Дерябин М.А. Оптимизация алгоритма расширения оснований в модулярном коде для гомоморфных шифров. Труды ИСП РАН, том 38, вып. 3, часть 1, 2026 г., стр. 33–44. DOI: 10.15514/ISPRAS–2026–38(3)–2.

Благодарности: Исследование выполнено при поддержки гранта Российского научного фонда № 25-71-30007.

Optimization of RNS Base Extension for Homomorphic Encryption Schemes

[1,2] *M.G. Babenko, ORCID: 0000-0001-7066-0061<mgbabenko@ncfu.ru>*
[1] *A.A. Sinitsyn, ORCID: 0009-0002-7740-4217 <antony@email.su>*
[3] *M.A. Deryabin, ORCID:* 0000-0002-6761-*<maxim.deryabin@gmail.com>*

[1] *Ivannikov Institute for System Programming of the Russian Academy of Sciences, 25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.*

[2] *North-Caucasus Federal University, 1, Pushkin Street, Stavropol 355017, Russia.*

[3] *Samsung Advanced Institute of Technology, Suwon 16678, Republic of Korea.*

Abstract. Homomorphic encryption allows data to be processed without decrypting it in a remote space (such as a cloud data processing system). Despite the fact that it is one of the key promising technologies for protecting personal data of users in the modern world, it faces the problem of low productivity. To improve data processing speed, most major homomorphic encryption systems use modular code through a residue number system (RNS) as an arithmetic basis for high performance computing. Among the complex operations required for homomorphic ciphers, a special place is occupied by the operation of expanding the base system of the RNS. Most popular is the previously proposed approach to this operation, based on calculating the approximate rank of a number in the RNS of floating point uses. To optimize the algorithm for expanding the bases of the RNS, estimates of the accuracy with which it is necessary to calculate the approximate rank of the number were previously obtained, and the authors used the classical theory of error, which does not consider the properties of the modular code. We propose a theoretical study that made it possible to assess the accuracy of calculating the approximate rank of a number. The proven theorem allows you to reduce the length of operands by more than 3 times compared to estimates of other authors. The simulation results show that the previously proposed algorithm optimization can increase the speed of the number scaling algorithm in the residue number system by an average of 46.15%.

Keywords: residue number system; RNS base extension; Chinese Remainder Theorem; rank of a number; approximation of the rank of a number.

For citation: Babenko M.G., Sinitsyn A.A., Deryabin M.A. Optimization of RNS Base Extension for Homomorphic Encryption Schemes. Trudy ISP RAN/Proc. ISP RAS, vol. 38, issue 3, part 1, 2026, pp. 33-44 (in Russian). DOI: 10.15514/ISPRAS-2026-38(3)-2.

Acknowledgements. The research was supported by the Russian Science Foundation Grant No 25-71-30007.

1. Введение

Гомоморфное шифрование это способ шифрования позволяющий выполнять арифметические операции сложения и умножения с зашифрованными числами. Например, гомоморфный шифр Brakerski-Fan-Vercautern (BFV) ориентирован на операции с целыми числами и в базовом виде позволяет выполнять ограниченное число точных целочисленных арифметических сложений и умножений [1-4]. Для расширения количества допустимых умножений используется вычислительно сложный алгоритм бутстраппинга (Bootstrapping) [5]. Бутстраппинг позволяет обновлять параметры схемы шифрования и сбрасывать ограничение на количество умножений.

Использование гомоморфных шифров неизбежно увеличивает объём хранимых, передаваемых и обрабатываемых данных накладывая дополнительные накладные расходы на системы обработки данных. Операции, необходимые для гомоморфных шифров, являются сложным сочетанием множества алгоритмов и математических систем, включая теоретико-числовое преобразование и систему остаточных классов. Данная работа посвящена анализу отдельного алгоритма, который используется как в базовой схеме BFV с использованием

системы остаточных классов, так и ее производных схем, например CKKS (Cheon-Kim-Kimn-Song) [6]. В свою очередь, схема BGV (Brakerski–Gentry–Vaikuntanathan) [1, 7-8], которая является альтернативой для BFV при обработки целых чисел и отличается более гибкой схемой редукции уровня шума и масштабирования, так же требует расширения системы оснований для операции смены модулей (modulus switching). Схема BGV менее удобна для реализации, тем не менее пользуется популярностью на практике.

В данной работе мы подробно анализируем алгоритм расширения системы оснований СОК, который требуется в процессе операции смены ключей системы шифрования, которая необходима для целого ряда утилитарных операций в BFV, таких как умножение двух шифртекстов, циклического сдвига вектора зашифрованных данных.

Расширение системы оснований происходит в два этапа: вычисление ранга числа и расчета нового основания на его основе. Для повышения производительности операции вычисления ранга числа наибольшей популярностью пользуются два метода, различающихся процессом вычисления ранга числа. Первый метод, основанный на точном вычислении ранга числа в системе остаточных классов [3], позволяет обеспечить асимптотически более высокую производительность по сравнению с наивным подходом. Второй метод основан на аппроксимации ранга, числа использующей комбинацию целочисленной и арифметики с плавающей точкой в дополнение к методам системы остаточных классов [4]. Этот метод требует относительно меньшего количества ресурсов и выполняется быстрее первого, однако он является приближенным и неизбежно приводит к погрешностям, связанным с переходом от вычислений с числами в несколько сотен бит к вещественным числам двойной точности, представленных в 64 битах. Для доказательства корректности работы второго метода использовалась теории погрешности, не учитывающая свойства системы остаточных классов.

В работе мы исследуем свойства алгоритма расширения оснований в системе остаточных классов основанного на аппроксимации ранга числа. Ключевыми результатами являются:

- Доказательство корректности предложенного метода: получены условия, при которых аппроксимированный ранг совпадает с точным рангом при заданных ограничениях гомоморфного шифрования;
- Исследование свойств метода вычисления точного ранга числа с использованием аппроксимированного ранга числа. Показано, что вычисления ранга числа на всем диапазоне системы остаточных классов требует ту же точность, что и функция определения знака числа Van Vu T. [9].

2. Математическая постановка задачи

Шифры BGV, CKKS и BFV оперируют элементами больших циклотомических колец, заданных по модулю целых, содержащих сотни бит. Реализация арифметических операций с числами гораздо большей разрядности чем это позволяют сделать базовые типы данных требует значительных вычислительных затрат, и одним из способов ускорения этих операций является использование системы остаточных классов. В частности, диапазон системы M целое число равное, $M = \prod_{i=1}^{k} p_i$, где p_i – модули системы остаточных классов. Модули остаточных классов являются попарно взаимно простыми числами, каждое из которых может быть представлено в виде одного машинного слова.

Из Китайской теоремы об остатках следует, что любое целое число $X \in Z_M$ может быть представлено в виде кортежа $(x_1, x_2, \cdots, x_k)$, где $\forall i: x_i = |X|_{p_i}$. Операции над X в Z_M могут быть реализованы посредством выполнения тех же операций над каждым компонентом x_i в своём кольце Z_{p_i}.

Как схема BGV, так и BFV включают операции масштабирования, которые нельзя напрямую реализовать над компонентами системы остаточных классов. В обеих схемах возникает

необходимость обрабатывать как положительные, так и отрицательные числа, поэтому мы будем считать, что $X \in \left[-\frac{M}{2}, \frac{M}{2}\right)$. Расширение оснований в системе остаточных классов задаётся следующим образом. Пусть $X \in Z_M$ задан в системе остаточных классов $(x_1, \dots, x_k)$, и требуется расширить основание, то есть вычислить $|X|_{p_{k+1}} \in Z_{p_{k+1}}$ для некоторого нового модуля p_{k+1} взаимно простого с M.

Используя Китайскую теорему об остатках, мы хотим вычислить $x_{k+1} = |X|_{p_{k+1}}$, то есть

$$x_{k+1} = |X|_{p_{k+1}} = \left| \sum_{i=1}^{k} \left| x_i |P_i^{-1}|_{p_i} \right|_{p_i} P_i - r_X M \right|_{p_{k+1}}$$

где $\forall i: P_i = \frac{M}{p_i}$ и $|P_i^{-1}|_{p_i}$ – мультипликативная инверсия P_i по модулю p_i.

Основная сложность в вычислении $x_{k+1} = |X|_{p_{k+1}}$ заключается в нахождении ранга числа r_X. Ранг числа r_X вычисляется в [3-4], используя следующую формулу:

$$r_X = \left\lfloor \sum_{i=1}^{k} \frac{1}{p_i} \left| x_i |P_i^{-1}|_{p_i} \right|_{p_i} \right\rfloor$$

где $y_i = \left| x_i |P_i^{-1}|_{p_i} \right|_{p_i}$ – целое число, $z_i = \frac{y_i}{p_i}$ – вычисляется с использованием арифметики с плавающей запятой.

После этого мы суммируем все z_i и округляем сумму к ближайшему целому числу:

$$r_X = \left\lfloor \sum_{i=1}^{k} z_i \right\rfloor$$

Таким образом, значения y_i и r_X мы можем непосредственно вычислить и

$$x_{k+1} = |X|_{p_{k+1}} = \left| \sum_{i=1}^{k} y_i |P_i|_{p_{k+1}} - r_X |M|_{p_{k+1}} \right|_{p_{k+1}}$$

Поскольку P_i являются заранее известными параметрами, мы можем предварительно вычислить все значения $|P_i|_{p_{k+1}}$ и $|M|_{p_{k+1}}$, так что вычисление сводится к вычислению скалярного произведения двух $(k+1)$-мерных векторов по модулю p_{k+1}. Для вычисления x_{k+1} необходимо k умножений с плавающей точкой, k умножений в кольце Z_{p_i} , $k+1$ целочисленных умножений и одна операция нахождения остатка от деления по модулю p_{k+1}. Мы считаем, что операция сложения и умножения имеют одну вычислительную сложность.

Единственным источником ошибок в данной методе являются операции с плавающей запятой при вычислении r_X: вместо точных значений $z_i = \frac{y_i}{p_i}$ используется приближенное значения $z_i^* = \frac{y_i}{p_i} + e_i$, где e_i – ошибка округления, возникавшая за счет использования чисел с плавающей точкой. В результате вычисляется значение

$$r_X^* = \left\lfloor \sum_{i=1}^{k} (z_i + e_i) \right\rfloor,$$

которое может отличаться от истинного значения $r_X = \left\lfloor \sum_{i=1}^{k} z_i \right\rfloor$.

При применении вышеописанной метода из работы [4] необходимо проверять, что полученное значение r_X^* не попадает в область возможной ошибки $Z + \frac{1}{2} \pm e$, где $e = \sum_{i=1}^{k} e_i$. Если r_X^* попадает в эту область, процедуру можно повторить с использованием арифметики более высокой точности (и, соответственно, меньшего e), пока результат не выйдет за пределы зоны неопределённости.

Для устранения вышеизложенного недостатка применим подход из работы [10] к поиску аппроксимации значения ранга числа.

Пусть $\bar{z}_i = \left\lfloor 2^N \cdot \frac{y_i}{p_i} \right\rfloor$, тогда аппроксимация ранга числа может быть вычислена следующим образом:

$$\bar{r}_X = \left\lfloor \frac{1}{2^N} \sum_{i=1}^{k} \bar{z}_i \right\rfloor$$

Потребуем, чтобы $\bar{r}_X = r_X$. Вычислим, какое значение необходимо взять для N, чтобы требования выполнилось $\bar{r}_X = r_X$.

3. Выбор параметров для метода аппроксимации ранга числа

Теорема 1. Если $N > \log_2 \frac{k}{1-2\delta}$ и $0 < \delta < \frac{1}{2}$ то $\forall X \in [-\delta \cdot M, \delta \cdot M) : r_X = \bar{r}_X$.

Доказательство

Пусть $\forall\, i: \left\lfloor 2^N \cdot \frac{y_i}{p_i} \right\rfloor = 2^N \cdot \frac{y_i}{p_i} + e_{X,i}$, где $-\frac{1}{2} < \ e_{X,i} \leq \frac{1}{2}$ и $-\frac{k}{2} < \sum_{i=1}^{k} e_{X,i} \ \leq \frac{k}{2}$. Следовательно

$$\bar{r}_X = \left\lfloor \frac{1}{2^N} \sum_{i=1}^{k} \left\lfloor 2^N \cdot \frac{y_i}{p_i} \right\rfloor \right\rfloor = \left\lfloor \sum_{i=1}^{k} \frac{y_i}{p_i} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} \right\rfloor$$

Учитывая, что $\sum_{i=1}^{k} \frac{y_i}{p_i} = r_X + \frac{X}{M}$ то

$$\bar{r}_X = r_X + \left\lfloor \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} \right\rfloor$$

Необходимым и достаточным условием является, чтобы $\forall X \in [-\delta \cdot M, \delta \cdot M) : r_X = \bar{r}_X$ выполнялось равенство:

$$\forall X \in [-\delta \cdot M, \delta \cdot M) : \left\lfloor \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} \right\rfloor = 0$$

Равносильно

$$\forall X \in [-\delta \cdot M, \delta \cdot M) : -\frac{1}{2} \leq \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{1}{2}$$

Учитывая, что $\forall X \in [-\delta \cdot M, \delta \cdot M) : r_X = \bar{r}_X$ если,

$$-\frac{1}{2} + \delta \leq \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{1}{2} - \delta$$

Учитывая, что $-\frac{k}{2} < \sum_{i=1}^{k} e_{X,i} \ \leq \frac{k}{2}$ то $\frac{k}{2^{N+1}} < \frac{1}{2} - \delta$, значит $N > \log_2 \frac{k}{1-2\delta}$.

Теорема доказана.

Покажем, что доказанная теорема 1 позволяет уменьшить размер операндов более чем в 3 раза по сравнению с работой [4] при одинаковых параметрах системы остаточных классов.

Пример 1. Для параметров заданных в [4]{Section 2.2: Correctness} $\left|\frac{X}{M}\right| \ll \frac{1}{4}$, то есть $\delta \ll \frac{1}{4}$ и $k \leq 32$, следовательно, $\log_2 \frac{k}{1-2\delta} < \log_2 64 = 6$, значит, при $N = 6$ условия теоремы 1 будут выполнены и $\forall X \in [-\delta \cdot M, \delta \cdot M) : r_X = \bar{r}_X$. При данных ограничениях системы остаточных классов достаточно проводить вычисления с точностью 6 знаков после запятой, что в $\frac{19}{6} = 3\frac{1}{6}$ раз меньше, чем предлагают использовать авторы работы [4].

Исследуем, какое N надо выбирать, если $\delta = 1/2$. Для этого докажем два утверждения. Первое утверждение: если диапазон системы остаточных классов не кратен 2, а второе утверждение относится к случаю, когда этот диапазон кратен 2.

Утверждение 1. Если $2 \nmid M$, $\delta = \frac{1}{2}$ и $N = \lceil \log_2(k \cdot M) \rceil$ то $\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right] : r_X = \bar{r}_X$.

Доказательство.

Так как

$$-\frac{1}{2} = -\frac{M+1}{2M} - \frac{1}{2M} \leq \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{M-1}{2M} + \frac{1}{2M} = \frac{1}{2}$$

Учитывая, что по условию утверждения $2 \nmid M$, то $\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right]$ и необходимое и достаточное условие можно будет записать в виде

$$\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right] : -\frac{1}{2} \leq \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{1}{2}.$$

Если $\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right] : \left|\frac{1}{2^N} \sum_{i=1}^{k} e_{X,i}\right| < \frac{1}{2M}$, то необходимое и достаточное условие выполняется $\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right]$. Учитывая, что $-\frac{k}{2} < \sum_{i=1}^{k} e_{X,i} \ \leq \frac{k}{2}$ то $\frac{k}{2^N} < \frac{1}{2M}$. Если выбрать N равное $N = \lceil \log_2(k \cdot M) \rceil$, то $\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right] : r_X = \bar{r}_X$.

Утверждение доказано.

Утверждение 2. Если $2 \mid M$, $\delta = \frac{1}{2}$ и $N = \lceil \log_2(k \cdot M) \rceil - 1$ то $\forall X \in \left[-\frac{M}{2}, \frac{M}{2} - 1\right] : r_X = \bar{r}_X$.

Доказательство.

Учитывая, что по условию утверждения $2 \mid M$, то $\forall X \in \left[-\frac{M}{2}, \frac{M}{2} - 1\right]$, и необходимое и достаточное условие можно записать в виде

$$\forall X \in \left[-\frac{M}{2}, \frac{M}{2} - 1\right] : -\frac{1}{2} \leq \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{1}{2}$$

Покажем, что при $X = -\frac{M}{2}$ и $N \geq 1$ выполняется равенство: $\sum_{i=1}^{k} e_{-M/2,i} = 0$. Без потери общности будем считать $2 \mid p_k$, тогда $-\frac{M}{2} \to \left(0, \cdots, 0, \frac{p_k}{2}\right)$, и $\sum_{i=1}^{k} e_{-M/2,i} = e_{-M/2,k}$.

Вычислим y_k при $X = -\frac{M}{2}$ получим:

$$y_k = \left| q |P_k^{-1}|_{p_k} \right|_{2 \cdot q} = q |P_k^{-1}|_{p_k} - 2 \cdot q \left\lfloor \frac{1}{2} |P_k^{-1}|_{p_k} \right\rfloor = q \left| |P_k^{-1}|_{p_k} \right|_2$$

Вычислим $2^N \cdot \frac{y_k}{p_k}$ при $N \geq 1$ и $X = -\frac{M}{2}$ получим:

$$2^N \cdot \frac{y_k}{p_k} = 2^N \cdot \frac{q \left| |P_k^{-1}|_{p_k} \right|_2}{2 \cdot q} = 2^{N-1} \left| |P_k^{-1}|_{p_k} \right|_2 \in Z$$

Следовательно, при $N \geq 1$, $e_{-M/2,k} = \left\lfloor 2^N \cdot \frac{y_k}{p_k} \right\rfloor - 2^N \cdot \frac{y_k}{p_k} = 0$.

Значит необходимое и достаточное условие можно будет записать в виде

$$\forall X \in \left[-\frac{M}{2} + 1, \frac{M}{2} - 1\right] : -\frac{1}{2} \leq \frac{X}{M} + \frac{1}{2^N} \sum_{i=1}^{k} e_{X,i} < \frac{1}{2}$$

Так как необходимое и достаточное условие можно представить в виде

$$-\frac{1}{2} = -\frac{1}{2} + \frac{1}{M} - \frac{1}{M} \le \frac{X}{M} + \frac{1}{2^N}\sum_{i=1}^{k} e_{X,i} < \frac{1}{2} - \frac{1}{M} + \frac{1}{M} = \frac{1}{2}$$

если выполняется неравенство

$$\forall X \in \left[-\frac{M+1}{2}, \frac{M-1}{2}\right] : \left|\frac{1}{2^N}\sum_{i=1}^{k} e_{X,i}\right| < \frac{1}{M}$$

то условие тоже выполняется.

Учитывая, что $-\frac{k}{2} < \sum_{i=1}^{k} e_{X,i} \le \frac{k}{2}$ то $\frac{k}{2^N} < \frac{1}{M}$. Если выбрать N равное $N = \lceil \log_2(k \cdot M) \rceil - 1$ то $\forall X \in \left[-\frac{M}{2}, \frac{M}{2} - 1\right] : r_X = \bar{r}_X$.

Утверждение доказано.

Из утверждения 1 и 2 следует, что вычисления ранга числа на всем диапазоне системы остаточных классов требует туже точность, что и функция определения знака числа Van Vu T. [9].

4. Моделирование

Моделирование проводилось под управлением операционной системы Ubuntu 25.04 Plucky в среде разработки Visual Studio Code 1.104.1 (Universal), процессор AMD Ryzen 9 7950X 16-Core Processor, оперативная память DDR5-6000MHz 64GB, на языке программирования Rust, версия: rustc 1.92.0-nightly (844264add 2025-10-14),

Моделирование производится с использованием модулей СОК p_i вида $p_i = 2^{47} + \alpha_i$, где $\alpha = [1, 3, 5, 9, 11, 15, 21, 23, 27, 29, 33, 35, 39, 41, 51, 53, 63, 65, 69, 71, 75, 83, 93, 95, 105, 111, 113, 131, 135, 141, 155, 159, 165, 173, 179, 189, 203, 219, 221, 225, 231, 233, 239, 243, 249, 261, 273, 281, 285, 299, 303, 309, 315,321, 329, 333, 335, 341, 359, 363, 369, 371, 375, 393, 401, 413, 419, 425, 443, 449, 453]$. Параметры алгоритма: $\delta = \frac{1}{4}$ и $N = 8$ для $k \in [80,128]$, $N = 9$ для $k \in [129,150]$. Заметно, что предлагаемое решение требует гораздо меньше разрядов для выполнения в сравнении с алгоритмом из [4], который работает вещественными числами с плавающей запятой двойной точности (как минимум).

При моделировании был реализован алгоритм расширения оснований в системе остаточных классов, где изменялась процедура вычисления ранга числа. В алгоритме расширение оснований из работы [4] ранг числа вычисляется с использованием алгоритма 1. В новый алгоритм алгоритме расширения основания ранг числа вычисляется с использованием алгоритма 2, и параметры алгоритма должны удовлетворять условиям теоремы 1.

Алгоритм 1. Вычисления r_X с использованием чисел с плавающей точкой [4].	**Алгоритм 2.** Вычисления r_X с использованием теоремы 1.
Input: $(y_1, y_2, \cdots, y_k)$, $(p_1, p_2, \cdots, p_k)$ **Output:** r_X – ранг числа	**Input:** $(y_1, y_2, \cdots, y_k)$, $(p_1, p_2, \cdots, p_k)$, N **Output:** r_X – ранг числа
1. $r \leftarrow 0$ 2. **for** $i = 1$ **to** k **do:** 2.1. $r \leftarrow r + \frac{double(y_i)}{double(p_i)}$ 3. **return** $\mathrm{round}(\boldsymbol{r})$	1. $r \leftarrow 0$ 2. **for** $i = 1$ **to** k **do:** 2.1. $r \leftarrow r + \left\lfloor \frac{y_i \ll N}{\boldsymbol{p_i}} \right\rfloor$ 3. **return** $\mathrm{round}\left(\frac{\boldsymbol{r}}{\boldsymbol{2^N}}\right)$

Для эксперимента заранее генерировались 1000 случайных чисел в диапазоне $[-\delta \cdot M, \delta \cdot M)$, представленных в системе остаточных классов, которые затем хранились в памяти компьютера. При моделировании рассматривались два сценария:

- *Сценарий 1.* Задано k оснований СОК, мы расширяем на одно основание СОК, что соответствует подходу в схеме шифрования BFV и CKKS. Результаты моделирования представлены на рис. 1.
- *Сценарий 2.* Задано k оснований СОК, мы расширяем на k оснований СОК, что имитирует вычисления для BGV. Результаты моделирования представлены на рис. 2. В данном случае ранг вычисляется единожды и используется для серии восстановлений.

Из данных, представленных на рис. 1 мы можем сделать вывод о том, что функция зависимости времени выполнения операции расширения оснований СОК от количества модулей для алгоритма из работы [4] выражается следующей закономерностью $t = 8.449 \cdot k + 666.88$ скоэффициентом детерминации $R^2 \approx 0.9875$, а функция зависимости времени выполнения операции расширения оснований СОК от количества модулей для предложенного алгоритма выражается следующей закономерностью $t = 0.9442 \cdot k + 114.34$ скоэффициентом детерминации $R^2 \approx 0.8883$. В среднем время работы алгоритма расширения оснований в СОК уменьшается на 84.5% за счет уменьшения разрядности операндов более, чем в 3 раза и переходе от чисел с плавающей точкой к целым числам. При этом стоит отметить, что стандартное отклонение для времени работы алгоритма расширения оснований в СОК от количества оснований в СОК изменяется в диапазоне от 43.00 до 488.76 причем максимальное стандартное отклонение достигается при $k = 139$, а минимальное при $k = 93$ [4]. Для предложенного алгоритма стандартное отклонение для времени работы алгоритма расширения оснований в СОК от количества оснований в СОК изменяется в диапазоне от 11.41 до 204.499 причем максимальное стандартное отклонение достигается при $k = 86$, а минимальное при $k = 84$.

По данным рис. 2, кубическая аппроксимация зависимости времени выполнения операции расширения оснований в СОК от числа модулей для алгоритма из работы [4] имеет вид: $t = -0.2636 \cdot k^3 + 9.9616 \cdot k^2 - 77.294k + 4781.4$ при коэффициенте детерминации $R^2 = 0.8741$; для предлагаемого алгоритма – $t = -0.1803 \cdot k^3 + 6.0088 \cdot k^2 - 49.937k + 2659.6$ при $R^2 = 0.5047$. В среднем время выполнения операции сокращается на 46.15% благодаря более чем трёхкратному уменьшению разрядности операндов и переходу от чисел с плавающей точкой к целочисленным. При этом стандартное отклонение времени для алгоритма из работы [4] изменяется в диапазоне от 387.12 до 2930.01 (минимум при $k = 88$, максимум при $k = 100$); для предлагаемого алгоритма – от 179.35 до 1466.43 (минимум при $k = 82$, максимум при $k = 96$).

5. Заключение

Гомоморфное шифрование позволяет выполнять вычисления над данными в зашифрованном виде в удалённой среде, такой как облачные платформы, и тем самым выступает одним из ключевых направлений защиты персональной информации. Главная практическая проблема таких систем - невысокая производительность. Для смягчения этой проблемы большинство современных реализаций опираются на модулярную арифметику в системе остаточных классов (СОК), где критически важной операцией является расширение системы оснований.

В данной работе предложено теоретическое обоснование точности вычисления аппроксимированного ранга с учётом структуры СОК и ограничений гомоморфного шифрования.

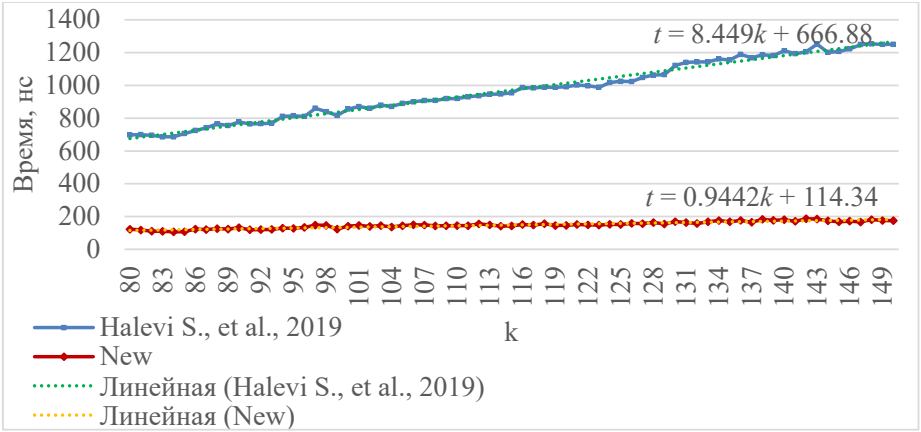

Рис. 1. Среднее время расширения на одно основание СОК.
Fig. 1. Average time of base extension in RNS by single modulus.

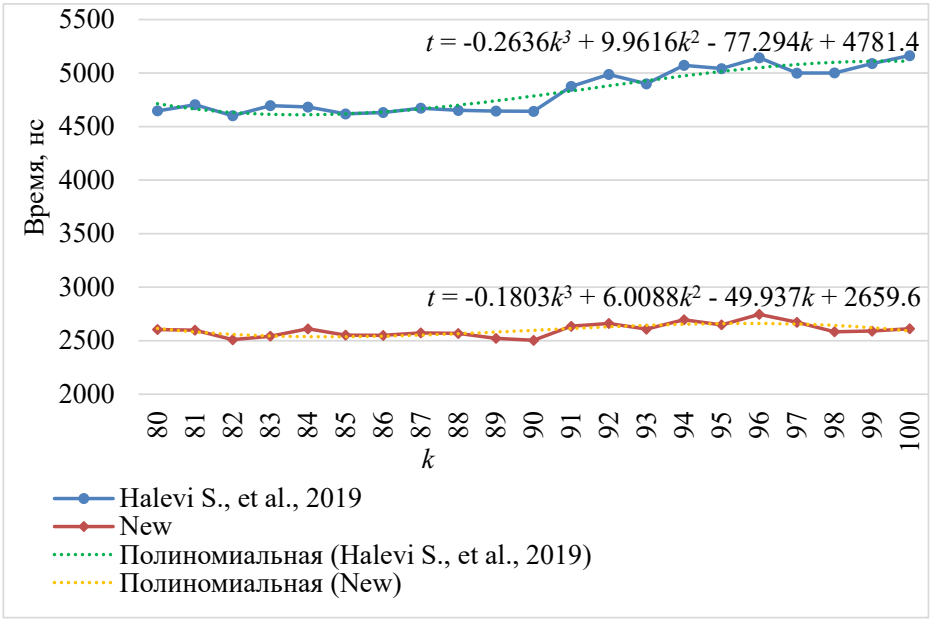

Рис. 2. Среднее время расширения оснований в СОК при удвоении количества оснований.
Fig. 2. Average time of base extension in RNS for doubling the number of moduli.

Полученная теорема даёт конструктивные условия выбора параметров, при которых можно более чем втрое сократить разрядность операндов по сравнению с оценками работы [4], отказаться от операций с плавающей точкой в пользу целочисленных вычислений и, тем самым, упростить реализацию как на CPU, так и на специализированных ускорителях. Результаты моделирования подтверждают практическую значимость анализа: оптимизированная версия алгоритма масштабирования чисел в СОК в среднем ускоряется на

46.15% относительно исходного варианта на базе алгоритма [4]. Ключевыми результатами статьи можно выделить, следующие:

1. Доказана корректность предложенного метода: получены условия, при которых аппроксимированный ранг совпадает с точным рангом при заданных ограничениях гомоморфного шифрования.
2. Исследованы свойства метода вычисления точного ранга числа с использованием аппроксимированного ранга; показано, что для вычисления ранга на всём диапазоне СОК требуется та же точность, что и для функции определения знака числа Van Vu T. [9].

Суммарно, предложенный теоретический и алгоритмический аппарат системно снижает вычислительную стоимость ключевых операций в СОК без потери корректности, создавая основу для дальнейшего ускорения практических схем гомоморфного шифрования. В частности, предложены подход будет полезен при проектировании специализированных ускорителей, где операции с плавающей запятой являются дорогостоящими и сложными для реализации.

Список литературы / References

[1]. Brakerski Z. Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. In Safavi-Naini R., Canetti R. (eds) Advances in Cryptology. CRYPTO 2012. CRYPTO 2012. Lecture Notes in Computer Science, vol. 7417, 2012, pp. 868-886. DOI: 10.1007/978-3-642-32009-5_50.
[2]. Fan J., Vercauteren F. Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, 2012. Available at: https://eprint.iacr.org/2012/144, accessed 05.11.2025.
[3]. Badawi A. Al, Polyakov Y., Kaung. M. M., Veeravalli B., Rohloff K. Implementation and performance evaluation of RNS variants of the BFV homomorphic encryption scheme. IEEE Transactions on Emerging Topics in Computing, 2019, vol. 9, no. 2, pp. 941-956. DOI: 10.1109/TETC.2019.2902799.
[4]. Halevi S., Polyakov Y., Shoup V. An improved RNS variant of the BFV homomorphic encryption scheme. In Matsui M. (eds) Topics in Cryptology. CT-RSA 2019. CT-RSA 2019. Lecture Notes in Computer Science, 2019, vol. 11405, pp 83-105. Springer, Cham. DOI: 10.1007/978-3-030-12612-4_5.
[5]. Geelen R., Vercauteren F. Bootstrapping for BGV and BFV Revisited. Journal of Cryptology, 2023, vol. 36, no. 2, pp. 12. DOI: 10.1007/s00145-023-09454-6.
[6]. Cheon J.H., Han K., Kim A., Kim M., Song Y. A full RNS variant of approximate homomorphic encryption. In: Cid, C., Jacobson Jr., M. (eds) Selected Areas in Cryptography – SAC 2018. SAC 2018. Lecture Notes in Computer Science, vol 11349, pp. 347-368. DOI: 10.1007/978-3-030-10970-7_16.
[7]. Brakerski Z., Vaikuntanathan V. Fully homomorphic encryption from ring-LWE and security for key dependent messages. In: Rogaway, P. (eds) Advances in Cryptology – CRYPTO 2011. CRYPTO 2011. Lecture Notes in Computer Science, vol 6841, pp. 505-524. DOI: 10.1007/978-3-642-22792-9_29.
[8]. Brakerski Z., Gentry C., Vaikuntanathan V. (Leveled) fully homomorphic encryption without bootstrapping. ACM Transactions on Computation Theory (TOCT), vol. 6, no. 3. 2014, pp. 1-36. DOI: 10.1145/2633600.
[9]. Van Vu T. Efficient implementations of the Chinese remainder theorem for sign detection and residue decoding. IEEE Transactions on Computers. vol. 100, no. 7, 1985, pp. 646-651. DOI: 10.1109/TC.1985.1676602.
[10]. Chervyakov N., Babenko M., Tchernykh A., Kucherov N., Miranda-López V., Cortés-Mendoza J. M. AR-RRNS: Configurable reliable distributed data storage systems for Internet of Things to ensure security. Future Generation Computer Systems, vol. 92, 2019, pp. 1080-1092. DOI: 10.1016/j.future.2017.09.061.

Информация об авторах / Information about authors

Михаил Григорьевич БАБЕНКО – доктор физико-математических наук, заведующий кафедры вычислительной математики и кибернетики факультета математики и компьютерных наук имени профессора Н.И. Червякова ФГАОУ ВПО «Северо-Кавказский федеральный университет». Сфера научных интересов: облачные вычисления,

высокопроизводительные вычисления, система остаточных классов, нейронные сети, криптография.

Mikhail Grigoryevich BABENKO – Dr. Sci. (Phys.-Math.), Head of the Department of Computational Mathematics and Cybernetics, Faculty of Mathematics and Computer Science named after Professor N.I. Chervyakov, North Caucasus Federal University. His research interests include cloud computing, high-performance computing, residue number systems, neural networks, cryptography

Антон Алексеевич СИНИЦЫН – аспирант 3 курса Института системного программирования Российской академии наук по специальности 2.3.5 «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей». Сфера научных интересов: гомоморфное шифрование, модулярная арифметика, криптография, безопасные вычисления, сохраняющие конфиденциальность.

Anton Alekseevich SINITSYN – 3rd-year postgraduate student at the Institute for System Programming of the Russian Academy of Sciences, specialty 2.3.5 “Mathematical and Software Support for Computational Systems, Complexes, and Computer Networks”. His research interests include homomorphic encryption, modular arithmetic, cryptography, secure privacy-preserving computing.

Максим Анатольевич ДЕРЯБИН – кандидат технических наук, научный сотрудник в Институте передовых технологий Samsung (Сувон, Южная Корея). Одной из основных тем его исследований является система остаточных чисел и её применение. Сфера научных интересов: криптографию на основе теории решёток, гомоморфное шифрование, вычислительную алгебру и теорию чисел.

Maxim Anatolyevich DERYABIN – Cand. Sci. (Tech.) in Computer Science, Staff Researcher at Samsung Advanced Institute of Technology, Suwon, South Korea. One of the major topics of his research was the residue number system and its applications. His current research interests include lattice-based cryptography, homomorphic encryption, computational algebra, and number theory.