Residue Number System (RNS)

Анатолий М. Георгиевский, 2025-09-19

{относится к оптимизациям ZKP в схемах BFV и CKKS}

Система остаточных классов (Residue Number System, RNS) является непозиционной системой целых чисел, основанной на китайской теореме об остатках (CRT).
В такой системе целое число x представляется его остатками $x_i = x \mod p_i$ по базису взаимно простых чисел $\mathcal{B} = \{p_0, \ldots, p_{k-1}\}$.
Множество $\mathcal{B} = \{p_0, \ldots, p_{k-1}\}$ формирует базис RNS, состоящий из k каналов. Модули p_i обычно выбираются с учетом ширины слова w, которая соответствует целевой архитектуре.
Важным преимуществом такой системы является то, что операции сложения, вычитания и умножения могут выполняться параллельно в каждом канале:

$$z_i = x_i \circ y_i \mod p_i, \text{ где } \circ \in \{+, -, \times\}$$

Традиционно рассматриваются системы $\{2^n + 1, 2^n, 2^n - 1\}$

Ряд работ по использованию RNS в доказательствах ZKP и FHE:

- [2016/510] A Full RNS Variant of FV like Somewhat Homomorphic Encryption Schemes
- [2018/117] An Improved RNS Variant of the BFV Homomorphic Encryption Scheme
- [2018/931] A Full RNS Variant of Approximate Homomorphic Encryption

Обозначения

Для целого числа $q \geq 2$ мы отождествляем кольцо $\mathbb{Z}_q$ с его отображением на симметричном интервале $\mathbb{Z} \cap [-q/2, q/2)$. Для произвольного действительного числа x мы обозначаем через $[x]_q$ отображение x на этот интервал, а именно, действительное число $x' \in [-q/2, q/2)$, такое что $x' - x$ делится на q. Мы также обозначаем через $\lfloor x \rfloor$, $\lceil x \rceil$ и $\lfloor x \rceil$ округление x вниз, вверх и до ближайшего целого числа, соответственно. Векторы мы обозначаем жирным шрифтом, и расширяем нотации $\lfloor \mathbf{x} \rfloor$, $\lceil \mathbf{x} \rceil$, $\lfloor \mathbf{x} \rceil$ на векторы поэлементно.

Мы выбираем множество из k взаимно простых модулей $\{p_0, \ldots, p_{k-1}\}$, где все числа целые больше 1, и пусть их произведение равно $P = \prod_{i=0}^{k-1} p_i$.

Для всех $i \in \{0, \ldots, k-1\}$ мы также обозначаем

$$\hat{p}_i = P/p_i \in \mathbb{Z} \quad \text{и} \quad \tilde{p}_i = \hat{p}_i^{-1} \pmod{q_i} \in \mathbb{Z}_{q_i},$$

где $\tilde{p}_i \in [-p_i/2, p_i/2)$ и $\hat{p}_i \cdot \tilde{p}_i = 1 \pmod{p_i}$.

Теорема об остатках (CRT)

Обозначим представление целого числа $x \in \mathbb{Z}_p$ относительно базиса RNS $\{p_0, \ldots, p_{k-1}\}$ через $x \sim (x_0, \ldots, x_{k-1})$, где $x_i = [x]_{p_i} \in \mathbb{Z}_{p_i}$. Формула, выражающая x через x_i, имеет вид

$$x = \sum_{i=0}^{k-1} x_i \cdot \tilde{p}_i \cdot \hat{p}_i \pmod{P}.$$

Эта формула может быть использована более чем одним способом для «реконструкции» значения $x \in \mathbb{Z}_p$ из $[x]_{\mathcal{B}}$. В данной работе мы используем следующие два факта:

$$x = \sum_{i=0}^{k-1} [x_i \cdot \tilde{p}_i]_{p_i} \cdot \hat{p}_i - e \cdot P \quad \text{для некоторого } e \in \mathbb{Z},$$

и

$$x = \sum_{i=0}^{k-1} x_i \cdot \tilde{p}_i \cdot \hat{p}_i - e' \cdot P \quad \text{для некоторого } e' \in \mathbb{Z},$$

где сумма во второй формуле берётся по $x_i \cdot \tilde{q}_i \cdot \hat{q}_i \in \big[-\frac{q_i q}{4}, \frac{q_i q}{4}\big)$.

Представление RNS в кольце

Пусть $\mathcal{B} = \{p_0, \ldots, p_{k-1}\}$ — это базис взаимно простых чисел, и пусть $P = \prod_{i=0}^{k-1} p_i$. Обозначим через $[\cdot]_{\mathcal{B}}$ отображение из $\mathbb{Z}_P \mapsto \mathbb{Z}_{p_0} \times \mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_{k-1}}$, определённое как $a \mapsto [a]_{\mathcal{B}} = ([a]_{p_i})_{0 \le i < k}$ -- отображение из множества целых чисел на множество остатков в базисе взаимно простых чисел. Это изоморфизм кольца по Теореме об остатках (CRT), и $[a]_{\mathcal{B}}$ называется представлением числа $a \in \mathbb{Z}_P$ в системе остаточных классов (RNS). Основное преимущество представления RNS заключается в возможности выполнения компонентных арифметических операций в малых кольцах $\mathbb{Z}_{p_i}$, что снижает асимптотическую и практическую вычислительную сложность. Этот изоморфизм кольца над целыми числами может быть естественным образом расширен до изоморфизма в кольце полиномов $[\cdot]_{\mathcal{B}} : \mathcal{R}_P \to \mathcal{R}_{p_0} \times \cdots \times \mathcal{R}_{p_{k-1}}$ путём пересчета коэффициентов над циклическими кольцами.

Расширение базиса CRT

Пусть $x \in \mathbb{Z}_P$ задано в представлении CRT $(x_0, \ldots, x_{k-1})$, и предположим, что мы хотим расширить базис CRT, вычислив $[x]_q \in \mathbb{Z}_q$ для некоторого другого модуля $q > 1$. Используя уравнение (2), мы хотели бы вычислить

$$[x]_q = \left[\left(\sum_{i=0}^{k-1} [x_i \cdot \tilde{p}_i]_{p_i} \cdot \hat{p}_i \right) - e \cdot P \right]_q .$$

Основная сложность здесь заключается в вычислении e, которое является целым числом в $\mathbb{Z}_k$. Формула для e выглядит так:

$$e = \left\lfloor \frac{\sum_{i=0}^{k-1} [x_i \cdot \tilde{p}_i]_{p_i} \cdot \hat{p}_i}{P} \right\rfloor = \left\lfloor \sum_{i=0}^{k-1} [x_i \cdot \tilde{p}_i]_{p_i} \cdot \frac{\hat{p}_i}{P} \right\rfloor = \left\lfloor \sum_{i=0}^{k-1} \frac{[x_i \cdot \tilde{p}_i]_{p_i}}{p_i} \right\rfloor .$$

Чтобы получить e, мы вычисляем для каждого $i \in \{0, \ldots, k-1\}$ элемент $\xi_i := [x_i \cdot \tilde{p}_i]_{p_i}$ используя арифметику целых чисел, а затем рациональное число $z_i := \xi_i / p_i$ в формате с плавающей точкой одинарной точности. Затем суммируем все z_i и округляем результат, чтобы получить e. {округление к меньшему для чисел без знака, проверить}:

$$e + \frac{x}{P} = \sum_{i=0}^{k-1} \frac{\xi_i}{p_i}, \quad e = \left\lfloor \sum_{i=0}^{k-1} \frac{\xi_i}{p_i} \right\rfloor ,$$

-- в такой форме должно быть справедливо для модульной арифметики без знака [23].

После того как мы получили значение e, а также все ξ_i, мы можем напрямую вычислить уравнение (2) по модулю q, чтобы получить

$$[x]_q = \left[\left(\sum_{i=0}^{k-1} \xi_i \cdot [\hat{p}_i]_q \right) - e \cdot [P]_q \right]_q .$$

Заметим, если все значения $[\hat{p}_i]_q$ и $[P]_q$ представить в качестве элементов вектора, то вычисление сводится к операции скалярного произведения векторов размерности $k+1$ по модулю q.

{данное описание достаточно полное, чтобы представить алгоритм расширения}

Преобразования базиса CRT

Прямое преобразование в RNS сводится к модульной операции на каждом базовом канале. Обратное преобразование может выполняться разными способами. Китайская теорема об остатках предоставляет вычислительную формулу в целевой системе чисел [2018/117]:

$$x + e \cdot P = \sum_{i=1}^{n} \left[x_i \cdot \hat{p}_i^{-1} \right]_{p_i} \cdot \hat{p}_i$$

где

$$\hat{p}_i \times \left(\frac{P}{p_i}\right)^{-1}_{p_i} \equiv 1 \pmod{P?}$$

Пусть $\mathcal{D} = \{p_0, \ldots, p_{k-1}, q_0, \ldots, q_{\ell-1}\}$ некоторый базис. Пусть $\mathcal{B} = \{p_0, \ldots, p_{k-1}\}$ и $\mathcal{C} = \{q_0, \ldots, q_{\ell-1}\}$ будут его подпространствами. Обозначим их произведения через $P = \prod_{i=0}^{k-1} p_i$ и $Q = \prod_{j=0}^{\ell-1} q_j$ соответственно. Тогда можно преобразовать RNS-представление $[a]_{\mathcal{C}} = (a^{(0)}, \ldots, a^{(\ell-1)}) \in \mathbb{Z}_{q_0} \times \cdots \times \mathbb{Z}_{q_{\ell-1}}$ целого числа $a \in \mathbb{Z}_Q$ в элемент $\mathbb{Z}_{p_0} \times \cdots \times \mathbb{Z}_{p_{k-1}}$ путём вычисления

$$\mathrm{Conv}_{\mathcal{C}\to\mathcal{B}}([a]_{\mathcal{C}}) = \left(\sum_{j=0}^{\ell-1} [a^{(j)} \cdot \hat{q}_j^{-1}]_{q_j} \cdot \hat{q}_j \pmod{p_i}\right)_{0 \le i < k},$$

где $\hat{q}_j = \prod_{i \neq j} q_i \in \mathbb{Z}$. Обратите внимание, что

$$\sum_{j=0}^{\ell-1} \left[a^{(j)} \cdot \hat{q}_j^{-1}\right]_{q_j} \cdot \hat{q}_j = a + Q \cdot e$$

для некоторого малого $e \in \mathbb{Z}$, удовлетворяющего $|a + Q \cdot e| \le (\ell/2) \cdot Q$. Это подразумевает, что $\mathrm{Conv}_{\mathcal{C}\to\mathcal{B}}([a]_C) = [a + Q \cdot e]_B$ может рассматриваться как RNS-представление целого числа $a + Q \cdot e$ относительно базиса $\mathcal{B}$.

- [2018/931]
  -- определяет две операции: увеличение и уменьшение размерности базиса, а также изменение базиса на основе этих двух операций.

Mixed Radix Conversion

RNS позволяет параллельно считать в числах с пониженной разрядностью. Но обратные операции связанные с вычислением знака, делением и сравеннием выполняются с использованием обратного преобразования в позиционную систему. Сравнение можно выполнить в позиционной системе Mixed Radix.

Алгоритм Гарнера

Рассмотрим набор модулей $(p_0, p_1, \ldots, p_{k-1})$, удовлетворяющих условию теоремы. Другой теоремой из теории чисел утверждается, что любое число $0 \leqslant x < M = p_0 \cdot p_1 \cdot \ldots \cdot p_{k-1}$ однозначно представимо в виде $x = x_0 + x_1 \cdot p_0 + x_2 \cdot p_0 \cdot p_1 + \cdots + x_{k-1} \cdot p_0 \cdot p_1 \cdot \ldots \cdot p_{k-1}$.

Вычислив по порядку все коэффициенты x_i для $i \in \{0, 1, \ldots, k-1\}$ мы сможем подставить их в формулу и найти искомое решение:

[Knuth2, 4.3.2]: Обозначим через $c_{ij} = p_i^{-1} \pmod{p_j}$, для $1 \leqslant i < j < k$ и рассмотрим выражение для x по модулю p_i получим:

$$
\begin{aligned}
x_0 &= r_0 \\
r_1 &= (x_0 + x_1 p_0) \pmod{p_1} \\
x_1 &= (r_1 - x_0) c_{01} \pmod{p_1} \\
r_2 &= (x_0 + x_1 p_0 + x_2 p_0 p_1) \pmod{p_2} \\
x_2 &= ((r_2 - x_0) c_{02} - x_1) c_{12} \pmod{p_2} \\
&\cdots \\
x_i &= (\ldots((r_i - x_0) c_{0,i} - x_1) c_{1,i} - \cdots - x_{i-1}) c_{(i-1),i} \pmod{p_i}.
\end{aligned}
$$

и так далее.

$$x = x_0 + x_1 p_0 + x_2 p_0 p_1 + \cdots + x_{k-1} p_0 p_1 \cdots p_{k-2}$$

- [Knuth2] Knuth, D. E. 2014. The Art of Computer Programming, Volume 2: Seminumerical Algorithms. Addison-Wesley Professional. ISBN:978-0-201-89684-8
- [26] Harvey L. Garner. 1959. The residue number system. In Papers Presented at the the March 3-5, 1959, Western Joint Computer Conference (IRE-AIEE-ACM '59 (Western)). Association for Computing Machinery, New York, NY, USA, 146–153. https://doi.org/10.1145/1457838.1457864

Algorithm. Mixed Radix Conversion

Requie: $\mathcal{B} = \{p_0, \ldots, p_{n-1}\}$ - набор из n взаимно простых модулей.
Requie: $a_i \equiv x \pmod{p_i}$ -- RNS представление $[a]_{\mathcal{B}}$
Шаг 1 precompute $\gamma_k = (\prod_{i=0}^{k-1} p_i)^{-1} \mod p_k$, for $k = 1, 2, ..., n-1$

1. for k from 1 to $n-1$
2. $\quad p \leftarrow p_0 \pmod{p_k}$
3. $\quad$ for i from 1 to $k-1$
4. $\quad\quad p \leftarrow p \cdot p_i \pmod{p_k}$
5. $\quad \gamma_k = p^{-1} \pmod{p_k}$

Шаг 2: Расчет коэффициентов MRC $\{v_i\}$ из RNS $\{a_i\}$

1. $v_0 \leftarrow a_0$

2. for k from 1 to $n-1$
3. $\quad t \leftarrow v_{k-1}$
4. $\quad$ for i from $k-2$ to 0
5. $\quad\quad t \leftarrow t \cdot p_i + v_i \pmod{p_k}$
6. $\quad v_k = (a_k - t)\gamma_k \pmod{p_k}$

Шаг 3: Расчет стандартного представления числа из MRC

1. $x \leftarrow v_{n-1}$
2. for k from $n-2$ to 0
3. $\quad x = x \cdot p_k + v_k$
4. return x

Дополнительная литература

- [22] N.S. Szabo, R.I. Tanaka. Residue Arithmetic and Its Applications to Computer Technology
- [23] Kawamura, S., Koike, M., Sano, F., Shimbo, A. (2000). Cox-Rower Architecture for Fast Parallel Montgomery Multiplication. In: Preneel, B. (eds) Advances in Cryptology — EUROCRYPT 2000. EUROCRYPT 2000. Lecture Notes in Computer Science, vol 1807. Springer, Berlin, Heidelberg. (https://doi.org/10.1007/3-540-45539-6_37)
- [24] J. . -C. Bajard, L. . -S. Didier and P. Kornerup, "An RNS Montgomery modular multiplication algorithm," in IEEE Transactions on Computers, vol. 47, no. 7, pp. 766-776, July 1998, (https://doi.org/10.1109/12.709376).
- [2025/1068] Efficient Modular Multiplication Using Vector Instructions on Commodity Hardware, 2025. Cryptology {ePrint} Archive, Paper 2025/1068
- [25] Kawamura, et al. Efficient Algorithms for Sign Detection in RNS Using Approximate Reciprocals, 2021