DOI: 10.15514/ISPRAS-2019-31(2)-13

Эффективное сравнение чисел в системе остаточных классов на основе позиционной характеристики

[1] *М.Г. Бабенко, ORCID: 0000-0001-7066-0061 <mgbabenko@ncfu.ru>*
[2,3,5] *А.Н. Черных, ORCID: 0000-0001-5029-5212 <chernykh@cicese.mx>*
[1] *Н.И. Червяков, ORCID: 0000-0002-4573-2032 <ncherviakov@ncfu.ru>*
[1] *В.А. Кучуков, ORCID: 0000-0002-1839-2765 <viktor-kuchukov@yandex.ru>*
[2] *В. Миранда-Лопес, ORCID: 0000-0002-1128-6660 <vmiranda@cicese.edu.mx>*
[2] *Р. Ривера-Родригес, ORCID: 0000-0002-1968-8525 <rrivera@cicese.mx>*
[4] *Чж. Ду, ORCID: 0000-0002-8435-1611 <duzh@tsinghua.edu.cn>*

[1] *Северо-Кавказский федеральный университет,*
355009, Россия, г. Ставрополь, ул. Пушкина, 1.
[2] *Центр научных исследований и высшего образования Энсенада,*
В.С. 22860, Мексика.
[3] *Институт системного программирования РАН им. В.П. Иванникова,*
109004, Россия, г. Москва, ул. А. Солженицына, д. 25.
[4] *Университет Цинхуа,*
Район Хайдянь, Пекин, 100084, КНР.
[5] *Южно-Уральский государственный университет,*
454080, Россия, г. Челябинск, ул. Ленина, 76.

Аннотация. Операция сравнения чисел широко используется при реализации большинства современных алгоритмов. Реализация алгоритма сравнения чисел в системе остаточных классов (СОК) состоит из двух этапов. Первый этап – вычисление позиционной характеристики модулярного числа. Второй этап – сравнение позиционных характеристик модулярных чисел в позиционной системе счисления. В статье предлагается новый эффективный алгоритм вычисления позиционной характеристики числа в СОК, основанный на использовании приближенного метода. Использование этого метода не требует дорогостоящих модульных операций, которые заменяются быстрыми битовыми операциями сдвиг вправо и взятия младших бит. Доказано, что в случае, когда динамический диапазон СОК является нечетным числом, размер операндов уменьшается на размер модуля. Если одно из оснований СОК является степенью двойки, то размер операндов меньше динамического диапазона.

Ключевые слова: система остаточных классов; немодульные операции; сравнение чисел; приближенный метод

Для цитирования: Бабенко М.Г., Черных А.Н., Червяков Н.И., Кучуков В.А., Миранда-Лопес В., Ривера-Родригес Р., Ду Чж. Эффективное сравнение чисел в системе остаточных классов на основе позиционной характеристики. Труды ИСП РАН, том 31, вып. 2, 2019 г., стр. 187-202. DOI: 10.15514/ISPRAS-2019-31(2)-13

Благодарности. Работа выполнена при поддержке стипендии Президента РФ молодым ученым и аспирантам, МК-341.2019.9, СП-2236.2018.5, а также грантов РФФИ 18-07-01224, 18-07-00109.

Efficient Number Comparison in the Residue Number System based on Positional Characteristics

[1] *M.G. Babenko <mgbabenko@ncfu.ru>*
[2,3,5] *A.N. Tchernykh <chernykh@cicese.mx>*
[1] *N.I, Chervyakov <ncherviakov@ncfu.ru>*
[1] *V.A. Kuchukov <vkuchukov@ncfu.ru>*
[2] *V. Miranda-López <vmiranda@cicese.edu.mx>*
[2] *R. Rivera-Rodriguez <rrivera@cicese.mx>*
[4] *Z. Du <duzh@tsinghua.edu.cn>*

[1] *North-Caucasus Federal University,*
1, Pushkin st., Stavropol, 355009, Russia.
[2] *CICESE Research Center,*
Ensenada, Baja California, 22860, Mexico.
[3] *Ivannikov Institute for System Programming of the Russian Academy of Sciences,*
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.
[4] *Tsinghua University,*
Haidian District, Beijing, 100084, P. R. China
[5] *South Ural State University,*
Chelyabinsk, 76 Lenina St., Chelyabinsk, 454080, Russia.

Abstract. An important operation for data processing is a number comparison. In Residue Number System (RNS), it consists of two steps: the computation of the positional characteristic of the number in RNS representation and comparison of its positional characteristics in the positional number system. In this paper, we propose a new efficient method to compute the positional characteristic based on the approximate method. The approximate method as a tool to compare numbers does not require resource-consuming non-modular operations that are replaced by fast bit right shift operations and taking the least significant bits. We prove that in case when the dynamic range of RNS is an odd number, the size of the operands is reduced by the size of the module. If one of the RNS moduli is a power of two, then the size of the operands is less than the dynamic range. It makes our method efficient for hardware implementation of cryptographic primitives and digital signal processing.

Keywords: residue number system, non-modular operation, number comparison, approximate method

For citation: Babenko M., Tchernykh A., Chervyakov N., Kuchukov V., Miranda-López V., Rivera-Rodriguez R., Du Z. Efficient Number Comparison in the Residue Number System based on Positional Characteristics. Trudy ISP RAN/Proc. ISP RAS, vol. 31, issue 2, 2019. pp. 187-202 (in Russian). DOI: 10.15514/ISPRAS-2019-31(2)-13

Acknowledgements. This work was partly supported by the scholarship of the President of the Russian Federation to young scientists and graduate students МК-341.2019.9, СП-2236.2018.5, and also grants of RFBR 18-07-01224, 18-07-00109

1. Введение

При использовании непозиционных систем счисления, таких как система остаточных классов (СОК), выполнение высокопроизводительных вычислений возможно за счет отсутствия переносов между разрядами. Однако во многих прикладных задачах возникает необходимость сравнения чисел. Данная операция является базовой при реализации большого числа алгоритмов защиты информации (Chang et al., 2015 [1], Chervyakov et al., 2017 [2], Sousa et al., 2016 [3]), цифровой обработки сигналов (Chervyakov et al., 2014) [4], систем беспроводной связи (Ye et al., 2018) [5], облачных вычислений (Tchernykh et al., 2016 [6], Miranda-López et al., 2017 [7], Tchernykh et al., 2017 [8], Babenko et al., 2017 [9]) и т.д.

Из-за непозиционной природы СОК немодульные операции, такие как сравнение чисел, определение знака числа и определение переполнения динамического диапазона, относятся к вычислительно сложным операциям.

В позиционной системе счисления существуют простые алгоритмы сравнения чисел, которые сводятся к их поразрядному сравнению. В СОК простых алгоритмов сравнения чисел нет (Szabo & Tanaka, 1969) [10]. Реализация алгоритма сравнения чисел в СОК состоит из двух этапов. Первый этап – вычисление позиционной характеристики (ПХ) модулярных чисел $X = (x_1, x_2, \dots, x_n)$ и $Y = (y_1, y_2, \dots, y_n)$. Второй этап – сравнение позиционных характеристик ПХ(X) и ПХ(Y) модулярных чисел в позиционной системе счисления (ПСС).

В качестве ПХ модулярного числа может выступать его представление в ПСС. Для перевода числа из СОК в ПСС можно использовать один из алгоритмов: Китайскую теорему об остатках (КТО), обобщенную позиционную систему счисления (Bi & Gross, 2008) [11], рекурсивный алгоритм сдваивания чисел (nCRT) (Wang, 2000) [12] и их модификации.

Большая вычислительная сложность алгоритмов вычисления искомого числа в двоичной системе счисления сподвигла исследователей на поиск его аппроксимации. С целью уменьшения вычислительной сложности операции сравнения чисел в СОК исследователи предложили в качестве ПХ модулярного числа использовать следующие функции: диагональная функция (Dimauro et al., 1993) [13], функция ядра (Burgess, 2003) [14], фактор-функция (Dimauro et al., 2003) [15], монотонная функция Pirlo (Pirlo and Impedovo, 2013) [16] и др. Предлагаемые алгоритмы вычисления ПХ позволяют уменьшить вычислительную сложность за счет уменьшения размерности операндов при выполнении операции деления с остатком.

Самым эффективным является подход, основанный на приближенном методе (Chervyakov et al., 2017) [17], так как он позволяет заменить операцию деления с остатком на операцию взятия старших бит числа. В статье мы предлагаем оптимизировать приближенный метод для вычисления операции сравнения чисел за счет уменьшения количества операций деления с остатком и улучшенной точности вычисления для корректной работы алгоритма.

Далее статья организована следующим образом. В разд. 2 нами рассмотрены основные положения СОК и ее свойства. В разд. 3 рассмотрены методы сравнения чисел, основанные на переводе чисел из СОК в ПСС. В разд. 4 проведен обзор методов сравнения на основе вычисления позиционных характеристик. В разд. 5 исследуется вопрос сравнения чисел в СОК с использованием методов определения знака числа. Разд. 6 посвящен модификации метода сравнения чисел и исследованию его свойств. В заключении представлены сравнение предложенного метода с существующими и основные выводы.

2. Система остаточных классов и ее свойства

Под остатком числа X по модулю p_i понимается число x_i, удовлетворяющее выражению $X = x_i + b \cdot p_i$ для некоторого числа b и $0 \le x_i < p_i$. Остаток от деления можно записать в терминах теории сравнений $x_i \equiv X \bmod p_i$, или для краткости $|X|_{p_i}$.

СОК определяется набором взаимно простых чисел p_i, называемых модулями, т.е. $\{p_1, p_2, \dots, p_n\}$, где НОД$(p_i, p_j) = 1$ для $i \neq j$, n – количество модулей. Любое число $X \in [0, P-1]$ может быть представлено в СОК единственным образом как $X = (x_1, x_2, \dots, x_n)$, где $x_i \equiv X \bmod p_i$, а $P = \prod_{i=1}^{n} p_i$ – динамический диапазон.

Особенностью СОК является возможность выполнения операций сложения, вычитания и умножения параллельно и независимо по каждому из модулей. Пусть даны два числа $X = (x_1, x_2, \dots, x_n)$ и $Y = (y_1, y_2, \dots, y_n)$, тогда, как показано в (Акушский & Юдицкий, 1968) [20], выполняется

$$C = X * Y = \left(|x_1 * y_1|_{p_1}, \dots, |x_n * y_n|_{p_n}\right),$$

где $*= \{+, -, \times\}$.

Однако, при выполнении арифметических операций возможна ситуация, когда результат выходит за диапазон $C \notin [0, P-1]$, т.е. происходит переполнение, и результат будет отличен от ожидаемого на размер диапазона. Для проверки корректности результата в статье (Chervyakov et al., 2017) [2] разработана схема, основанная на использовании свойств ранга числа. Преимущество предложенного подхода заключается в том, что он позволяет проверить корректность результата, не восстанавливая само число.

Использование метода приближенного вычисления ранга числа позволяет уменьшить вычислительную сложность алгоритмов перевода чисел из СОК в позиционную систему счисления.

3. Методы сравнения чисел, основанные на переводе чисел из СОК в ПСС

В большинстве методов задача сравнения чисел решается через перевод числа из СОК в ПСС и их сравнение.

3.1 Китайская теорема об остатках

Согласно (Omondi & Premkumar, 2007) [21], для перевода числа из СОК в ПСС используется стандартное восстановление с помощью КТО, которую можно записать формулой:

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

где $P_i = \frac{P}{p_i}$, а $|P_i^{-1}|_{p_i}$ – мультипликативная инверсия P_i по модулю p_i. Рассмотрим примеры восстановления числа по формуле (1) и сравнения чисел.

Пусть дана СОК $\{3,5,7\}$ и числа $X = (2,2,3)$, $Y = (1,3,4)$. Динамический диапазон данной системы остаточных классов равен $P = 3 \cdot 5 \cdot 7 = 105$.

Вычислим P_i:

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

Чтобы вычислить мультипликативную инверсию P_i, нужно найти такое x, которое удовлетворяет сравнению $x \cdot P_i \equiv 1 \bmod p_i$. Таким образом, $|P_1^{-1}|_3 = 2$, $|P_2^{-1}|_5 = 1$, $|P_3^{-1}|_7 = 1$. Таким образом, все необходимые для вычисления (1) данные получены. Найдем значение первого числа:

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

Найдем 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 $17 < 88$, значит $X < Y$.

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

3.2 Обобщенная позиционная система счисления (ОПСС)

ОПСС за счет своих свойств позволяет сравнивать два числа без прямого востановления самого сисла. Число в ОПСС задается кортежем $[a_1, a_2, \dots, a_n]$, а основаниями системы являются p_1, p_1p_2, $p_1p_2p_3$, …, $p_1, p_2, \dots, p_{n-1}$, где $p_1, p_2, \dots, p_n$ – модули СОК. Связь между двоичной системой счисления и ОПСС определяется по следующей формуле:

$$X = a_1 + a_2p_1 + a_3p_1p_2 + \cdots + a_np_1p_2 \dots p_{n-1},$$

Так как ОПСС является позиционной системой счисления, то сравнение чисел равносильно сравнению двух кортежей $[a_1, a_2, \dots, a_n]$ и $[b_1, b_2, \dots, b_n]$.

Для перевода числа $X = (x_1, x_2, \dots, x_n)$ из СОК в $[a_1, a_2, \dots, a_n]$ ОПСС используется следующий подход:

$a_1 = x_1$,

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

$a_3 = |(x_3 - a_1 - a_2 p_1) \cdot p_1^{-1} \cdot p_2^{-1}|_{p_3}$,

……

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

Эффективная реализация алгоритма сравнения чисел с использованием ОПСС представлена в работе (Isupov, 2016) [22].

Использование ОПСС позволяет уйти от вычисления остатка от деления на большое число P, но приводит к использованию большего количества модульных операций по модулям СОК.

3.3 Приближенный метод

Для исключения операции деления с остатком на большое простое число в статье (Van, 1985) [23] предложен приближенный метод, основанный на отображении, переводящем $[0, P)$ в $[0, 2)$. Для этого перепишем (1) в виде

$$X = \sum_{i=1}^{n} P_i \cdot x_i \cdot |P_i^{-1}|_{p_i} - P \cdot r_x, \tag{2}$$

для некоторого неотрицательного целого числа r_x - ранга числа. Разделив (2) на $\frac{P}{2}$, получим

$$X_s = \left(\frac{2}{P}\right) \cdot X = \sum_{i=1}^{n} \frac{2}{p_i} \cdot x_i \cdot |P_i^{-1}|_{p_i} - 2r_x. \tag{3}$$

Таким образом, из (3), X_s может быть вычислен как сумма дробных чисел с отбрасыванием кратной двум целой части результата. Это может быть получено довольно тривиально, поскольку вычисления выполняются в двоичном виде. Проиллюстрируем это на примере. Пусть дано число $(2, 2, 3)$, тогда по формуле (3) получим

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

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) выделить $N + 1$ бит, 1 – на целую часть и N – на дробную, и усекать оставшиеся биты, то ошибка в каждом слагаемом будет удовлетворять неравенству $0 \leq e_i < 2^{-N}$. И поскольку таких слагаемых n, то максимальная ошибка при усечении (3) будет $e = n2^{-t}$.

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

Кроме того, интервал между наибольшим положительным числом и 1 равен $\frac{2}{P}$ для четного P и $\frac{1}{P}$ для нечетного P.

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

$$n \cdot 2^{-N} \le \frac{2}{P}\text{, для четного } P \quad (4)$$

$$n \cdot 2^{-N} \le \frac{1}{P}\text{, для нечетного } P, \quad (5)$$

или

$$N \ge \lceil \log_2 P \cdot n \rceil - 1 \text{ для четного } P,$$
$$N \ge \lceil \log_2 P \cdot n \rceil \text{ для нечетного } P.$$

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

Рассмотрим численный пример. Пусть в СОК $\{3, 5, 7\}$ заданы два числа $1 = (1, 1, 1)$ и $104 = (2, 4, 6)$. Для данной системы $N \ge \lceil \log_2(105 \cdot 3) \rceil = 9$. Рассмотрим первое число по слагаемым:

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

Просуммируем по модулю 2 полученные слагаемые и получим 0.000001010.

Рассмотрим второе число:

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

Просуммируем по модулю 2 полученные 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 (4)-(5).

Стоит заметить, что использование данного подхода позволяет вычислять позиционную характеристику с применением следующей формулы:

$$V(X) = \left| \sum_{i=1}^{n} \left\lceil \frac{2}{p_i} \left| |P_i^{-1}|_{p_i} \cdot x_i \right|_{p_i} \right\rceil_{2^{-N}} \right|_2, \quad (6)$$

где $\lceil x \rceil_{2^{-N}} = \lceil 2^N x \rceil / 2^N$.

С целью уменьшения количества операций в приближенном методе в (Chervyakov et al., 2017) [17] предложено использовать следующую формулу:

$$C(X) = \left| \sum_{i=1}^{n} W_i x_i \right|_1, \quad (7)$$

где $W_i = \left\lfloor \frac{2^N |P_i^{-1}|_{p_i}}{p_i} \right\rfloor / 2^N$, $|x|_1$-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 при этом увеличились размеры операндов.

4. Методы сравнения чисел в СОК с использованием позиционных характеристик

С целью уменьшения вычислительной сложности алгоритма сравнения чисел исследователи (Dimauro et al., 1993) [13] предложили использовать монотонную диагональную функцию.

4.1 Диагональная функция

Отличным от вышеизложенных методов сравнения чисел является метод на основе специальной диагональной функции, которая определяется как сумма соответствующих коэффициентов P_i и называется методом суммы коэффициентов (Sum of Quotients Technique, SQT), описание которой можно найти, например, в (Dimauro et al., 1993) [13]. Диагональная функция представляет собой монотонно возрастающую функцию, на основе которой возможно сравнение чисел.

Диагональная функция имеет вид:

$$D(X) = \left\lfloor \frac{X}{p_1} \right\rfloor + \left\lfloor \frac{X}{p_2} \right\rfloor + \cdots + \left\lfloor \frac{X}{p_n} \right\rfloor. \quad (8)$$

Однако формула (8) является мало пригодной на практике. В связи с этим (Dimauro et al., 1993) [13] была предложена аналитическая функция для вычисления диагональной функции:

$$D(X) = \left| \sum_{i=1}^{n} k_i^* \cdot x_i \right|_{SQ} \quad (9)$$

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

Так как диагональная функция (9) 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 $D(X) = D(Y)$, и тогда $X < Y$, когда $x_i < y_i$, $i = 1, \dots, n$.

Рассмотрим пример сравнения чисел. Возьмем ранее использованные числа $X = (2, 2, 3)$ и $Y = (1, 3, 4)$. Для начала вычислим значения

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

Найдем значение диагональной функции:

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

Т.к. $D(X) < D(Y)$, то $X < Y$.

4.2 Функция ядра Акушского

Обобщив результат, полученный (Dimauro et al., 1993) [13], исследовательская группа (Pirlo & Impedovo, 2013) [16] предложила использовать минимальную функцию ядра Акушского

без критических ядер. Данный подход является аналогичным методу диагональной функции. Функция Pirlo имеет следующий вид:

$$Pi(X) = \left\lfloor \frac{X}{p_n} \right\rfloor \tag{10}$$

Однако формула (10) является мало пригодной на практике, в связи с этим была предложена аналитическая функция для вычисления Pirlo функции:

$$Pi(X) = \left| \sum_{i=1}^{n} k_i^{**} \cdot x_i \right|_{P_n} \tag{11}$$

где $k_i^{**} = \left\lfloor \frac{|P_i^{-1}|_{p_i} P_i}{p_n} \right\rfloor$.

Так как функция Pirlo (11) 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 $X < Y$, когда $x_n < y_n$.

Рассмотрим пример сравнения чисел. Возьмем ранее использовавшиеся числа $X = (2, 2, 3)$ и $Y = (1, 3, 4)$. Для начала вычислим значения:

$$P_3 = 15,$$
$$k_1^{**} = \left\lfloor \frac{|P_1^{-1}|_{p_1} P_1}{p_3} \right\rfloor = \left\lfloor \frac{2 \cdot 35}{7} \right\rfloor = 10,$$
$$k_2^{**} = \left\lfloor \frac{|P_2^{-1}|_{p_2} P_2}{p_3} \right\rfloor = \left\lfloor \frac{1 \cdot 21}{7} \right\rfloor = 3,$$
$$k_3^{*} = \left\lfloor \frac{|P_3^{-1}|_{p_3} P_3}{p_3} \right\rfloor = \left\lfloor \frac{1 \cdot 15}{7} \right\rfloor = 2.$$

Найдем значение функции Pirlo:

$$\mathrm{Pi(X)} = |2 \cdot 10 + 2 \cdot 3 + 3 \cdot 2|_{15} = 2,$$
$$\mathrm{Pi(Y)} = |1 \cdot 10 + 3 \cdot 3 + 4 \cdot 2|_{15} = 12$$

и поскольку $Pi(X) < Pi(Y)$, то $X < Y$.

Как показано в работе (Mohan, 2016) [24], функция Pirlo проигрывает Китайской теореме об остатках, так как требует дополнительных сравнений чисел.

5. Сравнение чисел на основе алгоритма определения знака числа

С целью оптимизации алгоритма сравнения чисел иногда целесообразно использовать на втором этапе вместо алгоритма сравнения алгоритм определения знака числа.

Некоторые приложения в СОК требуют использования отрицательных чисел. Для определения знака числа в СОК с отрицательными числами необходимо сравнить это число с серединой диапазона. Следует также обратить внимание, что в данном случае отрицательные числа идут за положительными, и для сравнения чисел сначала нужно определить их знак.

В СОК с модулями $\{p_1, p_2, \dots, p_n\}$ и динамическим диапазоном $P = \prod_{i=1}^{n} p_i$ может быть представлено число X, удовлетворяющее следующим соотношениям:

$$-\frac{P-1}{2} \leq X \leq \frac{P-1}{2}, \text{если } P \text{ нечетное},$$
$$-\frac{P}{2} \leq X \leq \frac{P}{2} - 1, \text{если } P \text{ четное}.$$

Тогда, согласно (Omondi & Premkumar, 2007) [21], если $X = (x_1, x_2, \dots, x_n)$, то отрицательным будет число $-X = (\overline{x_1}, \overline{x_2}, \dots, \overline{x_n})$, где $\overline{x_i}$ является дополнением x_i до модуля m_i. Например,

для СОК $\{3,5,7\}$ 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 СОК: 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 $\frac{P}{2}$):

$$S(X) = \begin{cases} 0, \text{если } 0 \le \mathrm{X} < \dfrac{\mathrm{P}-1}{2}, \\ 1, \text{если } \dfrac{\mathrm{P}-1}{2} \le \mathrm{X} < \mathrm{P}. \end{cases} \tag{12}$$

Рассмотрим на примере сравнение чисел с использованием отрицательных чисел. Пусть необходимо сравнить числа $\mathrm{X} = 17 = (2,2,3)$ и $Y = -8 = (1,2,6)$ в СОК $\{3,5,7\}$. Если $X > Y$, то $(X - Y) > 0$. Найдем разность

$$X - Y = (2-1, 2-2, 3-6) = (1,0,4).$$

Применим приближенную формулу на основе КТО и будем сравнивать результат с серединой диапазона, т.е. с $\frac{1}{2}$. Все константы предварительно вычислены в предыдущих примерах.

$$\frac{X}{P} = \left| \sum_{i=1}^{3} \frac{x_i \cdot |P_i^{-1}|_{p_i}}{p_i} \right|_1 = \left| \frac{2}{3} + \frac{4}{7} \right|_1 = \frac{5}{21} < \frac{1}{2}.$$

Очевидно, что поскольку полученное значение меньше середины диапазона, то оно положительное, и значит $X > Y$.

Стоит отметить, что для корректной работы алгоритма сравнения чисел на основе алгоритма определения знака числа требуется удвоение диапазонов СОК, что ведет к дополнительным вычислительным нагрузкам при обработке данных, однако в данном случае необходимо нахождение лишь одной позиционной характеристики числа

6. Модификация алгоритма сравнения чисел в СОК

В качестве позиционной характеристики рассмотрим следующую функцию:

$$f(X) = \left| \sum_{i=1}^{n} \overline{k_i}\ x_i \right|_{2^N},$$

где $\overline{k_i} = \left\lfloor \frac{2^N |P^{-1}|_{p_i}}{p_i} \right\rfloor$.

6.1 Сравнение числа в СОК с нечетным диапазоном

Лемма 1. Если $N = \lceil \log_2(n \cdot P - n) \rceil$, то справедливо следующее равенство:

$$\left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}\, x_i}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^{n} k_i x_i}{P} \right\rfloor, \tag{14}$$

где $\overline{k_i} = \left\lfloor \frac{2^N |P^{-1}|_{p_i}}{p_i} \right\rfloor$ и $k_i = |P^{-1}|_{p_i} P_i$.

Доказательство.

Так как k_i и $\overline{k_\iota}$ связаны равенством $\overline{k_\iota} = \frac{2^N k_i}{P} - \frac{|2^N k_i|_P}{P}$, то выражение $\left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, x_i}{2^N} \right\rfloor$ примет вид:

$$\left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, x_i}{2^N} \right\rfloor = \left\lfloor \frac{1}{P} \sum_{i=1}^n k_i\, x_i - \frac{1}{P \cdot 2^N} \sum_{i-1}^n |2^N k_i|_P \cdot x_i \right\rfloor \quad (15)$$

Подставим $\frac{1}{P}\sum_{i=1}^n k_i\, x_i = \left\lfloor \frac{1}{P}\sum_{i=1}^n k_i\, x_i \right\rfloor + \frac{X}{P}$ в (15), получим:

$$\left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, x_i}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^n k_i x_i}{P} \right\rfloor + \left\lfloor \frac{X}{P} - \frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P\, x_i \right\rfloor \quad (16)$$

Из (16) следует, что условие леммы 1 эквивалентно следующему неравенству:

$$0 \le \frac{X}{P} - \frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P\, x_i < 1 \quad (17)$$

Согласно Китайской теореме об остатках, X удовлетворяет условию $0 \le X < P$, следовательно, $0 \le \frac{X}{P} < 1$. Принимая во внимание, что $\frac{1}{P \cdot 2^N} \cdot \sum_{i=1}^n |2^N k_i|_P\, x_i \ge 0$, мы получаем, что правая часть двойного неравенства (17) верна для всех N.

Рассмотрим левую часть двойного неравенства (17). При $X = 0$ она выполняется для любого N. Пусть X удовлетворяет неравенству $1 \le X < P$, тогда левую часть неравенства (17) можно представить в следующем виде:

$$2^N \ge \frac{1}{X} \sum_{i=1}^n |2^N k_i|_P\, x_i \quad (18)$$

Так как $|2^N k_i|_P \le P - 1$, то $\sum_{i=1}^n |2^N k_i|_P\, x_i \le (P-1)\sum_{i=1}^n x_i$, следовательно, для всех $1 \le X < P$ справедливо следующее неравенство:

$$\frac{1}{X} \sum_{i=1}^n |2^N k_i|_P\, x_i \le n \cdot (P-1) \quad (19)$$

Из (18) и (19) следует, что если $N = \lceil \log_2(-n + n \cdot P) \rceil$, то левая часть неравенства (17) выполняется, следовательно, равенство (14) выполняется. Лемма доказана.

Теорема 1. Если $N = \lceil \log_2(-n + n \cdot P) \rceil$, то функция $f(X)$ – строго возрастающая.

Доказательство.

Для того чтобы $f(X)$ являлась строго возрастающей функцией, необходимо и достаточно, чтобы для всех целых чисел $1 \le X \le P - 1$ выполнялось следующее условие:

$$f(X) - f(X-1) > 0 \quad (20)$$

Так как $|X|_{2^N} = X - \left\lfloor \frac{X}{2^N} \right\rfloor \cdot 2^N$, то функцию $f(X)$ можно представить в следующем виде:

$$f(X) = \sum_{i=1}^n \overline{k_i}\; x_i - \left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, x_i}{2^N} \right\rfloor \cdot 2^N \quad (21)$$

Принимая во внимание, что $\sum_{i=1}^n \overline{k_i}\; \left(x_i - |x_i - 1|_{p_i}\right) = \sum_{i=1}^n \overline{k_i} - \sum_{x_i=0} \overline{k_\iota}\, p_i$, то получим, что

$$f(X) - f(X-1) =$$
$$= \sum_{i=1}^n \overline{k_i} - \sum_{x_i=0} \overline{k_\iota}\, p_i - \left(\left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, x_i}{2^N} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^n \overline{k_i}\, |x_i - 1|_{p_i}}{2^N} \right\rfloor \right) \cdot 2^N \quad (22)$$

Так как условие леммы 1 выполнено, то

$$\left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}\, x_i}{2^N} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}\, |x_i - 1|_{p_i}}{2^N} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^{n} k_i x_i}{P} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^{n} k_i |x_i - 1|_{p_i}}{P} \right\rfloor \quad (23)$$

Используя теорему из работы (Chervyakov, et. al., 2017) [2] и лемму 1, формула (23) примет вид:

$$\left\lfloor \frac{\sum_{i=1}^{n} k_i x_i}{P} \right\rfloor - \left\lfloor \frac{\sum_{i=1}^{n} k_i |x_i - 1|_{p_i}}{P} \right\rfloor = \left\lfloor \frac{\sum_{i=1}^{n} k_i}{P} \right\rfloor - \sum\nolimits_{x_i=0} |P_i^{-1}|_{p_i} = \\ = \left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}}{2^N} \right\rfloor - \sum\nolimits_{x_i=0} |P_i^{-1}|_{p_i} \quad (24)$$

Подставив (24) в (22), получим:

$$f(X) - f(X-1) = \\ = \sum_{i=1}^{n} \overline{k_i} - \sum\nolimits_{x_i=0} \overline{k_i}\, p_i - \left(\left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}}{2^N} \right\rfloor - \sum\nolimits_{x_i=0} |P_i^{-1}|_{p_i} \right) \cdot 2^N \quad (25)$$

Так как $\sum_{i=1}^{n} \overline{k_i} - \left\lfloor \frac{\sum_{i=1}^{n} \overline{k_i}}{2^N} \right\rfloor \cdot 2^N = \left| \sum_{i=1}^{n} \overline{k_i} \right|_{2^N}$ и $|P_i^{-1}|_{p_i} \cdot 2^N - \overline{k_i} p_i = \frac{|2^N k_i|_P}{P_i}$ то для всех $i = \overline{1,n}$ формула (25) примет вид:

$$f(X) - f(X-1) = \left| \sum_{i=1}^{n} \overline{k_i} \right|_{2^N} + \sum_{x_i=0} \frac{|2^N k_i|_P}{P_i} \quad (26)$$

Так как $\left| \sum_{i=1}^{n} \overline{k_i} \right|_{2^N} > 0$, то из (26) следует, что $f(X) - f(X-1) > 0$, и, следовательно, функция $f(X)$ строго возрастает. Теорема доказана.

Из теоремы 1 следует, что введенная функция является строго монотонной, следовательно, ее можно использовать в качестве позиционной характеристики для сравнения чисел в СОК.

Предложенный подход позволяет уменьшить вычислительную сложность алгоритма сравнения чисел в СОК. Эффективная аппаратная реализация операции $|x \cdot y|_{2^N}$ позволяет уменьшить логическую схему при аппаратной реализации по сравнению с классическим умножением двух чисел $x \cdot y$.

6.2 Сравнение чисел в СОК, если один из модулей равен степени двойки

Так как модули СОК являются взаимно простыми числами, следовательно, четный модуль только один. Значит, без потери общности будем считать, что n-ый модуль имеет вид $p_n = 2^t$. Так как $p_n = 2^t$, то используя свойство СОК, числа X, Y могут быть представлены в следующем виде:

$$X = A \cdot 2^t + x_n, Y = B \cdot 2^t + y_n. \quad (27)$$

Для сравнения чисел сравним A и B. Если A$<$ B, то $X < Y$. В случае, когда $A = B$, $X < Y$ при условии, что $x_n < y_n$.

Так как n-ый модуль СОК четный, следовательно, модули $p_1, p_2, \dots, p_{n-1}$ являются нечетными числами, тогда P_n - нечетное число. Коэффициенты A и B удовлетворяют неравествам: $0 \leq A < P_n$ и $0 \leq B < P_n$. Вычислив значения A и B в СОК по модулям $p_1, p_2, \dots, p_{n-1}$, мы можем сравнить их, используя введеную функцию $f(X)$.

Таким образом, алгоритм сравнение чисел X и Y будет иметь вид:

Алгоритм. Алгоритм сравнения чисел X и Y.

Input: $X \xrightarrow{RNS} (x_1, x_2, \dots, x_n)$,

$Y \xrightarrow{RNS} (y_1, y_2, \dots, y_n)$,

$p_1, p_2, \dots, p_{n-1}, p_n$,

$I_i = \left|\frac{1}{p_n}\right|_{p_i}$ для всех $i = \overline{1, n-1}$,

$\overline{k_\iota} = \left\lfloor 2^N \cdot |1/P_i^*|_{p_i}/p_i \right\rfloor$ для всех $i = \overline{1, n-1}$, где $N = \lceil \log_2(-n + nP_n) \rceil$ и $P_i^* = P_n/p_i$ для всех $i = \overline{1, n-1}$.

Output: $X > Y$-'10', $X < Y$-'01', $X = Y$ - '00'.

1. **For** $i \coloneqq 1$ **to** $n-1$ **do:**

1.1. $a_i \coloneqq |x_i - x_n|_{p_i}$; \\Parallel processing

1.2. $b_i \coloneqq |y_i - y_n|_{p_i}$; \\Parallel processing

2. **For** $i \coloneqq 1$ **to** $n-1$ **do:**

2.1. $a_i \coloneqq |a_i \cdot I_i|_{p_i}$; \\ Parallel processing

2.2. $b_i \coloneqq |b_i \cdot I_i|_{p_i}$; \\Parallel processing

3. $S_A = 0$; $S_B = 0$;

4. **For** $i \coloneqq 1$ **to** $n-1$ **do:**

4.1. $S_A = \left|S_A + \overline{k_\iota} \cdot a_i\right|_{2^N}$;

4.2. $S_B = \left|S_B + \overline{k_\iota} \cdot b_i\right|_{2^N}$;

5. IF $S_A > S_B$ **then return** '10'

6. **IF** $S_A < S_B$ **then return** '01'

7. **IF** $a_n > b_n$ **then return** '10'

8. **IF** $a_n < b_n$ **then return** '01'

9. **return** '01'

End.

Количество операций, необходимых для получения результата у данного алгоритма равно: умножений –$4n$, вычитаний – $2n$, сложений – n.

В таблице 1 представлены свойства методов вычисления позиционной характеристики. Предложенный метод позволяет уменьшить размер операндов по сравнению с алгоритмами из работ (Chervyakov et al., 2017[17], Van, 1985 [23]).

Табл. 1. Свойства алгоритмов сравнения чисел
Table 1. Properties of number comparison methods

Метод		$\lceil\cdot\rceil$	Количество операций '×'	Размер модуля	Вид модуля
КТО (Omondi, 2007)			$4n$	$\lfloor n \cdot \log_2 p_n \rfloor$	P
Диагональная функция (Pirlo, 1993)			$2n$	$\lfloor (n-1) \cdot \log_2 p_n + \log_2 n \rfloor$	SQ
ОПСС (Isupov, 2016)			$n \cdot \frac{n-1}{2}$	$\lceil \log_2 p_n \rceil$	p_i
Приближенный метод, (Van, 1985)	$\lvert P \rvert_2 = 1$	n	$2n$	$\lceil \log_2(Pn) \rceil$	2^N
Приближенный метод, (Chervyakov, 2017)			$2n$	$\lceil \log_2(P\rho) \rceil$	2^N
Наш метод			$2n$	$\lceil \log_2(-n + n \cdot P) \rceil$	2^N
Приближенный метод, (Van, 1985)	$\lvert P \rvert_2 = 0$	n	$2n$	$\lceil \log_2(Pn) \rceil - 1$	2^N
Приближенный метод, (Chervyakov, 2017)			$2n$	$\lceil \log_2(P\rho) \rceil - 1$	2^N
Наш метод			$4n$	$\lceil \log_2(-n + n \cdot P_n) \rceil$	2^N

8. Заключение

В статье рассмотрены методы сравнения чисел, представленных в СОК, что особенно важно в задачах цифровой обработки сигналов.

Из табл. 1 видно, что самым быстрым является предложенный модифицированный приближенный метод. Худший результат показал алгоритм, основанный на ОПСС.

Оставшиеся методы показали схожие результаты и их применение зависит непосредственно от решаемой задачи. Диагональная функция требует нахождения остатка по меньшему модулю, однако в случае равенства значений диагональных функций требуется дополнительное уточнение, что занимает дополнительное время.

Список литературы/References

[1]. Chang C.H., Molahosseini A.S., Zarandi A.A.E., Tay T.F. Residue number systems: A new paradigm to datapath optimization for low-power and high-performance digital signal processing applications. IEEE circuits and systems magazine, vol. 15? № 4, 2015, pp. 26-44.

[2]. 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.

[3]. Sousa L., Antao S., Martins P. Combining residue arithmetic to design efficient cryptographic circuits and systems. IEEE Circuits and Systems Magazine, vol. 16, № 4, 2016, pp. 6-32.

[4]. Chervyakov N.I., Lyakhov P.A., Babenko M. Digital filtering of images in a residue number system using finite-field wavelets. Automatic Control and Computer Sciences, vol. 48, № 3, 2014, pp. 180-189.

[5]. Ye R., Boukerche A., Wang H., Zhou X., Yan B. RESIDENT: a reliable residue number system-based data transmission mechanism for wireless sensor networks. Wireless Networks, vol. 24, № 2, 2018, pp. 597-610.

[6]. Tchernykh A., Schwiegelsohn U., Talbi E. G., Babenko M. Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability. Journal of Computational Science, 2016 (in Press), DOI: 10.1016/j.jocs.2016.11.011.

[7]. Miranda-López V., Tchernykh A., Cortés-Mendoza J.M., Babenko M., Radchenko G., Nesmachnow S., Du Z. Experimental Analysis of Secret Sharing Schemes for Cloud Storage Based on RNS. In Proc. of the Latin American High Performance Computing Conference, 2017, pp. 370-383.

[8]. Tchernykh A., Babenko M., Chervyakov N., Cortés-Mendoza J. M., Kucherov N., Miranda-López V., Deryabin M., Dvoryaninova I., Radchenko G. Towards mitigating uncertainty of data security breaches and collusion in cloud computing. In Proc. of the 28th International Workshop on Database and Expert Systems Applications (DEXA), 2017, pp. 137-141.

[9]. Babenko M., Chervyakov N., Tchernykh A., Kucherov N., Shabalina M., Vashchenko I., Radchenko G., & Murga D. Unfairness correction in P2P grids based on residue number system of a special form. In Proc. of the 28th International Workshop on Database and Expert Systems Applications (DEXA), 2017, pp. 147-151.

[10]. Szabo N.S., Tanaka R.I. Residue arithmetic and its applications to computer technology. N.Y., McGraw-Hill, 1967, 236 p.

[11]. Bi S., Gross W.J. The mixed-radix Chinese remainder theorem and its applications to residue comparison. IEEE Transactions on Computers, vol. 57. № 12, 2008, pp. 1624-1632.

[12]. Wang Y. Residue-to-binary converters based on new Chinese remainder theorems. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 47, № 3, 2000, pp. 197-205.

[13]. Dimauro G., Impedovo S., Pirlo G. A new technique for fast number comparison in the residue number system. IEEE transactions on computers, vol. 42, № 5, 1993, pp. 608-612.

[14]. Burgess N. Scaling an RNS number using the core function. In Proc. of the 16th IEEE Symposium on Computer Arithmetic, 2003. pp. 262-269.

[15]. Dimauro G., Impedovo S., Modugno R., Pirlo G., Stefanelli R. Residue-to-binary conversion by the" quotient function". IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing. vol. 50. № 8, 2003, pp. 488-493.

[16]. Pirlo G., Impedovo D. A new class of monotone functions of the residue number system. International Journal of Mathematical Models and Methods in Applied Sciences, vol. 7. № 9, 2013, pp. 803-809.

[17]. Chervyakov N.I., Molahosseini A.S., Lyakhov P.A., Babenko M.G., Deryabin M.A. Residue-to-binary conversion for general moduli sets based on approximate Chinese remainder theorem. International Journal of Computer Mathematics, vol. 94. № 9, 2017, pp.1833-1849.

[18]. 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, vol. 61, № 6, 2014, pp.1687-1700.

[19]. Phatak D.S., Houston S.D. New distributed algorithms for fast sign detection in residue number systems (RNS). Journal of Parallel and Distributed Computing, vol. 97, issue C, 2016, pp. 78-95.

[20]. Акушский И.Я., Юдицкий Д.И. Машинная арифметика в остаточных классах. М., Советское радио, 1968, 440 с. / Akushsky I. Ya., Yuditsky D. I. Computer arithmetic in residual classes. Moscow, Soviet Radio, 1968, 440 p. (in Russian).

[21]. Omondi A.R., Premkumar B. Residue number systems: theory and implementation. L., Imperial College Press, 2007, 296 p.

[22]. Isupov K. An Algorithm for Magnitude Comparison in RNS based on Mixed-Radix Conversion II. International Journal of Computer Applications, vol. 141, № 5, 2016.

[23]. Van Vu T. Efficient implementations of the Chinese remainder theorem for sign detection and residue decoding. IEEE Transactions on Computers, vol. 100, № 7, 1985, pp. 646-651.

[24]. Mohan P.A. RNS to binary conversion using diagonal function and Pirlo and Impedovo monotonic function. Circuits, Systems, and Signal Processing, vol. 35, № 3, 2016, pp. 1063-1076.

Информация об авторах / Information about authors

Михаил Григорьевич БАБЕНКО окончил Ставропольский государственный университет в 2007 году. Защитил кандидатскую диссертацию в 2011 г. Преподаватель кафедры прикладной математики и математического моделирования Северо-Кавказского федерального университета. Сфера научных интересов: алгебраические структуры в полях Галуа, модулярная арифметика, нейрокомпьютерные технологии, цифровая обработка сигналов, криптографические методы защиты информации.

Mikhail Grigorievitch BABENKO graduated from Stavropol State University in 2007. He defended his thesis in 2011. Currently he is a lecturer of the Department of Applied Mathematics and Mathematical Modeling of the North Caucasus Federal University. Research interests: Algebraic structures in the Galois fields, modular arithmetic, neurocomputer technologies, digital signal processing, cryptographic methods for protecting information.

Андрей Николаевич ЧЕРНЫХ получил степень кандидата наук в Институте точной механики и вычислительной техники РАН. В настоящее время он является профессором Центра научных исследований и высшего образования в Энсенаде, Нижняя Калифорния, Мексика. В научном плане его интересуют многоцелевая оптимизация распределения ресурсов в облачной среде, проблемы безопасности, планирования, эвристики и метаэвристики, энергосберегающие алгоритмы, интернет вещей и т.д.

Andrei TCHERNYKH received his PhD degree at the Institute of Precision Mechanics and Computer Engineering of the Russian Academy of Sciences. Now he is holding a full professor position in computer science at CICESE Research Center, Ensenada, Baja California, Mexico. He is interesting in grid and cloud research addressing multi-objective resource optimization, both, theoretical and experimental, security, uncertainty, scheduling, heuristics and meta-heuristics, adaptive resource allocation, energy-aware algorithms and Internet of Things.

Николай Иванович ЧЕРВЯКОВ – доктор технических наук, профессор, заведующий кафедрой прикладной математики и информатики Северо-Кавказского федерального университета с 2004 года. Сфера научных интересов: алгебраические структуры в полях Галуа, модулярная арифметика, нейрокомпьютерные технологии, цифровая обработка сигналов, криптографические методы защиты информации.

Nikolay Ivanovitch CHERVYAKOV – Doctor of Technical Sciences, Professor, Head of the Department of Applied Mathematics and Computer Science of the North Caucasus Federal University since 2004. Research interests: algebraic structures in the Galois fields, modular arithmetic, neurocomputer technologies, digital signal processing, cryptographic methods for protecting information.

Виктор Андреевич КУЧУКОВ является специалистом отдела научно-технической информации, наукометрии и экспортного контроля Управления науки и технологий Северо-Кавказского федерального университета. Его научные интересы включают распознавание образов, системы остаточных классов.

Viktor Andreevich KUCHUKOV is a specialist of the department of scientific and technical information, scientometrics and export control of the Department of Science and Technology of the North Caucasus Federal University. His research interests include pattern recognition, residual class systems.

Ванесса МИРАНДА-ЛОПЕС получила степень бакалавра в области электроники в Технологическом институте Соноры, Мексика в 2006 году и степень магистра в области компьютерных наук в исследовательском центре CICESE в 2010 году. Ее интересы включают облачные вычисления, сетевое планирование, большие данные, безопасность и электронный дизайн.

Vanessa MIRANDA-LÓPEZ received a Bachelor degree in electronics engineering from Technological Institute of Sonora, Mexico in 2006, and Master degree in computer sciences from CICESE Research Center in 2010. Her interests include cloud computing, grid scheduling, big data, security and electronic design.

Рауль РИВЕРА РОДРИГЕС получил степень доктора философии в Автономном университете Нижней Калифорнии, Британская Колумбия, Мексика. В настоящее время он является директором отделения телематики в исследовательском центре CICESE, B.C., Мексика. Научные интересы включают сети связи для HPC и BigData.

Raúl RIVERA RODRÍGUEZ obtained a PhD degree from Autonomous University of Baja California, B.C., Mexico. Currently he is a Director of the Telematics Division at CICESE Research Center, B.C., Mexico. Research interests include communications networks for HPC and BigData.

Чжихуэй ДУ получил степень PhD в области компьютерных наук и технологий в Пекинском университете, КНР в 1998 г. В настоящее время он работает доцентом на факультете Компьютерных наук и технологий университета Цинхуа, КНР. В число научных интересов входят параллельное программирование, высокопроизводительные / облачные / энергосберегающие вычисления и анализ больших данных.

Zhihui DU received the degree of PhD in Computer Science & Technology from Peking University, China in 1998. Currently he is the associate professor at the Department of Computer Science and Technology of Tsinghua University, China. His research interests include parallel computing, high performance/cloud/energy efficient computing, and Big Data analysis.