УДК 004.052.2

МОДИФИКАЦИЯ АЛГОРИТМА ВЫЧИСЛЕНИЯ ПОЗИЦИОННО-ИНТЕРВАЛЬНОЙ ХАРАКТЕРИСТИКИ МОДУЛЯРНЫХ КОДОВ ДЛЯ КОРРЕКЦИИ ОШИБОК

Чистоусов Н.К., Чипига А.Ф., Калмыков И.А., Ефременков И.Д., Калмыкова Н.И.
ФГАОУ ВО «Северо-Кавказский федеральный университет», Ставрополь, e-mail: kia762@yandex.ru

Для обеспечения более высокой имитостойкости систем опознавания «свой – чужой» надо сокращать время, которое тратится на аутентификацию претендента. В этом случае у нарушителя уменьшается вероятность правильного подбора сигнала ответчика. Для достижения такой цели в ряде работ предлагалось выполнять протоколы аутентификации с использованием параллельных вычислений. В этом случае наибольшее распространение получили параллельные арифметические коды, в частности модулярные коды. Среди данных кодов особое место занимают коды системы классов вычетов (СОК). Высокая скорость вычисления достигается за счет параллельного выполнения арифметических операций. Это происходит благодаря тому, что данные операции выполняются над малыми остатками. При этом в ходе выполнения данных операций обмен промежуточных результатов между модулями кода СОК не осуществляется. Следовательно, при возникновении ошибки она не будет оказывать влияния на другие остатки кода СОК. Значит, при введении избыточных модулей код СОК можно использовать для коррекции ошибок в остатках. При этом данная процедура базируется на вычислении позиционно-интервальной характеристики (ПИХ). Очевидно, что снижение схемных затрат на вычисление ПИХ повышает эффективность алгоритма вычисления ошибок в коде СОК. Поэтому модификация алгоритма вычисления позиционно-интервальной характеристики модулярных кодов для коррекции ошибок является актуальной задачей.

Ключевые слова: протоколы опознавания «свой – чужой», параллельные модулярные коды, позиционно-интервальная характеристика, коррекция ошибок в СОК

MODIFICATION OF THE ALGORITHM FOR CALCULATING THE POSITION-INTERVAL CHARACTERISTIC OF MODULAR CODES FOR ERROR CORRECTION

Chistousov N.K., Chipiga A.F., Kalmykov I.A., Efremenkov I.D., Kalmykova N.I.
Federal State Autonomous Educational Institution Higher Professional Education «North-Caucasian Federal University», Stavropol, e-mail: kia762@yandex.ru

To ensure a higher imitability of the «friend – foe» identification systems, it is necessary to reduce the time spent on authenticating the applicant. In this case, the violator's probability of correctly selecting the respondent's signal decreases. To achieve this goal, a number of papers have proposed to perform authentication protocols using parallel computing. In this case, parallel arithmetic codes, in particular, modular codes, have become the most widespread. Among these codes, a special place is occupied by the residue number system (RNS). High calculation speed is achieved due to parallel execution of arithmetic operations. This is achieved due to the fact that these operations are performed on small balances. At the same time, during the execution of these operations, operations, the exchange of intermediate results between the RNS code modules are not carried out. This means that if an error occurs, it will not affect other remnants of the RNS code. This means that when introducing redundant modules, the RNS code can be used to correct errors in the residuals. At the same time, this procedure is based on the calculation of the position-interval characteristic (PIH). It is obvious that reducing the circuit and time costs for calculating the PIH increases the efficiency of the algorithm for calculating errors in the remnants of the SOC code. Therefore, the modification of the algorithm for calculating the positional interval characteristic of modular codes for error correction is an urgent task.

Keywords: protocols of identification «friend – foe», parallel modular codes, position-interval characteristic, error correction in the RNS

В настоящее время реализацию таких глобальных отечественных проектов, связанных с освоением природных богатств, расположенных на шельфе Северного Ледовитого океана, невозможно представить без использования низкоорбитальных систем спутниковой связи (НССС). Однако по мере увеличения числа группировок НССС возрастает вероятность навязывания приемнику спутниковой связи, который располагается на необслуживаемом объекте управления (НОУ), ретрансляционной помехи. В качестве такой помехи выступает ранее перехваченный правильный сигнал управления, который передавался с космического аппарата (КА) приемнику НОУ. Чтобы предотвратить попытку навязывания перехваченного сигнала в работе [1], предлагается использовать систему опознавания КА. Для обеспечения высокой скорости определения статуса спутника в работах [2, 3] предлагается реализовать протоколы аутентификации с использованием параллельных кодов системы остаточных классов (СОК).

Однако избыточные коды СОК можно также применять для коррекции ошибок, которые могут возникнуть при вычислениях в протоколах аутентификации. Для достижения данной цели коды СОК используют позиционно-интервальную характеристику (ПИХ). Поэтому модификация алгоритма вычисления позиционной интервальной характеристики модулярных кодов для коррекции ошибок является актуальной задачей.

Основным достоинствам кодов СОК является параллельное выполнение модульных аддитивных и мультипликативных операций [4, 5]. Высокая скорость достигается за счет отсутствия обмена промежуточными результатами между основаниями СОК. Тогда ошибка, возникшая в остатке по одному основанию, не будет оказывать влияния на другие остатки кода СОК. Это свойство используется при построении избыточных модулярных кодов, в которых для коррекции ошибочного остатка необходимо вычислить позиционно-интервальную характеристику.

Целью работы является снижение схемных затрат на коррекцию ошибочного остатка СОК за счет модификации алгоритма вычисления ПИХ.

Материалы и методы исследования

Основные принципы построения кодов СОК достаточно полно раскрыты в работах [4–6]. В коде СОК кодовая комбинация представляет собой кортеж остатков:

$$Y = (y_1, y_2, ..., y_k), \quad (1)$$

где $Y \equiv y_i \bmod m_i$; m_i – основание кода СОК; $\underset{i \neq j}{НОД}(m_i, m_j) = 1$; $i = 1, ..., k$.

Произведение оснований кода СОК определяет размер рабочего диапазона:

$$M^k = \prod_{i=1}^{k} m_i . \quad (2)$$

Так как коды СОК представлены в алгебраической системе «кольцо», то справедливо:

$$C + Y = ((c_1 + y_1) \bmod m_1, (c_2 + y_2) \bmod m_2, ..., (c_k + y_k) \bmod m_k), \quad (3)$$

$$C - Y = ((c_1 - y_1) \bmod m_1, (c_2 - y_2) \bmod m_2, ..., (c_k - y_k) \bmod m_k), \quad (4)$$

$$C \cdot Y = ((c_1 \cdot y_1) \bmod m_1, (c_2 \cdot y_2) \bmod m_2, ..., (c_k \cdot y_k) \bmod m_k), \quad (5)$$

где $C \equiv c_i \bmod m_i$; $\{C, Y\} < M^k$; $i = 1, ..., k$.

В протоколе аутентификации [3], который обеспечивает минимальное время опознавания КА за счет кода СОК, используются: W – секретный ключ КА, а также числа C и B для генерации и проверки времени применения n-го сеансового ключа C(n) и B(n), где $n = 1, 2, ...$. Протокол представлен в таблице.

Протокол аутентификации в коде СОК

Ответчик		Запросчик
Секретные параметры в СОК $W = \{W_i\}, C = \{C_i\}, B = \{B_i\}$	$m_1, ..., m_k$ – модули $\{W, C, B\} < M^k$	
$U_1(n) = \left\| a_1^{W_1(n)} a_1^{C_1(n)} a_1^{B_1(n)} \right\|_{m_i}^{+}$	Исходный статус КА	
$W_i^*(n) = \left\| W_i + \Delta W_i(n) \right\|_{\varphi(m_i)}^{+}$, $C_i^*(n) = \left\| C_i + \Delta C_i(n) \right\|_{\varphi(m_i)}^{+}$, $B_i^*(n) = \left\| B_i + \Delta B_i(n) \right\|_{\varphi(m_i)}^{+}$.	Изменение секретных параметров протокола	
$U_i^*(n) = \left\| a_1^{W_i^*(n)} a_1^{C_i^*(n)} a_1^{B_i^*(n)} \right\| \bmod m_i$	Измененный статус	
	Вопрос	$H = (H_1(n), ..., H_k(n))$

Окончание таблицы

Ответчик		Запросчик
$\begin{cases} Q_i^1(n) = \left\|W_i^*(n) - H_i W_i(n)\right\|_{\varphi(m_i)}^+, \\ Q_i^2(n) = \left\|C_i^*(n) - H_i C_i(n)\right\|_{\varphi(m_i)}^+, \\ Q_i^3(n) = \left\|B_i^*(n) - H_i B_i(n)\right\|_{\varphi(m_i)}^+. \end{cases}$	Вычисляются ответы	
	Проверка ответов	$R_i(n) = \left\|U_i^{H_i(n)} a_i^{Q_i^1(n)} a_i^{Q_i^2(n)} a_i^{Q_i^3(n)}\right\|_{m_i}^+$ $\left\{R_i(n)\right\} = \left\{U_i^*(n)\right\}$ – КА «свой»

Использование модулярных кодов позволяет повысить скорость проводимых вычислений. В работе [5] показано, что для выполнения операции умножения двух 128-битовых операндов максимально требуется 7 CPU Clock Cycles (тактов центрального процессора). Если исходные данные представить в виде 4×32бит, то потребуется 5 тактов CPU Clock Cycles, что в 1,4 раза меньше. Таким образом, очевидно, что применение модулярных кодов позволяет сократить время на опознавание спутника.

Для построения кодов СОК, способных исправлять ошибки в одном остатке, используются два контрольных основания [6–9], для которых истинно неравенство:

$$m_{k+2} \cdot m_{k+1} > m_k \cdot m_{k-1}. \quad (6)$$

В результате получается полный диапазон кода СОК:

$$M^{k+2} = \prod_{i=1}^{k+2} m_i = M^k m_{k+1} m_{k+2}. \quad (7)$$

Тогда комбинация кода СОК считается разрешенной, если выполнено условие:

$$Y=(y_1, y_2, ..., y_k, y_{k+1}, y_{k+2}) < M^k. \quad (8)$$

Для проверки условия (8) можно использовать обратное преобразование из СОК в позиционный код (СОК-ПК) с использованием Китайской теоремы об остатках (КТО):

$$Y = \sum_{i=1}^{k+2} y_i B_i^{k+2} \bmod M^{k+2}, \quad (9)$$

где $B_i^{k+2} = u_i M^{k+2} m_i^{-1}$ – ортогональный базис для кода СОК, состоящего из k+2 оснований; $B_i^{k+2} \equiv 1 \bmod m_i$; u_i – вес ортогонального базиса; $u_i (M^{k+2} p_i^{-1}) \equiv 1 \bmod m_i$; $i = 1, ..., k$.

Условие (8) показывает, что для исправления ошибочного остатка по одному из оснований необходимо определить позицию числа $Y = (y_1, y_2, ..., y_{k+2})$ относительно рабочего диапазона. Поэтому для достижения поставленной цели в кодах СОК используются позиционно-интервальные характеристики. Непозиционные избыточные коды СОК характеризуются многовариантностью по использованию ПИХ. В работе [8] для исправления ошибки в одном основании предлагается использовать метод проекции, в котором поочередно из избыточной комбинации СОК удаляются остатки. Полученную комбинацию переводят в позиционный код и сравнивают с M^k. Условие (8) будет выполнено только при удалении ошибочного остатка. В качестве недостатка метода проекции можно отметить последовательный характер определения ошибочного остатка. Кроме того, необходимо выполнять перевод СОК-ПК при изменяющемся кортеже оснований, что требует пересчета ортогональных базисов для выполнения (9). В работе [9] предлагается использовать вычисления следа числа. Для определения ошибочного остатка необходимо из кодограммы СОК последовательно в течение k итераций вычитать константы нулевизации. Если код СОК не содержит ошибки, то в конечном результате должна получиться нулевая комбинация. В противном случае вычисленные остатки $\hat{y}_{k+1} \neq 0, \hat{y}_{k+2} \neq 0$. По величине полученного следа можно однозначно определить и исправить ошибочный остаток. Однако данный алгоритм не может быть реализован параллельно, так выбор констант нулевизации зависит от результата, полученного на предыдущей итерации.

Устранить данные недостатки можно с помощью алгоритма вычисления ПИХ, приведенного в работе [10]. Однако он обладает недостатком – вычисления ПИХ проводятся по большему модулю $\tilde{M}^{k+2} = m_{k+1}m_{k+2}$, что увеличивает схемные затраты. Для устранения данного недостатка проведем его модификацию.

Для проверки выполнения условия (8) найдем ПИХ, используя выражение:

$$D = \left[\frac{Y}{M^k}\right] = \left[\frac{\sum_{i=1}^{k+2} y_i B_i^{k+2} \bmod M^{k+2}}{M^k}\right] = \left[\frac{\sum_{i=1}^{k+2} y_i B_i^{k+2} - c_Y^{k+2} M^{k+2}}{M^k}\right], \quad (10)$$

где [ ] – целая вычисленного частного; c_Y^{k+2} – позиционная характеристика ранг числа Y, которая определяет количество переходов за M^{k+2} при выполнении перевода СОК-ПК.

В работе [7] доказано, что ортогональные базисы для полного B_i^{k+2} и рабочего B_i^k диапазонов подобны. Следовательно, для них справедливо:

$$B_i^{k+2} \equiv B_i^k \bmod M^k, \quad (11)$$

где $i = 1, ..., k$.

На основе свойства (11) справедливо равенство:

$$B_i^{k+2} = \left[\frac{B_i^{k+2}}{M^k}\right] + B_i^k = K_i M^k + B_i^k. \quad (12)$$

В этом случае выражение (10) можно представить:

$$D = \left[\frac{\sum_{i=1}^{k+2} y_i (K_i M^k + B_i^k) - c_Y^{k+2} M^{k+2}}{M^k}\right] = \left[\sum_{i=1}^{k+2} y_i K_i + c_y^k - c_y^{k+2} m_{k+1} m_{k+2}\right], \quad (13)$$

где $c_y^k = \left[\frac{\sum_{i=1}^{k} y_i B_i^k}{M^k}\right]$ – ранг числа Y в коде СОК, состоящего из k оснований.

Позиционно-интервальная характеристика, определяемая (10), может принимать значения от 0 до $\tilde{M}^{k+2} - 1$. Значит, используя изоморфизм, порожденный КТО, можно перейти от вычислений по одному модулю $\tilde{M}^{k+2}$ к параллельным вычислениям по контрольным основаниям m_{k+1}, m_{k+2}. Тогда:

$$\begin{aligned} D_1 &= \left(\sum_{i=1}^{k+2} y_i K_i + c_y^k - c_y^{k+2} m_{k+1} m_{k+2}\right) \bmod m_{k+1} = \left(\sum_{i=1}^{k+2} y_i K_i + c_y^k\right) \bmod m_{k+1}, \\ D_2 &= \left(\sum_{i=1}^{k+2} y_i K_i + c_y^k - c_y^{k+2} m_{k+1} m_{k+2}\right) \bmod m_{k+2} = \left(\sum_{i=1}^{k+2} y_i K_i + c_y^k\right) \bmod m_{k+2}. \end{aligned} \quad (14)$$

Для выполнения (13) используется LUT-таблица, содержащая $N_1 = \tilde{M}^{k+2} \times \tilde{M}^{k+2}$ ячеек памяти запоминающей матрицы. При использовании модифицированного алгоритма вычисления ПИХ потребуется $N_2 = \sum_{i=k+1}^{k+2} m_i^2$. Очевидно, что $N_2 < N_1$.

Результаты исследования и их обсуждение

Пусть имеем информационные модули $m_1 = 7$, $m_2 = 17$, $m_3 = 19$ и два контрольных $m_4 = 29$, $m_5 = 31$. Диапазоны кода СОК: рабочий $M^k = 2261$, полный $M^{k+2} = 2032639$, контрольный $\tilde{M}^{k+2} = 899$. Для данных оснований получаем ортогональные базисы:

$$B_1^5 = \frac{u_1 M^5}{m_1} = \frac{5 \cdot 2032639}{7} = 1451885, \quad B_2^5 = \frac{u_2 M^5}{m_2} = \frac{3 \cdot 2032639}{17} = 358701,$$

$$B_3^5 = \frac{u_3 M^5}{m_3} = 748867, \quad B_4^5 = \frac{u_4 M^5}{m_4} = 981274, \quad B_5^5 = \frac{u_5 M^5}{m_5} = 524552.$$

Представим их согласно выражению (12):

$$B_1^5 = K_1 M^3 + B_1^3 = 642M^3 + 323, \quad B_2^5 = K_2 M^3 + B_2^3 = 158M^3 + 1463,$$

$$B_3^5 = K_3 M^3 + B_3^3 = 331M^3 + 476, \quad B_4^5 = K_4 M^3 = 434M^3, \quad B_5^5 = K_5 M^3 = 232M^3.$$

Пусть на вход блока вычисления ПИХ подается код $Y = (5,14,11,27,15) < M^3$. Тогда ранг числа Y по информационным модулям равен:

$$c_Y^3 = \left[\sum_{i=1}^{3} y_i B_i^k \Big/ M^3 \right] = \left[(5 \cdot 323 + 14 \cdot 1463 + 11 \cdot 476)/2261 \right] = 12.$$

Тогда ПИХ равна:

$$D_1 = \left| \sum_{i=1}^{5} y_i K_i + c_y^3 \right|_{m_4}^{+} = \left| 5 \cdot 642 + 14 \cdot 158 + 11 \cdot 331 + 27 \cdot 434 + 15 \cdot 232 + 12 \right|_{29} = 0.$$

$$D_2 = \left| \sum_{i=1}^{5} y_i K_i + c_y^3 \right|_{m_5}^{+} = \left| 5 \cdot 642 + 14 \cdot 158 + 11 \cdot 331 + 27 \cdot 434 + 15 \cdot 232 + 12 \right|_{31} = 0.$$

Так как ПИХ равна нулю, то комбинация является разрешенной, т.е. $Y < M^k$.

Пусть в процессе выполнения протокола аутентификации произошла ошибка и на вход блока ПИХ поступила комбинация $Y^* = (0^*,14,11,27,15)$. Тогда ранг числа Y* равен:

$$c_{Y^*}^3 = \left[\sum_{i=1}^{3} y_i B_i^k \Big/ M^3 \right] = \left[(0 \cdot 323 + 14 \cdot 1463 + 11 \cdot 476)/2261 \right] = 11.$$

Тогда ПИХ равна

$$D_1 = \left| \sum_{i=1}^{5} y_i K_i + c_y^3 \right|_{m_4}^{+} = \left| 0 \cdot 642 + 14 \cdot 158 + 11 \cdot 331 + 27 \cdot 434 + 15 \cdot 232 + 12 \right|_{29} = 8.$$

$$D_2 = \left| \sum_{i=1}^{5} y_i K_i + c_y^3 \right|_{m_5}^{+} = \left| 0 \cdot 642 + 14 \cdot 158 + 11 \cdot 331 + 27 \cdot 434 + 15 \cdot 232 + 12 \right|_{31} = 13.$$

Получили: ПИХ равен $D = (D_i, D_2) = (8,13) = 385$), где $D \equiv D_i \bmod m_i$, i = 4,5.

Воспользуемся алгоритмом вычисления ПИХ [10]. Получаем:

$$D = \left| \sum_{i=1}^{5} y_i K_i + c_y^3 \right|_{\tilde{M}^5}^{+} = \left| 0 \cdot 642 + 14 \cdot 158 + 11 \cdot 331 + 27 \cdot 434 + 15 \cdot 232 + 12 \right|_{899}^{+} = 385.$$

Для этого значения ПИХ вектор ошибки $\Delta_{корр} = (2,0,0,0,0)$. Коррекция дает:

$$Y = Y^* - \Delta_{корр} = (0,14,11,27,15) - (2,0,0,0,0) = (5,14,11,27,15).$$

Модифицированный алгоритм вычисления ПИХ показал аналогичный результат по сравнению с прототипом, для которого требуется $N_1 = \tilde{M}^5 \times \tilde{M}^5 = 808201$ ячеек памяти LUT-таблицы. При этом для модифицированного алгоритма вычисления ПИХ используется $N_2 = \sum_{i=4}^{5} m_i^2 = 1802$ ячеек памяти запоминающей матрицы LUT-таблицы. Очевидно, что поставленная цель, направленная на снижение схемных затрат, достигнута.

Заключение

В статье показан протокол аутентификации спутника, реализованный в коде СОК. Применение данного кода за счет параллельных вычислений позволяет не только уменьшить время на опознавание КА, но и корректировать ошибки, которые могут возникнуть в процессе работы системы «свой – чужой» для НССС. С целью снижения схемных затрат на реализацию запросно-ответной системы была проведена модификация алгоритма вычисления ПИХ. Поставленная цель достигнута благодаря изоморфизму Китайской теоремы об остатках. В статье рассмотрен пример реализации модифицированного алгоритма вычисления ПИХ. Полученные результаты совпали с результатами прототипа, для которого требуется $N_1 = 808201$ ячеек памяти LUT-таблицы. При этом для модифицированного алгоритма вычисления ПИХ используется $N_2 = 1802$ ячеек памяти запоминающей матрицы LUT-таблицы. Очевидно, что поставленная цель, направленная на снижение схемных затрат, достигнута.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-37-90009.

Список литература

1. Пашинцев В.П., Ляхов А.В. Применение помехоустойчивого протокола аутентификации космического аппарата для низкоорбитальной системы спутниковой связи // Инфокоммуникационные технологии. 2015. № 2. С. 183–190.

2. Калмыков И.А., Чипига А.Ф. Алгоритм коррекции ошибок, возникающих при вычислении ответа на запрос отказоустойчивой системы опознавания спутника // Современные наукоемкие технологии. 2021. № 3. С. 96–100. DOI: 10.17513/snt.38069.

3. Чистоусов Н.К., Калмыкова Н.И. Разработка протоколов аутентификации низкоорбитальных космических аппаратов на основе параллельных кодов систем остаточных классов // Инженерный вестник Дона. 2021. № 4. [Электронный ресурс]. URL: ivdon.ru/ru/magazine/archive/n4y2021/6912 (дата обращения: 10.08.2021).

4. Червяков Н.И., Коляда А.А., Ляхов П.А. Модулярная арифметика и ее приложения в инфокоммуникационных технологиях. М.: ФИЗМАТЛИТ, 2017. 400 с.

5. Рокотянский М. Стоимость операций в тактах ЦП. [Электронный ресурс]. URL: https://habr.com/ru/company/otus/blog/343566 (дата обращения: 10.08.2021).

6. Ananda Mohan Residue Number Systems. Theory and Applications. Springer International Publishing Switzerland, 2016. 351 p.

7. Omondi A., Premkumar B. Residue Number Systems: Theory and Implementation. Imperial College Press. UK, 2007. 296 p.

8. Yones D. Comparative Study on Different Moduli Sets in RNS. International Conference on Computer System and Industrial Informatics (ICCSII), 2012. P. 1–6.

9. Mohan P.V. Residue Number Systems. Algorithms and Architectures. Springer Science + Business Media New York, 2002. 254 p.

10. Бабенко Л.К., Ефременков И.Д., Мирошников Д.А. Использование избыточных модулярных кодов при разработке отказоустойчивых запросно-ответных систем распознавания спутника // Фундаментальные исследования. 2017. № 12–2. С. 292–296.