УНИВЕРСИТЕТ ИТМО
НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
июль–август 2020 Том 20 № 4 ISSN 2226-1494 http://ntv.itmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
July–August 2020 Vol. 20 No 4 ISSN 2226-1494 http://ntv.itmo.ru/en/
НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

УДК 004.056.55 doi: 10.17586/2226-1494-2020-20-4-539-544

ЭФФЕКТИВНАЯ РЕАЛИЗАЦИЯ СОВРЕМЕННОЙ КРИПТОСИСТЕМЫ МАК-ЭЛИСА, ПОСТРОЕННОЙ НА ОБОБЩЕННЫХ (*L*, *G*)-КОДАХ

И.К. Носков[a], С.В. Беззатеев[a,b]

[a] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация
[b] Санкт-Петербургский государственный университет аэрокосмического приборостроения (ГУАП), Санкт-Петербург, 190000, Российская Федерация
Адрес для переписки: vanya170595@gmail.com

Информация о статье
Поступила в редакцию 28.05.20, принята к печати 29.06.20
Язык статьи — русский

Ссылка для цитирования: Носков И.К., Беззатеев С.В. Эффективная реализация современной криптосистемы Мак-Элиса, построенной на обобщенных (*L*, *G*)-кодах // Научно-технический вестник информационных технологий, механики и оптики. 2020. Т. 20. № 4. С. 539–544. doi: 10.17586/2226-1494-2020-20-4-539-544

Аннотация

Предмет исследования. Исследованы способы и подходы к реализации современной криптосистемы Мак-Элиса, построенной на сепарабельных обобщенных (*L*, *G*)-кодах. **Метод.** На основе анализа известных общедоступных источников по реализации современной криптосистемы Мак-Элиса предложен метод, который позволяет использовать обобщенные (*L*, *G*)-коды с нумераторами степени больше или равной второй без использования расширенного поля. **Основные результаты.** Разработаны подходы к реализации современной криптосистемы Мак-Элиса, построенной на обобщенных (*L*, *G*)-кодах, а именно: построение проверочной матрицы обобщенного (*L*, *G*)-кода, использующего сепарабельный многочлен Гоппы и нумераторы различных степеней; описан подход к реализации шифрования и расшифрования сообщений в современной криптосистеме Мак-Элиса; описан способ использования процедуры Ченя для нумераторов степени второй и выше без расширения поля. **Практическая значимость.** Предложенные способы можно применять в разработке криптографических систем, способных противостоять атакам с использованием квантовых компьютеров, что позволит обеспечить конфиденциальность данных, а также улучшить безопасность и производительность криптосистем. Областью применения результатов работы также могут являться аэрокосмические, автомобильные, железнодорожные, сетевые мультимедийные, телекоммуникационные и мобильные системы защиты информации.

Ключевые слова
коды Гоппа, обобщенные (*L*, *G*)-коды, алгоритмы декодирования, современная криптосистема Мак-Элиса, сепарабельный полином

doi: 10.17586/2226-1494-2020-20-4-539-544

EFFECTIVE IMPLEMENTATION OF MODERN MCELIECE CRYPTOSYSTEM ON GENERALIZED (*L*, *G*)-CODES

I.K. Noskov[a], S.V. Bezzateev[a,b]

[a] ITMO University, Saint Petersburg, 197101, Russian Federation
[b] Saint Petersburg State University of Aerospace Instrumentation (SUAI), Saint Petersburg, 190000, Russian Federation
Corresponding author: vanya170595@gmail.com

Article info
Received 28.05.20, accepted 29.06.20
Article in Russian

For citation: Noskov I.K., Bezzateev S.V. Effective implementation of modern McEliece cryptosystem on generalized (*L*, *G*)-codes. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2020, vol. 20, no. 4, pp. 539–544 (in Russian). doi: 10.17586/2226-1494-2020-20-4-539-544

Abstract

Subject of Research. The paper presents the study of methods and approaches to implementation of the modern McEliece cryptosystem based on separable generalized (*L*, *G*)-codes. **Method.** A method is proposed based on the analysis of the well-known public sources on implementation of the modern McEliece cryptosystem that uses the

generalized (L, G)-codes with locators of degree greater than or equal to the second one without using an extended field. **Main Results.** Approaches to implementation of the modern McEliece cryptosystem based on the generalized (L, G)-codes are developed, namely: creation of a parity check matrix for the generalized (L, G)-code using a separable Goppa polynomial and locators of various degrees, description of an approach to the implementation of encryption and decryption of messages in the modern McEliece cryptosystem, presentation of the Chein's procedure for numerators of degree greater than or equal to the second one without expanding the field. **Practical Relevance.** The proposed methods can be used in the development of cryptographic systems that can withstand attacks from quantum computers and ensure data confidentiality, as well as improve the security and performance of cryptosystems. Aerospace, automobile, railway, network multimedia, telecommunication and mobile information protection systems can also be the scope of the work results.

Keywords
Goppa codes, generalized (L, G)-codes, decoding algorithms, modern McEliece cryptosystem, separable polynomial

Введение

В настоящее время из-за активного развития квантовых компьютеров возникла потребность в постквантовой криптографии. Исследования показали, что для противодействия атакам, использующим квантовые вычисления, размер ключа в криптографических системах должен быть существенно увеличен, а многие известные криптосистемы вообще не могут быть использованы в условиях существования квантового компьютера. Некоторые криптосистемы, построенные для постквантовой криптографии, используют двоичные неприводимые коды Гоппы [1], что является проблемой, связанной с ограничением длины кода для выбранного порядка поля.

Для решения данной проблемы можно использовать обобщенные (L, G)-коды [2]. Проблема заключается в том, что для обобщенных (L, G)-кодов плохо изучены проблемы реализации, а именно: построение нумераторов второй степени и выше, алгоритм декодирования Паттерсона для сепарабельных многочленов и процедура Ченя для декодирования. В данной статье рассмотрены возможности использования обобщенных (L, G)-кодов с нумераторами второй степени и выше без вычислений в расширении поля, использования процедуры Ченя для данной реализации, а также подход к реализации шифрования и расшифрования в современной криптосистеме Мак-Элиса.

Современный вариант криптосистемы Мак-Элиса

Для начала необходимо ввести несколько определений.

Определение 1 [1]. Двоичный вектор $\mathbf{a} = (a_1, a_2, ..., a_n)$ является кодовым словом кода Гоппы для множества локаторов $L = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ и для многочлена $G(x)$, который называется многочленом Гоппы, тогда и только тогда, когда выполняется сравнение

$$\sum_{i=1}^{n} a_i \frac{1}{x-\alpha_i} \equiv 0 \bmod G(x),$$

где $\alpha_i \in GF(2^m)$, $n \le 2^m$, $G(x) \in F_{2^m}[x]$, $\deg G(x) = t$.

Определение 2 [1]. Код Гоппы называется сепарабельным, если $G(x)$ — сепарабельный многочлен.

Определение 3 [1]. Код Гоппы называется неприводимым, если $G(x)$ — неприводимый многочлен.

Утверждение 1 [1]. Сепарабельный двоичный код Гоппы имеет избыточность r и минимальное расстояние d, определяемые неравенствами:

$$r \le mt,\ d \ge 2t + 1.$$

Определение 4 [2]. Двоичный вектор $\mathbf{a} = (a_1, a_2, ..., a_n)$ является кодовым словом обобщенного (L, G)-кода для множества локаторов $L = \left\{ \frac{f_1'(x)}{f_1(x)}, \frac{f_2'(x)}{f_2(x)}, \ldots, \frac{f_n'(x)}{f_n(x)} \right\}$, где $f_i'(x)$ — формальная производная для $f_i(x)$, тогда и только тогда, когда выполняется сравнение

$$\sum_{i=1}^{n} a_i \frac{f_i'(x)}{f_i(x)} \equiv 0 \bmod G(x). \quad (1)$$

Расстояние такого кода определяется неравенством [2, 3]:

$$d_G \ge \frac{2t+1}{l},\ l = \max \deg f_i(x).$$

Чтобы построить проверочную матрицу для обобщенного (L, G)-кода, рациональную функцию можно представить в виде [4]:

$$\frac{f_i'(x)}{f_i(x)} \equiv s_i(x) \equiv b_{i,t-1}x^{t-1} + b_{i,t-2}x^{t-2} + \ldots + b_{i,1}x^1 + b_{i,0} \bmod G(x),\ b_{i,j} \in GF(2^m).$$

Тогда уравнение (1) для обобщенного (L, G)-кода может быть переписано в форме:

$$\sum_{i=1}^{n} a_i \frac{f_i'(x)}{f_i(x)} = \sum_{i=1}^{n} a_i s_i(x) \equiv \sum_{i=1}^{n} a_i b_{i,t-1}x^{t-1} + \sum_{i=1}^{n} a_i b_{i,t-2}x^{t-2} + \ldots + \sum_{i=1}^{n} a_i b_{i,1}x^1 + \sum_{i=1}^{n} a_i b_{i,0} \equiv 0 \bmod G(x), \quad (2)$$

а проверочная матрица будет выглядеть следующим образом:

$$\mathbf{H} = \begin{bmatrix} b_{1,t-1} & b_{2,t-1} & \cdots & b_{n,t-1} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1,0} & b_{2,0} & \cdots & b_{n,0} \end{bmatrix}. \quad (3)$$

Очевидно, что для любого кодового слова $\mathbf{a} = (a_1, a_2, ..., a_n)$ обобщенного (L, G)-кода определяемого сравнениями (1) и (2) будет выполняться соотношение

$$\mathbf{H} \cdot \mathbf{a}^T = 0. \quad (4)$$

Здесь матрица **H** состоит из элементов $b_{i,j} \in GF(2^m)$. Двоичная проверочная матрица получается путем представления каждого элемента $b_{i,j}$ в виде двоичного вектор-столбца из m бит. Таким образом, полученная двоичная матрица **H** будет состоять из mt строк и n столбцов.

Для построения современной криптосистемы Мак-Элиса [5, 6], использующей идеи классического алгоритма Мак-Элиса [7] и алгоритма Нидеррайтера [8], выбирается многочлен Гоппы $G(x)$ и множество локаторов L в качестве секретного ключа. По выбранным $G(x)$ и L строится проверочная матрица **H**. Используя метод Гаусса проверочная матрица **H** приводится к диагональному виду $\hat{\mathbf{H}}$

$$\hat{\mathbf{H}} = [\mathbf{E} \quad \mathbf{T}],$$

где **E** — единичная матрица, а матрица **T** является открытым ключом системы шифрования.

Реализация шифрования. На вход функции шифрования подается сообщение **b** длины 256 бит, которое требуется зашифровать, и матрица **T**. Алгоритм шифрования состоит из следующих шагов:

1) генерируется случайный двоичный вектор-столбец ошибки **e** длины n и веса Хэмминга $\tau = (d_G - 1)/2$;
2) строится матрица $\hat{\mathbf{H}} = [\mathbf{E} \quad \mathbf{T}]$;
3) находится двоичный вектор $\mathbf{c_0}$ длины mt

$$\mathbf{c_0} = \hat{\mathbf{H}}\mathbf{e};$$

4) вычисляется результат хеш-функции (например, SHA256) **h** длины 256 бит от вектора **e**

$$\mathbf{h} = Hash(\mathbf{e});$$

5) находится результат побитового сложения XOR вектора **h** и сообщения **b**

$$\mathbf{R} = \mathbf{b} \oplus \mathbf{h}.$$

Результатом шифрования будут вектора **R** и $\mathbf{c_0}$.

Реализация расшифрования. На вход функции расшифрования подаются двоичный вектор $\mathbf{c_0}$ длины mt бит, вектор **R** длины 256 бит, а также секретный ключ, который состоит из множества локаторов L и многочлена Гоппы $G(x)$. Алгоритм расшифрования состоит из следующих шагов:

1) находится вектор **v** длины n: $\mathbf{v} = (\mathbf{c_0}, 0, 0, \ldots, 0)$, где количество добавляемых нулей равно $k = n - mt$;
2) строится проверочная матрица **H**, используя соотношение (3);
3) находится синдром $\mathbf{s} = \mathbf{H}\mathbf{v}^T$;
4) для полученного синдрома находится многочлен локаторов позиций ошибок по одному из известных алгоритмов (Берлекэмпа–Месси [9, 10], расширенному алгоритму Евклида [11], Паттерсона [12]) с учетом не единичной степени локаторов позиций [3], и далее находится вектор **e** длины n с помощью процедуры Ченя [13];
5) вычисляется результат хеш-функции длины 256 бит от вектора **e**

$$\mathbf{h} = Hash(\mathbf{e});$$

6) находится результат битового сложения XOR вектора **h** и вектора **R**

$$\mathbf{b} = \mathbf{R} \oplus \mathbf{h}.$$

Результатом расшифрования будет вектор **b**.

Приведем более подробно схему работы алгоритма расшифрования. Легко увидеть, что значение синдрома $\mathbf{s} = \mathbf{H}\mathbf{v}$ равно $\mathbf{c_0}$, поскольку первые $n - k$ позиций вектора $\mathbf{v} = (\mathbf{c_0}, 0, 0, \ldots, 0)$ умножаются на единичную матрицу, а остальные позиции равны нулю. Так как $\mathbf{c_0} = \mathbf{H}\mathbf{e}$, где вес Хэмминга вектора **e** равен τ и $\mathbf{s} = \mathbf{H}\mathbf{v} = \mathbf{c_0} = \mathbf{H}\mathbf{e}$, следовательно, $\mathbf{v} = \mathbf{c} \oplus \mathbf{e}$, где **c** является кодовым словом и для него выполняется соотношение (4). Это кодовое слово находится на расстоянии Хэмминга равным τ от **v**, и такое кодовое слово, находящееся на расстоянии не большим τ от **v** — единственно, поскольку минимальное расстояние d_G используемого кода Гоппы не менее $2\tau + 1$.

Вычисление открытого ключа и секретного множества локаторов для обобщенных (L,G)-кодов с локаторами первой и второй степени

Для упрощения изложения, но без потери общности, далее будет рассмотрен случай построения проверочной матрицы и открытого ключа для обобщенного (L, G)-кода, содержащего локаторы первой и второй степени.

Проверочная матрица для обобщенного (L, G)-кода с сепарабельным многочленом $G(x)$, содержащая локаторы первой и второй степени может быть представлена в виде:

$$\mathbf{H} = \begin{bmatrix} \frac{1}{G(\alpha_i)} & \cdots & \frac{1}{G(\beta_j)} & + & \left(\frac{1}{G(\beta_j)}\right)^{2m} \\ \frac{\alpha_i}{G(\alpha_i)} & \cdots & \frac{\beta_j}{G(\beta_j)} & + & \left(\frac{\beta_j}{G(\beta_j)}\right)^{2m} \\ \vdots & \ddots & & & \vdots \\ \frac{\alpha_i^{t-1}}{G(\alpha_i)} & \cdots & \frac{\beta_j^{t-1}}{G(\beta_j)} & + & \left(\frac{\beta_j^{t-1}}{G(\beta_j)}\right)^{2m} \end{bmatrix},$$

где $\alpha_i \in GF(2^m)$, $\beta_i \in GF(2^{2m})$.

Чтобы не строить поле размера $GF(2^{2m})$, элемент данного поля можно представить [9] в виде:

$$\beta_j = \alpha_k z + \alpha_f, \tag{5}$$

где $\alpha_k, \alpha_f \in GF(2^m)$.

Опишем теперь сложение и умножение таких элементов, задав элемент $\beta_i = \alpha_m z + \alpha_n$.

$$\beta_i + \beta_j = \alpha_m z + \alpha_n + \alpha_k z + \alpha_f = (\alpha_k + \alpha_m)z + (\alpha_f + \alpha_n),$$

$$\beta_i\beta_j = (\alpha_m z + \alpha_n)(\alpha_k z + \alpha_f) = (\alpha_k\alpha_m)z^2 + (\alpha_k\alpha_n + \alpha_f\alpha_m)z + \alpha_f\alpha_n. \tag{6}$$

Для упрощения полученных соотношений можно редуцировать их по модулю неприводимого многочлена второй степени [9]. Известно [14], что существует неприводимый многочлен вида $f(z) = z^2 + z + \alpha_k$, $\alpha_k \in CF(2^m)$ для любого m. Таким образом, соотношение (6) можно переписать в виде:

$$\beta_i\beta_j = (\alpha_k z + \alpha_f)(\alpha_m z + \alpha_n) = (\alpha_k\alpha_m)z^2 + \\ + (\alpha_k\alpha_n + \alpha_f\alpha_m)z + \alpha_f\alpha_n \equiv (\alpha_k\alpha_n + \alpha_f\alpha_m + \alpha_k\alpha_m)z + \\ + (\alpha_f\alpha_n + \alpha_k\alpha_m)\mathrm{mod}(z^2 + z + \alpha_k).$$

Подставляя значение (5) в $G(\beta_j)$ по модулю неприводимого многочлена, получим

$$G(\beta_j) = G(\alpha_m z + \alpha_n) = \beta_q = \alpha_p z + \alpha_h,$$

где $\alpha_p, \alpha_n \in GF(2^m)$; $\beta_q \in GF(2^{2m})$.

Для нахождения значений в проверочной матрице **H** для локаторов второй степени необходимо найти значение, обратное значению $G(\beta_j)$.

Пусть $\alpha_d z + \alpha_v = (\alpha_p z + \alpha_n)^{-1}$. Тогда

$$(\alpha_d z + \alpha_v)(\alpha_p z + \alpha_n) = 1.$$

Из этого уравнения получим

$$(\alpha_d z + \alpha_v)(\alpha_p z + \alpha_n) = \alpha_d\alpha_p z^2 + (\alpha_d\alpha_n + \alpha_v\alpha_p)z + \alpha_v\alpha_n \equiv \\ \equiv \alpha_d\alpha_p(z + \alpha_k) + (\alpha_d\alpha_n + \alpha_v\alpha_p)z + \alpha_v\alpha_n \equiv \\ \equiv (\alpha_d\alpha_p + \alpha_d\alpha_n + \alpha_v\alpha_p)z + (\alpha_v\alpha_n + \alpha_d\alpha_p\alpha_k) \equiv \\ \equiv 1\mathrm{mod}(z^2 + z + \alpha_k).$$

Теперь можно перейти к следующей системе уравнений:

$$\begin{cases} \alpha_d\alpha_p + \alpha_d\alpha_n + \alpha_v\alpha_p = 0, \\ \alpha_v\alpha_n + \alpha_d\alpha_p\alpha_k = 1. \end{cases}$$

Подставляя в первое уравнение значение α_d из второго уравнения получим

$$\begin{cases} \alpha_k^{-1} + \alpha_v\alpha_n\alpha_k^{-1} + \alpha_p^{-1}\alpha_k^{-1}\alpha_n + \alpha_v\alpha_n^2\alpha_p^{-1}\alpha_k^{-1} + \alpha_v\alpha_p = 0, \\ \alpha_d = \alpha_p^{-1}\alpha_k^{-1} + \alpha_v\alpha_n\alpha_p^{-1}\alpha_k^{-1}. \end{cases}$$

Подставляя во второе уравнение значение α_v из первого уравнения получим

$$\begin{cases} \alpha_v = \dfrac{\alpha_p^{-1}\alpha_k^{-1}\alpha_n + \alpha_k^{-1}}{\alpha_n\alpha_k^{-1} + \alpha_n^2 + \alpha_p^{-1}\alpha_k^{-1} + \alpha_p}, \\ \alpha_d = \alpha_p^{-1}\alpha_k^{-1} + \alpha_v\alpha_n\alpha_p^{-1}\alpha_k^{-1}. \end{cases}$$

Тогда

$$\begin{cases} \alpha_v = \dfrac{\alpha_n + \alpha_p}{\alpha_n^2 + \alpha_n\alpha_p + \alpha_p^2}, \\ \alpha_d = \dfrac{\alpha_p}{\alpha_n^2\alpha_k + \alpha_n\alpha_p\alpha_k + \alpha_p^2\alpha_k}. \end{cases}$$

Зная, что $\dfrac{1}{G(\beta_j)} = \alpha_d z + \alpha_v$ можно найти значения элементов в столбцах проверочной матрицы **H** для локаторов второй степени.

Для вычисления открытого ключа системы Мак-Элиса необходимо привести проверочную матрицу **H** к виду:

$$\hat{\mathbf{H}} = [\mathbf{E} \quad \mathbf{T}],$$

где **E** — единичная матрица; **T** — оставшаяся матрица (открытый ключ системы). Для построения открытой проверочной матрицы проверочная матрица **H** приводится по Гауссу. На первом этапе осуществляется так называемый прямой ход, когда путем элементарных преобразований над столбцами исходную матрицу **H** приводят к ступенчатой форме, а именно, среди элементов первой строки матрицы выбирают ненулевой, перемещают его на первую позицию перестановкой столбцов (если поменялся 1-й и j-й столбцы, то необходимо поменять 1-й и j-й локаторы во множестве локаторов L). Затем получившаяся после перестановки первая строка складывается по модулю 2 с остальными строками, где на первой позиции элемент равен 1, обнуляя тем самым столбец под единичным элементом, стоящим на первой позиции в первой строке. Затем среди элементов второй строки матрицы находят ненулевой и аналогичным образом перемещают его на вторую позицию перестановкой столбцов, при этом меняются местами соответствующие локаторы во множестве локаторов L. Затем получившаяся после перестановки вторая строка складывается по модулю 2 с находящимися ниже остальными строками, где на второй позиции элемент равен 1, обнуляя тем самым столбец под единичным элементом, стоящим на второй позиции во второй строке. Эти действия выполняются со всеми строками матрицы **H**. На втором этапе осуществляется так называемый обратный ход. Эта процедура повторяет прямой ход, но начинается с последней строки и идет вверх. После этого получившаяся единичная матрица **E** отбрасывается, и оставшаяся часть **T** матрицы $\hat{\mathbf{H}}$ является открытым ключом. Дополнительным результатом выполнения процедуры Гаусса над матрицей $\hat{\mathbf{H}}$ является получение секретного ключа $\hat{L}$ — переупорядоченного множества локаторов.

В результате выполнения такой процедуры Гаусса получается матрица $\hat{\mathbf{H}}$ и новое множество локаторов $\hat{L}$ с переставленными соответствующим образом элементами исходного множества L.

Реализация шифрования и расшифрования

Реализация шифрования. Для реализации процедуры шифрования необходимо создать двоичный вектор-столбец **e** длины n и веса Хэмминга $t/2$, где n — длина кода; t — степень многочлена Гоппы.

Далее данный вектор разбивается на две части следующим образом

$$\mathbf{e}^T = (\mathbf{e_1}|\mathbf{e_2}),$$

где $\mathbf{e_1}$ — вектор длины mt; $\mathbf{e_2}$ — вектор длины $n - mt$.

Затем находится результат перемножения матрицы **T** на вектор $\mathbf{e_1}$, и результат складывается с вектором $\mathbf{e_2}$

$$\mathbf{c_0} = \mathbf{T}\mathbf{e_1}^T \oplus \mathbf{e_2}^T.$$

Полученный вектор $\mathbf{c_0}$ и будет являться результатом шифрования. Отметим, что данная процедура равносильна умножению матрицы $\hat{\mathbf{H}} = [\mathbf{E} \quad \mathbf{T}]$ на вектор-столбец $\mathbf{e} = (\mathbf{e_1}^T|\mathbf{e_2}^T)$ и позволяет не расширять матрицу **T** до матрицы $\hat{\mathbf{H}}$, что приводит к ускорению вычислений.

Реализация расшифрования. Для реализации процедуры расшифрования находится следующая матрица

с элементами из $GF(2^m)$, для построения которой используются первые $n - mt$ локаторов из множества $\hat{L}$:

$$\mathbf{H_1} = \begin{bmatrix} \frac{1}{G(\alpha_i)} & \cdots & \frac{1}{G(\beta_j)} & + & \left(\frac{1}{G(\beta_j)}\right)^{2m} \\ \frac{\alpha_i}{G(\alpha_i)} & \cdots & \frac{\beta_j}{G(\beta_j)} & + & \left(\frac{\beta_j}{G(\beta_j)}\right)^{2m} \\ \vdots & \ddots & & & \vdots \\ \frac{\alpha_i^{t-1}}{G(\alpha_i)} & \cdots & \frac{\beta_j^{t-1}}{G(\beta_j)} & + & \left(\frac{\beta_j^{t-1}}{G(\beta_j)}\right)^{2m} \end{bmatrix}.$$

Матрица $\mathbf{H_1}$, состоящая из элементов поля $GF(2^{2m})$, представленных в виде (3) умножается на вектор-столбец $\mathbf{c_0}$

$$\mathbf{c} = \mathbf{H_1 c_0}.$$

Такой вариант процедуры вычисления синдрома позволяет не расширять вектор $\mathbf{c_0}$ до вектора $\mathbf{v}$, а также использовать проверочную матрицу меньшего размера. Для полученного значения синдрома $\mathbf{c}$ с помощью одного из алгоритмов: расширенный алгоритм Евклида, алгоритм Берлекэмпа–Месси или алгоритм Паттерсона [15] находится многочлен локаторов позиций ошибок.

Нахождение позиций ошибок с помощью процедуры Ченя. Затем из многочлена локаторов позиций ошибок находятся позиции ошибок, т. е. вектор $\mathbf{e}$. Для этого можно использовать процедуру Ченя, которая в классическом варианте алгоритма декодирования помехоустойчивых кодов позволяет найти позиции вектора ошибки, на которых находятся единицы для множества локаторов первой степени. Для данной модификации алгоритма декодирования также можно использовать процедуру Ченя. Это возможно, так как полученный после декодирования полином можно представить как

$$\alpha_{i_t} x^t + \alpha_{i_{t-1}} x^{t-1} + \ldots + \alpha_{i_0} = \alpha_{i_t}(x + \alpha_{j_0})(x + \alpha_{j_1})\ldots(x + \alpha_{j_{t-1}}),$$

где α_{j_k} — корни данного многочлена.

Чтобы использовать процедуру Ченя, можно представить элемент поля $GF(2^{2m})$ как в (3).

Полином локаторов ошибок, состоящий из нумераторов первой и второй степени, можно представить следующим образом

$$\alpha_{i_t} x^t + \alpha_{i_{t-1}} x^{t-1} + \ldots + \alpha_{i_0} = \\ = \alpha_{i_t} \prod_{v=0}^{\lambda-1} (x^{k_v} + \alpha_{v,i_{k_v-1}} x^{k_v-1} + \ldots + \alpha_{v,i_0}) \qquad (7)$$

где $x^{k_v} + \alpha_{v,i_{k_v-1}} x^{k_v-1} + \ldots + \alpha_{v,i_0}$ — неприводимый многочлен первой или второй степени ($k_v \in \{1,2\}$) над полем $GF(2^m)$ и $t = \sum_{v=0}^{\lambda-1} k_v$.

Любой неприводимый многочлен второй степени над $GF(2^m)$ можно представить как:

$$x^2 + \alpha_{i_1} x + \alpha_{i_0} = (x + \beta)(x + \beta^{2m}), \qquad (8)$$

где $\beta \in GF(2^{2m})$.

Если при подстановке элемента β, представленного как в (1), уравнение (8) обратится в ноль, значит и уравнение (7) также обращается в ноль.

В результате будет найден двоичный вектор, содержащий единицы на позициях, занумерованных локаторами, при подстановке которых уравнение (7) обратилось в ноль.

Пример 1. Рассмотрим полином локаторов ошибок $s(x)$ над полем $GF(2^3)$, задаваемым порождающим многочленом $x^3 + x + 1$, который содержит нумераторы первой и второй степени

$$s(x) = x^5 + \alpha^1 x^4 + \alpha^5 x^3 + \alpha^5 x^2 + \alpha^4 x + \alpha^4. \qquad (9)$$

Пусть элемент, соответствующий проверяемому локатору позиции, равен β^f:

$$s(\beta^f) = (\beta^f)^5 + \alpha^1(\beta^f)^4 + \alpha^5(\beta^f)^3 + \alpha^5(\beta^f)^2 + \alpha^4(\beta^f) + \alpha^4. \qquad (10)$$

Тогда $\beta^f = \alpha^1 + \alpha^2 z$, и проверим, является ли данный элемент корнем уравнения (9).

Найдем значения β^f, $(\beta^f)^2,\ldots,(\beta^f)^5$ используя представления элементов поля $GF(2^6)$ как многочленов с коэффициентами из поля $GF(2^3)$ по модулю неприводимого многочлена $f(z) = z^2 + z + 1$:

$$\beta^f = \alpha^1 + \alpha^2 z,$$

$$(\beta^f)^2 = \alpha^2 + \alpha^4 z^2 = \alpha^2 + \alpha^4(z + 1) \equiv \\ \equiv \alpha^1 + \alpha^2 z \bmod(z^2 + z + 1),$$

$$(\beta^f)^3 = \beta^f(\beta^f)^2 = \alpha^2 + (\alpha^3 + \alpha^5)z + \alpha^6 z^2 \equiv \\ \equiv \alpha^2 + \alpha^2 z + \alpha^6(z + 1) \equiv \alpha^0 + \alpha^0 z \bmod(z^2 + z + 1),$$

$$(\beta^f)^4 = (\alpha^1 + \alpha^2 z)^4 \equiv \alpha^4 + \alpha^1 z^4 \equiv \alpha^4 + \alpha^1 z \equiv \\ \equiv \alpha^4 + \alpha^1 z \bmod(z^2 + z + 1),$$

$$(\beta^f)^5 = \beta^f(\beta^f)^4 = (\alpha^1 + \alpha^2 z)(\alpha^4 + \alpha^1 z) \equiv \\ \equiv \alpha^2 + \alpha^1 z \bmod(z^2 + z + 1).$$

Теперь подставим полученные значения в (10)

$$s(x) = \alpha^2 + \alpha^1 z + \alpha^1(\alpha^4 + \alpha^1 z) + \alpha^5(\alpha^0 + \alpha^0 z) + \\ + \alpha^5(\alpha^1 + \alpha^4 z) + \alpha^4(\alpha^1 + \alpha^2 z) + \alpha^4 = 0.$$

Так как данное уравнение обратилось в ноль при подстановке $\beta^f = \alpha^1 + \alpha^2 z$, значит $\alpha^1 + \alpha^2 z$ является его корнем.

Заключение

В статье рассмотрен один из вариантов реализации современной криптосистемы Мак-Элиса, а именно предложен алгоритм, использующий обобщенные (L, G)-коды с нумераторами второй степени и выше без вычислений в расширении исходного поля, предложен вариант использования процедуры Ченя для данной реализации, а также подход к реализации шифрования и расшифрования.

Литература

1. Гоппа В.Д. Новый класс линейных корректирующих кодов // Проблемы передачи информации. 1970. Т. 6. № 3. С. 24–30.
2. Bezzateev S.V., Shekhunova N.A. One generalization of Goppa codes // Proc. IEEE International Symposium on Information Theory (ISIT 1997). 1997. P. 299. doi: 10.1109/ISIT.1997.613221
3. Zeh A., Wachter-Zeh A., Bezzateev S.V. Decoding cyclic codes up to a new bound on the minimum distance // IEEE Transaction on Information Theory. 2012. V. 58. N 6. P. 3951–3960. doi: 10.1109/TIT.2012.2185924
4. MacWilliams F.J., Sloane N.J.A. The Theory of Error-Correcting Codes. Amsterdam: Elsevier Science, 1977. 762 p.
5. Wang W., Szefer J., Niederhagen R. FPGA-based key generator for the niederreiter cryptosystem using binary Goppa codes // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2017. V. 10529. P. 253–274. doi: 10.1007/978-3-319-66787-4_13
6. Bernstein D., Chou T., Lange T., Maurich I., Misoczki R., Niederhagen R., Persichetti E., Peters C., Schwabe P., Sendrier N., Szefer J., Wang W. Classic McEliece: conservative code-based cryptography: Project documentation [Электронный ресурс]. URL: https://classic.mceliece.org/nist/mceliece-20190331.pdf, свободный. Яз. англ. (дата обращения: 27.05.2020).
7. McEliece R.J. A public-key cryptosystem based on algebraic coding theory: Technical report. NASA, 1978.
8. Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory // Problems of Control and Information Theory. 1986. V. 15. N 2. P. 159–166.
9. Berlekamp E.R. Algebraic Coding Theory. McGraw-Hill, 1968. 466 p.
10. Massey J. Shift-register synthesis and BCH decoding // IEEE Transactions on Information Theory. 1969. V. 15. N 1. P. 122–127. doi: 10.1109/TIT.1969.1054260
11. Sugiyama Y., Kasahara M., Hirasawa S., Namekawa T. A method for solving key equation for decoding Goppa codes // Information and Control. 1975. V. 27. N 1. P. 87–99. doi: 10.1016/S0019-9958(75)90090-X
12. Patterson N.J. The algebraic decoding of Goppa code // IEEE Transaction on Information Theory. 1975. V. 21. N 2. P. 203–207. doi: 10.1109/TIT.1975.1055350
13. Chien R.T., Watson T.J. Cyclic decoding procedures for Bose Chaudhuri-Hocquenghem codes // IEEE Transactions on Information Theory. 1964. V. 10. N 4. P. 357–363. doi: 10.1109/TIT.1964.1053699
14. Véron P. Goppa codes and trace operator // IEEE Transactions on Information Theory. 1998. V. 44. N 1. P. 290–294. doi: 10.1109/18.651048
15. Bezzateev S.V., Noskov I.K. Patterson algorithm for decoding separable binary Goppa codes // Proc. 2019 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2019. P. 8840650. doi: 10.1109/WECONF.2019.8840650

Авторы

Носков Иван Константинович — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, Scopus ID: 57207734046, ORCID ID: 0000-0001-7758-097X, vanya170595@gmail.com

Беззатеев Сергей Валентинович — доктор технических наук, доцент, профессор практики, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация; заведующий кафедрой, Санкт-Петербургский государственный университет аэрокосмического приборостроения (ГУАП), Санкт Петербург, 190000, Российская Федерация, Scopus ID: 6602425996, ORCID ID: 0000-0002-0924-6221, bsv@aanet.ru

References

1. Goppa V.D. A new class of linear correcting codes. *Problems of Information Transmission*, 1970, vol. 6, no. 3, pp. 207–212.
2. Bezzateev S.V., Shekhunova N.A. One generalization of Goppa codes. *Proc. IEEE International Symposium on Information Theory (ISIT 1997)*, 1997, pp. 299. doi: 10.1109/ISIT.1997.613221
3. Zeh A., Wachter-Zeh A., Bezzateev S.V. Decoding cyclic codes up to a new bound on the minimum distance. *IEEE Transaction on Information Theory*, 2012, vol. 58, no. 6, pp. 3951–3960. doi: 10.1109/TIT.2012.2185924
4. MacWilliams F.J., Sloane N.J.A. *The Theory of Error-Correcting Codes*. Amsterdam, Elsevier Science, 1977, 762 p.
5. Wang W., Szefer J., Niederhagen R. FPGA-based key generator for the niederreiter cryptosystem using binary Goppa codes. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10529, pp. 253–274. doi: 10.1007/978-3-319-66787-4_13
6. Bernstein D., Chou T., Lange T., Maurich I., Misoczki R., Niederhagen R., Persichetti E., Peters C., Schwabe P., Sendrier N., Szefer J., Wang W. *Classic McEliece: conservative code-based cryptography*. *Project documentation.* Available at: https://classic.mceliece.org/nist/mceliece-20190331.pdf (accessed: 27.05.2020).
7. McEliece R.J. *A public-key cryptosystem based on algebraic coding theory*. Technical report. NASA, 1978.
8. Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 1986, vol. 15, no. 2, pp. 159–166.
9. Berlekamp E.R. *Algebraic Coding Theory*. McGraw-Hill, 1968, 466 p.
10. Massey J. Shift-register synthesis and BCH decoding. *IEEE Transactions on Information Theory*, 1969, vol. 15, no. 1, pp. 122–127. doi: 10.1109/TIT.1969.1054260
11. Sugiyama Y., Kasahara M., Hirasawa S., Namekawa T. A method for solving key equation for decoding Goppa codes. *Information and Control*, 1975, vol. 27, no. 1, pp. 87–99. doi: 10.1016/S0019-9958(75)90090-X
12. Patterson N.J. The algebraic decoding of Goppa code. *IEEE Transaction on Information Theory*, 1975, vol. 21, no. 2, pp. 203–207. doi: 10.1109/TIT.1975.1055350
13. Chien R.T., Watson T.J. Cyclic decoding procedures for Bose Chaudhuri-Hocquenghem codes. *IEEE Transactions on Information Theory*, 1964, vol. 10, no. 4, pp. 357–363. doi: 10.1109/TIT.1964.1053699
14. Véron P. Goppa codes and trace operator. *IEEE Transactions on Information Theory*, 1998, vol. 44, no. 1, pp. 290–294. doi: 10.1109/18.651048
15. Bezzateev S.V., Noskov I.K. Patterson algorithm for decoding separable binary Goppa codes. *Proc. 2019 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*, 2019, pp. 8840650. doi: 10.1109/WECONF.2019.8840650

Authors

Ivan K. Noskov — Postgraduate, ITMO University, Saint Petersburg, 197101, Russian Federation, Scopus ID: 57207734046, ORCID ID: 0000-0001-7758-097X, vanya170595@gmail.com

Sergey V. Bezzateev — D.Sc., Associate Professor, Professor of Practice, ITMO University, Saint Petersburg, 197101, Russian Federation; Head of Chair, Saint Petersburg State University of Aerospace Instrumentation (SUAI), Saint Petersburg, 190000, Russian Federation, Scopus ID: 6602425996, ORCID ID: 0000-0002-0924-6221, bsv@aanet.ru