Научная статья
DOI: 10.18287/2541-7525-2020-26-3-17-29

УДК 519.725

Дата: поступления статьи: 13.03.2020
после рецензирования: 27.03.2020
принятия статьи: 25.05.2020

С.М. Рацеев
Ульяновский государственный университет,
г. Ульяновск, Российская Федерация
E-mail: ratseevsm@mail.ru. ORCID: https://orcid.org/0000-0003-4995-9418
О.И. Череватенко
Ульяновский государственный педагогический университет
имени И.Н. Ульянова, г. Ульяновск, Российская Федерация
E-mail: choi2008@mail.ru. ORCID: https://orcid.org/0000-0003-3931-9425

ОБ АЛГОРИТМАХ ДЕКОДИРОВАНИЯ ОБОБЩЕННЫХ КОДОВ РИДА — СОЛОМОНА НА СЛУЧАЙ ОШИБОК И СТИРАНИЙ

АННОТАЦИЯ

В статье приводятся алгоритмы декодирования обобщенных кодов Рида — Соломона на случай ошибок и стираний. Данные алгоритмы строятся на основе алгоритма Гао, алгоритма Сугиямы, алгоритма Берлекэмпа–Месси (алгоритма Питерсона — Горенстейна — Цирлера). Первый из данных алгоритмов относится к алгоритмам бессиндромного декодирования, остальные — к алгоритмам синдромного декодирования. Актуальность данных алгоритмов состоит в том, что они применимы для декодирования кодов Гоппы, которые лежат в основе некоторых перспективных постквантовых криптосистем. При этом данные алгоритмы применимы для кодов Гоппы над произвольным полем, в отличие от хорошо известного алгоритма декодирования Паттерсона для двоичных кодов Гоппы.

Ключевые слова: помехоустойчивые коды, коды Рида — Соломона, коды Гоппы, декодирование кода.

Цитирование. Рацеев С.М., Череватенко О.И. Об алгоритмах декодирования обобщенных кодов Рида — Соломона // Вестник Самарского университета. Естественнонаучная серия. 2020. Т. 26, № 3. С. 17–29. DOI: http://doi.org/10.18287/2541-7525-2020-26-3-17-29.

Информация о конфликте интересов: авторы и рецензенты заявляют об отсутствии конфликта интересов.

Рацеев Сергей Михайлович — доктор физико-математических наук, доцент, профессор кафедры информационной безопасности и теории управления, Ульяновский государственный университет, 432017, Российская Федерация, г. Ульяновск, ул. Льва Толстого, 42.

Череватенко Ольга Ивановна — кандидат физико-математических наук, доцент, доцент кафедры высшей математики, Ульяновский государственный педагогический университет имени И.Н. Ульянова, 432071, Российская Федерация, г. Ульяновск, площадь Ленина, 4/5.

Введение

Пусть $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{n-1})$, где α_i — различные элементы поля $F = GF(q)$, $y = (y_0, y_1, \ldots, y_{n-1})$ — ненулевые (не обязательно различные) элементы из F. Тогда обобщенный код Рида — Соломона, обозначаемый $GRS_k(\alpha, y)$, состоит из всех кодовых векторов вида:

$$u = (y_0 b(\alpha_0), y_1 b(\alpha_1), \ldots, y_{n-1} b(\alpha_{n-1})), \tag{1}$$

где $b(x)$ — информационные многочлены над полем F степени не выше $k-1$. Кодовое расстояние кода $GRS_k(\alpha, y)$ равно $d = n-k+1$. Если $n = q-1$, вектор y состоит из единиц и $\alpha_i = \alpha^i$, $i = 0, 1, \ldots, n-1$, где α — примитивный элемент поля F, то в этом случае получаем код Рида — Соломона (РС).

Заметим, что, в отличие от кодов РС, в обобщенных кодах РС одна из компонент вектора α может быть нулевой, что нужно учитывать для некоторых алгоритмов декодирования.

Нам понадобится вид проверочной матрицы кода $GRS_k(\alpha, y)$ (см., напр., [1]).

Теорема 1. Код, дуальный $GRS_k(\alpha, y)$-коду, является $GRS_{n-k}(\alpha, w)$-кодом для некоторого вектора w, причем в качестве вектора w можно взять $w = (w_0, w_1, \ldots, w_{n-1})$, где

$$w_i = \frac{1}{y_i \prod_{j \neq i}(\alpha_i - \alpha_j)}, \quad i = 0, 1, \ldots, n-1.$$

Из данной теоремы следует, что проверочная матрица H кода $GRS_k(\alpha, y)$ равна порождающей матрице кода $GRS_{n-k}(\alpha, w)$:

$$H = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_0 & \alpha_1 & \ldots & \alpha_{n-1} \\ \ldots & \ldots & \ldots & \ldots \\ \alpha_0^{n-k-1} & \alpha_1^{n-k-1} & \ldots & \alpha_{n-1}^{n-k-1} \end{pmatrix} \begin{pmatrix} w_0 & 0 & \ldots & 0 \\ 0 & w_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & w_{n-1} \end{pmatrix}. \tag{2}$$

Матрица H содержит $n - k = d - 1$ строк и n столбцов.

Для декодирования кодов Рида — Соломона на случай ошибок хорошо известны следующие алгоритмы [1–3]: алгоритм Гао, алгоритм Сугиямы, алгоритм Берлекэмпа–Месси, алгоритм Питерсона — Горенстейна — Цирлера. В дополнение к этим алгоритмам можно добавить алгоритм поиска ошибок Форни. Для обобщенных кодов Рида — Соломона и кодов Гоппы подобные алгоритмы рассматривались в работах [4–6].

Для декодирования кодов Гоппы хорошо известен алгоритм Паттерсона [7]. Этот алгоритм предлагается для использования в криптосистеме Мак-Элиса [8; 9]. Но алгоритм Паттерсона применим только для двоичных кодов Гоппы и не применим для случая, когда в канале связи действуют ошибки и стирания. В работе [10] приводится алгоритм списочного декодирования двоичных кодов Гоппы. При этом такой вариант не применим для криптосистемы Мак-Элиса, так как для нее нужны алгоритмы однозначного декодирования.

В данной статье приводятся алгоритмы декодирования для обобщенных кодов РС на случай ошибок и стираний: декодирование на основе алгоритма Гао, на основе алгоритма Сугиямы, на основе алгоритма Берлекэмпа–Месси (алгоритма Питерсона — Горенстейна — Цирлера). Понятно, что любой из этих алгоритмов применим и для случая канала связи только с ошибками. Актуальность таких алгоритмов декодирования состоит в том, что все алгоритмы декодирования для обобщенных кодов РС можно применять для декодирования кодов Гоппы, при этом именно на основе кодов Гоппы строятся некоторые перспективные постквантовые криптосистемы [11].

1. Декодирование ОРС кодов на основе алгоритма Гао на случай ошибок и стираний

Предположим, что в канале связи действуют ошибки и стирания. Пусть кодовый вектор $u \in \in GRS_k(\alpha, y)$ получен на основе информационного вектора b с помощью правила (1), а после передачи вектора u на приемной стороне получен вектор v, в котором t ошибок и s стираний. Пусть S — позиции стертых символов в векторе v. На основе векторов v, α, y составим соответствующие векторы $\widetilde{v}$, β, z путем удаления всех компонент с номерами из множества S. Рассмотрим код $GRS_k(\beta, z)$ длины $\widetilde{n} = n - s$ и размерности $\widetilde{k} = k$, который получается из кода $GRS_k(\alpha, y)$ путем выкалывания компонент с номерами из множества S. Для кодового расстояния кода $GRS_k(\beta, z)$ выполнено равенство $\widetilde{d} = \widetilde{n} - \widetilde{k} + 1 = n - s - k + 1$. Если $d \geqslant 2t + s + 1$, то для кодового расстояния $\widetilde{d}$ кода $GRS_k(\beta, z)$ выполнено неравенство $\widetilde{d} \geqslant 2t + 1$. Тогда вектор $\widetilde{v}$, в котором только ошибки, можно декодировать.

Для описания алгоритма из данного параграфа будем следовать работам [2; 12].

На основе компонент вектора β определим многочлен:

$$m(x) = (x - \beta_0)(x - \beta_1) \ldots (x - \beta_{\widetilde{n}-1}).$$

Пусть $X_1 = \beta_{i_1}, \ldots, X_t = \beta_{i_t}$ — локаторы ошибок. В данном алгоритме многочлен локаторов ошибок запишем в виде:

$$\sigma(x) = (x - X_1) \ldots (x - X_t).$$

Если ошибок не было, то будем полагать, что $\sigma(x) = 1$. Пусть $\widetilde{u}$ — вектор, полученный из u путем выкалывания компонент с номерами из S. Понятно, что $\widetilde{u} \in GRS_k(\beta, z)$. Так как $n - k + 1 = d \geqslant 2t + s + 1$,

то $n-s \geqslant 2t+k \geqslant k$, поэтому вектор $\widetilde{u}$ получен с помощью кодирования информационного многочлена $b(x) = b_0 + b_1x + \ldots + b_{k-1}x^{k-1}$ (на основе которого получен вектор u) с помощью правила:

$$\widetilde{u} = (z_0 b(\beta_0), z_1 b(\beta_1), \ldots, z_{\widetilde{n}-1} b(\beta_{\widetilde{n}-1})).$$

Если $\widetilde{v}_i = \widetilde{u}_i$, то $\widetilde{v}_i = z_i b(\beta_i)$. Если $\widetilde{v}_i \neq \widetilde{u}_i$, то на позиции i произошла ошибка, поэтому $\sigma(\beta_i) = 0$. Из этого следует, что

$$\sigma(\beta_i) z_i^{-1} \widetilde{v}_i = \sigma(\beta_i) b(\beta_i), \quad i = 0, 1, \ldots, \widetilde{n}-1.$$

Обозначим $p(x) = \sigma(x)b(x)$. Тогда:

$$\sigma(\beta_i) z_i^{-1} \widetilde{v}_i = p(\beta_i), \quad i = 0, 1, \ldots, \widetilde{n}-1.$$

Построим интерполяционный многочлен Лагранжа $f(x)$ степени не выше $\widetilde{n}-1$, проходящий через точки $(\beta_0, z_0^{-1}\widetilde{v}_0)$, $(\beta_1, z_1^{-1}\widetilde{v}_1), \ldots,$ $(\beta_{\widetilde{n}-1}, z_{\widetilde{n}-1}^{-1}\widetilde{v}_{\widetilde{n}-1})$:

$$f(\beta_i) = z_i^{-1}\widetilde{v}_i, \quad i = 0, 1, \ldots, \widetilde{n}-1, \quad \deg f(x) \leqslant \widetilde{n}-1.$$

Тогда из равенств:

$$\sigma(\beta_i) f(\beta_i) = p(\beta_i), \quad i = 0, 1, \ldots, \widetilde{n}-1,$$

получаем сравнение:

$$\sigma(x) f(x) \equiv p(x) \pmod{m(x)}. \tag{3}$$

Алгоритм 1 (декодирование ОРС кодов на основе алгоритма Гао на случай ошибок и стираний).
Вход: принятый вектор v.

Выход: исходный информационный вектор b, если в соответствующем кодовом векторе u произошло s стираний и не более t ошибок при $d \geqslant 2t+s+1$.

1. Пусть S — позиции стертых символов в векторе v. На основе векторов v, α, y составить соответствующие векторы $\widetilde{v}$, β, z путем удаления всех компонент с номерами из множества S. После этого вектор $\widetilde{v}$ рассматривается как вектор, в котором только ошибки и который соответствует некоторому кодовому вектору кода $GRS_k(\beta, z)$ длины $\widetilde{n} = n - s$ и размерности $\widetilde{k} = k$. Определяется многочлен:

$$m(x) = \prod_{i=0}^{\widetilde{n}-1} (x - \beta_i).$$

2. Интерполяция. Строится интерполяционный многочлен $f(x)$, для которого

$$f(\beta_i) = z_i^{-1}\widetilde{v}_i, \quad i = 0, 1, \ldots, \widetilde{n}-1.$$

3. Незаконченный обобщенный алгоритм Евклида. Пусть $r_{-1}(x) = m(x)$, $r_0(x) = f(x)$, $v_{-1}(x) = 0$, $v_0(x) = 1$. Производится последовательность действий обобщенного алгоритма Евклида:

$$r_{i-2}(x) = r_{i-1}(x) q_{i-1}(x) + r_i(x),$$
$$v_i(x) = v_{i-2}(x) - v_{i-1}(x) q_{i-1}(x), \quad i \geqslant 1,$$

до тех пор, пока не достигается такого $r_j(x)$, для которого:

$$\deg r_{j-1}(x) \geqslant \frac{\widetilde{n}+\widetilde{k}}{2}, \quad \deg r_j(x) < \frac{\widetilde{n}+\widetilde{k}}{2}.$$

4. Деление. Информационный многочлен кода $GRS_k(\beta, z)$, соответствующий кодовому вектору u, равен $b(x) = \frac{r_j(x)}{v_j(x)}$.

Теорема 2. Если в кодовом векторе произошло t ошибок и s стираний, причем $d \geqslant 2t+s+1$, то алгоритм декодирования 1 всегда приводит к единственному решению, а именно к исходному информационному вектору b.

Доказательство. После применения к вектору v шага 1 алгоритма 1 получим вектор $\widetilde{v}$, в котором только ошибки. Однозначность декодирования вектора $\widetilde{v}$ с помощью шагов 2-4 алгоритма 1 следует из теоремы 1 работы [5]. □

Пример 1. Рассмотрим обобщенный код РС над полем $GF(11)$ с параметрами $n = 9$, $k = 4$, $d = 6$, $\alpha = (0, 1, \ldots, 8)$, $y = (2, 1, 3, 1, 4, 1, 5, 1, 6)$.

Данный $[9, 4, 6]$-код $GRS_4(\alpha, y)$ может исправлять до двух ошибок и одно стирание, либо одну ошибку и до трех стираний, либо до пяти стираний. Рассмотрим случай двух ошибок и одного стирания.

Пусть $b = (4, 2, 1, 7)$ — информационный вектор, который соответствует многочлену $b(x) = 4 + 2x + x^2 + 7x^3$. После кодирования вектора b получаем кодовый вектор:

$$u = (y_0 b(0), y_1 b(1), \ldots, y_8 b(8)) = (8, 3, 6, 10, 1, 1, 10, 4, 8).$$

Пусть после отправки вектора u на приемном конце получен вектор v:

$$v = (1, 3, 6, 10, 9, 1, 10, *, 8),$$

т. е. произошли две ошибки на 0-й и 4-й позициях (нумеруя с нуля) и одно стирание на 7-й позиции.

1. Удалив в векторе v стертые символы, получим новый вектор:

$$\widetilde{v} = (1, 3, 6, 10, 9, 1, 10, 8),$$

в котором только две ошибки. Пусть β и z — векторы длины 8, которые получаются соответственно из векторов α и y путем удаления 7-й компоненты:

$$\beta = (0, 1, 2, 3, 4, 5, 6, 8), \quad z = (2, 1, 3, 1, 4, 1, 5, 6).$$

Множество S позиций стертых символов равно $S = \{7\}$. Составляем многочлен $m(x)$:

$$m(x) = x(x-1)(x-2)(x-3)(x-4)(x-5)(x-6)(x-8) =$$
$$= 4x + 4x^2 + 6x^3 + 2x^4 + 10x^5 + 2x^6 + 4x^7 + x^8.$$

Ниже приведена матрица Вандермонда V на основе вектора β, обратная к ней матрица V^{-1} и диагональная матрица Z на основе вектора z:

$$V = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 & 5 & 6 & 8 \\ 0 & 1 & 4 & 9 & 5 & 3 & 3 & 9 \\ 0 & 1 & 8 & 5 & 9 & 4 & 7 & 6 \\ 0 & 1 & 5 & 4 & 3 & 9 & 9 & 4 \\ 0 & 1 & 10 & 1 & 1 & 1 & 10 & 10 \\ 0 & 1 & 9 & 3 & 4 & 5 & 5 & 3 \\ 0 & 1 & 7 & 9 & 5 & 3 & 8 & 2 \end{pmatrix}, \quad V^{-1} = \begin{pmatrix} 1 & 1 & 7 & 6 & 8 & 6 & 1 & 3 \\ 0 & 10 & 9 & 2 & 7 & 10 & 4 & 3 \\ 0 & 1 & 7 & 5 & 3 & 4 & 8 & 5 \\ 0 & 7 & 2 & 2 & 6 & 3 & 10 & 3 \\ 0 & 9 & 3 & 6 & 6 & 2 & 5 & 2 \\ 0 & 1 & 10 & 9 & 10 & 10 & 8 & 7 \\ 0 & 3 & 9 & 6 & 8 & 7 & 10 & 1 \\ 0 & 1 & 8 & 8 & 7 & 2 & 9 & 9 \end{pmatrix},$$

$$Z = \mathrm{Diag}(2, 1, 3, 1, 4, 1, 5, 6).$$

2. Интерполяция. Вычисляем коэффициенты многочлена $f(x) = f_0 + f_1 x + \ldots + f_7 x^7$:

$$(f_0, f_1, \ldots, f_7) = \widetilde{v} Z^{-1} V^{-1} = (6, 0, 10, 9, 6, 5, 1, 10),$$
$$f(x) = 6 + 10x^2 + 9x^3 + 6x^4 + 5x^5 + x^6 + 10x^7.$$

3. Применение неполного обобщенного алгоритма Евклида. Определяем $r_{-1}(x) = m(x)$, $r_0(x) = f(x)$, $v_{-1}(x) = 0$, $v_0(x) = 1$ и применяем алгоритм Евклида:

$$\begin{aligned} r_{-1}(x) &= r_0(x) q_0(x) + r_1(x), \\ q_0(x) &= 6 + 10x, \\ r_1(x) &= 8 + 10x + 10x^2 + 6x^3 + 8x^4 + 8x^5 + x^6, \\ v_1(x) &= v_{-1}(x) - q_0(x) v_0(x) = 5 + x, \\ r_0(x) &= r_1(x) q_1(x) + r_2(x), \\ q_1(x) &= 9 + 10x, \\ r_2(x) &= 6x + 7x^2 + 9x^3 + 6x^4 + 7x^5, \\ v_2(x) &= v_0(x) - q_1(x) v_1(x) = 7x + x^2. \end{aligned}$$

Так как $(\widetilde{n} + \widetilde{k})/2 = 6$, $\deg r_1(x) = 6$, $\deg r_2(x) = 5$, то после второго шага алгоритма Евклида останавливаемся.

4. Деление:

$$b(x) = \frac{r_2(x)}{v_2(x)} = 4 + 2x + x^2 + 7x^3.$$

2. Декодирование ОРС кодов на основе алгоритма Сугиямы на случай ошибок и стираний

Пусть v — полученный на приемной стороне вектор, в котором могут быть ошибки и стирания. Пусть t — максимальное число возможных ошибок при фиксированном числе стираний s в векторе v, $d \geqslant 2t+s+1$, $t = [(d-s-1)/2]$. Так как позиции стертых символов известны, то заменим эти символы в векторе v, например, на нули и будем обращаться с полученным вектором $\widetilde{v}$ как с вектором, содержащим только ошибки. Пусть ошибки произошли на позициях $i_1, \ldots, i_t$, а стирания на позициях $i_{t+1}, \ldots, i_{t+s}$. При этом известны только позиции $i_{t+1}, \ldots, i_{t+s}$. После того как на данные позиции поместили нули, с

какими-то позициями могли угадать (если в кодовом векторе там действительно стояли нули). Поэтому $\widetilde{v} = u + e$, где e — вектор ошибок веса не более $t + s$.

Вычисляя синдромный вектор, получаем:

$$S = \widetilde{v}H^T = eH^T = \left(\ \dots, \quad e_{i_1}, \quad \dots, \quad e_{i_{t+s}}, \quad \dots \ \right) \times$$

$$\times \left(\begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_0 & \alpha_1 & \dots & \alpha_{n-1} \\ \dots & \dots & \dots & \dots \\ \alpha_0^{n-k-1} & \alpha_1^{n-k-1} & \dots & \alpha_{n-1}^{n-k-1} \end{pmatrix} \begin{pmatrix} w_0 & 0 & \dots & 0 \\ 0 & w_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & w_{n-1} \end{pmatrix} \right)^T =$$

$$= \begin{pmatrix} e_{i_1}w_{i_1} + \ldots + e_{i_{t+s}}w_{i_{t+s}} \\ e_{i_1}w_{i_1}\alpha_{i_1} + \ldots + e_{i_{t+s}}w_{i_{t+s}}\alpha_{i_{t+s}} \\ \dots \\ e_{i_1}w_{i_1}\alpha_{i_1}^{n-k-1} + \ldots + e_{i_{t+s}}w_{i_{t+s}}\alpha_{i_{t+s}}^{n-k-1} \end{pmatrix}^T.$$

Пусть $X_1 = \alpha_{i_1}, \dots X_t = \alpha_{i_t}$ — неизвестные локаторы ошибок, $X_{t+1} = \alpha_{i_{t+1}}, \dots, X_{t+s} = \alpha_{i_{t+s}}$ — известные локаторы стираний, $Y_1 = e_{i_1}, \dots, Y_{t+s} = e_{i_{t+s}}$ — значения ошибок. Обозначим $Z_j = Y_j w_{i_j}$, $j = 1, \dots, t+s$. Тогда:

$$\begin{aligned} &S_0 = Z_1 + \ldots + Z_t + Z_{t+1} + \ldots + Z_{t+s}, \\ &S_1 = Z_1X_1 + \ldots + Z_tX_t + Z_{t+1}X_{t+1} + \ldots + Z_{t+s}Z_{t+s}, \\ &\dots \\ &S_{2t+s-1} = Z_1X_1^{2t+s-1} + \ldots + Z_tX_t^{2t+s-1} + Z_{t+1}X_{t+1}^{2t+s-1} + \ldots + Z_{t+s}X_{t+s}^{2t+s-1}. \end{aligned} \tag{4}$$

При этом из $d = n - k + 1 \geqslant 2t + s + 1$ следует, что $n - k \geqslant 2t + s$. Запишем синдромный многочлен в виде:

$$S(x) = \sum_{i=0}^{2t+s-1} S_i x^i = \sum_{i=0}^{2t+s-1} \left(\sum_{j=1}^{t+s} Z_j X_j^i \right) x^i = \sum_{j=1}^{t+s} Z_j \left(\sum_{i=0}^{2t+s-1} (X_j x)^i \right) =$$

$$= \sum_{j=1}^{t+s} Z_j \frac{1 - (X_j x)^{2t+s}}{1 - X_j x} = \sum_{j=1}^{t+s} \frac{Z_j}{1 - X_j x} - x^{2t+s} \sum_{j=1}^{t+s} \frac{Z_j X_j^{2t+s}}{1 - X_j x}.$$

Полагая

$$\widetilde{\sigma}(x) = \prod_{i=1}^{t+s}(1 - X_i x) = \sum_{i=0}^{t+s} \widetilde{\sigma}_i x^i, \quad \widetilde{\sigma}_0 = 1,$$

$$\widetilde{\omega}(x) = \sum_{i=1}^{t+s} Z_i \prod_{\substack{1 \leqslant j \leqslant t+s, \\ j \neq i}} (1 - X_j x), \quad \widetilde{\Phi}(x) = \sum_{i=1}^{t+s} Z_i X_i^{2t+s} \prod_{\substack{1 \leqslant j \leqslant t+s, \\ j \neq i}} (1 - X_j x),$$

после приведения всех дробей к общему знаменателю получим:

$$S(x) = \frac{\widetilde{\omega}(x)}{\widetilde{\sigma}(x)} - x^{2t+s} \frac{\widetilde{\Phi}(x)}{\widetilde{\sigma}(x)}.$$

Тогда

$$S(x)\widetilde{\sigma}(x) = \widetilde{\omega}(x) - x^{2t+s}\widetilde{\Phi}(x).$$

Данное выражение называют ключевым уравнением, которому можно придать иной вид:

$$\widetilde{\sigma}(x)S(x) \equiv \widetilde{\omega}(x) \pmod{x^{2t+s}}. \tag{5}$$

Заметим, что $\widetilde{\sigma}(x) = \sigma(x)\nu(x)$, где $\sigma(x)$ — это многочлен неизвестных локаторов ошибок, $\nu(x)$ — многочлен известных локаторов стираний:

$$\widetilde{\sigma}(x) = \prod_{i=1}^{t}(1 - X_i x) \prod_{i=1}^{s}(1 - X_{t+i} x) = \sigma(x)\nu(x).$$

Введем в рассмотрение многочлен $\widetilde{S}(x) = S(x)\nu(x)$ — модифицированный синдромный многочлен. Тогда ключевое уравнение (5) примет вид:

$$\sigma(x)\widetilde{S}(x) \equiv \widetilde{\omega}(x) \pmod{x^{2t+s}}, \tag{6}$$

где

$$\deg \sigma(x) \leqslant t, \quad \deg \widetilde{\omega}(x) \leqslant t + s - 1, \quad \sigma(0) = 1. \tag{7}$$

Рассмотрим сравнение

$$a(x)\widetilde{S}(x) \equiv b(x) \pmod{x^{2t+s}} \tag{8}$$

относительно неизвестных многочленов $a(x), b(x) \in F[x]$ с условием

$$\deg a(x) \leqslant t, \ \deg b(x) \leqslant t+s-1, \ a(0)=1. \tag{9}$$

Из (6) и (7) следует, что сравнение (8) с условием (9) имеет решение.

Теорема 3. 1. Многочлены $a(x)$ и $b(x)$ являются решением сравнения (8) с условием (9) тогда и только тогда, когда для некоторого многочлена $\mu(x) \in F[x]$ выполнены равенства $a(x)=\mu(x)\sigma(x)$, $b(x)=$ $=\mu(x)\widetilde{\omega}(x)$.

2. Многочлены $\sigma(x)$ и $\widetilde{\omega}(x)$ являются единственным решением сравнения (8) с условием (9) и условием взаимной простоты.

Доказательство. 1. Если $a(x)=\mu(x)\sigma(x)$, $b(x)=\mu(x)\widetilde{\omega}(x)$, то

$$a(x)\widetilde{S}(x) \equiv \mu(x)\sigma(x)\widetilde{S}(x) \equiv \mu(x)\widetilde{\omega}(x) \equiv b(x) \pmod{x^{2t+s}}.$$

Обратно, пусть $a(x)$ и $b(x)$ — некоторое решение сравнения (8) с условием (9). Рассмотрим два сравнения:

$$b(x) \equiv a(x)\widetilde{S}(x) \pmod{x^{2t+s}}, \quad \widetilde{\omega}(x) \equiv \sigma(x)\widetilde{S}(x) \pmod{x^{2t+s}}.$$

Умножив первое сравнение на $\sigma(x)$, а второе на $a(x)$, получим:

$$b(x)\sigma(x) \equiv a(x)\sigma(x)\widetilde{S}(x) \equiv \widetilde{\omega}(x)a(x) \pmod{x^{2t+s}}.$$

Учитывая первые два неравенства из условия (9) для многочленов $a(x)$, $b(x)$, $\sigma(x)$, $\widetilde{\omega}(x)$, из сравнения $b(x)\sigma(x) \equiv \widetilde{\omega}(x)a(x) \pmod{x^{2t+s}}$ следует равенство:

$$b(x)\sigma(x) = \widetilde{\omega}(x)a(x). \tag{10}$$

Так как $\sigma(x) \,|\, \widetilde{\omega}(x)a(x)$ и $\sigma(x)$ и $\widetilde{\omega}(x)$ взаимно просты, то $\sigma(x) \,|\, a(x)$. Поэтому найдется многочлен $\mu(x)$, для которого $a(x)=\mu(x)\sigma(x)$. При этом из (10) следует, что:

$$\frac{b(x)}{\widetilde{\omega}(x)} = \frac{a(x)}{\sigma(x)} = \mu(x).$$

Таким образом, $a(x)=\mu(x)\sigma(x)$, $b(x)=\mu(x)\widetilde{\omega}(x)$.

2. Пусть $a(x)$ и $b(x)$ — некоторое решение сравнения (8) с условием (9), причем $a(x)$ и $b(x)$ взаимно просты. Из пункта 1 следует, что для некоторого многочлена $\mu(x)$ выполнено $a(x)=\mu(x)\sigma(x)$, $b(x)=$ $=\mu(x)\widetilde{\omega}(x)$. В силу взаимной простоты $a(x)$ и $b(x)$ многочлен $\mu(x)$ должен являться константой. А в силу условия $\sigma(0)=a(0)=1$ эта константа равна единице, поэтому $\alpha(x)=\sigma(x)$, $b(x)=\widetilde{\omega}(x)$. □

Определим для обобщенного алгоритма Евклида следующие многочлены:

$$r_{-1}(x) = x^{2t+s}, \ r_0(x) = \widetilde{S}(x),$$
$$u_{-1}(x) = 1, \ u_0(x) = 0, \quad v_{-1}(x) = 0, \ v_0(x) = 1.$$

Произведем последовательность действий обобщенного алгоритма Евклида ($i \geqslant 1$):

$$\begin{aligned} r_{i-2}(x) &= r_{i-1}(x)q_i(x) + r_i(x), \\ u_i(x) &= u_{i-2}(x) - u_{i-1}(x)q_{i-1}(x), \\ v_i(x) &= v_{i-2}(x) - v_{i-1}(x)q_{i-1}(x). \end{aligned}$$

При этом будем получать такие равенства:

$$u_i(x)x^{2t+s} + v_i(x)\widetilde{S}(x) = r_i(x),$$

из которых следуют сравнения:

$$r_i(x) \equiv v_i(x)\widetilde{S}(x) \pmod{x^{2t+s}}.$$

Учитывая, что степени остатков $r_i(x)$ строго убывают, будем применять алгоритм Евклида до тех пор, пока не достигнем такого $r_j(x)$, что

$$\deg r_{j-1}(x) \geqslant t+s, \quad \deg r_j(x) \leqslant t+s-1. \tag{11}$$

Тогда в качестве $a(x)$ и $b(x)$ возьмем такие многочлены:

$$a(x) = \lambda v_j(x), \quad b(x) = \lambda r_j(x), \tag{12}$$

где константа $\lambda \in F$ задается так, чтобы удовлетворялось условие $a(0)=1$ (в теореме 4 приводится обоснование того, что $v_j(0) \neq 0$, поэтому такая константа существует). В этом случае:

$$b(x) \equiv \lambda r_j(x) \equiv \lambda v_j(x)\widetilde{S}(x) \equiv a(x)\widetilde{S}(x) \pmod{x^{2t+s}},$$

$$\deg b(x) = \deg r_j(x) \leqslant t + s - 1,$$
$$\deg a(x) = \deg v_j(x) = \deg x^{2t+s} - \deg r_{j-1}(x) \leqslant 2t + s - t - s = t.$$

Поэтому такой алгоритм приводит к решению $a(x)$ и $b(x)$ сравнения (8) с условием (9).

Теорема 4. Пусть $v_j(x)$ и $r_j(x)$ — многочлены из обобщенного алгоритма Евклида с условием (11). Тогда найдется такая ненулевая константа $\lambda \in F$, для которой $\sigma(x) = \lambda v_j(x)$, $\widetilde{\omega}(x) = \lambda r_j(x)$.

Доказательство. Для многочленов $v_j(x)$ и $r_j(x)$, а также для многочленов $\sigma(x)$ и $\widetilde{\omega}(x)$ выполнены равенства:

$$u_j(x)x^{2t+s} + v_j(x)\widetilde{S}(x) = r_j(x), \tag{13}$$
$$\widetilde{\Phi}(x)x^{2t+s} + \sigma(x)\widetilde{S}(x) = \widetilde{\omega}(x). \tag{14}$$

Домножив обе части первого равенства на $\sigma(x)$, а второго — на $v_j(x)$, получим:

$$\begin{array}{l} \sigma(x)u_j(x)x^{2t+s} + \sigma(x)v_j(x)\widetilde{S}(x) = \sigma(x)r_j(x), \\ v_j(x)\widetilde{\Phi}(x)x^{2t+s} + v_j(x)\sigma(x)\widetilde{S}(x) = v_j(x)\widetilde{\omega}(x). \end{array} \tag{15}$$

Из данных равенств следует сравнение:

$$\sigma(x)r_j(x) \equiv v_j(x)\widetilde{\omega}(x) \pmod{x^{2t+s}}.$$

Учитывая степени многочленов в данном сравнении, получаем равенство:

$$\sigma(x)r_j(x) = v_j(x)\widetilde{\omega}(x).$$

Поэтому из (15) с учетом последнего равенства следует такое равенство:

$$\sigma(x)u_j(x) = v_j(x)\widetilde{\Phi}(x).$$

Из свойства взаимной простоты многочленов $u_j(x)$ и $v_j(x)$ следует, что $v_j(x) \,|\, \sigma(x)$, поэтому для некоторого многочлена $\mu(x)$ выполнено $\sigma(x) = \mu(x)v_j(x)$. Подставим это равенство в (14):

$$\widetilde{\Phi}(x)x^{2t+s} + \mu(x)v_j(x)\widetilde{S}(x) = \widetilde{\omega}(x).$$

Теперь домножим равенство (13) на $\mu(x)$:

$$\mu(x)u_j(x)x^{2t+s} + \mu(x)v_j(x)\widetilde{S}(x) = \mu(x)r_j(x).$$

Учитывая степени многочленов $\omega(x)$, $\mu(x)$ и $r_j(x)$, из последних двух равенств следует равенство $\widetilde{\omega}(x) = \mu(x)r_j(x)$.

Таким образом, $\sigma(x) = \mu(x)v_j(x)$, $\widetilde{\omega}(x) = \mu(x)r_j(x)$. Так как многочлены $\sigma(x)$ и $\widetilde{\omega}(x)$ взаимно просты, то многочлен $\mu(x)$ является ненулевой константой. □

Замечание. Вернемся к вопросу о наличии в векторе α нулевой компоненты. В предыдущем алгоритме этот факт не имел значения. Здесь же нужно это учитывать. Предположим, что $\alpha_i = 0$. Пусть на i-й позиции кодового вектора произошла ошибка. Так как $X_0 = \alpha_i = 0$, то данный локатор ошибки не оказывает влияния на многочлен $\sigma(x)$. С помощью данного многочлена можно найти только ненулевые локаторы ошибок и соответствующие им значения ошибок. Поэтому в самом конце следующего алгоритма после нахождения всех ошибок, соответствующих ненулевым локаторам, необходимо проверить, была ли еще одна ошибка на i-й позиции. Пусть $\widetilde{u}$ — вектор, в котором исправлены все ошибки в векторе $\widetilde{v}$, соответствующие ненулевым локаторам. Так как i-й столбец матрицы H имеет только одно ненулевое значение на первой позиции, равное w_i, то необходимо найти значение Z_0, равное скалярному произведению вектора $\widetilde{u}$ на первую строку матрицы H. Если $Z_0 = 0$, то на i-й позиции ошибок не было. В противном случае значение ошибки на i-й позиции равно $Y_0 = Z_0 w_i^{-1}$.

Пусть теперь на i-й позиции произошло стирание. Тогда значение ошибки равно $Y_0 = Z_0 w_i^{-1}$.

Алгоритм 2 (декодирование ОРС кодов на основе алгоритма Сугиямы на случай ошибок и стираний).

Вход: принятый вектор v, в котором s стираний и не более t ошибок.

Выход: исходный кодовый вектор u, если $d \geqslant 2t + s + 1$.

1. Определяется $t = [(d-s-1)/2]$. В векторе v все стирания заменяются нулями, получая тем самым вектор $\widetilde{v}$. Находятся компоненты $S_0, S_1, \ldots, S_{2t+s-1}$ синдромного вектора $\widetilde{v}H^T$. Если они все равны нулю, то возвращается вектор $\widetilde{v}$, и процедура окончена.

Вычисляются значения локаторов стираний $X_{t+1} = \alpha_{i_{t+1}}, \ldots, X_{t+s} = \alpha_{i_{t+s}}$ на основе известных позиций стираний $i_{t+1}, \ldots, i_{t+s}$. Вычисляются коэффициенты модифицированного синдромного многочлена $\widetilde{S}(x)$.

2. Пусть $r_{-1}(x) = x^{2t+s}$, $r_0(x) = \widetilde{S}(x)$, $v_{-1}(x) = 0$, $v_0(x) = 1$. С помощью обобщенного алгоритма Евклида производится последовательность вычислений ($i \geqslant 1$):

$$r_{i-2}(x) = r_{i-1}(x)q_{i-1}(x) + r_i(x),$$

$$v_i(x) = v_{i-2}(x) - v_{i-1}(x)q_{i-1}(x).$$

Процесс прекращается, как только для некоторого $r_j(x)$ будет выполнено:

$$\deg r_{j-1}(x) \geqslant t+s, \quad \deg r_j(x) \leqslant t+s-1. \tag{16}$$

Тогда:

$$\sigma(x) = \lambda v_j(x), \quad \widetilde{\omega}(x) = \lambda r_j(x),$$

где константа $\lambda \in F$ задается так, чтобы удовлетворялось условие $\sigma(0) = 1$. Пусть $l = \deg \sigma(x)$.

3. Отыскиваются l корней многочлена $\sigma(x)$ последовательной подстановкой в него ненулевых элементов поля F. При этом локаторы ошибок — это величины, обратные корням многочлена $\sigma(x)$.

4. При вычислении значений ошибок выполняется один из следующих пунктов.

4.1. Если среди локаторов стираний $X_{t+1}, \ldots, X_{t+s}$ имеется нулевое значение (в противном случае переходим в пункт 4.2), скажем, $X_p = 0$, то пусть:

$$M = \{1, \ldots, l\} \cup \{t+1, \ldots, t+s\} \backslash \{p\}$$

— множество индексов локаторов ошибок и стираний без учета индекса p. Находятся Z_j, $j \in M$, например, с помощью алгоритма Форни для обобщенных кодов РС:

$$Z_j = \frac{\widetilde{\omega}(X_j^{-1})}{\prod_{i \in M \backslash \{j\}} (1 - X_i X_j^{-1})}, \quad j \in M. \tag{17}$$

После этого находятся значения ошибок $Y_j = Z_j / w_{i_j}$, $j \in M$. У вектора $\widetilde{v}$ из i_j-го символа, $X_j = \alpha_{i_j}$, вычитается значение Y_j, $j \in M$. При этом получается вектор $\widetilde{u}$. Пусть для некоторого i выполнено $\alpha_i = 0$ (в противном случае все локаторы стираний были бы ненулевые). Вычисляется значение Z_p, равное скалярному произведению вектора $\widetilde{u}$ на первую строку матрицы H. Вычисляется значение ошибки $Y_p = Z_p / w_i$. Осталось в векторе $\widetilde{u}$ из i-го символа вычесть Y_p.

4.2. Если условие 4.1 не выполнено, то пусть $M = \{1, \ldots, l\} \cup \{t+1, \ldots, t+s\}$. По формуле (17) находятся значения Z_j, затем значения ошибок $Y_j = Z_j / w_{i_j}$, $j \in M$. У вектора $\widetilde{v}$ из i_j-го символа, $X_j = \alpha_{i_j}$, вычитается значение Y_j, $j \in M$. При этом получается вектор $\widetilde{u}$.

Если $\alpha_i = 0$ для некоторого i, то вычисляется значение Z_0, равное скалярному произведению вектора $\widetilde{u}$ на первую строку матрицы H. Если $Z_0 \neq 0$, то вычисляется значение ошибки $Y_0 = Z_0 / w_i$. Осталось в векторе $\widetilde{u}$ из i-го символа вычесть Y_0.

Пример 2. Продолжим рассмотрение примера 1, в котором рассматривался код $GRS_4(\alpha, y)$. Порождающая и проверочная матрицы кода $GRS_4(\alpha, y)$, учитывая теорему 1, будут иметь вид:

$$G = \begin{pmatrix} 2 & 1 & 3 & 1 & 4 & 1 & 5 & 1 & 6 \\ 0 & 1 & 6 & 3 & 5 & 5 & 8 & 7 & 4 \\ 0 & 1 & 1 & 9 & 9 & 3 & 4 & 5 & 10 \\ 0 & 1 & 2 & 5 & 3 & 4 & 2 & 2 & 3 \end{pmatrix}, H = \begin{pmatrix} 10 & 5 & 7 & 2 & 9 & 2 & 2 & 5 & 7 \\ 0 & 5 & 3 & 6 & 3 & 10 & 1 & 2 & 1 \\ 0 & 5 & 6 & 7 & 1 & 6 & 6 & 3 & 8 \\ 0 & 5 & 1 & 10 & 4 & 8 & 3 & 10 & 9 \\ 0 & 5 & 2 & 8 & 5 & 7 & 7 & 4 & 6 \end{pmatrix},$$

где первая строка матрицы H совпадает с вектором w.

Пусть на приемном конце получен тот же вектор $v = (1, 3, 6, 10, 9, 1, 10, *, 8)$. Применим алгоритм декодирования 2.

1. Определяем $s = 1$, $t = [(d - s - 1)/2] = 2$. Заменим в векторе v все стирания нулями:

$$\widetilde{v} = (1, 3, 6, 10, 9, 1, 10, 0, 8).$$

Находим синдромный вектор $(S_0, S_1, S_2, S_3, S_4) = \widetilde{v} H^T = (4, 5, 7, 3, 2)$. Вычисляем известные локаторы стираний: $X_3 = \alpha_7 = 7$. Поэтому

$$\widetilde{S}(x) = S(x)\nu(x) = (4 + 5x + 7x^2 + 3x^3 + 2x^4)(1 - 7x) =$$

$$= 4 + 10x + 5x^2 + 9x^3 + 3x^4 + 8x^5.$$

2. Определяем $r_{-1}(x) = x^5$, $r_0(x) = \widetilde{S}(x)$, $v_{-1}(x) = 0$, $v_0(x) = 1$. Выполняем неполный алгоритм Евклида:

$$\begin{aligned} r_{-1}(x) &= r_0(x)q_0(x) + r_1(x),\\ q_0(x) &= 7,\\ r_1(x) &= 5 + 7x + 9x^2 + 3x^3 + x^4,\\ v_1(x) &= v_{-1}(x) - q_0(x)v_0(x) = 4,\\ r_0(x) &= r_1(x)q_1(x) + r_2(x),\\ q_1(x) &= 1 + 8x,\\ r_2(x) &= 10 + 7x + 6x^2,\\ v_2(x) &= v_0(x) - q_1(x)v_1(x) = 8 + x. \end{aligned}$$

Так как $t + s = 3$, $\deg r_1(x) = 4$, $\deg r_2(x) < 3$, то после второго шага останавливаемся. Тогда

$$\sigma(x) = \lambda v_2(x), \quad \widetilde{\omega}(x) = \lambda r_2(x).$$

При $\lambda = 7$ получаем $\sigma(0) = 1$, поэтому:

$$\sigma(x) = 1 + 7x, \quad \widetilde{\omega}(x) = 4 + 5x + 9x^2.$$

3. Корнем многочлена $\sigma(x)$ является $x_1 = 3$, поэтому $X_1 = x_1^{-1} = 4 = \alpha_4$. Это значит, что ошибка произошла на 4-й позиции. Итак, на 4-й позиции вектора $\widetilde{v}$ точно имеется ошибка, а на позиции 7, возможно, есть ошибки (после замены стертых символов нулями мы могли поставить некоторые символы верно).

4. Так как среди локаторов стираний нет нулевых, то попадаем в пункт 4.2 алгоритма 2. Находим значения ошибок:

$$Z_1 = \frac{\widetilde{\omega}(X_1^{-1})}{(1 - X_3X_1^{-1})} = 6, \quad Y_1 = Z_1w_4^{-1} = 8,$$

$$Z_3 = \frac{\widetilde{\omega}(X_3^{-1})}{(1 - X_1X_3^{-1})} = 2, \quad Y_3 = Z_3w_7^{-1} = 7.$$

Вычитая из 4-й и 7-й позиций в векторе $\widetilde{v}$ соответственно значения 8 и 7, получаем вектор:

$$\widetilde{u} = (1, 3, 6, 10, 1, 1, 10, 4, 8).$$

Осталось проверить, была ли ошибка на 0-й позиции, которая соответствует элементу $\alpha_0 = 0$. Вычисляя скалярное произведение вектора $\widetilde{u}$ на первую строку матрицы H, получаем $Z_0 = 7$. Это означает, что на 0-й позиции имеется ошибка со значением $Y_0 = Z_0w_0^{-1} = 4$. Окончательно:

$$u = \widetilde{u} - (4, 0, 0, 0, 0, 0, 0) = (8, 3, 6, 10, 1, 1, 10, 4, 8).$$

3. Декодирование ОРС кодов на основе алгоритма Берлекэмпа — Месси (алгоритма Питерсона — Горенстейна — Цирлера)

Продолжим рассмотрение сравнения (6). Пусть $d \geqslant 2t + s + 1$,

$$\widetilde{S}(x) = \widetilde{S}_0 + \widetilde{S}_1x + \ldots + \widetilde{S}_{2t+2s-1}x^{2t+2s-1} =$$

$$= S(x)\nu(x) = (S_0 + S_1x + \ldots + S_{2t+s-1}x^{2t+s-1})(\nu_0 + \nu_1x + \ldots + \nu_sx^s),$$

где $\nu_0 = 1$, $\nu_i = (-1)^i\sigma_i(X_{t+1}, \ldots, X_{t+s})$ — элементарный симметрический многочлен от $X_{t+1}, \ldots, X_{t+s}$, $i = 1, \ldots, s$.

Так как в сравнении (6) $\deg\widetilde{\omega}(x) \leqslant t + s - 1$, $\deg\widetilde{S}(x) \leqslant 2t + 2s - 1$, $\deg\sigma(x) \leqslant t$, то необходимым условием выполнения данного сравнения является тот факт, что коэффициенты многочлена $\sigma(x)\widetilde{S}(x)$ при степенях $j = t + s, t + s + 1, \ldots, 2t + s - 1$ равны нулю. Поэтому получаем такую систему линейных уравнений:

$$\begin{cases} \sigma_0\widetilde{S}_{s+t} + \sigma_1\widetilde{S}_{s+t-1} + \ldots + \sigma_t\widetilde{S}_s = 0,\\ \sigma_0\widetilde{S}_{s+t+1} + \sigma_1\widetilde{S}_{s+t} + \ldots + \sigma_t\widetilde{S}_{s+1} = 0,\\ \ldots\\ \sigma_0\widetilde{S}_{s+2t-1} + \sigma_1\widetilde{S}_{s+2t-2} + \ldots + \sigma_t\widetilde{S}_{s+t-1} = 0. \end{cases}$$

Так как $\sigma_0 = 1$, то данная система в матричной форме примет такой вид:

$$\begin{pmatrix} \widetilde{S}_s & \widetilde{S}_{s+1} & \ldots & \widetilde{S}_{s+t-1}\\ \widetilde{S}_{s+1} & \widetilde{S}_{s+2} & \ldots & \widetilde{S}_{s+t}\\ \ldots & \ldots & \ldots & \ldots\\ \widetilde{S}_{s+t-1} & \widetilde{S}_{s+t} & \ldots & \widetilde{S}_{s+2t-2} \end{pmatrix} \begin{pmatrix} \sigma_t\\ \sigma_{t-1}\\ \ldots\\ \sigma_1 \end{pmatrix} = \begin{pmatrix} -\widetilde{S}_{s+t}\\ -\widetilde{S}_{s+t+1}\\ \ldots\\ -\widetilde{S}_{s+2t-1} \end{pmatrix}. \tag{18}$$

Обозначим матрицу этой системы через $M(t,s)$. Выясним, в каком случае эта система разрешима.

Лемма. Для любого $j = 0, 1, \ldots, 2t-2$ выполнено равенство:

$$\widetilde{S}_{s+j} = \sum_{k=1}^{t} Z_k X_k^j \prod_{i=1}^{s}(X_k - X_{t+i}).$$

Доказательство. Пусть, как и ранее, $\nu_i = (-1)^i \sigma_i(X_{t+1}, \ldots, X_{t+s})$ — элементарный симметрический многочлен от $X_{t+1}, \ldots, X_{t+s}$. Обозначим $\sigma_i = \sigma_i(X_{t+1}, \ldots, X_{t+s})$. Учитывая определение многочлена $\widetilde{S}(x)$ и равенство:

$$\sum_{i=0}^{s}(-1)^{s-i}\sigma_{s-i}X_k^i = \begin{cases} 0, & t+1 \leqslant k \leqslant t+s, \\ \prod_{i=1}^{s}(X_k - X_{t+i}), & k \notin \{t+1, \ldots, t+s\}, \end{cases}$$

получаем:

$$\widetilde{S}_{s+j} = \sum_{i=0}^{s} S_{j+i}\nu_{s-i} = \sum_{i=0}^{s}\sum_{k=1}^{t+s} Z_k X_k^{j+i}(-1)^{s-i}\sigma_{s-i} =$$

$$= \sum_{k=1}^{t+s} Z_k X_k^j \sum_{i=0}^{s} X_k^i (-1)^{s-i}\sigma_{s-i} = \sum_{k=1}^{t} Z_k X_k^j \sum_{i=0}^{s} X_k^i (-1)^{s-i}\sigma_{s-i} =$$

$$= \sum_{k=1}^{t} Z_k X_k^j \prod_{i=1}^{s}(X_k - X_{t+i}).$$

□

Теорема 5. Пусть произошло s стираний. Матрица $M(t,s)$ невырождена тогда и только тогда, когда произошло t ошибок.

Доказательство. Обозначим через A, B, C следующие квадратные матрицы порядка t:

$$A = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ X_1 & X_2 & \ldots & X_t \\ \ldots & \ldots & \ldots & \ldots \\ X_1^{t-1} & X_2^{t-1} & \ldots & X_t^{t-1} \end{pmatrix}, B = \begin{pmatrix} Z_1 & 0 & \ldots & 0 \\ 0 & Z_2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & Z_t \end{pmatrix},$$

$$C = \begin{pmatrix} \prod_{i=1}^{s}(X_1 - X_{t+i}) & 0 & \ldots & 0 \\ 0 & \prod_{i=1}^{s}(X_2 - X_{t+i}) & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \prod_{i=1}^{s}(X_t - X_{t+i}) \end{pmatrix}.$$

Покажем, что $M(t,s) = ABCA^T$. По лемме элемент матрицы $M(t,s)$ с индексами i и j равен:

$$(M(t,s))_{ij} = \widetilde{S}_{s+i+j-2} = \sum_{k=1}^{t} Z_k X_k^{i+j-2} \prod_{l=1}^{s}(X_k - X_{t+l}).$$

Учитывая, что

$$(A)_{ij} = X_j^{i-1}, \quad (BC)_{ij} = \delta_{ij} Z_i \prod_{k=1}^{s}(X_i - X_{t+k}),$$

где δ_{ij} — символ Кронекера, найдем элемент с соответствующими индексами матрицы $ABCA^T$:

$$(ABCA^T)_{ij} = \sum_{m=1}^{t}(ABC)_{im}(A^T)_{mj} = \sum_{m=1}^{t}\sum_{k=1}^{t} A_{ik}(BC)_{km}A_{jm} =$$

$$= \sum_{m=1}^{t}\sum_{k=1}^{t} X_k^{i-1}\delta_{km} Z_k \prod_{l=1}^{s}(X_k - X_{t+l}) X_m^{j-1} = \sum_{k=1}^{t} Z_k X_k^{i+j-2} \prod_{l=1}^{s}(X_k - X_{t+l}).$$

Следовательно, $M(t,s) = ABCA^T$.

Если произошло t ошибок, то все $X_1, \ldots, X_{t+s}$ различные, а все $Z_1, \ldots, Z_t$ отличны от нуля, поэтому определитель матрицы $ABCA^T$ отличен от нуля. Если произошло менее чем t ошибок, то хотя бы один диагональный элемент матрицы B равен нулю, поэтому матрица $ABCA^T$ будет вырожденной. □

Алгоритм 3 (декодирование ОРС кодов на основе алгоритма Берлекэмпа — Месси на случай ошибок и стираний).

Вход: принятый вектор v, в котором s стираний и не более t ошибок.

Выход: исходный кодовый вектор u, если $d \geqslant 2t+s+1$.

1. Определяется $t = [(d-s-1)/2]$. В векторе v все стирания заменяются нулями, получая тем самым вектор $\widetilde{v}$. Находятся компоненты $S_0, S_1, \ldots, S_{2t+s-1}$ синдромного вектора $\widetilde{v}H^T$. Если они все равны нулю, то возвращается вектор $\widetilde{v}$, и процедура окончена.

Вычисляются значения локаторов стираний $X_{t+1} = \alpha_{i_{t+1}}, \ldots, X_{t+s} = \alpha_{i_{t+s}}$ на основе известных позиций стираний $i_{t+1}, \ldots, i_{t+s}$. Вычисляются коэффициенты модифицированного синдромного многочлена $\widetilde{S}(x)$.

2. Определяется $h := t$.

Цикл: пока $|M(h,s)| = 0$, переопределить $h := h-1$.

Если $h > 0$, то находятся $\sigma_1, \ldots, \sigma_h$ — решение системы (18). Это можно сделать с помощью алгоритма Берлекэмпа — Месси (или методом Гаусса). После этого составляется многочлен $\sigma(x)$. Пусть $l = \deg\sigma(x)$.

3. Отыскиваются l корней многочлена $\sigma(x)$ последовательной подстановкой в него ненулевых элементов поля F. При этом локаторы ошибок — это величины, обратные корням многочлена $\sigma(x)$.

4. При вычислении значений ошибок выполняется один из следующих пунктов.

4.1. Если среди локаторов стираний $X_{t+1}, \ldots, X_{t+s}$ имеется нулевое значение (в противном случае переходим в пункт 4.2), скажем, $X_p = 0$, то пусть

$$M = \{1, \ldots, l\} \cup \{t+1, \ldots, t+s\} \backslash \{p\}.$$

Находятся Z_j, $j \in M$, например, с помощью формул Форни (17). После этого находятся значения ошибок $Y_j = Z_j/w_{i_j}$, $j \in M$. У вектора $\widetilde{v}$ из i_j-го символа, $X_j = \alpha_{i_j}$, вычитается значение Y_j, $j \in M$. При этом получается вектор $\widetilde{u}$. Пусть для некоторого i выполнено $\alpha_i = 0$. Вычисляется значение Z_p, равное скалярному произведению вектора $\widetilde{u}$ на первую строку матрицы H. Вычисляется значение ошибки $Y_p = Z_p/w_i$. Осталось в векторе $\widetilde{u}$ из i-го символа вычесть Y_p.

4.2. Если условие 4.1 не выполнено, то пусть $M = \{1, \ldots, l\} \cup \{t+1, \ldots, t+s\}$. По формуле (17) находятся значения Z_j, затем значения ошибок $Y_j = Z_j/w_{i_j}$, $j \in M$. У вектора $\widetilde{v}$ из i_j-го символа, $X_j = \alpha_{i_j}$, вычитается значение Y_j, $j \in M$. При этом получается вектор $\widetilde{u}$.

Если $\alpha_i = 0$ для некоторого i и $\deg\sigma(x) < h$, то вычисляется значение Z_0, равное скалярному произведению вектора $\widetilde{u}$ на первую строку матрицы H, а затем вычисляется значение ошибки $Y_0 = Z_0/w_i$. Осталось в векторе $\widetilde{u}$ из i-го символа вычесть Y_0.

Пример 3. Продолжим рассматривать примеры 1 и 2. Пусть на приемной стороне получен все тот же вектор $v = (1,3,6,10,9,1,10,*,8)$. После замены стертых символов нулями получаем вектор $\widetilde{v} = (1,3,6,10,9,1,10,0,8)$. Компоненты синдромного вектора $\widetilde{S}$ вычислены в предыдущем примере: $\widetilde{S} = (4,10,5,9,3,8)$. Определяем $s = 1$, $t = [(d-s-1)/2] = 2$. Составляем матрицу системы (18):

$$\left(\begin{array}{cc|c} \widetilde{S}_1 & \widetilde{S}_2 & -\widetilde{S}_3 \\ \widetilde{S}_2 & \widetilde{S}_3 & -\widetilde{S}_4 \end{array}\right) = \left(\begin{array}{cc|c} 10 & 5 & -9 \\ 5 & 9 & -3 \end{array}\right).$$

Так как определитель матрицы $M(2,1)$ ненулевой, то из данной системы находим $\sigma_1 = 7$, $\sigma_2 = 0$. Поэтому $\sigma(x) = 1 + 7x$. После этого осталось повторить шаги 3 и 4 предыдущего примера. Только проверять, была ли ошибка на 0-й позиции, не нужно, так как факт ее наличия следует из неравенств $\deg\sigma(x) < 2$, $|M(2,1)| \neq 0$.

Литература

[1] Блейхут Р. Теория и практика кодов, контролирующих ошибки / пер. с англ. Москва: Мир, 1986. 576 с. URL: http://publ.lib.ru/ARCHIVES/B/BLEYHUT_Richard_E/_Bleyhut_R.E..html.

[2] Gao S. A new algorithm for decoding Reed–Solomon codes. In: Bhargava V.K., Poor H.V., Tarokh V., Yoon S. (eds) // Communications, Information and Network Security. The Springer International Series in Engineering and Computer Science (Communications and Information Theory). V. 712. Boston: Springer, MA. DOI: https://doi.org/10.1007/978-1-4757-3789-9_5.

[3] Huffman W.C., Pless V. Fundamentals of Error-Correcting Codes. Cambridge: Cambridge University Press, 2003. 646 p. DOI: https://doi.org/10.1017/CBO9780511807077.

[4] Рацеев С.М. Об алгоритмах декодирования кодов Гоппы // Челяб. физ.-матем. журн. 2020. Т. 5, № 3. С. 327–341. DOI: https://doi.org/10.47475/2500-0101-2020-15307.

[5] Рацеев С.М., Череватенко О.И. О простом алгоритме декодировании кодов БЧХ, кодов Рида — Соломона и кодов Гоппы // Вестник СибГУТИ. 2020. № 3. С. 3–14. URL: https://www.elibrary.ru/item.asp?id=44408789.

[6] Рацеев С.М., Череватенко О.И. Об алгоритмах декодирования обобщенных кодов Рида — Соломона // Системы и средства информатики. 2020. Т. 30, № 4. С. 83–94. DOI: https://doi.org/10.14357/08696527200408.

[7] Patterson N.J. The algebraic decoding of Goppa codes // IEEE Transactions on Information Theory. 1975. Vol. 21, № 2. P. 203–207. DOI: https://doi.org/10.1109/TIT.1975.1055350.

[8] McEliece R.J. A Public-Key Cryptosystem Based On Algebraic Coding Theory // DSN Progress Report 1978. 42–44. P. 114–116.

[9] Marek Repka, Pavol Zaj. Overview of the Mceliece Cryptosystem and its Security // Tatra Mountains Mathematical Publications. 2014. Vol. 60. P. 57–83. DOI: http://doi.org/10.2478/tmmp-2014-0025.

[10] Bernstein Daniel J. List decoding for binary Goppa codes. In: Chee Y.M. et al. (eds) Coding and Cryptology. IWCC 2011. Lecture Notes in Computer Science, vol 6639. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-20901-7_4.

[11] Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process: Internal Report 8240. National Institute of Standards and Technology, January, 2019. 27 p. DOI: https://doi.org/10.6028/NIST.IR.8240

[12] Федоренко С.В. Простой алгоритм декодирования алгебраических кодов // Информационно-управляющие системы. 2008. № 3. С. 23–27. URL: https://cyberleninka.ru/article/n/prostoy-algoritm-dekodirovaniya-algebraicheskih-kodov/viewer.

Scientific article

DOI: 10.18287/2541-7525-2020-26-3-17-29

Submited: 13.03.2020
Revised: 27.03.2020
Accepted: 25.05.2020

S.M. Ratseev
Ulyanovsk State University, Ulyanovsk, Russian Federation
E-mail: ratseevsm@mail.ru. ORCID: https://orcid.org/0000-0003-4995-9418
O.I. Cherevatenko
Ulyanovsk State University of Education, Ulyanovsk, Russian Federation
E-mail: chai@pisem.net. ORCID: https://orcid.org/0000-0003-3931-9425

ON DECODING ALGORITHMS FOR GENERALIZED REED — SOLOMON CODES WITH ERRORS AND ERASURES

ABSTRACT

The article is devoted to the decoding algorithms for generalized Reed — Solomon codes with errors and erasures. These algorithms are based on Gao algorithm, Sugiyama algorithm, Berlekamp — Massey algorithm (Peterson — Gorenstein — Zierler algorithm). The first of these algorithms belongs to syndrome-free decoding algorithms, the others — to syndrome decoding algorithms. The relevance of these algorithms is that they are applicable for decoding Goppa codes, which are the basis of some promising post-quantum cryptosystems. These algorithms are applicable for Goppa codes over an arbitrary field, as opposed to the well-known Patterson decoding algorithm for binary Goppa codes.

Key words: error-correcting codes, Reed — Solomon codes, Goppa codes, code decoding.

Citation. Ratseev S.M., Cherevatenko O.I. On decoding algorithms for generalized Reed — Solomon codes with errors and erasures. *Vestnik Samarskogo universiteta. Estestvennonauchnaia seriia = Vestnik of Samara University. Natural Science Series*, 2020, vol. 26, no. 3, pp. 17–29. DOI: http://doi.org/10.18287/2541-7525-2020-26-3-17-29. (In Russ.)

Information about the conflict of interests: authors and reviewers declare no conflict of interests.

Ratseev Sergey Mihaylovich — Doctor of Physical and Mathematical Sciences, associate professor, Department of Information Security and Control Theory, Ulyanovsk State University, 42, Leo Tolstoy Street, Ulyanovsk, 432017, Russian Federation.

Cherevatenko Olga Ivanovna — Candidate of Physical and Matheatical Sciences, associate professor, Department of Higher Mathematics, Ulyanovsk State University of Education, 4/5, Lenin Square, Ulyanovsk, 432063, Russian Federation.

References

[1] Blahut, Richard E. Theory and practice of error control codes. Translation from English. Moscow: Mir, 1986, 576 p. Available at: http://publ.lib.ru/ARCHIVES/B/BLEYHUT_Richard_E/_Bleyhut_R.E..html. (In Russ.)

[2] Gao S. A new algorithm for decoding Reed—Solomon codes. In: Bhargava V.K., Poor H.V., Tarokh V., Yoon S. (eds) Communications, Information and Network Security. The Springer International Series in Engineering and Computer Science (Communications and Information Theory), vol 712. Springer, Boston, MA. DOI: https://doi.org/10.1007/978-1-4757-3789-9_5.

[3] Huffman W. Cary. Fundamentals of Error-Correcting Codes. Cambridge: Cambridge University Press, 2003. 646 p. DOI: https://doi.org/10.1017/CBO9780511807077.

[4] Ratseev S.M. On decoding algorithms for Goppa codes. *Chelyabinskiy Fiziko-Matematicheskiy Zhurnal* [Chelyabinsk Physical and Mthematical Journal], 2020, vol. 5, no. 3, pp. 327–341. DOI: https://doi.org/10.47475/2500-0101-2020-15307. (In Russ.)

[5] Ratseev S.M., Cherevatenko O.I. On a simple algorithm for decoding BCH codes, Reed — Solomon codes, and Goppa codes. *(*Vestnik SibGUTI), 2020, no. 3(51), pp. 3–14. Available at: https://www.elibrary.ru/item.asp?id=44408789. (In Russ.)

[6] Ratseev S.M., Cherevatenko O.I. On decoding algorithms for generalized Reed — Solomon codes. *Sistemy i sredstva informatiki* [Systems and Means of Informatics], 2020, vol. 30, no. 4, pp. 83–94. DOI: https://doi.org/10.14357/08696527200408. (In Russ.)

[7] Patterson N.J. The algebraic decoding of Goppa codes. *IEEE Transactions on Information Theory*, 1975, vol. 21, issue 2, pp. 203–207. DOI: https://doi.org/10.1109/TIT.1975.1055350.

[8] McEliece R.J. A Public-Key Cryptosystem Based On Algebraic Coding Theory. *DSN Progress Report*, 1978, vol. 42–44, pp. 114–116.

[9] Marek Repka, Pavol Zaj. Overview of the Mceliece Cryptosystem and its Security. *Tatra Mountains Mathematical Publications*, 2014, vol. 60, pp. 57–83. DOI: http://doi.org/10.2478/tmmp-2014-0025.

[10] Bernstein Daniel J. List decoding for binary Goppa codes. In: Chee Y.M. et al. (eds) Coding and Cryptology. IWCC 2011. Lecture Notes in Computer Science, vol 6639. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-20901-7_4.

[11] Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process: Internal Report 8240. National Institute of Standards and Technology, January, 2019, 27 p. DOI: https://doi.org/10.6028/NIST.IR.8240.

[12] Fedorenko S.V. A simple algorithm for decoding algebraic codes. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2008, no. 3, pp. 23–27. Available at: https://cyberleninka.ru/article/n/prostoy-algoritm-dekodirovaniya-algebraicheskih-kodov/viewer. (In Russ.)