Математика и ее приложения : ЖИМО. 2012. Вып. 1 (9). С. 31–40

УДК 512.543

В. М. Деундяк[1], Е. С. Чекунов[2]

Математическая модель списочного декодера Бернштейна

Ключевые слова: бинарные коды Гоппы, списочный декодер, решетки над кольцами многочленов, LLL–алгоритм, минимальный вектор решетки.

В работе рассмотрены две версии LLL–алгоритма: алгоритм приведения базиса многомерной решетки над кольцом многочленов с коэффициентами из поля рациональных функций и быстрый алгоритм приведения базиса двумерной решетки над кольцом многочленов с коэффициентами из поля Галуа. На основе применения этих алгоритмов и алгоритма Паттерсона декодирования бинарных кодов Гоппы построена математическая модель списочного декодера Бернштейна для таких кодов.

Key words: binary Goppa codes, list decoding algorithm, lattices over polynomial rings, LLL–algorithm, minimum-length vector of lattice.

We consider two versions of LLL–algorithm: the basis reduction algorithm for n–dimensional lattice over ring of polynomials with coefficients in the field of rational functions and fast basis reduction algorithm for two–dimensional lattice over ring of polynomials with coefficients in a Galois field. Based on the application of LLL–algorithms and Patterson's decoding algorithm of binary Goppa codes we construct the mathematical model of Bernstein's list decoding algorithm for binary Goppa codes.

1. Введение

С ростом производительности вычислительных средств становится актуальным вопрос усиления защиты кодовых криптосистем типа Мак-Элиса [7] от несанкционированного доступа (НСД). Это достигается как за счет выбора помехоустойчивого кода, так и методов кодирования и декодирования. Известно, что классическая криптосистема Мак-Элиса на бинарных кодах Гоппы до сих пор является стойкой к атакам на секретный ключ [3], [4], [9]. Однако для таких криптосистем существуют детерминированные алгоритмы атак на шифрограмму (см., например, [4]). Отметим, что атака из [3], самая быстрая из такого класса атак, позволяет восстанавливать шифрограмму за неделю с помощью кластера из 200 компьютеров 2.4 ГГц Core 2 Quad. Чтобы противостоять такой атаке, в [3] предложено усилить криптосистему за счет увеличения параметров кода, увеличения количества искусственных ошибок в протоколе и замены стандартного метода декодирования Паттерсона на списочный декодер [2].

[1]Южный федеральный университет; ФГНУ НИИ «Спецвузавтоматика»; E-mail: vlade@math.rsu.ru

[2]Южный федеральный университет; E-mail: echekunov@gmail.com

Цель данной работы — построить математическую модель списочного декодера Бернштейна на основе подхода из [2]. Для этого разработаны две версии LLL–алгоритма: алгоритм приведения базиса многомерной решетки над кольцом многочленов с коэффициентами из поля рациональных функций и быстрый алгоритм приведения базиса двумерной решетки над кольцом многочленов с коэффициентами из поля Галуа. В дальнейшем на основе полученной математической модели предполагается построить имитационную модель усиленной кодовой криптосистемы Мак-Элиса на кодах Гоппы для использования в схемах защиты данных.

2. Решетки над кольцами многочленов и двумерный алгоритм приведения базиса решетки

В 2.1 приведена общая информация о решетках над кольцами многочленов и определена двумерная решетка Бернштейна над кольцом многочленов с коэффициентами из поля Галуа. В 2.2 построен быстрый алгоритм приведения базиса решетки Бернштейна.

2.1. Решетки над кольцами многочленов и двумерная решетка Бернштейна. Пусть $|X|$ мощность произвольного множества X, $\mathbb{F}[x]$ — кольцо многочленов с коэффициентами из поля Галуа $\mathbb{F}$. Определим функцию $\deg : \mathbb{F}[x] \to \mathbb{Z}$, которая каждому многочлену из $\mathbb{F}[x]$ ставит в соответствие его степень. $\mathbb{F}[x]$–решеткой, согласно [6], будем называть пару $(\mathcal{L},\ \mu)$, состоящую из свободного $\mathbb{F}[x]$–модуля $\mathcal{L}$ и функции $\mu : \mathcal{L} \to \mathbb{Z}$, такой, что:

$$\forall a, b \in \mathcal{L} : \mu(a+b) \le \max\{\mu(a);\ \mu(b)\},$$

$$\forall \alpha \in \mathbb{F}[x],\ a \in \mathcal{L} : \mu(\alpha a) = \deg \alpha + \mu(a),$$

$$\forall a \in \mathcal{L} : \mu(a) \ge 0,\ \mu(0) = 0,$$

$$\forall z \in \mathbb{Z} : |\{a \in \mathcal{L} : \mu(a) \le z\}| < \infty.$$

Решетку $\Lambda = (\mathcal{L}, \mu)$ будем называть n–мерной, если модуль $\mathcal{L}$ имеет ранг n. Пусть $\Lambda = (\mathcal{L}, \mu)$ — n–мерная $\mathbb{F}[x]$–решетка с базисом $b_1, b_2, \ldots, b_n$. Числа $\lambda_1, \lambda_2, \ldots, \lambda_n \in \mathbb{Z}$ называются последовательными минимумами решетки Λ, если существуют $v_1, v_2, \ldots, v_n \in \Lambda$ такие, что $\lambda_1 = \mu(v_1)$ — минимум $\mu(x)$ для всех $0 \ne x \in \Lambda$, $\lambda_2 = \mu(v_2)$ — минимум $\mu(x)$ для всех $x \in \Lambda$, линейно независимых с v_1, ... и, наконец, $\lambda_n = \mu(v_n)$ — минимум $\mu(x)$ для всех $x \in \Lambda$, линейно независимых с $v_1, v_2, \ldots, v_{n-1}$. Из определения следует, что $0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$. Свяжем с базисом решетки матрицу $B = (b_{ij}), i, j = 1 \ldots n$, строками которой являются базисные элементы решетки. Определителем решетки $\det(\Lambda)$ называют определитель матрицы B. Известно, что определитель решетки не зависит от выбора базиса [6].

Рассмотрим конечное поле $\mathbb{F}_{2^m}$, кольцо многочленов $\mathbb{F}_{2^m}[x]$ и поле рациональных функций $\mathbb{F}_{2^m}(x)$ над $\mathbb{F}_{2^m}$. Продолжим функцию deg в $\mathbb{F}_{2^m}(x)$ по следующему правилу:

$$\forall a \in \mathbb{F}_{2^m}(x) : \deg a = \deg \frac{\varphi}{\psi} = \deg \varphi - \deg \psi, \text{ где } \varphi, \psi \in \mathbb{F}_{2^m}[x].$$

Согласно [2], определим функцию $\|\cdot\| : \mathbb{F}_{2^m}(x) \to \mathbb{N} \cup \{0\}$ по правилу

$$\forall a \in \mathbb{F}_{2^m}(x) : \|a\| = \begin{cases} 2^{\deg a}, & \text{если } a \neq 0; \\ 0, & \text{если } a = 0. \end{cases} \tag{1}$$

Лемма 1. *Функция $\|\cdot\|$ — неархимедова норма в $\mathbb{F}_{2^m}(x)$.*

Доказательство. Покажем выполнение всех свойств для неархимедовой нормы. По определению $\|a\| \geq 0$ для всех $a \in \mathbb{F}_{2^m}(x)$ и $\|a\| = 0$, если $a = 0$. Рассмотрим произвольные элементы $a,\ b \in \mathbb{F}_{2^m}(x)$. Тогда

$$\|a \cdot b\| = 2^{\deg a \cdot b} = 2^{\deg a + \deg b} = 2^{\deg a} \cdot 2^{\deg b} = \|a\| \cdot \|b\|,$$

$$\|a + b\| = 2^{\deg(a+b)} \leq 2^{\max\{\deg a; \deg b\}} = \max\{2^{\deg a}; 2^{\deg b}\} = \max\{\|a\|; \|b\|\}.$$

■

Рассмотрим двумерную $\mathbb{F}[x]$–решетку Λ с некоторой нормой μ. Базис $a,\ b$ называется приведенным, если для любого $r \in \mathbb{F}[x]$: $\mu(a) \leq \mu(b) \leq \leq \mu(a + rb)$. Следующее утверждение показывает, что векторы приведенного базиса — минимальные (по норме μ) в решетке.

Лемма 2. *Пусть $\lambda_1, \lambda_2 \in \mathbb{Z}$ — последовательные минимумы двумерной решетки Λ. Если базис $b_1, b_2 \in \Lambda$ приведенный, то $\mu(b_i) = \lambda_i,\ i = 1, 2$.*

Доказательство. По определению $\mu(b_2 + rb_1) \geq \mu(b_2) \geq \mu(b_1)$. Пусть $v = \alpha_1 b_1 + \alpha_2 b_2$, где $v \neq 0$, $\alpha_1, \alpha_2 \in \mathbb{F}[x]$, — произвольный элемент решетки. Достаточно доказать, что $\mu(v) \geq \mu(b_1)$. Если $\alpha_2 = 0$, то $\mu(v) \geq \mu(b_1)$. Если $\alpha_2 \neq 0$, то $\alpha_1 = r\alpha_2 + s$, где $r, s \in \mathbb{F}[x]$, $0 \leq \deg s < \deg \alpha_2$. Тогда $v = sb_1 + \alpha_2(b_2 + rb_1)$. Применим неравенство треугольника:

$$\mu(v) \geq \mu(\alpha_2(b_2 + rb_1)) - \mu(sb_1) = (\deg \alpha_2 - \deg s)\mu(b_2 + rb_1) +$$

$$+ \deg s(\mu(b_2 + rb_1) - \mu(b_1)) \geq \mu(b_2 + rb_1) \geq \mu(b_2) \geq \mu(b_1).$$

■

В свободном $\mathbb{F}_{2^m}[x]$–модуле $\mathcal{L}_2$ ранга 2 определим функцию μ_2 по правилу: $\mu_2(a) = ||a_1^2 + xa_2^2||$ для всех $a = (a_1,\ a_2) \in \mathcal{L}_2$, где $||\cdot||$ — норма (1). Пара $\Lambda_2 = (\mathcal{L}_2, \mu_2)$ является $\mathbb{F}_{2^m}[x]$–решеткой Бернштейна [2].

2.2. Алгоритм приведения базиса двумерной решетки Бернштейна. Чтобы построить быстрый алгоритм приведения базиса двумерной решетки над кольцом многочленов, модифицируем алгоритм Ленстры из [5] на случай $\mathbb{F}_{2^m}[x]$–модулей и нормы μ_2.

Пусть $\bar{a} = (a_1, b_1), \bar{b} = (a_2, b_2)$ — базис решетки Λ_2. Определим

$$f(r) = \mu_2((a_2 - ra_1,\ b_2 - rb_2)) = (a_2 - ra_1)^2 + x(b_2 - rb_1)^2,\ r \in \mathbb{F}_{2^m}[x].$$

Поскольку $a_i, b_i \in \mathbb{F}_{2^m}[x]$, то $f(r) = a_2^2 + xb_2^2 + r^2(a_1^2 + xb_1^2)$. Функция f принимает минимальное значение, если $r = \lfloor\sqrt{(a_2^2 + xb_2^2)/(a_1^2 + xb_1^2)}\rfloor$. Отметим, что r можно взять равным $\lfloor a_2/a_1 \rfloor$, если $\deg b_i < \deg a_i$.

Алгоритм приведения базиса решетки $\bar{a},\ \bar{b} \in \Lambda_2$ представлен ниже. В алгоритме при условии, что $\deg b_i < \deg a_i$, базисный элемент $\bar{b}$ заменяется наименьшим по норме $\bar{b} - r\bar{a}$, где $r = \lfloor a_2/a_1 \rfloor$ — частное при делении многочленов. На выходе алгоритма получаем приведенный базис решетки $\bar{a}',\ \bar{b}' \in \Lambda_2$. Отметим, что минимальным вектором решетки будет $\bar{a}'$.

`Алгоритм 1. Reduce2BasisLattice.`
`Вход: базис решетки` $\{\bar{a};\ \bar{b}\} \subset \Lambda_2$, `где` $\bar{a} = (a_1, b_1), \bar{b} = (a_2, b_2)$.
`Выход: приведенный базис решетки` $\{\bar{a}';\ \bar{b}'\} \subset \Lambda_2$.
1: `if` $\mu(\bar{a}) > \mu(\bar{b})$ `then`
2: $\bar{a} \leftrightarrow \bar{b}$ `end if`
3: `while` $\mu(\bar{a}) < \mu(\bar{b})$ `do`
4: $r \leftarrow \lfloor a_2/a_1 \rfloor;\ \bar{b} \leftarrow \bar{b} - r\bar{a};$
5: $\bar{a} \leftrightarrow \bar{b}$ `end while`
6: `return` $\{\bar{a};\ \bar{b}\}$

Поскольку норма $\mu(\bar{b})$ уменьшается на шаге (4) по крайней мере на единицу, то цикл (3 – 5) завершит работу за конечное число шагов.

3. n–мерная решетка Бернштейна и алгоритм приведения базиса многомерной решетки Бернштейна

В 3.1 определена n–мерная решетка Бернштейна и доказаны вспомогательные утверждения. Алгоритм приведения базиса многомерной решетки над кольцом многочленов с коэффициентами из поля рациональных функций и его представление в псевдокодах содержится в 3.2.

3.1. n–мерная решетка Бернштейна. Рассмотрим n–мерную $\mathbb{F}_{2^m}[x]$–решетку $\Lambda_n = (\mathcal{L}_n, \nu_n) \subset (\mathbb{F}_{2^m}[x])^n$, состоящую из свободного $\mathbb{F}_{2^m}[x]$–модуля $\mathcal{L}_n$ ранга n и нормы ν_n, такой, что

$$\forall \varphi = (\varphi_1, \varphi_2, \ldots, \varphi_n) \in \Lambda_n : \nu_n(\varphi) = \max\{||\varphi_i||, \varphi_i \in \mathbb{F}_{2^m}[x]\}. \qquad (2)$$

Решетку Λ_n назовем n–мерной решеткой Бернштейна.

Пусть $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n \in \Lambda_n$ — некоторый базис решетки Λ_n, где $\bar{b}_i = (b_{i1}, b_{i2}, \ldots, b_{in})$, $b_{ij} \in \mathbb{F}_{2^m}[x]$. Мерой неортогональности решетки Λ_n, порожденной базисом $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n$, назовем величину $\text{mes} = \text{mes}(\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n) = (\prod_{i=1}^n \nu_n(\bar{b}_i))/||\det B||$, где $B = (b_{ij})$, $i, j = 1 \ldots n$. Доказательство следующей леммы вытекает из определения $\det B$ и леммы 1.

Лемма 3. *Для любого базиса* $mes(\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n) \geq 1$.

Лемма 4. *Если* $\bar{x} = \sum_{i=1}^n r_i \bar{b}_i \in \Lambda_n$, *то* $\nu_n(r_i \bar{b}_i) \leq mes \cdot \nu_n(x)$, $1 \leq i \leq n$.

Доказательство. Пусть B — матрица, строками которой являются базисные векторы $\bar{b}_i$, $i \in [1, n]$. Тогда $B^{-1} = A^{\tau}/\det B$, где A^{τ} — транспонированная матрица алгебраических дополнений. Рассмотрим i-й столбец $\bar{b}_i^{-1}$ матрицы B^{-1} как вектор. Тогда $\nu_n(\bar{b}^{-1}) \leq (\prod_{j=1}^n \nu_n(\bar{b}_j))/(\nu_n(\bar{b}_i)||detB||) = \text{mes}/\nu_n(\bar{b}_i)$. По условию $\bar{r} = B^{-1}\bar{x}$. Поэтому $||r_i|| \leq \nu_n(\bar{b}_i^{-1})\nu_n(\bar{x})$ для всех $i \in [1, n]$. Тогда $\nu_n(r_i\bar{b}_i) \leq \nu_n(\bar{b}_i^{-1})\nu_n(\bar{x})\nu_n(\bar{b}_i) = \text{mes} \cdot \nu_n(\bar{x})$. ■

Лемма 5. *Пусть* $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n$ — *базис решетки* Λ_n *такой, что* $mes(\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n) = 1$ *и* $\nu_n(\bar{b}_i) \leq \nu_n(\bar{b}_j), 1 \leq i < j \leq n$. *Тогда* $\nu_n(\bar{b}_j)$, $1 \leq j \leq n$ — j*-й последовательный минимум решетки* Λ_n *и, в частности,* $\nu_n(\bar{b}_1) \leq \nu_n(\bar{x})$ *для всех* $\bar{x} \in \Lambda_n, \bar{x} \neq 0$.

Доказательство. Пусть $\lambda_j = \nu_n(\bar{x})$, где $\bar{x} = \sum_{i=1}^n r_i \bar{b}_i$, — j-й последовательный минимум решетки Λ_n для некоторого $j \in [1, n]$. Достаточно показать, что $\nu_n(\bar{x}) \geq \nu_n(\bar{b}_j)$. Перенумеруем базис $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n$, не нарушая условие леммы, так, чтобы для некоторого $i_0 \in \{j, j+1, \ldots, n\}$ выполнялось $r_{i_0} \neq 0$. Номер i_0 существует, так как $\nu_n(\bar{x})$ — j-й последовательный минимум. Тогда по лемме 4 $\nu_n(\bar{x}) \geq \nu_n(r_{i_0}\bar{b}_{i_0}) \geq \nu_n(\bar{b}_j)$. ■

Базис $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n$ называется приведенным по Ленстре, если после его перестановки $\bar{b}_1', \bar{b}_2', \ldots, \bar{b}_n'$ выполняется:

$$\nu_n(\bar{b}_i') \leq \nu_n(\bar{b}_j') \qquad 1 \leq i < j \leq n, \tag{3}$$

$$\|b_{ii}'\| \geq \|b_{ij}'\| \qquad 1 \leq i < j \leq n, \tag{4}$$

$$\|b_{ii}'\| > \|b_{ij}'\| \qquad 1 \leq j < i \leq n. \tag{5}$$

Отметим, что $\nu_n(\bar{b}_i) = \nu_n(\bar{b}_i'), 1 \leq i \leq n$. Если для базиса $\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n$ выполняются условия (4) – (5), тогда элементы матрицы B обладают следующим свойством:

$$B = \begin{pmatrix} = \nu_n(\bar{b}_1) & \leq \nu_n(\bar{b}_1) & \leq \nu_n(\bar{b}_1) & \ldots & \leq \nu_n(\bar{b}_1) \\ < \nu_n(\bar{b}_2) & = \nu_n(\bar{b}_2) & \leq \nu_n(\bar{b}_2) & \ldots & \leq \nu_n(\bar{b}_2) \\ < \nu_n(\bar{b}_3) & < \nu_n(\bar{b}_3) & = \nu_n(\bar{b}_3) & \ldots & \leq \nu_n(\bar{b}_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ < \nu_n(\bar{b}_n) & < \nu_n(\bar{b}_n) & < \nu_n(\bar{b}_n) & \ldots & = \nu_n(\bar{b}_n) \end{pmatrix}.$$

В этом случае $\|\det(B)\| = \prod_{i=1}^{n} \nu_n(\bar{b}_i)$, поэтому $\mathrm{mes}(\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_n) = 1$. Согласно (3) и леммы 5, $\nu_n(\bar{b}_j)$ — j-й последовательный минимум решетки Λ, а $\bar{b}_1$ — вектор с минимальной нормой ν_n.

3.2. Алгоритм приведения базиса n–мерной решетки Бернштейна. Пусть $\nu_n(\bar{b}_0) = 0$ и для некоторого $k \in \{0, 1, \ldots, n\}$ выполняется:

$$\nu_n(\bar{b}_i) \le \nu_n(\bar{b}_j) \qquad 1 \le i < j \le k, \tag{6}$$

$$\nu_n(\bar{b}_k) \le \nu_n(\bar{b}_j) \qquad k < j \le n, \tag{7}$$

$$\|b_{ii}\| \ge \|b_{ij}\| \quad 1 \le i \le k, i < j \le n, \tag{8}$$

$$\|b_{ii}\| > \|b_{ij}\| \qquad 1 \le j < i \le k. \tag{9}$$

Отметим, что для $k = 0$ условия (6) – (9) выполняются. Если $k = n$, тогда базис является приведенным по Ленстре и алгоритм останавливается. Пусть $k < n$. Перенумеруем базисные векторы $\bar{b}_{k+1}, \ldots, \bar{b}_n$ так, что $\nu_n(\bar{b}_{k+1}) = \min\{\nu_n(\bar{b}_i) : k+1 \le i \le n\}$. Пусть $a_{ij} \in \mathbb{F}_{2^m}$ — коэффициент многочлена b_{ij} при степени $x^{\log \nu_n(\bar{b}_i)}$, $1 \le i \le k+1,\ 1 \le j \le k$. Рассмотрим

$$\sum_{i=1}^{k} a_{ij} r_i = a_{k+1,j}, \quad 1 \le j \le k. \tag{10}$$

Из условий (8) – (9) следует, что $a_{ii} \ne 0$ при $1 \le i \le k$ и $a_{ij} = 0$ при $1 \le j < i \le k$. Значит, матрица коэффициентов $A = \{a_{ij}\}$ — верхнетреугольная ранга k, поэтому система уравнений (10) имеет единственное решение $(r_1, r_2, \ldots, r_k), r_i \in \mathbb{F}_{2^m}$.

Пусть $\bar{b}'_{k+1} = \bar{b}_{k+1} - \sum_{i=1}^{k} r_i \bar{b}_i x^{\log \nu_n(\bar{b}_{k+1}) - \log \nu_n(\bar{b}_i)}$, тогда $\nu_n(\bar{b}'_{k+1}) \le \le \nu_n(\bar{b}_{k+1})$ и $\bar{b}^*_{k+1} \in \mathbb{F}_{2^m}[x]^n$, так как выполнены условия (6) и (7). Более того, из (10) следует, что $\|b_{k+1,i}\| < \nu_n(\bar{b}_{k+1})$, при $1 \le i \le k$. Если $\nu_n(\bar{b}'_{k+1}) = \nu_n(\bar{b}_{k+1})$, то меняем $\bar{b}_{k+1}$ на $\bar{b}'_{k+1}$. Сделаем перестановку координат $\bar{b}_1, \ldots, \bar{b}_n$ так, что $\|b_{k+1,k+1}\| = \nu_n(\bar{b}_{k+1})$. Отметим, что перестановка никак не повлияет на первые k координат. Увеличиваем k на единицу.

Если $\nu_n(\bar{b}'_{k+1}) < \nu_n(\bar{b}_{k+1})$, то меняем местами $\bar{b}_{k+1}$ и $\bar{b}'_{k+1}$. Затем присваиваем k наибольшее значение $l \in \{0, 1, \ldots, k\}$ такое, что $\nu_n(\bar{b}_l) \le \nu_n(\bar{b}_{k+1})$. После всех преобразований условия (6) – (9) будут выполнены для l. Продолжим процесс и на выходе алгоритма получим базис, приведенный по Ленстре. Отметим, что в алгоритме происходит перестановка координат исходного базиса, поэтому необходимо хранить соответствующую перестановку, чтобы восстановить порядок следования координат в исходном базисе. Ниже представлена подробная схема алгоритма.

`Алгоритм 2. ReduceNBasisLattice.`

`Вход: матрица` $B = (-\bar{b}_i-)$, `где` $\{\bar{b}_1, \ldots, \bar{b}_n\}$ — `базис решетки` Λ.

`Выход: приведенный по Ленстре базис решетки` $\{\bar{b}_1, \ldots, \bar{b}_n\} \subset \Lambda$.

`Комментарий:` $P = (p_{ij}), i, j = 1, \ldots, n$ — `матрица перестановок,` p_i — `строка матрицы` P.

1: $k \leftarrow 0,\ \{a_{ij}\} \leftarrow 0,\ P \leftarrow E$
2: **while** $k < n$ **do**
3: $\min \leftarrow k+1$
4: **for** $i = k+1$ to n **do**
5: **if** $\nu_n(\bar{b}_i) < \nu_n(\bar{b}_{\min})$ **then**
6: $\bar{b}_{\min} \leftarrow \bar{b}_i, \min \leftarrow i$
7: **end if end for**
8: $\bar{b}_{k+1} \leftarrow \bar{b}_{\min}$
9: $a_{k+1,k+1} \leftarrow b_{k+1,k+1}[\log \nu_n(\bar{b}_{k+1})]$
10: **for** $i = 1$ to $k+1$ **do** $a_{ii} \leftarrow b_{ii}[\log \nu_n(\bar{b}_i)]$
11: **for** $j = i$ to k **do** $a_{ij} \leftarrow b_{ij}[\log \nu_n(\bar{b}_i)]$
12: **end for end for**
13: **BackGauss** (**in:** $k, \{a_{ij}\}$; **out:** $\bar{r}$)
14: $\bar{b}'_{k+1} = \bar{b}_{k+1} - \sum_{i=1}^{k} r_i \bar{b}_i x^{\log \nu_n(\bar{b}_{k+1}) - \log \nu_n(\bar{b}_i)}$
15: **if** $\nu_n(\bar{b}'_{k+1}) = \nu_n(\bar{b}_{k+1})$ **then**
16: $\bar{b}_{k+1} \leftarrow \bar{b}'_{k+1}$
17: **for** $j = k+1$ to n **do**
18: **if** $\|b_{jj}\| = \nu_n(\bar{b}_j)$ **then**
19: **for** $i = 1$ to n **do** $b_{i,k+1} \leftrightarrow b_{ij}$ **end for**
20: $p_{k+1} \leftrightarrow p_j$ **end if end for**
21: $k \leftarrow k+1$ **end if**
22: **if** $\nu_n(\bar{b}'_{k+1}) < \nu_n(\bar{b}_{k+1})$ **then**
23: $\bar{b}_{k+1} \leftarrow \bar{b}'_{k+1}$
24: **for** $l = 0$ to k **do**
25: **if** $\nu_n(\bar{b}_l) \le \nu_n(\bar{b}_{k+1})$ **then** $\max \leftarrow l$
26: **end if end for**
27: $k \leftarrow \max$
28: **end if end while**
29: $B \cdot P^{\intercal}$
30: **return** $\{\bar{b}_1;\ \bar{b}_2; \ldots; \bar{b}_n\}$

Подпрограмма: BackGauss
Вход: параметр $k \le n$, матрица $\{a_{ij}\}, 1 \le i \le j \le n$
Выход: вектор $\bar{r} = (r_1, r_2, \ldots, r_k)$

1: $r_k \leftarrow \frac{1}{a_{kk}} a_{k+1,k}$
2: **for** $i = k$ **downto** 1 **do**
3: $\text{sum} \leftarrow 0$
4: **for** $j = i+1$ to k **do** $\text{sum} \leftarrow \text{sum} + a_{ij} r_j$ **end for**
5: $r_i \leftarrow \frac{1}{a_{ii}}(\text{sum} + a_{k+1,i})$ **end for**
6: **return** $(r_1, r_2, \ldots, r_k)$

4. Математическая модель списочного декодера Бернштейна

В 4.1 представлена общая информация о бинарных кодах Гоппы. В 4.2 построена математическая модель списочного декодера Бернштейна с учетом алгоритмов 1 и 2 из разделов 2 и 3 соответственно.

4.1. Бинарные коды Гоппы. Пусть $g(x) \in \mathbb{F}_{2^m}[x]$ — многочлен с коэффициентами из поля $\mathbb{F}_{2^m}$ степени t. Рассмотрим множество элементов поля $L = \{\alpha_1,\ \alpha_2, \ldots, \alpha_n\}$ $(\subseteq\ \mathbb{F}_{2^m})$, $0 < n \leq 2^m$ такое, что все элементы в L различные и $g(\alpha_i) \neq 0$ для всех $i \in [1,\ n]$. Множество L называют носителем и обычно в качестве L выбирают все поле $\mathbb{F}_{2^m}$. Свяжем с каждым вектором $\bar{v} = (v_1,\ v_2, \ldots,\ v_n) \in \mathbb{F}_2^n$ рациональную функцию $R_{\bar{v}}(x) = \sum_{i=1}^{n} v_i (x - \alpha_i)^{-1} \in \mathbb{F}_{2^m}(x)$. Говорят, что вектор $\bar{v}$ принадлежит бинарному коду Гоппы $\bar{v} \in \Gamma(L, g)$, если $R_{\bar{v}}(x) \equiv 0 \pmod{g(x)}$ [1].

Код Гоппы является линейным кодом длины $n = |L|$, размерности $k \geq n - mt$ и с минимальным кодовым расстоянием $d \geq 2t + 1$ [1].

4.2. Списочный декодер Паттерсона — Бернштейна. Пусть $\Gamma(\alpha_1, \ldots, \alpha_n, g), \alpha_i \in \mathbb{F}_{2^m}$ бинарный код Гоппы длины n с порождающим многочленом $g(x)$ степени t и носителем $L = \{\alpha_1, \ldots, \alpha_n\}$. Стандартный декодер Паттерсона [8] исправляет до t ошибок. Опишем списочный декодер Бернштейна согласно [2], позволяющий исправлять дополнительно u ошибок, где величина u зависит от параметров кода и равна $t^2/(2n)$. Пусть $\bar{w} = (w_1, \ldots, w_n) \in \mathbb{F}_2^n$ — пришедшее по каналу слово и $h(x) = \prod_{i=1}^{n}(x - \alpha_i)$. Определим многочлен локаторов ошибок $\sigma_{\bar{e}}(x) = \prod_{i=1}^{n}(x - \alpha_i)^{e_i}$, где $\bar{e} = (e_1, \ldots, e_n) \in \mathbb{F}_2^n$ — вектор ошибок веса $t + u$. Тогда $\sigma'_{\bar{e}}(x) \equiv \sigma_{\bar{e}}(x) R_{\bar{w}}(x) \pmod{g(x)}$. По определению $\sigma'_{\bar{e}}(x)/\sigma_{\bar{e}}(x) \equiv 0 \pmod{g(x)}$. Значит $\sigma'_{\bar{e}} \equiv 0 \pmod{g(x)}$. Поскольку $\sigma_{\bar{e}}(x)$ — многочлен над полем характеристики 2, то представим его в следующем виде $\sigma_{\bar{e}}(x) = a(x)^2 + x b(x)^2$ для некоторых $a(x), b(x), \deg a(x) \leq \lfloor (t+u)/2 \rfloor, \deg b(x) \leq \lfloor (t+u-1)/2 \rfloor$. Тогда $\sigma'_{\bar{e}}(x) = b(x)^2 = g(x) f(x)$, для любого $f(x) \in \mathbb{F}_{2^m}[x]$. Положим $f(x) = g(x)$, тогда $\sigma'_{\bar{e}}(x) \equiv 0 \pmod{g(x)^2}$. То есть, $b(x)^2 = \sigma'_{\bar{e}}(x) = \sigma_{\bar{e}}(x) R_{\bar{w}}(x) = (a(x)^2 + x b(x)^2) R_{\bar{w}} \pmod{g(x)^2}$, где $a(x) = b(x)\sqrt{R_{\bar{w}}(x)^{-1} - x} \pmod{g(x)}$. Последнее уравнение будем решать с помощью быстрого алгоритма `Reduce2BasisLattice` из раздела 2. Пусть $\bar{a} = (s, 1), \bar{b} = (g, 0)$, где $s(x) = \sqrt{R_{\bar{w}}(x)^{-1} - x} \pmod{g(x)}$ — базис решетки $\Lambda_2 \subseteq (\mathbb{F}_{2^m}[x])^2$ с нормой μ_2. Отметим, что квадратный корень извлекается, поскольку все вычисления в поле $\mathbb{F}_{2^m}$. Применим к базису решетки Λ_2 алгоритм `Reduce2BasisLattice`. На выходе алгоритма получим пару базисных векторов $\bar{a'} = (\alpha_0, \beta_0), \bar{b'} = (\alpha_1, \beta_1)$ решетки Λ_2 с минимальной нормой. Вычислим $\sigma_0 = \alpha_0^2 + x\beta_0^2$ и $\sigma_1 = \alpha_1^2 + x\beta_1^2$, $t_0 = \deg \sigma_0,\ g_0 = 2\lfloor (u + t - t_0)/2 \rfloor,\ g_1 = 2\lfloor (u + t_0 - t - 1)/2 \rfloor$ и $\theta = g_1 - g_0$.

Найдем многочлен $\delta \in \mathbb{F}_{2^m}[x]$ такой, что $\sigma_1\delta \bmod h = \sigma_0$ с помощью алгоритма Евклида `EuclidEx`. Последнее уравнение имеет решение, если НОД$(\sigma_1,\ h) = 1$. Если НОД$(\sigma_1,\ h) \neq 1$, то $(\alpha_1, \beta_1) = (\alpha_1, \beta_1) + \sqrt{d}(\alpha_0,\ \beta_0)$, $\sigma_1 = \sigma_1 + d\sigma_0$, для некоторого $d \in \mathbb{F}_{2^m}$. Зафиксируем целое число $r > 0$. Вычислим $l = r\sqrt{n/(g_0 + g_1)}$. Значение параметра r выбирается близкое к $\sqrt{2(u-1)n}$ так, чтобы $(g_0 + g_1)(l-1)/2k + n(k+1)/2l < t$.

Для заданных l и r определим l–мерную решетку $\Lambda(x, z) \subset \mathbb{F}_{2^m}(x)[z]$, порожденную базисными многочленами $1, (x^\theta z + \delta(x))/h(x),\ (x^\theta z + \delta(x))^2/h^2(x), \ldots, (x^\theta z + \delta(x))^r/h^r(x),\ x^\theta z(x^\theta z + \delta(x))^r/h^r(x), (x^\theta z)^2(x^\theta z + \delta(x))^r/h^r(x), \ldots, (x^\theta z)^{l-r-1}(x^\theta z + \delta(x))^r/h^r(x)$. Запишем элементы базиса решетки по возрастанию степени z, то есть в виде $\varphi_i(x, z) = \varphi_0(x) + \varphi_1(x)z + \varphi_2(x)z^2 + \ldots + \varphi_{l-1}(x)z^{l-1},\ \varphi_i(x) \in \mathbb{F}_{2^m}(x),\ i = 0, \ldots, l-1$. Поставим в соответствие каждому такому элементу $\varphi_i(x, z)$ l–мерный вектор $\overline{\varphi_i(x)} = (\varphi_0(x), \ldots, \varphi_{l-1}(x))$. Получим решетку $\Lambda_l(x) \subset (\mathbb{F}_{2^m}(x))^l$ с базисом $\overline{\varphi_0(x)}, \ldots, \overline{\varphi_{l-1}(x)}$. Применим к последней решетке алгоритм `ReduceNBasisLattice` из раздела 3 и найдем минимальный по норме $v_n(\overline{\varphi})$ вектор $\overline{\psi(x)} = (\psi_0(x), \ldots, \psi_{l-1}(x)) \in \Lambda_l(x)$.

Представим полученный вектор $\overline{\psi(x)}$ как многочлен двух переменных $\psi(x, z)$. Применим алгоритм факторизации `PolyFactoriz` к многочлену $\psi(x, z)$ по переменной z. Найдем корни вида $q_0^2/(x^\theta q_1^2)$, где НОД$(q_0,\ q_1) = 1$ и $q_0, q_1 \in \mathbb{F}_{2^m}[x]$. Для всех таких корней вычислим $\sigma = q_0^2\sigma_0 + q_1^2\sigma_1$. Если σ делит h, то выводим $\bar{c} = (c_1, c_2, \ldots, c_n) \in \Gamma(L, g)$ такое, что $c_i - w_i \pmod 2 = 1$, где $\bar{w} = (w_1, w_2, \ldots, w_n)$ — пришедшее по каналу слово, а i такое, что $\sigma(\alpha_i) = 0$.

Ниже представлена подробная схема алгоритма списочного декодирования бинарного кода Гоппы.

`Алгоритм 3. ListDecoderPatBern.`

`Вход:` $\Gamma(\alpha_1, \ldots, \alpha_n, g)$ — `код Гоппы длины` n, $\deg g(x) = t$;
$\bar{w} \in \mathbb{F}_2^n$ — `пришедшее по каналу слово.`

`Выход: вектор ошибки` $\bar{e} \in \mathbb{F}_2^n$.

1: `if` $R_{\bar{w}} \pmod{g(x)} = 0$ `then`
2: $\bar{e} \leftarrow \bar{w}$, `return` $\bar{e}$ `end if`
3: $\bar{c} \leftarrow \bar{w},\ h(x) \leftarrow \prod_{i=1}^n (x - \alpha_i)$
4: $s(x) \leftarrow \sqrt{R_{\bar{w}}(x)^{-1} - x} \pmod{g(x)}$
5: $\bar{a} \leftarrow (s(x), 1), \bar{b} \leftarrow (g(x), 0)$
6: `Reduce2BasisLattice(in:`$\bar{a}, \bar{b}$`; out:` $\bar{a}', \bar{b}'$`)`
7: $\sigma_0 \leftarrow \alpha_0^2 + x\beta_0^2, \sigma_1 \leftarrow \alpha_1^2 + x\beta_1^2,\ t_0 \leftarrow \deg\sigma_0$
8: $g_0 \leftarrow 2\lfloor(u + t - t_0)/2\rfloor,\ g_1 \leftarrow 2\lfloor(u + t_0 - t - 1)/2\rfloor,\ \theta \leftarrow g_1 - g_0$
9: `while НОД(`$\sigma_1,\ h$`)` $\neq 1$ `do`
10: $d \leftarrow$ `NextFieldEl // генерирует случайный элемент поля` $\mathbb{F}_{2^m}$
11: $(\alpha_1, \beta_1) \leftarrow (\alpha_1, \beta_1) + \sqrt{d}(\alpha_0,\ \beta_0)$
12: $\sigma_1 \leftarrow \sigma_1 + d\sigma_0$ `end while`
13: `EuclidEx (in:` $\sigma_0(x), \sigma_1(x), h(x)$`; out:` $\delta(x)$`)`

14: ChooseLatticeParameters (in: t, g_0, g_1; out: r, l) // определяет параметры решетки r, l
15: LatticeBasisInit (in: $\theta, r, l, \delta(x), h(x)$; out: $\overline{\varphi_0(x)}, \ldots, \overline{\varphi_{l-1}(x)}$) // формирует базис решетки для алгоритма 2
16: ReduceNBasisLattice (in: $\overline{\varphi_0} \ldots \overline{\varphi_{l-1}}$; out: $\overline{\psi_0} \ldots \overline{\psi_{l-1}}$)
17: PolyFactoriz (in: $\psi(x, z)$; out: $Z = \{\zeta(x) : \psi(x, \zeta(x)) = 0\}$)
18: while $z_1(x)/z_2(x)$ in Z do
19: $q_0(x)/q_1(x) \leftarrow \sqrt{x^\theta z_1(x)/z_2(x)}$
20: if НОД$(q_0(x),\ q_1(x)) = 1$ then
21: $\overline{q_j(x)} \leftarrow (q_0(x),\ q_1(x))$
22: $j \leftarrow j + 1$ end if end while
23: while $i \le j$ do
24: $\sigma_i(x) \leftarrow q_0^2(x)\sigma_0(x) + q_1^2(x)\sigma_1(x)$
25: $d(x) \leftarrow$ НОД$(\sigma_i(x),\ h(x))$
26: if $d \ne 1$ then
27: ChienSearch (in: $d, \alpha_1, \ldots, \alpha_n$; out: $I = \{i : d(\alpha_i) = 0\}$)
28: while i in I do $\bar{c}_i \leftarrow \bar{w}_i + 1 \pmod 2$ end while end if
29: end while

Используя [2], можно показать, что алгоритм 3 работает корректно и позволяет гарантированно исправлять $\lfloor n - \sqrt{n(n - 2t - 2)} \rfloor$ ошибок.

Список литературы

1. *Гоппа В. Д.* Новый класс линейных корректирующих кодов // Проблемы передачи информации. 1970. Т. 6, вып. 3. С. 24–30.
2. *Bernstein D. J.* List decoding for binary Goppa codes. 2008. URL: http://cr.yp.to/papers.html#goppalist (дата обращения: 01.10.2012).
3. *Bernstein D. J., Lange T., Peters C.* Attacking and defending the McEliece cryptosystem // Lecture Notes In Computer Science. 2008. Vol. 5299. P. 31–46.
4. *Engelbert D., Overbeck R., Schmidt A.* A summary of McEliece-type cryptosystems and their security // Journal of Mathematical Cryptology. 2007. Vol. 1(2). P. 151–199.
5. *Lenstra A. K.* Factoring multivariate polynomials over finite fields // J. Comput. System Sci. 1985. Vol. 30(2). P. 235–248.
6. *Lenstra H. W.* Lattices. Algorithmic number theory: lattices, number fields, curves and cryptography. Cambridge : Cambridge Univ. Press, 2008. P. 127–181.
7. *McEliece R. J.* A public key cryptosystem based on algebraic coding theory // DSN progress report. 42–44. 1978. P. 114–116.
8. *Patterson N. J.* The algebraic decoding of Goppa codes // IEEE Transactionson Information Theory. 1975. Vol. 21(2). P. 203–207.
9. *Wieschebrink C.* Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes // PQCrypto. 2010. P. 61–72.

Поступила в редакцию 26.11.2012.