В. М. Сидельников, Открытое шифрование на основе двоичных кодов Рида–Маллера, *Дискрет. матем.*, 1994, том 6, выпуск 2, 3–20

Дискретная математика

том 6 выпуск 2 * 1994

УДК 519.72

Открытое шифрование на основе двоичных кодов Рида–Маллера

© 1994 г. В. М. Сидельников

Рассматривается кодовая система открытого шифрования (см. [1, 2]), в которой используется низкоскоростной код Рида–Маллера порядка r (код RM_r) длины $N = 2^m$ и алгоритм декодирования работы [3]. Предложена модификация этой системы, которая существенно повышает ее стойкость к нападению.

Основная часть работы посвящена исследованиям сложности дешифрования как исходной (с кодом RM_r) так и модифицированной системы шифрования. Основной вывод состоит в том, что рассмотренные кодовые системы, особенно модифицированная, имеют при $N \geqslant 1024$ высокую стойкость к нападению, скорость передачи близкую к 1 и невысокую сложность как шифрования так и расшифрования.

1. Описание системы открытого шифрования

Двоичный код Рида–Маллера r-го порядка (код RM_r) длины $N = 2^m$ образован векторами вида $\Omega_f = (f(\alpha_1), \ldots, f(\alpha_N))$, где $f(x)$ — булева функция от m переменных, порядок нелинейности которой не превосходит r, $\{\alpha_1, \ldots, \alpha_N\} = (\mathbf{F}_2)^m$ — множество всех двоичных векторов длины m, которое является линейным пространством размерности m над полем $\mathbf{F}_2$ из двух элементов. Число информационных разрядов (размерность) кода RM_r над $\mathbf{F}_2$ равна $k(r) = \sum_{j=0}^{r} \binom{m}{j}$, а кодовое расстояние $d = 2^{m-r}$, т. е. код исправляет любую комбинацию из $2^{m-r-1} - 1$ ошибок [4]. В работе [3] предложены эффективные алгоритмы декодирования кода RM_r, которые исправляют "почти все" ошибки кратности до $t(N, m) = N/2 - O(m^r N^{1/2})$, т. е. существенно больше, чем $d/2$. Сложность этих алгоритмов равна $O(m^{r-1} N^2)$ операций в поле $\mathbf{F}_2$.

Например, алгоритм декодирования из [3] кода RM_3 длины $N = 1024$ и $N = 2048$ ($m = 10, 11$) и размерности 176 и 232, как показывают эксперименты, "почти всегда" исправляет 200 и 420 ошибок.

Пусть $R \rightarrow$ фиксированная порождающая матрица размера $k(r) \times N$ кода RM_r длины N. Обозначим через $\mathcal{E}_r$ ансамбль, состоящий из всевозможных матриц вида $E = H \cdot R \cdot \Gamma$, где H пробегает множество всех невырожденных матриц над $\mathbf{F}_2$ размера $k(r) \times k(r)$, а Γ — множество всех перестановочных матриц размера $N \times N$, т. е. матриц с элементами из $\mathbf{F}_2$, у которых в каждой строке и каждом

столбце имеется только один ненулевой элемент. Вычислим число элементов в ансамбле $\mathcal{E}_r$.

Определим группу автоморфизмов G_r кода RM_r как множество перестановочных матриц Γ, для которых $R \cdot \Gamma = H \cdot R$, где H — матрица над $\mathbf{F}_2$ размера $k \times k$. Очевидно,

$$|\mathcal{E}_r| = h_k N! \, |G_r|^{-1}, \qquad k = k(r),$$

где h_k — число всех невырожденных двоичных матриц размера $k \times k$, $k = k(r)$, $(h_k = (2^k - 1)(2^k - 2)\dots(2^k - 2^{k-1}))$, $N!$ — число различных матриц Γ и $|G_r|$ — мощность группы G_r автоморфизмов кода. Как известно [4], группа G_r представляет собой полную аффинную группу пространства $(\mathbf{F}_2)^m$ и, следовательно, $|G_r| = 2^m(2^m - 1)\dots(2^m - 2^{m-1})$. Таким образом,

$$|\mathcal{E}_r| = (N!)(2^k - 1)(2^k - 2)\dots(2^k - 2^{k-1})/2^m(2^m - 1)\dots(2^m - 2^{m-1}).$$

Отметим, что ансамбль кодов $|\mathcal{K}_r| = \{\mathcal{K}(E);\ E \in \mathcal{E}_r\}$, где $\mathcal{K}(E)$ — линейный код над $\mathbf{F}_2$ с порождающей матрицей E, содержит

$$|\mathcal{K}_r| = (N!)/2^m(2^m - 1)\dots(2^m - 2^{m-1})$$

элементов, ибо коды с порождающими матрицами $R \cdot \Gamma$ и $H \cdot R \cdot \Gamma$ совпадают.

Передача секретного сообщения абонента $\mathcal{Y}$, предназначенного абоненту $\mathcal{X}$, предваряется следующими действиями. Абонент $\mathcal{X}$ случайно, равновероятно и независимо от других абонентов выбирает матрицы $H = H_{\mathcal{X}}$ и $\Gamma = \Gamma_{\mathcal{X}}$ и вычисляет матрицу $E_{\mathcal{X}} = H_{\mathcal{X}} \cdot R \cdot \Gamma_{\mathcal{X}}$ из ансамбля $\mathcal{E}_r$. Матрица $E_{\mathcal{X}}$ является открытым (общедоступным для всех абонентов) ключом, а матрицы $H_{\mathcal{X}}$, $\Gamma_{\mathcal{X}}$ — секретным ключом абонента $\mathcal{X}$.

Шифрованная информация $\mathbf{b}$ (криптограмма), которую абонент $\mathcal{Y}$ передает по общедоступному каналу абоненту $\mathcal{X}$, в системе Маклиса [1] представляет собой двоичный вектор длины N вида $\mathbf{b} = \mathbf{a}E + \mathbf{e}E = E_{\mathcal{X}}$, где $\mathbf{a}$ — двоичный вектор длины $k = k(r)$, несущий конфиденциальную информацию абонента $\mathcal{Y}$, а $\mathbf{e}$ — секретный вектор ошибок веса, не превосходящего t, который случайно и равновероятно выбирается абонентом $\mathcal{Y}$ среди всех векторов веса не выше t.

Абонент $\mathcal{X}$, получив вектор $\mathbf{b}$, восстанавливает кодовый вектор $\mathbf{a}E$ следующим образом. Сначала он строит вектор $\mathbf{b}' = \mathbf{b}\Gamma^{-1}$, который, очевидно, является вектором кода RM_r с порождающей матрицей R, искаженным не более, чем в t разрядах. Затем с помощью какого-либо алгоритма декодирования кода RM_r находится вектор $\mathbf{a}'$, который удовлетворяет условию $\mathbf{b}' = \mathbf{a}'R + \mathbf{e}'$, где $w(\mathbf{e}') \leqslant t$ и $\mathbf{a}'R$ — кодовый вектор кода RM_r.

Мы будем предполагать, что $t > (d-1)/2$, но t меньше некоторой границы, при которой алгоритм декодирования работает "почти всегда" правильно. Именно для этого случая в [3] предложен алгоритм декодирования. Как показано в [3], для вероятности $P(r, N)$ неправильного декодирования кода RM_r длины N в случае $t = (1 - \varepsilon)N/2$, $\varepsilon > 0$, $r = \mathrm{const}$, $N \to \infty$, справедлива оценка $P(r, N) < \exp(-cN)$, $c = c(\varepsilon, r) > 0$. Поэтому при подходящем t декодирование будет неправильным с пренебрежимо малой вероятностью. Этой возможностью мы будем пренебрегать, т. е. полагать, что всегда $\mathbf{a}' = \mathbf{a}$.

Шифрованная информация $\mathbf{c}$ в системе Нидеррайтера [2] представляет собой двоичный вектор длины $N-k$ вида $\mathbf{c} = \mathbf{e}D$, где $D = D_{\mathcal{X}}$ — некоторая проверочная

матрица кода $\mathcal{K}(E_X)$ размера $N \times (N-k)$, а $\mathbf{e}$ — вектор длины N и веса, не превосходящего t, который несет конфиденциальную информацию абонента $\mathcal{Y}$.

В теории кодирования вектор $\mathbf{c}$ называют синдромом вектора $\mathbf{e}$. Отметим, что матрицы D и E связаны соотношением $E \cdot D^T = 0$, где D^T — транспонированная матрица D. Строки матрицы D являются базисом подпространства размерности $N-k$, ортогонального пространству строк матрицы E.

Абонент $\mathcal{X}$, получив сообщение $\mathbf{c}$, находит какой-либо вектор $\mathbf{b}$, который является решением уравнения $\mathbf{x}D^T = \mathbf{c}$. Очевидно, вектор $\mathbf{b}$ является вектором вида $\mathbf{b} = \mathbf{a}E + \mathbf{e}$ при некотором неизвестном $\mathbf{a}$. Затем абонент $\mathcal{X}$ также, как в системе Маклиса, декодирует вектор $\mathbf{b}\Gamma^{-1} = \mathbf{b}' = \mathbf{a}'R + \mathbf{e}'$, но вместо кодового вектора $\mathbf{a}'R$ находит вектор $\mathbf{e}'$, а затем и вектор $\mathbf{e} = \mathbf{e}'\Gamma$.

Как и выше, предполагаем, что используемый алгоритм декодирования кода RM_r всегда правильно восстанавливает вектор ошибок $\mathbf{e}$.

Системы Маклиса и Нидеррайтера обладают одинаковой стойкостью к нападению, ибо криптографическая атака на одну из систем может быть легко трансформирована в атаку на другую.

Действительно, при известном синдроме $\mathbf{c} = \mathbf{e}D$ нетрудно вычислить вектор $\mathbf{b} = \mathbf{a}E + \mathbf{e}$ с некоторым вектором $\mathbf{a}$ такой, что $\mathbf{c} = \mathbf{b}D$. Вектор $\mathbf{b}$ мы будем рассматривать как криптограмму в системе Маклиса. Если для системы Маклиса найдена криптографическая атака со сложностью T, т. е. известен алгоритм вычисления вектора $\mathbf{a}$ (конфиденциальная информация в системе Маклиса), то вектор $\mathbf{e}$ (конфиденциальная информация в системе Нидеррайтера), очевидно, представляется в виде $\mathbf{e} = \mathbf{a}E + \mathbf{b}$, т. е. сложность определения $\mathbf{e}$ по существу совпадает со сложностью определения $\mathbf{a}$.

Наоборот, если для системы Нидеррайтера известна криптографическая атака со сложностью T, то используя в качестве криптограммы этой системы вектор $(\mathbf{a}E + \mathbf{e})D^T = \mathbf{e}D^T$, вычислим вектор ошибок $\mathbf{e}$, а затем и вектор $\mathbf{a}$.

Соображения, использованные в предыдущих двух абзацах любезно сообщены автору Г. А. Кабатянским.

Две эти системы различаются скоростью передачи. У системы Нидеррайтера она всегда выше, поэтому далее будем рассматривать только ее. Вместе с тем будем предполагать, не оговаривая этого особо, что криптограммой этой системы является N-мерный вектор $\mathbf{b}$, который является каким-либо решением системы $\mathbf{x}D = \mathbf{c}$, где $\mathbf{c} = \mathbf{e}D$ и $\mathbf{e}$ — вектор веса не выше t. Это связано с тем, что алгоритм декодирования кода RM_r, рассмотренный в [3], и некоторые известные криптографические атаки оперируют с искаженным кодовым вектором $\mathbf{b}$.

Шифрование сообщения $\mathbf{e}$ состоит в вычислении его синдрома и поэтому его сложность равна $O((N-k)N)$ операций. Сложность расшифрования (сложность восстановления вектора $\mathbf{e}$) определяется, в основном, трудоемкостью алгоритма декодирования кода RM_r и при использовании алгоритма декодирования работы [3] равна $O(N^2(\log N)^{r-1})$ операций.

Как известно [5], кодовые системы открытого шифрования имеют большую скорость шифрования по сравнению с другими подобными системами, например, с системой RCA. Вместе с тем они обладают, по меньшей мере, двумя недостатками. Во-первых, скорость передачи у кодовой системы всегда меньше 1 (обычно меньше $1/2$), в то время как в системе RCA она равна 1. Во-вторых, открытый ключ (в рассматриваемой кодовой системе — матрица E) имеет объем примерно в k раз больший, чем у упомянутой системы RCA. Кроме того, ра-

бот по оценке стойкости кодовых систем известно значительно меньше, чем для системы RCA [6]. Настоящая работа, как надеется автор, позволит частично преодолеть эти недостатки.

В системе открытого шифрования Нидеррайтера в качестве открытой информации выступают векторы $\mathbf{e}$ веса t и менее. Для ее реализации необходимо иметь способ отображения множества всех двоичных векторов длины n в подмножество W_t векторов длины N веса не выше t, где $n \leqslant \tau(t,N) = \left[\log_2 \sum_{i=0}^{t} \binom{N}{i}\right]$. Два из многих возможных эффективно реализуемых отображений и их обратных приведены в §5.

При кодировании n-мерных векторов из $(\mathbf{F}_q)^n$ векторами из множества W_t скорость передачи $R = R(N,t)$ в системе Нидеррайтера равна $R = n/(N-k)$ и не превосходит $\tau(t,N)/(N-k)$, где $N-k$ — длина синдрома. При использовании кода RM_3 и одного простейшего метода кодирования из §4 имеем $R(1024,200) = 576/848 = 0{,}68$, в то время как $\tau(200,1024)/848 = 0{,}85$.

Система Нидеррайтера полностью определяется как порождающей матрицей E, так и проверочной матрицей D. Так как в работе рассматриваем только низкоскоростные коды RM_r (r фиксировано, $N \to \infty$), то открытым ключом этой системы естественно считать матрицу E, которая содержит меньше строк, чем матрица D, хотя криптограмма $\mathbf{c} = \mathbf{e}D$ реально строится с помощью матрицы D.

Переход от системы Маклиса к системе Нидеррайтера полезен не только с точки зрения повышения скорости передачи, но и, что более важно, позволяет с помощью несложной модернизации существенно усилить ее стойкость к криптографическим атакам. Анализу сложности дешифрования модернизированной системы посвящен §3. Описание модернизированной системы приведено ниже.

Рассмотрим новую систему шифрования, в которой порождающая матрица E образована из u различных матриц $E_1, \ldots, E_u$ размера $k \times N$ каждая. Матрицы E_i, $i = 1, \ldots, u$, имеют вид $E_i = H_i R$ (R — порождающая матрица кода RM_r), где матрицы $H_1, \ldots, H_u$ — невырожденные матрицы размера $k \times k$ с элементами из поля $\mathbf{F}_2$. Общедоступная матрица $E_X = E$ абонента X имеет размер $k \times uN$ и вид

$$E = \|E_1 \ldots E_u\| \cdot \Gamma, \tag{1}$$

где Γ — перестановочная матрица размера $uN \times uN$.

Ансамбль $\mathcal{E}_{u,r}$ состоит из матриц E вида (1), где матрицы $H_1, \ldots, H_u$ независимо одна от другой пробегают множество всех невырожденных матриц над $\mathbf{F}_2$ размера $k \times k$, а Γ пробегает множество всех перестановочных матриц размера $uN \times uN$. По существу, $\mathcal{E}_{u,r} = (\mathcal{E}_r)^u \cdot \mathcal{G}$, где $\mathcal{G}$ — множество всех перестановочных матриц Γ. Предполагается, что в качестве матрицы E модернизированной системы выбирается матрица, являющаяся реализацией случайной величины равномерно распределенной на множестве $\mathcal{E}_{u,r}$.

Число элементов ансамбля $\mathcal{E}_{u,r}$ вычислить не удалось. Предположительно, $\|\mathcal{E}_{u,r}\| = (uN)!(h_k)^u \|G_r\|^{-u}(u!)^{-1}$.

Матрица E используется точно так же, как описано выше для системы Нидеррайтера. В частности, информация, передаваемая в рассматриваемой системе, представляет собой вектор $\mathbf{e}$ длины uN и веса, не превосходящего t_u, а криптограмма имеет вид $\mathbf{c} = \mathbf{e}D$, где D — проверочная матрица кода $\mathcal{K}(E)$.

Величину t_u естественно взять возможно большей, так как она определяет

скорость передачи сообщений, а также сложность некоторых методов дешифрования системы. Рассмотрим задачу выбора допустимого значения t_u.

Абонент $\mathcal{X}$, получив вектор $\mathbf{c} = \mathbf{e}D$, $w(\mathbf{e}) \leqslant t_u$, строит векторы $\mathbf{b} = \mathbf{a} \cdot E + \mathbf{e}$ и $\mathbf{d} = \mathbf{b} \cdot \Gamma^{-1}$. Вектор $\mathbf{d}$, очевидно, представляет собой последовательность $(\mathbf{d}'_1, \ldots, \mathbf{d}'_u)$ из u искаженных векторов кода RM_r длины N так, что $\mathbf{d}'_i = \mathbf{d}_i + \mathbf{e}_i$, $\mathbf{d}_i = \mathbf{a} \cdot H_i \cdot R$, $i = 1, \ldots, u$, $w(\mathbf{e}_i) + \ldots + w(\mathbf{e}_u) = w(\mathbf{e}) \leqslant t_u$, и $(\mathbf{e}_1, \ldots, \mathbf{e}_u) = \mathbf{e} \cdot \Gamma^{-1}$.

Пусть $t_u = ut + u - 1$, где t — максимальное число ошибок, которое может исправить алгоритм декодирования кода RM_r длины N. Тогда $w(\mathbf{e}_i) \leqslant t$, по крайней мере, для одного значения i. Таким образом, алгоритм декодирования, примененный последовательно к каждому $\mathbf{d}'_i$, правильно восстановит, по меньшей мере, один из векторов $\mathbf{e}_i$, а именно тот, который имеет минимальный вес. Зная вектор $\mathbf{e}_i = \mathbf{d}'_i + \mathbf{a} \cdot H_i \cdot R$, можно определить и информационный вектор $\mathbf{a}$. Все остальные векторы $\mathbf{e}_1, \ldots, \mathbf{e}_u$ могут быть затем восстановлены очевидным образом: $\mathbf{e}_j = \mathbf{d}'_j + \mathbf{a} \cdot H_j \cdot R$.

Отметим, что в рассмотренной модернизированной системе скорость передачи не намного ниже скорости передачи исходной системы с ансамблем $\mathcal{E}_r$. Вместе с тем, использование ансамбля $\mathcal{E}_{u,r}$, как будет показано в §4, существенно повышает стойкость шифрования. Всюду далее полагается, что $2r \leqslant m - 2$.

2. Простейшие методы анализа системы открытого шифрования

В настоящем разделе оценивается сверху сложность дешифрования (измеряемая числом операций в поле $\mathbf{F}_2$) системы открытого шифрования с ансамблем $\mathcal{E}_r$. Под дешифрованием понимается алгоритм (называемый алгоритмом дешифрования), который позволяет при известных матрицах R и E, векторе $\mathbf{b}$, числе ошибок t найти неизвестные вектор $\mathbf{a}$ и вектор $\mathbf{e}$ веса $w(\mathbf{e}) \leqslant t$, такие, что $\mathbf{b} = \mathbf{a}E + \mathbf{e}$. Естественно рассматривать два вида алгоритмов дешифрования.

Тип 1. По известному вектору $\mathbf{b} = \mathbf{a}E + \mathbf{e}$, известному числу t и известной матрице E алгоритм находит вектор $\mathbf{e}$, $w(\mathbf{e}) \leqslant t$, без определения матриц H и Γ.

Тип 2. По известным матрицам E и R алгоритм находит невырожденную матрицу H' размера $k \times k$ с элементами из поля $\mathbf{F}_2$ и перестановочную матрицу Γ' размера $N \times N$, которые являются решением уравнения

$$E = H \cdot R \cdot \Gamma, \tag{2}$$

т. е. решает уравнение (2) относительно матриц H и Γ. Затем с помощью какого-либо алгоритма декодирования находятся вектор $\mathbf{e}$, $w(\mathbf{e}) \leqslant t$.

После решения уравнения (2) можно производить дешифрование многих сообщений $\mathbf{b}$ с малой сложностью, поэтому алгоритмы типа 2 предпочтительнее алгоритмов типа 1. Основная часть настоящей работы посвящена оценкам сложности решения уравнения (2), т. е. исследованию алгоритмов типа 2.

Переходим к обсуждению известных [6–15] алгоритмов дешифрования типа 1. С точки зрения внешнего наблюдателя, не знающего свойств ансамбля $\mathcal{E}_r$, матрица E представляет собой матрицу без видимых закономерностей или, как

иногда говорят, матрицу общего положения. Как известно [6], задача декодирования двоичного линейного кода с порождающей матрицей общего положения и скоростью передачи отличной от 0 и 1 является NP-полной, т. е. с этих позиций алгоритмы типа 1 предположительно являются достаточно сложными. Вместе с тем естественно получить явные оценки сложности этих алгоритмов при конкретных значениях N, t и r.

Отметим, что переборный алгоритм декодирования по критерию минимума расстояния кода применим к любому коду и имеет, очевидно, сложность $N2^k$. Кроме того известны и другие алгоритмы декодирования произвольного кода со сложностью меньшей, чем $N2^k$ [7–12]. Рассмотрим некоторые из них.

Достаточно эффективный алгоритм декодирования кода K с произвольной порождающей матрицей E, описанный в [8, 9, 14, 15], заключается в следующем. В линейном коде выбирается, например, с помощью случайных бросаний набор из k информационных разрядов кода. Из знаков вектора $\mathbf{b}$, находящихся в этих разрядах, формируется кодовый вектор $\widehat{\mathbf{b}}$, $\widehat{\mathbf{b}} \in K$, который сравнивается с $\mathbf{b}$. Если векторы $\mathbf{b}$ и $\widehat{\mathbf{b}}$ отличаются один от другого более, чем в t разрядах, то алгоритм декодирования переходит к следующему набору из k информационных разрядов, ибо в этом случае выбранные k информационных разрядов содержат ошибки. Декодирование считается законченным, если очередной кодовый вектор $\widehat{\mathbf{b}}$ отличается от искаженного кодового вектора $\mathbf{b}$ не более, чем в t разрядах. Результатом работы алгоритма является последний кодовый вектор $\widehat{\mathbf{b}}$ и вектор ошибок $\mathbf{e} = \mathbf{b} + \widehat{\mathbf{b}}$.

Заметим, что алгоритм заканчивает работу, если отсутствуют ошибки в очередном наборе из k информационных разрядов. Если предположить, что все ошибки равновероятны, то вероятность того, что фиксированный набор из k информационных разрядов не содержит ошибок, очевидно, равна

$$P_t(N,k) = \binom{N-k}{t}\binom{N}{t}^{-1}.$$

Среднее число операций $U(N,k,t)$, требуемых для реализации этого алгоритма, приблизительно равно

$$U(N,k,t) = (P_t(N,k))^{-1}S(N,k),$$

где $(P_t(N,k))^{-1}$ — среднее число актов декодирования до нахождения набора из k информационных разрядов, в которых отсутствуют ошибки, и $S(N,k)$ — число операций необходимых для построения кодового вектора $\widehat{\mathbf{b}}$.

Величина $S(N,k)$ примерно равна $Nk + k^3$. Следовательно

$$U(N,k,t) = \binom{N-k}{t}^{-1}\binom{N}{t}(Nk + k^3). \tag{3}$$

Описанный алгоритм допускает модернизацию (см. [14, 15]). А именно, если вектор $\widehat{\mathbf{b}}$ отличается от $\mathbf{b}$ более, чем в t разрядах, то прежде, чем перейти к выборке следующих новых k информационных разрядов, производится изменение, например, первого информационного разряда. В результате чего образуется вектор $\widehat{\mathbf{b}}_1$, который может быть построен, исходя из вектора $\widehat{\mathbf{b}}$, со сложностью примерно равной Nk, меньшей, чем $S(N,k)$. Последовательно изменяя

на противоположный каждый из k информационных разрядов, получим векторы $\widehat{\mathbf{b}}_1, \ldots, \widehat{\mathbf{b}}_k$, которые также последовательно сравниваются с $\mathbf{b}$. Декодирование заканчивается, если очередной кодовый вектор $\widehat{\mathbf{b}}_i$ отличается от искаженного кодового вектора $\mathbf{b}$ не более, чем в t разрядах. Если такого вектора не нашлось, то происходит переход к новой выборке информационных разрядов. Отметим, что для построения векторов $\widehat{\mathbf{b}}_1, \ldots, \widehat{\mathbf{b}}_k$ необходимо затратить приблизительно Nk^2+k^3 операций. Декодирование заканчивается на данной выборке из k информационных разрядов, если в ней имеется не более одного искаженного разряда. Поэтому модернизированный алгоритм имеет сложность, примерно равную

$$U_1(N,k,t) = \left(\binom{N-k}{t} + \binom{N-k}{t-1}\binom{k}{1}\right)^{-1} \binom{N}{t}(Nk^2+k^3).$$

Отметим, что $U_1(N,k,t)/U(N,k,t) = t(N+k)/(N-k-t)(N+k^2)$. В работах [14, 15] используется рабочая функция $U(N,k,t)$, которая не сильно отличается от $U_1(N,k,t)$. Поэтому далее будет рассматриваться только функция $U(N,k,t)$.

Повидимому, трудоемкость других известных алгоритмов декодирования общих линейных кодов в рассматриваемых случаях выше, чем $U(N,k,t)$.

Простейший анализ сообщения (3) показывает, что зависимость величины $U(N,k,t)$ от t весьма сильная. Поэтому увеличение параметра t, т. е. использование более эффективных алгоритмов декодирования, приводит не только к увеличению скорости передачи, но и к увеличению стойкости к нападению рассмотренным выше методом. Отметим, что $U(1024,176,200) = 1{,}5 \cdot 10^{25}$.

3. Оценка сложности решения в случае $u = 1$

В качестве порождающей матрицы R кода RM_r возьмем матрицу, у которой i-й столбец R_i состоит из значений элементарных конъюнкций $q_I(x) = x_{i_1} \ldots x_{i_s}$, $I = \{i_1, \ldots, i_s\}$, $0 < i_1 < \ldots < i_s \leqslant m$, $s = 0, \ldots, r$, в точке $\alpha_j = (\alpha_1^j, \ldots, \alpha_m^j)$, $\alpha_j \in (\mathbf{F}_2)^m = \{\alpha_1, \ldots, \alpha_N\}$. Таким образом, $R_j = (q_{I_1}(\alpha_j), \ldots, q_{I_k}(\alpha_j))^T$, $k = \sum_{j=0}^{r} \binom{m}{j}$, где $q_{I_1}(x), \ldots, q_{I_k}(x)$ — всевозможные конъюнкции степени не выше r, занумерованные в каком-либо порядке.

Строки матрицы R занумерованы элементарными конъюнкциями. Таким же образом будем нумеровать столбцы матрицы H, а именно, наборами чисел $(i_1, \ldots, i_s)$, $0 < i_1 < \ldots < i_s \leqslant m$, $s = 0, \ldots, r$, так, что $H = \|h^u_{i_1,\ldots,i_s}\|$, $u = 0, \ldots, k$. Ясно, что i-й столбец E_i матрицы E можно представить в виде

$$E_i = E(\beta_i) = (f_1(\beta_i), \ldots, f_k(\beta_i))^T, \tag{4}$$

где $f_u(x) = \sum_{s=0}^{r} \sum_{0<i_1<\ldots<i_s\leqslant m} h^u_{i_1,\ldots,i_s} x_{i_1} \ldots x_{i_s}$, $u = 1, \ldots, k$ — булевы функции, порядок нелинейности которых не превосходит r, и $\beta_j = \sigma(\alpha_j)$, где $\sigma(\cdot)$ — перестановка элементов m-мерного пространства $(\mathbf{F}_2)^m = \{\alpha_1, \ldots, \alpha_N\}$, которая соответствует перестановочной матрице Γ. Напомним, что Γ переставляет столбцы матрицы R, которые занумерованы m-мерными векторами из $(\mathbf{F}_2)^m$. Так как H — невырожденная матрица, функции $f_u(x)$, $u = 1, \ldots, k$, являются линейно независимыми над $\mathbf{F}_2$.

Как было отмечено, построение алгоритма дешифрования типа 2 сводится к нахождению какого-либо решения уравнения (2) относительно неизвестных

матриц H и Γ, где предполагается, что матрицы $E = \|e_{i,j}\|$, $i = 1, \ldots, k$, $j = 1, \ldots, N$, и R известны.

Уравнение (2) очевидным образом может быть представлено как система, которая содержит kN уравнений с $k^2 + mN$ неизвестными, принимающими значение из $\mathbf{F}_2$ (k^2 неизвестных элементов матрицы H и mN неизвестных $\sigma(\alpha_j)$).

Отметим, что для кода RM_3 длины 1024 ($m = 10$) эта система имеет 40116 неизвестных и более 180000 уравнений. Каждое уравнение имеет первый порядок нелинейности относительно переменных h и третий — относительно остальных переменных.

Рассматриваемые ниже способы решения уравнения (2) направлены на раздельное определение неизвестных матриц H и Γ: сначала определяется перестановочная матрица Γ, а затем матрица H. Предварительно рассмотрим задачу о числе решений уравнения (2).

Если Γ' — перестановочная матрица, для которой

$$R \cdot \Gamma' \cdot R = H' \cdot R \tag{5}$$

при некоторой невырожденной матрице H', то перестановка σ' столбцов матрицы R, соответствующая Γ', называется элементом группы автоморфизмов кода с порождающей матрицей R. Совокупность всех таких σ' образует группу автоморфизмов G_r кода RM_r. Известно [4], что группа G_r совпадает с полной аффинной группой пространства $(\mathbf{F}_2)^m$ и состоит из всех аффинных перестановок вида $\sigma: x \to Ax + \beta$, где A — невырожденная матрица размера $m \times m$ и $\beta \in (\mathbf{F}_2)^m$.

Отсюда вытекает, что если Γ — решение уравнения (2) и Γ' — аффинная перестановочная матрица (т. е. матрица, которой соответствует аффинная перестановка σ'), то матрица $\Gamma \cdot \Gamma'$ также является решением уравнения (2).

Очевидно, любой упорядоченный набор векторов $\delta_1, \ldots, \delta_m$ из $(\mathbf{F}_2)^m$, содержащий m линейно независимых над $\mathbf{F}_2$ элементов, с помощью некоторой линейной перестановки σ можно перевести в стандартный упорядоченный набор векторов (стандартный базис) $\Xi = \{\xi_1, \ldots, \xi_m\}$, где ξ_i — вектор, у которого только одна координата, стоящая на i-ом месте, равна единице. Кроме того, с помощью аффинной перестановки можно перевести в Ξ и некоторые наборы векторов, содержащие $m - 1$ линейно независимых векторов.

Будем говорить, что набор $\mathrm{B} = \{\beta_1, \ldots, \beta_m\}$ (первые m элементов в (4)) обладает свойством (А), если существует аффинная перестановка, которая переводит набор B в набор Ξ. Предполагая, что Γ является реализацией случайной величины равномерно распределенной на множестве всех перестановок матриц размера $N \times N$, нетрудно подсчитать вероятность события, заключающегося в том, что набор B обладает свойством (А). Эта вероятность близка к 1/2. В дальнейшем, для простоты изложения, будем предполагать, что набор B обладает свойством (А). Это предположение для рассматриваемых ниже методов является предположением в пользу нападающей стороны, поскольку несколько уменьшает действительную сложность дешифрования системы.

Из этого предположения вытекает, что найдется перестановка σ' из группы автоморфизмов кода RM_r, которая переводит векторы β_i, $i = 1, \ldots, m$, (первые m аргументов булевых функций в (6)) в множество Ξ. Замена Γ на $\Gamma \cdot \Gamma'$ в (2), где Γ' — перестановочная матрица, соответствующая σ', с учетом (5) приводит

к тому, что $\beta_i = \xi_i$, $i = 1, \ldots, m$. Другими словами, будем предполагать, что первые m векторов β_i являются векторами $\xi_1, \ldots, \xi_m$.

Лемма 1. *Пусть* $\psi_{j_s} \in (\mathbf{F}_2)^m$ *и* $\{E(\psi_{j_s})\}$, $s = 1, \ldots, v$, $v = 2^{r+1}$, — *множество столбцов матрицы* E, *определенных равенством* (4). *Соотношение*

$$\sum_{s=1}^{v} E(\psi_{j_s}) = 0 \tag{6}$$

имеет место тогда и только тогда, когда множество $\Psi = \{\psi_{j_1}, \ldots, \psi_{j_v}\}$ *является смежным классом* $L + \gamma$, *где* L — *подпространство пространства* $(\mathbf{F}_2)^m$ *размерности* $r + 1$, *а* γ — *элемент* $(\mathbf{F}_2)^m$.

Доказательство. Легко показать, что $\sum_{\alpha \in L+\gamma} q_I(\alpha) = 0$, где $q_I(x) = x_{i_1} \ldots x_{i_s}$, $s \leqslant r$, — произвольная конъюнкция. Следовательно, для любой булевой функции $f(x)$ порядка нелинейности не выше r справедливо равенство $\sum_{\alpha \in L+\gamma} f(\alpha) = 0$, т. е. (6) выполнено, если $\Psi = L + \gamma$.

Пусть булева функция $g(x)$ такова, что $g(\psi_{j_s}) = 1$ для $s = 1, \ldots, v$ и $g(\psi) = 0$ для остальных векторов ψ из $(\mathbf{F}_2)^m$, т. е. $g(x)$ — характеристическая функция множества Ψ. Координатами столбца $E(x)$ являются линейно независимые функции. Поэтому из соотношения (6) вытекает, что для любой функции $f(x)$ порядка нелинейности не выше r справедливо равенство

$$\sum_{\beta \in (\mathbf{F}_2)^m} f(\beta) g(\beta) = 0.$$

Другими словами, вектор $(g(\beta_1), \ldots, g(\beta_N))$ принадлежит коду $\mathrm{RM}_r^{\perp}$, двойственному к коду RM_r. Как известно [4], кодом, двойственным к коду RM_r, является код RM_{m-r-1}, т. е. функция $g(x)$ является функцией порядка нелинейности не выше $m - r - 1$. Также известно, что код RM_{m-r-1} имеет кодовое расстояние, равное $v = 2^{r+1}$, и булевы функции $g_0(x)$, определяющие векторы минимального веса, имеют вид $g_0(x) = l_1(x) \ldots l_{r+1}(x)$, где $l_u(x) = a_{u,1}x_1 + \ldots + a_{u,m}x_m + a_{u.0}$ — аффинная функция.

Последнее доказывает лемму, ибо код с порождающей матрицей E является кодом RM_r с переставленными коэффициентами, а функция $g_0(x)$, очевидно, принимает значения 1 только на элементах некоторого смежного класса по подпространству размерности $r + 1$.

Обозначим через $L(\delta_1, \ldots, \delta_s)$ пространство, натянутое на векторы $\delta_1, \ldots, \delta_s$ из $(\mathbf{F}_2)^m$, через $\hat{L}$ — пространство $L(\xi_{j_1}, \ldots, \xi_{j_{r+1}})$ и через $\hat{L}_0$ — подпространство размерности r пространства $\hat{L}$, состоящее из всевозможных векторов $\xi = \sum_{i=1}^{r+1} a_i \xi_{j_i}$, у которых $\sum_{i+1}^{r+1} a_i = 0$. Напомним, что ξ_i — вектор с одной единичной координатой.

Лемма 2. *Пусть* $L + \gamma$ — *смежный класс, для которого*

$$L + \gamma \supset \{\xi_{j_1}, \ldots, \xi_{j_{r+1}}\}. \tag{7}$$

Тогда при $\gamma \in \hat{L}_0$ *пространство* L *совпадает с* $\hat{L}$, *а при* $\gamma \notin \hat{L}_0$ *пространство имеет вид* $L = \hat{L}_0 \cup \hat{L}_0 + \lambda$, *где* $\lambda = \xi_{j_1} + \gamma$.

Доказательство. Из (7) вытекает, что

$$L \supset L(\xi_{j_1}+\gamma,\ldots,\xi_{j_{r+1}}+\gamma) = \hat{L}_0 \cup \hat{L}_0 + \xi_{j_1} + \gamma \supset \{\xi_{j_1}+\gamma,\ldots,\xi_{j_{r+1}}+\gamma\}.$$

Если $\gamma \in \hat{L}_0$, то $L \supset \hat{L}_0 \cup \hat{L}_0 + \gamma_{j_1} = \hat{L}$, т. е. $l = \hat{L}$. Если же $\gamma \notin \hat{L}_0$, то в качестве L со свойством (7) можно взять подпространство $L = \hat{L}_0 \cup \hat{L}_0 + \lambda$, $\lambda = \xi_{j_1} + \gamma$. Лемма доказана.

Следствие 1. *При фиксированном множестве* $\{\xi_{j_1},\ldots,\xi_{j_{r+1}}\}$ *число пар* (L,γ), *для которых выполнено* (10), *равно числу различных смежных классов по подпространству* $\hat{L}_0$, *равному* 2^{m-r}.

Как было отмечено, можно предполагать, что $\beta_j = \xi_j$, $1 \leqslant j \leqslant m$. Будем изучать алгоритмы, которые позволяют вычислять значения неизвестных векторов β_j с $j > m$, которые определяют j-й столбец матрицы E (см. (4)).

Пусть $\Phi = \{\xi_{j_1},\ldots,\xi_{j_{r+1}}\}$, $0 < j_1 < \ldots < j_{r+1} \leqslant m$, — $(r+1)$-элементное подмножество Ξ. Как следует из следствия 1, для каждого Φ имеется 2^{m-r} множеств $\mathrm{B} = \{\beta_{i_1},\ldots,\beta_{i_{v-r-1}}\}$, $v = 2^{r+1}$, таких, что $\Phi \cup \mathrm{B} = L + \gamma = \hat{L}_0 \cup (\hat{L}_0 + \xi_{j_1} + \gamma)$, т. е. таких, что

$$G = G(\Phi \cup \mathrm{B}) = \sum_{\beta \in \Phi \cup \mathrm{B}} E(\beta) = \sum_{u=1}^{r+1} E(\beta_{i_u}) + \sum_{s=1}^{v-r-1} E(\beta_{i_s}) = 0. \tag{8}$$

Таким образом, для подмножества $\mathcal{G} = \{E(\beta_{j_1}),\ldots,E(\beta_{j_{r+1}})\}$ первых m столбцов матрицы E существует 2^{m-r} различных подмножеств

$$\mathcal{D} = \{E(\beta_{i_1}),\ldots,E(\beta_{i_{v-r-1}})\},$$

для каждого из которых выполнено соотношение (8). Предположим, что мы каким-либо способом, например, способом описанным ниже, нашли какие-либо два $\mathcal{D}_1$ и $\mathcal{D}_2$ из этих 2^{m-r} множеств $\mathcal{D}$. Каждому множеству $\mathcal{G} \cup \mathcal{D}_i$, $i = 1,2$, соответствует множество $\Phi \cup \mathrm{B}_i$ элементов из $(\mathbf{F}_2)^m$, которые определяют эти столбцы в соответствии с представлением (4). По лемме 2 пересечение $\Lambda = (\Phi \cup \mathrm{B}_1) \cap (\Phi \cup \mathrm{B}_2)$ совпадает с подпространством $\hat{L}_0$ (определение приведено перед формулировкой леммы 2), элементы которого тем самым становятся известными. Таким образом, мы для заданного фиксированного множества Φ определим множество $\Lambda = \hat{L}_0$ тех β_j, которые образуют $\hat{L}_0$.

Пусть Φ' — $(r+1)$-элементное подмножество Ξ, такое что $\Phi' \cap \Phi = \varnothing$. Проделаем аналогичную работу по поиску множества $\hat{L}'_0$ для набора Φ'. Ввиду того, что пересечение $\hat{L}_0 \cap \hat{L}'_0$ равно нулевому вектору, мы тем самым найдем тот столбец $E_j = ((\Phi \cup \mathrm{B}_1) \cap (\Phi \cup \mathrm{B}_2)) \cap ((\Phi' \cup \mathrm{B}'_1) \cap (\Phi' \cup \mathrm{B}'_2))$ матрицы E, для которого $\beta_j = 0$.

Если в качестве Φ и Φ' взять множества, имеющие два общих элемента, скажем, ξ_{s_1} и ξ_{s_2}, то, как легко установить, пересечение $\hat{L}_0 \cap \hat{L}'_0$ совпадает с множеством $\{0, \xi_{s_1} + \xi_{s_2}\}$, что позволяет определить столбец E_j, для которого $\beta_j = \xi_{s_1} + \xi_{s_2}$.

Продолжая этот процесс, можно достаточно простым и очевидным образом найти значения β_j для всех столбцов матрицы E. Оценим сложность этого алгоритма нахождения элементов β_j.

Рассматриваемый алгоритм требует для своей реализации нахождения двух различных соотношений вида (8) при заданном $(r+1)$-элементном множестве Φ. Очевидным способом нахождения соотношений (8) при фиксированном Φ является последовательный перебор всевозможных множеств $\mathcal{D}$ столбцов матрицы E, содержащих $s = 2^{r+1} - r - 1 = v = r = 1$ элементов, до тех пор, пока сумма $G = G(\Phi \cup \mathrm{B})$ не станет равной 0. Ввиду того, что при фиксированном Φ имеется ровно 2^{m-r} таких множеств B, математическое ожидание числа актов выбора $\mathcal{D}$ до получения нулевого значения суммы G при случайном и равновероятном выборе множеств $\mathcal{D}$, очевидно, равно $\binom{N-m}{s}2^{r-m}$.

Если учесть, что эту работу в простейшем варианте алгоритма нахождения всех β_j необходимо проделать дважды для всех множеств Φ, т. е. $2\binom{m}{r+1}$ раз, то общая трудоемкость нахождения нужных соотношений вида (8) будет не меньше $\binom{m}{r+1}\binom{N-m}{s}2^{r-m+1}$ даже без учета затрат по вычислению сумм G.

Величина $\binom{m}{r+1}\binom{N}{s}2^{r-m+1}$ при $N = 1024$, $m = 10$ и $r = 3$ равна $8 \cdot 10^{27}$. Отметим, что идеи работы [16] позволяют сократить трудоемкость поиска соотношений вида (8), что позволяет уменьшить при фиксированном множестве Φ число операций необходимых для нахождения множеств B, для которых сумма G равна нулю. Здесь на этом останавливаться не будем, а приведем расчеты сложности алгоритма поиска соотношений вида (8), основанного на хорошо известных идеях использования памяти.

Зафиксируем множество Φ. Пусть $q = [s/2]$ и $w = s - q$, т. е. q, w — разбиение числа $s = 2^{r+1} - r - 1$ на две приблизительно равные части. Произведем случайно и равновероятно M_1 и M_2 выборок с возвращением q-элементных множеств Δ и, соответственно, w-элементных множеств Ω из множества $N - m$ последних столбцов матрицы E. Для каждой выборки вычислим сумму $G(\Delta) = \sum_{\beta \in \Delta} E(\beta)$ ее элементов.

Память составим из блоков, адресами которых являются k-мерные двоичные векторы. Блок памяти содержит информацию о номерах столбцов q-элементного множества и суммы $G(\Delta)$ его элементов, сложенной с фиксированным вектором $G(\Phi) = \sum_{\beta \in \Phi} E(\beta)$. Адресом этого блока является k-мерный вектор $D = G(\Delta) + G(\Phi)$. Таким образом, блок q-элементного множества содержит примерно $k + mq$ двоичных ячеек, в которых записаны координаты k-мерного вектора D и q различных m-мерных векторов, которые являются номерами столбцов множества Δ. Совершенно аналогично, но без сложения с фиксированным вектором $G(\Phi)$, образуется блок с информацией о w-элементных множествах. Адресом этого блока служит сумма $G(\Omega)$ его элементов.

Из построения блоков следует, что объединение множеств Δ и Ω блоков с одинаковыми адресами дает множество $\mathrm{B} = \Delta \cup \Omega$ с нулевой суммой

$$G = \sum_{\beta \in \Phi} E(\beta) + \sum_{\beta \in \mathrm{B}} E(\beta),$$

т. е. множество B, для которого выполнено (8).

Все $M_1 + M_2$ блоков образуют массив памяти с числом ячеек памяти, приблизительно равным

$$T(M_1, M_2) = (M_1 + M_2)(k + m(2^{r+1} - r - 1)/2). \qquad (9)$$

Массив памяти каким-либо образом сортируется по значениям адресов блоков. Например, блоки упорядочиваются в соответствии с лексикографическим

порядком их адресов. В результате блоки с одинаковыми адресами станут находиться по соседству один к другому, т. е. их можно будет легко выделить.

При фиксированном множестве Φ получим оценки для чисел M_1 и M_2, которые обеспечивают высокую вероятность появления в памяти двух блоков с одинаковыми адресами.

Множество B, $|\mathrm{B}| = 2^{r+1} - r - 1 = s$, с нулевой суммой $G = G(\Phi \cup \mathrm{B})$ (при заданном множестве Φ) можно разбить $\binom{s}{q}$ способами на два множества с q и w элементами, $q + w = s$. Следовательно, при случайном и равновероятном выборе q и w-элементных множеств математическое ожидание числа пар выбранных q-элементных и w-элементных множеств, объединение которых составят заданное множество B, как легко установить, равно $A(M_1, M_2) = M_1 M_2 \binom{s}{q} \binom{N-m}{q}^{-1} \binom{N-m}{w}^{-1}$. Число всех множеств $\mathcal{D}$ с нулевой суммой G равно 2^{m-r} (следствие 1). Поэтому, если целые числа M_1 и M_2 выбраны так, чтобы величина $2^{m-r} A(M_1, M_2)$ была существенно большей, чем 2, то с вероятностью, близкой к 1, из выбранных q-элементных и w-элементных множеств можно будет составить по меньшей мере два множества B с нулевой суммой G. Отсюда вытекает, что при достаточно большой постоянной C условие

$$M_1 M_2 > C 2^{r-m+1} \binom{s}{q}^{-1} \binom{N-m}{q} \binom{N-m}{w} \geqslant C 2^{r-m+1} \binom{N-m}{s} \tag{10}$$

является достаточным для того, чтобы с вероятностью, близкой к 1, можно было при фиксированном множестве Φ найти два множества B с нулевой суммой G.

Очевидно, для минимизации величины $M_1 + M_2$ необходимо положить $M_1 = M_2 = M$. В этом случае условие (10) будет иметь вид

$$M > \left(C 2^{r-m+1} \binom{N-m}{s} \right)^{1/2},$$

а оценка (9) примет вид

$$T(M, M) > \left(C 2^{r-m+1} \binom{N-m}{s} \right)^{1/2} (2k + m(2^{r+1} - r - 1)) = Q(m, r). \tag{11}$$

Величина $Q(m, r)$, т. е. нижняя оценка числа ячеек памяти, при $m = 10, 11$ и $r = 3$, $C = 1$ равна $1{,}9 \cdot 10^{15}$, $1{,}7 \cdot 10^{17}$.

Подсчитаем число операций $U(m, r)$, необходимых для реализации алгоритма по нахождению двух множеств B с нулевой суммой G для всех множеств Φ с помощью алгоритма с памятью. При фиксированном множестве Φ для упорядочивания памяти объема $T(m, r)$ с помощью известных алгоритмов (см. [18]) необходимо проделать не менее $U_1(m, r) = T(m, r) \log_2 T(m, r)$ операций. Число различных множеств Φ равно $\binom{m}{r+1}$. Следовательно,

$$U(m, r) = U_1(m, r) \binom{m}{r+m} = \binom{m}{r+1} T(m, r) \log_2 T(m, r). \tag{12}$$

Заметим, что оценки снизу, вытекающие из (11) и (12), величины $U(m, r)$ при $m = 10, 11$, $r = 3$ и $C = 1$ равны $2{,}0 \cdot 10^{19}$ и $3{,}2 \cdot 10^{21}$. Реально эти оценки выше, ибо они получены при существенных упрощающих предположениях.

Дальнейшие шаги алгоритма дешифрования, в частности, нахождение матрицы H в системе (2) при известной матрице Γ требуют числа операций существенно меньшего, чем $U(m,r)$, и в дальнейшем учитываться не будет.

4. Анализ усиленной системы открытого шифрования

Так же как в §2, сложность $U(uN,k,t')$ алгоритма дешифрования типа 1 определяется соотношением (5). Отметим, что величина $U(uN,k,t')$ при $N\to\infty$, u, $r=\text{const}$, по порядку совпадает с $U(N,k,t)$.

Переходим к эвристическому обсуждению сложности алгоритма дешифрования типа 2 с помощью методов §2 и §3.

Во-первых, отметим, что уравнение (2) в данном случае имеет вид

$$E=\|H_1R\dots H_uR\|\cdot\Gamma \tag{13}$$

Соображения, подобные соображениям в начале §3, показывают, что вместе с матрицей Γ решением (13) являются матрица $\Gamma\cdot\Gamma'$,

$$\Gamma'=\left\|\begin{array}{ccccc}\Gamma'_1 & 0 & \dots & 0 & 0\\ \dots & \dots & \dots & \dots & \dots\\ 0 & 0 & \dots & 0 & \Gamma'_u\end{array}\right\|,$$

где Γ'_i, $i=1,\dots,u$ — перестановочные матрицы размера $N\times N$, соответствующие, вообще говоря, различным аффинным перестановкам $\mathbf{F}_{2^m}$.

Отсюда вытекает, что в отличие от §3 нельзя считать первые m векторов β_j равными ξ_j, так как среди первых столбцов E могут быть столбцы матрицы R, умноженные на матрицы H_i с различными значениями i. Будем называть столбец E_k матрицы E столбцом вида i, если он имеет вид $E_k=H_iR_j$, где R_j — j-й столбец матрицы R.

Для того, чтобы можно было использовать метод определения значений перестановки $\sigma(\cdot)$, рассмотренный в §2, т. е. метод нахождения матрицы Γ, необходимо сначала каким-либо образом отделить столбцы вида, например, 1 от столбцов вида i, $i\neq 1$. Это можно сделать, избежав прямого перебора столбцов E, следующим образом.

Во-первых, отметим, что кодом, двойственным к коду RM_r, является код RM_{m-r-1} с кодовым расстоянием $d=2^{r+1}$. Число векторов C_d кода RM_{m-r-1} минимального веса d равно (см. [4])

$$S_d=2^{m-r-1}\prod_{i=0}^{r}\frac{2^{m-1}-1}{2^{r-i-1}-1}. \tag{14}$$

Предположим, что матрицы Γ и H_i, $i=1,\dots,u$, являются реализациями независимых при различных i случайных величин равномерно распределенных, соответственно, на множестве всех перестановочных матриц и множестве всех невырожденных матриц с элементами из $\mathbf{F}_2$. Из (13) вытекает следующее утверждение.

Лемма 3. *Пусть E_j, $j = 1, \ldots, uN$ — столбцы матрицы E, $v = 2^{r+1}$, и $G = \sum_{s=1}^{v} E_{j_s}$. Тогда при постоянном r и $m \to \infty$*

(A) *вероятность $\mathbf{P}(G = 0 \mid A)$ события $G = 0$ при условии A состоит в том, что все столбцы E_{j_s} являются столбцами одного вида, равна $S_d/\binom{N}{v} = C2^{m(r+2-2^{r+1})}$ при некоторой постоянной C.*

(B) *вероятность $\mathbf{P}(G = 0 \mid B)$ события $G = 0$ при условии B состоит в том, что столбцы E_{j_s} не являются столбцами одного вида равна 2^{-k}, где $k = \sum_{i=0}^{r} \binom{m}{i}$.*

Следует обратить внимание на то, что при фиксированном r, и $m \to \infty$ вероятность $\mathbf{P}(G = 0 \mid A)$ существенно больше вероятности $\mathbf{P}(G = 0 \mid B)$. Поэтому, если $G = 0$, то с вероятностью, близкой к единице, все столбцы E_{j_s}, $s = 1, \ldots, v$, являются столбцами одного вида.

Несколько усиливая позицию нападающей стороны, будем предполагать, что выполнено следующее условие: $G = 0$ тогда и только тогда, когда столбцы E_{j_s}, $s = 1, \ldots, v$, являются столбцами одного вида.

Используя это предположение, столбцы матрицы E можно рассортировать по видам следующим образом.

Комплекты (множества) $\mathcal{J} = \{E_{j_1}, \ldots, E_{j_v}\}$ и $\mathcal{I} = \{E_{i_1}, \ldots, E_{i_v}\}$ столбцов E с нулевой суммой ($G = \sum_{s=1}^{v} E_{j_s} = 0$) назовем зависимыми, если они имеют непустое пересечение. Комплекты $\mathcal{J}$ и $\mathcal{I}$ назовем связанными, если найдется последовательность $\mathcal{J}_1, \ldots, \mathcal{J}_s$, $\mathcal{J}_1 = \mathcal{J}$, $\mathcal{J}_s = \mathcal{I}$, с зависимыми парами $\mathcal{J}_i$, $\mathcal{J}_{i+1}$ комплектов, $i = 1, \ldots, s-1$. Пусть $\mathcal{M}$ — множество с максимальным числом элементов, состоящее из связанных комплектов $\mathcal{J}$ столбцов матрицы E, и E' — подмножество столбцов E, образованное объединением всех комплектов из $\mathcal{M}$. Как следует из предположения, все столбцы из E' являются столбцами одного вида. Таким образом, если найти достаточное число связанных комплектов с нулевой суммой, т. е. комплектов $\mathcal{J} = \{E_{j_1}, \ldots, E_{j_v}\}$, для которых $G = 0$, то их объединение будет образовывать множество столбцов одного вида. Всего необходимо найти u классов связанных комплектов для того, чтобы рассортировать все столбцы E по их принадлежности к тому или иному виду.

Таким образом, задача сортировки столбцов сводится к задаче нахождения достаточно большого (содержащего не менее $uN/2^r$ элементов) множества связанных комплектов с нулевой суммой.

Эта задача может быть решена с использованием памяти совершенно аналогично тому, как это было сделано в §3. Перед подсчетом необходимого для этого числа операций сделаем следующее замечание.

Как следует из предположения, число соотношений вида (13) среди столбцов матрицы E равно uC_d, ибо каждому соотношению (13) соответствует вектор веса d из пространства, ортогонального к пространству строк матрицы E, т. е. вектор из кода, двойственному к коду с порождающей матрицей E. Таким образом, память заполняется суммами из $v/2$ случайно выбранных столбцов и информацией о номерах столбцов, входящих в сумму. Для того, чтобы получить не менее $N/2^r$ комплектов с нулевой суммой, число таких сумм должно

быть порядка

$$Y = Y(u, N, r) = \binom{uN}{d/2}\left(2uN\left(2^r C_d\binom{d}{d/2}\right)^{-1}\right)^{1/2}. \qquad (15)$$

Действительно, фиксированный комплект столбцов матрицы E можно разбить $\binom{d}{d/2}$ способами на два множества по $d/2$ элементов в каждом множестве. Следовательно, при случайном выборе u комплектов из $d/2$ столбцов матрицы E математическое ожидание числа пар тех комплектов, которые составят фиксированный набор столбцов с нулевой суммой, очевидно, равно $V(Y) = \binom{Y}{2}\binom{d}{d/2}P^2$, где $P = \binom{uN}{d/2}^{-1}$ — вероятность выбора фиксированного комплекта из $d/2$ столбцов E. Отсюда и из условия $V(Y) \geqslant uN/2^r$ вытекает соотношение (14).

Таким образом, память должна состоять примерно из $Y(u, N, r)$ блоков, имеющих примерно $k + (\log_2 uN)d/2 = k + 2^{-1}md\log_2 u$ ячеек. Общий объем памяти оценивается снизу величиной

$$X(u, N, r) = (k + 2^{-1}md\log_2 u)Y(u, N, r).$$

Например, $X(4, 1024, 3) = 1{,}1\cdot 10^{24}$, что, очевидно, очень далеко от возможного в настоящее время. Таким образом, задача сортировки столбцов матрицы E по видам сводится к поиску векторов веса 2^{r+1} в коде с проверочной матрицей E, который является достаточно сложной задачей.

Случайный перебор комплектов из m столбцов матрицы E до встречи комплекта, содержащего, например, только столбцы вида 1 требует порядка u^m шагов. Поэтому, если в первом алгоритме из §3 определения матрицы Г, в котором не используется память, в качестве начальных столбцов $E_1, \ldots, E_m$ взять случайно набранный комплект из m столбцов E, то вероятность того, что все столбцы комплекта имеют один вид равна, приблизительно, u^{-m+1}. Другими словами, сложность определения матрицы Г в рассматриваемом случае увеличится по сравнению со сложностью алгоритма из §3, по меньшей мере, в u^m раз.

Кроме того, отметим, что сложность алгоритма из §3 с памятью существенно увеличится, например, из-за того, что оценка для величины M определяющей объем памяти будет иметь вид $M > \left(2^{r-m+1}\binom{uN-m}{s}\right)^{1/2}$, т. е. увеличится примерно в $u^{s/2}$ раз ($s = 2^{r+1} - r - 1$).

Подводя итоги, следует отметить, что методы дешифрования, разработанные в §§2, 3, к усиленной системе применимы после значительного усложнения алгоритма дешифрования. Сложность этого усложненного алгоритма в u^v, а объем требуемой памяти, по меньшей мере, в $u^{s/2}$ выше, чем для исходного алгоритма дешифрования немодернизированной системы ($v = 2^{r+1}$, $s = 2^{r+1} - r - 1$). Вместе с тем сложность зашифрования и расшифрования, а также скорость передачи сообщений для усложненной системы практически не меняется по сравнению с исходной системой секретной связи, описанной в §1.

По мнению автора, эта система при $u \geqslant 4$, $N \geqslant 1024$, $r \geqslant 3$ будут иметь стойкость к нападению существенно более высокую, чем системы рассмотренные выше. Например, нижняя оценка памяти $Q(m, r)$ (см. (11)), требуемая для дешифрования методами §3 рассматриваемой системы с $u = 4$, $m = 10$, $N = 1024$,

$r = 3$, $t' = 803$, будет равна $Q(10,3) = 1{,}2 \cdot 10^{17}$ двоичных ячеек, а общая трудоемкость дешифрования $U'(10,3)$, которая подсчитывается с существенными упрощающими предположениями для нападающей стороны, будет не меньше $1{,}2 \cdot 10^{24}$.

5. Представление двоичной информации в виде векторов веса, не превосходящего t

В настоящем разделе рассматривается следующая задача. Пусть $B(N,t)$ — множество всех двоичных векторов длины N и веса не превосходящего t, $0 \leqslant t \leqslant N$. Необходимо построить эффективный алгоритм, который устанавливает взаимно однозначное соответствие между двоичными n-мерными векторами из $(\mathbf{F}_2)^n$, где $n \leqslant [\log_2 |B(N,t)|]$ и $[x]$ — целая часть числа x, и векторами некоторого подмножества W_t, $|W_t| = 2^n$, множества $B(N,t)$.

Слово эффективность означает, что алгоритм должен быстро для любого b, $b \in (\mathbf{F}_2)^n$, находить соответствующий ему вектор $f(b)$, $f(b) \in W_t$, и, наоборот, для каждого вектора c, $c \in W_t$, быстро вычислять вектор $f^{-1}(c)$ из $(\mathbf{F}_2)^n$. Кроме того, будем требовать, чтобы число n при заданных N и t было достаточно близко к числу $\log_2 |B(N,t)|$, верхней оценке n.

Простейший способ решения этой задачи состоит в следующем. Число N представим в виде $N = sp + p_0$, $0 \leqslant p_0 < p$, $s = [N/p]$. Для простоты дальнейших рассуждений предположим, что $p_0 = 0$, т. е. $s = N/p$. Пусть W'_t — множество p-мерных векторов веса не превосходящего $t' = [pt/N]$, с числом элементов 2^q, где $q = [\log_2 B(p,t')]$. Установим каким-либо образом соответствие между элементами множеств W'_t и $(\mathbf{F}_2)^q$. Для получения требуемого соответствия между множествами $(\mathbf{F}_2)^n$ и W_t, где $n = sq$, поступим следующим образом: n-мерный двоичный вектор разобьем на s q-мерных. Каждый q-мерный вектор заменим на соответствующий ему p-мерный. В результате получим N-мерный вектор веса, не превосходящего $t's \leqslant t$. Совершенно так же устанавливается и обратное соответствие между W_t и $(\mathbf{F}_2)^n$.

Таким образом, множество W_t является произведением s комплектов множества W'_t и сложность прямого и обратного отображений, очевидно, определяется величиной q. Отметим, что для рассмотренного отображения величина $u = sq$ при малых значениях q существенно меньше, чем $\log_2 |B(N,t)|$. Другими словами, множество W_t составляет малую часть множества $B(N,t)$. Например, при $N = 1024$, $t = 200$ находим, что $\log_2 |B(1024,200)| = 724{,}6$, $n = 16$, $s = 64$, $t' = 3$, $q = [\log_2 |B(16,3)|] = 9$, $u = 9 \cdot 64 = 576$.

Рассмотрим принципиально иной способ представления двоичной информации в виде вектора длины N и веса не превосходящего t, который требует для своей реализации весьма небольшую память. Пусть K — двоичный линейный код длины N с числом информационных разрядов k, который имеет радиус покрытия не меньше t. Предположим, что информационными являются первые k разрядов кода K. Пусть $u = N - k$, $\mathbf{x}$ — двоичный вектор длины u и $\mathbf{y}$ — двоичный вектор длины N, полученный из $\mathbf{x}$ добавлением к нему слева k нулей. Подберем в коде K вектор $\mathbf{b}$, отстоящий от $\mathbf{y}$ на расстояние Хемминга не больше t, т. е. $w(\mathbf{y} + \mathbf{b}) \leqslant t$. Таким образом, вектор $\mathbf{e} = \mathbf{b} + \mathbf{y}$ имеет вес не более t и вектор $\mathbf{x}$ может быть восстановлен из вектора $\mathbf{b}$ следующим способом. С

помощью первых k разрядов вектора $\mathbf{e}$ восстанавливается вектор $\mathbf{b}$. Вектор $\mathbf{x}$ по построению совпадает с последними u разрядами вектора $\mathbf{e}+\mathbf{b}$.

Сложность этого алгоритма в основном определяется сложностью нахождения вектора $\mathbf{e}$, находящегося на расстоянии не более t от вектора $\mathbf{b}$. Алгоритмов решения этой задачи, кроме так называемого, корреляционного (см. [7, 18]), неизвестно. Корреляционный алгоритм, по крайней мере, для кодов RM_r, $r > 1$, имеет сложность реализации, по порядку равную числу элементов этого кода. Для кода RM_1 известен корреляционный алгоритм со сложностью $N \log_2 N$ операций, основанный на быстром умножении вектора на матрицу Адамара [4].

Следует также отметить, что радиус покрытия известен только для очень узкого класса кодов. Например, для кодов RM_1 длины $N = 2^m$, m — четное, он равен $1/2(N - (N)^{1/2})$ [4], для нескольких известных совершенных кодов он равен $(d-1)/2$, для кодов БЧХ, исправляющих две ошибки, он равен 3.

В качестве примера рассмотрим совершенный двоичный код Голлея длины 23 и размерности 12 с радиусом покрытия 3. Этот код позволяет взаимно однозначно отображать двоичные векторы длины 11 во все векторы длины 23 и веса 3.

Список литературы

1. McEliece R. J. A public-key cryptosystem based on algebraic coding theory. DSN Progress Report 42–44, Jet Propulsion Lab., Pasadena, CA, January–February, 1978, pp. 114–116.
2. Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory. *Probl. Control and Inform. Theory* (1986) **15**, 19–34.
3. Сидельников В. М., Першаков А. С. Декодирование кодов Рида–Маллера при большом числе ошибок. *Пробл. перед. инф.* (1992) **28**, №3, 80–94.
4. Мак-Вильямс Ф. Д., Слоэн Н. Дж. *Теория кодов, исправляющих ошибки.* Связь, Москва, 1979.
5. Riek J. R. Observations on the application of error-correcting codes to public key encryption. In: *Intern. Carnahan Conf. on Security Technology.* 1990, pp. 15–18.
6. Cryptology and Computational Number Theory. In: *Proc. Symp. Appl. Math.* 1989, **42**.
7. Berlekamp E. R., McEliece R. J., van Tilborg H. T. A. On the interent intractability of certain coding problem. *IEEE Trans.* (1978) **24**, 384–386.
8. Зайцев Г. В., Зиновьев В. А., Семаков Н. В. Быстрое корреляционное декодирование блочных кодов. В кн.: *Кодирование и передача дискретных сообщений в системах связи.* Наука, Москва, 1976, с. 74–85.
9. Евсеев Г. С. О сложности декодирования линейных кодов. *Пробл. перед. инф.* (1983) **19**, №1.
10. Крук Е. А. Границы для сложности декодирования линейных кодов. *Пробл. перед. инф.* (1989) **25**, №3, 103–107.
11. Бассалыго Л. А., Зяблов В. В., Пинскер М. С. Проблемы сложности в теории корректирующих кодов. *Пробл. перед. инф.* (1977) **13**, 5–13.
12. Корякин Ю. Д. Быстрое корреляционное декодирование кодов Рида–Маллера. *Пробл. перед. инф.* (1987) **23**, №2, 40–49.
13. Levitin L. B., Hartman C. P. A new approach to the general minimum distance decoding problem: the zero-neighbors algorithm. *IEEE Trans.* (1985) **31**, 378–384.

14. Ntafos G. C., Hakimi G. L. On the complexity of some coding problems. *IEEE Trans.* (1981) **27**, 794–796.

15. Coffey J. T., Goodman R. M. The complexity of information decoding. *IEEE Trans. Information Theory* (1990) **36**, 1031–1037.

16. Adams C. M., Meijer H. Security-related comments regarding McEliece's public-key cryptosystem. *Lecture Notes in Computer Sci.* (1988) **293**, 224–228.

17. Lee P. J., Brickell E. F. An observation on the security of the McEliece public-key cryptosystem. *Lecture Notes in Computer Sci.* (1988) **230**, 224–228.

18. Leon J. G. A probabilistic algorithm for computing weights of large error-correcting codes. *IEEE Trans.* (1988) **34**, 1354–1359.

19. Кнут Д. *Искусство программирования для ЭВМ, т. 3: Сортировка и поиск.* Мир, Москва, 1979.

Статья поступила 19.05.93.