УДК 519.725

А. Л. Шишкин

Московский физико-технический институт (государственный университет)

Комбинированный метод построения многокомпонентных сетевых кодов

Предложен новый метод построения многокомпонентных сетевых кодов на основе ранговых подкодов. Метод сочетает в себе «жадный» лексикографический перебор при поиске компонент сетевого кода, а также использование неравномерно ограниченных ранговых кодов для кодирования внутри компонент. Приведены примеры кодов, получены оценки мощности, осуществлено сравнение с верхней границей.

Ключевые слова: коды над подпространствами, коды постоянного веса, «жадный» перебор.

1. Введение

Рассмотрим $\mathbb{K}_q$ — конечное поле из q элементов. Построим над $\mathbb{K}_q$ n-мерное векторное пространство $\mathbb{K}_q^n$ и обозначим через $\mathcal{A}(n)$ множество всех его подпространств. Подпространство $\mathcal{V}$ из $\mathcal{A}(n)$, имеющее размерность $k \leqslant n$, будет представлять собой совокупность q^k векторов длины n из $\mathbb{K}_q^n$.

Подпространство $\mathcal{V}$ можно рассматривать как линейную оболочку над k векторами из $\mathbb{K}_q^n$ или, что эквивалентно, над матрицей размера $k \times n$ с элементами из $\mathbb{K}_q$. Таким образом, каждая матрица V размерности $k \times n$ над полем $\mathbb{K}_q$ однозначно задает некоторое k-мерное подпространство $\mathcal{V}$ из $\mathcal{A}(n)$. Однако каждое подпространство из $\mathcal{A}(n)$ может иметь несколько порождающих его матриц: умножение матрицы V на невырожденную матрицу T размерности $k \times k$ даст матрицу $\tilde{V} = TV$ размера $k \times n$, задающую то же подпространство, что и матрица V.

В связи с этим подпространства удобно задавать с помощью матриц в приведенной ступенчатой форме, к которой произвольная матрица размера $k \times n$ может быть приведена с помощью метода гауссовых исключений. Приведенная ступенчатая форма матриц характеризуется следующими свойствами:

- Первый справа ненулевой элемент каждой строки равен 1. Он называется *ведущим элементом.*
- Каждый ведущий элемент является единственным ненулевым элементом в своем столбце.
- Ведущий элемент следующей строки всегда расположен правее ведущего элемента предыдущей строки.
- На остальные элементы матрицы ограничений не накладывается. Они называются *свободными элементами.*

Ниже приводится пример матрицы размером 4×8 в приведенной ступенчатой форме (символами «*» обозначены свободные элементы):

$$\begin{pmatrix} 0 & 1 & * & 0 & 0 & * & 0 & * \\ 0 & 0 & 0 & 1 & 0 & * & 0 & * \\ 0 & 0 & 0 & 0 & 1 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & * \end{pmatrix}. \tag{1}$$

Между k-мерными подпространствами из $\mathcal{A}(n)$ и матрицами размера $k \times n$ над $\mathbb{K}_q$ в приведенной ступенчатой форме можно установить взаимно однозначное соответствие.

Расположение ведущих элементов матрицы размером $k \times n$ в приведенной ступенчатой форме можно описать с помощью мультииндекса $\mathcal{I} = \{i_1, i_2, \ldots, i_k\}$, определяемого как набор целых чисел, соответствующих номерам столбцов, в которых находятся ведущие элементы всех k строк, начиная с первой. Еще один способ описания расположения ведущих элементов — с помощью двоичного индекс-вектора v длины n, который содержит 1 в тех своих компонентах, номера которых соответствуют номерам столбцов матрицы с ведущими элементами, и 0 в остальных элементах.

К примеру, матрица размером 4×8 из приведенного выше примера содержит ведущие элементы во втором, четвертом, пятом и седьмом столбцах. Она будет описываться мультииндексом $\mathcal{I} = \{2, 4, 5, 7\}$ и индекс-вектором $v = (0, 1, 0, 1, 1, 0, 1, 0)$.

Число свободных элементов зависит от мультииндекса, соответствующего матрице в приведенной ступенчатой форме. Для матрицы размером $k \times n$ с мультииндексом $\mathcal{I} = \{i_1, i_2, \ldots, i_k\}$ число свободных элементов будет равно

$$f = f_{k,n}(\mathcal{I}) = kn - \frac{k(k-1)}{2} - (i_1 + i_2 + \cdots + i_k). \tag{2}$$

При этом количество различных матриц с одним и тем же мультииндексом $\mathcal{I}$ (а значит, и количество соответствующих им подпространств) будет равно q^f.

2. Коды над подпространствами

Пусть $\mathcal{U}$ и $\mathcal{V}$ — два произвольных подпространства $\mathcal{A}(n)$. Между ними можно ввести расстояние, называемое *расстоянием Грассмана* [1]:

$$d_{sub}(\mathcal{U}, \mathcal{V}) = \dim(\mathcal{U} \cup \mathcal{V}) - \dim(\mathcal{U} \cap \mathcal{V}). \tag{3}$$

Пусть $\mathcal{U}$ имеет размерность m и задается порождающей матрицей U размером $m \times n$, а $\mathcal{V}$ имеет размерность k и задается порождающей матрицей V размером $k \times n$. Тогда расстояние Грассмана между этими подпространствами можно вычислить следующим образом:

$$d_{sub}(\mathcal{U}, \mathcal{V}) = 2\mathrm{rank}\left(\begin{bmatrix} U \\ V \end{bmatrix}\right) - m - k. \tag{4}$$

В случае подпространств равной размерности ($m = k$) расстояние d_{sub} будет четным для любых $\mathcal{U}$ и $\mathcal{V}$.

С использованием расстояния Грассмана вводится понятие *кодов над подпространствами*. Подпространственным $[n, M, d_{sub}, k]$-кодом называется множество k-мерных подпространств $\mathcal{A}(n)$, содержащее M элементов, причем расстояние Грассмана между любой парой подпространств не меньше d_{sub}.

Подпространственные коды находят широкое применение в теории случайного сетевого кодирования [2–4]. Важной задачей при построении подпространственных кодов является максимизация их мощности M при фиксированной исправляющей способности, определяемой расстоянием d_{sub}.

Расстояние Грассмана тесно связано с ранговым расстоянием, используемым при построении ранговых кодов [1, 5]. Пусть подпространства $\mathcal{U}$ и $\mathcal{V}$ размерности k, заданные матрицами U и V, имеют мультииндексы $\mathcal{I}_{\mathcal{U}}$ и $\mathcal{I}_{\mathcal{V}}$, а также вектор-индексы $v_{\mathcal{U}}$ и $v_{\mathcal{V}}$.

Для иллюстрации связи с ранговыми кодами введем понятие *подматрицы ведущих элементов*. Для матрицы U в приведенной ступенчатой форме совокупность ее столбцов, содержащих ведущие элементы, называется *подматрицей ведущих элементов* и обозначается U_l. Совокупность оставшихся столбцов называется *подматрицей свободных элементов* и обозначается U_f. Заметим, что U_f содержит все свободные элементы матрицы U. Далее возможны два случая.

Если мультииндексы $\mathcal{I}_{\mathcal{U}}$ и $\mathcal{I}_{\mathcal{V}}$ совпадают, то совпадает также расположение свободных элементов в соответствующих им матрицах, и выражение для грассманова расстояния между подпространствами можно записать в следующем виде:

$$\begin{aligned} d_{sub}(\mathcal{U},\mathcal{V}) = 2\mathrm{rank}\left(\begin{bmatrix} U \\ V \end{bmatrix}\right) - 2k = 2\mathrm{rank}\left(\begin{bmatrix} U_l|U_f \\ V_l|V_f \end{bmatrix}\right) - 2k = \\ = 2k + 2\mathrm{rank}([U_f - V_f]) - 2k = 2\mathrm{rank}([U_f - V_f]). \end{aligned} \quad (5)$$

В промежуточных выкладках использован тот факт, что для подпространств размерности k ранг матрицы ведущих элементов в точности равен k. Отметим, что итоговое выражение в точности соответствует удвоенному ранговому расстоянию между матрицами свободных элементов U_f и V_f.

Таким образом, для подпространств внутри одного мультииндекса задача построения кода максимальной мощности при заданном подпространственном расстоянии d_{sub} сводится к построению рангового кода максимальной мощности с ранговым расстоянием $d_{rank} = d_{sub}/2$.

Рассмотрим теперь случай различных мультииндексов $\mathcal{I}_{\mathcal{U}}$ и $\mathcal{I}_{\mathcal{V}}$. В этом случае количество линейно независимых столбцов в матрице $\begin{bmatrix} U \\ V \end{bmatrix}$ не меньше, чем количество ненулевых элементов в векторе $v_{\mathcal{U}} \vee v_{\mathcal{V}}$. Следовательно,

$$\begin{aligned} d_{sub}(\mathcal{U},\mathcal{V}) = 2\mathrm{rank}\left(\begin{bmatrix} U \\ V \end{bmatrix}\right) - 2k \geqslant 2(v_{\mathcal{U}} \vee v_{\mathcal{V}}) - 2k = \\ = 2(2k - v_{\mathcal{U}} \wedge v_{\mathcal{V}}) - 2k = 2k - 2v_{\mathcal{U}} \wedge v_{\mathcal{V}} = v_{\mathcal{U}} \oplus v_{\mathcal{V}} = d_{ham}(v_{\mathcal{U}}; v_{\mathcal{V}}), \end{aligned} \quad (6)$$

где d_{ham} обозначает расстояние в хэмминговой метрике между двоичными вектор-индексами матриц.

Таким образом, для подпространств с различными мультииндексами грассманово расстояние может быть оценено снизу через хэммингово расстояние между вектор-индексами, соответствующими подпространствам.

Теперь задачу построения подпространственных кодов можно декомпозировать на две составляющие: сначала нужно построить в хэмминоговой метрике код постоянного веса из вектор-индексов с минимальным расстоянием $d_{ham} = d_{sub}$, а затем для каждого вектор-индекса построить в соответствующем подпространстве вложенный ранговый код на подматрице свободных элементов с минимальным расстоянием $d_{rank} = d_{sub}/2$.

Кодовые слова подпространственного кода, соответствующие одному вектор-индексу, называют принадлежащими одной кодовой компоненте. Код, содержащий кодовые слова из нескольких кодовых компонент, называется *многокомпонентным кодом.*

Для мощности многокомпонентых подпространственных кодов с параметрами $n, k, d_{rank} = d_{sub}/2$ существует верхняя граница [6]:

$$M \leqslant \frac{\begin{bmatrix} n \\ k - d_{rank} + 1 \end{bmatrix}}{\begin{bmatrix} k \\ k - d_{rank} + 1 \end{bmatrix}}, \quad (7)$$

где

$$\begin{bmatrix} n \\ s \end{bmatrix} = \frac{(q^n - 1)(q^n - q)\dots(q^n - q^{s-1})}{(q^s - 1)(q^s - q)\dots(q^s - q^{s-1})}. \quad (8)$$

Рассмотрим некоторые примеры алгоритмов построения подпространственных кодов и конкретные примеры известных конструкций.

3. Существующие методы построения подпространственных кодов

Исторически первой конструкцией подпространственных кодов, использующей ранговые коды в качестве подкодов внутри одной компоненты, является код Сильвы–Кеттера–Кшишанга (СКК-код) [7]. Для подпространств $\mathcal{A}(n)$ размерности k используется только одна кодовая компонента с мультииндексом $\mathcal{I} = \{1, 2, \ldots, k\}$. Подматрица свободных элементов при этом имеет размеры $k \times (n-k)$ и состоит только из свободных элементов. Для параметров $n = 8, k = 4$ матрица этого кода имеет вид

$$\begin{pmatrix} 1 & 0 & 0 & 0 & * & * & * & * \\ 0 & 1 & 0 & 0 & * & * & * & * \\ 0 & 0 & 1 & 0 & * & * & * & * \\ 0 & 0 & 0 & 1 & * & * & * & * \end{pmatrix}. \tag{9}$$

Мощность СКК-кода может быть найдена следующим образом [1]:

$$M = \begin{cases} q^{(n-k)(k-d_{rank}+1)}, & \text{если } n \geqslant 2k; \\ q^{k(n-k-d_{rank}+1)}, & \text{если } n < 2k. \end{cases} \tag{10}$$

СКК-код используется в качестве первой компоненты во всех конструкциях многокомпонентных кодов и дает компоненту с наибольшей мощностью. Однако суммарная мощность у любой многокомпонентной конструкции превосходит мощность СКК-кода.

В конструкции Габидулина–Боссерта [8] используется подпространственное расстояние $d_{sub} = 2k$ — максимально возможное для подпространств размерности k. В такой конструкции матрицы кодовых компонент в приведенной ступенчатой форме получаются из матрицы СКК-кода путем добавления нулевых префиксных матриц. Код Габидулина–Боссерта с параметрами $n = 8, k = 4$ будет состоять из двух компонент:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & * & * & * & * \\ 0 & 1 & 0 & 0 & * & * & * & * \\ 0 & 0 & 1 & 0 & * & * & * & * \\ 0 & 0 & 0 & 1 & * & * & * & * \end{pmatrix}, \tag{11}$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{12}$$

Эти коды для значений n, кратных k, достигают верхней границы мощности [8]. Однако ограничение на $d_{sub} = 2k$ существенно уменьшает свободу разработчика при выборе параметров проектируемого кода.

Более гибкой с точки зрения допустимых параметров кода является конструкция Габидулина–Пилипчук [1], использующая для поиска мультииндексов, соответствующих кодовым компонентам, комбинаторные блок-схемы. В качестве подкодов внутри кодовых компонент в этой конструкции используются ранговые коды с ограничениями на кодовые символы. Такие подкоды достигают границы Синглтона и позволяют использовать те же алгоритмы кодирования и декодирования, что и для классических кодов с максимальным ранговым расстоянием [1].

Недостатком этого подхода является ограничение на выбор параметров кода, связанное с использованием комбинаторных блок-схем. Для поиска кодовых компонент можно использовать любой двоичный код постоянного веса в хэмминговой метрике, однако не каждый такой код является блок-схемой [9]. Также недостатком этого метода является отсутствие универсального алгоритма определения мощности рангового подкода для произвольной матрицы свободных элементов. В результате для некоторых кодовых компонент построение оптимального рангового подкода становится сложной задачей [1].

В работах Этзиона и Зильберштейна [10,11] для поиска кодовых компонент используется полный перебор по всем двоичным векторам длины n с хэмминговым весом k. Двоичные векторы при этом предварительно сортируются в лексикографическом порядке. Это позволяет отказаться от блок-схем и дает полную свободу в выборе параметров многокомпонентного кода. Однако этот алгоритм также не предлагает универсальный способ определения мощности ранговых подкодов.

В данной работе описан подход, совмещающий преимущества методов Габидулина–Пилипчук и Этзиона–Зильберштейна. Для поиска кодовых компонент используется перебор по всем двоичным векторам, а в качестве подкодов используются ранговые коды с ограничениями на информационные символы. При этом для ранговых подкодов используется универсальный алгоритм определения мощности, позволяющий автоматизированно строить коды с большим числом компонент. Дополнительной особенностью метода является то, что поиск по двоичным векторам осуществляется с помощью «жадного» перебора: компоненты выбираются не в лексикографическом порядке, а в порядке уменьшения своей мощности. Новый метод позволяет построить коды, совокупная мощность компонент которых в ряде случаев превосходит ранее известные конструкции с теми же параметрами.

4. Новый метод построения многокомпонентных подпространственных кодов

На первой стадии предлагаемого метода для кода с параметрами n, k, d_{sub} формируется список из всех двоичных векторов с длиной n и хэмминговым весом k. Количество таких векторов равно C_k^n. Для программной реализации построения этого списка удобно использовать алгоритм Нараяны.

На второй стадии каждый вектор v из полученного списка рассматривается в качестве индекс-вектора матрицы размером $k \times n$ в приведенной ступенчатой форме. По восстановленной из вектора матрице V определяется ее матрица свободных элементов V_f. Расположение свободных элементов в этой матрице используется для определения мощности рангового подкода, который можно разместить в соответствующей V кодовой компоненте.

Алгоритм определения мощности подкода с ранговым расстоянием $d_{rank} = d_{sub}/2$ следующий. Вычисляется $d_{rank} - 1$ — число проверочных символов рангового подкода. Фиксируется горизонтальное расположение символов рангового подкода (в строках матрицы V_f). Определяется количество свободных элементов в строке матрицы V_f с номером $d_{rank} - 1$ — оно равно размерности рангового подкода N. Определяется i_h — общее количество свободных элементов в строках матрицы V_f с номерами от d_{rank} до $\min\{N, k\}$.

Мощность рангового подкода при горизонтальном расположении символов равна $C_h = q^{i_h}$. Аналогичная процедура повторяется для вертикального расположения символов рангового подкода (в столбцах матрицы V_f) и определяется мощность рангового подкода для этого случая: $C_v = q^{i_v}$. Итоговая мощность рангового подкода определяется как $\max\{C_h, C_v\}$.

На третьей стадии используются полученные ранее списки вектор-индексов и мощностей соответствующих им подкодов длиной C_k^n. На этой стадии осуществляется «жадный» перебор компонент с максимальной мощностью и добавление их к строящемуся многокомпонентному коду. Первым добавляется СКК-код, имеющий максимальную мощность среди всех компонент. Далее из оставшихся ищется вектор-индекс с максимальной мощностью. Если он находится на хэмминговом расстоянии не менее $d_{ham} = d_{sub}$ от каждого из уже добавленных к коду вектор-индексов, то он также добавляется к коду и изымается из списка. В противном случае ищется вектор-индекс со следующей по убыванию мощностью рангового подкода и также проверяется на минимальное расстояние до уже сформированного подпространственного кода.

Поиск прекращается и код объявляется построенным в тот момент, когда любой из оставшихся вектор-индексов находится на расстоянии, меньшем d_{ham} от какого-либо вектор-индекса из добавленных к коду.

5. Примеры новых кодовых конструкций

Рассмотрим пример использования нового метода для кода с параметрами $n = 7, k = 3, d_{sub} = 4$. Список всех двоичных векторов длины n с хэмминговым весом k (их количество равно $C_3^7 = 35$) имеет вид

$$\begin{array}{ccccccc} \{1110000, & 1101000, & 1100100, & 1100010, & 1100001, & 1011000, & 1010100, \\ 1010010, & 1010001, & 1001100, & 1001010, & 1001001, & 1000110, & 1000101, \\ 1000011, & 0111000, & 0110100, & 0110010, & 0110001, & 0101100, & 0101010, \\ 0101001, & 0100110, & 0100101, & 0100011, & 0011100, & 0011010, & 0011001, \\ 0010110, & 0010101, & 0010011, & 0001110, & 0001101, & 0001011, & 0000111\}. \end{array} \quad (13)$$

Рассмотрим алгоритм определения мощности ранговых подкодов на примере вектор-индекса $v = \{0011010\}$. Соответсвующая ему матрица в приведенной ступенчатой форме:

$$V = \begin{pmatrix} 0 & 0 & 1 & 0 & * & 0 & * \\ 0 & 0 & 0 & 1 & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 1 & * \end{pmatrix}. \quad (14)$$

Количество кодовых символов рангового подкода равно $d_{rank} - 1 = d_{sub}/2 - 1 = 1$. Для горизонтального расположения символов рангового подкода количество свободных элементов в строке с номером $d_{rank} - 1$ равно 2, то есть размерность рангового подкода $N = 2$. Общее количество свободных элементов в строках с номерами от $d_{rank} = 2$ до $\min\{N, k\} = 2$ равняется 2. Следовательно, $i_h = 2$ и $C_h = q_h^i = q^2$.

Рассматривая аналогично вертикальное расположение, получаем $N = 3$ и $i_v = 2$, что дает $C_v = q_v^i = q^2$. Окончательно мощность рангового подкода равна $\max\{C_h, C_v\} = \max\{q^2, q^2\} = q^2$.

Повторяя аналогичные действия для остальных индекс-векторов, получаем список из $C_3^7 = 35$ мощностей кодовых компонент: $\{q^8, q^7, q^6, q^5, q^4, q^6, q^5, q^4, q^3, q^4, q^3, q^2, q^2, q, q^0, q^6, q^5, q^4, q^3, q^4, q^3, q^2, q^2, q, q^0, q^3, q^2, q^2, q, q, q^0, q^0, q^0, q^0, q^0\}$.

В качестве первой компоненты выбираем СКК-код с мощностью q^8 (его вектор-индекс первый в списке). Следующим по мощности подкодом обладает второй вектор-индекс — его мощность равна q^7. Однако он находится на хэмминговом расстоянии 2 от вектор-индекса СКК-кода, что меньше требуемого $d_{ham} = d_{sub} = 4$. Первым в списке по убыванию мощностей подкодов, удовлетворяющим требованию на минимальное расстояние, является 10-й вектор-индекс (мощность равна q^4). Его мы и добавляем к коду.

Продолжая поиск вектор-индексов по описанной процедуре, получаем многокомпонентный код из 7 компонент со следующими мультииндексами: $\mathcal{I}_1 = \{1, 2, 3\}$, $\mathcal{I}_2 = \{1, 4, 5\}$, $\mathcal{I}_3 = \{2, 4, 6\}$, $\mathcal{I}_4 = \{3, 4, 7\}$, $\mathcal{I}_5 = \{2, 5, 7\}$, $\mathcal{I}_6 = \{3, 5, 6\}$, $\mathcal{I}_7 = \{1, 6, 7\}$. Его мощность равна $M = q^8 + q^4 + q^3 + q^2 + q + q + 1$. Верхняя граница для мощности при тех же параметрах равна $M_{\max} = q^8 + q^6 + q^5 + q^4 + q^3 + q^2 + 1$. Этот код совпадает с кодом, построенным в работе [1] для тех же параметров, отличаясь лишь порядком мультииндексов.

Рассмотрим теперь код с параметрами $n = 13, k = 4, d_{sub} = 6$. Описанный метод позволяет построить для этих параметров многокомпонентный код со следующими мультииндексами:
$\mathcal{I}_1 = \{1, 2, 3, 4\}$, $\mathcal{I}_2 = \{1, 5, 6, 7\}$, $\mathcal{I}_3 = \{2, 5, 8, 9\}$, $\mathcal{I}_4 = \{3, 6, 8, 10\}$, $\mathcal{I}_5 = \{4, 7, 8, 11\}$, $\mathcal{I}_6 = \{3, 7, 9, 12\}$, $\mathcal{I}_7 = \{4, 5, 10, 12\}$, $\mathcal{I}_8 = \{4, 6, 9, 13\}$, $\mathcal{I}_9 = \{1, 9, 10, 11\}$, $\mathcal{I}_{10} = \{2, 6, 11, 12\}$, $\mathcal{I}_{11} = \{2, 7, 10, 13\}$, $\mathcal{I}_{12} = \{3, 5, 11, 13\}$, $\mathcal{I}_{13} = \{1, 8, 12, 13\}$.

Мощность такого кода равна $M = q^{18} + q^{12} + q^8 + q^7 + q^6 + q^4 + 2q^3 + 3q^2 + q + 1$. В работе [1] для тех же параметров был построен код с мощностью $q^{18} + q^{12} + 2q^6 + 2q^5 + 3q^4 + 2q^3 + q^2 + 1$. Таким образом, новый метод позволяет улучшить ранее известную конструкцию. Верхняя граница для мощности такого кода равна $M_{\max} = q^{18} + q^{15} + q^{14} + q^{12} + q^{11} + q^{10} + q^9 + q^8 + q^7 + q^6 + q^4 + q^3 + 1$.

В завершение рассмотрим код с параметрами $n = 31, k = 3, d_{sub} = 4$ для размерности базового поля $q = 2$. Этот код состоит из 155 компонент, его мощность при построении по описанному методу равняется $M = 81955583110556320$ кодовых слов. Верхняя граница составляет $M_{\max} = 109802047904403264$ кодовых слова. Отношение фактической мощности к верхней границе составляет примерно 0,746.

Отметим также, что предложенный метод построения для $d_{sub} = 2k$ совпадает в методом Габидулина–Боссерта.

6. Вывод

В работе предложен новый метод построения многокомпонентных сетевых кодов, сочетающий в себе «жадный» перебор мультииндексов в лексикографическом порядке, а также использование ранговых подкодов с ограничениями на информационные символы в качестве подкодов внутри компонент. Предложен универсальный метод построения подкодов с известной мощностью для любых конфигураций матриц свободных элементов.

Приведены примеры новых кодов, которые в некоторых случаях превосходят по характеристикам ранее известные конструкции. Для них определена фактическая мощность и осуществлено сравнение с верхней границей, которой при определенных параметрах эти коды достигают.

Литература

1. *Габидулин Э. М., Пилипчук Н. И.* Ранговые подкоды в многокомпонентном сетевом кодировании // Проблемы передачи информации. — 2013. — Т. 49, вып. 1. — С. 46–60.
2. *Koetter R., Kschischang F. R.* Coding for Errors and Erasures in Random Network Coding // IEEE Transactions on Information Theory. — 2008. — V. 54, N 8. — P. 3579–3591.
3. *Skachek V.* Recursive Code Construction for Random Networks // IEEE Transactions on Information Theory. — 2010. — V. 56, N 3. — P. 1378–1382.
4. *Ahlswede R., Aydinian H.* On error control codes for random network coding // Workshop on Network Coding, Theory, and Applications, 2009. NetCod ’09. — 2009. — P. 68–73.
5. *Габидулин Э. М.* Теория кодов с максимальным ранговым расстоянием // Проблемы передачи информации. — 1985. — Т. 21, вып. 1. — С. 1–12.
6. *Wang X.* Linear Authentication Codes from Free Modules: Bounds and Constructions // WSEAS Transactions on Mathematics. — 2013. — V. 12, N 2. — P. 201–210.
7. *Silva D., Kschischang F. R., Koetter R.* A Rank-Metric Approach to Error Control in Random Network Coding // IEEE Transactions on Information Theory. — 2008. — V. 54, N 9. — P. 3951–3967.
8. *Габидулин Э. М., Боссерт М.* Алгебраические коды для сетевого кодирования // Проблемы передачи информации. — 2009. — Т. 45, вып. 4. — С. 3–18.
9. *Холл М.* Комбинаторика. — М. : Мир, 1970.
10. *Etzion T., Silberstein N.* Error-Correcting Codes in Projective Spaces via Rank-Metric Codes and Ferrers Diagrams // IEEE Transactions on Information Theory. — 2011. — V. 55, N 7. — P. 2909–2919.
11. *Etzion T., Silberstein N.* Large Constant Dimension Codes and Lexicodes // Advances in Mathematics of Communications. — 2011. — V. 5, N 2. — P. 177–189.

Поступила в редакцию 26.04.2014.