ОБРАБОТКА И ПЕРЕДАЧА ИНФОРМАЦИИ

УДК 681.325.3

Н. В. Пацей, доцент (БГТУ)

МЕТОДЫ ГЕНЕРАЦИИ ПРОВЕРОЧНЫХ МАТРИЦ ИТЕРАТИВНОГО БЛОЧНОГО КОДА С НИЗКОЙ ПЛОТНОСТЬЮ ПРОВЕРОК НА ЧЕТНОСТЬ

Статья посвящена проблеме построения проверочных матриц кода с низкой плотностью проверок на четность для заданных характеристик скорости и длины кода. Рассмотрен метод генерации проверочных матриц на основе случайной перестановочной подматрицы, известный как квазициклическое регулярное кодирование, и структурированный метод генерации матрицы на основе евклидово-геометрического кода с возможностью удаления строк и деления столбцов. Выполнено моделирование кодов.

The article is devoted to the problem of construction parity check matrixes for low density parity check codes with given characteristic of code rate and code length which used in. Method of generating check matrixes on base of random permutation sub-matrices arrangement, known as quasi-cyclic regular code and method of structured generating check matrixes on base of Euclid geometric codes with possibility rows deletion and columns division are described. Simulation analysis of low density parity check codes performance and equalizability is done.

Введение. Проблема построения, адаптации и модификации известных кодов по-прежнему остается одной из актуальных задач в области теории кодирования.

Высокая производительность и корректирующая способность, получаемая при итеративном декодировании турбокодов, стимулировали активные исследования по применению данного метода к декодированию других видов кодов. В частности, оказалось, что возможно получить на порядок лучшие характеристики при использовании низкоплотностных (low density parity check – LDPC) кодов, предложенных Галлагером [1]. Этим объясняется выбор Network Working Group кодов с низкой плотностью проверок на четность в качестве стандарта IETF RFC5170 для Интернет-коммуникаций [2] и интеграция кодов в стандарты беспроводной связи и цифрового ТВ (DVB-S2).

Одной из задач, связанной с построением низкоплотностных кодов, является генерация проверочных матриц с заданными свойствами.

Основная часть. Код с малой плотностью проверок на четность задается проверочной матрицей H, обладающей свойством разряженности, т. е. ее строки и столбцы содержат малое число ненулевых позиций по сравнению с размерностью матрицы. Проверочная H и генераторная G матрицы должны удовлетворять следующему условию:

$$GH^T = 0.$$

Процесс кодирования последовательности $(m_1, m_2, \ldots, m_k)$ заключается в получении кодированной последовательности $(c_1, c_2, \ldots, c_n)$:

$$C = (m_1, m_2, \ldots, m_k)G = (c_1, c_2, \ldots, c_n)$$

при условии, что:

$$H(c_1, c_2, \ldots, c_n)^T = 0.$$

В настоящее время используются два принципа построения проверочных матриц кода.

Первый основан на генерации начальной проверочной матрицы с помощью псевдослучайного генератора. Коды, полученные таким методом, называют случайными (random-like codes). Основной недостаток случайных кодов – необходимость применения алгоритмов удаления циклов (в особенности коротких) и нестабильные рабочие характеристики кода.

Второй базируется на использовании специальных структурированных методов, основанных на группах, конечных полях и т. п. Несмотря на то, что лучшие результаты по исправлению ошибок дают именно случайные LDPC-коды, структурированные методы позволяют использовать оптимизацию процедур хранения, кодирования и декодирования, а также получать коды с более предсказуемыми и устойчивыми характеристиками.

Изначально Галлагером в [1] была предложена следующая структура проверочной матрицы H с параметрами (n, λ, ρ), где n – количество столбцов матрицы, которая имеет ровно λ единиц в каждом столбце и ρ единиц в каждой строке (остальные нули). Пример матрицы для (20, 3, 4) LDPC-кода представлен формулой (1).

$$H = \begin{pmatrix} H_1 \\ H_2 \\ H_3 \end{pmatrix} = \begin{pmatrix}
1&1&1&1&&&&&&&&&&&&\\
&&&&1&1&1&1&&&&&&&&\\
&&&&&&&&1&1&1&1&&&&\\
&&&&&&&&&&&&1&1&1&1\\
1&&&&1&&&&1&&&&1&&&\\
&1&&&&1&&&&1&&&&1&&\\
&&1&&&&1&&&&1&&&&1&\\
&&&1&&&&1&&&&1&&&&1\\
1&&&&&1&&&&1&&&&1&&\\
&1&&&&&1&&&&1&&&&&\\
&&1&&&&&1&&&&1&&&&\\
&&&1&&&&&1&&&&1&&&\\
&&&&1&&&&1&&&&1&&&\\
&&&&&1&&&&1&&&&1&&\\
&&&&&&1&&&&1&&&&1&\\
&&&&&&&1&&&&1&&&&1
\end{pmatrix}, \quad (1)$$

$$H_{EG(2,2^2)} = \begin{pmatrix}
1&1&&1&&&1&&&&&&&&&\\
&1&1&&1&&&1&&&&&&&&\\
&&1&1&&1&&&1&&&&&&&\\
&&&1&1&&1&&&1&&&&&&\\
&&&&1&1&&1&&&1&&&&&\\
&&&&&1&1&&1&&&1&&&&\\
&&&&&&1&1&&1&&&1&&&\\
&&&&&&&1&1&&1&&&1&&\\
1&&&&&&&&1&1&&1&&&1&\\
&1&&&&&&&&1&1&&1&&&\\
&&1&&&&&&&&1&1&&1&&\\
&&&1&&&&&&&&1&1&&1&\\
1&&&&1&&&&&&&&1&1&&\\
&1&&&&1&&&&&&&&1&1&\\
1&&1&&&&1&&&&&&&&1&
\end{pmatrix}. \quad (2)$$

Как видно, структурно она состоит из трех подматриц, в каждой из которых содержится только одна единица в столбце: в H_1 каждая i-я строка содержит единицы в столбцах от $(i-1)k+1$ до ik; H_2 и H_3 получены путем случайной перестановки столбцов матрицы H_1. Все строки и столбы матрицы H имеют одинаковые веса, что делает код регулярным.

Поскольку использовалась случайная перестановка и нет строго определенного правила, то в дальнейшем необходимо проведение компьютерного поиска для выбора из потенциального множества кодов с наилучшими характеристиками. В любом случае поиск требует затрат времени.

Более универсальным является структурированный метод построения кода на основе евклидово-геометрических кодов $EG(m, 2^s)$ [3]. Данный метод позволяет приблизиться к границе Шеннона при BER (Bit Error Rate), равной 10^{-4}.

Евклидово-геометрические коды строятся как система инциденций геометрии $EG(m, p^s)$. Код имеет следующие характеристики: длина кодового слова – $n = 2^{2s} - 1$; длина информационного слова – $k = 2^{2s} - 3^s$; количество избыточных бит – $n - k = 3^s - 1$; минимальное расстояние – $d_{\min} = 2^s + 1$. Поскольку число единиц в проверочной матрице евклидово-геометрического кода мало по сравнению с размером матрицы, то такой код сам по себе можно рассматривать как LDPC-код. Проверочная матрица H_{EG} строится следующим образом: строки проверочной матрицы соответствуют линиям евклидовой геометрии, столбцы – ненулевым точкам в $EG(m, p^s)$. Элементы матрицы H_{EG} определяются из векторов инциденций линий евклидовой геометрии:

$$H_{EG}(i,j) = \begin{cases} 1, \text{ если точка лежит на прямой } i, \\ 0, \text{ в противном случае.} \end{cases}$$

Если ввести обозначение $q = p^s$, то матрица H_{EG} имеет $n = q^m$ столбцов и $r = q^{m-1}(q^m - 1)/(q-1)$ строк. Каждый столбец матрицы содержит $\lambda = q^{m-1}(q^m - 1)/(q-1)$ единиц, каждая строка содержит $\rho = p^s$ единиц. Например, евклидово-геометрический код $EG(2, 2^2)$ будет иметь проверочную матрицу (2), что соответствует структуре регулярного LDPC (15, 4, 4).

Позднее в [4] было доказано, что нерегулярные LDPC-коды (веса столбцов и строк описываются с помощью функций $\lambda(i)$ и $\rho(i)$, задающих долю столбцов и строк с весом i) имеют лучшие характеристики по сравнению с регулярными.

Для получения оптимального ансамбля и скорости, а также длины кода в [5] предлагается метод построения нерегулярной проверочной матрицы с использованием операций удаления строк и деления столбцов H_{EG}.

Первый шаг метода состоит в определении величины максимального веса столбца dl матрицы. Например, $dl = 4$ (данная цифра выбрана с целью упрощения дальнейших расчетов и выборок, в реальных условия значение dl выбирается значительно больше).

Второй шаг – выбор евклидово-геометрического кода. Для рассматриваемого примера $EG(2, 2^s)$. В общем случае необходимо выбрать s таким образом, чтобы $2^{s-1} < dl < 2^{s+1}$. Для $dl = 4$ величина s будет равна 2, следовательно, необходимо выбрать код $EG(2, 2^2)$.

Затем делятся столбцы и распределяются веса матрицы. Разделение и понижение веса столбца выполняется для уменьшения количества циклов длины 6, которые присутствуют в евклидово-геометрических кодах.

Существует несколько способов разделения столбцов. Например, столбец с постоянным весом p^s делится случайным образом на несколько столбцов с меньшим весом, в сумме равным p^s. Для рассматриваемого примера столбец с весом 4 делится на один с весом 2 и два с весом 1. Тогда $\rho_{(2)} = 0{,}3125$, $\rho_{(1)} = 0{,}6875$. Число столбцов R_{EG} евклидово-геометрического кода $EG(2, 2^2)$, составляющих базис, будет:

$$R_{EG} = 2^s \cdot 2^s - 1.$$

После разделения число столбцов становится R_T.

Однако более корректно применение способа случайного разделения, основанного на латинских квадратах. Для этого выбирается простое число P такое, что $P \geq 2^s$. Создается базовая случайная последовательность $C(i)$ длины $P - s$:

$$C(i + 1) = G_0 C(i) \bmod P,$$

где G_0 принадлежит полю Галуа $GF(P)$. Числа, большие 2^s, удаляются из последовательности $C(i)$, и ее длина становится равной 2^s. На основе $C(i)$ генерируется перестановочный шаблон $LB_j(i)$ с шагом $S(j) = j$ для $j = 1, 2, ..., 2^s$. Тогда j-я последовательность латинского квадрата столбца q и строки i вычисляется как

$$L_{jp}(i) = LB_j(((q + i - 2) \bmod 2^s) + 1)$$

для $j = 1, 2, ..., 2^s$, $i = 1, 2, ..., 2^s$ и $q = 1, 2, ..., 2^s$.

Преимущество рассмотренного способа удаления состоит в том, что передающая и принимающая стороны могут генерировать одинаковую случайную последовательность.

Еще один способ разделения столбцов – поиск на основе линейного программирования оптимальных генераторных функций $\rho(x)$ и $\lambda(x)$ так, чтобы их распределение было максимально приближено к гауссовской аппроксимации (возможно итеративное вычисление этих распределений). Для рассматриваемого примера с весом $dl = 4$ и скоростью кода $r = 0{,}5$ распределения $\rho(x)$ и $\lambda(x)$ приведены в табл. 1.

Таблица 1

Распределения весов

Веса x	Доля столбцов с заданным весом $\lambda(x)$	Доля строк с заданным весом $\rho(x)$
1	–	0,6
2	0,2795	0,3
3	0,3477	–
4	0,2	–

Следующий шаг – задание скорости кода r, например 0,5, и установление информационной длины K на основании зависимости: $K = Nr$. Например, если $N = 30$, то $K = 30 \cdot 0{,}5 = 15$.

Последний шаг – удаление строк. Количество удаляемых строк должно вычисляться с учетом длины информационной последовательности: $D_r = R_T - K$. Иными словами, D_r строк должны быть удалены так, чтобы новые вероятности $\rho(x)$ были как можно ближе к первоначальным.

Существует другой способ удаления строк. Строки удаляются сразу в базовом евклидово-геометрическом коде $EG(2, 2^s)$. Число удаляемых строк: $D_{rED} = R_{ED}(R_T - K) / R_T$.

После удаления D_{rED} строк пересчитываются распределения $\rho(x)$ и $\lambda(x)$. Например, для $H_{EG(2,2^2)}$ необходимо удалить 5 строк. Затем выписываются номера столбцов, в которых есть единицы, сортируются и удаляются пять последних. Получаем новые распределения для веса $dl = 4$ и скорости кода $r = 0{,}5$, приведенные в табл. 2.

Таблица 2

Модифицированное распределение весов

Проверочные разряды		
Веса x	Доля столбцов с заданным весом $\lambda(x)$	Количество столбцов с заданным весом
1	0,0250	1
2	0,3000	6
3	0,3750	5
4	0,3000	3
Информационные разряды		
Веса x	Доля строк с заданным весом $\rho(x)$	Количество строк с заданным весом
4	1	10

Перед разделением столбцов матрицы составим таблицу возможных вариантов делений. Например, столбец веса 4 может быть разделен четырьмя возможными способам: 1×4, 2×2, 3×1 + 1×1, 4×1.

На основе подобных таблиц происходит разделение евклидово-геометрического базиса кода и конфигурация проверочной матрицы нерегулярного LDPC-кода. Метод деления можно применять как к столбцам, так и к строкам. Минимальное расстояние кода проверочной матрицы H_{GF} рассчитываться как $d_{\min} \geq \lambda + 1$.

Исследования показывают, что при разделении на основе определенного алгоритма улучшаются корректирующие свойства кода. Однако случайное разделение столбцов и нерегулярность кода (по сравнению с регулярными при одинаковой скорости) приводит к значительному увеличению производительности.

Описанные выше методы были построены на базе евклидово-геометрических кодов. Очевидно, что это не единственные коды, которые можно использовать. Проективно-геометрические и другие коды также подходят в качестве базиса.

Существует еще один метод [6] построения LDPC-кодов, существенно отличающийся от рассмотренного выше. В основе его лежит свойство – любой циклический сдвиг кодового слова есть также кодовое слово. Такой блоковый код называется квазициклическим (QC-LDPC).

Проверочная матрица кода представляет собой сочетание $H = [H_1, H_2, ..., H_p]$ циклических матриц:

$$H_{QC} = \begin{pmatrix} P_{a1,1} & P_{a1,2} & \dots & P_{a1,c-1} & P_{a1,c} \\ P_{a2,1} & P_{a2,2} & \dots & P_{a2,c-1} & P_{a2,c} \\ \dots & \dots & \dots & \dots & \dots \\ P_{am,1} & P_{am,2} & \dots & P_{am,c-1} & P_{am,c} \end{pmatrix}. \quad (3)$$

где aj, k представляет собой циклический сдвиг столбцов матрицы на i разрядов. Таким образом, P_i – перестановочная матрица размера $L \times L$, которая образуется в результате циклического сдвига столбцов на i позиций. Можно получить серию элементарных перестановочных матриц, например, для размера $L = 5$:

$$P_0 = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix} \dots P_4 = \begin{pmatrix} & & & & 1 \\ 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \end{pmatrix}. \quad (4)$$

В реальных системах L выбирается достаточно большим, например $L = 87$, $L = 101$ и т. д.

Выбор единичной матрицы в качестве P_0 не является обязательным. Например, для $L = 5$ P_0 может быть следующей:

$$P_0 = \begin{pmatrix} 1 & & 1 & 1 & \\ & 1 & & 1 & 1 \\ 1 & & 1 & & 1 \\ 1 & 1 & & 1 & \\ & 1 & 1 & & 1 \end{pmatrix}. \quad (5)$$

В общем случае циклическая матрица описывается ассоциируемым полиномом:

$$p_i(x) = \sum_{j=0}^{L-1} (P_i)_{oj} x^j.$$

Первый способ генерации проверочной матрицы квазициклического LDPC-кода основан на случайном распределении перестановочных матриц с заданным распределением $\rho(x)$ и $\lambda(x)$ (в зависимости от их выбора могут быть построены регулярные и нерегулярные коды). Если $L = 47$, $m = 6$, $j = 12$, то общий размер матрицы QC-LDPC кода H со скоростью 0,5 будет $Lm \times Lj$ или 282×564 (6).

Прочерки означают нулевые матрицы, цифры – номер перестановочной матрицы.

Второй способ получения проверочной матрицы основан на выборе двух чисел a и b, принадлежащих ненулевым элементам поля Галуа $GF(L)$, где L – простое число. Тогда заполнение матрицы H размером $Lm \times Lj$ – расстановка перестановочных матриц. В общем случае (s, t)-элемент матрицы равен $P_{s,t} = b^{(s-1)} a^{(t-1)} \bmod L$ для $1 \leq s \leq m$, $1 \leq t \leq j$ (7).

$$H_{QC(282,564)} = \begin{pmatrix} 41 & - & 11 & - & - & 23 & 1 & 0 & - & - & - & - \\ 3 & 28 & - & - & - & 16 & - & 0 & 0 & - & - & - \\ 30 & - & - & 24 & - & 3 & - & - & 0 & 0 & - & - \\ - & 37 & - & 18 & - & 0 & - & - & - & 0 & 0 & - \\ 2 & - & 29 & - & 35 & - & - & - & - & - & 0 & 0 \\ 31 & 42 & - & 13 & - & - & 1 & - & - & - & - & 0 \end{pmatrix}, \quad (6)$$

$$H_{QC(Lm,Lj)} = \begin{pmatrix} P_1 & P_a & P_{a^2} & \dots & P_{a^{j-1}} \\ P_b & P_{ab} & P_{a^2 b} & \dots & P_{a^{j-1} b} \\ \dots & \dots & \dots & \dots & \dots \\ P_{b^{m-1}} & P_{ba^{m-1}} & P_{a^2 b^{m-1}} & \dots & P_{a^{j-1} b^{m-1}} \end{pmatrix}, \quad (7)$$

$$H_{QC(282,564)} = \begin{pmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 17 & 34 & 21 & 42 & 37 & 27 \\ 5 & 10 & 20 & 40 & 33 & 19 & 38 & 29 & 11 & 22 & 44 & 41 \\ 25 & 3 & 6 & 12 & 24 & 1 & 2 & 4 & 8 & 16 & 32 & 17 \\ 31 & 15 & 30 & 13 & 26 & 5 & 10 & 20 & 40 & 33 & 19 & 38 \\ 14 & 28 & 9 & 18 & 36 & 25 & 3 & 6 & 12 & 24 & 1 & 2 \\ 23 & 46 & 45 & 43 & 39 & 31 & 15 & 30 & 13 & 26 & 5 & 10 \end{pmatrix}. \quad (8)$$

Получаем регулярный квазициклический LDPC-код с $\lambda = m - 1$, $\sigma = j - 1$ и скоростью $r \geq 1 - (m / j)$. Для данной матрицы длина самого короткого цикла будет 8 (что значительно больше, чем у LDPC-кодов, построенных на основе евклидово-геометрических кодов). Большая величина длины цикла позволяет эффективно использовать декодирование с распространением доверия. Примером для $L = 47$, $m = 6$, $j = 12$, $a = 2$ и $b = 5$ из $GF(47)$ матрица QC-LDPC-кода H будет матрица (8). Очевидно, что приведенный базовый метод построения квазициклических кодов является основой для модификаций и оптимизации отдельных шагов построения поверочных матриц (например, удлинение, усечение, выкалывание и т. п.).

Заключение. По результатам синтеза коды с проверочными квазициклическими низкоплотностными матрицами, полученными на основе случайных перестановок, имеют производительность в 1,5 раза выше, чем коды с матрицами на основе структурированных евклидово-геометрических кодов при одинаковой скорости кода (0,5), при одинаковом количестве проверочных и информационных разрядов (283, 564). Однако корректирующая способность кодов на основе квазициклических матриц нестабильна и изменяется в диапазоне от 10^{-3} до 10^{-5} (при уровне SNR = 6 dB).

Литература

1. Gallager, R. G. Low Density Parity Check Codes / R. G. Gallager. – Cambridge, MA: MIT. – Press, 1963. – 90 p.

2. Roca, V. RFC 5170 on Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes / V. Roca, C. Neumann, D. Furodet [Electronic resource]. – Mode of access: http://search.usa.gov. – Date of access: 25.06.2008.

3. Kou, Y. Low Density Parity Check Codes based on Finite Geometries: A Rediscovery / Y. Kou, S. Lin, M. P. C. Fossorier R. // IEEE Trans. Inform. Theory. – 2001. – Vol. 47. – P. 2711–2736.

4. Improved low-dencity parity-check codes using irregular graphs and brelief propogation / M. G. Luby [et al.] // Proc. of IEEE Intern. Symposium on Inform. Theory. – Cambridge. Mass, 1998. – P. 171.

5. LDPC code inspection matrix generation method and inspection matrix generation device: pat. US 7089479B2, H03M 13/00 / Wataru Matsumoto [Electronic resource]. – Mode of access: http://search.usa.gov. – Date of access: 10.03.2010.

6. LDPC block and convolutional codes based on circulant matrices / R. M. Tanner [et al.] // IEEE Trans. on Inform. Theory. – 2004. – Vol. 50, № 12. – P. 2966–2984.

Поступила в редакцию 31.03.2010