SHIMAN DMITRY, PATSEI NATALLIA, Belarussian State Technological University

METHOD FOR VARIABLE LOW-DENSITY PARITY CHECK ENCODING

Предложен метод для динамического изменения параметров кодов низкой плотности проверок на четность в соответствии с показателем качества канала связи. Рассмотрено устройство для хранения, динамического выбора проверочной матрицы и корректирующего кодирования. Представлены результаты моделирования характеристик низкоплотностных кодов с переменными параметрами, которые доказывают эффективность предложенного метода.

It is proposed a method for dynamically changing the low-density parity check code parameters in accordance with communication channel quality monitoring and device for storing, dynamic selection of parity check matrices and error-correcting encoding. Finally, resultsof thesimulationcodes characteristics and performance, which prove the effectiveness of proposed method, are presented.

Reliable data transfer is ensured by using channel detecting and correcting errors coding schemes. Low-density parity check (LDPC) codes suggested for use in a variety of transmission system, such as satellite communications, wireless transmissions, fiber optics, and in media storage [1, 2].

The LDPC code can be described by a parity check matrix H dimension of $(n\text{-}k)n$, where k is the systematic bits number and n is a total value of the systematic bits and the parity bits number (code word length). A generator matrix G corresponded to the parity check matrix H. An input data-encoding method is generally used H instead of G. Therefore, matrix H is the most important factor in the encoding/decoding method on base of LDPC codes [3].

Meanwhile, in the latest mobile and wireless communication system a variable code rate $r(r = k/n)$ scheme generally employed. It reduces the errors probabilities and increases the channel bandwidth [4].

A method and apparatus for dynamically (in transmission) changing the low-density parity check code matrixis presented in this article. This leads to the possibility of code rate, code block length and H matrix density control. For example, with the good channel status can be used the parity bits reducing scheme. In case of poor channel, status can be increased with the row and/or column weight of H matrix and number of check bits in code word, which leads to code rate decreasing. Such LDPC encoder/decoder provides multiple block lengths and code rates by supporting a different parity check matrixes. Let call it variable LDPC (VLDPC) codes.

Communication System With VLDPC Codec

Exemplary communication system can be represented as a transmitter and receiver, between which the wireless communication channel as a transmission medium is located (Fig. 1). Describes how to send/receive simplified timing signals are omitted.

For dynamic monitoring of the channel quality is added CRC calculation blocks to the receiving and transmitting side. It base on the use of cyclic codes measurement in actually running channel (carried real traffic). Such scheme cannot locate a bit error (objective measurement accuracy is limited by the block size – CRC-4, CRC-6, CRC-16 or CRC-32 and it is possible the situation with compensation error). The measurement accuracy is corresponded to BLER (block error rate or block errors frequency). Despite this, the obvious advantage of this method that it is not needed disconnection for measuring channel performance.

CRC block connected with analyzer (see Fig. 1). The analyzer compares the CRC values, and performs management LDPC codec functions. It would be desirable to use long-term analysis throughout the day. However, in this case, the system will slowly adapt to the changing quality of

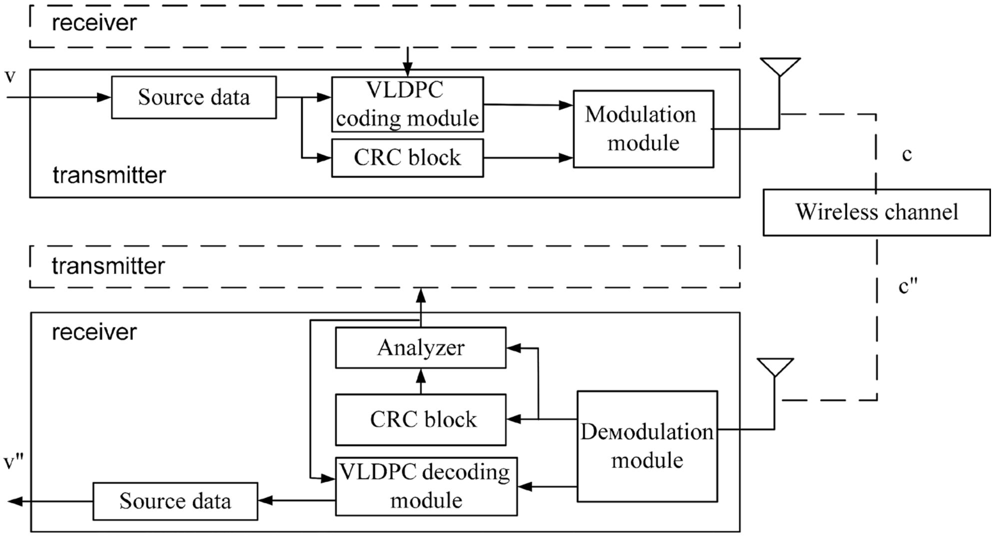

Fig. 1. Structure of wireless error-correction communication system on base of VLDPC

the channel. In presented apparatus the measurement period will depend of data transferred amount. Analyzer will perform the division of the channel quality into six categories and subcategories depending on the BLER, that shown in Table 1.

TABLE 1. **The Channel Category**

Category	BLER
A (high)	BLER ->0
B (good)	BLER < 10^{-7}
C (normal)	10^{-7}< BLER < 10^{-5}
D (below the normal)	10^{-5}< BLER < 10^{-4}
E (low)	10^{-4}< BLER < 10^{-3}
F (degradation)	BLER < 10^{-3}

Then analyzer determines one of LDPC code from Table 2 in accordance with the channel category.

TABLE 2. **VLDPC Parameters**

r	Long		Short	
	K	*n*	*k*	*n*
1/4	16 200	64 800		
1/3	21 600	64 800	5 400	16 200
2/5	25 920	64 800	6 480	16 200
1/2	32 400	64 800		
4/9			7 200	16 200
3/5	38 880	64 800	9 720	16 200
2/3	43 200	64 800	10 800	16 200
3/4	48 600	64 800		
4/5	51 840	64 800		
5/6	54 000	64 800		
7/9			12 600	16 200
8/9	57 600	64 800	14 400	16 200
9/10	58 320	64 800		
wc	64, 128, 256		16, 64, 128	

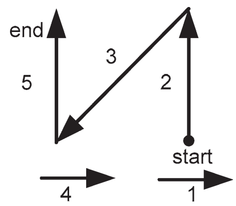

Fig. 2. Direction of code selection in Table 2 in case of sequential movement from *A* to *F* category

The block size *n* selected in accordance with the ITU-T G.821, G.826 and M.2100 recommendations.

If the transfer channel belongs to the category A (Table 1) control unit will select (16200, 14400, 16) code. In the case of a gradual channel quality reduction control unit will increase matrix density *wc* form 16 to 128 and then decrease code rate, providing values *r* = 8/9, 7/9, 2/3, 4/9, 3/5, 2/5, 1/3. If the channel quality has not changed or continues to fall it switches to a long block length *n* = 64800 (VLDPC (64800, 58320, 64)). The next stageis increasing matrix density *wc* from 64 to 256 and decreasing code rate: *r* = 9/10, 8/9, 5/6, 4/5, 3/4, 2/3, 3/5, 1/2, 2/5, 1/3, 1/4. Graphically, in case of gradual decline channel quality from A to F the motion in Table 2 represented as shown in Fig. 2.

VLDPC Encoder

Functionally encoding data processing apparatus (Fig. 3) have blocks for control, generating, storing memory, registers and module two adder. The input coder parameters: systematic block length (*k*), code word length (*n*) and column weight (*wc*) can be changed during the information transfer due to category (Table 1). It will give

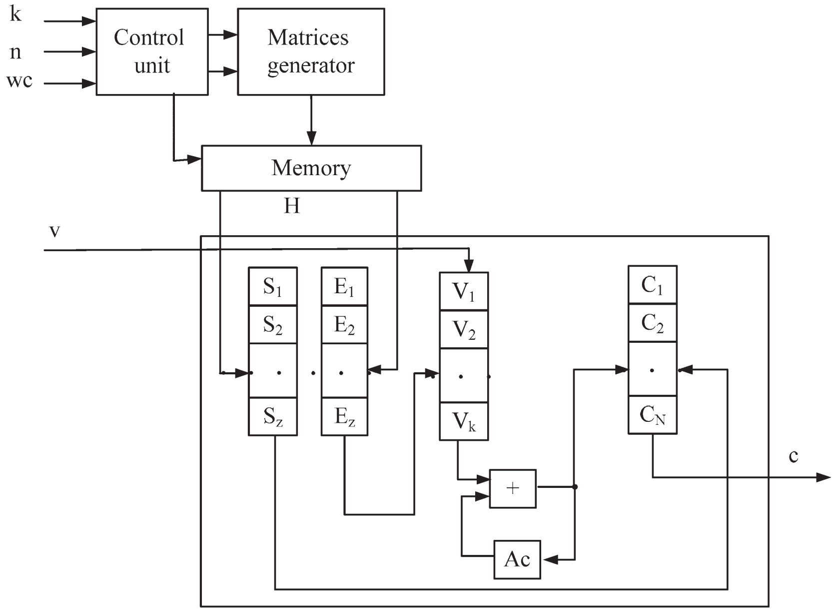

Fig. 3. VLDPC encoder

a wide range of codes with different correction capability.

The control unit connected with memory (RAM), which specifies the address of the desired parity check matrix. In case of absence parity check matrix in memory (an array of memory cells cleared – all cells keep the character «0»), it will generate a new parity check matrix.

A vector can represent the code word:

$$c = [v \mid p], \tag{1}$$

where$v$ – input vector of length k (systematic bits that must be encoded), and p –a check bits vector of length $(n - k)$. Structural parity check matrix H of VLDPC code divided into two sub-matrixes:

$$H = [Hd \mid Hp], \tag{2}$$

where $Hp$is a dual diagonal matrix containing check part (corresponding redundant bits code word) of the form:

$$Hp = \begin{bmatrix} I_d & - & \dots & - \\ - & I_d & \dots & - \\ \dots & \dots & \dots & - \\ - & - & \dots & I_d \end{bmatrix}, \tag{3}$$

where$I_d$is a dual diagonal matrix:

$$I_d = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 1 & 1 \end{bmatrix}. \tag{4}$$

According to

$$c^T H = 0, \tag{5}$$

obtain

$$[Hd \mid Hp][v \mid p]^T = 0, \tag{6}$$

then

$$p = ((inv(Hp)Hd)v, \tag{7}$$

where *inv(Hp)* is the inversion of the matrix.

For *Hd* matrix in the first phase of encoder process square base matrix P_0 will be generated [3–4]. The size m of the matrix P_0 affects to the structure of verification matrix *Hd*. It is possible to manage the code corrective ability by increasing or decreasing the m value. Then permutation matrixes of P_0 must be determined by using cyclic shift right/left operation of the rows/columns. The operation is repeated $m - 1$ times, giving a $P_1, \dots, P_{m-1}$ shifted matrixes. The next step – placing shift matrices in *Hd* [4]. To control the code rate r it can be used the schemes of parity matrixes H transformation (lengthening, shortening, puncturing) [5].

For efficient memory storing and accelerating encoder process in (7) and then in (1), matrix H storage can be organized through two arrays $S_{1\dots z}$ and $E_{1\dots z}$ size z called «string» and «end», where z is total number of entries in the matrix H (Fig. 3). It will store the locations of ones instead of store the whole matrix directly [6]. The array $S_{1\dots z}$ determined the location of ones in each row. The array $E_{1\dots z}$ indicating the end of a row. For example, if

$$\acute{I}\, d = \begin{bmatrix} 1 & & & 1 & \\ & 1 & & & 1 \\ & & 1 & & 1 \\ 1 & & & 1 & \\ & 1 & 1 & & \end{bmatrix} \qquad (8)$$

it will be stored as shown in Table 3.

TABLE 3. **The Storage Of S and E Arrays In Memory**

Address (z = 10)	1	2	3	4	5	6	7	8	9	10
$S_{1..z}$	0	3	1	4	2	4	0	3	1	2
$E_{1..z}$	0	1	0	1	0	1	0	1	0	1

Together $S_{1...z}$ and $E_{1...z}$ determine the bits involved in multiplication (7) with input array $V_{1...k}$ and the locations in the code vector c stored in array $C_{1...n}$ as shown in Fig. 3. The described solution will increase the encoder functionality and flexibility.

Decoding mode based on a probabilistic method with belief propagation. It detailed description is given in [7].

Simulations Results

The encoding/decoding apparatus was implemented using C# application.

Fig. 4 displays the performance of three codes, VLDPC, permanent LDPC (16200, 14400, 16) code and LDPC with variable code rate (code rate was decreasing with increasing SNR and has following values r = 8/9, 7/9, 2/3, 4/9, 3/5, 2/5, 1/3).

The parameter of VLDPC code changedepending of channel quality and in accordance

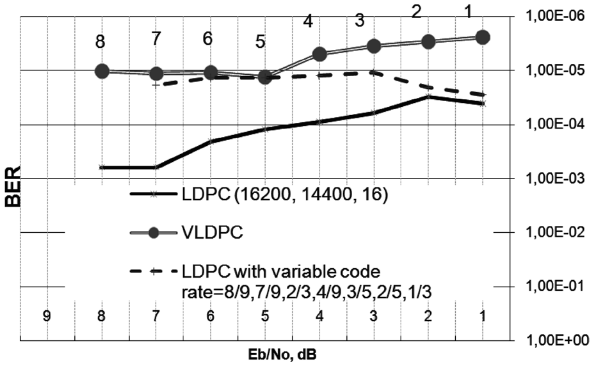

Fig. 4. VLDPC vs LDPCs

with Table 2. The marked dots (1–8) in Fig. 4 determine the values of code parameters n, k, wc:

TABLE 4. **VLDPC Code Parameters During Simulation**

№	1	2	3	4	5	6	7	8
n	16200	16200	16200	16200	64800	64800	64800	64800
k	14400	10800	9720	5400	58320	58320	43200	32400
wc	64	16	64	64	64	128	64	128
r	8/9	2/3	3/5	1/3	9/8	9/8	2/3	1/2
dir.	$\uparrow wc$	$\downarrow r \downarrow wc$	$\downarrow r \uparrow wc$	$\downarrow r$	$\uparrow n \uparrow r$	$\uparrow wc$	$\downarrow r \downarrow wc$	$\downarrow r \uparrow wc$

There are another roe in Table 4, which show direction of code optimization.

Conclusion

As shown, the suggested method performs well result in comparison with the permanent LDPC and variable code rate LDPC-codes. Nevertheless, should bear in mind that codes with low code rate r = 1/3, 1/2 and big wc, for example, 128 increase the decoding time. Therefore, it is necessary for a good channel quality dynamically choose faster LDPC codes.

References

1. **Roca, V.** RFC 5170 on Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes / V. Roca, C. Neumann, D. Furodet.– [Online]. Available: https:// tools. ietf. org/html/rfc5170.

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

3. **Kou, Y.** Low Density Parity Check Codesbased on Finite Geometries: A Rediscover / Kou, Y. S. Lin, R. Fossorier – IEEE Trans. Inform. Theory,2001, Vol.47, P. 2711–2736.

4. **Tanner, R. M.** LDPC block and convolutional codes based on circulant matrices / R. M. Tanner, D. Sridhara, A. Sridharan, D. J. Costello, T. E. Fuja–IEEE Trans. on Inform. Theory, 2004, Vol. 50, № 12, P. 2966–2984.

5. **Пацей, Н. В.** Моделированиепеременныхкодовнизкойплотностипроверокначетность / Н. В. Пацей // ТрудыБГТУ. Сер. VI. Физ.-мат. науки и информатика. Вып. XIX. –Мн.: БГТУ, 2011.– С. 132–135.

6. **Su, Jia-ning.** An efficient low complexity LDPC encoder based on LU factorization with pivoting /Jia-ning Su, Zhi Liu, Ke Liu, Bo Shen, Hao Min –ASIC & System Key Lab, China, 2006.

7. **MacKay, D. J. C.** Information Theory, Inference, and Learning Algorithms/ D. J. C. MacKay– Cambridge University Press, 2003.