# INFORMATION TECHNOLOGY, COMPUTER SCIENCE, AND MANAGEMENT

UDC 519.72; 519.876.5 https://doi.org/10.23947/2687-1653-2021-21-1-96-104

## On the modification of bit-flipping decoder of LDPC-codes

**S. S. Gurskiy, N. S. Mogilevskaya**

Southern Federal University (Rostov-on-Don, Russian Federation)

*Introduction.* In all types of digital communication, error control coding techniques are used. Many digital communication standards, such as Wi-Fi and 5G, use low density parity check (LDPC) codes. These codes are popular because they provide building encoders and decoders with low computational complexity. This work objective is to increase the error correcting capability of the well-known bit-flipping decoder (BF) of LDPC-codes. For this purpose, a modification of the decoder is built, which enables to dynamically control one of its main parameters whose choice affects significantly the quality of decoding.

*Materials and Methods.* The well-known bit-flipping decoder of binary LDPC-codes is considered. This decoder has several parameters that are not rigidly bound with the code parameters. The dependence of the decoding quality on the selection of the output parameters of the bit-flipping decoder was investigated through simulation modeling. It is shown that the decoding results in this case are significantly affected by the input parameter of the decoder — threshold $T$. A modification of the BF-decoder of binary LDPC-codes has been developed, in which it is proposed to set the threshold dynamically during the execution of the algorithm depending on the error rate. A comparative analysis of the error-correcting capability of decoders is carried out by the simulation modeling method.

*Results.* A lemma on the maximum value of the decoder threshold $T$ is formulated and proved. Upper bounds for the number of operations are found for the original and modified decoders. A simulation model that implements a digital noise-immune communication channel has been built. In the model, the initial data is encoded with a given LDPC-code, then it is made noisy by additive uniformly distributed errors, and thereafter, it is decoded in turn by the bit-flipping algorithm with different threshold $T$ parameters, as well as by a modified decoder. Based on the input and output data, the correction capacity of the decoders used is estimated. Experiments have shown that the error-correcting capability of the modified decoder in the range of the real error rate is higher than that of the original decoder, regardless of the selection of its parameters.

*Discussion and Conclusions.* The lemma, proved in the paper, sets the upper bound on the threshold value in the original decoder, which simplifies its adjustment. The developed modification of the decoder has a better error-correcting capability compared to the original decoder. Nevertheless, the complexity of the modification is slightly increased compared to the original algorithm. It has been pointed out that the decoding quality of a modified decoder develops with a decrease in the number of cycles in the Tanner graph and an increase in the length of the code.

***Keywords:*** LDPC-codes, error-correcting capability, dynamic threshold, binary symmetric channel, experimental research.

***For citation:*** S. S. Gurskiy, N. S. Mogilevskaya. On the modification of bit-flipping decoder of LDPC-codes. Advanced Engineering Research, 2021, vol. 21, no. 1, p. 96–104. https://doi.org/10.23947/2687-1653-2021-21-1-96-104

**Introduction.** In 1963, in [1], R. Gallager first described a class of linear block codes whose check matrix contains a small number of nonzero elements. Such codes are commonly referred to as low-density parity check codes, or LDPC codes. For them, it is possible to build encoders and decoders with low computational complexity. Thus, when using LDPC codes, the data transfer rate is not significantly limited. Many modern studies are devoted to LDPC codes and their decoders [2-5]. LDPC codes are widely used in various digital communication standards, such as Wi-Fi, 5G, and optical communication [6, 7]. However, despite the popularity of these codes, some of the problems associated with them require research and solution. One of them is building new decoders and improving the existing ones.

This work objective is to increase the error-correcting capability of the well-known bit-flipping decoder of LDPC codes (hereinafter referred to as the BF decoder). To do this, a modification of the decoder is built, which enables to dynamically control one of its key parameters, whose selection affects significantly the quality of decoding.

**Materials and Methods.** The key parameters of binary LDPC codes are length $N$, dimension $K$ and minimum code distance $d$. The information words $[N, K, d]$ of the $C$-code are vectors $\bar{m} = (m_1, m_2, \dots, m_K) \in F_2^K$, where $F_2$ is the Galois field of cardinality 2, and the codewords are vectors $\bar{c} = (c_1, c_2, \dots, c_N) \in F_2^N$ [8]. It is convenient to set the LDPC codes with the check $(N - K) \times N$ matrix $H$. Most of its elements are zero [1], so it is more convenient to store it not entirely, but storing only the positions of nonzero elements rowwise.

There are regular [9] and irregular [10] LDPC codes. In regular codes, all rows and columns of the check matrices contain a fixed number of single elements ($k$ and $j$, respectively), otherwise the code is called irregular. For convenience, check matrices of regular LDPC codes will be called regular matrices, and irregular LDPC codes — irregular.

Regular LDPC codes have a number of advantages: easily evaluated code parameters, easy storage of matrices, low computational complexity of encoding and decoding algorithms, etc. In addition, regular code decoders correct errors evenly, unlike irregular ones, which correct errors in some parts of the codeword worse than in others. However, the problem of generating regular matrices with given properties is complex, and brute-force methods are often used to solve it.

To discuss the properties of the matrix $H$ it is convenient to use the corresponding Tanner graph $G = (V, E)$, where $E$ — a set of edges, and $V = S \cup R$ — a set of vertices, $S$ — a set of rows of the matrix $H$, and $R$ — a set of its columns [11]. Each nonzero element $H$, standing in the $i$-th row and the $j$-th column, defines an edge connecting the $i$-th vertex of the set $S$ and the $j$-th vertex of the set $R$. Fig. 1 shows an example of a regular check matrix 3×6 with parameters $k = 4$ and $j = 2$, and the corresponding Tanner graph.

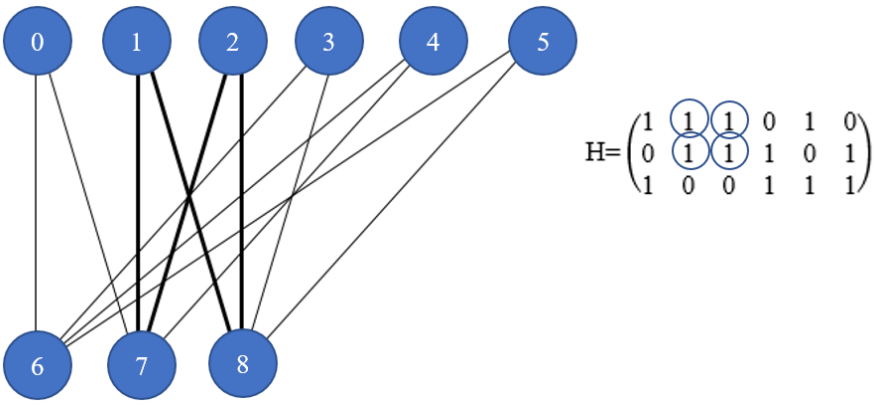


Fig. 1. The cycle in Tanner graph and in the check matrix

The top row of the graph vertices corresponds to the columns of the matrix $H$, and the bottom row is connected to the rows of $H$. An important characteristic of the check matrix $H$ of the LDPC code is the presence and type of cycles in the corresponding Tanner graph. A cycle is a sequence of adjacent vertices of a graph in which the first and last vertices coincide. The length of this sequence is called the cycle length. The minimum cycle length in a graph is called the girth. If the graph contains no cycles, its girth is assumed to be infinite. An example of a cycle of length 4 is highlighted in bold lines in the graph (Fig. 1).

The error-correcting capabilities depend not only on the key parameters of the LDPC codes, but also on the structure of the check matrix $H$. On the one hand, the presence of cycles of small lengths (such as 4 and 6) impairs noticeably the error-correcting capability of the decoder [12]. On the other hand, the code that corresponds to Tanner graph without cycles does not correct errors, since its minimum code distance is 2. Thus, the task of constructing check matrices of regular LDPC codes is multiparametric. When solving it, you need to monitor the key parameters of the code, as well as the cycles in Tanner graph corresponding to the check matrix.

Consider the well-known BF-decoder of the LDPC code $C$ in a convenient form [13].

Input: LDPC code $C$ with parameters $[N, K, d]$, given by the check matrix

$$H = \begin{pmatrix} h_{11} & h_{12} & \dots & h_{1N} \\ h_{21} & h_{22} & \dots & h_{2N} \\ \dots & \dots & \ddots & \dots \\ h_{(N-K)1} & h_{(N-K)2} & \dots & h_{(N-K)N} \end{pmatrix}. \quad (1)$$

Vector $\bar{c}' = \bar{c} + \bar{e}$, $\bar{c} \in C(\subset F_2^N)$, $\bar{e}\,(\in F_2^N)$ — error vector; $p$ — the number of iterations of the algorithm; $T$ — threshold value.

Output: code vector $\bar{c} \in C(\subset F_2^N)$.

Step 1. Let the counter $r$ be equal to zero.

Step 2. Calculate the syndrome $\bar{s} = \bar{c}' H^T$. If $\bar{s} = \bar{0}$ or $r = p$, then go to step 5.

Step 3. Select the unit coordinates from the vector $\bar{s} = (s_1, s_2, \dots, s_{N-K})$, i.e., $s_i = 1$, $i = \overline{1, (N-K)}$. Compose the set $L = \{i | s_i = 1\}$. Calculate $\bar{h}' = (h'_1, h'_2, \dots, h'_N)$, where

$$h'_l = \sum_{i \in L} h_{il}. \quad (2)$$

The values $h_{il}$, $l = 1, \dots, N$ should be assumed to be nonnegative integers. Thus, $\bar{h}' \in \mathbb{N}_0^N$, where $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$.

Step 4. In the vector $\bar{h}' = (h'_1, h'_2, \dots, h'_N)$, we find all the elements $h'_i > T$. Among them, we select random $h'_l$ and invert the bit $c'_l$ of the vector $\bar{c}$. Add a unit to the counter $r$ and go to step 2.

Step 5. $\bar{c} \coloneqq \bar{c}'$.

The research carried out in this work allows us to make some observations on the BF-decoder.

*Observation* 1. The input parameter $p$ sets the maximum number of iterations of the algorithm from the 2nd to the 4th steps, but the decoder can recover the codeword in fewer iterations.

*Observation* 2. When selecting the parameter $T$, the following considerations should be taken into account. If the parameter $d$ of the used $[N, K, d]$-code $C$ is known, then it can be applied to calculate $t$ — the number of reliably recoverable errors, and then the number of decoder iterations is limited to this value:

$$p = t = \left\lfloor \frac{d-1}{2} \right\rfloor. \quad (3)$$

Here, $\lfloor x \rfloor$ — rounding the number $x$ to a smaller integer. If the parameter $d$ is unknown, then it can be estimated using the Singleton bound [5]

$$d \le N - K + 1$$

and, using (3), we obtain

$$p = \left\lfloor \frac{N-K}{2} \right\rfloor.$$

*Observation* 3. The structure of the decoder is such that the recovery of the correct codeword is not guaranteed, even if in the noisy word $\bar{c}' = \bar{c} + \bar{e}$, no more than $t$ errors occurred (3).

*Observation* 4. In the literature, for regular check matrices in the BF decoder, it is recommended to select the threshold $T$ depending on the weight $j$ of the column of the matrix $H$, namely, $T = \frac{j}{2}$. For irregular matrices, such recommendations are not given in the literature. The error-correcting capability of the BF decoder can be worsened by an unsuccessful selection of threshold $T$. If its value is large, at step 4 of the decoder, the vector $\bar{h}'$ may not have a coordinate that exceeds the threshold $T$, therefore, the erroneous bits will not be corrected. If you select a small value of $T$ in step 4 of the BF decoder, several coordinates, whose value exceeds the threshold, may appear in the vector $\bar{h}'$. Among them, there may be coordinates that do not contain an error. Thus, the selection of the parameter $T$ can significantly affect the decoding quality.

**Research Results.** We formulate and prove a lemma on the maximum possible value of the threshold $T$. Then we modify the BF decoder so that the threshold is set dynamically during the decoding process, and conduct a comparative analysis of the original and modified decoding algorithms.

*Lemma*. Let the binary $[N, K, d]$-code $C$ be given by the check matrix $H$ having a fixed number of $j$ unit elements in each column. Then the maximum threshold value $T$ for the BF decoder of such LDPC code $C$ cannot be greater than

$$T = j - 1. \tag{4}$$

Proof. Let the vector $\bar{c}' = \bar{c} + \bar{e}$ be obtained from the transmission channel, where $\bar{c} \in C$ — is the correct codeword, $\bar{e} \in F_2^N$ — the error vector with the Hamming weight $w(\bar{e})$. If $w(\bar{e}) = 0$, then, in step 2, the vector-syndrome $\bar{s} = \bar{0}$. Hence, the algorithm will go to step 5 and return $\bar{c}'$ as the answer. In this case, the threshold value is not used. If $w(\bar{e}) > 0$, then the regularity of $H$ implies the validity of the inequality $h_l' \le j$, where $h_l'$ — the elements of the vector $\bar{h}'$. The inverting of the bit $c_l$ of the vector $\bar{c}'$ occurs in the algorithm only if $h_l' > T$. Therefore,

$$T < h_l' \le j.$$

Thus, the formula (4) is correct.

We will make changes to the BF decoder that will allow us to determine the threshold value dynamically, depending on the degree of damage to the code vector in the transmission channel.

Input: $[N, K, d]$-code $C$ given by the above check matrix (1). Vector $\bar{c}' = \bar{c} + \bar{e}$, where $\bar{c} \in C (\subset F_2^N), \bar{e} (\in F_2^N)$ — the error vector; $p$ — the number of iterations of the algorithm; $T$ — some threshold value selected in advance.

Output: code vector $\bar{c} \in C (\subset F_2^N)$.

Step 1. Let the counter $r$ be equal to zero.

Step 2. Calculate the syndrome $\bar{s} = \bar{c}' H^T$. If $\bar{s} = (0, \dots, 0)$ or $r = p$, then go to step 7.

Step 3. Select the unit coordinates from the vector $\bar{s} = (s_1, s_2, \dots, s_{N-K})$, i.e., $s_i = 1$, $i = \overline{1, (N-K)}$. Compose the set $L = \{i | s_i = 1\}$. Calculate $\bar{h}' = (h_1', h_2', \dots, h_N')$, where $h_l'$ is the same as in the original decoder (2). When summing the value $h_{il}$ $l = 1, \dots, N$, we should assume nonnegative integers. Thus, $\bar{h}' \in \mathbb{N}_0^N$, где $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$.

Step 4. Initialize the threshold value $T \coloneqq \max(h_l')_{l=1,\dots,N} - 1$.

Step 5. If $T \ge 0$

Select an arbitrary element $h_q'$ 'of the vector $\bar{h}'$ — such as $h_q' > T$.

Invert the bit $c_q'$.

Step 6. Add a unit to the counter $r$ and go to step 2.

Step 7. $\bar{c} \coloneqq \bar{c}'$.

*Observation* 5. The modified algorithm generally performs fewer iterations than the BF decoder since the threshold is selected dynamically in step 4. Therefore, the decoder does not perform useless iterations that do not change the bits of the vector $\bar{c}'$. The threshold value in the modified decoder depends on the number of errors that damaged the codeword, and is immediately set so that the noisy codeword $\bar{c}'$ is guaranteed to be changed.

Let us estimate from above the number of addition, comparison and assignment operations in both decoders. In the original BF decoder of the $[N, K, d]$-code $C$, $p(kK + (N-K)N + 1)$ addition operations, $p(3N - 2K + 2)$ comparison operations and $p\big((N-K)(k+1) + 2N + 3\big) + 1$ assignment operations are performed. In the BF decoder with dynamic threshold, $p(kK + (N-K)N + 3)$ addition operations, $p(5N - 2K + 3)$ comparison operations and $p\big((N-K)(k+1) + 2N + 4\big) + 1$ assignment operations are performed. Here, $p$ — the decoder parameter that sets the maximum number of operations, $k$ — weight of the code check matrix rows. Note that when implementing the algorithm, the multiplication and division operations are not actually used, as long as at the second step, it is convenient to use addition operations instead of multiplication to calculate the syndrome $\bar{s}$. Recall that the matrix $H$ has a sparse structure, and its rows are conveniently stored as a list of nonzero element numbers. Therefore, instead of multiplying the vector $\bar{c}'$ by the matrix $H$, it is required to sum the coordinates of the vector $\bar{c}'$, whose numbers coincide with the numbers of nonzero elements in the corresponding row of the matrix $H$.

Compared to the original algorithm, the modified BF decoder performs more operations, but moderately: the number of comparison operations has increased by $p(2N+1)$, assignment operations — by $p$, addition operations — by $2p$.

For a comparative study of the error-correcting capability of the original and modified decoding algorithms, a software tool has been created that implements a simulation model of a binary symmetric perfectly synchronized noise-immune communication channel according to [14-16]. To provide noise immunity, the model uses LDPC codes and BF decoders (original and with dynamic threshold). Errors in the channel were modeled as independent and uniformly distributed.

The experiments used purposely found check matrices that specify LDPC codes. We describe the key parameters of these matrices using the standard notation of the key parameters of the code, as well as: $j$ and $k$ — the weight of each column and the weight of each row of the check matrix, respectively; $\omega_4, \omega_6$ — 4 and 6 cycles in Tanner graph corresponding to the check matrix.

Regular matrix $H_1$: $N = 20, K = 5, j = 3, k = 4, d = 6, \omega_4 = 0, \omega_6 = 41$.

Regular matrix $H_2$: $N = 28, K = 7, j = 3, k = 4, d = 6, \omega_4 = 0, \omega_6 = 42$.

Regular matrix $H_3$: $N = 28, K = 7, j = 3, k = 4, d = 6, \omega_4 = 0, \omega_6 = 29$.

Irregular matrix $H_4$: $N = 32, K = 5, j = 3, d = 12, \omega_4 = 0, \omega_6 = 0$.

Using these matrices, LDPC codes were constructed and simulation experiments were conducted. Fig. 2–5 show graphs of the dependence of the error-correcting capability of the constructed LDPC codes on the error probability in the channel. For the rationale for selecting the threshold values $T = 1$ and $T = 2$ in the BF decoder, see Observations 3, 4 and Lemma.

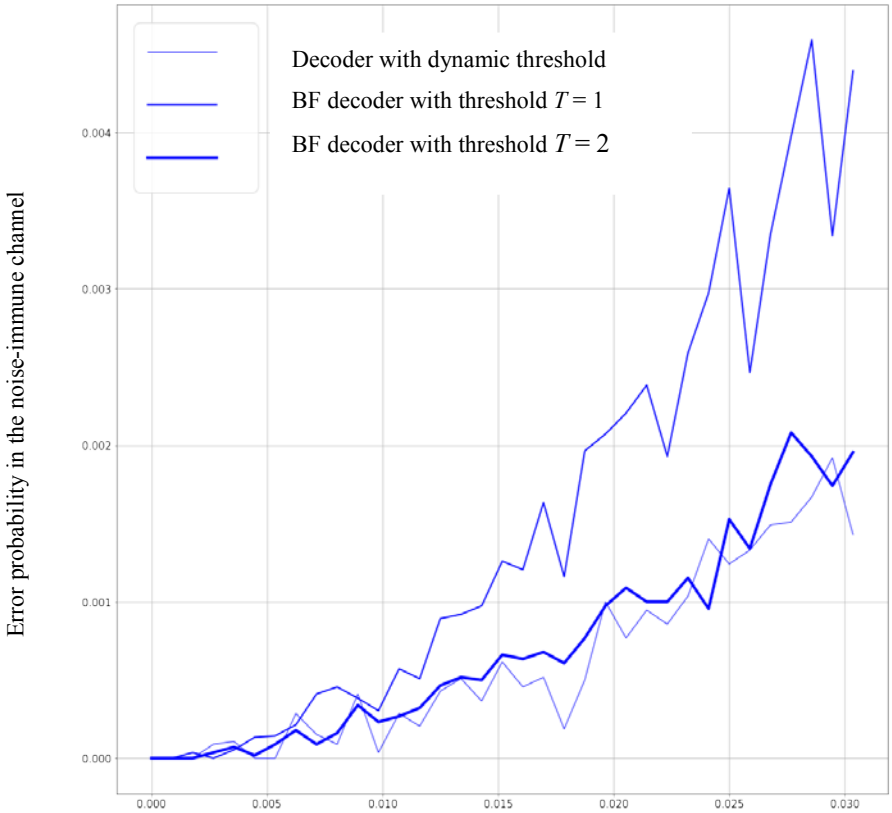


Fig. 2. Graph of the decoder error-correcting capability for LDPC codes given by the matrix $H_2$

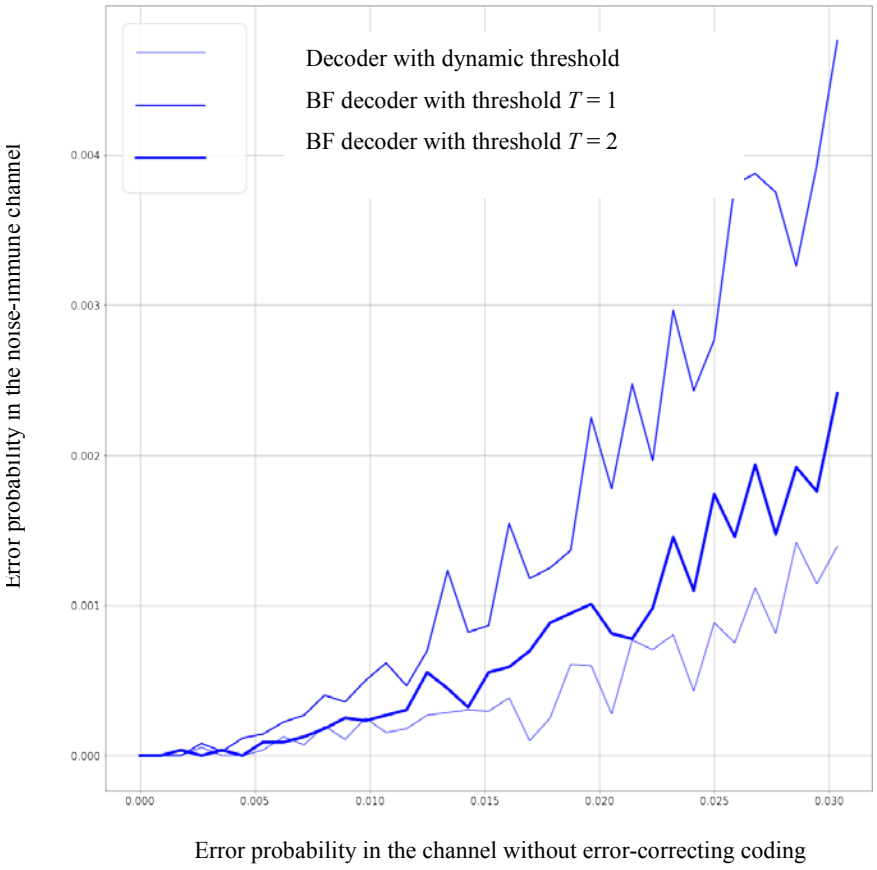


Fig. 3. Graph of the decoder error-correcting capability for LDPC codes given by the matrix $H_3$

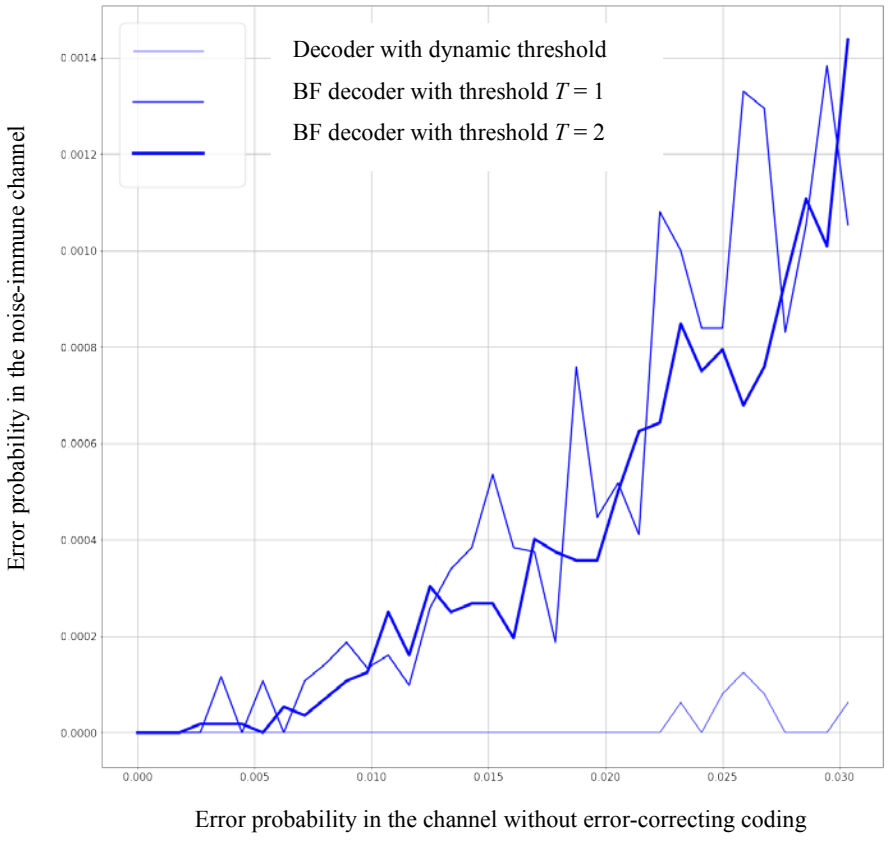


Fig. 4. Graph of the decoder error-correcting capability for LDPC codes given by the matrix $H_4$

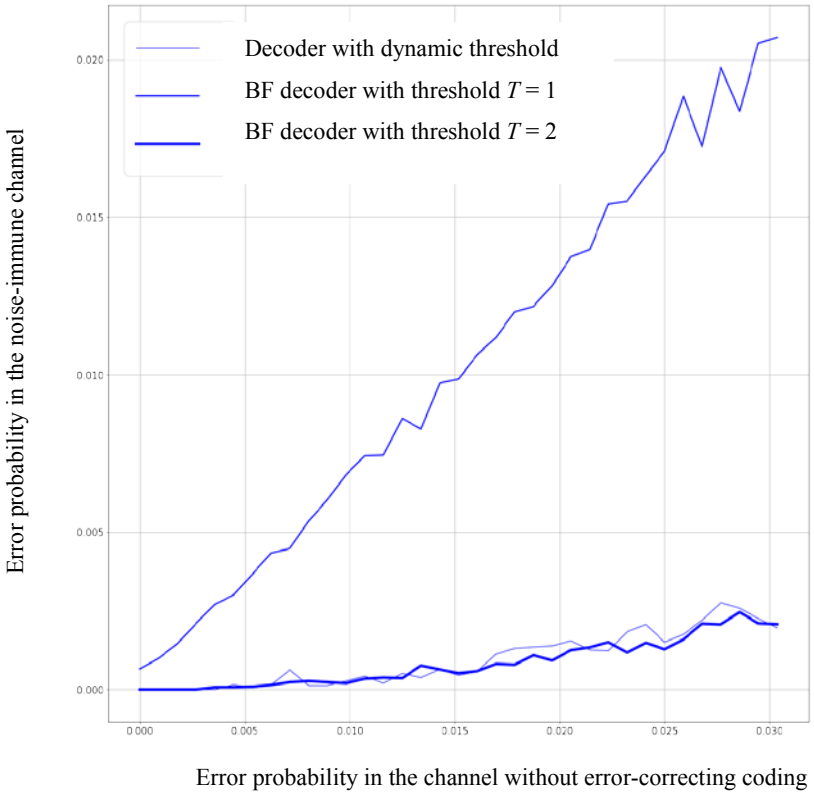


Fig. 5. Graph of the decoder error-correcting capability for LDPC codes given by the matrix $H_1$

In the range of the real error level [8, 13, 14] in Fig. 2–4, it can be observed that the BF decoder at the threshold value $T = 2$ shows better results than at $T = 1$, and the modified BF decoder has a better error-correcting capability compared to the original one.

The decoders show similar efficiency at small values of the code length, but when it is increased, the modified decoder shows better results. Specifically, if the error probability in the non-noise-immune channel is 0.05, the difference in the error probability in the noise-immune channel when using a BF decoder with the threshold $T = 2$ and $T = 1$, is from 0.005 to 0.03 in favor of using a larger threshold value. If a BF decoder with the threshold $T = 2$ and a modified decoder are used, this difference varies depending on the LDPC code in the range from 0.001 to 0.003. If the error probability in the non-noise-immune channel is 0.1, the error probability in the noise-immune channel when using a BF decoder with the threshold $T = 2$ is less than with the threshold $T = 1$ by the value from 0.001 to 0.02. When using a BF decoder with the threshold $T = 2$ and a modified decoder, this difference varies in the range from 0.002 to 0.01depending on the LDPC code.

Both decoders are sensitive to the number of cycles in Tanner graph corresponding to the LDPC code check matrix. The greater the ratio of the number of cycles to the total number of elements in the matrix, the worse any BF decoder corrects errors. During the experiments, it was interesting to find out whether it is possible to increase the number of cycles in the matrix so that the modified decoder will show worse results compared to the BF decoder. Experimentally, the matrix $H_1$ containing 41 cycles of length 6 was found. The results of the study on the error-correcting capability of decoders for this matrix are shown in Fig. 5. Note, however, that the matrix $H_2$ contains even more cycles of length 6, namely, 42. The fundamental difference between the matrices $H_1$ and $H_2$ is in the density of units:

— in $H_1$ — 60 unit elements per 300 matrix elements,

— in $H_2$ — 84 units per 588 matrix elements.

Recall that the feature of LDPC codes is the sparse structure of the check matrix, so $H_2$ is more typical for LDPC codes.

**Discussion and Conclusions.** The paper considers a bit-flipping decoder for binary LDPC codes. Recommendations on the selection of such input parameters of the decoder as the threshold and the number of iterations

of the algorithm are given. A lemma on the maximum value of the decoder threshold is formulated and proved. A modification of the BF decoder of binary LDPC codes has been developed, in which it is proposed to set the threshold dynamically during the execution of the algorithm depending on the resulting syndrome. For the original and modified decoders, upper estimates of the number of operations are found. These estimates show that the modification complicates the decoder only slightly. The conducted simulation experiments demonstrate better error-correcting capability of the modified decoder in relation to the original one. The experiments also showed the dependence of the decoding quality on the degree of matrix sparsity and the number of cycles of length 6 in the Tanner graph corresponding to the check matrix of the LDPC code. Thus, the problem of constructing check matrices with a small number of short cycles arises, which is the subject of further research.

**References**

1. Gallager R. Low-density parity-check codes. IRE Transactions on information theory. 1962;1:21–28.
2. Milicevic M, Feng Ch, Zhang LM, et al. Quasi-cyclic multi-edge LDPC codes for long-distance quantum cryptography. NPJ Quantum Information. 2018;4(1):1–9. DOI: 10.1038/s41534-018-0070-6
3. Chen P, Cai K, Zheng S. Rate-Adaptive Protograph LDPC Codes for Multi-Level-Cell NAND Flash Memory. IEEE Communications Letters. 2018;22(6):1112–1115. DOI: 10.1109/LCOMM.2018.2814985
4. Baldi M, Barenghi A, Chiaraluce F, et al. A Post-quantum Key Encapsulation Mechanism Based on QC-LDPC Codes. Post-Quantum Cryptography. In: PQCrypto 2018: Lecture Notes in Computer Science. Cham: Springer. 2018;10786:3–24. DOI: 10.1007/978-3-319-79063-3_1
5. Maity RK, Singh RA, Mazumdar A. Robust Gradient Descent via Moment Encoding and LDPC Codes. In: IEEE International Symposium on Information Theory (ISIT). Paris: IEEE; 2019. P. 2734–2738. DOI: 10.1109/ISIT.2019.8849514
6. Li H, Bai B, Mu X, et al. Algebra-Assisted Construction of Quasi-Cyclic LDPC Codes for 5G New Radio. IEEE Access. 2018;6:50229–50244. DOI: 10.1109/ACCESS.2018.2868963
7. Cai Z, Hao J, Tan PH, et al. Efficient encoding of IEEE 802.11n LDPC codes. Electronics Letters. 2006;42(25):1471–1472.
8. Kolesnik VD. Kodirovanie pri peredache i khranenii informatsii. [Coding in the transmission and storage of information]. Moscow: Vysshaya shkola; 2009. 550 p. (In Russ.)
9. Tong Zhang, Parhi KK. Joint (3,k)-regular LDPC code and decoder/encoder design. IEEE Transactions on Signal Processing. 2004;52(4):1065–1079. DOI: 10.1109/TSP.2004.823508
10. Yang M, Ryan WE, Yan Li. Design of efficiently encodable moderate-length high-rate irregular LDPC codes. IEEE Transactions on Communications. 2004;52(4):564–571.
11. Malema GA. Low-Density Parity-Check Codes: Construction and Implementation. University of Adelaide; 2007. 160 p. Available from: URL: https://digital.library.adelaide.edu.au/dspace/bitstream/2440/45525/8/02whole.pdf (accessed: 07.06.2020).
12. Etzion T, Trachtenberg A, Vardy A. Which Codes Have Cycle-Free Tanner Graphs? IEEE Transactions on Information Theory. 2006;52(9):4219–4223. DOI: 10.1109/TIT.2006.880060
13. Morelos-Zaragoza R. Iskusstvo pomekhoustoichivogo kodirovaniya. Metody, algoritmy, primenenie [The art of noise-immune coding. Methods, algorithms, and applications]. Moscow: Tekhnosfera; 2006. P. 259–262. (In Russ.)
14. Deundyak VM, Mogilevskaya NS. Metody otsenki primenimosti pomekhoustoichivogo kodirovaniya v kanalakh svyazi [Methods for evaluating the applicability of noise-immune coding in communication channels]. Rostov-on-Don: DSTU Publ. Centre; 2007. 85 p. (In Russ.)
15. Deundyak VM, Zhdanova MA, Mogilevskaya NS. Reshenie zadachi podbora modeli istochnika oshibok v IS OPSAPK [Solution to error source model selection problem in IS EASECC]. Vestnik of DSTU. 2017;17(4):107–115. DOI: 10.23947/1992-5980-2017-17-4-107-115 (In Russ.)

16. Deundyak VM, Mogilevskaya NS. Imitatsionnaya model' tsifrovogo kanala peredachi dannykh i algebraicheskie metody pomekhoustoichivogo kodirovaniya [The simulation model of digital channel of data transmission and algebraic methods of error-correcting coding]. Vestnik of DSTU. 2001;1(1):98–104. (In Russ.)

Submitted 26.12.2020
Scheduled in the issue 25.01.2021

*About the Authors:*

**Gurskiy, Semen S.,** undergraduate student of the Algebra and Discrete Mathematics Department, Southern Federal University (8a, Milchakova St., Rostov-on-Don, 344090, RF), ORCID: https://orcid.org/0000-0002-4738-2363, nor-ber@list.ru

**Mogilevskaya, Nadezhda S**., associative professor of the Algebra and Discrete Mathematics Department, Southern Federal University (8a, Milchakova St., Rostov-on-Don, 344090, RF), Cand.Sci. (Eng.), associate professor, ORCID: http://orcid.org/0000-0003-1357-5869, nadezhda.mogilevskaia@yandex.ru

*Claimed contributorship*

S. S. Gurskiy: modification of the bit-flipping decoder; software implementation of channel models; conducting computational experiments; text preparation. N. S. Mogilevskaya: academic advising; task formulation; analysis of the research results; the text revision; formulation of conclusions.

*All authors have read and approved the final manuscript.*