Algorithm of Multithreshold Decoding for Non-Binary Self-Orthogonal Concatenated Codes

Valery Zolotarev
Institute of Space Researches
Russian Academy of Sciences
Moscow, Russian Federation
zolotasd@yandex.ru

Gennady Ovechkin
Department of Computational and Applied Mathematics
Ryazan State Radio engineering University,
Ryazan, Russian Federation
g_ovechkin@mail.ru

Yerzhan Seitkulov
The Institute of Information Security and Cryptology
L.N. Gumilyov Eurasian National University
Astana, Republic of Kazakhstan
seitkulov_y@enu.kz

Dina Satybaldina, Nurlan Tashatov, Vitaly Mishin
Department of Information Technologies
L.N. Gumilyov Eurasian National University
Astana, Republic of Kazakhstan
satybaldina_dzh@enu.kz, tash.nur@ mail.ru

***Abstract*—Non-binary multithreshold decoding (qMTD) for q-ary self-orthogonal codes (qSOC) is considered. The SER performance of qMTD is shown to be close to the results provided by optimum total search methods, which are not realizable for non-binary codes in general. qMTD decoders are compared with different decoders for Reed-Solomon codes. The performance provided with qMTD in some cases is unattainable with classical decoders for arbitrary long Reed-Solomon codes. The result of concatenation of qSOC with simple to decode outer codes is described. Method of improving of qMTD decoder's performance for qSOC is offered. Some simulated results obtained by using these two decoding techniques (the base and modified ways) are presented as well. Comparison of the results showed that the change in the threshold element's algorithm can significantly improve speed of qMTD work. It's shown that for a larger gain this modification qMTD should be used after conventional decoding iterations.**

***Index Terms*— Communication, error-correction coding, multithreshold decoding, non-binary codes, concatenated codes.**

I. Introduction

In his seminal paper in 1948, Shannon formalized the communications problem and showed that it was possible to encode messages in such a way that the number of extra bits transmitted was as small as possible [1]. The value of error-correcting codes for information transmission, both on Earth and from space, was immediately apparent, and a wide variety of codes were constructed which achieved both economy of transmission and error-correction capacity.

Convolutional codes with Viterbi decoders [2], turbo codes [3], low-density parity-check codes [4] and other codes are used in modern communication systems now. However, these codes are still very complex for decoding or inefficient. Below we shall consider high performance and very simple iterative decoder which one is the evolution of threshold decoder (TD) [5] for linear convolutional or block codes.

TD is used for decoding of self-orthogonal codes [6]. This decoder implements one of the least complex decoding methods, but its error correcting ability is weak. To improve the performance of TD many authors in the seventies of the twentieth century introduced several schemes of repeated decoding with TD. However, these schemes were inefficient due to essential error grouping at the decoder output. This problem was solved later in the development process of new method which is called multithreshold decoding (MTD) [7–10]. It allows to build software MTD decoders which are about hundreds times faster than other decoding algorithms comparable on performance [9]. Hardware MTD versions implemented on simple Xilinx or Altera FPGAs show practically unlimited throughput even in case of data transmission through high-speed channels with large noise level [10].

In some systems, it is convenient to work with data having a byte structure. Until recently there were no effective and simultaneously enough simple decoding methods for non-binary (symbolic) codes, except decoders for Reed-Solomon (RS) codes. However short RS codes of length up to n=255 symbols do not provide levels of reliability necessary nowadays. Decoders for long RS codes appear to be too complex and their essential simplification is rather problematic. Their real correcting possibilities are also very restricted. Recently many experts began to develop decoders for q-ary low-density parity-check (qLDPC) codes [11, 12]. The given methods, certainly, possess very high performance. However, complexity of their decoders, especially at large alphabet size q, appears to be too high for practical application.

In fact, J.Massey considered these codes and proved Theorems 1÷4 for these codes in [5]. But then he spoke negatively about these codes possibilities in sections 1.2, 6.2, 6.5, 6.6 and 8.2 of the same book and no longer engaged in the topic.

The generalization of MTD for q-ary symmetric channels (qSC) was offered in [13, 14]. The value of this method shows that the majority algorithms provide almost optimal performance and have only linear computational complexity, as usually optimum methods are characterized by exponential complexity. Therefore, the application of q-ary MTD (qMTD) can be especially useful.

In present paper some new important qMTD properties are discussed. The other parts of the paper are arranged in the following way. Section II gives the concept of the q-ary multithreshold decoding. The method of improving of qMTD decoder's performance for self-orthogonal codes is considered in Section III. Simulation results for two decoding techniques (the base and modified ways) are shown in Section IV. Section V shows the main conclusions of the paper

II. THE NON-BINARY MULTITHRESHOLD DECODING BACKGROUND

Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts. True Type 1 or Open Type fonts are required. Please embed all fonts, in particular symbol fonts, as well, for math, etc.

Let consider usual q-ary, $q > 2$, symmetrical channel (qSC) with an error probability $p_s > 0$, when a transmission any initial character of a code transforms it to one of stayed (q-1) characters incidentally, separately and with equiprobability. For such a channel the optimum decoder solution will be such, probably, unique code word among q^{nR} possible ones, which word differs from the received word in a minimum number of code characters. (Here it was supposed, that n - code length expressed by a number of a code characters, R - code rate, $R=k/n < 1$).

Let it be further a linear non-binary code, which check matrix H has the same view, as well as in a binary case, i.e. it consists of zeroes and ones. Let this matrix corresponds to self-orthogonal systematic block or convolutional code (SOCC). In this case code words of minimum weight d, where d - is a minimum code distance, have an alone non-zero character i_k, with value q_i, $q_i > 0$, in its information part. A generating matrix G contain only zeroes and ones, the operations of the encoder and decoder with checking characters of a code formation and calculation of a syndrome S in the received word are only addings. Thus, coding and decoding do not need processing in non-binary fields or in rings for integers. It is only enough to arrange integer group. It essentially simplifies principally all coding procedures and subsequent decoding.

The example of a scheme realizing the operation of encoding by block qSOC is given on Fig. 1. Such code is characterized by the parameters: code length n=26 symbols, data part length k=13 symbols, code rate R=1/2, code distance d=5.

Let's assume that encoder has performed encoding of data vector U and received code vector $A = [U, V]$, where $V= U\cdot G$. Note that in this example and below when multiplication, addition, subtraction of vectors and matrices are made, module

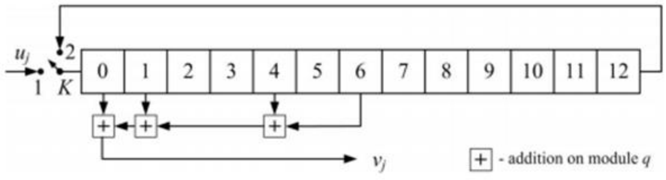

Fig. 1. Encoder block SOCC with R = 1/2 and k = 13.

arithmetic is applied. When code vector A having the length n with k data symbols on qSC is transmitted decoder is entered with vector Q, generally speaking, having differences from original code vector due to errors in the channel: $Q=A + E$, where E – channel error vector of qSC type.

Operating algorithm of qMTD during vector Q decoding is the following [10].

1. Syndrome vector is calculated $S = \mathrm{H}\cdot\mathrm{Q}^{\mathrm{T}}$. Difference register D is reset. This register will contain data symbols changed by decoder. Note that the number of nonzero elements of D and S vectors will always determine the distance between message Q received from the channel and code word being the current solution of qMTD. The task of decoder is to find such code word which demands minimal number of nonzero elements of D and S vectors. This step totally corresponds to binary case.

2. For arbitrarily chosen decoded q-ary data symbol i_j of the received message let's count the number of two most frequent values of checks s_j of syndrome vector S from total number of all checks relating to symbol i_j, and symbol d_j of D vector, corresponding to i_j symbol. Let the values of these two checks be equal to h_0 and h_1, and their number be equal to m_0 and m_1 correspondingly when $m_0 \geq m_1$.

This step is an analogue of sum reception procedure on a threshold element in binary MTD.

3. If $m_0–m_1 > T$, where T – a value of a threshold (some integer number), then from i_j, d_j and all checks regarding i_j error estimation equal to h_0 is subtracted. This step is analogous to comparison of a sum with a threshold in binary decoder and change of decoded symbol and correction via feedback of all syndrome symbols being the checks for decoded symbol.

4. The choice of new i_m, $m \neq j$ is made, next step is clause 2.

Such attempts of decoding according to cl. 2…4 can be repeated for each symbol of received message several times [10].

The example of qMTD implementation for encoder from Fig. 1 is given on Fig. 2.

III. QMTD PERFORMANCE

The SER performance of decoders for codes of rate R=1/2 over qSC is shown in fig. 3. On the horizontal axis channel SER P_0 is presented and on the vertical axis average SER after decoding is shown. Here curves 4 and 5 correspond to qMTD for codes of length n=4000 and n=32000 one-byte symbols (q=256) accordingly.

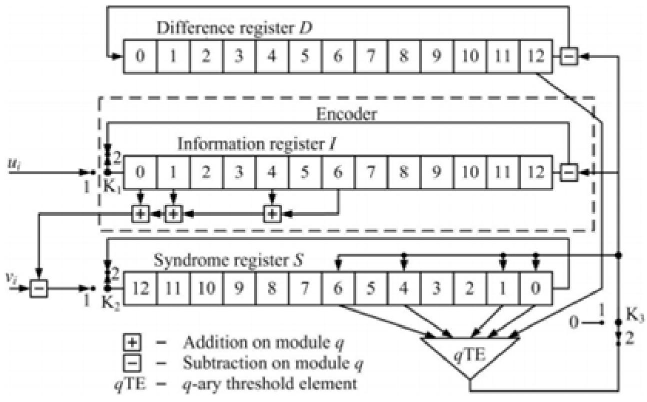

Fig. 2. qMTD for block qSOCC.

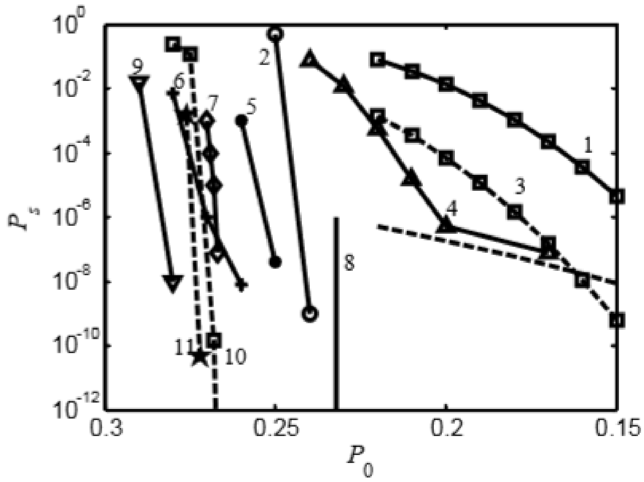

Fig. 3. SER performance of rate one-half RS codes and qMTD over qSC.

Dotted line with no marker in fig. 3 shows the lower bound P_{OD} on the symbol error probability of OD (optimum decoder) for the first code. It's seen that qMTD can achieve OD performance at high noise level. To achieve the optimum decision or to get close to it, qMTD for code with q = 256 requires from 5 up to 20 decoding iterations. It completely corresponds to MTD for binary codes [7-10]. For comparison in fig. 3 the performance of (255, 128) RS code over GF(2^8) is also shown by curve 1. As it follows from fig. 3 qMTD provides much better performance than decoder for RS code with symbols of the same size due to greater length of used codes and to good qMTD decisions convergence to the OD decisions.

It should be noted the complexity of known methods for increasing correcting ability of RS codes as Sudan algorithm and others is proportional n^3. This leads to the difference in complexity with qMTD in 10^9 times for codes of length about 30000 symbols. And performance improvement for RS codes decoding in this case is insignificant. It's shown by curve 3 in fig. 3 which corresponds to Sudan decoder performance for (255, 128) RS code. Further we shall describe simulation results for codes with larger alphabet size q. The performance of qMTD for codes with R=1/2, n=32000 and q=2^{16} (two-byte symbols) is presented in fig. 3 by curve 6. We note that a very simple for implementation qMTD for the code of length 32000 symbols appears to be capable to provide error correcting ability essentially unattainable even for RS code of length 65535 over GF(2^{16}) (curve 2 in fig. 3), a decoder for which is too complex for implementation. Thus qMTD for two-byte symbols practically is not more complex than one-byte one as even usual microprocessors simply and quickly work and with one-byte symbols, and with 2 and even with 4-byte symbols. For example the performance of qMTD for code with four-byte symbols (q=232) is shown in fig. 3 by curve 9. Note that P_0 on the horizontal axis is error probability in one-byte, two-byte or four-byte symbols for different codes. It should be noted that other decoding algorithms with acceptable complexity besides qMTD which can provide the same performance are unknown now.

Note that for estimating of such low SER decoding about 10^{11} or even more q-ary symbols was performed. It was possible due to very low complexity of qMTD which software versions work with rate about 10^{11} symbols per hour.

For communication and data storage systems due to different restrictions high-rate q-ary codes are very useful. The performance of qMTD for codes with R=7/8, n=48000 symbols and q=256 is shown in fig. 4 by curve 3 and performance of decoders for RS code with R=7/8 over GF(2^8) is presented by curve 1. It's seen that here qMTD outperform decoders for RS codes significantly. Similar relation between performance of these error correction methods remains at using higher code rate R=19/20.

For this code rate the performance of qMTD for codes with q=256 is shown in fig. 4 by curve 4 and curve 2 presents the performance of decoders for RS code over GF(2^8). In this case qMTD is much more effective than RS codes decoder too. Comparing decoders for RS codes of length n = 255, R = 7/8 and R = 19/20 it is clear that the latter code is much less effective then the former one and it is much more difficult to provide good efficiency at redundancy reduction. Nevertheless the performance of low redundancy codes with qMTD decoding appears rather high and can essentially increase error correcting ability if the chosen codes have enough large lengths. The performance of qMTD for code with two-byte symbols and R=7/8 is shown in fig. 4 by curve 5.

It should be noted that to achieve such results with qMTD it is necessary to select used codes carefully. The main criterion at codes searching is their resistance to error- propagation effect [10]. For illustration of the statement in fig. 3 by curve 7 and in fig. 4 by curve 6 the performance of qMTD for codes with q=256, R=1/2 and R=7/8 is presented accordingly. The applied codes were selected even more carefully than before.

Let's consider simulation results for several concatenated schemes based on qMTD. In [10, 13] it's shown the using outer single modulo q check code with qSOC allows to improve SER performance on 1..3 decimal orders at about 2% redundancy increasing.

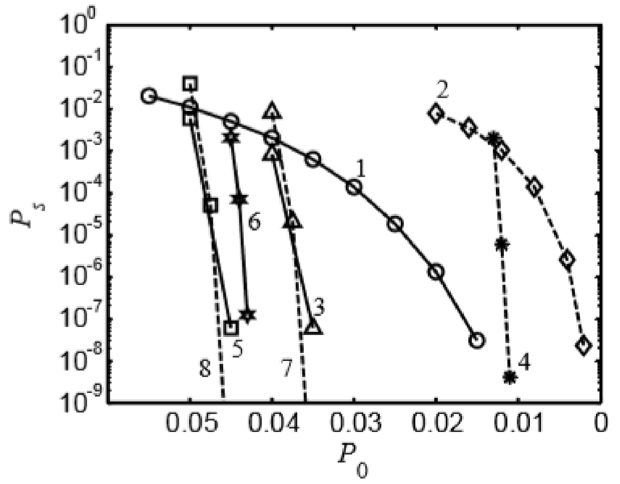

Fig. 4. SER performance of high-rate RS codes and *q*MTD over *q*SC.

Complexity increasing in this case is 20% only in comparison with base *q*MTD. SER performance of concatenated codes consisting of a *q*SOC and code with single modulo *q* check is shown in fig. 4 [10]. Here SER for concatenated codes with an internal *q*SOC of rate R = 7/8 and an external code with single modulo q check of length L = 190 are presented by curve 7 for q = 256 (one-byte symbols) and by curve 8 for q = 216 (two-byte symbols).

Other codes suitable for concatenation with *q*SOC are modified *q*-ary Hamming codes [10]. Such codes have less length in comparison with known *q*-ary Hamming codes and have no restriction on alphabet size *q*. For example performance of such scheme is presented in fig. 3 by curve 10. It's shown the using of modified Hamming codes as outer code provides *q*MTD SER performance improving on 3..5 decimal orders. Later improvement of *q*MTD performance is possible with using outer high rate *q*SOC in concatenated scheme [15]. This outer *q*SOC is decoded with other qMTD which one can to use information about reliability of decoding symbols generated with *q*MTD of internal code. The performance example for such scheme is shown in fig. 3 by curve 11. This scheme allows to provide SER about 10-10 at 27.5% symbol channel errors. Such result is unattainable for other described codes.

IV. Algorithm of Work Acceleration of a Threshold Element of Non- Binary Multithreshold Decoder

In the analysis of non-binary MTD scheme shown in Fig. 2, it was shown that *q*MTD is a device consisting only of shift registers, adders, subtractors modulo *q* and non-binary threshold element (*q*TE). Among the most difficult elements of the decoder has its threshold element. Therefore, in order to speed up *q*MTD, you need to speed up *q*TE.

In the course of research work *q*MTD it was noticed that often on adjacent iterations of decoding for one information symbol does not change the information delivered to the threshold element. It is therefore proposed to modify the work *q*TE so that in the decoding process if possible, the values of calculated threshold elements of previous iterations, and only in case of necessity make their conversion . On the basis of this idea has been developed *q*MTD modified scheme shown in Fig. 5.

Note that in this scheme will charge an extra register, the elements of which indicate whether reprocess the information that comes to the threshold element with the syndrome and the difference registers. In this case, since the various iterations may change the threshold *q*TE, you need to remember the value of the difference m_0-m_1 and the correction value h_0, which is used in triggering threshold element.

Each element of the flags recalculation register may contain only two values: 0 or 1. In a first iteration of the decoding register is filled with signs of translation units.

The procedure for decoding a received message *q*MTD is modified as follows:

1. Selected at random and the decoded information symbol i_j received message.

2. If the element of the flags recalculation register corresponding to the information symbol i_j, is 1, are calculated by the two most frequent inspections. These two values of inspections are h_0 and h_1, and their number m_0 and m_1 power respectively, wherein $m_0 \geq m_1$. If element of the flags recalculation register is 0, the difference value m_0-m_1 is set equal to the threshold value of the current register and the value h_0 - value of the current element of the corrections register.

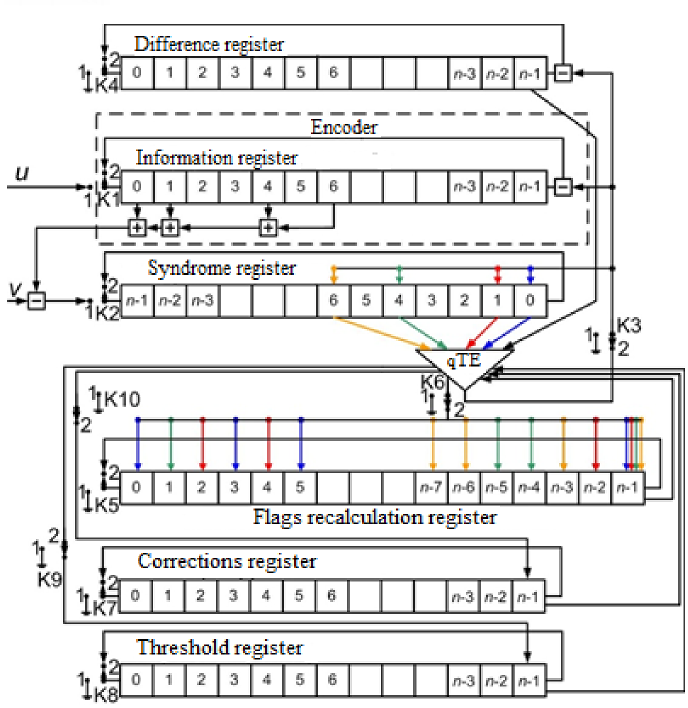

Fig. 5. The modified non-binary block *q*MTD SOCC with R = 1/2, d = 5.

3. If m_0-m_1≤T, is set to 0 the value of element of the flags recalculation register of the corresponding information symbol i_j, in the current element threshold register is stored difference m_0-m_1, a current member register offsets - the value of h0. If m_0-m_1> T, then the i_j, d_j and all checks concerning i_j subtracted estimate of error, which is equal to h_0. Are also set to 1, all the elements of the flags recalculation register, the corresponding data symbols are involved in the formation of modified character of the syndrome register.

4. The transition to the new arbitrary i_m, $m \neq j$ and then go to step 2.

In p. 3 $(d - 1)^2 \bullet nk / nr$ items need only for change the flags recalculation register in the case of correction of each information symbol, where *nk* and *nr* - number of information and verification of branches encoder, respectively.

Table 1 shows a comparison of time decoding the information and common-modified qMTD. To decode unused qSOCC with R = 7/8 and d = 7, the error probability in the channel formed le and P_0 = 0.04 and P_0 = 0.001. The amount of information to decode the 10^8 byte characters.

TABLE I. Time of Decoding the Information qMTD

The error probability in the channel	**qMTD**	**Modification of qMTD**
P_0=0.04	16 MINUTES	8 MINUTES
P_0=0.001	15 MINUTES	5 MINUTES

As a result of the using of the modified algorithm of qPE performance of decoding has increased in 2 times at P_0 = 0.04 and 3 times at P_0 = 0.001. Note that using this modification of qPE decoding performance as compared with conventional qPE not reduced. Note that for a larger gain in terms of transactions this modification qMTD in some cases, should be used after conventional decoding iterations

V. CONCLUSION

The efficiency of *q*MTD algorithms in SER and in complexity is in many times better than the efficiency of decoders for Reed-Solomon codes. This is proved with the effective transfer of binary multithreshold decoding ideas on very simply organized non-binary codes of any big length. Other codes and decoding algorithms with similar complexity and error correction ability are not known nowadays.

In our paper new important results on the modification *q*TE are obtained. It's shown that in *q*-ary symmetric channel characteristics modified *q*MTD can provide a performance in 2 ÷ 3 time better than the base *q*MTD. The use of such *q*MTD to correct errors in the systems memory byte data structure can improve the speed of multiple versions of software algorithms for encoding and decoding in the implementation of special versions of decoders with fast threshold elements.

Thus, this level of error correcting ability achieved with different *q*MTD algorithms allows solving problems of high reliability maintenance for transmission and data storage without any additional completion of these algorithms or only in the process of their insignificant adaptation to the possible additional requirements arising in large-scale digital systems.

ACKNOWLEDGMENT

This work was supported by the Russian fund of fundamental researches (grant No. 12-07-00418), the grant of the President of the Russian Federation (grant MD-639.2014.9) and the Science Committee of Ministry of Education and Science of the Kazakhstan Republic (grant No. 144-04.02.2014).

REFERENCES

[1] C.E. Shannon (1948, July/October). A Mathematical Theory of Communication. Bell System Technical Journal. 27 (3):379–423.

[2] A.J. Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm, IEEE Trans., 1967, IT-13, pp.260–269.

[3] C. Berrou, A. Glavieux A., P. Thitimajshima, Near Shannon Limit Error- Correcting Coding and Decoding: Turbo-Codes, in Proc. ICC'93, Geneva, 1993, pp.1064–1070.

[4] D.J.C. MacKay, R.M. Neal, Near Shannon limit performance of low-density parity check codes, IEEE Electronics Letters, 1996, vol.32, no.18, pp.1645–1646.

[5] J. Massey, Threshold decoding, M.I.T. Press, Cambridge, Massachusetts, 1963.

[6] R.L. Townsend, E.J. Weldon, Self-Orthogonal Quasi-Cyclic Codes. IEEE Trans., 1967, IT-13, pp.183–195.

[7] V.V. Zolotarev, Way of a noiseproof code decoding. Invention patent №2377722 of 2009 with priority at 21 June, 2007.

[8] V.V. Zolotarev, “The Multithreshold Decoder Performance in Gaussian Channels”. In Proc. 7-th International Symposium on Communication Theory and applications, 2003, Ambleside, UK, pp.18-22.

[9] V.V. Zolotarev, Theory and algorithms of multithreshold decoding. Moscow: Radio and communication, Hot Line - Telecom, 2006, 270 p. (in Russian).

[10] V.V. Zolotaryov, J.B. Zubarev, G.V. Ovechkin, Multithreshold decoders and optimization theory of the coding. Hotline - Telecom, Moscow, 2012 (in Russian).

[11] D.Declercq, M.Fossorier, “Extended minsum algorithm for decoding LDPC codes over GF (q)”, In Proc. IEEE International Symp. on Inf. Theory, 2005, pp.464-468.

[12] F. Zhang, H. Pfister, “List-Message Passing Achieves Capacity on the q-ary Symmetric Channel for Large q”, In Proc. IEEE Global Telecom. Conf., Washington, 2007. pp.283–287.

[13] V.V. Zolotarev Generalization of MTD algorithm at non-binary codes. Mobile systems, Moscow, 2007, №3, pp.39-42 (in Russian).

[14] V.V. Zolotarev, G.V. Ovechkin, S.V. Averin, “Algorithm of multithreshold decoding for self-orthogonal codes over Gaussian channels”, in Proc. 11-th ISCTA’09, July, UK, Ambleside, 2009.

V.V. Zolotarev, G.V. Ovechkin, “Concatenated methods for performance improvement of multithreshold decoders for non-binary codes”, In Proc. 14-th DSPA. Moscow, 2012