UDC 004.056.5

# MODIFIED ALGORITHM FOR SEARCHING THE ROOTS OF THE ERROR LOCATORS POLYNOMINAL WHILE DECODING BCH CODES

**Krylova V. A.** – PhD, Associate Professor of the Department of automation and control in technical systems, National Technical University «Kharkiv Polytechnic Institute», Kharkov, Ukraine.

**Tverytnykova E. E.** – Dr. Sc., Professor of the Department of information and measuring technologies and systems, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine.

**Vasylchenkov O. G.** – PhD, Associate Professor of the Department of automation and control in technical systems, National Technical University «Kharkiv Polytechnic Institute», Kharkov, Ukraine.

**Kolisnyk T. P. –** PhD, Associate Professor of the Department of information technology and cybersecurity, Kharkiv National University of Internal Affairs, Kharkov, Ukraine.

## ABSTRACT

**Context.** In telecommunications and information systems with an increased noise component the noise-resistant cyclic BCH and Reed-Solomon codes are used. The adjustment and correcting errors in a message require some effective decoding methods. One of the stages in the procedure of decoding RS and BCH codes to determine the position of distortions is the search for the roots of the error locator polynomial. The calculation of polynomial roots, especially for codes with significant correction capacity is a laborious task requiring high computational complexity. That is why the improvement of BCH and RS codes decoding methods providing to reduce the computational complexity is an urgent task.

**Objective.** The investigation and synthesis of the accelerated roots search algorithm of the error locator polynomial presented as an affine polynomial with coefficients in the finite fields, which allows accelerating the process of BCH and RS code decoding.

**Method.** The classical roots search method based on the Chan's algorithm is performed using the arithmetic of the Galois finite fields and the laborious calculation, in this case depends on the number of addition and multiplication operations. For linearized polynomials, the roots search procedure based on binary arithmetic is performed taking into account the values obtained at the previous stages of the calculation, which provides the minimum number of arithmetic operations.

**Results.** An accelerated algorithm for calculating the values of the error locator polynomial at all points of the GF($2^m$) finite field for linearized polynomials based on the Berlekamp-Massey method has been developed. The algorithm contains a minimum number of addition operations, due to the use at each stage of the calculations the values obtained at the previous step, as well as the addition in the finite field GF(2). A modified roots search method for affine polynomials over the finite fields has been proposed to determine error positions in the code word while decoding the cyclic BCH and RS codes.

**Conclusions.** The scientific newness of the work is to improve the algorithm of calculating the roots of the error locator polynomial, which coefficients belong to the elements of the finite field. At the same time it simplifies the procedure for cyclic BCH and RS codes decoding, due to reducing the computational complexity of one of the decoding stages, especially finding the error positions using the modified Berlekamp-Massey algorithm. These facts are confirmed by the simulation program results of the roots search of the error locator polynomial algorithm. It is shown, that the application of the accelerated method permits to reach a gain on speed of 1.5 times.

**KEYWORDS:** BCH codes, error locator polynomial, Chan's search, Berlekamp-Massey algorithm and Reed-Solomon codes.

## ABBREVIATIONS

BM is a Berlekamp-Massey;
BCH is a Bose-Chaudhuri-Hocquenghem;
RS is a Reed-Solomon.

## NOMENCLATURE

*A* is a binary matrix of linearized polynomials;
*F*(*x*) is a linearized polynomial;
$f_i$ is a coefficient to the GF($2^m$) finite field;
GF($2^m$) is a finite field Galois;
*l* is a field order;
*m* is a natural number;
*p*(*x*) is a generating polynomial;
*t* is a times;
*Y* is a binary vector of zero coefficient of the error locator polynomial;
ν is a degree of the errors locators polynomial;
$\alpha^i$ is an element of GF($2^m$) finite field;
$\beta^i$ is a binary vector field element α;
σ(*x*) is an errors locators polynomial;
$\sigma_0$ is a zero coefficient of locators of errors;
$\sigma_i$ is an element of the finite field GF($2^m$).

## INTRODUCTION

One of the ways to protect information from errors in digital communication systems is to use error-correcting codes detecting and correcting errors in the information transmission channel. The requirements for encoding and decoding methods and procedures by the reference to the spectral and energy efficiency of a communication system give the task of constructing simplified algorithms for correcting errors in transmitted information. In modern information systems, cyclic BCH and Reed-Solomon codes, which require high redundancy, are the most used to ensure their corrective abilities. Moreover, the processing time of information in a decoding device, which depends on the complexity of the encoding and decoding algorithms, limits significantly the operating time of the error protection system.

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

BCH and Reed-Solomon codes used in modern information storage and transmission systems built on classical error detection and correction procedures, at some decoding stages, due to a significant amount of data, have high arithmetic complexity. This leads to limited potential possibilities of the above said correction codes to provide a given probability of information loss.

In this case, in order to increase the operation speed of information protection systems from errors in information data transmission systems, it is necessary to improve and refine existing algorithms for decoding of cyclic BCH codes for improving the correcting ability of the code by reducing the computational complexity of encoding and decoding procedures with high redundancy.

**The object of research** is the decoding process of cyclic RS and BCH codes in the digital communication systems.

**The subject** of research is a method for finding the roots of the error locator polynomial whose coefficients belong to in the Galois fields.

**The aim** of this work is to study the decoding methods of RS and BCH codes, allowing detecting and correcting errors in the code sequence, as well as the devel-o$^{pm}$ent of an accelerated algorithm for calculating the roots of the error polynomial, the implementation of the algorithm into Visual Studio and the subsequent analysis of the results.

## 1 PROBLEM STATEMENT

The ways of BCH and RS codes decoding are quite well developed in theory and practice, which is presented in [1], but nevertheless the implementation of decoding algorithms is quite a laborious task, especially if the finite fields of a large order are used.

A typical procedure for the decoding of cyclic RS codes is proposed by the author R.E. Blahut [2] and consists of the following stages:

– the calculation of syndrome components (syndrome vector);

– the formation of a key equation and finding of the error locator polynomial by one of the methods – Peterson, Berlekamp-Massey or the Euclidean algorithm;

– the searching for the roots of the error locator polynomial using the Chan’s method – a complete enumeration of all values;

– the calculation of the error values polynomial and error character determination based on the Forney algorithm;

– the correction of erroneous characters.

The GF($2^m$) finite field contains $2^m$ elements (1, $\alpha^1$, $\alpha^2$, $\alpha^3$, ...), each of which is represented by a binary vector of $m$ bits and in the practice PC code mostly uses the calculations in the GF($2^m$) finite fields.

One of the most time-consuming stages of RS code decoding is searching for the roots of the error locator polynomial

$$\sigma(x) = \sigma_0 + \sigma_1 x + \sigma_2 x^2 + ... + \sigma_v x^v . \qquad (1)$$

It is known that the Chan’s method can be used to find the roots of the error locator polynomial (1) degree ν in the field GF($2^m$). However, the Chan’s algorithm requires multiplication of each coefficient $\sigma_i$ in the formula (1) by the element of field GF($2^m$) using α degrees.

Therefore, to correct errors in the code sequence, the Chan’s method is used, which in fact is a complete enumeration of all elements, which significantly affects the effectiveness of the decoding device. If the decoder has a known error locator polynomial $\sigma(x) = \sum_{i=0}^{t} \sigma_i x^i$ , which roots are mutual to the error position locators, then the Chan’s procedure can be applied to each of the locators 1, $\alpha^1$, $\alpha^2$, $\alpha^3$, …, $\alpha^l$ ($l = 2^m - 1$) to check if the symbol displayed at the moment is an error [1]. I.e., for all non-zero elements of $\alpha^k$ field GF($2^m$), the condition $\sigma(\alpha^k) = 0$ is checked, and its implementation indicates that $\alpha^{-k}$ is a mutual root of the error locator polynomial to the erroneous symbol. In this case, the number of operations to implement the Chan’s procedure is $2t$ ($2^m - 1$).

Thus, the Chan’s procedure for searching the roots of the polynomial error becomes quite complicated for calculations in large finite fields ($m$>8) and for error locators polynomials of a large degree, because it requires a significant number of operations.

Therefore, the task of this work is to improve the method for determining the positions of distortions in the code word and to reduce the computational complexity of the algorithm for finding the roots of the error polynomial in the finite fields of GF($2^m$) when decoding RS and BCH codes.

## 2 REVIEW OF THE LITERATURE

The principles of encoding, more modern methods of errors control, code applications for the design of real error control systems are covered in [3]. With the help of a special class of $p$-polynomials the authors offered a fast algorithm for finding polynomial roots of the degree less than 5. In this algorithm the polynomial defined in the formula (1) is transformed into an affine polynomial above the GF($2^m$) field. Then the roots can be found by solving a system from $m$ linear equations with $m$ unknown above the GF(2) field. Therefore, at the minor values of ν, the advantage of the above algorithm over the Chan’s method is that the calculations in the GF($2^m$) field (multiplying $\sigma_i$ coefficients by α degrees) required in the standard Chan’s search method are completely excluded.

The searching method for the roots of the polynomial on the base of some transformation that makes it possible to group some polynomial constituents of the degree no higher than 11 into multiple affine polynomials is proposed in [4]. This method makes it possible to accelerate the calculations, but this algorithm has its disadvantages. For example, it can be used for polynomials which degrees cannot be higher than 11. The building and decoding of BCH codes for different communication systems are described in [5]. The authors propose an improved algorithm for searching the roots of polynomials over the

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

finite fields. This algorithm significantly accelerates the process of BCH codes decoding. Also the improved hybrid algorithm of polynomial roots finding over finite fields is considered in [6] and [7] articles. The authors have combined the Jiang's algorithm, which is based on the decomposition of the error locator polynomial in the sum of multiple affine polynomials with modified analytical methods for solving polynomials with small degree in radicals. A number of works have been devoted to different issues of BCH and RS decoding, where simple equation for Reed-Solomon codes on the base of uniting of two algorithms (Berlekamp-Massey algorithm and Euclidean) was proposed [8], the new approach to calculating the total error locator polynomial [9–10], the dependence of indicators of reliability and efficiency of information transmission on the parameters of Reed-Solomon codes are given in [11], the main stages of Reed-Solomon codes encoding (decoding) for practical application [12] were reviewed.

The analysis of publications has shown the necessity to study the procedures of cyclic BCH codes decoding, as well as the methods of determining error positions in the code word, based on the search of the roots of the error locator polynomial. The calculations of the roots values are carried out using the elements of finite fields, taking into account the degree of error polynomial, which significantly affects the calculation time. Therefore, the analysis results of known methods of roots search will be presented further, and also the accelerated algorithms of definition of errors positions in a code word at decoding RS and BCH codes will be described.

## 3 MATERIALS AND METHODS

The solution of the problem of searching the roots of the error locator polynomial (1), which $\sigma_i$ coefficients belong to the GF($2^m$) finite field, using the Berlekamp algorithm is based on a special class of polynomials. These polynomials, which roots can be found much easier, are called *p*-polynomials or linearized polynomials [1].

$$F(x) = \sum_i f_i x^{2^i} \; . \tag{2}$$

If we suggest that the error locators polynomial has the degree $v=2^i$, then it can be represented as

$$\sigma(x) = F(x) + \sigma_0, \quad \sigma_0 \in \mathrm{GF}(2^m) \, . \tag{3}$$

The polynomial of the type (3) is called affine polynomial [6].

If the polynomial is checked when searching for roots, then we obtain

$$F(x) = \sigma_0, \quad \sigma_0 \in \mathrm{GF}(2^m) \, . \tag{4}$$

For any GF($2^m$) finite field the standard basis is a set of m elements 1, $\alpha^1$, $\alpha^2$, $\alpha^3$, … $\alpha^{m-1}$ [14]. The values of the linearized polynomial $F_0(\alpha^0)$ $F_1(\alpha^1)$ ….. $F_{m-1}(\alpha^{m-1})$ at the points of the standard basis of the GF($2^m$) finite field can be calculated using (2). The obtained result lies in the GF($2^m$) field, but for further calculations it is reasonable to present it as a binary vector corresponding to the field element

$$F_i(\alpha^i) = \{a_0\, a_1\, a_2 \ldots a_{m-1}\}, \quad a_k \in \mathrm{GF}(2) \, .$$

We denote the searching root of the error polynomial as $\alpha \in \mathrm{GF}(2^m)$ field element through a binary vector $\beta^i = \{b_0\, b_1\, b_2 \ldots b_{m-1}\}, \;\; b_k \in \mathrm{GF}(2).$

Then it is derived from the formula (1) that the zero coefficient of error locator polynomials represented as $Y = \{y_0\; y_1 \ldots y_{m-1}\}$ binary vector can be obtained by multiplying $\beta^i = \{b_0\, b_1\, b_2 \ldots b_{m-1}\}, \;\; b_k \in \mathrm{GF}(2)$ the vector of the line by a matrix *A* over field GF($2^m$)

$$[b_0\, b_1\, b_2 \ldots b_{m-1}] \begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} \cdots a_{0,m-1} \\ a_{1,0} & a_{1,1} & a_{1,2} \;\cdots\; a_{1,m-1} \\ \cdots\cdots\cdots\cdots\cdots\cdots & & \\ a_{m-1,0} & a_{m-1,1} & \cdots\; a_{m-1,m-1} \end{bmatrix} = [y_0\, y_1 \cdots y_{m-1}] \, . \tag{5}$$

Therefore, the decomposition coefficients of *F*(*x*) polynomial on a standard basis can be obtained

$$F(x) = \sum_{k=0}^{m-1} b_k F(\alpha^k), \quad b_k \in \mathrm{GF}(2) \, . \tag{6}$$

For example, it is necessary to find the roots of the error locator polynomial that coefficients belong to the GF($2^3$) finite field, generated by the primitive polynomial $p(x)=x^3+x+1$:

$$\sigma(x) = x^2 + \alpha^6 x + \alpha^6 \, . \tag{7}$$

We represent the error polynomial as $\sigma(x) = F(x) + \sigma_0$, where the $F(x) = x^2 + \alpha^6 x$ linearized polynomial above the GF($2^3$) field and the zero coefficient is $\sigma_0 = \alpha^6$. Then, in accordance with (4), we obtain

$$F(x) = x^2 + \alpha^6 x = \alpha^6 \, . \tag{8}$$

The element of the GF($2^3$) $\alpha^6$ finite field is comparable to the $[\,y_2\, y_1\, y_0\,]$ binary vector. If the $\beta^i = \{b_0\, b_1\, b_2\,\}, \;\; b_k \in \mathrm{GF}(2)$ searching roots of the error locator polynomial, then in accordance with (6) we obtain

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

$$b_0F(\alpha^0)+b_1F(\alpha^1)+b_2F(\alpha^2)=\left[y_0\ y_1\ y_2\right]. \qquad (9)$$

The basis vectors $F_2(\alpha^2)$, $F_1(\alpha^1)$ и $F_0(\alpha^0)$ taking into account (8) can be calculated as::

$$F_0(\alpha^0)=\alpha^0+\alpha^6=\alpha^2,$$

$$F_1(\alpha^1)=\alpha^2+\alpha^7=\alpha^6,$$

$$F_2(\alpha^2)=\alpha^4+\alpha^1=\alpha^2.$$

Each $F(\alpha^i)$ basis vector above the GF($2^3$) field corresponds to a binary

$$F(\alpha^i)=\{a_0\ a_1\ a_2\},\quad a_k\in\ \mathrm{GF}(2)$$

$$F(\alpha^0)=\{1\,0\,0\},\quad F(\alpha^1)=\{1\,0\,1\},\quad F(\alpha^2)=\{1\,0\,0\}.$$

Then, in accordance with (5), we have a system of the linear binary equations

$$\begin{bmatrix} b_0 & b_1 & b_2 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}.$$

Therefore, the roots of the error locator polynomial (7) can be found by solving simultaneously three equations with three unknown $\{b_0\ b_1\ b_2\},\ \ b_k\in\mathrm{GF}(2)$. It is obvious that the elements of the GF($2^3$) finite field $\alpha^1$ {0 1 0} and $\alpha^5$ {1 1 1} are the roots of the polynomial (7). This example shows that to find the roots of the polynomial (7) by means of the Berlekamp algorithm it is necessary to calculate only three $F(\alpha^2)$, $F(\alpha^1)$ and $F(\alpha^0)$ in the GF($2^3$) field instead of calculating all values $F(\alpha^0)$, $F(\alpha^1)$ …. $F(\alpha^6)$, required for searching the roots by the Chan's method.

For the decrease in complexity and quantity of computational operations at finding the roots of errors locator polynomial by the Berlekamp method it is expedient to use earlier calculated values at the following stage of algorithm. For this purpose we will use the property of the linearized polynomial

$$F(x+y)=F(x)+F(y). \qquad (10)$$

Thus, in accordance with the property (10), we obtain

$$F(\alpha^i+\alpha^j)=F(\alpha^i)+F(\alpha^j) \qquad (11)$$

or

$$F(\alpha^k)+F(\alpha^i)=F(\alpha^j),\quad \alpha^k=\alpha^i+\alpha^j.$$

Consider such a pair of $\alpha^k$ and $\alpha^i$ elements of the finite field GF($2^m$) as a standard basis (binary vector of $m$ length), which would differ from each other in one position. Then in the formula (11) the $\alpha^j$ field element is in this case a single vector of $m$ length. As it has been pointed out earlier such elements (binary sequence weight 1) are basic in the finite field GF($2^m$) □ 1, $\alpha^1$, $\alpha^2$, $\alpha^3$, … $\alpha^{m-1}$.

Hence, if we rank (order) the elements of the GF($2^m$) finite field in such a way that the nearby vectors will differ exactly in one position. Then, at each step of the algorithm for searching the roots of the error locator polynomial, the calculations are reduced to a single addition □ the previous value and the value of the linearized polynomial at the points of the standard basis $F(\alpha^0)$ $F(\alpha^1)$ ….. $F(\alpha^{m-1})$ of the GF($2^m$) finite field.

Thus, in order to calculate all values of the error locator polynomial represented as the affine polynomial (3) in all points of the finite field $\beta^i\in\mathrm{GF}(2^m),\ i=0,1,\dots,m-1$, it is necessary to complete the following steps

$$F(\beta^i)=F(\beta^{i-1})+F_j(\alpha^j), \qquad (12)$$

($\alpha^j=\beta^i\oplus\beta^{i-1}$ – corresponds to one of the basic elements of the GF($2^m$) field).

The expression (12) provides the procedure for finding the polynomial value set $\sigma(x)$ at all points $\alpha^i\in\mathrm{GF}(2^m)$. The calculation requires ordering of all the elements of the field, the presence of the previous value $F(\beta^{i-1})$ and the previously found values of the basis vectors $F(\alpha^i)$.

Thus, to find the roots of the error locator polynomial, and therefore to calculate the polynomial value (3) at all points of the finite field it is necessary to use the following algorithm:

1. To set the value of zero coefficient of error locators polynomial $\sigma_0$ and linearized polynomial $F(x)$.
2. To find the values of the linearized polynomial $F(x)$ at the points of the standard basis of the finite field GF($2^m$)

$$F_k=F(\alpha^i),\ i=0,1,\dots,m-1.$$

3. To perform the initialization $F(\beta^0)=0$.
4. To arrange all elements of the $\alpha^i\in\mathrm{GF}(2^m)$ finite field in the form of binary vectors in such a way that two any nearby vectors will differ exactly in one position.
5. To calculate the value of

$$F(\beta^i)=F(\beta^{i-1})+F_j(\alpha^j),\ \alpha^j=\beta^i\oplus\beta^{i-1}.$$

6. If $F(\beta^i)=\sigma_0$, then the field element $\beta^i$ is the polynomial root of $\sigma(x)$.

## 4 EXPERIMENTS

To confirm the functionality of the above-mentioned theoretical calculations we will perform the procedure of searching the roots of the error locator polynomial, which coefficients belong to the GF($2^4$) finite field, generated by the primitive polynomial $p(x)=x^4+x+1$:

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

$$\sigma(x) = x^2 + \alpha^5 x + 1 . \qquad (13)$$

1. We represent the error polynomial (13) as $\sigma(x) = F(x) + \sigma_0$, where the linearized polynomial $F(x) = x^2 + \alpha^5 x$ above the GF($2^4$) and the zero coefficient $\sigma_0 = 1$.

2. We will find the values of the linearized polynomial at the points of the standard basis of the GF($2^4$) finite field

$F_0(\alpha^0) = \alpha^{10}$ (1010), $F_1(\alpha^1) = \alpha^5$ (1011), $F_2(\alpha^2) =$
$= \alpha^8$ (1110), $F_3(\alpha^3) = \alpha^0$ (0001).

3. We will perform the initialization $F(\beta^0) = 0$.

4. The elements of the GF($2^4$) finite field, decomposed as the binary vectors, should be ordered so that the nearby vectors will differ in one position.

Table 1 – The GF($2^4$) Finite Field Element Ordering

| Field Element | Binary Vector | Field Element | Binary Vector |
|---|---|---|---|
| $\beta^0(0)$ | 0000 | $\beta^8(\alpha^{13})$ | 1100 |
| $\beta^1(\alpha^0)$ | 0001 | $\beta^9(\alpha^{13})$ | 1101 |
| $\beta^2(\alpha^{12})$ | 0011 | $\beta^{10}(\alpha^{13})$ | 1111 |
| $\beta^3(\alpha^1)$ | 0010 | $\beta^{11}(\alpha^{13})$ | 1110 |
| $\beta^4(\alpha^{13})$ | 0110 | $\beta^{12}(\alpha^{13})$ | 1010 |
| $\beta^5(\alpha^7)$ | 0111 | $\beta^{13}(\alpha^{13})$ | 1011 |
| $\beta^6(\alpha^{13})$ | 0101 | $\beta^{14}(\alpha^{13})$ | 1001 |
| $\beta^7(\alpha^{13})$ | 0100 | $\beta^{15}(\alpha^{13})$ | 1000 |

5. We will perform step-by-step calculations in accordance with (12).

1) $\beta^0 \oplus \beta^1 = 0001$ – corresponds to the basic element $\alpha^0$ the GF($2^4$) field

$F(\beta^1) = F(\beta^0) + F_0(\alpha^0)$ = 0000+1010=1010;

2) $\beta^1 \oplus \beta^2 = 0010$ – corresponds to the basic element $\alpha^1$ the GF($2^4$) field

$F(\beta^2) = F(\beta^1) + F_1(\alpha^1)$ = 1010+1011 = 0001;

3) $\beta^2 \oplus \beta^3 = 0001$ – corresponds to the basic element $\alpha^0$ the GF($2^4$) field

$F(\beta^3) = F(\beta^2) + F_0(\alpha^0)$ = 0001 + 1010 = 1011;

4) $\beta^3 \oplus \beta^4 = 0100$ – corresponds to the basic element $\alpha^2$ the GF($2^4$) field

$F(\beta^4) = F(\beta^3) + F_2(\alpha^2)$ = 1011 + 1110 = 0101;

…

14) $\beta^{13} \oplus \beta^{14} = 0010$ – corresponds to the basic element $\alpha^1$ the *GF*($2^4$) field

$F(\beta^{14}) = F(\beta^{13}) + F_1(\alpha^1)$ = 0000 + 1011 = 1011;

15) $\beta^{14} \oplus \beta^{15} = 0001$ – corresponds to the basic element $\alpha^0$ the *GF*($2^4$) field

$F(\beta^{15}) = F(\beta^{14}) + F_0(\alpha^0)$ = 1011 + 1010 = 0001.

As $F(\beta^2) = 1$ и $F(\beta^{15}) = 1$, the roots of the polynomial (13) are consequently the elements of the field $\beta^2$ and $\beta^{15}$, $\alpha^{12}$ and $\alpha^3$ respectively.

To estimate the real efficiency of the proposed modified algorithm for calculating the roots of the error locator polynomial, the software simulation in *C*++ language has been implemented. The multiplication in the GF($2^8$) finite fields for the Chan's method was carried out using the tables of logarithms and antilogarithms. The calculations of the roots and the comparison of results were carried out only for the linearized error polynomials and for the elements of the $\alpha^0$, $\alpha^1$, …, $\alpha^{254}$ field.

## 5 RESULTS

Figure 1 represents the measurement results of the computation speed for the modified root search method and for the Chan's method.

Based on the results, it can be concluded that the application of the modified algorithm for searching the roots of error locator polynomials, presented as the linearized polynomials, makes it possible to achieve a speed gain of 1.5 times in comparison with the Chan's method.

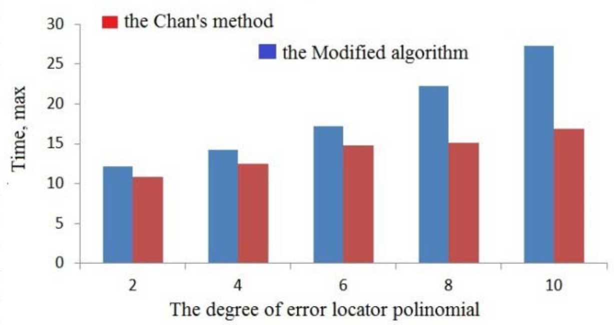

Figure 1 – The speed of the error polynomial roots computation

For the GF($2^8$) finite field, the average number of operations to search for the of polynomial roots of 8 degree is 4100 for the Chan's procedure and 1532 for the proposed modified root search method. For 16 degree of polynomial, the average number of operations is 8240 and 3562 respectively.

## 6 DISCUSSION

As all polynomials of the second degree are linearized, their roots can be found by solving the corresponding system of linear equations by the Berlekamp-Massey method [7]. Also, a linearized polynomial of error loca-

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

tors with a nonzero coefficient can be represented as an affine polynomial. In this case, the root search procedure is reduced to the addition of binary vectors in the GF(2) field GF as the elements of the GF($2^m$) finite field, which allows reducing significantly the number of addition and multiplication operations of the field elements.

If each element $\alpha^i$ of the GF($2^m$) finite field can be compared to a binary vector of $m$ length, the field elements can always be ordered in such a way that two nearby vectors will differ from each other exactly in the same position, for example, using the Gray code.

The described algorithm allows finding the roots of the error locator polynomial if the polynomial (1) is represented as an affine polynomial (3). However, this root-searching procedure can also be generalized in the case of arbitrary polynomials if they are broken down into a set of affine polynomials.

## CONCLUSIONS

**In this work the acute scientific** task of the accelerated search of the roots of the error locator polynomial which define the errors positions in the accepted code sequence at the stage of cyclic BCH and RS codes decoding is solved.

**Also the effective algorithm** of the roots calculation of the errors polynomial above the GF($2^m$) finite fields is developed. The improved algorithm for calculating the roots of the polynomial errors with the coefficients in the finite field based on the Berlekamp-Massey algorithm for the linearized polynomials has been proposed, which provides a minimum number of arithmetic operations due to the use of data obtained from the previous stages of calculations. The proposed algorithm reduces the complexity of root calculations at one point of the finite field due to the application of a special arrangement of all elements of the finite field.

**Practical newness** of the work results lies in the implementation of the program model of the code message decoding system, which allows conducting the error correction, both for the classical method of searching the roots, and for the developed modified algorithm for calculating the positions of the disturbed characters. The modeling results have confirmed the efficiency of the proposed algorithm for the root searching of the error locator polynomial while the cyclic BCH and RS codes decoding, which allows achieving the speed gain by 1.5 times, compared with the Chan’s search method.

## ACKNOWLEDGEMENTS

The work is supported by National Technical University of Ukraine «Kharkiv Polytechnic Institute».

## REFERENCES

1. Berlekamp E. R. Algebraic Coding Theory. New York, McGraw-Hill, 1968, 466 p.
2. Blahut Richard E. Algebraic Codes for Data Transmission. Cambridge, Cambridge University Press, 2003, 482 p.
3. Lin S., Costello D. J. Error control coding: fundamentals and applications. Prentice-Hall Inc, Printed in the United States of America, 2004, 624 p.
4. Truong T. K., Jeng J. H., Reed I. S. Fast algorithm for computing the roots of error locator polynomials up to degree 11 in Reed-Solomon decoders, *IEEE Transactions on Communications*, 2001, Vol. 49, Issue 5, pp. 779–783. DOI:10.1109/26.923801.
5. Nabipour S., Javidan J., Zare Gholamreza F. Error Detection Mechanism Based on Bch Decoder and Root Finding of Polynomial Over Finite Fields, *Journal of Mathematics and Computer Science,* 2014, Issue 4, pp. 271–281. DOI: http://dx.doi.org/10.22436/jmcs.012.04.03.
6. Fedorenko S. V., Trifonov P. V. Finding roots of polynomials over finite fields, *IEEE Transactions on Communications*, 2002, Vol. 50, Issue 11, pp. 1709–1711. DOI: 10.1109 / TCOMM.2002.805269.
7. Fedorenko S., Trifonov P., Costa E., Haas H. Improved hybrid algorithm for finding roots of error-locator polynomials, *European Transactions on Telecommunications,* 2003, Vol. 14, Issue 5, pp. 411–416. DOI: https://doi.org/10.1002/ett.936.
8. Bras-Amorós M., Michael O’Sullivan E. The Symmetric Key Equation for Reed–Solomon Codes and a New Perspective on the Berlekamp-Massey Algorithm, *Symmetry*, 2019, Vol. 11 (1357). DOI: 10.3390/sym11111357.
9. Ceria M., Mora T., Sala M. Help: a sparse error locator polynomial for BCH codes, *Applicable Algebra in Engineering, Communication and Computing,* 2020, Vol. 31, pp. 215–233. DOI: https://doi.org/10.1007/s00200-020-00427-x.
10. Almuzakkia M. Z., Oharac K. Computing general error locator polynomial of 3-error-correcting BCH codes via syndrome varieties using minimal polynomial [Electronic resource], *ISCS, Selected Papers*, 2015, pp. 80–85. Access mode: https://mafiadoc.com/computing-general-error-locator-polynomial_5bad280f097c479e798b4727.html.
11. Freyman V. I. Research of the reed-solomon codes characteristic for realization within control systems devices, *Radio Electronics, Computer Science, Control*, 2019, Vol. 3, pp. 143–151. DOI: https://doi.org/10.15588/1607-3274-2019-3-1.
12. Liang Z., Zhang W. Efficient Berlekamp-Massey Algorithm and Architecture for Reed-Solomon Decoder, *Journal of Signal Processing Systems*, 2017, Vol. 86, Issue 1, pp. 51–65. DOI: https://doi.org/10.1007/s11265-015-1094-1.
13. Caruso F., Orsini E., Sala M., Tinnirello C. On the Shape of the General Error Locator Polynomial for Cyclic Codes / // *IEEE Transactions on Information Theory*, 2017, Vol. 63, Issue 6, pp. 3641–3657. DOI: 10.1109/TIT.2017.2692213.
14. Bucerzan D., Dragoi V., Richmond T. The simple roots problem [Electronic resource], *Proceedings of the Romanian Academy, (Special issue), Cryptology Science*, 2017, Vol. 18, pp. 317–332. Access mode: https://acad.ro/sectii2002/proceedings/doc2017-4s/03artSupl.pdf.

Received 19.06.2020.
Accepted 20.08.2020.

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

УДК 004.056.5

# МОДИФІКОВАНИЙ АЛГОРИТМ ПОШУКУ КОРЕНІВ ПОЛІНОМА ЛОКАТОРІВ ПОМИЛОК ПРИ ДЕКОДУВАННІ БЧХ КОДІВ

**Крилова В. А.** – канд. техн. наук, доцент кафедри автоматизації та управління в технічних системах Національного технічного університету «Харківський політехнічний інститут», Харків, Україна.

**Тверитникова О. Є.** – д-р наук, професор кафедри інформаційно-вимірювальних технологій та систем Національного технічного університету «Харківський політехнічний інститут», Харків, Україна.

**Васильченков О. Г.** – канд. техн. наук, доцент кафедри автоматизації та управління в технічних системах Національного технічного університету «Харківський політехнічний інститут», Харків, Україна.

**Колісник Т. П.** – канд. пед. наук, доцент кафедри інформаційних технологій та кібербезпеки Харківського національного університету внутрішніх справ, Харків, Україна.

## АНОТАЦІЯ

**Актуальність**. У телекомунікаційних та інформаційних системах зв'язку з підвищеною шумовою складової використовуються перешкодостійкі циклічні БЧХ та коди Ріда-Соломона. Коригування та виправлення помилок в повідомленні вимагає ефективних методів декодування. Одним з етапів процедури декодування РС і БЧХ кодів для визначення позицій спотворень є пошук коренів полінома локаторів помилок. Обчислення коренів многочлена, особливо у кодів зі значною коректує здатністю, є трудомісткою завданням, що вимагає високої обчислювальної складності. Тому удосконалення методів декодування БЧХ і РС кодів, що дозволяють зменшити складність обчислень, є актуальним завданням.

**Мета роботи**. Дослідження і синтез прискореного алгоритму пошуку коренів полінома локаторів помилок, представленого у вигляді афінного многочлена з коефіцієнтами в кінцевих полях, який дозволяє прискорити процес декодування БЧХ і РС кодів.

**Метод.** Класичний метод пошуку коренів на базі алгоритму Ченя виконується за допомогою арифметики кінцевих полів Галуа і трудомісткість розрахунків, в даному випадку, залежить від кількості операцій додавання і множення. Для линеаризиваних поліномів процедура пошуку коренів, заснована на двійковій арифметиці та здійснюється з урахуванням значень отриманих на попередніх етапах обчислення, що забезпечує мінімальне число арифметичних операцій.

**Результати**. Розроблено прискорений алгоритм обчислення значень полінома локаторів помилок у всіх точках кінцевого поля GF ($2^m$) для линеаризированих многочленів на базі методу Берлекемпа-Мессі. Алгоритм містить мінімальну кількість операцій додавань, за рахунок використання на кожному етапі обчислень, значень отриманих на попередньому кроці, а також виконання складання в кінцевому полі GF(2). Запропоновано модифікований метод пошуку коренів для афінних поліномів над кінцевими полями, що дозволяє визначити позиції помилок в кодовому слові під час декодування циклічних БЧХ і РС кодів.

**Висновки**. Наукова новизна роботи полягає в удосконаленні алгоритму обчислення коренів многочлена локаторів помилок, коефіцієнти якого належать до елементів кінцевого поля. При цьому спрощується процедура декодування циклічних БЧХ і РС кодів, за рахунок зниження обчислювальної складності одного з етапів декодування – знаходження позицій помилок з використанням модифікованого алгоритму Берлекемпа-Мессі. Дані факти підтверджені результатами програмного моделювання алгоритму пошуку коренів полінома локаторів помилок. Показано, що застосування прискореного методу дозволяє досягти виграшу по швидкодії в 1,5 рази.

**КЛЮЧОВІ СЛОВА**: БЧХ коди, поліном локаторів помилок, пошук Ченя, алгоритм Берлекемпа-Мессі, коди Ріда-Соломона.

УДК 004.056.5

# МОДИФИЦИРОВАННЫЙ АЛГОРИТМ ПОИСКА КОРНЕЙ ПОЛИНОМА ЛОКАТОРОВ ОШИБОК ПРИ ДЕКОДИРОВАНИИ БЧХ КОДОВ

**Крылова В. А.** – канд. техн. наук, доцент кафедры автоматизации и управления в технических системах Национального технического университета «Харьковский политехнический институт», Харьков, Украина.

**Тверитникова Е. Е.** – д-р наук, профессор кафедры информационно-измерительных технологий и систем Национального технического университета «Харьковский политехнический институт», Харьков, Украина.

**Васильченков О. Г.** – канд. техн. наук, доцент кафедры автоматизации и управления в технических системах Национального технического университета «Харьковский политехнический институт», Харьков, Украина.

**Колесник Т. П.** – канд. пед. наук, доцент кафедры информационных технологий и кибербезопасности Харьковского национального университета внутренних дел, Харьков, Украина.

## АННОТАЦИЯ

**Актуальность.** В телекоммуникационных и информационных системах связи с повышенной шумовой составляющей используются помехоустойчивые циклические БЧХ и коды Рида-Соломона. Корректировка и исправление ошибок в сообщении требует эффективных методов декодирования. Одним из этапов процедуры декодирования РС и БЧХ кодов для определения позиций искажений является поиск корней полинома локаторов ошибок. Вычисление корней многочлена, особенно у кодов со значительной корректирующей способностью, является трудоемкой задачей, требующей высокой вычислительной сложности. Поэтому усовершенствование методов декодирования БЧХ и РС кодов, позволяющих уменьшить сложность вычислений, является актуальной задачей.

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14

**Цель исследования.** Исследование и синтез ускоренного алгоритма поиска корней полинома локаторов ошибок, представленного в виде аффинного многочлена с коэффициентами в конечных полях, который позволяет ускорить процесс декодирования БЧХ и РС кодов.

**Метод.** Классический метод поиска корней на базе алгоритма Ченя выполняется с помощью арифметики конечных полей Галуа и трудоемкость расчетов, в данном случае, зависит от количества операций сложения и умножения. Для линеаризированных полиномов процедура поиска корней, основанная на двоичной арифметике, осуществляется с учетом значений полученных на предыдущих этапах вычисления, что обеспечивает минимальное число арифметических операций.

**Результаты.** Разработан ускоренный алгоритм вычисления значений полинома локаторов ошибок во всех точках конечного поля GF($2^m$) для линеаризированных многочленов на базе метода Берлекэмпа-Месси. Алгоритм содержит минимальное число операций сложений, за счет использования на каждом этапе вычислений, значений полученных на предыдущем шаге, а также выполнения сложения в конечном поле GF(2). Предложен модифицированный метод поиска корней для аффинных полиномов над конечными полями, позволяющий определить позиции ошибок в кодовом слове при декодировании циклических БЧХ и РС кодов.

**Выводы.** Научная новизна работы состоит в усовершенствовании алгоритма вычисления корней многочлена локаторов ошибок, коэффициенты которого принадлежат элементам конечного поля. При этом упрощается процедура декодирования циклических БЧХ и РС кодов, за счет снижения вычислительной сложности одного из этапов декодирования – нахождения позиций ошибок с использованием модифицированного алгоритма Берлекэмпа-Месси. Данные факты подтверждены результатами программного моделирования алгоритма поиска корней полинома локаторов ошибок. Показано, что применение ускоренного метода позволяет достичь выигрыша по быстродействию в 1,5 раза.

**KEYWORDS:** БЧХ коды, полином локаторов ошибок, поиск Ченя, алгоритм Берлекэмпа-Месси, коды Рида-Соломона.

## ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Berlekamp E. R. Algebraic Coding Theory / E. R. Berlekamp. – New York : McGraw-Hill, 1968. – 466 p.
2. Blahut Richard E. Algebraic Codes for Data Transmission / E. Richard Blahut. – Cambridge : Cambridge University Press, 2003. – 482 p.
3. Lin S. Error control coding: fundamentals and applications / S. Lin, D. J. Costello. Prentice-Hall Inc : Printed in the United States of America, 2004. – 624 p.
4. Truong T. K. Fast algorithm for computing the roots of error locator polynomials up to degree 11 in Reed-Solomon decoders / T. K. Truong, J. H. Jeng, I. S. Reed // IEEE Transactions on Communications. – 2001. – Vol. 49, Issue 5. – P. 779–783. DOI:10.1109/26.923801.
5. Nabipour S. Error Detection Mechanism Based on Bch Decoder and Root Finding of Polynomial Over Finite Fields / S. Nabipour, J. Javidan, F. Zare Gholamreza // Journal of Mathematics and Computer Science. – 2014, Issue 4. – P. 271–281. DOI: http://dx.doi.org/10.22436/jmcs.012.04.03.
6. Fedorenko S. V. Finding roots of polynomials over finite fields / S. V. Fedorenko, P. V. Trifonov // IEEE Transactions on Communications. – 2002. – Vol. 50, Issue 11. – P. 1709–1711. DOI: 10.1109 / TCOMM.2002.805269.
7. Improved hybrid algorithm for finding roots of error-locator polynomials / [S. Fedorenko, P. Trifonov, E. Costa, H. Haas] // European Transactions on Telecommunications. – 2003. – Vol. 14, Issue 5. – P. 411–416. DOI: https://doi.org/10.1002/ett.936.
8. Bras-Amorós M. The Symmetric Key Equation for Reed–Solomon Codes and a New Perspective on the Berlekamp–Massey Algorithm / M. Bras-Amorós, E. Michael O’Sullivan // Symmetry – 2019. – Vol. 11 (1357). DOI: 10.3390/sym11111357.
9. Ceria M. Help: a sparse error locator polynomial for BCH codes / M. Ceria, T. Mora, M. Sala // Applicable Algebra in Engineering, Communication and Computing. – 2020. – Vol. 31. – P. 215–233. DOI: https://doi.org/10.1007/s00200-020-00427-x.
10. Almuzakkia M. Z. Computing general error locator polynomial of 3-error-correcting BCH codes via syndrome varieties using minimal polynomial [Electronic resource] / M. Z. Almuzakkia, K. Oharac // ISCS, Selected Papers. – 2015. – P. 80–85. – Access mode: https://mafiadoc.com/computing-general-error-locator-polynomial_5bad280f097c479e798b4727.html.
11. Фрейман В. И. Исследование характеристик кодов Рида-Соломона для реализации в устройствах систем управления / В. И. Фрейман // Радіоелектроніка, інформатика, управління. – 2019 (3). – С. 143–151. DOI: https://doi.org/10.15588/1607-3274-2019-3-1.
12. Liang Z. Efficient Berlekamp-Massey Algorithm and Architecture for Reed-Solomon Decoder / Z. Liang, W. Zhang // Journal of Signal Processing Systems. – 2017. – Vol. 86, Issue 1. – P. 51–65. DOI: https://doi.org/10.1007/s11265-015-1094-1.
13. On the Shape of the General Error Locator Polynomial for Cyclic Codes / F. Caruso, E. Orsini, M. Sala, C. Tinnirello // IEEE Transactions on Information Theory. – 2017. – Vol. 63, Issue 6. – P. 3641–3657. DOI: 10.1109/TIT.2017.2692213.
14. Bucerzan D. The simple roots problem [Электронный ресурс] / D. Bucerzan, V. Dragoi, T. Richmond // Proceedings of the Romanian Academy, (Special issue), Cryptology Science. – 2017. – Vol. 18. – P. 317–332. – Режим доступа: https://acad.ro/sectii2002/proceedings/doc2017-4s/03artSupl.pdf.

© Krylova V. A., Tverytnykova E. E., Vasylchenkov O. G., Kolisnyk T. P., 2020
DOI 10.15588/1607-3274-2020-3-14