ON THE POWER TO DETECT ERRORS OF ONE ERROR-DETECTING CODE

Prof. Ilievska N. PhD.
Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University, Skopje, Macedonia
natasa.ilievska@finki.ukim.mk

Abstract: *When messages are transmitted through the communication channel, due to the noises in the channel, they can be incorrectly transmitted. Therefore, the receiver must ensure that it has the correct message. Similarly, the data stored in the storage media due to different circumstances can be corrupted. In order to check whether the data is corrupted or to check whether the receiver received the correct message, so-called error-detecting codes are used. When using such a code, it is important to know the power of the code to detect errors. In this paper we will analyze the ability to detect errors of one such error-detecting code. We compare the error-detecting capabilities of the code in a case when a quasigroup of order 4, order 8 and order 16 is used for coding for three different lengths of the redundancy. At the end we made a conclusion about the best choice of parameters from the aspect of the ability of the code to surely detect errors.*

Keywords: ERROR-DETECTING CODE, QUASIGROUP, CODED BLOCK, SYMBOL, ALPHABET

1. Introduction

Nowadays, when we live in the age of fast communication networks through which huge amounts of data are transmitted on a daily basis, but also huge amount of data is stored on the storage media, it is important to ensure security and reliability (accuracy) of transmitted or stored data. In order to achieve this goal, the standard approach is the data to be first encrypted, then the encrypted data is coded and transmitted through the channel. The received data is first decoded and then decrypted. But also, there are some solutions in which it is made effort the encryption and coding procedures to be combined into one algorithm, so-called crypt-coding algorithm.

This paper is focused on the coding part of the communication system. The role of the codes is to ensure a reliability and accuracy of the data. This means that they are used in order to ensure that data is not corrupted during transmission through the channel (or while it is stored on some storage medium, when used in such medium).

In [1] we defined an error-detecting code. This code is defined using quasigroups. The performances of the defined code depend on the quasigroup used for coding. An important parameter that defines performances of the code from the aspect of the ability of the code to detect errors is the number of errors that the code surely detects. This number is the maximum number of incorrectly transmitted bits up to which the code will detect the error for sure. In this paper we compare the number of errors that the code surely detects when a linear quasigroup of order 4, order 8 and order 16 is used for coding.

The paper is organized in a following way: In Section 2 are given the basic mathematical definitions and the definition of the code. In Section 3 we compare the ability of the code to detect errors for sure when a quasigroup of order 4, order 8 and order 16 is used for coding. We consider the cases when the length of the redundancy is 8, 12 and 16 bits when the quasigroups of order 4 and order 16 are used for coding and redundancy of 9, 12 and 15 bits when the quasigroup of order 8 is used for coding. At the and we conclude the paper.

2. Code for error-detection

Quasigroup is an algebraic structure that is used in coding theory and cryptography. The code that we consider in the paper is also defined using this algebraic structure. For that reason, first we will slightly explain this term.

Let Q be a set and * is a binary operation. If Q is closed under the operation *, i.e., if for all x, y∈Q, the quasigroup product $x*y \in Q$, then the structure $(Q, *)$ is called a groupoid. A groupoid $(Q, *)$ in which for all $u, v \in Q$, the equations $x*u=v$ and $u*y=v$ have unique solution by x and y is called quasigroup. Order of a finite quasigroup is the number of elements in the quasigroup. In this paper, if the order of the quasigroup is n, then we will take that the elements of the quasigroups are the integers from 0 to n-1, i.e., $Q=\{0, 1, 2, \ldots, n-1\}$.

In the code considered in this paper we use a so-called linear quasigroups. A quasigroup $(Q, *)$ of order 2^q is linear if there are non-singular binary matrices A and B of order $q \times q$ and a binary matrix C of order $1 \times q$, such that

$$(\forall\, x, y \in Q)\ \boldsymbol{x*y} = \boldsymbol{x}A + \boldsymbol{y}B + C \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the binary representations of $\boldsymbol{x}$ and $\boldsymbol{y}$ as $1 \times q$ vector, $\boldsymbol{x*y}$ is the binary representations of the quasigroup product $x*y$ as $1 \times q$ vector and + is a binary addition.

The code that we consider in this paper is defined in a following way. Let for coding be used a linear quasigroup Q and let the input block be $a_0 a_1 \ldots a_{n-1}$ ($a_i \in Q$, for all $i \in \{0, 1, \ldots, n-1\}$). Then, the redundant symbols are defined in a following way:

$$\boldsymbol{d_i} = \boldsymbol{a_i} A^{n-2} + \sum_{j=1}^{n-2} \boldsymbol{a_{i+j}} B A^{n-j-2} + C \sum_{j=0}^{n-3} A^j,\ \ i=0, 1, \ldots, r \tag{2}$$

where $r \in N$ is the parameter of the code such that $1 \le r \le n-1$, $\boldsymbol{a_i}$ is the binary representations of the information symbol a_i, A, B and C are the binary matrices that satisfy the equation (1) and + is a binary addition. With equation (2), the redundant symbols are obtained in a binary form.

Now, the redundant symbols are concatenated on the input block $\boldsymbol{a_0 a_1 \ldots a_{n-1}}$, which gives the coded block $\boldsymbol{a_0 a_1 \ldots a_{n-1} d_0 d_1 \ldots d_r}$. The coded block is in a binary form and is transmitted through a binary symmetric channel. Since there are noises in the channel, some symbols may be incorrectly transmitted. This may lead to situation in which the recipient receives inaccurate data.

When the receiver receives the output block, in order to check the accuracy of the data, it calculates the redundant symbols that correspond to the received information block. If the calculated symbols match with the received ones, then the receiver accepts the output block. If there is a mismatch in at least one symbol, the receiver does not accept the received output block and asks for repeated transmission of that block.

But, as with any error-detecting code there is a possibility that the code will not detect the errors in transmission, i.e., it is possible to have complete match of the received and calculated redundant symbols, although the block is not correctly transmitted. We started the investigation of the error-detecting properties of the code in [1] and [2] by obtaining the probability of undetected errors, and then we continue the research in a direction of the number of errors that the code surely detects ([3], [4], [5]).

3. Comparison and analysis of the power of the code to surely detect errors

The number of errors that the code surely detects depends on the length of the input block, but also on the parameter r of the

model and the quasigroup used for coding. In this section we will compare the results for the number of errors that the code detects for sure when a linear quasigroup of order 4, order 8 and order 16 is used for coding. We consider the cases when the length of the redundancy is 8, 12 and 16 bits (i.e., 9, 12 and 15 bits when the quasigroup of order 8 is used for coding).

First, for coding was used the following linear quasigroup of order 4:

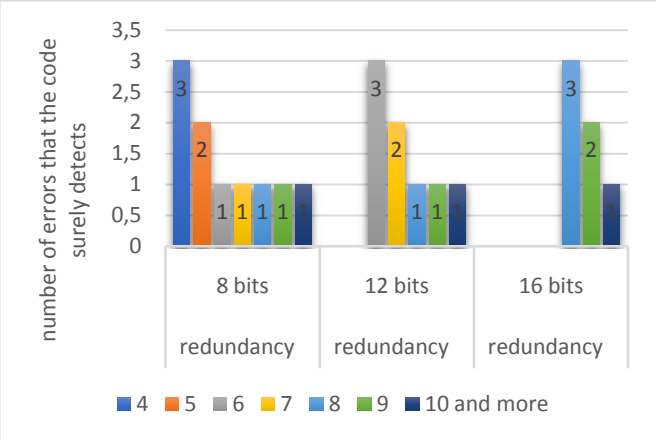
*	0	1	2	3
0	0	1	3	2
1	3	2	0	1
2	2	3	1	0
3	1	0	2	3

Since this is liner quasigroup, there are binary matrices such that (1) is satisfied. This binary matrices A, B and C are the following:

$$A = \begin{matrix} 1 & 0 \\ 1 & 1 \end{matrix}, B = \begin{matrix} 1 & 1 \\ 0 & 1 \end{matrix}, C = [0 \quad 0]$$

The number of errors that the code surely detects when the linear quasigroup of order 4 is used for coding is given in Fig. 1 ([3]).

***Figure 1**: Number of errors that the code surely detects when the linear quasigroup of order 4 is used for coding.*



The length of the input block is expressed as number of characters from the quasigroup used for coding and is denoted with the color of the pillar. The blue pillar represents the input blocks with length 4 characters from the quasigroup, the orange is for the input blocks with length 5 characters from the quasigroup, the gray pillar is for input blocks with length 6 characters, the yellow for input blocks with length 7 characters, light blue is for the blocks of length 8 characters, the green one for blocks of length 9 characters, while the results for the input blocks of length greater than or equal to 10 characters are represented with the dark blue pillar. In the first section of Fig.1 are given the results when the redundancy is 8 bits, in the second section are the results when the redundancy is 12 bits and in the third section are the results when the redundancy is 16 bits.

The quasigroup of order 8 used for coding is:

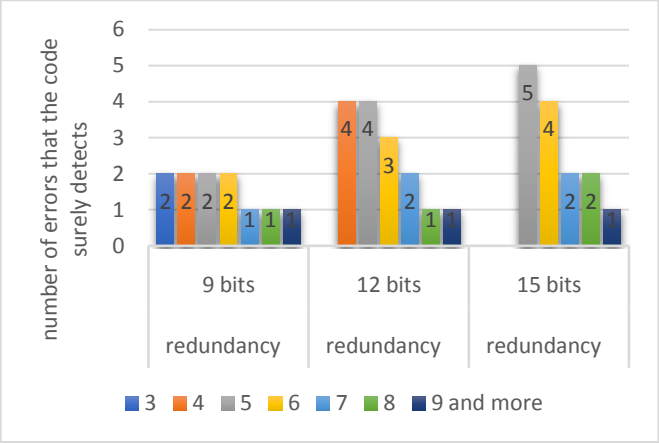
*	0	1	2	3	4	5	6	7
0	0	5	7	2	3	6	4	1
1	7	2	0	5	4	1	3	6
2	3	6	4	1	0	5	7	2
3	4	1	3	6	7	2	0	5
4	5	0	2	7	6	3	1	4
5	2	7	5	0	1	4	6	3
6	6	3	1	4	5	0	2	7
7	1	4	6	3	2	7	5	0

This linear quasigroup is represented by the following binary matrices:

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, C = [0 \quad 0 \quad 0]$$

The number of errors that the code surely detects when this quasigroup is used for coding is given in Fig. 2 ([4]).

***Figure 2**: Number of errors that the code surely detects when the linear quasigroup of order 8 is used for coding.*



The linear quasigroup of order 16 used for coding is:

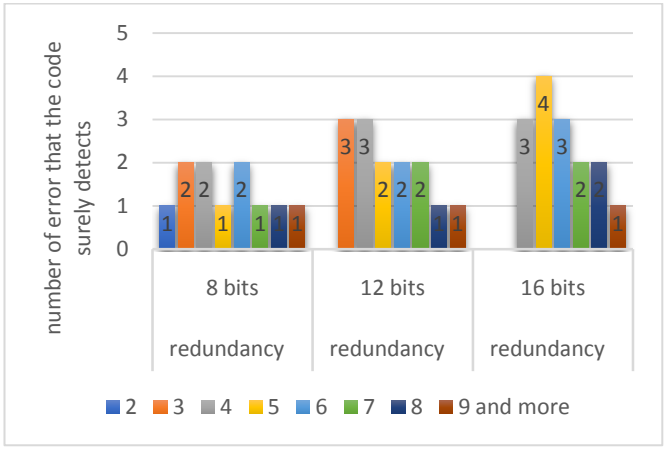
*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	7	F	8	D	A	2	5	B	C	4	3	6	1	9	E
1	F	8	0	7	2	5	D	A	4	3	B	C	9	E	6	1
2	D	A	2	5	0	7	F	8	6	1	9	E	B	C	4	3
3	2	5	D	A	F	8	0	7	9	E	6	1	4	3	B	C
4	B	C	4	3	6	1	9	E	0	7	F	8	D	A	2	5
5	4	3	B	C	9	E	6	1	F	8	0	7	2	5	D	A
6	6	1	9	E	B	C	4	3	D	A	2	5	0	7	F	8
7	9	E	6	1	4	3	B	C	2	5	D	A	F	8	0	7
8	7	0	8	F	A	D	5	2	C	B	3	4	1	6	E	9
9	8	F	7	0	5	2	A	D	3	4	C	B	E	9	1	6
A	A	D	5	2	7	0	8	F	1	6	E	9	C	B	3	4
B	5	2	A	D	8	F	7	0	E	9	1	6	3	4	C	B
C	C	B	3	4	1	6	E	9	7	0	8	F	A	D	5	2
D	3	4	C	B	E	9	1	6	8	F	7	0	5	2	A	D
E	1	6	E	9	C	B	3	4	A	D	5	2	7	0	8	F
F	E	9	1	6	3	4	C	B	5	2	A	D	8	F	7	0

The corresponding matrices are:

$$A = \begin{matrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{matrix}, B = \begin{matrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{matrix}, C = [0 \quad 0 \quad 0 \quad 0]$$

The number of errors that the code surely detects when the quasigroup of order 16 is used for coding is given in Fig. 3 ([5]).

***Figure 3**: Number of errors that the code surely detects when the linear quasigroup of order 16 is used for coding.*



In order to compare the results for quasigroups of different order and to see how the order of the quasigroups affect the number of errors that the code surely detects, we will express the length of the input blocks in bits. Each element from a quasigroup of order 2^q is represented with q bits in the binary representation. This means that the element from a quasigroup of order 4 is represented by 2 bits, of order 8 by 3 bits, while the element from a quasigroup of order 16 by 3 bits in a binary form. Using this fact and the results represented in Fig. 1, Fig. 2 and Fig. 3, we obtain Table 1, Table 2 and Table 3.

In the first columns of Table 1 – Table 3 is given the length n of the input block, now expressed as number of bits. In the second, third and fourth column is given the number of errors that the code surely detects when the linear quasigroup of order 4, order 8 and order 16 is used for coding, respectively. Since each element from a quasigroup of order 4 is represented by 2 bits, the length of input blocks in binary form when this quasigroup is used for coding is a multiple of 2. Therefore, there are values for the quasigroup of order 4 only on the positions that are multiple of 2. Similarly, since each element from a quasigroup of order 8 is represented by 3 bits in the binary form, in the case when the quasigroup of order 8 is used for coding the length of input blocks is a multiple of 3. Analogous, when a quasigroup of order 16 is used for coding, the length of the input block is multiple of 4. In order to compare the error-detecting capabilities of the code when quasigroups of different orders are used for coding, we should compare the number of errors that the code surely detects when code rates are equal. Therefore, we will compare this number when the lengths of the input blocks expressed in bits are equal and the redundancies also have equal length. In Table 1 are given the results when the redundancy is 8 bits (i.e., 9 bits when the quasigroup of order 8 is used for coding), in Table 2 are the results obtained when the redundancy is 12 bits and in Table 3 are represented the results for the number of errors that the code surely detects when the redundancy is 16 bits (i.e., 15 bits when the quasigroup of order 8 is used for coding).

***Table 1**: Number of errors that the code surely detects when the redundancy is 8 bits (i.e., 9 bits for quasigroup od order 8).*

n	**quasi. order 4**	**quasi. order 8**	**quasi. order 16**
8	3		1
9		2	
10	2		
11			
12	1	2	2
13			
14	1		
15		2	
16	1		2
17			
18	1	2	
19			
20	1		1
21		1	
22	1		
23			
24	1	1	2
25			
26	1		
27		1	
≥28	1		1

***Table 2**: Number of errors that the code surely detects when the redundancy is 12 bits.*

n	**quasi. order 4**	**quasi. order 8**	**quasi. order 16**
12	3	4	3
13			
14	2		
15		4	
16	1		3
17			
18	1	3	
19			
20	1		2
21		2	
22	1		
23			
24	1	1	2
25			
26	1		
27		1	
28	1		2
29			
30	1	1	
31			
≥32	1		1

***Table 3**: Number of errors that the code surely detects when the redundancy is 16 bits (i.e., 15 bits for quasigroup od order 8).*

n	**quasi. order 4**	**quasi. order 8**	**quasi. order 16**
15		5	
16	3		3
17			
18	2	4	
19			
20	1		4
21		2	
22	1		
23			
24	1	2	3
25			
26	1		
27		1	
28	1		2
29			
30	1	1	
31			
32	1		2
33		1	
34	1		
35			
≥36	1	1	1

From Table1 – Table 3 we can see that for longer input blocks, the number of errors that the code surely detects is equal, regardless of the order of the quasigroup used for coding, but also regardless of the length of the redundancy. Namely, when the redundancy is 8 bits (i.e., 9 bits for the quasigroup of order 8) and the length of the input block is greater than or equal to 28 bits, the code surely detects 1 incorrectly transmitted bit regardless which of the three quasigroups is used for coding (Table 1). When the redundancy is 12 bits and the length of the input blocks is greater than or equal to 32 bit the code also detects 1 incorrectly transmitted bit for all three quasigroups (Table 2). The same conclusion holds when the redundancy is 16 bits (i.e., 15 bits for the quasigroup of order 8) and the length of the input blocks is greater than or equal to 36 bits (Table 3). But, for shorter input blocks, there is a difference in the error-detecting capability of the code, depending on which quasigroup is used for coding.

From Table1 – Table 3 we can see that whenever the input blocks have equal lengths, except when the length of the input

blocks is 8 bits and the redundancy is 8 bits, the number of errors that the code surely detects when the quasigroup of order 16 is used for coding is greater than or equal to the number of errors that the code surely detects when the quasigroup of order 4 is used for coding. But, in the case when the redundancy is 8 bits, when the input block has length 10 bits for quasigroup of order 4 the code rate is better (larger) than when the input block has length 8 bits for quasigroup of order 16, and also the number of surely detected errors is larger.

When comparing the quasigroups of order 4 and order 8, we see that the quasigroup of order 8 provides greater than or equal number of errors that the code surely detects than the quasigroup of order 4 always when the lengths of the input blocks are equal and lengths of the redundancies are equal. The quasigroup of order 4 is in advantage when the redundancy is 8 bits and the length of the input block is 8 bits, while the length of the input block is 9 bits for quasigroup of order 8 (equal code rates, but larger number of surely detected errors), but also when the length of the input block is 10 bits for quasigroup of order 4 and 9 bit for order 8 (better code rate for the quasigroup of order 4 while equal number of surely detected errors).

From all above we can conclude that for short input blocks, the quasigroup of order 4 gives smallest number of errors that are surely detected from the three considered quasigroups when the code rates are equal for a given length of the redundancy (except in the few indicated cases). But on other hand, if we want to achieve largest number of surely detected errors when the redundancy is 8 bits, we should use exactly the quasigroup of order 4 for coding and to divide the input message into blocks of length 8 bits and to code each of them separately. But then the code rate is 1/2.

The situation is not so simple when comparing the quasigroups of order 8 and order 16. In the cases of equal lengths of the input blocks, the quasigroup of order 8 provide greater number of surely detected incorrectly transmitted bits than the quasigroup of order 16 only in the case when the redundancy and the length of the input block are 12 bits. In all other cases, the quasigroup of order 16 gives greater or equal number of surely detected errors. But, on the other hand there are cases when for a given length of the redundancy, the quasigroup of order 8 achieves better code rate and larger or equal number of surely detected errors than the quasigroup of order 16. For example, when the redundancy is 8 bits, in the cases when the quasigroup of order 8 is used and the length of the input block is 9, 12, 15 or 18 bits the code rate is greater than or equal as when the quasigroup of order 16 is used and the length of the input block is 8 bits, but also the quasigroup of order 8 gives larger number of surely detected errors. Other such cases when the quasigroup of order 8 is in advantage are: When the length of the redundancy is 12 bits, the input block is 15 or 18 bits for quasigroup of order 8 and 12 bits for quasigroup of order 16, the input block is 18 bits for the quasigroup of order 8 and 16 bits for quasigroup of order 16, the input block is 21 bits for the quasigroup of order 8 and 20 bits for quasigroup of order 16; also when redundancy is 16 (i.e., 15) bits and the length of the input block is 15 or 18 bits for the quasigroup of order 8 and 16 bits for quasigroup of order 16. In these cases, the quasigroup of order 8 provide better code rate, but also higher number of surely detected errors or equal number of surely detected errors for better code rate.

When the redundancy is 12 bits, the best result is achieved when the quasigroup of order 8 is used and the length of the input block is 15 bits. The same number of surely detected errors is obtained also when the length of the input block is 12 bits, but since in this case the rate of the code is smaller, it is better to divide the input messages in blocks of length 15 bits. When the redundancy is 16 bits, the largest number of surely detected errors will be obtained if the input messages are divided in blocks of length 15 bits and the quasigroup of order 8 is used for coding. This is also generally the best choice of redundancy length, input block length and quasigroup for coding from all considered cases from the aspect of the number of errors that the code surely detects. In this case is achieved the largest number of surely detected incorrectly transmitted bits, i.e., the number of errors that the code detects for sure in this case is 5 bits.

4. Conclusion

In this paper we compared the ability to detect errors from the aspect of the number of errors that it surely detects of one error-detecting code based on linear quasigroups in the cases when a linear quasigroup of order 4, order 8 and order 16 is used for coding and the redundancy is 8, 12 and 16 bits (i.e., 9, 12 and 15 bits for quasigroups of order 8). For longer input blocks, for each of the considered cases for the length of the redundancy, the number of errors that the code surely detects is equal, regardless which of the three quasigroups is used for coding. When the length of the redundancy is 8 (i.e., 9 bits) this conclusion holds when the input blocks have length greater than or equal to 28 bits. When the redundancy is 12 bits this numbers are equal when the length of the input block is greater than or equal to 32 bits, while when the redundancy is 16 (i.e., 15 bits) the conclusion holds for input blocks with length greater than or equal to 36 bits.

For shorter input blocks, the quasigroup of order 4 has smaller number of surely detected incorrectly transmitted bits than the quasigroup of order 8 and order 16 when the code rates are equal for given length of the redundancy. The quasigroup of order 4 has better performances than the other two quasigroups in only few cases.

For short input blocks, there is no general conclusion which of the quasigroups of order 8 and order 16 has greater number of surely detected incorrectly transmitted bits. It depends on the length of the input blocks. For some lengths one quasigroup yields better results, for some the other.

From the aspect of the number of errors that the code surely detects it is best to divide the input messages into blocks of length 15 bits and to code each block individually using the quasigroup of order 8, such that 15 redundant bits are added on each input block (i.e., to choose r=4).

5. References

[1] Ilievska N., Gligoroski D., "Simulation of some new models of error-detecting codes," Proceedings of the 22nd Telecommunications Forum Telfor, Belgrade, Serbia, 2014, pp. 395-398.

[2] Ilievska N., Gligoroski D., "Simulation of a Quasigroup Error-Detecting Linear Code," Proceedings of the 38th International ICT Convention MIPRO CTI – Telecommunications & Information, Opatija, Croatia, 2015, pp. 483 – 488.

[3] Ilievska N., "Towards the Fixed Length Redundancy Code," Proceedings of the 27th Telecommunications Forum Telfor, Belgrade, Serbia, 2019, pp. 241-244.

[4] Ilievska N., "Modeling the number of errors that the code surely detects," Proceedings of the III International Scientific Conference Mathmodel, Borovets, Bulgaria, 2019.

[5] Ilievska N., "Towards the number of errors that the code detects for sure," Journal of Engineering Science and Technology Review, accepted.