ON A STREAM ERROR-CORRECTING CODE

S. Markovski, V. Bakeva

Institute of Informatics, Faculty of Natural Sciences and Mathematics
Sts. Cyril and Methodius University
Arhimedova bb, P.O.Box 162, Skopje, Macedonia
{smile, verica}@ii.edu.mk

Abstract: We use quasigroups to define suitable codes of stream nature. These codes are error-correcting with high probability.

Keywords: quasigroup, code, error-correcting code.

1. Preliminaries

Error-correcting codes are widely used in applications such as returning pictures from deep space, design of registration numbers and so on. Error-correcting codes are used to correct errors when messages are transmitted through a noisy communication channel. The channel may be a telephone line, a high frequency radio link, or a satellite communication link. The noise may be produced by human errors, lightings, thermal noises, imperfections in equipment, etc., and may result in errors so that the data received is different from data, that is sent. The object of an error-correcting code is to encode the data, by adding a certain amount of redundancy to the message, so that the original message can be recovered if (not too many) errors have been occurred. Here we define error-correcting codes using quasigroups.

A quasigroup is a groupoid on a set A with binary operation * satisfying the law

$$(\forall x, y \in A)(\exists! u, v \in A)(x * u = y \,\&\, v * x = y).$$

In what follows we consider only the set A = {0,1}, and $*$ will denote a quasigroup operation on the set A. There are only two quasigroup operations on the set A, and here we took (A, *) to be defined by the table

*	0	1
0	1	0
1	0	1

It follows from the obtained results that it does not matter which quasigroup operation will be chosen.

2. Description of the system

Let $a_1a_2...a_n...$be a source message, where $a_i \in A = \{0,1\}$ for each i, and let $b_0 \in A$ be a given (known) binary letter. The sequence $b_1b_2...b_n...$ is obtained from the sequence $a_1a_2...a_n...$ such that $b_i=b_{i-1}*a_i$ for each i = 1,2,… We sent the sequence $a_1b_1a_2$ $b_2...a_nb_n...$through a noisy channel. Since there are noises in the channel, the sequences obtained in the exit of the channel can be different than the sent ones. We consider binary symmetrical channel, which means that 0 can be replaced by 1 (and opposite, 1 by 0) with probability p ($0< p<1/2$). This communication channel can be represented as in Fig. 1.

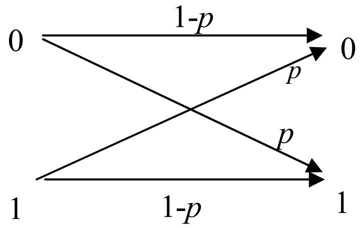

Figure 1

The bit wise sequence of errors $e_1e_2e_3$…. is defined in such a way that $e_i= 1$ if an error on i-th place appears, otherwise $e_i= 0$. The distance between two consecutive errors in a sequence of errors is the number of zeros between them. The error distance is the smallest distance between any two consecutive errors. If $a_1a_2...a_n...$ is a source message, then the obtained message is $c_1c_2...c_n...$, where $c_i = a_i + e_i$ and + denotes the addition modulo 2.

The reason why the sequence $a_1b_1a_2b_2...a_nb_n...$is sent, is for checking the correctness of the transmission of the binary data $a_1a_2 ...a_n$ …. The checking of the correctness and correction of the incorrect transmitted data can start immediately after receiving of the first two letters of the message. Namely, when the letters a_1 and b_1 are received, it can be checked if $b_0*a_1=b_1$, after that if $b_1*a_2=b_2$, and so on. Since there are noises in the channel, some of the equations in the previous sequence should not be satisfied. We propose the following algorithm for error correction:

if $b_{i-1}*a_i \neq b_i$ and $b_i*a_{i+1} \neq b_{i+1}$	then $b_i \leftarrow 1 - b_i$
if $b_{i-1}*a_i \neq b_i$ and $b_i*a_{i+1} = b_{i+1}$	then $\mathrm{a_i} \leftarrow 1 - \mathrm{a_i}$

Theorem: If the error distance is at least 3 then all of the errors will be corrected (i.e. the obtained message at the exit of the channel will be identical with the source one).

Proof: Let us obtain the subsequence $b_{i-1}a_i'b_ia_{i+1}b_{i+1}$ where a_i' denotes that the symbol a_i is transmitted incorrectly, and suppose that the next error will appear after b_{i+1}. Then, applying our algorithm we have the following:

$$b_{i-1} * a_i' \neq b_i \quad \text{and} \quad b_i * a_{i+1} = b_{i+1}$$

and we correct a_i.

Similarly, if the obtained subsequence is $b_{i-1}a_ib_i'a_{i+1}b_{i+1}$ where b_i' denotes that the symbol b_i is transmitted incorrectly, and suppose that the next error will appear after b_{i+1}. Then

$$b_{i-1} * a_i \neq b_i' \quad \text{and} \quad b_i' * a_{i+1} \neq b_{i+1}$$

and according to our algorithm we correct b_i. □

Given any source sequence $a_1a_2 \ldots a_n \ldots$, the error distance may not be at least three. We have several situations:

The errors distance is 0:

Let the obtained subsequence be $b_{i-1}a_i'b_i'a_{i+1}b_{i+1}a_{i+2}b_{i+2}$. Then

$$b_{i-1} * a_i' = b_i', \qquad b_i' * a_{i+1} \neq b_{i+1} \qquad \text{and} \qquad b_{i+1} * a_{i+2} = b_{i+2}$$

and we will correct a_{i+1} by $1 - a_{i+1}$. The decoded message will contain the subsequence $a_i'a_{i+1}'a_{i+2}$ instead of $a_ia_{i+1}a_{i+2}$.

Let the obtained subsequence be $b_{i-1}a_ib_i'a_{i+1}'b_{i+1}a_{i+2}b_{i+2}$. Then

$$b_{i-1} * a_i \neq b_i', \qquad b_i' * a_{i+1}' = b_{i+1} \qquad (\text{and} \qquad b_{i+1} * a_{i+2} = b_{i+2})$$

and we will correct a_i by $1 - a_i$. The decoded message will contain again the subsequence $a_i'a_{i+1}'a_{i+2}$ instead of $a_ia_{i+1}a_{i+2}$.

The errors distance is 1:

2.1. Let the subsequence $b_{i-1}a_i'b_ia_{i+1}'b_{i+1}a_{i+2}b_{i+2}$ be obtained. Then

$$b_{i-1} * a_i' \neq b_i, \qquad b_i * a_{i+1}' \neq b_{i+1} \qquad (\text{and} \qquad b_{i+1} * a_{i+2} = b_{i+2})$$

and b_i will be replaced by $1-b_i$, but the incorrectness in the subsequence $a'_i a'_{i+1} a_{i+2}$ will remain.

2.2. Let the subsequence $b_{i-1} a_i b'_i a_{i+1} b'_{i+1} a_{i+2} b_{i+2} a_{i+3} b_{i+3}$ be obtained. Then

$$b_{i-1} * a_i \neq b'_i, \qquad b'_i * a_{i+1} = b'_{i+1} \text{ and}$$

$$b'_{i+1} * a_{i+2} \neq b_{i+2}, \qquad b_{i+2} * a_{i+3} = b_{i+3}.$$

We will correct a_i by $1-a_i$ and a_{i+1} by $1-a_{i+1}$, which means that the obtained subsequence will be $a'_i a_{i+1} a'_{i+2} a_{i+3}$ instead $a_i\, a_{i+1}\, a_{i+2}\, a_{i+3}$.

Let the errors happen on distance 2:

Let the subsequence $b_{i-1} a'_i b_i a_{i+1} b'_{i+1} a_{i+2} b_{i+2} a_{i+3} b_{i+3}$ be obtained. In that case, we have

$$b_{i-1} * a'_i \neq b_i, \qquad b_i * a_{i+1} \neq b'_{i+1}$$

which means that b_i will be replaced by $1-b_i$, and after that

$$b'_{i+1} * a_{i+2} \neq b_{i+2}, \qquad b_{i+2} * a_{i+3} = b_{i+3}$$

imply that a_{i+2} will be replaced by $1-a_{i+2}$, i.e. the obtained sequence is $a'_i a_{i+1} a'_{i+2} a_{i+3}$.

Let the obtained subsequence be $b_{i-1} a_i b'_i a_{i+1} b_{i+1} a'_{i+2} b_{i+2} a_{i+3} b_{i+3}$. Here,

$$b_{i-1} * a_i \neq b'_i, \qquad b'_i * a_{i+1} \neq b_{i+1},$$

and b'_i will be corrected by $1-b'_i$. After that,

$$b_{i+1} * a'_{i+2} \neq b_{i+2}, \qquad b_{i+2} * a_{i+3} = b_{i+3}$$

which means that a'_{i+2} will be replaced by $1-a'_{i+2}$. In that case, the subsequence $a_i\, a_{i+1}\, a_{i+2}\, a_{i+3}$ will be correctly received.

From the previous discussion, we can conclude that, in the worst case, a subsequence of a given message will be incorrectly received if the errors will happen on the distance less than 2. The probability of that event is

$$p^2 + p(1-p)p + p(1-p)^2 p = p^2\left[3(1-p) + p^2\right]$$

which is approximately $3p^2$.

We have made several experiments in order to obtain that probability experimentally. Namely, we generated random error sequences of the length 10^6 consisting of 0s and 1s with a given probability p that 1 will be generated i.e. an error will happen. In these files we determined the frequency of 1s, which are on error distance less than 3. For different values of p, the results obtained experimentally and the theoretical ones are given in Table 1 and Fig. 2.

Probability of error in channel	Experimental probability of uncorrected transfer	Theoretical probability of uncorrected transfer
0.200	0.097590	0.097600000
0.100	0.027088	0.027100000
0.050	0.007132	0.007132500
0.010	0.000295	0.000297000
0.005	0.000076	0.000074600
0.001	0.000003	0.000002997

Table 1

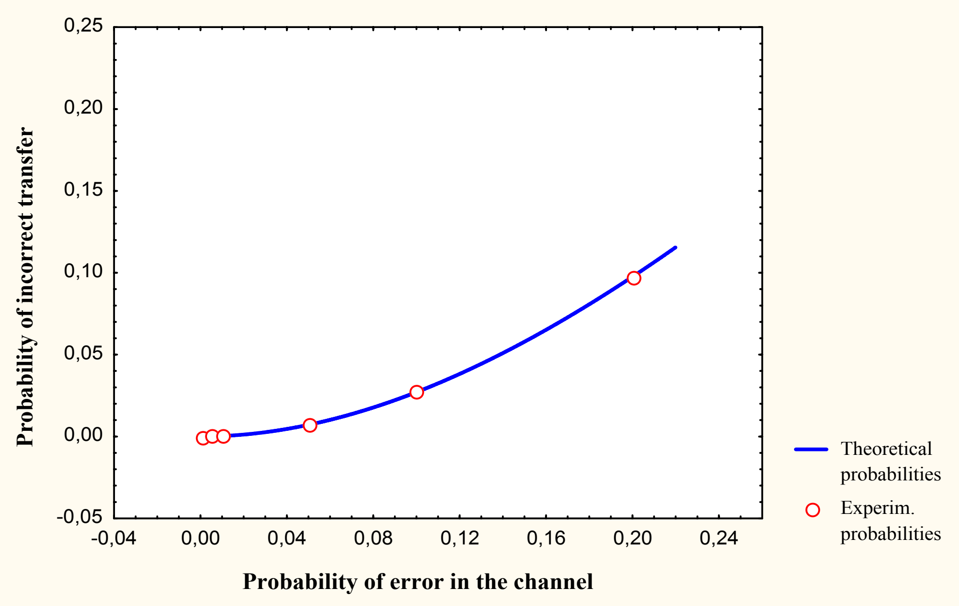

Figure 2

3. Conclusions

In this note we construct a code by using a quasigroup operation, which correct incorrectly transmitted symbols with high probability. The advantage of our code is in its stream nature. All codes which we know use blocks with fixed or variable length as code words and decoding can start after receiving of a block. Here, the decoding can start immediately after receiving of the first two letters. It is clear that our method can be defined as suitable block code as well. We mention here that the quasigroup method of coding can be applied to any finite set A $=\{l_1, l_2, \ldots, l_k\}$ of letters on which a quasigroup operation is defined.

4. References:

1. Hill, Raymond (1986): *A First Course in Coding Theory*, Clarendon Press, Oxford.