Analysis of Linear Block Codes as Sources with Memory

Valeriu Munteanu, Daniela Tarniceriu, Gheorghe Zaharia

Analysis of Linear Block Codes as Sources with Memory

Valeriu MUNTEANU[1], Daniela TARNICERIU[1], and Gheorghe ZAHARIA[2]
[1] *"Gheorghe Asachi" Technical University Iasi, Faculty of Electronics, Telecommunications and Information Technology, 700506, Romania,*
[2] *IETR – INSA, UMR CNRS 6164 Rennes, France*
vmuntean@etc.tuiasi.ro

***Abstract*— The linear, binary, block codes with no equally likely probabilities for the binary symbols are analyzed. The encoding graph for systematic linear block codes is proposed. These codes are seen as sources with memory and the information quantities H(S,X), H(S), H(X), H(X|S), H(S|X), I(S,X) are derived. On the base of these quantities, the code performances are analyzed.**

***Index Terms*— information quantities, linear, block codes, sources with memory.**

I. INTRODUCTION

Generally, for a linear, block code the encoding operation is performed according to relation [1–5]:

$$[v]=[i_1 i_2 ... i_k][G] \tag{1}$$

where $[v]$ is the code word;

i_j, $j=\overline{1,k}$, are the information symbols;

[G] is the generator matrix.

The number of rows in the generator matrix is equal to the number of the information symbols, *k*, and the number of columns is equal to the code word length, *n*.

For error correction, besides the *k* information symbols, *m* parity - check symbols have to be added, so that, the code word length is:

$$n=m+k \tag{2}$$

In this paper only linear, binary, block codes will be analyzed. The analysis can be also extended for a non-binary alphabet, at the expense of computation complexity, the conclusions being the same.

Since from rel. (1) 2^k distinct code words have to result, the rank of the generator matrix [G] must be equal to *k*. This means that, by elementary transformations, the generator matrix can be expressed in the equivalent form:

$$[G]=\begin{bmatrix} 1 & 0 & ... & 0 & 0 & p_{1,1} & p_{1,2} & ... & p_{1,m} \\ 0 & 1 & ... & 0 & 0 & p_{2,1} & p_{2,2} & ... & p_{2,m} \\ ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & 1 & 0 & p_{k-1,1} & p_{k-1,2} & ... & p_{k-1,m} \\ 0 & 0 & ... & 0 & 1 & p_{k,1} & p_{k,2} & ... & p_{k,m} \end{bmatrix}=[I_k P] \tag{3}$$

where $[I_k]$ denotes the identity matrix of rank *k*;

$p_{i,j}\in\{0,1\}$.

If the generator matrix is as in (3), the code is systematic, with the information symbols placed on the first *k* positions in the code word, that is:

$$[v]=[i_1 i_2 ... i_k c_1 c_2 ... c_m] \tag{4}$$

where $c_j\in\{0,1\}$, $j=\overline{1,m}$, are parity - check symbols.

For the seek of generality we assume that the binary symbols are provided by the binary, memoryless source, characterized by the distribution

$$X:\begin{pmatrix} x_1=0 & x_2=1 \\ 1-p & p \end{pmatrix}, 0<p<1 \tag{5}$$

The average information per information symbol is calculated by the entropy [6-11].

$$\begin{aligned} H(X)&=-\sum_{j=1}^{2} p(x_j)\log_2 p(x_j)= \\ &=-(1-p)\log_2(1-p)-p\log_2 p=H_b(p) \end{aligned} \tag{6}$$

To simplify the writing, in the following, we will no longer specify the logarithm base; it being understood it is equal to 2.

II. THE ENCODING GRAPH ATTACHED TO A LINEAR, BINARY, BLOCK CODE

The number of levels for the graph corresponding to a linear, binary, block code is equal to the code word length. There are 2^k nodes placed on the last level, corresponding to the 2^k code words.

On the first level in the graph, there are 2^1 nodes, on the second one, 2^2, and so on, on the level k, 2^k nodes. As on the last level, (n), also 2^k nodes exist, this means that from the level *k*, no nodes diversifies into two branches. Up to the level k-2, inclusive, the number of nodes can be calculated with relation

$$2^1+2^2+...+2^{k-2}=2^{k-1}-2 \tag{7}$$

The first node on the level k-1 has the index 2^{k-1}-1 and the last node on this level, the index 2^{k-1}-1+2^{k-1}-1=2^k-2. So, the first node on the level k has the index 2^k-1 and the last node on this level, the index 2^k-1+2^k-1=2^{k+1}-2.

The number of levels between *k* and *n* will be, obviously, *n*-*k*+1. The total number of nodes in the encoding graph, excepting the root, is:

$$2^1+2^2+...+2^{k-1}+(n-k+1)2^k=2^k(n-k+1)-2 \tag{8}$$

The index of the last node in the encoding graph will be $2^k(n-k+2)-2$, while the index of the first node on the level *n* will have the index

$2^k(n-k+2)-2-2^k+1=2^k(n-k+1)-1$.

The index of the last node on the level *n*-1 will be $2^k(n-k+1)-2$ and that of the first node on the same level will be $2^k(n-k+1)-2-2^k+1=2^k(n-k)-1$.

The graph corresponding to a linear, binary, block code is given in Fig. 1.

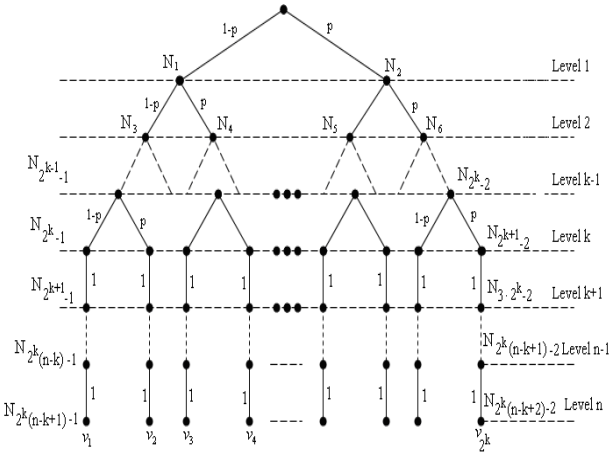

Fig. 1. The encoding graph

Since the code is systematic, with the information symbols grouped on the first *k* places in the code word, on the branches of the encoding graph up to level *k*, inclusive, all the binary distinct sequences of length *k* result. In the encoding graph the parity check symbols are placed between levels *k* and *n*. The parity check symbols calculated by means of relation (1) will be also "0" or "1", but they will be provided with probability "1" from each node between levels *k* and *n*. This means that the parity check symbols carry no information.

So, the average information per code word will be determined only by the average information per each information symbol in the code word. If $H(V)$ denotes the average information per code word, we have obviously

$$H(V) = kH(X) = kH_b(p) \tag{9}$$

Since the probability of a parent node is the sum of probabilities of its children, it results that on a certain level, *i*, $1 \le i \le k$, we have

$$p\ \ N_{2^i-1}\ = (1-p)^i \tag{10}$$

$$p\ \ N_{2^{i+1}-2}\ = p^i \tag{11}$$

On the same level there are a number of C_i^1 nodes of probability $p(1-p)^{i-1}$, a number of C_i^2 nodes of probability $p^2(1-p)^{i-2}$ a.s.o. a number of C_i^{i-1} nodes, of probability $p^{i-1}(1-p)$.

The probability attached to each branch in the graph is equal to the ratio between the probability of the child and the probability of its parent. Since between levels *k* and *n* no nodes splits into another two ones, a parent has only one child, both of the same probability, and so the branch probabilities between levels *k* and *n* will be equal to unity.

On a certain level, *i*, $1 \le i \le k$, the node probabilities will be equal to those of nodes on the level *k*.

III. LINEAR, BINARY, BLOCK CODES SEEN AS SOURCES WITH MEMORY

According to the encoding graph in Fig. 1, when a terminal node is reached the source will deliver another code word. To emphasize this, we link the terminal nodes in the graph with the nodes on the first level, as shown Fig. 2.

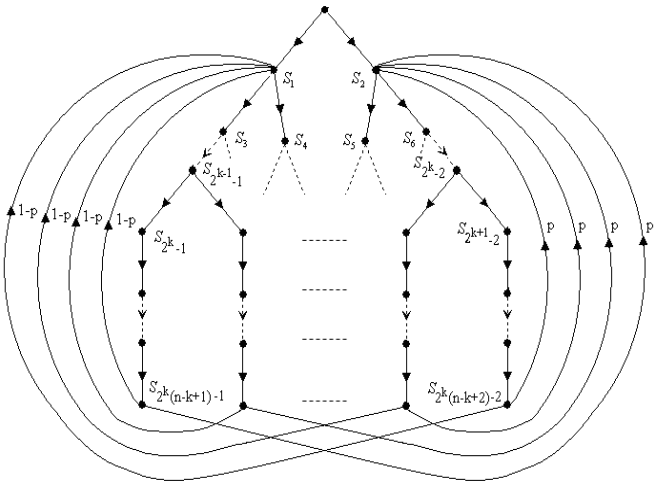

Fig. 2 The graph of the source with memory

Each node N_i in the encoding graph will correspond to a state S_i of the source with memory denoted in the following with $(X|S)$, because the providing probabilities of a binary symbol depend on the state from which it is generated.

The probability of a stationary state is obtained as the ratio between the node probability corresponding to that state in the encoding graph and the average length of the code words [12-14].

As all the code words have the same length, (*n*), the average code word length is also *n*. If $p(S_i)$ denotes the probabilities of the source with memory, then

$$p(S_i) = \frac{p(N_i)}{n},\ i = 1,2,...,2^k(n-k+2)-2 \tag{12}$$

The probabilities of branches in the graph of the source with memory (Fig. 2) are the same as those in the encoding graph (Fig. 1). This is because the probability of a branch in the graph of the source with memory is calculated as the ratio between the stationary probability of the state in which it reaches and that from where it starts. Denoting these states by S_i and S_j, respectively, then

$$p_{ij} = \frac{p(S_i)}{p(S_j)} = \frac{p(N_i)}{p(N_j)}, \tag{13}$$

that is, the ratio between the parent and the child probabilities in the encoding graph.

Let $p(x_j \mid S_i)$, $i = 1,2,...,2^k(n-k+2)-2$, $j = 1,2,$ denotes the probability that the source will deliver the message s_j given the state S_i. Considering the graph in Fig. 1, we can write

$$p(x_j \mid S_i) = \begin{cases} 1-p, \text{for } i = 1,2,...,2^k-2 \text{ and} \\ i = 2^k(n-k+1)-1,...,2^k(n-k+2)-2;\ j=1; \\ p, \text{ for } i = 1,2,...,2^k-2 \text{ and} \\ i = 2^k(n-k+1)-1,...,2^k(n-k+2)-2;\ j=2; \\ 1 \text{ or } 0, \text{ for } i = 2^k-1,...,2^k(n-k+1)-2, \\ \text{so that when } p(x_1 \mid S_i) = 1, \\ \text{then } p(x_2 \mid S_i) = 0, \text{ and conversely} \end{cases} \tag{14}$$

We denote by $p(x_j, S_i)$, $i = 1,2,...,2^k(n-k+2)-2$; $j = 1,2$; the probability that the source with memory is in the state S_i and it supplies the message x_j. These probabilities can be calculated as

$$p(x_j, S_i) = p(S_i)\,p(x_j \mid S_i),$$

$$i = 1,2,...,2^k(n-k+2)-2;\ j = 1,2 \tag{15}$$

Let $p(S_i \mid x_j)$ denote the probability that the source is in the state S_i, if it had supplied the message x_j. These probabilities can be determined by means of relation

$$p(S_i \mid x_j) = \frac{p(x_j, S_i)}{p(x_j)} \tag{16}$$

Theorem 1

The average information per symbol in a code word can be obtained with

$$H(X \mid S) = \frac{k}{n} H_b(p) \tag{17}$$

Proof

$$H(X \mid S) = -\sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(x_j \mid S_i) =$$

$$-\sum_{i=1}^{2^k-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(x_j \mid S_i) -$$

$$+\sum_{i=2^k(n-k+1)-1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(x_j \mid S_i) =$$

$$-\log(1-p) \sum_{i=1}^{2^k-2} \sum_{j=1}^{2} p(x_j, S_i) - \log p \sum_{i=1}^{2^k-2} \sum_{j=1}^{2} p(x_j, S_i) -$$

$$-\log(1-p) \sum_{i=2^k(n-k+1)-1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) -$$

$$+\log p \sum_{i=2^k(n-k+1)-1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) = \frac{k}{n} H_b(p).$$

Theorem 2

For fixed *k* and *n*, the average redundancy per symbol in a code word can be obtained with

$$I(X,S) = \left(1 - \frac{k}{n}\right) H_b(p) \tag{18}$$

Proof

$$I(X,S) = \sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log \frac{p(x_j, S_i)}{p(x_j)p(S_i)} =$$

$$= \sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(S_i) p(x_j \mid S_i) \log \frac{p(x_j \mid S_i)}{p(x_j)} \tag{19}$$

For levels $1 \le i \le k$ and $i = n$, from the encoding graph, we have

$$p(x_1) = p(x_1 \mid S_i) = 1 - p \tag{20}$$

$$p(x_2) = p(x_2 \mid S_i) = p \tag{21}$$

For levels $k \le i \le n-1$, from the encoding graph, we have

$$p(x_1 \mid S_i) = 1;\ p(x_2 \mid S_i) = 0;\ \ p(x_1) = 1 - p\,, \tag{22}$$

$$p(x_1 \mid S_i) = 0;\ p(x_2 \mid S_i) = 1;\ \ p(x_2) = p\,, \tag{23}$$

Considering (5), (12), (20) – (23) in (19), we have

$$I(X,S) = \left(1 - \frac{k}{n}\right) H_b(p).$$

In stationary regime, the state distribution of the source with memory is

$$S: \begin{pmatrix} S_1 & S_2 & \cdots & S_{2^k(n-k+2)-2} \\ p(S_1) & p(S_2) & \cdots & p(S_{2^k(n-k+2)-2}) \end{pmatrix} \tag{24}$$

The source in (24) is discrete, complete and memoryless.

Theorem 3

The average information per stationary state is calculated as entropy

$$H(S) = \log n + \frac{k(2n-k+1)}{2n} H_b(p) \tag{25}$$

Proof

$$H(S) = -\sum_{i=1}^{2^k(n-k+2)-2} p(S_i) \log p(S_i) =$$

$$= -\frac{1}{n} \sum_{j=1}^{k-1} \sum_{i=0}^{j} C_j^i p^i (1-p)^{j-i} \log \frac{p^i (1-p)^{j-i}}{n} -$$

$$-\frac{(n-k+1)}{n} \sum_{i=0}^{k} C_k^i p^i (1-p)^{k-i} \log \frac{p^i (1-p)^{k-i}}{n} =$$

$$= -\frac{1}{n} (1-p) \sum_{j=1}^{k-1} \sum_{i=0}^{j} (j-i) C_j^i p^i (1-p)^{j-i-1} \log(1-p) +$$

$$p \sum_{j=1}^{k-1} \sum_{i=0}^{j} i C_j^i p^{i-1} (1-p)^{j-i} \log p - \sum_{j=1}^{k-1} \sum_{i=0}^{j} C_j^i p^i (1-p)^{j-i} \log n \Bigg] -$$

$$-\frac{n-k+1}{n} \Bigg[(1-p) \sum_{i=0}^{k} (k-i) C_k^i p^i (1-p)^{k-i-1} \log(1-p) +$$

$$+ p \sum_{i=0}^{k} i C_k^i p^{i-1} (1-p)^{k-i} \log p - \sum_{i=0}^{k} C_k^i p^i (1-p)^{k-i} \log n \Bigg] =$$

$$= \frac{k(2n-k+1)}{2n} H_b(p) + \log n$$

Theorem 4

The entropies H(X,S) and H(S|X) are calculated by means of relations:

$$H(X,S) = \log n + \frac{k(2n-k+3)}{2n} H_b(p) \tag{26}$$

and

$$H(S \mid X) = \log n + \frac{k(2n-k+3)-2n}{2n} H_b(p)\,, \tag{27}$$

respectively.

Proof

$$H(X,S) = -\sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(x_j, S_i) =$$

$$-\sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log \Big[p(S_i) p(x_j \mid S_i) \Big] =$$

$$-\sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(S_i) - \tag{28}$$

$$-\sum_{i=1}^{2^k(n-k+2)-2} \sum_{j=1}^{2} p(x_j, S_i) \log p(x_j \mid S_i) = H(S) + H(X \mid S)$$

Considering (17) and (25) from (28), relation (26) results.

Analogously, we have

$$H(S \mid X) = -\sum_{i=1}^{2^k(n-k+2)-2}\sum_{j=1}^{2} p(x_j, S_i)\log p(S_i \mid x_j)$$
$$= -\sum_{i=1}^{2^k(n-k+2)-2}\sum_{j=1}^{2} p(x_j, S_i)\log \frac{p(x_j, S_i)}{p(x_j)} =$$
$$-\sum_{i=1}^{2^k(n-k+2)-2}\sum_{j=1}^{2} p(x_j, S_i)\log p(x_j, S_i) + \quad (29)$$
$$\sum_{i=1}^{2^k(n-k+2)-2}\sum_{j=1}^{2} p(x_j, S_i)\log p(x_j) = H(S) - H(X)$$

Considering (6) and (25), from (29), relation (27) results.

IV. CONCLUSIONS

The main original contributions of the paper are:

1. The analysis of binary block codes as sources with memory, when the symbols 1 and 0 are supplied with probabilities *p* and 1-*p*, respectively.

2. In this approach the following information quantities have been highlighted: the average information per symbol H(X|S), the average redundancy per symbol I(X,S), the average information per stationary state H(S) and the entropies H(X,S) and H(S|X).

3. Between the information quantities calculated above, the following relationships can be established:

$$H(X,S) = H(X) + H(S \mid X) = H(S) + H(X \mid S) \quad (30)$$

$$I(X,S) = H(X) - H(X \mid S) = H(S) - H(S \mid X) = \quad (31)$$
$$= H(X) + H(S) - S(X,S)$$

They can be easily verified by replacing the relations above.

4. We make the following correspondences:

a. The set *X* of binary information symbols and the set of symbols at the input of a discrete memoryless channel;

b. The set of states of the source with memory {S} and the set of symbols at the output of a discrete, memoryless channel.

Then:

- The entropy of the field at the channel input is identical to that of the field consisting of the set of binary symbols, *X*, when information symbols are generated;
- The entropy of the field at the channel output is identical to the state entropy, *H*(*S*);
- The mutual information of the channel is identical to the redundancy on each symbol in a code word;
- The equivocation of the channel is identical to the average information per symbol in a code word, H(X|S);
- The prevarication of the channel is identical to the entropy *H*(*S*|*X*) of the source with memory;
- The joint entropy of the channel is identical to the entropy *H*(*S*,*X*) of the source with memory.

5. It is well known that the number of errors that can be corrected depend on the code redundancy. The larger the code redundancy is, the larger is the number of errors the code can correct. Since the code word redundancy is equal to *nI*(*X*,*S*), the maximum value of the code redundancy, for *k* and *n* fixed, is obtained along with the maximum value of redundancy per symbol in a code word. According to (18), the maximum value of I(X,S), for *k* and *n* fixed, is obtained along with maximum value of binary entropy $H_b(p)$. It becomes maximum when *p*=1/2. This means that to obtain a maximum redundancy per symbol in a code word, for *k* and *n* fixed, the symbols "0" and "1" have to be equally likely. The information symbols, as well as the code words, result also equally likely.

In this case (*p*=1/2) the average information per symbol in a code word becomes maximum, equal to *k*/*n* and so do the average information per code word, equal to *k*.

6. If $H_b(p)$ is the average information per information symbol, obviously, the average information per code word, containing *k* information symbols, is equal to $k\,H_b(p)$. As the code word length is *n*, the average information per symbol in a code word is $\frac{kH_b(p)}{n}$, the same as the entropy in (17).

7. The maximum possible average information per code word is obtained when all the *n* binary symbols in a code word would have the average information of $H_b(p)$, when

$$\max[H(V)] = nH_b(p)$$

In fact, the average information of each code word is equal to $H(V) = kH_b(p)$. Then, the redundancy per a code word is

$$\max[H(V)] - H(V) = (n-k)H_b(p).$$

This means that the redundancy per symbol in a code word is

$$\frac{n-k}{n}H_b(p),$$

that is, the information quantity *I*(*X*,*S*) in (18).

REFERENCES

[1] T. M. Cover, J. A. Thomas. Elements of Information Theory. John Wiley and Sons, 1991.

[2] R. Yeung. A First Course in Information Theory. Kluwer Academic Publisher, 2002.

[3] V. Munteanu, D. Tarniceriu. Elements of Information Theory. Ed. Cermi, 2007.

[4] V. Munteanu, Teoria codarii informatiei. Ed. Politehnium, 2009.

[5] G. R. Gallager, Information Theory and Reliable Communications. Wiley, 1968.

[6] R. E. Blahut, R. Koetter, Codes, Graphs and Systems. Kluwer Academic Publishers, 2001.

[7] V. Munteanu, D. Tarniceriu, "New Possibilities to Characterize Ergodic Sources with Memory", Buletinul Institutului Politehnic Iasi, Tomul L (LIV), Fasc. 1-2, pp. 22-30, 2004.

[8] V. Munteanu, D. Tarniceriu "An Encoding Procedure of Sources with Memory", Buletinul Institutului Politehnic Iasi, Tomul L (LIV), Fasc. 1-2, pp.19-25, 2005

[9] V. Munteanu, D. Tarniceriu, G. Zaharia, "Analysis of lossless compression for a large class of sources of information", Proc. ISSCS, vol. 2, pp. 545-548, July 9-10, 2009, Iasi, Romania.

[10] D. Tarniceriu, V. Munteanu, G. Zaharia, "Information analysis for a large class of discrete sources", , Proc. ISSCS, vol. 2, vol. 2, pp. 553-556, July 9-10, 2009, Iasi, Romania.

[11] G. Zaharia, V. Munteanu, D. Tarniceriu, "Tight bounds on the codeword lengths and average codeword length for d-ary Huffman codes" – part 1, 2, , Proc. ISSCS, vol. 2,, pp. 537-544, July 9-10, 2009, Iasi, Romania.

[12] G. Zaharia, I. Cleju "The series of auxiliary sources and the characterization function of information sources", Jubilee Scientific Session "75 years of electrical education in Iasi" (in Romanian), 16-17 mai, 1986, pp. 21-24.

[13] G. Zaharia, I. Cleju "Monotony theorem of characterization functions" Jubilee Scientific Session "75 years of electrical education in Iasi" (in Romanian), 16-17 mai, 1986, pp. 25-28.

[14] G. Zaharia, I. Cleju "Criteria for establishing the type of secondary information sources", Jubilee Scientific Session "75 years of electrical education in Iasi", (in Romanian), 16-17 mai, 1986, pp. 35-38.