ABBYY PDF Transformer 2.0 Click here to buy www.ABBYY.com

Correcting Codes with Floating Protection

Kuriata Eugeniusz
Institute of Control and Computation Engineering
University of Zielona Gora
email: e.kuriata@issi.uz.zgora.pl

Abstract

In this article a method of information transfer, which provides the higher level of credibility only for the selected symbols in code word has been presented. This system of transfer, which provides for information such parameters, is characterised by parameters of circular start-stop character of transmission. Constructions of asymptotic perfect codes generating have been presented.

Key words: UEP-codes, floating protection, error probability, asymptotic perfect codes, information value

1. Introduction

In order to increase the credibility of the sent information, correcting codes are applied. Codes, in which for all symbols the equal fault-tolerance is provided are the most often used. For these codes the probability of faulty-decoding of code word has been assumed as a quality criterion.

The carried out analyse of dispatching systems show, that information about a state of definite objects is more sensitive than information about other states of these devices, taking under consideration the aspect of security, therefore it should be dispatched to a decision making centre with the correspondingly higher level of credibility.

It has been established in the work [7] that in a specified system for a particular structure it is possible to separate such their states, where their deformity has a more significant influence on the security of the system. As an example, in nuclear power stations if we view them from the perspective of a possible danger of the reactor overheating, the state of “graphite cores being raised up”, which corresponds to the slowing down of the on- going reactions, will be safer than the state of “graphite cores being lowered down”. Of course, by applying other criterions, it may occur that information about the fact that “the core has been raised up” becomes more important. Accordingly, depending on the adopted evaluation criterion, information may have different value. In our further dissertations, this type of information will be referred to as a priority one.

In [7] it has been shown that if in control systems, an order in a commend channel has been transferred to set a particular device (within a group of given devices) into a state more answerable for security, then during a control of this commend completion (report channel) a report which informs that this commend has been not carried out will be a priority one.

From the point of view of security in automatic systems, most often the priority information becomes the one, which gives information about any change in a state of device (in comparison with the previous cycle), with the assumption, that the automatics of security functions correctly.

The analysis has proved that in control systems for transfer of information about the state of executive automatic devices, it is reasonable to use systems with characteristics of sporadic-circular type (start-stop) [9]. These systems are characterised by the fact, that within the set time interval (cycle), information about the state of all devices is sent to a dispatcher centre, while for the significant information (the priority one) a respectively higher level of credibility, in relation to the rest of information, is provided.

As any random bit in an information block in any moment may become a priority one, in practice, correcting codes are applied, which protect the whole information block against the equal amount of errors, while a security level is defined accordingly to demands of the priority bit. Such a solution is not optimal, since by providing for the less significant bits the same level of protection as for the priority bit, we are forced to reduce the rate of transfer. If we are restricted by the capacity of a communication channel, the reduction of transfer rate may make effective management of the technological process impossible. The solution to this problem may be, if it is possible, extending time in the cycle of information gathering.

There will be presented below a method of generating codes with unequal symbols protection (UEP-codes) permitting for a random bit in an informative part of code word to provide a higher level of protection against errors, while for the other bits a correspondingly lower one.

The most common situation is when in a transferred information, only its part is a priority information (in automatics it may be information, for example, about a change in a state of a particular device), therefore, it is advisable to provide a higher level of credibility only for the priority one, and for the rest of information an respectively lower level. It is obvious that the protection level should be adequate to the value of information. By the term of the value of information we mean these maximum profits, which by minimizing losses, may be gained from that amount of information. It is also obvious, that even in the process of incorrect information decoding; some information symbols may be decoded correctly.

Described in literature UEP-codes [1, 2, 16, 17] are used in such cases, where the place of bits is “known” to the encoder and decoder. This limitation leads to a situation, where the practical interest in UEP-codes is inconsiderable. In majority of cases the most significant bit in code word is known for the sending side (coder), while the receiving side does not have any information where the place of such a symbol in code word is. This has contributed to the situation, that regardlessly of bits’ value and their protection level against errors, these bits are treated as equally important, and it means they are of the equal value.

Of course every symbol in code word is significant, but in many cases an error of the most important bit may lead to a situation, which would be much more serious in consequences, than in a case of an error of less significant symbols.

In connection with the inability to provide in the same code word the “higher” level of protection only for a priority symbol (which may be any random symbol), most often in such cases correcting codes are applied, which provide the equal level of protection for all symbols, i.e. such a level as is required for the priory symbol.

2. Problems of Diversified Protection of Information

Using codes protecting information against the equal amount of errors is justified only then, when all symbols in information block, both for a receiver and a sender, are equally important. Treating the likelihood of incorrect decoding of code word as a measure of quality, in a secret way a rule of the equal value of all bits is adopted, and consequently, the equal level of protection for all signs [3].

For UEP-codes protecting particular bits against a different amount of errors, this measure of quality is not applicable. In these codes every *i*-th symbol has a defined t_i protection level. It means that if a code word $\overline{X}$ came into being *f* of independent errors, then these information symbols, for which $t_i \le f$, will be decoded correctly even in a situation when code word will be decoded incorrectly. It has been assumed here that during the decoding of UEP-codes the maximum likelihood method is used.

Currently used symbols for the determined symbols provide an expected level of credibility [2]. However, there are many situations, where the position (the following number in an information block) of the most important bit (bits groups) may alter and in every code word these bits will be located on different positions. It means that a symbol, which is now the most important (priority) one may be any x_p bit (*p* = 1, 2,..., *k*) among *k* information symbols. In the situation where the place of the priority symbol may change, applying UEP-codes, which provide for c_1 of this symbol protection against t_1 errors, for c_2 of this symbol protection against t_2 errors, ..., c_z of this symbol is protected from t_z errors, where $t_1 < t_2 < ... < t_z$ is not purposeful, as it may appear that in a particular word the priority bit is protected in the weakest way.

In the next part a method which allows to use UEP-codes in systems will be presented, in which the most important bit (a group of bits) “is floating” i.e. any information symbol x_p (*p* = 1, 2,…, *k*) may be a priority one.

In the theory of coding, on the assumption that all code words are protected against the equal amount of errors, it is presupposed that all bits in code word are equally important, it is acknowledged in a secret way, that the information enclosed in them is equally important [3, 4, 13, 13]. It means that all information symbols are protected against the equal amount of errors, therefore an equal level of credibility is provided for all of them. On the other hand, most of the used in practice codes, either provide an unequal (different) protection for particular code words or a different protection of particular bits in code words [10, 11, 12]. The problem lies in the fact, that potential codes properties are known only for a small number of them. Therefore, in the coding theory the term of the minimal Hamming distance $d_{\min}$ has been used, which represents a parameter defining the guaranteed code correcting properties, whereas this parameter refers to bit in code word or of code word, which has the weakest protection.

It means that a code providing for the c_1-th symbol a protection against t_1 errors, and for the c_2-th symbol a protection against t_2 errors, ..., c_z-th symbol is protected against t_z errors , where $t_1 < t_2 < ... < t_z$, protects code word against t_1 errors.

From the analysis of the value of transmitted information [6] it emerges that for the transfer of information concerning a state of automatics devices, it is justified to use the correcting codes, which provide an unequal protection for information symbols (or group of symbols). Employing such codes enables to provide for the priority information a protection against $t_2 = \frac{d_2 - 1}{2}$ errors, whereas for the rest part of information against $t_1 = \frac{d_1 - 1}{2}$ errors, where $t_1 < t_2$, while the priority and non-priority symbols are located in the same code word (fig.1).

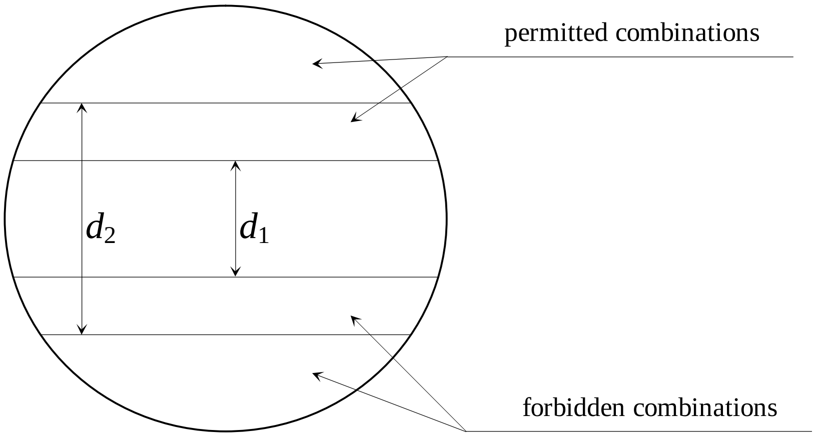


Fig.1 Information encoding giving consideration to its value

The figure 1 shows that symbols protected by a distance d_2, as a result of information deformity, have to overcome a greater “trench” than symbols protected by a distance d_1. It means that in a case when $t_1 < f < t_2$, then symbols protected by a distance d_2 will be decoded correctly, while the symbols protected by a distance d_1 may be distorted.

It is justifiable to apply correcting codes, which differentiate the level of protection for particular code words (an unequal commend protection) in a commend channel. In this situation to encode priority commends, the aptly selected code words have been used, which are more resistant to noise interference, and as a result of this they can be provided with the higher level of credibility (fig.2).

It is visible on the figure 2 that in order to encode information in a not associated commend, for example, in a case of a security hazard, it is possible to employ codes or code words protecting information against the smaller amount of errors (d_1), whereas for the priority information to use code words with protection d_2. These codes have been widely applicable in broadband communication channels [5].

There are two report files $\overline{X}$ and $\overline{Y}$. For each pair x and , $x \in \overline{X}$ and $y \in \overline{Y}$, there exists such a code word $\overline{K} = z(x, y)$ of the length n, from which it is possible to reproduce x from a set $\overline{X}$ and y from a set $\overline{Y}$, unless the amount of errors is bigger than, correspondingly, t_1 and t_2 (fig.3).

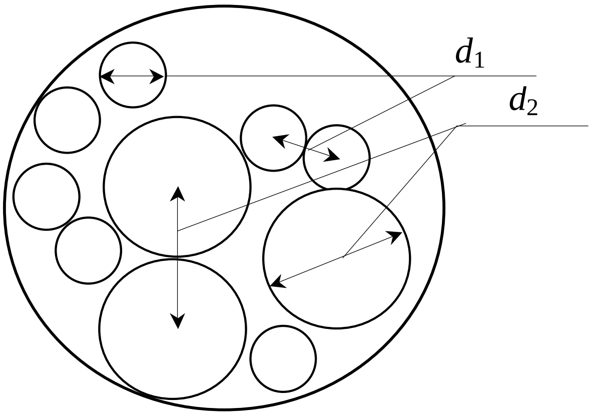


Fig.2 Codes with unequal protection of commend

Code

$$\hat{K} = \left\{\hat{R} = z(x, y); x \in \hat{X}, y \in \hat{Y}\right\} \qquad (1)$$

has parameters

$$d_1\left(\hat{R}\right) = \min_{y_1, y_2 \in \hat{Y}} \left\{ \min_{\substack{x_1, x_2 \in \hat{X} \\ x_1 \neq x_2}} d\left[z(x_1, y_1), z(x_2, y_2)\right] \right\} \qquad (2)$$

and

$$d_2\left(\hat{R}\right) = \min_{x_1, x_2 \in \hat{X}} \left\{ \min_{\substack{y_1, y_2 \in \hat{Y} \\ y_1 \neq y_2}} d\left[z(x_1, y_1), z(x_2, y_2)\right] \right\} \qquad (3)$$

where

d(α, β) – the Hamming distance between α and β.

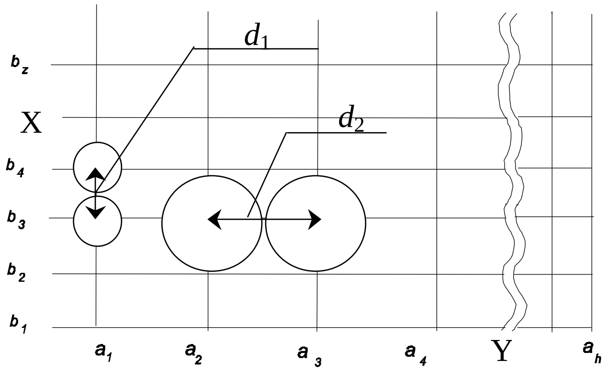


Fig.3 Graphical representation of unequal protection of commend

Let us analyse the set of all code words $\overline{K} = \bigcup_{y \in \overline{Y}} \overline{K}_y$ which are the sum of subsets $\overline{K} = \left\{z(x, y); x \in \overline{X}\right\}$. Every subset $\overline{K}_y$ represents a code with a distance d_2.

As $d\{z(x_1, y), z(x_2, y)\} \geq d_1$, while $x_1 \neq x_2$, two optional subsets $\overline{K}_y$ and $\overline{K}_{y_k}$, where $y \neq y_k$ are situated from each other in a distance at least d_2.

Codes with such properties may be generated by appropriate connection of other codes, for instance

$$|G| = \left\| \begin{matrix} \dfrac{|G_1|}{|0|} \end{matrix} \; |G_2| \right\| \qquad (4)$$

where

$|G_1|$ - matrix of code with parameters (n_1, k_1, d_1) of dimensions $(n_1 \times k_1)$,

$|G_2|$ - matrix of code $\overline{K}_2$ with parameters (n_2, k_2, d_2) of dimensions $(n_2 \times k_2)$,

$|0|$ - matrix of dimensions $(n_1 \times (k_2 - k_1))$.

Matrix $|G|$ will be of dimensions $((n_1 + n_2) \times k_2)$.

By means of this type of solutions, with a defined amount of errors, code word including priority information will decode it correctly, whereas non-priority information may have errors.

3. Generating of UEP-codes by a method of orthogonal matrices

On the basis of the presented [15] constructive methods of connecting matrices it may be possible to generate UEP-codes. The recognized codes will be taken as base matrices

There has been presented below a method of generating asymptotically perfect codes, which are characterised by the property that: for one optional information symbol there will be a possibility to provide a higher level of protection, while for other symbols a protection against one error will be provided.

As a base code the Hamming code H_N with parameters $n = 2N - 1$, $k = 2N - 1 - N$, $d = 3$ will be employed. Let us create matrix of type C_{III} (for $N \geq 3$) of dimension $((2^N + N + 1) \times (2^{N+1} + N + 1))$ [15].

$$C_{III} = \left\| \begin{array}{c|c|cc} D_1 & 0_1 & \multicolumn{2}{c}{H} \\ \hline 0_2 & D_2 & 0_3 & \multicolumn{1}{|c}{H^T} \end{array} \right\|, \qquad (5)$$

where

D_1 - diagonal matrix of dimensions $((N + 1) \times (N + 1))$,

D_2 - diagonal matrix of dimensions $(2^N \times 2^N)$,

0_1 - zero matrix $((N + 1) \times 2^N)$,

0_2 - zero matrix $(2^N \times (N + 1))$,

0_3 - zero matrix $(2^N \times (2^N - N - 1))$,

H - control matrix of the Hamming code $((N + 1) \times 2^N)$,

H^T - transposed control matrix of the Hamming code $(2^N \times (N + 1))$.

After shifting some of columns in matrix $\overline{C}_{III}$ we will get matrix in a form of

$$C'_{III} = \left\| \begin{array}{c:c} D & \begin{array}{c} H \\ \hdashline \begin{array}{c:c} 0 & H^{T} \end{array} \end{array} \end{array} \right\| \qquad (6)$$

where:

H - control matrix H of the Hamming code $\left((N+1)\times 2^{N}\right)$,

H^{T} - transposed control matrix of the Hamming code $\left(2^{N}\times(N+1)\right)$.

0 - zero matrix $\left(2^{N}\times\left(2^{N}-N-1\right)\right)$,

D - diagonal matrix of dimensions $\left(\left(2^{N}+N+1\right)\times\left(2^{N}+N+1\right)\right)$,

The generated codes (fig.4) will have two groups of symbols protected in different ways: one group will protected against t_1 errors, and the other group – against t_2 errors, while $t_1 < t_2$.

Parameters of the generated codes will be as following:

- length of code combination $n = 2^{N+1} + N + 1$,
- amount of symbols protected against t_2 errors: $k_1 = N + 1$,
- correcting property of symbols protected against t_2 errors $d_2 = 2N - 1$,
- amount of symbols protected against t_1 errors $k_2 = 2^{N} - 1$,
- correcting property of symbols protected against t_1 errors $d_2 = 3$.

There are situations when applying UEP-codes with so-called “floating protection” is recommended – where one optional symbol, in comparison with other symbols, has the higher correcting property. Therefore, in order to be able to perform the set tasks - a method of generating the optimal linear UEP-codes has been worked out. Matrices connection will take place accordingly to the following rule [15]:

$$C_{IV} = \left\| \begin{array}{c:c} 1 & \begin{array}{c} \tilde{H}' \\ \hdashline \tilde{H}'^{T} \end{array} \end{array} \right\| \qquad (7)$$

Dimensions of matrix C_{IV} will be $\left(\left(2^{N}+N+1\right)\times\left(2^{N+1}+N+1\right)\right)$, while constituent submatrices:

1 - diagonal identity matrix of dimensions $\left(\left(2^{N}+N+1\right)\times\left(2^{N}+N+1\right)\right)$;

$\tilde{H}'$ - control matrix of the extended Hamming code of dimensions $((N+1)\times 2^{N})$;

$\tilde{H}'^{T}$ - transposed parity tests matrix of the extended Hamming code of dimensions: $\left(2^{N}\times(N+1)\right)$ while columns of the matrix $\tilde{H}'^{T}$ are positioned opposite the ones vectors of the matrix $\tilde{H}'$ (fig. 4).

It is known that the minimal weight of any optional code word $\min\limits_{i}\left(wt\left(\overline{X}\right)\right)$, where $i = 1,\ldots,M$,cannot be smaller than $d_{\min}$ of the code. In a code generated by means of the structure of matrix $C_{IV} = \left\| \begin{array}{c:c} 1 & \begin{array}{c} \tilde{H}' \\ \hdashline \tilde{H}'^{T} \end{array} \end{array} \right\|$, presented on the example of dependency (7) and illustrated on the figure 4, there is one row (row x6), which has the weight less than

$d_{\min}=3$. When we remove this row (x6) and one column (06) the generated code will reach the requested parameters.

$$C'_{IV}\left|\begin{array}{ccccccccccccccccccccccccccccccccccccc}
1&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0\\
0&1&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&1&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0\\
0&0&0&1&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0\\
0&0&0&0&1&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{0}&\mathbf{1}\\
0&0&0&0&0&\mathbf{0}&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1\\
0&0&0&0&0&\mathbf{0}&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&1&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&1\\
0&0&0&0&0&\mathbf{0}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&1&0&0&0&1&0&1&1&1
\end{array}\right|\begin{array}{l}
x_{1}\\ x_{2}\\ x_{3}\\ x_{4}\\ x_{5}\\ \mathbf{x_{6}}\\ x_{7}\\ x_{8}\\ x_{9}\\ x_{10}\\ x_{11}\\ x_{12}\\ x_{13}\\ x_{14}\\ x_{15}\\ x_{16}\\ x_{17}\\ x_{18}\\ x_{19}\\ x_{20}\\ x_{21}
\end{array}$$

$$\begin{array}{ccccccccccccccccccccccccccccccccccccc}
01&02&03&04&05&\mathbf{06}&07&08&09&10&11&12&13&14&15&16&17&18&19&20&21&22&22&23&24&25&26&27&28&29&30&31&32&33&34&35&36
\end{array}$$

Fig.4 Code generation by means of construction $C'_{IV}=\left\|\begin{array}{c:c}1&\begin{array}{c}\tilde{H}'\\ \hdashline \tilde{H}'^{T}\end{array}\end{array}\right\|$

The matrix C_{IV} after reorganizing some columns (fig.5) will be of size $\left(2^{N}+N\right)\times\left(2^{N+1}+N\right)$.

The employed construction will have a form of:

$$C'_{IV}=\left\|\begin{array}{c:c:c:c}1_{H}&0&\multicolumn{2}{c}{\tilde{H}'}\\ \hdashline 0_{p}&1&0_{u}&\tilde{H}'^{T}\end{array}\right\| \tag{8}$$

where

1_{H} - diagonal matrix of dimensions $\left((N+1)\times(N+1)\right)$,

0 - zero matrix $\left((N+1)\times\left(2^{N}-1\right)\right)$,

$\tilde{H}'$ - control matrix of the extended Hamming code $\left((N+1)\times 2^{N}\right)$,

0_{p} - zero matrix of dimensions $\left(\left(2^{N}-1\right)\times(N+1)\right)$,

1 - diagonal ones matrix of dimensions $\left(\left(2^{N}-1\right)\times\left(2^{N}-1\right)\right)$,

0_{u} - zero matrix of dimensions $\left(\left(2^{N}-1\right)\times\left(2^{N}-n-2\right)\right)$,

$\tilde{H}'^{T}$ - transposed matrix of the extended Hamming code $\left(\left(2^{N}-1\right)\times(N+1)\right)$.

$$G_{IV} = \begin{vmatrix}
\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&0&0&\mathbf{1}&0&0&0&0\\
0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&\mathbf{1}&0&0&0\\
0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&0&\mathbf{1}&\mathbf{1}&0&0&\mathbf{1}&\mathbf{1}&0&0&0&\mathbf{1}&0&0\\
0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&\mathbf{1}&0&\mathbf{1}&\mathbf{1}&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&0&0&\mathbf{1}&0\\
0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&\mathbf{1}\\
0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&\mathbf{1}&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&\mathbf{1}&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&0&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&0&\mathbf{1}&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&0&0&0&0&0&0&0&0&0&0&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}&\mathbf{1}
\end{vmatrix}$$

Fig.5. Generating matrix of code G_{IV} with floating protection

For codes that protect all symbols against the equal amount of errors, defining the biggest code, which includes the greatest amount of code words with the assigned parameters, i.e. with the assigned length *n* and the minimal Hamming distance *d*, is done by means of the evaluation of the upper and lower bounds of size *M (n, d)*.

The upper bound is the Hamming bound (bound of spherical packing), which is defined by the dependency

$$q^n \geq M \sum_{i=0}^{t} C_n^i (q-1)^i ,$$

while the lower bound is the Gilbert-Sachs bound

$$q^r \geq \sum_{i=0}^{d-2} C_{n-1}^i (q-1)^i .$$

For perfect codes these bounds are overlapping [15].

The code generated by means of this construction (7) reaches the Hamming upper bound and the Gilbert lower bound [7, 15].

5. Codes with floating protection

Let it exist $\wp$ a linear systematical UEP-code, assigned by the matrix C_{IV}, in which the symbol on the first position is protected against t_2 errors (u_0), while the other $(k-2)$ of symbols are protected against t_1 errors, where $(t_t > t_1)$, by assumption that only the decoder "knows" on which position of the report $\overline{U} = u_0 u_1 \ldots u_{k-1}$ the priority symbol x_p is. In order to have on the sender's site a possibility to identify the priority symbol, it is necessary to send additional information, which will enable to define the place (the position) of the priority symbol x_p $(p = 0,1,2,...,k-1)$ in code word $\overline{X}$. It is necessary to transfer additional number of symbols for the correct identification of the priority symbol

$$N = \log k\,. \qquad (9)$$

For these N symbols the equal level of protection, as for the priority bit, should be provided, as any errors in this group of bits may have similar consequences, as in the case of the priority bit error. These N bits defining the number of the priority bit in code word will be referred to as a numerator.

If there exists $\Im$ UEP-code assigned with the parity test matrix C_{IV}, where the first $(N+1)$ rows represent symbols protected against t_2 errors, then the other $(k-2)$ rows represent symbols protected against t_1 errors (fig.6).

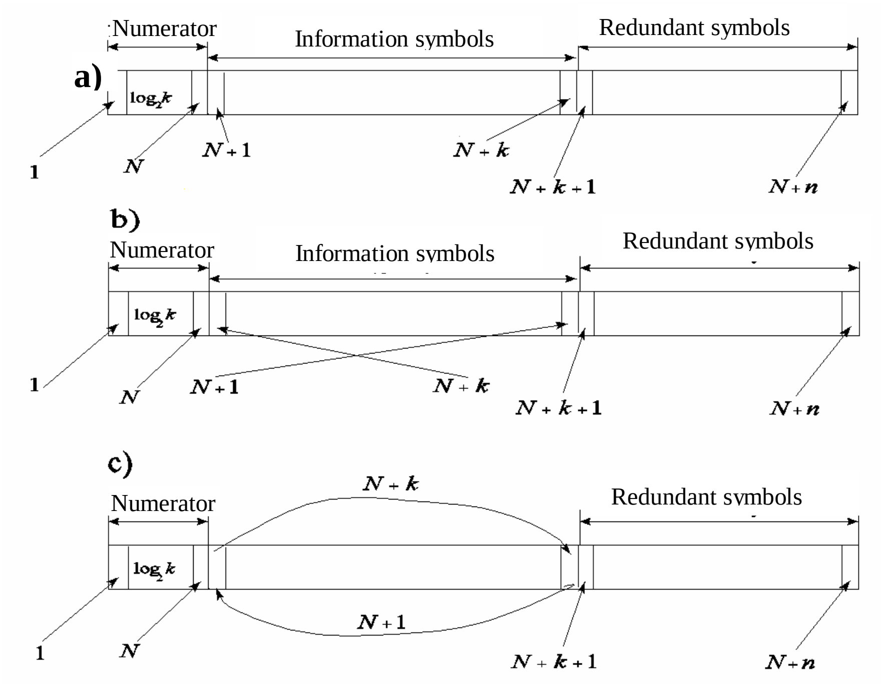


Fig. 6. Coding with floating protection

As the coder knows the place of the priority bit in a report $\overline{U}$, before the proper encoding takes place, it may perform an transposition between positions of the priority symbol u_p which lie on the p –th position of report $\overline{U}$ with the symbol being on the first position u_0 of the report $\overline{U}$. As a result of transposition of the mentioned bits we will get an transformed report $\overline{U}'$. After the operation of transposing the considered bits, the proper encoding of the report $\overline{U}'$ into code word $\overline{X} = x_0, x_1, ..., x_{k-1}, ..., x_n$ takes place.

$$\begin{bmatrix} x_1 \\ \vdots \\ x_k \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \bar{I}_{k+N} \\ \hline -\bar{A} \end{bmatrix} \cdot \begin{bmatrix} N_1 \\ \vdots \\ N_N \\ U_0 \\ \vdots \\ U_{k-1} \end{bmatrix} \tag{10}$$

where:

$\bar{I}_{k+N}$ - diagonal matrix of dimensions $\{(k+N)\times(k+N)\}$,

$-\bar{A}$ - assigned matrix of dimensions $\{(n-k-N)\times(k+N)\}$,

$N_1 \ldots N_N$ - numerator,

$u_0, u_1 \ldots, u_{k-1}$ - commend $\hat{U}$.

The idea of such encoding is performed in such a way, that if the priority symbol is situated in a group of the lower protected symbols, then it is relocated to a group of the higher protected symbols, and a non-priority symbol from the higher protected group is taking the place of the priority one. Encoding and decoding is performed by means of former assigned matrices.

In [6, 7] a mechanism of coding with so called “floating protection” has been presented, which enables to provide the higher level of credibility for the priority bit group, regardless of the place of this group in code word.

Decoding of a vector $\overline{Y} = \overline{X} + \overline{E}$, where $\overline{E}$ means an error vector may be performed by using any known method. After the operation of decoding we are making a transposition of *p*-th bit with u_0 bit in a received modified commend $\overline{U}'$, and consequently we will get a commend $\overline{U}$.

On the fig.6b an operation of making a transposition of the priority bit has been presented, when it is on the weakly protected positions, whereas on the fig. 6c an operation of transposition of the priority bit positions in a commend $\overline{U}'$ with a bit, which is situated on the first position in a commend. The number of the priority bit is encoded in a numerator.

Thanks to applying such a coding, a particular bit (or a group of bits) will have the required level of credibility. From the fig.7 it is visible that any bit (from *k* symbols) in a commend $\overline{U}$ may become the priority bit (group of bits).

In [8] it has been established that with the assigned correcting properties of a code, i.e. when it is essential to protect $k_1$symbols against t_1 errors and k_2 symbols against t_2 errors , while $t_2 > t_1$ and $R_1 > 0$, where $R_1 = \frac{k_1}{n}$, with not too big *n*, the amount of redundant symbols satisfies an inequity:

$$r \geq k_1 \log_2 n + c(t_1, t_2, R_2), \tag{11}$$

where constant $c(t_1, t_2, R_2)$ is dependent from t_1 ,t_2 as well as from R_2, and is not dependent from *n*.

In the event of necessity to protect all $(R_1 + R_2)n$ information symbols against t_2 errors, the redundancy will not be smaller than:

$$r = t_2 \log_2 n + c(t_2). \qquad (12)$$

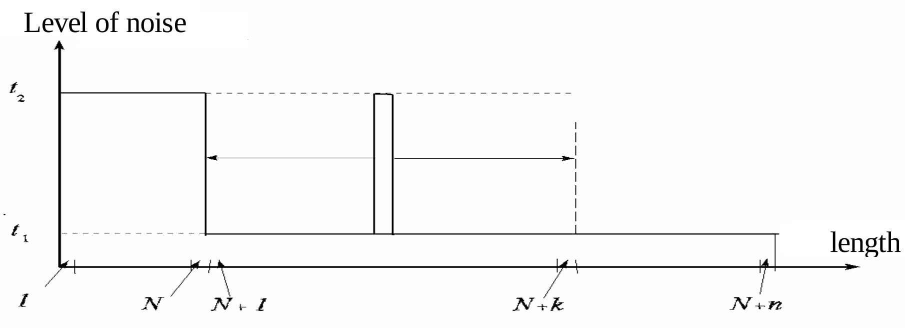


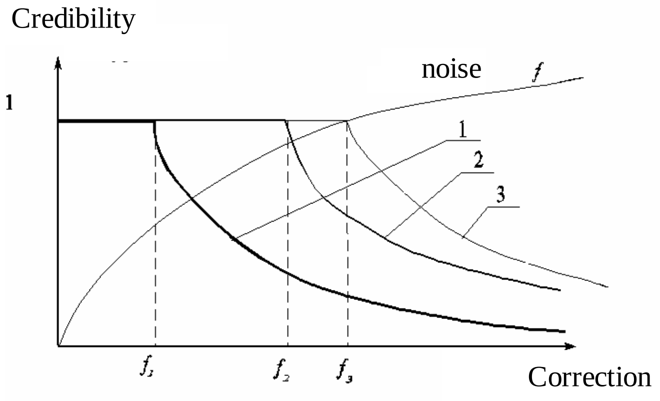
Fig. 7 Protection of the priority bit in command $\overline{U}$

It is important to observe that the constant $c(t_2)$ is dependant from t_2, but not from *n*.

It comes out that in a case of existence of even a small amount of information symbols, which will be provided with the lower level of credibility $(t_2 > t_1)$, we may quite significantly reduce the amount of redundant symbols.

6. Probability of errors

The probability of code word miscoding has been taken as a quality criterion for codes protecting all symbols against the equal number of errors,



1 - Credibility of code word and the weakest protected bit

2 - Non-priority group

3 - Priority group

Fig. 8. Information credibility dependency from code correction properties and a level of noise

In the case of unequal symbols protection codes (UEP-codes), every information symbol has an individual level of protection, and the probability of faulty-decoding of code word, as a quality criterion for the sent information, is not applicable. In this situation it is recommended to use the bit error probability, which will be different for particular groups of symbols.

If the protection level for a bit with the lowest credibility is t_1 errors, then ipso facto, the probability of code word errors of UEP-code is $d_{\min} = t_1$. It means that if the noise level is bigger than the correcting properties of the weakest protected bit, but is smaller than the correcting properties of the priority bit ($t_1 < f < t_2$), then it may be possible that decoding of code word may be incorrect, however in this word , the priority bits will be decoded correctly.

In case, where a code protects particular bits against different amount of errors, then the probability of faulty-decoding of code word is defined in the same method as for symbols protected against the smallest amount of errors $t_1^{\min}$

$$P_{bl} = 1 - \sum_{i=0}^{t_1^{\min}} Ci_n\, p^i (1-p)^{n-i},$$

whereas the probability of any individual bit error is defined by the dependency

$$\frac{1}{z} P_{bl} = P_{symb} = P_{bl} = 1 - \sum_{i=0}^{t_1^{\min}} Ci_n\, p^i (1-p)^{n-i},$$

where

z – amount of the symbols in a group with the weakest protection.

The dependency of information credibility from code correcting properties and a level of noise has been presented on the fig.8.

6. Conclusion

Applying UEP-codes enables to achieve for the selected (the priority) symbols the adequately high level of credibility, while for the other symbols, correspondingly, a lower one. It also makes possible the creation of new class of information transfer systems. In such a system, in a report channel a significant information (the level of importance is established accordingly to the applied criterion), is provided with the adequately higher level of credibility (as in start-stop systems), while the auxiliary information with respectively lower ones (as it is in systems with the circular information gathering). In a commend channel it is advisable to apply codes, which protect particular commends in a different way.

7. Bibliography

1. Altenkamp D. and Mehlhorm K. – Codes: Unequal Probabilities, Unequal Letter Cost, Journal ACM, vol. 27 N3 1980r

2. Masnik B. And Wolf J., On linear Unequal Error Protection Codes, IEEE Trans. Inf. Th., vol 13 N4.

3. Bloch E.L., Zjablov V.V. – Obobshchonnyje kaskadnye kody, Moskva, Svjaz, 1976

4. Brillouin L., Science and Information Theory, New York, London, Academic Press Inc., 1956.

5. Cower T. M., Shirokoveshchatelnyje kanaly, Kiberneticheskij Sbornik, N11, Moskva, Mir, 1974

6. Kuriata E., Correcting Codes with Floating Protection, in: The National Telecommunication and Teleinformatics Symposium '95. Bydgoszcz, 1995.

7. Kuriata E., Isledowanie i princypy postroienia sredstv kanaloobrazovania dla sistem dispetcherskoi centralizaci, thesis PhD, MIIT, Moskva, 1982.

8. Kacman G. L., Granicy moshchnosti linejnych kodov s nieravnoj zashchitoj informacionnych simvolov, PPI, N2, 1980.

9. Kuriata E., Information Reliability and Security, in: III PRST, Department of Telecommunications AGH Krakow,1996.

10. Kuriata E., About Codes with Unequal Symbols Correction, unpublished work, library. IKSAiP, 0162/V/BP/85.

11. Kuriata E., Decoding the Linear Codes, in: Performance Evaluation, Reliability & Exploatation of Computers Systems, Ossolineum, 1989.

12. Kuriata E., Method of Information Encoding with Differential Protection Level against Errors of Particular Bit Groups of Transferred Information, patent RP 155077.

13. MacWilliams F. J. and Sloane N. J. A., The Theory of Error - Correcting Codes, Nord-Holland publishing company, 1977

14. Peterson W. W. And Weldon E. J., Error – Correcting Codes, 1972.

15.Kuriata E., Constructing Codes with Unequal Error Protection of Two Sets of Information Symbols, Int. Journal AMCS (being printed).

16. Wolf J.K. – Error Locating Cades – a new Concept in Error Control, IEEE Inf. Th. vol. 9, N2, 1963.

17. Zinoviev V.A., Zjablov V.V.– Kody z nieravnoj zashchitoj informacionnych simvolov, PPI, N3, 1979r.