Binary Quasi Equidistant and Reflected Codes in Mixed Numeration Systems

Evgeny Beletsky[1] and Anatoly Beletsky[1]

[1]Department of Electronics, National Aviation University of Kiev,
1, av. Cosmonaut Komarov, 03680, Kiev, Ukraine

ebeletskiy@gmail.com, abelnau@ukr.net

Abstract. The problem of constructing quasi equidistant and reflected binary Gray code sequences and code in a mixed factorial, Fibonacci and binomial numeration systems is considered in the article. Some combinatorial constructions and machine algorithms synthesis sequences, based on the method of directed enumeration are offered. For selected parameters of sequences all quasi equidistant (for individual cases - reflected) codes with Hamming distance equal to 1 are found.

Keywords. Reflected codes, quasi equidistant sequence, Hamming distance

Key terms. Research, CodingTheory, MathematicalModelling

1 Introduction

Coding theory is one of the most important areas of modern applied mathematics. Beginning of the formation of mathematical coding theory dates back to 1948, when it was published a famous article by Claude Shannon [1]. The growth of codes originally was stimulated by tasks of communication. Later constructed codes found many other applications. Now codes are using to protect data in a computer memory, cryptography, data compression, etc.

The work is devoted to a rather small, but extremely important for applications subset of so-called quasi-equidistant and reflected codes. The class of quasi equidistant codes are sequences of uniform (i.e., containing the same number of bits) of binary code combinations in which any adjacent (neighboring) code sets (words) are at the same Hamming d distance equal to a fixed number of natural numbers (i.e. d = 1, 2, …) [2]. Equidistant sets include such codes in which any two words (code combinations) are at the same distance d [3].

Finally, we shall refer to the reflected subset quasi equidistant codes with distance d =1, the formation of which is based on the principle of mirror reflection? [4]. But if we restrict ourselves to only one mirror, the code sequence will contain the original sequence, after which is the same sequence just re-written in reverse order, which is unacceptable, since it leads to code repetition. The elimination of repetition can be provided by initial expansion of the number of digits combinations. The essence of the "mirror" reflection of the expansion is explained below as an example of canonical reflected Gray codes and in other sections of this article.

The main objective of this study is to develop algorithms for constructing quasi-equidistant and reflected binary Gray codes as well as code sequences in a mixed factorial, Fibonacci and binomial bases. The method of direct enumeration is the base of algorithms of computer sequences synthesis.

2 Basic of Number System

The history of discrete mathematics and computer science is directly related to the development and introduction of newer principles of representation and encoding digital information, which are based on the *numeration system of numbers*. By a numeration system we understand the way of image sets of numbers using a limited set of characters that form its alphabet, in which the characters (elements of the alphabet) are located in the established order, occupying a certain positions [5]. Any numeration system should be composed of a finite set of non-negative numbers — a range that it encodes. It always includes the number 0 and then follows the natural numbers starting with 1 [6].

There are various numeration system (as well as methods for their classification), whose number is constantly growing. All systems can be divided into the following main classes: positional, not positional and mixed. In the positional numeration systems the same numeric characters (digit) has different meanings in its description depending on the location (level) where it is resides.

By positional numeration system is generally understood the p numeration system, which is defined by an integer $p > 1$ — is called a base of numeration system. Unsigned integer N in p numeration system is represented as a finite linear combination of powers of

$$N = \sum_{k=1}^{n} \alpha_k p^k \ , \tag{1}$$

where α_k are integers satisfying the inequality $0 \le \alpha_k \le (p-1)$, n – the number of digits of the number. The simplest examples of positioning systems (1) can be binary, decimal, and other numeration systems.

In *no positional numeration systems* the value which indicated by the digit does not depend on the position in a number. At the same time the system may impose restrictions on the position of numbers, for example, that they are in descending order. The Roman and many other systems belong to not positional systems.

The *mixed numeration system* is a generalization of the p system, and often refers to the positional numeration systems. The base of mixed numeration system is an increasing sequence of numbers p_k, $k = 1, 2, \dots$, and each N number is presented like linear combination:

$$N = \sum_{k=1}^{n} \alpha_k p_k \ ,$$

there are some restrictions exist for α_k coefficient.

One of the known examples of the mixed system is a factorial numeration system, in which the bases are the sequence of factorials $p_k = k!$. Another commonly used *Fibonacci* numeration system is a system that is based on Fibonacci numbers. The *Binomial system* in the form in which it is presented in the relevant section of this article, we will also include to a mixed numeration system.

A positive integer is depicted in an arbitrary numeration system as a sequence of symbols $[N] = \alpha_n \alpha_{n-1} \ldots \alpha_k \ldots \alpha_2 \alpha_1$, where $[N]$ - the number representation in this numeration system, besides each α_k symbol takes r_k bit in general case (if binary alphabet is using).

Note the following general characteristics of quasi equidistant codes with Hamming distance $d = 1$. Let's agree each code sequence starts with zero code. And as result of this agreement the following code after the zero code should be placed with weights 1 and 2, and Further weight codes must alternate *even* (E) — *odd* (O) under the scheme

$$012\text{OEOE} \ldots \text{E(O)} \ . \tag{2}$$

Scheme (2) is a symbolic form of the tree sequence code combinations. Let's n_e and n_o to be the amount of even and odd code words in a sequence. If the sequence (2) ends up with odd code combination this means $n_e = n_o$, and if even — $n_e = n_o + 1$. This becomes evident:

Statement 1. *Inequality*

$$0 \le (n_e - n_o) \le 1 \ , \tag{3}$$

is a necessary (but not always sufficient) condition for the construction of quasi equidistant codes.

3 Sequences of Gray Codes

Classic Gray codes [7] may be called canonical, since for arbitrary length sequence of combinations are not only quasi equidistant, but also reflected. Let's $G(n)-$ sequence of n-bites classical Gray codes. To construct $(n+1)-$ bites reflected Gray Codes, let's us note as $G_{rc}(n+1)-$ codes, it is just enough to prefix for each source code $G(n)$ the 0 digit and 1 to the left of code group $G^R(n)$ constructed by reflected (reflex or reverse) mirror of $G(n)$ sequence, i.e.

$$G_{rc}(n+1) = 0G(n) \,\|\, 1G^{R}(n) \ , \qquad (4)$$

where || - is a symbol of concatenation (conjunction of sequences).

According to (4), $G_{rc}(n+1) \equiv G(n+1)$ and as a result sequences of Gray codes of $G(n)$ number of digits $n \geq 2$ are both quasi equidistant and reflected, and besides the line of reflection goes through $2^{n-1}-$ and $(2^{n-1}+1)-$ code combinations. On the basis of the canonical code $G(n)$, $n \geq 2$, the equidistant Gray codes can be constructed. For example, Tab. 1 show the three 12-bit code quasi equidistant sequences, one of which corresponds to the canonical version of the Gray code.

The first six variants of sequences in the table constructed of canonical option 1 as a result of a variety column rearrangement saving the Hamming distance $d = 1$ of related code combinations. Variants 7-12 are formed as a result of inverse none zero rearrangements of code combinations from appropriate variants 1-6.

Table 1. Three bit quasi equidistant Gray code

Variants of sequence											
1	2	3	4	5	6	7	8	9	10	11	12
000	000	000	000	000	000	000	000	000	000	000	000
001	100	100	001	010	010	100	001	010	010	001	100
011	110	101	101	110	011	101	101	110	011	011	110
010	010	001	100	100	001	111	111	111	111	111	111
110	011	011	110	101	101	110	011	011	110	101	101
111	111	111	111	111	111	010	010	001	100	100	001
101	101	110	011	011	110	011	110	101	101	110	011
100	001	010	010	001	100	001	100	100	001	010	010

The first six variants of sequences in the table constructed of canonical option 1 as a result of a variety column rearrangement saving the Hamming distance $d = 1$ of related code combinations. Variants 7-12 are formed as a result of inverse none zero rearrangements of code combinations from appropriate variants 1-6. As follows from Tab. 1 the only variants 1 (canonical) and 6 of Gray codes belong to a set of three bites reflected codes. At the same time each three bite sequence by (4) statement produce subset of four bite reflected Gray codes. Thereby it is true:

Statement 2. *All amounts* $L_{\text{от}}^{(G)}(n)$ *of reflected Gray codes of n number of digits is defined by*

$$L_{rc}^{(G)}(n+1) = \begin{cases} n, & \text{if } n \leq 2; \\ 2n!, & \text{if } n \geq 3. \end{cases}$$

3 Factorial Sequence

The integer positive number N in factorial number of numeration system can be represented as

$$N = \sum_{k=1}^{n} \alpha_k k!, \quad 0 \le \alpha_k \le k \tag{5}$$

where $k = 1, 2, \ldots, n;\quad 0 \le \alpha_k \le k.$ Extended form of (5) statement is

$$N = \alpha_n \cdot n! + \alpha_{n-1} \cdot (n-1)! + \ldots + \alpha_2 \cdot 2! + \alpha_1 \cdot 1!\ , \tag{6}$$

Statement (6) is so called numerical, or digital, function [8] of factorial system. There are first 120 decimal numbers (Tab. 2) defined by their α_k coefficients in factorial numeration system.

Table 2. Binary representations of decimal numbers of factorial numeration system

N	$[N_k]_{Fakt}$	N	$[N_k]_{Fakt}$	N	$[N_k]_{Fakt}$	N	$[N_k]_{Fakt}$	N	$[N_k]_{Fakt}$
0	0	24	100000	48	1000000	72	1100000	96	10000000
1	1	25	100001	49	1000001	73	1100001	97	10000001
2	10	26	100010	50	1000010	74	1100010	98	10000010
3	11	27	100011	51	1000011	75	1100011	99	10000011
4	100	28	100100	52	1000100	76	1100100	100	10000100
5	101	29	100101	53	1000101	77	1100101	101	10000101
6	1000	30	101000	54	1001000	78	1101000	102	10001000
7	1001	31	101001	55	1001001	79	1101001	103	10001001
8	1010	32	101010	56	1001010	80	1101010	104	10001010
9	1011	33	101011	57	1001011	81	1101011	105	10001011
10	1100	34	101100	58	1001100	82	1101100	106	10001100
11	1101	35	101101	59	1001101	83	1101101	107	10001101
12	10000	36	110000	60	1010000	84	1110000	108	10010000
13	10001	37	110001	61	1010001	85	1110001	109	10010001
14	10010	38	110010	62	1010010	86	1110010	110	10010010
15	10011	39	110011	63	1010011	87	1110011	111	10010011
16	10100	40	110100	64	1010100	88	1110100	112	10010100
17	10101	41	110101	65	1010101	89	1110101	113	10010101
18	11000	42	111000	66	1011000	90	1111000	114	10011000
19	11001	43	111001	67	1011001	91	1111001	115	10011001
20	11010	44	11110	68	1011010	92	1111010	116	10011010
21	11011	45	111011	69	1011011	93	1111011	117	10011011
22	11100	46	111100	70	1011100	94	1111100	118	10011100
23	11101	47	111101	71	1011101	95	1111101	119	10011101

Let's mark $\Phi(k)$ – sequence of n bite factorial codes. In the case where number of digits of code combination from code set $\Phi(k)$ less than k , it is prefixed with required amount of zeros. Let's $\Phi_d(k)$ – sequence of quasi equidistant k – bite factorial codes with Hamming distances among related combinations equal to d . Based on data from Tab. 2 it is easy to create (Tab. 3) sequences $\Phi_1(k)$ for $k = 1$ (singular case), and also $k = 2$ and $k = 3$ created by columns rearrangement of base sequences (variant 1).

Table 3. Sequences of quasi equidistant Factorial Codes

$\Phi_1(k)$								
$k = 1$	$k = 2$		$k = 3$					
1	1	2	1	2	3	4	5	6
0	00	00	000	000	000	000	000	000
1	01	10	010	010	100	100	001	001
	11	11	011	**110**	101	**110**	011	101
	10	01	001	100	001	010	010	100
			101	101	011	011	**110**	**110**
			100	001	010	001	100	010

Table 3 illustrates one possible method of synthesis of quasi equidistant codes. Its idea is in the following. At the very first stage the base sequence of quasi equidistant codes of n number of digits is created by means of some method (for example, the method of direct search which is examined below). On the second stage a variety of all possible rearrangements of base sequence columns (check out Tab. 3, the correspondent values are of number 1) is done which results in formation of $n!$ different quasi equidistant codes. And finally on the third stage the sequences which contain restricted code combinations are excluded from $n!$ sequences. Such combinations are 110 codes from Tab. 3 highlighted with bold type. So from six three bite sequences the only two generate quasi equidistant factorial sequences. Starting from $k = 4$ apart from quasi equidistant sets it is possible to create reflected factorial codes $\Phi_{rc}(k)$. Starting from $k = 4$ apart from quasi equidistant sets it is possible to create reflected factorial codes $\Phi_{rc}(k)$. The algorithm of reflected codes creation depends on their number of digits. In particular, here is easily provable by direct verification.

Statement 3. *The set of uniform reflected factorial codes defined by recurrence relation*

$$\Phi_{rc}(k) = 0\Phi_1(k-1) \,\|\, 1\Phi_1^R(k-1) \,,$$

Let's discuss the problem of synthesis of quasi equidistant factorial codes with a number of digits $n = \overline{4,7}$. So taking the data from Tab. 3 let's construct a preliminary weights distribution of n – bite code combinations resulting in Tab. 4. The amount of codes with even and odd weights in current table for all variants n are satisfying inequality (3) and this means, that all required conditions for quasi equidistant factorial codes creation are met.

Schema (2) of uniform codes $\Phi(4)$ weights interchanges, according to Tab. 4, is

$$012O2O2O2O2O \quad (7)$$

Table 4. Weights distribution of code words $\Phi(n)$

Code weight	The bit of code combinations			
	$n = 4$	$n = 5$	$n = 6$	$n = 7$
0	1	1	1	1
1	4	5	6	7
2	5	9	14	20
3	2	7	16	30
4		2	9	25
5			2	11
6				2
n_e	6	12	24	48
n_o	6	12	24	48
In all	12	24	48	96

At that from 5 odd elements (O) of sequence (7) two elements are equal 3 and the rest – 1. Which means, that there are ten possible variants of quasi equidistant factorial code trees of number of digits $n = 4$, from whose the one, for depiction, is shown on Fig. 1.

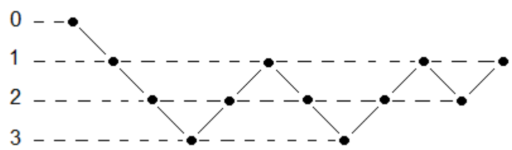

Fig. 1. Tree $\Phi_1(4)$ of sequence 012321232121

The symbolic form of the *tree* of code combination sequence $\Phi_1(5)$ can be represented by schema

$$012OEOEOEOEOEOEOEOEOEOEO\,, \quad (8)$$

One of variants is shown on Fig. 2.

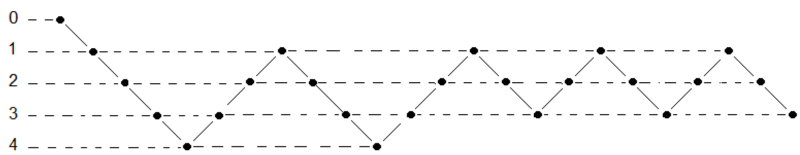

Fig. 2. The variant of tree sequence $\Phi_1(5)$

Let's go to validation to the whole amount of trees variants $\Phi_1(5)$. First of all pay attention (Fig. 2) the code combinations with weight of 4 must reside between codes with weights equal 3. This is required to provide a distance between related combinations equal to 1. Merge code pairs with weights equal to 3 among whose code with weights equal to 4 are reside. By that we can get rid of two code pairs with weights 3 and 4 in column $n = 5$ Tab. 4 and schema (8) rewrite as

$$012O2O2O2O2O2O2O2O2O\,. \tag{9}$$

There are group of nine odd (O) code combinations which contains four codes with weight equal to 1 and five with weight equal to 3 in the schema (9). It is evident the 126 variant of not complete trees of sequence $\Phi_1(5)$ exists, equal to number of nine by four combinations. And now take into consideration that in each of 126 variants of symbolic form (9) because of the operation, inversed to "merge" operation described above, it is possible to restore entire schemas of trees (8). Because of 10 possible methods of inverse operation means the entire amount of trees $\Phi_1(5)$ construction equal to 1260. Performing by the same method validation of amount of trees $L_\Phi(6)$ of $\Phi_1(6)$ sequences we get $L_\Phi(6)$ =1513512. With increasing of number of digits n the complexity of combinatorial validation $L_\Phi(n)$ and amount of trees $\Phi_1(n)$ dramatically increases. For example, all 10 variants of trees $\Phi_1(4)$ are shown in Tab. 5.

Table 5. Trees $\Phi_1(4)$

№	Tree variant		№	Tree variant
1	012323212121		6	012123212321
2	012321232121		7	012123212123
3	012321212321		8	012121232321
4	012321212123		9	012121232123
5	012123232121		10	012121212323

First of all we construct ranged by weights v sequence of uniform codes $\Phi(4)$ (Tab. 6).

Table 6. Ranged $\Phi(4)$ codes

№	Code weight v			
	0	1	2	3
1	0000	0001	0011	0111
2		0010	0101	1011
3		0100	0110	
4		1000	1001	
5			1010	

In correspondence with a schema of sixth tree variant (Tab. 5) the first two code sequences, which will be called *layers* of tree branch, choose 0000 and 0001 codes.

We could choose 0010 layer instead of 0001. The third layer to choose would be a code with weight equal to 2, the one which consist of 0001 code with Hamming distance equal to 1. Suitable ones are codes in columns with 1, 2 and 4 numbers of Tab. 6. The code with smaller number will be considered as a base, the rest – alternative. Keep moving the same way with codes choosing for $\Phi_1(4)$ sequence, using the schema of chosen tree, we have a Tab. 7.

Table 7. Synthesis of of $\Phi_1(4)$ branch

№	Code weight	Base code	Alternative code	
1	0	0000		
2	1	0001	0010	
3	2	0011	0101	1001
4	1	0010		
5	2	0110		

The ninth layer of tree under synthesis should be a code with weight equal to 2, moreover it must reside from previous code with distance equal to 1.

But there is no such a code, which were not used in Tab. 6. In order to cope with this deadlock we will do the following. We will go up through columns of and will do a substitution in this row with a nearest alternative code located from the right of it. In this case we should substitute base code 0011 with alternative code 0101 and afterwards continue the synthesis procedure for $\Phi_1(4)$. An example of quasi equidistant codes $\Phi_1(4)$ synthesized by method of direct enumeration is shown in Tab. 8.

Table 8. $\Phi_1(4)$ Sequences, correspondent to 012321212321 tree

Number of tiers	Tree	The branch of the tree									
		1	2	3	4	5	6	7	8	9	10
0	0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
1	1	0001	0001	0010	0010	0010	0010	0100	0100	0100	0100
2	2	1001	1001	0011	0011	1010	1010	0101	0101	1100	1100
3	3	1011	1101	1011	1011	1011	1011	1101	1101	1101	1101
4	2	0011	0101	1010	1010	0011	0011	1100	1100	0101	0101
5	1	0010	0100	1000	1000	0001	0001	1000	1000	0001	0001
Number of tiers	Tree	The branch of the tree									
		1	2	3	4	5	6	7	8	9	10
6	2	1010	1100	1001	1100	0101	1001	1001	1010	0011	1001
7	1	1000	1000	0001	0100	0100	1000	0001	0010	0010	1000
8	2	1100	1010	0101	0101	1100	1100	0011	0011	1010	1010
9	3	1101	1011	1101	1101	1101	1101	1011	1011	1011	1011
10	2	0101	0011	1100	1001	1001	0101	1010	1001	1001	0011
11	1	0100	0010	0100	0001	1000	0100	0010	0001	1000	0010

4 Fibonacci Sequences

Fibonacci codes are generalized concept of classical binary code [9]. Any nonnegative integer $N = 0, 1, 2, \ldots$ can be exclusively represented by a numerical Fibonacci function

$$N = \alpha_n F_n + \alpha_{n-1} F_{n-1} + \ldots + \alpha_k F_k + \ldots \alpha_2 F_2 + \alpha_1 F_1 \quad (10)$$

Besides the sequence $\{ \alpha_k \}$ in (1) doesn't contain pairs of neighbor unities which are provided by equivalent conversion called "folding' operation: $011 \to 100$. This operation makes it possible to represent Fibonacci number as so called "minimal" form, the code combination of which will have minimal weight.

For example, [10],

$$\underline{011}11\underline{011}001 \to 10\underline{011}100001 \to 10100100001 \; . \quad (11)$$

The codes which are underlined in example (11) are codes for which folding operation was performed. As it follows from this example the folding operations resulted in weights decreasing of code combinations. Namely, the amount of units in the final code is less than in the original one.

Using the folding operation it is easy to come to a representational algorithm of multidigit binary Fibonacci numbers. As an example let's consider a method of representation of natural sequence of decimal numbers (including zero) by four digits numbers of Fibonacci codes. We need to agree to label code numbers from right to left assuming the smaller (the very right) number the correspond to number 1, then number 2 and so on. We choose such a coding method of first three decimal numbers 0, 1 and 2:

$$\begin{aligned} 0_{10} &\to 0000 \;; \\ 1_{10} &\to 0001 \;; \\ 2_{10} &\to 0010. \end{aligned} \quad (12)$$

A conversion from decimal number k_{10} to $(k+1)_{10}$ number in Fibonacci codes (label them as F_k and F_{k+1} correspondingly) will be performed using a rule: if there is 0 in a smaller position F_k then it is substituted with 1 in F_{k+1} code. If there is 1 in a smaller position F_k then this 1 goes to the second position and writes as 0 in a smaller position. This rule is using in system (12) while conversion from F_1 to F_2 .

Let's represent number 3_{10} with Fibonacci code. But before we go, following the rule described above we will get code $3_{10} \to 00011$ which by folding operation would be represented in its minimal form

$$3_{10} \rightarrow 0100. \tag{13}$$

According to statements (12) and (13), the smaller positions of Fibonacci codes are using for decimal numbers 1, 2 and 3 representations correspondingly. Those values are generalized by the following recurrent block synthesis algorithm of binary Fibonacci sequences. Let's $F(k)-$ is a set of Fibonacci numbers of the same length including 0. Then we have:

Statement 4. *A set of $k-$ bite Fibonacci numbers of the same length is defined by recurrent correlation*

$$F(k) = 10 \,\|\, F(k-2)\,. \tag{14}$$

The proving of just formulated statement can be easily performed by a method of direct verification. In the right part of (14) the $F(k-2)$ set is consisted of $(k-2)-$ position numbers.

From this it is followed that if any subset of Fibonacci numbers, included in $F(k-2)$, contain digits the number of digits of whose are less than $k-2$ then those numbers are prefixed with required amount of zeros. Algorithm (14) is right for any value $k \geq 2$. Indeed, if $k = 2$ then

$$F(2) = 10 \,\|\, F(0)\,.$$

As long as $F(0)$ set is empty then $F(2)$ set contains the only Fibonacci digit 10, which corresponds to decimal digit 2_{10}.

There are Fibonacci codes for limited sequence of decimal numbers calculated using recurrent formula considering initial condition (12) in Tab. 9. Zeros, which are located to the left of bigger unit in Fibonacci coders, have been removed.

You can see values n in column F of Tab. 9, equal to number of codes which can be created by a fixed number of binary positions. For example, $F = 3$ means the four bite combinations, which contain 1 in its older position, can be created three Fibonacci codes. Writing down the values from F column we will get sequence 1, 1, 2, 3, 5, 8, 13,… which is classical sequence of Fibonacci numbers.

Now go to estimation of variants of quasi equidistant Fibonacci code trees. For this purpose based on data from Tab. 9 let's create a preliminary table of distribution of code combinations weights, included in $F(k)$, $k = \overline{4,7}$, (Tab. 10). By analysis of data from Tab. 10 we have the following conclusion. Quasi equidistant sequences of four digit Fibonacci numbers are end up with code combinations with weight of 1, five or six number of digits with weight of 2 and seven numbers of digits with odd weight equal to 1 or 3.

Table 9. Fibonacci numbers

k_{10}	F_k	F		k_{10}	F_k	F		k_{10}	F_k	F
0	0			13	100000			21	1000000	
1	1	1		14	100001			22	1000001	
2	10	1		15	100010			23	1000010	
3	100	2		16	100100			24	1000100	
4	101			17	100101			25	1000101	
5	1000			18	101000			26	1001000	
6	1001	3		19	101001	8		27	1001001	13
7	1010			20	101010			28	1001010	
8	10000							29	1010000	
9	10001							30	1010001	
10	10010	5						31	1010010	
11	10100							32	1010100	
12	10101							33	1010101	

Table 10. Distribution of code combinations weights $F(k)$

All code combinations	Number of code digits (k)			
	4	5	6	7
0	1	1	1	1
1	4	5	6	7
2	3	6	10	15
3		1	4	10
4				1
$n_{ч}$	4	7	11	17
$n_{н}$	4	6	10	17
All together	8	13	21	34

It is not that complicated to perform a calculation $L_F(k)$ of quantity of variants for quasi equidistant Fibonacci sequence $F_1(k)$ trees.

The result of this calculation for chosen *k* parameters is shown in Tab. 11.

Table 11. Power of tree subset $F_1(k)$

	Amount of tree variants $F_1(k)$			
k	4	5	6	7
$L_F(k)$	1	5	126	205920

For reflected Fibonacci codes it is right the following

Statement 5. *A set of even k bite reflected Fibonacci codes is defined by recurrent correlation*

$$\Phi_{\text{от}}(k) = 00F_1(k-2) + 10F_1^{\,R}(k-2)\,, \tag{15}$$

where $F_1^{\,R}(k)-$ sequence is inversed to $F_1(k)$, i.e. the sequence of quasi equidistant codes $F_1(k)$ written in reverse order.

As an example (Tab. 12) of calculated using a computer a branch of one tree $F_1(6)$.

Table 12. Sequences $F_1(k)$ of tree 012321232123232121212

Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
0	0	000000	000000	000000	000000	000000	000000	000000	000000
1	1	000001	000001	000010	000100	000100	001000	001000	010000
2	2	000101	010001	100010	000101	010100	001010	101000	010001
3	3	010101	010101	101010	010101	010101	101010	101010	010101
4	2	010100	000101	001010	010001	000101	101000	100010	010100
5	1	000100	000100	001000	000001	000001	100000	100000	000100
6	2	100100	100100	101000	100001	100001	100001	100001	000101
7	3	100101	100101	101001	100101	100101	101001	101001	100101
8	2	100001	100001	001001	100100	000000	001001	001001	100100
9	1	100000	100000	000001	100000	100000	000001	000001	100000
10	2	100010	100010	010001	100010	100010	010001	010001	100010
11	3	101010	101010	010101	101010	101010	010101	010101	101010
12	2	101000	101000	000101	101000	101000	000101	000101	101000
13	3	101001	101001	100101	101001	101001	100101	100101	101001
14	2	001001	001001	100001	001001	001001	100100	100100	100001
15	1	001000	001000	100000	001000	001000	000100	000100	000001
16	2	001010	001010	100100	001010	001010	010100	010100	001001
17	1	000010	000010	000100	000010	000010	010000	010000	001000
Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
18	2	010010	010010	010100	010010	010010	010010	010010	001010
19	1	010000	010000	010000	010000	010000	000010	000010	000010
20	2	010001	010100	010010	010100	010001	100010	001010	010100

5 Binomial Sequences

There are many known methods for binomial codes creation and based on them – binomial sequences [11]. We will consider two ways of even binomial codes synthesis in this unit. First of them we will call an "algorithm A. Borysenko", and the second one an "algorithm of A. Beletsky", which is called as *alternative* algorithm here in after.

The whole idea of first algorithm of uneven binary binomial codes, which correlate to algorithm of full summarized binomial arithmetic, is described in [12], page 124. Of course any uneven binary code can be converted to even code of *n* number of digits (length). For this purpose it is just enough to prefix the code combination such amount of zeros so the common number of digits became equal to n.

To construct algorithms of binomial arithmetic by Borysenko it is enough to define two parameters *k* and *n*, the first one defines the maximal amount of units in codes, the second one by value $r = n - 1$, defines the maximal length of uneven binomial number. A decimal zero in Borysenko's binomial code is written down as $l = n - k$ of zeros, the range *P* of binomial numbers is defined by formula $F_{\max} = P - 1$. Here are a number of examples of binomial numbers B_x (algorithm A. Borysenko), creation whose correspond to decimal value *x* (Tab. 13).

Table 13. Variants of binomial number sequences

$n=6,\ k=4$					$n=6,\ k=2$					$n=6,\ k=3$				
x	B_x		x	B_x	x	B_x		x	B_x	x	B_x		x	B_x
0	00		10	11010	0	0000		10	10000	0	000		10	1000
1	010		11	11011	1	00010		11	10001	1	0010		11	10010
2	0110		12	11100	2	00011		12	1001	2	00110		12	10011
3	01110		13	11101	3	00100		13	101	3	00111		13	10100
4	01111		14	1111	4	00101		14	11	4	0100		14	10101
5	100				5	0011				5	01010		15	1011
6	1010				6	01000				6	01011		16	11000
7	10110				7	01001				7	01100		17	11001
8	10111				8	0101				8	01101		18	1101
9	1100				9	011				9	0111		19	111

Let's label $B(n,k)$ – sequence of binomial numbers created by Borysenko's algorithm. From analysis of Tab. 4 we get the following conclusion.

Statement 6. *Direct and inverse binomial sequences are linked with correlation*

$$B(n,k) \equiv \overline{B}^{R}(n, n-k),$$

where $\overline{B}^R(n, n-k)$ *– sequence of binomial codes, which first of all is written in reverse order to codes in* $B(n, k)$ *and secondly each position of* $\overline{B}^R(n, n-k)$ *forms by result of inversion (i.e. substitution of 0 to 1 and vice versa) of corresponding positions* $B(n, k)$.

Let's find out a possibility of quasi equidistant codes $B_1(n, k)$ creation based on set of binomial numbers $B(n, k)$. For this purpose using the data from Tab. 13 lets create a table of code combination weights distribution (Tab. 14) included in $B(n, k)$ set. According to data from Tab. 14 and also values n_e and n_o comparison, received for many other parameters n and k, we can conclude the inequality (3) for codes $B(n, k)$ is not true and as sequence it is true

Table 14. Distribution of code combination weights $B(n, k)$

Weight of code combination	$B(6,4)$	$B(6,2)$	$B(6,3)$
0	1	1	1
1	2	4	3
2	3	10	6
3	4		10
4	5		
$n_ч$	9	11	7
$n_н$	6	4	13
All together	15	15	20

Statement 7. *Binomial codes do not form quasi equidistant sequences.*

Let's move to creation of alternative binomial codes. Introduce numeric function

$$B = \alpha_n C_n^{\alpha_n} + \alpha_{n-1} C_{n-1}^{\alpha_{n-1}} + \cdots + \alpha_k C_k^{\alpha_k} + \cdots + \alpha_1 C_1^{\alpha_1} \tag{15}$$

where

$$C_l^k = \binom{k}{l} = \frac{k \cdot (k-1) \cdots (k+1-l)}{l!},$$

- binomial coefficient which is equal to number of k and l combinations. The coefficients α_k are defined by a correlation $\alpha_k = \overline{0, \lceil k/2 \rceil}$, in which $\lceil x \rceil$ means rounding of number *x* to the nearest integer above.

Series (15) is presented in form of binary coefficients α_k for each of who's the limited number of positions equal to number of digits and required for binary value $\lceil k/2 \rceil$ representation is assigned.

Coefficient unambiguously defines the value of monomial $\alpha_k C_k^{\alpha_k}$, as it is shown in Tab. 15 (in which for example purpose the value $k = 7$ is chosen).

Table 15. An example of monomial series calculation (16)

α_7	0	1	2	3	4
$C_7^{\alpha_7}$	1	7	21	35	35
$\alpha_7 C_7^{\alpha_7}$	0	7	42	105	140

For a sequence of binomial codes created by numerical function (15), let's introduce a label $B(n,r)$ in which n parameter will be called a *power* of a function, and $r-$ *order* of function, which is equal to coefficient α_n. A fragment of binomial codes is shown in Tab. 16.

Table 16. The sequence of binomial numbers $B(4,2)$

N	α_3		α_2	α_1	N	α_4		α_3		α_2	α_1
0				0	10		1	0	1	1	1
1				1	11		1	1	0	0	1
					12		1	1	0	1	0
2			1	0	13		1	1	0	1	1
3			1	1							
					14	1	0	0	0	1	0
4		1	0	1	15	1	0	0	0	1	1
5		1	1	0	16	1	0	0	1	0	1
6		1	1	1	17	1	0	0	1	1	0
					18	1	0	0	1	1	1
7	1	0	0	1	19	1	0	1	0	0	1
N	α_3		α_2	α_1	N	α_4		α_3		α_2	α_1
8	1	0	1	0	20	1	0	1	0	1	0
9	1	0	1	1	21	1	0	1	0	1	1

In order to decide a question regarding the possibility of quasi equidistant binomial sequences creation let's create a table of a set of code combinations weights (Tab. 17).

Table 17. Distribution of weights of code combinations $B_1(n,r)$

Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10
0	1	1	1	1	1	1	1	1
Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10
1	2	2	2	2	2	2	2	2
2	3	5	5	6	6	6	6	6
3	1	2	4	9	9	12	12	12
Weight	Amount of digits of binomial sequence							
	3	4	5	6	7	8	9	10

4			2	4	7	15	15	17
5					2	12	12	23
6						4	8	29
7							2	18
8								4
Even	4	6	8	1	14	26	30	57
Odd	3	4	6	11	13	26	28	55
In all	7	10	14	22	27	52	58	112
Sign	+	–	–	+	+	+	–	–

As an example check Tab. 18, where results of quasi equidistant codes creation by a method of direct enumeration based on one of trees for $B(4,2)$ is shown.

Table 18. Results of computer code synthesis $B_1(4,2)$

Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
0	0	000000	000000	000000	000000	000000	000000	000000	000000
1	1	000001	000001	000001	000001	000001	000001	000001	000001
2	2	000101	000101	000101	000101	000101	000101	000101	000101
3	3	100101	100101	100101	100101	100101	100101	100101	100101
4	4	100111	100111	100111	100111	100111	100111	100111	100111
5	3	100011	100011	100110	100110	100110	100110	100110	100110
6	2	000011	100010	100010	100010	100010	100010	100010	100010
7	3	001011	101010	100011	100011	101010	101010	101010	101010
8	2	001010	001010	000011	000011	001010	001010	001010	001010
9	3	011010	011010	001011	001011	001011	011010	011010	011010
10	4	011011	011011	011011	101011	011011	011011	011011	011011
11	3	011001	011001	011001	101001	011001	001011	011001	011001
12	2	001001	001001	001001	001001	001001	001001	001001	001001
13	3	101001	101001	101001	011001	101001	101001	101001	101001
14	4	101011	101011	101011	011011	101011	101011	101011	101011
15	3	101010	001011	101010	011010	100011	100011	100011	001011
16	2	100010	000011	001010	001010	000011	000011	000011	000011
17	1	000010	000010	000010	000010	000010	000010	000010	000010
18	2	000110	000110	000110	000110	000110	000110	000110	000110
19	1	000010	000010	000010	000010	000010	000010	000010	000010
Number of tiers	Tree	The branch of the tree							
		1	2	3	4	5	6	7	8
20	2	000110	000110	000110	000110	000110	000110	000110	000110
21	3	000111	000111	000111	000111	000111	000111	000111	000111
22	4	010111	010111	010111	010111	010111	010111	010111	010111
23	3	010110	010110	010110	010110	010110	010110	010110	010110

A feature of alternative binomial codes is that they do not allow creating quasi equidistant codes in a full manner as it is visible from Tab. 18. In particular, for all sequences shown in Tab. 18, the latest codes (highlighted) reside from previous codes with a Hamming distance equal 3 but not 1, as it is required for sequence $B_1(4,2)$. This feature of alternative binomial codes is visible in all possible variants $B_1(n,r)$.

6 Conclusions

The main result of this research is formation of generalized conditions for quasi equidistant and reflected codes existence which are produced by even consistent binary code combinations in a mixed numeration systems. Except of Gray codes the Fibonacci, factorial and binomial codes with Hamming distance between related code combinations equal to 1, are also included in a set of such codes. The main method for synthesis of quasi equidistant codes is a method of computer direct enumeration. The results of this research can be easily generalized and applied for cases where Hamming distance is more than 1.

References

1. Shannon, C. T.: A Mathematical Theory of Communication. Bell. Syst. Tech. J., 27, 379 – 423, 623 – 656 (1948)
2. Efimenko, V. V., Karpjuk, B. V., Stukalin, Iu. A.: An Algorithm for Synthesis of Binary Quasi Equidistant Codes. Journal of Acad. Sience, USSR, AVTOMETRIJA, 5, 109–115 (1968) (In Russian)
3. Bogdanov, G. T., Zinovjev, V. A. Todorov, T. J.: On the Construction of Quasi Equidistant Codes. Journal of Problems of Information Transmission, 43(4), 13–36 (2007) (In Russian)
4. Beletsky, A. Y., Beletsky E. A.: Quasi Equidistant Codes. NAU Publishing , Kiev (2008) (In Russian)
5. Banja, E. N., Selivanov, V. L.: About the Features of the Construction of Various Number Systems. Journal of NTUU "KPI" Informatics, Management and Computer Science, 49, 68–73 (2008) (In Russian)
6. Borysenko, A. A., Cherednychenko, V. B.: Number Systems in Computing. Bulletin of the SSU, Engineering Series, 4, 162–177 (2009) (In Russian)
7. Grey, F.: Pulse Code Communication, Pat. USA, № 2632058 (1953)
8. Borysenko, A. A.: Discrete Mathematic. Textbook publishing house SSU (2007) (In Russian)
9. Stahov, A. P.: Codes of Golden Proportion. Radio Communication, Moscow (1984) (In Russian)
10. Stahov, A., P.: Fibonacci Codes, http://goldenmuseum.com/1010FibCodes_rus.html
11. Zanten, A. Ja.: Binomial System and Enumerations of Combinatorial Objects. Journal of Discrete Analysis and Operations Research, Series 1. 6, 12–18 (1999) (In Russian)
12. Borysenko, A. A.: Binomial Count. Theory and practice. Publishing house SSU (2004) (In Russian)