

**Kulyk I.,  
Berezhna O.,  
Shevchenko M.**

## **DEVELOPMENT OF DATA COMPRESSING CODING METHODS ON BASIS OF BINARY BINOMIAL NUMBERS**

*Об'єктом дослідження є методи стискаючого кодування, які застосовуються для економного представлення даних в інформаційних системах. Методи стиску можуть використовуватися на різних етапах обробки даних, при передачі повідомлень і їх зберіганні. Одними із самих проблемних місць застосування методів стиску є високі вимоги до обчислювальних ресурсів, значні апаратно-програмні витрати при їхній реалізації й невисока швидкість кодування/декодування. При цьому особливий інтерес викликають методи, над результатами стиску яких можливі обчислювальні операції без їхнього зворотного відновлення.*

*У рамках підходу, коли в основі будь-якого коду можна виявити структурну систему числення, розроблені математичні моделі стиску на основі двійкових біноміальних чисел. У структурі послідовностей визначаються відповідні їм біноміальні числа на основі систем кодоформуючих обмежень. Як результат, кожній комбінації ставиться у відповідність біноміальне число, яке являє собою її стислий образ.*

*У ході дослідження сформульовані теореми про взаємно однозначну відповідність вихідних послідовностей і двійкових біноміальних чисел, які демонструють способи реалізації відображень на основі простих аналітичних співвідношень. Наведені приклади підтверджують простоту перетворень при стискаючому кодуванні й декодуванні.*

*Отримані моделі процесів стиску й відновлення, які характеризуються невеликою кількістю простих операцій. Як наслідок, розглянуті методи характеризуються високою швидкістю при сприйнятливих коефіцієнтах стиску. При цьому обсяг апаратно-програмних витрат при практичній реалізації є невеликим. Додатковим позитивним ефектом є те, що стислі образи мають властивості чисел.*

*Результати дослідження демонструють ефективність застосування стиску на основі двійкових біноміальних чисел в інформаційних системах з метою збільшення їх продуктивності й зменшення вартості обробки даних. При цьому в порівнянні з аналогічними відомими методами спостерігаються мінімальні витрати на їхнє впровадження при досягненні високої швидкості перетворень і гарному ступеню стиску двійкових даних будь-якого виду.*

**Ключові слова:** двійкові біноміальні числа, біноміальні системи числення, стиснення двійкової інформації.

### **1. Introduction**

The use of compressive data coding aims, first of all, to increase the productivity of information systems, among which computerized control systems, data archiving systems, distributed databases, etc. can be attributed. Compressing information allows one to achieve performance in two ways [1, 2]:

- 1) reducing the time spent on the transmission of information by communication channels;
- 2) increasing in memory capacity to store information that is used in the system.

Improving the performance of the system, according to the first direction, is that over the same period of time, the amount of information that is transmitted over the communication channel increases. Thus, the channel capacity increases, but, conditionally speaking, «virtually» without changing its real characteristics. Thus, the information system is able to process more data for the same period of time.

According to the second direction, an increase in the performance of an information system is possible due to:

- increasing the amount of compressed data in the memory of the same capacity;
- reducing the amount of memory for storing compressed data with the same amount of information.

Thus, information compression is a low-cost, in economic terms, way to increase the performance of data processing in various information systems. In this regard, the development and study of methods and compression algorithms is an urgent task, which has not only scientific but also technical and economic value.

### **2. The object of research and its technological audit**

*The object of research* is lossless data compression methods designed to eliminate informational redundancy of messages and minimize their length, i. e., reduce the bitness of their representation.

Compression coding methods are characterized [1–3]:

- type of compressible data;
- models of the processes of compression and recovery based on analytical ratios or statistical dependencies;
- algorithms (device diagrams) for data compression and recovery.

The use of data compression in information systems allows to:

- 1) reduce the cost of their creation by reducing the cost of memory devices and the possibility of using low-cost low-speed communication channels;

2) increase their productivity by increasing the speed of information transfer through the existing channels and increasing the amount of stored data in the existing memory without significant costs for their modernization.

In this case, of particular interest are the methods, on the results of the compression of which it is possible to carry out computational operations without reverse restoration to the original sequences.

One of the most problematic places for the introduction of compression is the high demands on the computing resources of the system, the significant costs of its implementation, and the low coding/decoding speed. To improve the above characteristics of data compression in information systems, compression methods based on binary binomial numbers are proposed for consideration.

### 3. The aim and objectives of research

The aim of research is time minimization of compression and recovery of binary sequences with a limit on the amount of hardware and software costs in the practical implementation of compression methods based on binary binomial numbers.

The objectives of the research are formulated as follows:

1. Construction of a mathematical model of the method of compressing binary  $n$ -bit equilibrium combinations with a fixed number  $0 < k < n$  of units based on binary  $(n, k)$ -binomial numbers.
2. Construction of a mathematical model of a generalized method of compressing binary  $n$ -bit sequences with a variable number  $0 < k < n$  of units based on binary  $(n, k)$ -binomial numbers.

### 4. Research of existing solutions of the problem

A huge role, as before, in information systems is played by compression methods without loss of information, which is explained by the fact that [3, 4]:

- 1) lossless compression coding is a more universal solution due to the fact that it, as a rule, does not focus on a specific, narrowly specialized type of information, but can work at once with a multitude of data types;
- 2) a significant number of lossless compression methods deal with the binary representation of information, which further emphasizes the universal nature of such methods;
- 3) for many information systems, the use of lossless compression is uncontested due to the uncertainty of the value criterion of the data used or the lack of a psychophysiological factor of information perception. Such systems, for example, include automation systems for scientific experiments, automatic control systems, and distributed databases.

Perspective methods are lossless compression algorithms and algorithms based on structural numbers [5, 6], which are generated by structural number systems. The basic idea of compressing information transformation on the basis of structural number systems is that in the structure of any code sequence it is possible to identify the corresponding structural number. Thus, by assigning their structural numbers to the initial code sequences, one can significantly reduce information redundancy.

A special place among the structural number systems is occupied by binary binomial systems with parameters  $n$  and  $k$ , which generate binary  $(n, k)$ -binomial numbers [6, 7].

The numerical binary  $(n, k)$ -binomial function, which defines the decimal quantitative equivalent  $F_j = \text{dec } X_j$  of the binary binomial number  $X_j$ , has the form [6, 7]:

$$F_j = \text{dec } X_j = \sum_{i=1}^r x_i C_{n-i}^{k-q_i},$$

where  $X_j = x_1 x_2 \dots x_i \dots x_r$ ,  $r < n$ ,  $X_j \in X$ ,  $j = 1, 2, \dots, C_n^k$ ;  $q_i$  – the sum of single digits  $x_i$  from the first digit to  $(i-1)$ -th inclusive:

$$q_i = \sum_{t=1}^{i-1} x_t, \quad q_i \leq k.$$

The generated  $(n, k)$ -binomial numbers  $X_j$  must satisfy the following code-forming constraints [6, 8]:

$$\begin{cases} l = n - k, \\ x_r = 0, \end{cases} \quad \text{and} \quad \begin{cases} q = k, \\ x_r = 1, \end{cases}$$

where  $q$  and  $l$  – the numbers of units and zeros in the binary binomial number  $X_j$ .

In [9, 10], methods of coding information sources using a combinatorial number system are given, but the structural numbers are not used in coding, but the direct transition from a compressible sequence to a binary number is carried out. Such a transition is characterized by computational complexity. Although in [9] an attempt is made to consider binomial codes, but without a systematic approach in the absence of code-forming restrictions.

In papers [11, 12] methods of compressive coding based on numbering functions are considered, but, on the one hand, these methods have significant computational complexity, and, on the other hand, numbers are not considered as such when considering these methods. In addition, compressible sequences are characterized by complex combinatorial constraints, which limit the use of the proposed methods.

In [13], a compression model based on binary binomial numbers is presented, but only for one type of binary sequences – equilibrium combinations, which are of very limited distribution.

The theoretical and practical basis of this work is the fact that the basis of any compressible binary sequences can reveal the corresponding structural numbers generated by the structural number systems [6]. Such an approach to the development of compression coding methods is characterized by the universality of problem solving, as well as obtaining compressed images with numerical characteristics. This allows, if necessary, to carry out their computational processing without restoring them.

In this work, binary equilibrium combinations are considered as compressible sequences that can be obtained from any binary sequence by simply counting the number of binary units contained in it.

### 5. Methods of research

The implementation of the binomial representation  $\varphi^{-1}: Y \rightarrow X$  in the framework of constructing a mathematical model of the enumeration for the initial binary  $n$ -bit sequences of the form  $Y_j \in Y[n, k]$  is a compression:

$$f_b: Y[n, k] \rightarrow X[n, k], \quad (1)$$

of equilibrium combinations  $Y_j$  based on binary  $(n, k)$ -binomial numbers  $X_j \in X[n, k]$  [14]. In turn, the implementation of a binomial representation  $\varphi: X \rightarrow Y$  within the framework of the construction of a mathematical model of generation for  $Y_j \in Y[n, k]$  means the restoration of:

$$f_b^{-1}: X[n, k] \rightarrow Y[n, k], \quad (2)$$

original equilibrium combinations  $Y_j$  based on binary  $(n, k)$ -binomial numbers.

The following Theorem 1, which present without proof, gives the properties of the representation  $f_b$  and the method for its practical implementation. Let's agree on the concatenation operation to be denoted by the symbol  $\ll/\gg$ .

**Theorem 1.** Any binary sequence  $Y_j = y_1 y_2 \dots y_i \dots y_n$ ,  $Y_j \in Y[n, k]$ ,  $j = 1, C_n^k$  composed of  $n$  digits  $y_i$ , the sum of the values of which is equal  $k$ , can be assigned a single binary  $(n, k)$ -binomial number  $X_j = x_1 x_2 \dots x_i \dots x_r$ ,  $X_j \in X[n, k]$ ,  $r < n$ , using a function  $X_j = f_b(Y_j)$  of the form:

$$X_j = x_1 x_2 \dots x_i \dots x_r = \begin{cases} y_1 y_2 \dots y_i \dots y_{n-1} 0 / 00 \dots 0, \\ y_1 y_2 \dots y_i \dots y_{n-1} 1 / 11 \dots 1. \end{cases} \quad (3)$$

Thus, the representation  $f_b: Y[n, k] \rightarrow X[n, k]$ , which is defined by Theorem 1, will be called the compression method based on binary  $(n, k)$ -binomial numbers or binomial compression.

The simulation of the  $f_b$  compression process of binary equilibrium combinations  $Y_j = y_1 y_2 \dots y_i \dots y_n$  based on binary  $(n, k)$ -binomial numbers  $X_j$ , using Theorem 1 and function (3), consists of the following stages.

*Stage 1.* The value of the last digit  $y_n$  is determined in an  $n$ -bit equilibrium combination  $Y_j = y_1 y_2 \dots y_i \dots y_n$  having the  $k$  units.

*Stage 2.* If  $y_n = 0$ , then:

$$X_j = Y_j / 00 \dots 0 = y_1 y_2 \dots y_i \dots y_{n-1} 0 / 00 \dots 0 = x_1 x_2 \dots x_i \dots x_{r-1} 1,$$

that is, from the combination  $Y_j = y_1 y_2 \dots y_i \dots y_{n-1} 0$  all zero digits are discarded, starting with  $y_n = 0$ , until the first binary unit appears  $y_r = 1$ , which will represent the value of the last digit  $x_r = y_r = 1$  of the desired  $(n, k)$ -binomial number  $X_j = x_1 x_2 \dots x_i \dots x_{r-1} 1$ . Otherwise:

$$X_j = Y_j / 11 \dots 1 = y_1 y_2 \dots y_i \dots y_{n-1} 1 / 11 \dots 1 = x_1 x_2 \dots x_i \dots x_{r-1} 0,$$

that is, all single digits are discarded from the combination  $Y_j = y_1 y_2 \dots y_i \dots y_{n-1} 1$ , starting with  $y_n = 1$ , until the first binary zero appears  $y_r = 0$ , which will represent the value of the last digit of the sought  $(n, k)$ -binomial number. In both cases, the values of the remaining digits remain unchanged:  $x_1 = y_1$ ,  $x_2 = y_2$ , ...,  $x_{r-1} = y_{r-1}$ .

Theorem 2, which present without proof, gives the properties of the representation  $f_b^{-1}$  and the method of its practical implementation. Let's now introduce into consideration the operation of concatenation, which we denote as  $\ll++\gg$ . This action  $x_r = y_r = 0$  is inverse to the concatenation operation.

**Theorem 2.** To every binary  $(n, k)$ -binomial number  $X_j = x_1 x_2 \dots x_i \dots x_r$ ,  $X_j \in X[n, k]$ ,  $r < n$ , one can associate a single binary equilibrium combination  $Y_j = y_1 y_2 \dots y_i \dots y_n$ ,  $Y_j \in Y[n, k]$ ,  $j = 1, C_n^k$ , composed of  $n$  digits  $y_i$ , the sum of whose values  $k$  is equal, using a function  $Y_j = f_b^{-1}(X_j)$  of the form:

$$Y_j = y_1 y_2 \dots y_i \dots y_n = \begin{cases} x_1 x_2 \dots x_i \dots x_{r-1} 0 ++ 11 \dots 1, \\ x_1 x_2 \dots x_i \dots x_{r-1} 1 ++ 00 \dots 0. \end{cases} \quad (4)$$

Modeling the process of recovering  $f_b^{-1}$  binary equilibrium combinations  $Y_j = y_1 y_2 \dots y_i \dots y_n$  based on binary  $(n, k)$ -binomial numbers  $X_j$ , using Theorem 2 and function (4), consists of the following stages.

*Stage 1.* the value of the last digit  $x_r$  is determined in the binary  $(n, k)$ -binomial  $r$ -bit number  $X_j = x_1 x_2 \dots x_i \dots x_r$ ,  $X_j \in X[n, k]$ ,  $r < n$ .

*Stage 2.* If  $x_r = 0$ , then:

$$Y_j = X_j ++ 11 \dots 1 = x_1 x_2 \dots x_i \dots x_{r-1} 0 ++ 11 \dots 1 = y_1 y_2 \dots y_i \dots y_{n-1} 1,$$

that is, single digits 11...1:  $y_{r+1} = y_{r+2} = \dots = y_n = 1$  are added to the binary binomial number  $X_j = x_1 x_2 \dots x_i \dots x_{r-1} 0$ : so that the total number of digits of the desired binary equilibrium combination  $Y_j$  is  $n$ ,  $Y_j \in Y[n, k]$ . Otherwise:

$$Y_j = X_j ++ 00 \dots 0 = x_1 x_2 \dots x_i \dots x_{r-1} 1 ++ 00 \dots 0 = y_1 y_2 \dots y_i \dots y_{n-1} 0,$$

i. e. zero bits 00...0:  $y_{r+1} = y_{r+2} = \dots = y_n = 0$  are added to the binary binomial number  $X_j = x_1 x_2 \dots x_i \dots x_{r-1} 1$ : so that the total number of bits of the desired binary equilibrium combination  $Y_j$  is  $n$ ,  $Y_j \in Y[n, k]$ . In both cases, the values of the remaining digits remain unchanged  $x_1 = y_1$ ,  $x_2 = y_2$ , ...,  $y_r = x_r = 1$ .

The representation  $f_b: Y[n, k] \rightarrow X[n, k]$  is bijective, because the correspondences  $X_j = f_b(Y_j)$  and  $Y_j = f_b^{-1}(X_j)$  are functional (Theorems 1 and 2), that is, each element  $Y_j \in Y[n, k]$  has a single image  $X_j \in X[n, k]$ , and each element  $X_j \in X[n, k]$  has a unique prototype  $Y_j \in Y[n, k]$ .

The representation of the form  $f_b$  and  $f_b^{-1}$  operate with binary  $n$ -bit equilibrium combinations  $Y_j \in Y[n, k]$ , i. e. the number of  $k$  units is a constant value. It should be noted that, based on the properties of binary  $(n, k)$ -binomial numbers [2],  $k_{\min} = 1$  and  $k_{\max} = n - 1$ . More general is the case when  $k$  can take any values from the specified range  $0 \leq k \leq n$ , and the compressible array  $A$  is a set:

$$A = \bigcup_{k=0}^n Y[n, k] \text{ and } A_j \in A = \{0, 1\}^n, \quad j = \overline{1, 2^n},$$

of binary  $n$ -bit sequences  $A_j$ , for which there is no limitation of the form by the number of  $k$  units.

Taking into account the fact that binary  $(n, k)$ -binomial numbers  $X_j$  are prefix only for a  $k$  constant value [6], then for one-to-one recovery  $A_j \in A = \{0, 1\}^n$  from the  $X_j$  numbers one should additionally use the value of the  $k$  units expressed in binary  $\text{Bink}$ .

When compressing  $A_j$  it is necessary to use a function  $f_w$  that associates the initial sequence  $A_j$  with a sample  $(k, Y_j)$ , where  $Y_j = A_j$ . Further, if the obtained value  $k$  satisfies the inequality  $0 < k < n$ , then to compress the equilibrium combination  $Y_j$  corresponding to  $A_j$ ,  $f_b$  coding based on binary binomial numbers is used. At the same time, a compressed form  $\text{Bink}$  is added to the compressed combinations for unambiguous recovery, i. e., additional coding  $f_k$  is performed. If the value  $k$  satisfies the system of equalities  $(k=0) \vee (k=n)$ , then the resulting coded result combination will consist only of  $\text{Bink}$ , that is, the only coding method  $f_k$  is used.

Thus, let's consider the representation of the form:

$$f_{bg} : A \rightarrow Z,$$

which is set by the corresponding function:

$$Z_j = f_{bg}(A_j),$$

where  $A_j = a_1 a_2 \dots a_i \dots a_n$ ,  $A_j \in A = \{0, 1\}^n$ ,  $Z_j = (\text{Bin } k, X_j)$  or  $Z_j = \text{Bin } k$ ,  $Z_j \in Z$ ,  $j = 1, 2^n$ . The following Theorem 3, which present without proof, gives the properties of the representation  $f_{bg}$  and the method of its realization.

**Theorem 3.** Any binary sequence  $A_j = a_1 a_2 \dots a_i \dots a_n$ ,  $A_j \in A = \{0, 1\}^n$ ,  $j = 1, 2^n$  can be put in a one-to-one correspondence binary combination  $Z_j \in Z$  of the following form:

1) if, then:

$$Z_j = \text{Bin } k ++ X_j, \tag{5}$$

where  $k = \sum_{i=1}^n a_i$  and  $X_j = f_b(Y_j)$ ,  $X_j \in X[n, k]$ ,  $Y_j \in Y[n, k]$ ;

2) otherwise, if  $(k=0) \vee (k=n)$ , then:

$$Z_j = \text{Bin } k. \tag{6}$$

The representation  $f_{bg} : A \rightarrow Z$  is also bijective, since each element  $A_j$  has a single image, and each element of  $Z_j$  – a unique prototype for all  $A_j \in A$  and  $Z_j \in Z$ .

The methods of practical realization of the representations  $f_{bg}$  and  $f_{bg}^{-1}$  indicated in Theorem 3 in (5) and (6) may be different. The chosen approaches to building  $f_{bg}$  and  $f_{bg}^{-1}$  coding methods and the models of processes formed for them ultimately influence the speed and volume of hardware and software costs in their practical implementation.

Representation of  $f_{bg} : A \rightarrow Z$  is called a generalized method of compression based on binary  $(n, k)$ -binomial numbers (or generalized binomial compression), which is given by the following complex function of the form:

$$f_{bg} = \begin{cases} f_k \circ f_w, & (k=0) \vee (k=n), \\ f_k \circ f_b \circ f_w, & 0 < k < n, \end{cases} \tag{7}$$

where  $Z$  – the set of resulting sequences  $Z_j$ :

$$Z = Z_o \cup Z_b,$$

$$Z_o = Q \times \emptyset = \{Z_j / Z_j = \text{Bin } k, (k=0) \vee (k=n)\},$$

$$Z_b = Q \times X[n, k] = \{Z_j / Z_j = (\text{Bin } k, X_j), 0 < k < n-1\},$$

$$Q = \{\text{Bin } k / 0 \leq k \leq n\}, \quad Y_j \in Y[n, k], \quad Y_j = f_w(A_j);$$

$f_k$  – a function  $Z_j = f_k(A_j)$  that associates the source sequence  $Y_j = A_j$  with a binary record  $\text{Bin } k$  of the number of units  $k$ , where,  $(k=0) \vee (k=n)$  and which defines the display of the form:

$$f_k : Y[n, k] \rightarrow Z_o,$$

or a function  $Z_j = f_k(X_j)$  that associates with the binary  $(n, k)$ -binary number the resulting sequence  $Z_j = \text{Bin } k + X_j$ , if  $0 < k < n$ , and which defines a representation of the form:

$$f_k : X[n, k] \rightarrow Z_b;$$

$f_w$  – function  $Y_j = f_w(A_j)$  that associates the initial sequence  $A_j$  with an ordered sample of the form  $(k, Y_j)$ , where  $Y_j = A_j$ , and defining the representation of the form:

$$f_w : A \rightarrow M,$$

$$M = \{(k, Y_j) / 0 < k < n, Y_j \in Y[n, k]\}.$$

In turn, the inverse representation  $f_{bg}^{-1} : Z \rightarrow A$ , which is given by the inverse of a complex function:

$$f_{bg}^{-1} = \begin{cases} f_w^{-1} \circ f_k^{-1}, & (k=0) \vee (k=n), \\ f_w^{-1} \circ f_b^{-1} \circ f_k^{-1}, & 0 < k < n, \end{cases} \tag{8}$$

is a recovery based on the available values  $k$  of the original binary sequences  $A_j$ . In the case  $0 < k < n$   $A_j$  recovery is made on the basis of  $\text{Bin } k$  and binary and  $(n, k)$ -binomial numbers  $X_j \in X[n, k]$ , and in the case of  $k=0$  or  $k=n$  on the basis of  $\text{Bin } k$  by generating  $n$  zeros or units, respectively. This kind of recovery will be called generalized binomial.

The methods for implementing complex functions (7), (8) on the subdomain of the definition  $k \in \{1, 2, \dots, n-1\}$  for compression  $f_{bg}$  are similar as for functions (3), (4) of compression  $f_b$  over the entire range of values  $k$ , i. e., over the entire range  $0 < k < n$ . The methods for implementing complex functions (7), (8) on a subdomain of the definition  $k \in \{0, n\}$  are determined by a simple operation of calculating  $k$  units in compression  $f_{bg}$  and forming the initial zero or unit sequence  $A_j$  in recovery  $f_{bg}^{-1}$ .

Theorems 1 and 2, which substantiate the methods of implementing the correspondence formulated by Theorem 3, as well as Theorem 3 itself, follow the models of the processes of generalized binomial compression  $f_{bg}$  and recovery  $f_{bg}^{-1}$  of binary sequences.

The process of compressing of  $f_{bg}$  binary sequences  $A_j = a_1 a_2 \dots a_i \dots a_n$ ,  $A_j \in A = \{0, 1\}^n$ ,  $j = 1, 2^n$  is simulated on the basis of Theorems 1 and 3, function (3) and consists of the following stages.

*Stage 1.* Determine the number  $s$  of digits for the binary representation  $\text{Bin } k$  of the number of  $k$  units,  $0 \leq k \leq n$  of the initial  $n$ -bit sequence  $A_j = a_1 a_2 \dots a_i \dots a_n$ :

$$s = \lceil \log_2(n+1) \rceil.$$

*Stage 2.* Calculate the number  $k$  of binary units in the original  $n$ -bit sequence  $A_j = a_1 a_2 \dots a_i \dots a_n$ :

$$k = \sum_{i=1}^n a_i,$$

thereby realizing the function and determining the class of equilibrium combinations to which.

*Stage 3.* The conversion of the number of  $k$  units to its binary form  $\text{Bin } k$  is done, consisting of  $s$  digits.

*Stage 4.* If the number  $k$  satisfies the system of equalities  $(k=0) \vee (k=n)$ , that is  $k \in \{0, n\}$ , then the result will be a combination of the form  $Z_j = \text{Bin } k$ ,  $Z_j \in Z_o$ . Otherwise, the existing value  $n$  and the calculated value  $k$  are the parameters of the binary  $(n, k)$ -binomial number system and the transition to the next stages is carried out to implement the coding  $f_k(f_b(Y_j)) = Z_j$ .

Stage 5. Determined in an  $n$ -bit equilibrium combination  $Y_j = y_1 y_2 \dots y_i \dots y_n$  having the number of  $k$  units, the value of the last digit  $y_n$ .

Stage 6. If  $y_n = 0$ , then:

$$X_j = Y_j / 00 \dots 0 = y_1 y_2 \dots y_i \dots y_{n-1} 0 / 00 \dots 0 = x_1 x_2 \dots x_i \dots x_{r-1} 1,$$

that is, all zero digits are discarded from the combination  $Y_j = y_1 y_2 \dots y_i \dots y_{n-1} 0$ , starting with  $y_n = 0$ , until the first binary unit  $y_r = 1$  appears, which will represent the value of the last digit  $x_r = y_r = 1$  of the  $(n, k)$ -binomial number. Otherwise:

$$X_j = Y_j / 11 \dots 1 = y_1 y_2 \dots y_i \dots y_{n-1} 1 / 11 \dots 1 = x_1 x_2 \dots x_i \dots x_{r-1} 0,$$

that is, all single digits are discarded from the combination  $Y_j = y_1 y_2 \dots y_i \dots y_{n-1} 1$ , starting with  $y_n = 1$ , until the first binary zero appears  $y_r = 0$ , which will represent the value of the last digit  $x_r = y_r = 0$  of the  $(n, k)$ -binomial number  $X_j = x_1 x_2 \dots x_i \dots x_{r-1} 0$ . In both cases, the values of the remaining digits remain unchanged  $x_1 = y_1, x_2 = y_2, \dots, x_{r-1} = y_{r-1}$ .

Stage 7. The concatenation of binary values  $\text{Bin } k$  and  $(n, k)$ -binomial number  $X_j$ , that is, the coding of the form  $f_k(X_j) = Z_j$  for the case  $0 < k < n$  or  $k \in \{1, 2, \dots, n-1\}$ :

$$Z_j = \text{Bin } k + X_j,$$

thereby obtaining the resultant combination  $Z_j \in Z_b$ .

The simulation of the process of recovery of  $f_{bg}^{-1}$  sequences  $A_j = a_1 a_2 \dots a_i \dots a_n, A_j \in A = \{0, 1\}^n, j = 1, 2^n$ , from combinations of images  $Z_j, Z_j \in Z_o \cup Z_b$ , is carried out on the basis of Theorems 2, 3 and function (4). In the model that implements  $f_{bg}^{-1}$ , as the main stages, the stages from the recovery model  $f_b^{-1}$  with the known  $\text{Bin } k$  are used, and stages are introduced that form sequences only from zeros or only from units when  $k=0$  or  $k=n$ , respectively.

### 6. Research results

The results of the representation  $f_b: Y[n, k] \rightarrow X[n, k]$  and, therefore,  $f_b^{-1}: X[n, k] \rightarrow Y[n, k]$  with  $n=8$  and  $k=2$  for some combinations are given in Table 1.

Shaded cells in the Table 1 means the dropped bits of the equilibrium combinations  $Y_j$  according to (3) or the added digits to the binary binomial numbers  $X_j$  according to (4),  $1 \leq j \leq C_8^2$ . The average compression ratio of the equilibrium combinations that make up the data array in Table 1 has a value of 1.94. The average compression time  $f_b$  is defined as the average time of the number of machine cycles required to search for and discard binary digits, and is 3.2 machine cycles.

As an example, let's compress  $f_b$  of the original equilibrium combination  $Y_j \in \in Y[16, 4] - Y_j = 1000100001100000$ . In accordance with (3), we obtain a compressed image  $Y_j$  in the form  $X_j \in X[16, 4]$ :

$$X_j = 1000100001100000/00000 = 10001000011.$$

The length of the  $f_b^{-1}$  resulting combination is reduced from 16 binary digits to 11, i. e., 1.45 times.

**Table 1**

Correspondence between  $Y[8, 2]$  and  $X[8, 2]$  for some equilibrium combinations

$j$	Equilibrium code $Y[8, 2]$	Set $X[8, 2]$
0	0 0 0 0 0 0 1 1	0 0 0 0 0 0
9	0 0 0 1 1 0 0 0	0 0 0 1 1
14	0 0 1 1 0 0 0 0	0 0 1 1
15	0 1 0 0 0 0 0 1	0 1 0 0 0 0 0
18	0 1 0 0 1 0 0 0	0 1 0 0 1
19	0 1 0 1 0 0 0 0	0 1 0 1
20	0 1 1 0 0 0 0 0	0 1 1
22	1 0 0 0 0 0 1 0	1 0 0 0 0 0 0
27	1 1 0 0 0 0 0 0	1 1

Recovery of an equilibrium combination  $Y_j \in Y[20, 16]$ , having a binary binomial number  $X_j \in X[20, 16] - X_j = 010010$ , is made according to (4) as follows:

$$Y_j = 010010 + +111111111111 = 010010111111111111.$$

The results of the representations for  $f_{bg}: A \rightarrow Z$  and  $f_{bg}^{-1}: Z \rightarrow A$  for  $n=24$  where  $A_j \in A = \{0, 1\}^{24}, 1 \leq j \leq 2^{24}$  are given in Table 2. If  $0 < k < 23$ , then a complex coding function  $f_k \circ f_b \circ f_w$  is used, otherwise, i. e. with  $(k=0) \vee (k=24)$  – a complex coding function  $f_k \circ f_w$ . The ratio of the lengths of the binary representation  $A_j$  and  $Z_j$  for Table 2 ranges from 0.86 to 4.8, and their average value for the entire table under consideration is approximately 2.15. The average time  $T$  of the generalized binomial compression  $f_{bg}$  for the information array Table 2 is now defined as the sum of the time  $t_1$  required to count the number of  $k$  units, and the average time  $t_2$  of compression of the array. Taking for the simplest, but the most costly way of counting units  $t_1 = n = 24$ , let's obtain  $T = t_1 + t_2 = 24 + 6.6 = 30.6$  machine tact.

**Table 2**

The correspondence between some binary  $A_j$  and  $Z_j$  at  $n=24$

Binary sequence $A_j$	Binary combination $Z_j$		Form $f_{bg}$
	Bin $k$	$X_j$ or empty line	
000000000000000000000000	00000	-	$f_k \circ f_w$
0000000000000000000001001	00010	00000000000000000000100	$f_k \circ f_b \circ f_w$
000000001000011000000000	00011	000000001000011	$f_k \circ f_b \circ f_w$
001100001110011100100000	01001	0011000011100111001	$f_k \circ f_b \circ f_w$
011100010000000000000000	00100	01110001	$f_k \circ f_b \circ f_w$
011111011001010000101100	01100	0111110110010100001011	$f_k \circ f_b \circ f_w$
100000000000000000000000	00001	1	$f_k \circ f_b \circ f_w$
111000010001110111111111	10000	111000010001110	$f_k \circ f_b \circ f_w$
111111111111111111111110	10111	111111111111111111111	$f_k \circ f_b \circ f_w$
111111111111111111111111	11000	-	$f_k \circ f_w$

Analysis of the results of binomial compression of equilibrium combinations based on binary binomial numbers demonstrates a high coding and decoding speed with a fairly good compression ratio.

The time of binomial compression or recovery will be limited by the value  $t \leq n - k$  of if  $k \leq \lfloor n/2 \rfloor$ , or  $t \leq k$  if  $k > \lfloor n/2 \rfloor$ .

An analysis of the results of generalized binomial compression of binary sequences, which can have variable unit values, also shows a relatively high processing speed with a fairly good compression ratio.

But by the time  $t_2 = t$  of actual compression, the time  $t_1$  of calculating the number of units is added here, which can be made significantly smaller than the value  $n$  [15].

It should be noted that when  $n \rightarrow \infty$  the compression ratio for methods that use binary binomial numbers, will asymptotically tend to the value of the reciprocal of the entropy of the original source of information [9].

For both compression methods, the required amount of hardware and software costs during practical implementation is low, since compression uses simple operations of viewing bits of compressible combinations, checking their values and counting the number of binary units.

## 7. SWOT analysis of research results

*Strengths.* The strengths of binary binomial based compression methods are as follows.

1. The use of binary binomial numbers for compression allows to increase the performance of information systems. This is due to the reduction of the transmission time of compressed information and the reduction of the required amount of memory for its storage. At the same time, data compression ratios in the information system are quite high.

2. Compression methods based on binary binomial numbers have high speed with a low level of hardware and software costs. This allows the methods in question to work in real time, and the amount of the cost of their implementation is minimal.

3. The compression methods studied have a universal application and are aimed at processing common binary sequences, for which only the number of units containing them needs to be calculated.

4. Compressed images obtained by compressing coding have the properties of numbers, which gives an additional positive effect when using the considered compression methods. It is possible to perform various arithmetic operations, comparison and ordering operations on them without restoring the original data, which is a very useful quality, for example, for distributed databases.

*Weaknesses.* The weaknesses of compression methods based on binary binomial numbers are as follows.

1. In the case of a variable value of the number of units in binary sequences, it is necessary to use the service word for each compressed combination in order to further unambiguously restore it, which somewhat reduces the degree of binomial compression.

2. The compression degree will also decrease and may be less than one with an approximately equal number of binary zeros and units, as well as with their uniform arrangement in the discharge grid of compressible sequences.

*Opportunities.* The opportunities and prospects for further research on compression based on binary binomial numbers are as follows.

1. Detection and use of boundary values of the number of units, at which compression based on binary binomial numbers is appropriate, will significantly reduce the coding time and improve the degree of binomial compression.

2. Systems of code-forming constraints for binary binomial numbers provide the ability to control the occurrence of errors during compression of binary sequences, as well as during their transmission over a communication channel and storage in memory. This allows improving the noise immunity of information systems.

A good prospect of using compression based on binary binomial numbers in various information systems in order to increase their productivity, improve their technical and economic characteristics and reduce the cost of data processing is determined by the following characteristics of the developed models:

- speed;
- universality in relation to data types;
- low cost of implementation;
- numerical characteristics of compressed images.

*Threats.* A negative factor in the implementation of compression methods based on binary binomial numbers is the predominant use of binary data in the system, in which the probabilities of the appearance of zeros and units are approximately equal. This significantly reduces the effectiveness of the methods in question and may limit the increase in the performance of the information system.

## 8. Conclusions

1. A mathematical model is constructed for the method of compressing equilibrium combinations on the basis of binary binomial numbers. The basis of the obtained mathematical model consists of Theorems 1, 2 and the developed models of the processes of binomial compression and recovery for sequences with a constant number of units.

2. A mathematical model of the generalized binomial compression method of binary sequences with a variable number of units has been constructed. The basis of the obtained mathematical model consists of Theorem 3 and the developed model of the generalized binomial compression process (the model of the generalized binomial recovery process is based on the recovery model for equilibrium combinations).

The developed models of the processes of compression and recovery based on binary binomial numbers are characterized by a small number of simple operations, which ensures high speed of coding and decoding, as well as a low amount of hardware and software costs.

## References

1. Sayood Kh. Introduction to Data Compression. Morgan Kaufmann, 2017. 790 p.
2. Metody szhatia dannyh. Ustroistvo arhivatorov, szhatie izobrazheni i video / Vatolin D., Ratushniak A., Smirnov M., Jukin V. Moscow: DIALOG-MIFI, 2003. 384 p.
3. Smirnov M. A. Obzor primeneniya metodov bezusherbnogo szhatiya dannyh v SUBD. Saint Petersburg, 2004. 58 p. URL: [http://compression.ru/download/articles/db/smirnov\\_2003\\_database\\_compression\\_review.pdf](http://compression.ru/download/articles/db/smirnov_2003_database_compression_review.pdf)

4. Sayood K., Memon N. Lossless Compression Handbook. Academic Press, 2012. 488 p. doi: <http://doi.org/10.1201/9781420041163-101>
5. Borysenko O. A. Chislo i sistemi chislennya v elektronnyh cifrovih sistemah // Bulletin of Sumy State University. 2007. Issue 4. P. 71–76.
6. Borysenko O. A., Kulyk I. A. Binomial'noe kodirovanie: monograph. Sumy: Sumy State University, 2010. 206 p.
7. Borysenko A. A. Binomialnyi schet. Teoriya i praktika: monograph. Sumy: ITD Universitetskaya kniga, 2004. 170 p.
8. Kulyk I. A., Cherednichenko V. B., Kostel S. V. Algoritm generirovaniya dvoichnyh binomialnyh chisel na osnove minimalnyh sistem kodoobrazuyushih ogranichenii // Bulletin of Sumy State University. Series Technical Sciences. 2008. Issue 2. P. 45–52.
9. Schalkwijk J. An algorithm for source coding // IEEE Transactions on Information Theory. 1972. Vol. 18, Issue 3. P. 395–399. doi: <http://doi.org/10.1109/tit.1972.1054832>
10. Cover T. Enumerative source encoding // IEEE Transactions on Information Theory. 1973. Vol. 19, Issue 1. P. 73–77. doi: <http://doi.org/10.1109/tit.1973.1054929>
11. Amelkin V. A. Metody numeracionnogo kodirovaniya. Novosibirsk: Nauka, 1986. 155 p.
12. Amelkin V. A. Perechislitelnye zadachi seriinyh posledovatel'nostei. Novosibirsk: IVMiMG SO RAN, 2008. 317 p.
13. Kulyk I. A., Borysenko O. A., Onoriukpe A. Modeli szhatija i vosstanovlenija dannyh na osnove dvoichnyh binomial'nyh chisel: proceeding. Methods of information coding, defence and compression. Vinnytsya: Vinnitsa National Technical University, 2016. P. 101–105.
14. Kulyk I. A., Skordina E. M., Kostel S. V. Generirovanie kodov-sochetanii dlya resheniya informacionnyh zadach IUS // Automated management systems and automation devices. 2011. Vol. 155. P. 15–23.
15. Combinatorial Algorithms: Theory and Practice / Greenfield T., Reingold E. M., Nievergeld J., Deo N. // The Statistician. 1978. Vol. 27, Issue 2. P. 138. doi: <http://doi.org/10.2307/2987917>

---

**Kulyk Igor**, PhD, Associate Professor, Department of Electronics and Computing, Sumy State University, Ukraine, ORCID: <http://orcid.org/0000-0003-2403-8674>, e-mail: [i.kulyk@ekt.sumdu.edu.ua](mailto:i.kulyk@ekt.sumdu.edu.ua)

---

**Berezhna Olga**, PhD, Associate Professor, Department of Electronics and Computing, Sumy State University, Ukraine, ORCID: <http://orcid.org/0000-0001-7105-1276>, e-mail: [o.berezhna@ekt.sumdu.edu.ua](mailto:o.berezhna@ekt.sumdu.edu.ua)

---

**Shevchenko Marina**, Department of Electronics and Computing, Sumy State University, Ukraine, ORCID: <http://orcid.org/0000-0002-1434-5996>, e-mail: [marishev1993@gmail.com](mailto:marishev1993@gmail.com)