ISSN 0005-1179, Automation and Remote Control, 2011, Vol. 72, No. 7, pp. 1497–1526. © Pleiades Publishing, Ltd., 2011.
Original Russian Text © V.S. Vykhovanets, 2011, published in Avtomatika i Telemekhanika, 2011, No. 7, pp. 159–189.

Analytical Identification of Discrete Objects

V. S. Vykhovanets

Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia

Received April 23, 2009

Abstract—Consideration was given to the problem of identification of discrete objects which was reduced to analytical identification of the tabular functions. By the analytical identification is meant representation of the function of many variables defined on finite range spaces by a general bracket formula on the basis of arbitrary binary operations. Described was a formula generation procedure based on calculation of binary operations, order of occurrence of variables, and places of bracket arrangement. Estimates of formula complexities were presented.

DOI: 10.1134/S0005117911070174

1. INTRODUCTION

Mathematical models differing in the operators defining the relation of the characteristics of the modeled objects find wide use in the problems of control. This operator may be differential (difference), integral (summary), or functional (operational), as well as their combination [1, 2]. Since there are no strict approaches to the problem of selecting the type of a model, the intuition, knowledge and experience of the researcher come to the foreground.[1]

Determination of the model type is followed by formulating and solving the problem of *structural* identification which enables one to specify the primitive conjectures and acquire additional information about the modeled object. For the time being, there exist no formalized procedures to select the model structure [3]. The model structure is established from the observed data and the available a priori knowledge of the object. Here, the methods of enumeration on the given class of models play the major part [4].

The best studied is the functional differential model [5] defined by the equations enabling one to express the vector of object output characteristics $\mathbf{Y}$ in terms of its vector of input characteristics $\mathbf{X}$ with allowance for a vector of internal variables (state vector) $\mathbf{Q}$:

$$\begin{cases} \mathbf{Y} = \mathbf{F}(\mathbf{X}, \mathbf{Q}, t) \\ \dot{\mathbf{Q}} = \mathbf{G}(\mathbf{X}, \mathbf{Q}, t), \end{cases} \tag{1}$$

where $\mathbf{F}$ is the vector of output functions, $\mathbf{G}$ is the vector of transition functions, and $\dot{\mathbf{Q}}$ is the vector of the first derivatives with time t of the state variables.[2] In the general case, the vectors $\mathbf{F}$ and $\mathbf{G}$ are interpreted as nonlinear operators of structure defined to within the vector of state

[1] By the model type is meant the assembly of equations relating the characteristics of the modeled object prior to the introduction of the cause-effect relations (before dividing the characteristics into dependent and independent). Such equations are exemplified by those of mechanics, heat conduction, electrodynamics, gas dynamics, quantum physics, and so on.

[2] Extraction of the object characteristics and their division into the input and output ones is an informal procedure carried out proceeding from the general formulation of the control problem. Two approaches to determination of the internal state variables are seen: the internal variables are defined either on the basis of the same informal concepts or by analyzing in the course of modeling the empirical data captured by active or passive experiment.

variables $\mathbf{Q}$.[3] Depending on the structure of the operators $\mathbf{F}$ and $\mathbf{G}$, the control theory determines models of many different kinds [6] such as static, dynamic, integral and differential, stochastic and determinate, stationary and nonstationary, lumped and distributed, linear and nonlinear, one-dimensional and multidimensional, inertial and inertialess, continuous and discrete, and so on.

To coordinate the established form of the model with the modeled object, solved is the following identification problem reducible to restoration of the continuous functions of many variables from their sampled values. The source data are acquired in this case from experiment and represented most frequently in the tabular (matrix) form. The problem of function restoration proves to be especially complicated if the experimental data are polluted by noise or are represented by small samples.[4]

At that, the parametric methods of identification are constructed under the hypothesis of existence of a restorable dependency given to within a finite set of constants, and the nonparametric methods realize a certain smoothing of the experimental data, the desired function itself is assumed to be nonparametrizable, that is, it cannot be defined by a finite expansion in a set of elementary (standard) functions [9].

Therefore, for the *parametric* identification, the form of the model is assumed to be given, and in the course of observation of the object determined are the numerical model parameters for which the calculated values of the model responses are best coordinated with the experimental data. Despite the abundance of works on parametric identification, used are only a few approaches that are reducible to the least-squares method, stochastic approximation, and gradient search [10, 11].

In its turn, for the *nonparametric* identification, the model is defined to within the functional dependencies to be established. For the nonparametric identification, the theoretical studies either make use of the methods seeking the desired function as an expansion into an infinite series in the orthogonal sets of functions [12, 13] or determine the coefficients of the differential (integral, variational) equations whose solution represents the desired functional dependency [14]. To construct these equations, however, one needs the same a priori assumptions about the properties of the determined functions because in practice it is possible to obtain only the values of these functions at a finite number of the points of the definition domain [15, 16].

We notice that the physical objects are usually conceived and described in continuous time and with continuous characteristics (states). However, all methods of identification assume discrete time and discrete values of the characteristics.[5] Therefore, models with continuous and discrete variations of states are introduced and used depending on the nature of variation of the object state, those with continuous and discrete variation of time, depending on the time of such variation (Table 1).[6]

[3] Additional variables are included in Eqs. (1) to enable model adjustment. In the case at hand, it is unessential because the state variables can be regarded as containing the adjustable model parameters (in this case, the corresponding variables depend on the modeled object) and also the parameters of object perturbation by the environment (in this case, the corresponding variables depend on the state of the environment).

[4] This approach is allied to the classical data analysis where posed is the problem of determining a description of the experimental data such that it enables one to determine some characteristics of the object under consideration from the well-known values of its other characteristics [7, 8].

[5] All physical objects observed in practice are finite discrete both in time and characteristics. This follows from the fact that registration of the infinite set of the values of the object characteristics does not seem possible even in principle.

[6] In the table, the sign of plus implies the "following discrete value" and indicates to the discreteness of the variable. For example, t^+ means the following discrete time instant, and $\mathbf{Q}^+$, the following state vector varying discretely in each its element, that is, the values of elements belong to some finite set (in contrast to the elements $\mathbf{Q}$ assuming values on the continuous sets). In this case, $\mathbf{Q}^+(t^+)$ is interpreted as the following discrete state at the following discrete time instant.

Table 1. Classes of mathematical models

State	Time	
	Continuous	Discrete
Continuous	$\dot{\mathbf{Q}}(t) = \mathbf{G}(\mathbf{X}(t), \mathbf{Q}(t), t)$ (continuous)	$\mathbf{Q}(t^+) = \mathbf{G}(\mathbf{X}(t), \mathbf{Q}(t), t)$ (discrete time [17])
Discrete	$\mathbf{Q}^+(t) = \mathbf{G}(\mathbf{X}(t), \mathbf{Q}(t), t)$ (discrete state [18])	$\mathbf{Q}^+(t^+) = \mathbf{G}(\mathbf{X}(t), \mathbf{Q}(t), t)$ (discrete or automata)

The main distinction of these models lies in the fact that the time discretization and quantization in values are used to represent the same continuous (smooth) functions. These models should be referred to as discrete-continuous because their identification requires the best reproduction by an envelope of the discrete sequence of some initial continuous function [19, 20]. In this case one also needs an a priori assumption about the form of the expressed functional dependency, that is, we again come to the problem of nonparametric identification.

However, there exists a class of identification problems that do not yield to the existing approaches based on the a priori hypotheses about the form of the considered functional dependency. Such problems arise, in particular, at identification of the objects characterized by the lack of assumptions about the informal interpretation of the experimental data obtained.

It seems reasonable to introduce one more type of identification called the *analytical* one. With the analytical identification, determined is the expression of the identified function from its sampled values but without a priori assumptions about its continuous original.[7]

In distinction to the existing methods of identification of discrete objects where the form of the identified function is known in advance, at the analytical identification the relation between the current state of the object and its characteristics is expressed as a formula directly representing the experimental data with a given precision.

In terms of problem formulation, the analytical identification is allied to the methods of deriving formulas in the finite algebras [21, 22] where the functional constructions are carried out in a predefined basis of operations. Two methods—algebraic and spectral—dominate in this area. In the algebraic method, a function formula is sought in the given algebra and then subjected to identical transformations to obtain the desired formula [23]. The spectral method is based on expanding the function into a series in some set of spectral (orthogonal) functions whose formulas are known [25].[8]

However, in distinction to the methods of functional constructions, at analytical identification the operation basis and the construction of the desired formula are not known in advance, and the main aim of identification lies in determining compact representations of the function from its sampled values.

The present paper considers the problem of analytical identification of the discrete objects which was not explicitly discussed in the literature. In distinction to the similar problems of structural,

[7] The fundamental difference of the analytical identification from the nonparametric identification lies in the fact that at the nonparametric identification sought are the continuous functions that are assumed to be smooth, that is, it is possible to predict with given accuracy their values in the neighborhood of some point of the definition domain. At the same time, the behavior of the discrete functions describing an object at the analytical identification does not have such feature. It is also impossible to use such fundamental notion as the passage to the limit.

[8] It deserves to pay attention to the combinatorial complexity of the search of the minimal formulas in the algebraic method [24] and overwhelming computational difficulties of the spectral method that are due to the inversion of matrices of high dimensions [25]. To determine the inverse matrices and accelerate the discrete transform in the latter case, the spectral basis is selected so that the matrices of the direct and inverse transform admit factorization [26].

nonparametric, and parametric identification, as well as functional constructions and analysis of data, the problem of analytical identification is formulated differently. At analytical identification, to obtain efficient mathematical models of discrete objects, construction of the formula is determined in the course of identification, the variables and functions take on values on arbitrary finite sets, and the best operations relating the values of variables with the values of the identified function are calculated.

2. FORMAL PROBLEM DEFINITION

We assume that notions such as set, membership in set, subset, ordered set, Cartesian product set, relation, function and essential dependency of function on variable need no explanation.[9]

The tabular representations of the object functions will be used as the semantic theory. Then, the syntactic theory is represented by the set of formulas obtained to express these functions on the basis of the rules for their generation and calculation that are defined in what follows. Therefore, the developed theory is of totally formal nature.

2.1. Discrete Functions

We introduce notation for the family of sets $\mathbf{N}_k$ such that for $k = 0$ we have[10] the set of natural numbers $\mathbf{N}_0 = [0, 1, 2, \ldots]$ and for $k \neq 0$, $\mathbf{N}_k = [0, 1, \ldots, k-1]$. We assume that on the sets $\mathbf{N}_k$ given are the nonstrict order relation $\leq$ and the equality relation $=$ making these sets linearly ordered.[11]

Definition 1. By the function dimension is meant the number of variables on which the function depends, possibly unessentially.

Definition 2. By the significance of a function (variable) is meant the number of set elements on which this function (variable) possesses values.

Let the discrete function f of dimension n possess values on the set $\mathbf{N}_{k_f}$ and depend on the variables $\mathbf{X} = [x_0, x_1, \ldots, x_{n-1}]$ possessing values on the sets, $\mathbf{N}_{k_0}, \mathbf{N}_{k_1}, \ldots, \mathbf{N}_{k_{n-1}}$, where k_f is the significance of the function f and $k_0, k_1, \ldots, k_{n-1}$ are, respectively, the meanings of the variables $x_0, x_1, \ldots, x_{n-1}$. The set $\mathbf{N}_{k_f}$ is the range space of this function, and the set $\mathbf{\Lambda} = \mathbf{N}_0 \times \mathbf{N}_1 \times \ldots \times \mathbf{N}_{n-1}$, where $\times$ is the symbol of the Cartesian set product, is the domain of definition. Table 2 [28, p. 383] where the corresponding values of the function are given for all or part of values of the variables represents a form of definition of the discrete function. In this case, the maximal number of rows in the table is equal to the product of the significances of its variables.

Table 2. Tabular form of a discrete function

x_0	x_1	$\ldots$	x_{n-1}	f
0	0	$\ldots$	0	f_0
1	0	$\ldots$	0	f_1
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
$k_0 - 1$	0	$\ldots$	0	f_{k_0-1}
0	1	$\ldots$	0	f_{k_0}
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
$k_0 - 1$	$k_1 - 1$	$\ldots$	$k_n - 1$	f_{m-1}

[9] Definitions of the aforementioned basic notions can be found, for example, in [27, 28].

[10] Here and below the sign of equality $=$ is used as the equivalence relation reflecting identity (indistinguishability, sameness) of the compared elements. The sign $\equiv$ is used in turn to express an equivalence relation only in a given sense. Stated differently, two equal elements are always equivalent, but there may be unequal equivalent elements.

[11] In distinction to the unordered set denoted using the parentheses, the ordered set (sequence, vector, matrix) is expressed by a bracketed list of the symbols of elements. The symbols of elements are separated in the set by comma which may be omitted if this does not give rise to an ambiguous interpretation.

Definition 3. By the representation of the function f of the significance k_f and dimension n is meant the ordered set $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ where $\mathbf{F}$ is the vector of values of a function of length m, $\mathbf{K}$ is the vector of significances of the variables of length n, and $\mathbf{X}$ is the vector of notations (names) of the variables[12] of length n:

$$\begin{gathered}\mathbf{F} = [f_q \mid f_q \in \mathbf{N}_{k_f};\ q = \overline{0,\ m-1}];\\ \mathbf{K} = [k_i \mid k_i \in \mathbf{N}_0;\ i = \overline{0,\ n-1}];\\ \mathbf{X} = [x_i \mid i = \overline{0,\ n-1}],^{13}\end{gathered} \tag{2}$$

for which the following condition is satisfied:

$$m = \prod_{i=0}^{n-1} k_i.^{14} \tag{3}$$

Condition (3) asserts that the number of elements in the function vector $\mathbf{F}$ is equal to the number of elements in the set $\mathbf{\Lambda}$ consisting of all possible vectors $\mathbf{X}$ of values of variables (rows of Table 2).

We establish the following one-to-one relation between the set $\mathbf{\Lambda}$ consisting of m vectors of values of the variables $\boldsymbol{\lambda}_q = [\lambda_{iq} |\ i = \overline{0,\ n-1}]$, $q \in \mathbf{N}_m$ and the vector $\mathbf{F}$:

$$\mathbf{F}[x_0\ x_1\ \ldots\ x_{n-1}] = f_q, \quad q = \sum_{i=0}^{n-1} x_i \prod_{j=0}^{i-1} k_j, \tag{4}$$

where the symbols of variables x_i are replaced by their values λ_{iq} from the corresponding vector $\boldsymbol{\lambda}_q$ (qth row of Table 2).[15]

For inverse transformation of the index of element $q \in \mathbf{N}_m$ in the vector of values of the variables $\boldsymbol{\lambda}_q = [\lambda_{iq} \mid i = \overline{0,\ n-1}]$, we use the following recurrent rule:

$$\left.\begin{aligned}\lambda_{iq} &= q_i \bmod k_i\\ q_{i+1} &= q_i \operatorname{div} k_i;\end{aligned}\right\} \quad i = \overline{0,\ n-1} \tag{5}$$

with the initial conditions $q_0 = q$, where the binary operation mod is the residue of integer division of the first operand by the second operand, and div is the integer part of this division.[16]

Example 1. We define the representation of a discrete function whose vector of values is expressed by the 24 decimal digits of the number $\pi \approx 3.14159265358979323846265$ taken modulo three (residues of the integer division of the digits by 3) and determine the four variables so that

[12] Use of the vector of names of variables is due to the fact that at modeling of discrete objects a semantic interpretation in the form of name is assigned to each variable. Additionally, since the discrete object is described by more than one function, at modeling one also has to assign individual names to the variables, which allows one to identify common variables of two or more functions.

[13] The sign | employed within the brackets separates the set elements from the formalized description of the procedure intended to list or recognize them. In its turn, the bar over a list of two elements denotes an ordered set beginning with the first element of the list and ending with the last one.

[14] Here and below we make use of the arithmetics of natural numbers with operations of addition + and multiplication $\cdot$ and also with their bounded quantors of sum Σ and product Π.

[15] The considered one-to-one correspondence is equivalent to representation of the index of the element of the vector $\mathbf{F}$ in the positional calculus with the bases defined by the vector of significances of the variables $\mathbf{K}$ which is expressed as a matter of fact by (4).

[16] The proof of single-valuedness of (4) and its invertibility by the recurrent rule (5) is omitted because of its triviality.

Table 3. Example of incompletely defined discrete function

x_0	x_1	x_2	x_3	f
0	0	0	0	0
1	0	0	0	1
0	0	1	1	0
0	1	1	1	0
1	1	0	2	1
1	0	1	2	2
1	1	1	2	2

the product of their significances be equal to the length of the vector of the function.[17] As the result, we have the desired representation [**F K X**]:

$$\mathbf{F} = [0\,1\,1\,1\,2\,0\,2\,0\,2\,0\,2\,2\,0\,1\,0\,0\,2\,0\,2\,1\,0\,2\,0\,2];$$
$$\mathbf{K} = [2\,2\,2\,3];$$
$$\mathbf{X} = [x_0\,x_1\,x_2\,x_3]. \quad \blacklozenge$$

Definition 4. By the incompletely defined discrete function is meant the discrete function such that its values are given not for all vectors of values of their variables.

Example 2. Let us define an incomplete discrete function as Table 3. In this case, the table has smaller number of rows as compared with the table of completely defined function.

Analysis of Table 3 suggests that the function has significance 3 and its variables, the significances 2, 2, 2, and 3, respectively. We make use of formulas (4) and put down the representation [**F K X**] of this function:

$$\mathbf{F} = [01 * * * * * * * * * * 0 * 0 * * * * 1 * 2 * 2];$$
$$\mathbf{K} = [2\ 2\ 2\ 3];$$
$$\mathbf{X} = [x_0\ x_1\ x_2\ x_3],$$

where the undefined values are denoted by asterisk $*$ meaning any of the possible values from the set $\mathbf{N}_3$. ♦

Definition 5. Two representations are equivalent if they define the same discrete,—possibly, incomplete—function.

Example 3. Let us establish the representation [**F′ K′ X′**] which is equivalent to the representation [**F K X**] of the function form Example 2. For that, we change the order of the variables in Table 3 by permuting the columns of the variables x_2 and x_3. As the result, we obtain the representation [**F′ K′ X′**]:

$$\mathbf{F}' = [0\,1 * * * * * * * * * 1 * * * * 0 * 0 * * 2 * 2];$$
$$\mathbf{K}' = [2\ 2\ 3\ 2];$$
$$\mathbf{X}' = [x_0\ x_1\ x_3\ x_2]$$

defining the same discrete function because for the same values of variables the values of function as established from the representations [**F K X**] and [**F′ K′ X′**] are equal. ♦

[17] It deserves noting that on one vector **F** of length m it is possible to define a set of functions differing in the order and significance of the variables, that is, for each representation of the number m as a product of n integers greater than one, one can define $n!$ functions where ! is the sign of factorial function. Stated differently, the number of discrete functions is much greater than the number of their vectors.

2.2. Discrete Operations

Definition 6. By the discrete operation is meant a discrete function depending essentially on its variables.

Depending on their dimensions, we discriminate the unary (one-place), binary (two-place) and, in the general case, r-ary (r-place) operations. If a discrete function of dimension n depends essentially only on r of its variables, then this function should be regarded as an r-ary operation.

Since any operation is a function, to define operations the same forms can be used as to define the discrete functions. However, for visualization the unary operations will be defined by vectors and the binary ones, by matrices.

Definition 7. The result of application of a unary (binary) operation defined by the vector $\neg$ (matrix $\circ$)

$$\neg = [c_i \mid i = \overline{0, k_0 - 1}] \quad (\circ = [c_{ij} \mid i = \overline{0, k_0 - 1},\ j = \overline{0, k_1 - 1}])$$

to the variable a (variables a and b) is represented by the element $c_a (c_{ab})$, that is, $\neg a = c_a$ $(a \circ b = c_{ab})$, where k_0 is the significance of the variable of the unary operation (k_0 and k_1 are, respectively, the significances of the first and second variables binary operations).[18]

Example 4. We define the unary and binary operations, respectively, by the vector $\neg$ and matrix $\circ$:

$$\neg = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}, \quad \circ = \begin{bmatrix} 0 & 1 \\ 2 & 0 \\ 1 & 1 \end{bmatrix}.$$

Then, the values of the unary operation $\neg$ can be obtained for different values of its unique variable: $\neg 0 = 1$, $\neg 1 = 0$, $\neg 2 = 2$. Similarly, for the binary operation $\circ$, we have : $0 \circ 0 = 0$, $0 \circ 1 = 1$, $1 \circ 0 = 2$, $1 \circ 1 = 0$, $2 \circ 0 = 1$, $2 \circ 1 = 1$. ♦

One can readily see that with the knowledge of the values of operation for all vectors of values of variables one can construct its representation or table. The inverse transition from a representation of table to the vector for unary operations or matrix for binary ones is possible as well.

2.3. Composition and Decomposition

At decomposition, the function f is represented as a composition of the functions g and $h_0, h_1, \ldots,$ h_{u-1} having as a rule smaller dimension than f:

$$f(X) = g(h_0(X_0),\ h_1(X_1),\ \ldots,\ h_{u-1}(X_{u-1})), \tag{6}$$

where the functions h_i depend on some subsets X_i of the set of variables X of the function f $(i = \overline{0,\ u-1})$, g, on u variables for which the functions h_i are substituted. The functions h_i themselves also can be expressed in terms of formulas like (6) which in this case are called the subformulas.[19]

Definition 8. By the formula of a discrete function is meant expression (6) relating the value of the function with the values of its variables through a composition of some set of other (auxiliary) functions (operations).

[18] Sometimes the variable (argument) of operation is called the operand.

[19] The traditional notation $f(x_0, x_1, \ldots, x_{n-1})$ or its equivalent notations $f[x_0, x_1, \ldots, x_{n-1}]$ and $f([x_0, x_1, \ldots, x_{n-1}])$ will be used to denote the function f and the vector of its variables $[x_0, x_1, \ldots, x_{n-1}]$.

To determine composition (6), used are various formal systems defined as a rule by analytical constructions of the formulas generated by them such as the disjunctive, conjunctive, literal, interval, Zhegalkin, Reed–Muller, number-theoretic, polynomial, Rademacher, Walsh, Haar, Vilenkin–Chrestenson, and other systems [29]. The main aim of the analytical construction is to order construction of formulas and make it regular and efficient. Each analytical construction is in essence one of the plausible schemes of decomposition within whose framework a formula expression of the function is sought.

In practice, it is desired to construct the decomposition of a function from the variables and operations of small-dimensionality and significance belonging to some their set or base, at that the formula with the least number of operations is regarded as the best one.[20] Proceeding from this, we give some additional definitions that will be required in what follows.

Definition 9. Two formulas are equivalent if they express the same function.[21]

Definition 10. By the formula length is meant the number of operations (auxiliary functions) involved in it.

Definition 11. By the significance of a formula is meant the number k equal to the maximal significance of the operations (auxiliary functions) involved in it.

2.4. Analytical Design

The table of a discrete function that was obtained by observing the object parameters will be used as the source data for its analytical identification.[22] If one has to identify an object with a given precision, then the domains of parameter definitions are decomposed into a finite number of disjoint subdomains, and the values of the parameters are replaced by the numbers of the subdomains to which they belong.

The problem of analytical identification of the discrete function is reduced to the analytical design of formulas which lies in seeking an expression of the discrete function in terms of the unary and binary operations. It is assumed that the function may be defined incompletely, and operations may be any.

In view of the fact that the main aim of identification lies in hardware or software realization of the model obtained on a discrete computing facility, we confine the range of significances of the formula operations by some number k which is the significance of the designed formulas (Definition 11).

Efficiency of formulas is estimated in terms of their lengths. At that, of two formulas of the same function, all other things being equal, more efficient is that of smaller length, that is, smaller number of operations.

[20] The discrete computing (modeling) devices usually realize the hardwired unary and binary operations and the operations over higher number of places, are realized by software. In particular, the C-like languages use the ternary operation of arithmetic branching (conditional calculation) like $x?y : z$, and the access to the elements of the multidimensional array $F[x, y, \ldots, z]$ may be regarded as an operation with the number of places equal to the number of measurements of this array.

[21] A formula expresses a function if the function values for all values of the variable are equal to the result of calculating the formula for the same values of variable. We notice that the statements like "formula expresses function" and "formula of function" are semantically equivalent.

[22] The need for using functional representations to model the discrete objects is due among other things to the deterministic and irreversible nature of the modern modeling devices. It is assumed at using them that all calculated maps have a single resulting value, that is, are functions. The knowledge of the result does not enable one to restore uniquely the values of variables that gave rise to this result, that is, they are irreversible functions. Moreover, all hardwired functions are operations.

3. ANALYTICAL CONSTRUCTIONS OF FORMULAS

The diversity of analytical constructions is due to the attempts to design efficient formulas for constrained sets of functions. At that, a set of functions having more efficient representation as compared with the formulas of another analytical construction can be established for each analytical construction.

We are going to generalize the existing constructions of formulas and determine a construction having the best expressive potentialities, that is, the greatest set of effectively representable functions. For the found construction, we present a procedure of formula design and prove its convergence.

3.1. Analytical Construction of General Form

Let us consider an extremely general analytical construction describing the set of formulas with unary and binary operations under an arbitrary arrangement of brackets and present it in terms of the formal grammar[23] [T, N, P, Φ] defined over some terminal alphabet T adequate for putting down all formulas and having a nonterminal alphabet N = {Δ, X, Ξ, Θ, Φ}, as well as the inference or production rules P like

$$\Phi \to \Delta \mid \mathrm{X} \mid \Xi\,\Phi \mid (\Phi\,\Theta\,\Phi), \tag{7}$$

where Φ is the grammar axiom (always nonterminal symbol); → is the sign of inference (text replacement); Δ is the terminal entry of a constant expressed by the arbitrary element $\mathbf{N}_k$; X is the terminal entry of the variable expressed by an arbitrary element **X**; Ξ (Θ) is the terminal entry of a unary (binary) operation, and the vertical bar divides the right hand sides of productions (after the sign of inference) having the same left side (before the sign of inference).[24] Let us consider an example demonstrating the main notions of the theory of formal grammars.

Example 5. Let us derive a formula described by productions (7) by successive application of the grammar productions beginning from the axiom Φ:

$$\Phi \xrightarrow{4} (\Phi\,\Theta\,\Phi) \xrightarrow{3} (\Xi\,\Phi\,\Theta\,\Phi) \xrightarrow{4} (\Xi\,\Phi\,\Theta(\Phi\,\Theta\,\Phi)) \xrightarrow{3} (\Xi\,\Phi\,\Theta(\Xi\,\Phi\,\Theta\,\Xi\,\Phi)) \xrightarrow{2} (\Xi\,\mathrm{X}\,\Theta(\Xi\,\mathrm{X}\,\Theta\,\Xi\,\mathrm{X})),$$

where the number of production from (7) is indicated over the sign of inference. The inference is completed when the inferred row has no entries of nonterminal symbols. After the replacement of the entries of nonterminal symbols Ξ, X, and Θ by the terminal symbols (terms) expressing them, we obtain the final formula

$$(\neg_0 x_0 \circ_0 (\neg_1 x_1 \circ_1 \neg_2 x_2)), \tag{8}$$

where $\neg_i$ $(i = \overline{0,\,2})$ are the symbols of the unary operations, $\circ_j$ $(j = \overline{0,\,1})$ are the symbols of the binary operations, x_t $(t = \overline{0,\,2})$ are the variables from **X**. ♦

It is assumed at interpretation of the inferred formulas that the unary operations have higher priority over the binary ones. To specify the inferred formulas, one needs to determine the operations involved in them.

[23] We describe the formula construction using the syntactic facilities offered by the well developed theory of formal languages and grammars (see, for example, [30]), which enables us to use the semantic results of this theory for informal interpretation of the properties of the designed formulas, that is, the theory of formal languages and grammars is used as a metatheory of the theory of analytical identification under consideration.

[24] One of the informal propositions of the metatheory which can be made from the form of productions (7) states that since all formulas are generated by a context-free grammar, the considered set of formulas is resolvable, that is, for an arbitrary sequence of brackets, constants, variables, and operations, one can establish its membership in the set of formulas with the use of constructive means. One has to use the pushdown automaton as such facility because the finite automaton fails to resolve this problem [31].

Example 6. The operations for formula (8) are defined as the following vectors and matrices:

$$\neg_0 = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}, \quad \neg_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad \neg_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$
$$\circ_0 = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}, \quad \circ_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}, \tag{9}$$

where in the order of listing the significances of operations are 3, 2, 2, 3, and 3. Substitution of the vectors and matrices from (9) to (8) provides the following formula of length 5 and significance 3:

$$\begin{bmatrix} \underline{1} \\ 2 \\ 0 \end{bmatrix} x_0 \begin{bmatrix} 2 & 0 & 0 \\ 0 & \underline{2} & 1 \\ 1 & 1 & 2 \end{bmatrix} \left(\begin{bmatrix} 1 \\ 0 \\ \underline{1} \end{bmatrix} x_1 \begin{bmatrix} 0 & 1 \\ \underline{1} & 0 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ \underline{0} \end{bmatrix} x_2 \right). \tag{10}$$

Analysis of formula (10) shows that the variables have significances $k_0 = 3$, $k_1 = 3$, and $k_2 = 2$. The next example explains the underlined elements of the matrices and vectors in (10). ♦

Example 7. We demonstrate determination of the values of the function expressed by the formula from the last example. Let $\boldsymbol{\lambda} = [0\,2\,1]$ be the vector of values of the variables $\mathbf{X} = [x_0 x_1 x_2]$, that is, $x_0 = 0$, $x_1 = 2$, $x_2 = 1$. Then, the element $\neg_0 0 \circ_0 (\neg_1 2 \circ_1 \neg_2 1)$ is the value of the considered function. The unary operations from (9) allow us to determine the results of these operations $\neg_0 0 = 1$, $\neg_1 2 = 1$ and $\neg_2 1 = 0$ which are underlined in the calculated formula (10). The results of binary operations $1 \circ_1 0 = 1$ and $1 \circ_0 1 = 2$ (also underlined) are calculated in a similar manner. As the result, we find that element 2 is the desired value of the function. ♦

3.2. Canonical Analytical Construction

Definition 12. By the canonical is meant the formula having the analytical construction[25]

$$\mathrm{K} \to \Delta \mid \Xi\, \mathrm{X} \mid \Psi; \quad \Psi \to \mathrm{X} \mid (\Psi\, \Theta\, \Psi), \tag{11}$$

where Δ is the entry of constant, X is the entry of variable, $\Xi(\Theta)$ is the entry of unary (binary) operation, and Ψ is the canonical subformula.[26]

Theorem 1. *For any formula of general construction* (7) *and length l, there exists an equivalent formula with canonical analytical construction* (11) *and length at most l.*

Theorem 1 is proved in the Appendix. The following definition and theorem will be required in what follows.

Definition 13. Analytical constructions are equivalent if the sets of formulas generated by them are equal.

[25] It is assumed as before that the analytical construction is defined by the formal grammar [T, N, P, K]. In what follows, we omit the unessential details and identify the set of grammar's productions (or its axiom) with the analytical construction of the formula because the properties of the inferred formulas follow directly from the productions.

[26] Apart from the axiom, the set of nonterminal symbols of the grammar can comprise also other nonterminal symbols. The parts of formula inferred from these nonterminal symbols are called the subformulas.

Theorem 2. *The canonical analytical construction* (11) *is equivalent to the analytical construction* M*:*

$$\mathrm{M} \to \Delta \mid \Xi\,\mathrm{X} \mid \Psi; \quad \Psi \to \mathrm{X} \mid (\mathrm{X}\,\Theta\,\Psi) \mid ((\mathrm{X}\,\Theta\,\Psi)\,\Theta\,\Psi).^{27} \tag{12}$$

Theorem 2 is proved in the Appendix.

Theorems 1 and 2 were proved for the first time. It immediately follows from them that the canonical analytical construction is not equivalent to the general analytical construction (7) either as (11) or as (12). This fact is evident from the proof of Theorem 1 where any formula of the canonical analytical construction in form (11) is derived as a formula of the general analytical construction. The inverse, however, is not true because the redundant entries of the unary operations that are present in the analytical construction Φ are eliminated from the formulas of the canonical constructions K and M. At that, the sets of functions expressed by all these constructions are equal.

3.3. Canonization of Formulas

To reduce the general formula (7) to the canonical representation (12), we apply the procedure used to prove Theorem 1.

Example 8. We canonize the formula from Example 6. It is found from (10) that application of the unary operation $\neg_1$ to the variable x_1 is equivalent to a transformation of the binary operation $\circ_1$ such that row 1 is taken instead of the zero row of its matrix, row 0 instead of the first row, and row 1 is copied instead of row 2. As the result of such transformations, we obtain

$$\begin{bmatrix}1\\2\\0\end{bmatrix} x_0 \begin{bmatrix}2&0&0\\0&2&1\\1&1&2\end{bmatrix} \left(x_1 \begin{bmatrix}1&0\\0&1\\1&0\end{bmatrix} \begin{bmatrix}1\\0\end{bmatrix} x_2 \right).$$

In its turn, execution of the unary operation $\neg_2$ is equivalent to the permutation[28] [1 0] of the columns of the matrix of the operation $\circ_1$,

$$\begin{bmatrix}1\\2\\0\end{bmatrix} x_0 \begin{bmatrix}2&0&0\\0&2&1\\1&1&2\end{bmatrix} \left(x_1 \begin{bmatrix}0&1\\1&0\\0&1\end{bmatrix} x_2 \right).$$

Now, we analyze the binary operation $\circ_0$ and find that it has significance 2. Consequently, only the two of its first columns are used in the calculations of the next binary operation (to the left of the brackets). Therefore, the last column may be eliminated.

$$\begin{bmatrix}1\\2\\0\end{bmatrix} x_0 \begin{bmatrix}2&0\\0&2\\1&1\end{bmatrix} \left(x_1 \begin{bmatrix}0&1\\1&0\\0&1\end{bmatrix} x_2 \right).$$

[27] A stronger assertion about the construction M is valid. It is possible to demonstrate that the construction K is equivalent to the construction M such that in it the entry of the variable X in (XΘΨ) implies that there is no entry of this variable in Ψ. The last fact is possible owing to the arbitrary choice of the operations and places of brackets in the constructions. Such construction M is expressed, however, by a context-dependent grammar which is complicated (has context productions) and awkward (has a great quantity of them).

[28] Here and below the permutation of n elements is denoted by a vector of length n all of whose elements are distinct and belong to $\mathbf{N}_n$.

In its turn, the unary operation $\neg_0$ in the resulting formula gives rise to the permutation [1 2 0] of the row of the first binary operation:

$$x_0 \begin{bmatrix} 0 & 2 \\ 1 & 1 \\ 2 & 0 \end{bmatrix} \left(x_1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} x_2 \right).$$

As the result, we obtain the canonical formula $x_0 \bullet_0 (x_1 \bullet_1 x_2)$ where

$$\bullet_0 = \begin{bmatrix} 0 & 2 \\ 1 & 1 \\ 2 & 0 \end{bmatrix}, \quad \bullet_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The last operation executed in the formula is $\bullet_0$. Consequently, there are precisely two other canonical formulas defined by the number of permutations of the elements of the range space of the operation $\bullet_1$. Having carried out the permutation [1 0] of the elements of range space of the operation $\bullet_1$ and the same permutation of the columns of the operation $\bullet_0$, we obtain the operations of the second canonical formula $x_0 \div_0 (x_1 \div_1 x_2)$:

$$\div_0 = \begin{bmatrix} 2 & 0 \\ 1 & 1 \\ 0 & 2 \end{bmatrix}, \quad \div_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Three permutations of the subformulas with the binary operations are possible for each formula obtained as the result of permutations of the elements of the operation definition domains. In particular, for the formula with operations $\div_0$ and $\div_1$ we have

$$x_0 \div_0 (x_1 \div_1 x_2), \quad (x_1 \div_1 x_2) \div_0 x_0, \quad x_0 \div_0 (x_2 \div'_1 x_1), \quad (x_2 \div'_1 x_1) \div'_0 x_0,$$

where the formula of the matrix of corresponding operations is transposed to retain equivalence

$$\div'_0 = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 1 & 2 \end{bmatrix}, \quad \div'_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

We make sure that the original and canonical formulas are equivalent.[29] ♦

4. ANALYTICAL IDENTIFICATION

According to the aforementioned formulation of the problem of identification, the representation of the discrete function [**F K X**] is used as the source data for construction of the formulas expressing this function. If the view of the formula is of importance, we make use of the representation [**F K X**] to denote the entire set of function's formulas that can be obtained from this representation.

[29] It deserves noting that it is possible to minimize the number of different operations appearing in the formula. As was shown in Example 8, for that purpose it is required to execute the permutations of the elements of the range spaces of the formula operations and permutations of their variables such that one operation be transformed to the extent possible into another.

4.1. Identification Procedure

Prior to describing the identification procedure, we define on the set of vectors a metric generated by some relation of equivalence of the elements $\equiv$.[30]

Definition 14. Let given be two vectors $\mathbf{F} = [f_i \mid i = \overline{0,\ m-1}]$ and $\mathbf{G} = [g_i \mid i = \overline{0,\ m-1}]$ of the same lengths. The value of the function $d(\mathbf{F},\ \mathbf{G}) \in \mathbf{N}_m$ which is equal to the number of elements $\mathbf{F}$ not equivalent to the corresponding elements $\mathbf{G}$[31] is called the distance between $\mathbf{F}$ and $\mathbf{G}$ relative to the equivalence relation $\equiv$ of the elements:

$$d(\mathbf{F},\ \mathbf{G}) = \sum_{i=0}^{m-1} \delta(f_i,\ g_i), \quad \delta(a,\ b) = \begin{cases} 0 & \text{if} \quad a \equiv b \\ 1 & \text{if} \quad a \not\equiv b. \end{cases}$$

We describe a procedure of analytical identification constructing the function formula according to the productions of the canonical analytical construction M (12). We assume that a certain number k bounding the significance of the designed formula is given.

Procedure 1. [32] Identification of the discrete function f defined by the representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$:

(1) if $\mathbf{F} = [c\ c\ \ldots\ c]$, that is, the function vector consists of the constant c, then the procedure is completed with formula c (the first production of the analytical construction M);

(2) if $\mathbf{X} = [x_i]$, the procedure is completed with the formula x_i for $\mathbf{F} = [0\ 1\ \ldots\ k_i - 1]$ (third and fourth productions) or with formula $\neg\, x_i$ for $\mathbf{F} \neq [0\ 1\ \ldots\ k_i - 1]$ (second production), where k_i is the significance of the variable x_i, $\neg$ is the desired unary operation, $\neg = [f_q \mid q = \overline{0,\ k_i - 1}]$, and f_q is the $q$the element of the vector $\mathbf{F}$;

(3) otherwise, determine the variable $x_i \in \mathbf{X}$, binary operation $\otimes$, and the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ of the function f' such that the significance of this function does not exceed k and the vector of values $\mathbf{G}$ of the formula $x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ is away of the vector $\mathbf{F}$ by the least distance d, where $\psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ is the formula obtained for the function f' with the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ with the use of the present procedure;[33]

(4) if after the last step it turns out that $d(\mathbf{F},\ \mathbf{G}) = 0$, the procedure is completed with the formula $(x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'])$ which is canonized in advance (fifth production);

(5) otherwise, we determine for $d(\mathbf{F},\ \mathbf{G}) > 0$ the binary operation $\oplus$ and representation $[\mathbf{F}''\ \mathbf{K}''\ \mathbf{X}'']$ of the function f'' such that the significance of this function does not exceed k and the vector of values $\mathbf{G}$ of the formula $(x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']) \oplus \psi[\mathbf{F}''\ \mathbf{K}''\ \mathbf{X}'']$ is equal to the vector $\mathbf{F}$, where $\psi[\mathbf{F}''\ \mathbf{K}''\ \mathbf{X}'']$ is the formula obtained for the function f'' with the representation $[\mathbf{F}''\ \mathbf{K}''\ \mathbf{X}'']$ using the present procedure (sixth production);

(6) the procedure is completed by the final formula $((x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']) \oplus \psi[\mathbf{F}''\ \mathbf{K}''\ \mathbf{X}''])$ which is canonized in advance. ♦

Steps (1) and (2) of Procedure 1 are trivial. Detailed explanation and justification of the rest of the steps is given below.

---

[30] The equivalence relation $\equiv$ must be defined with regard for the possible uncertain value of the compared elements that previously was expressed by the sign $*$ in the vectors of the incompletely defined functions. In this case, any element is regarded as equivalent $*$.

[31] It is possible to demonstrate that the discrete function d is a metric in the arithmetic of natural numbers relative to the relation $\equiv$ because it satisfies the following axioms: $d(\mathbf{A},\ \mathbf{B}) = 0$ if and only if $\mathbf{A} \equiv\ \mathbf{B}$ (identity axiom); $d(\mathbf{A},\ \mathbf{B}) = d(\mathbf{B},\ \mathbf{A})$ (symmetry axiom); $d(\mathbf{A},\ \mathbf{B}) \leq d(\mathbf{A},\ \mathbf{C}) + d(\mathbf{C}, \mathbf{B})$ (triangle axiom), where $\mathbf{A},\ \mathbf{B},\ \mathbf{C}$ are any vectors of the same length.

[32] The relation between "procedure" and "algorithm" is similar to the relation between "assertion" and "theorem" (see Footnote 36): procedure is a description of a finite and effective sequence of actions executed at the semantic level, whereas only syntactic means are required to describe an algorithm. It is assumed that by formalization of informal constructions any procedure can be expressed in the formal terms of some algorithmic model (Church–Turing thesis [33, p. 88]).

[33] It is assumed that at realization of Procedure 1 for determination of the formula ψ its recursive call is used.

4.2. Initial Approximation

At Step 3 of Procedure 1, some variable x_i is specified for which determined are the operation $\otimes$ and the function f' with the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ such that the significance of this function does not exceed k and the vector of values $\mathbf{G}$ of the formula $x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ is away from the initial vector $\mathbf{F}$ by the least distance.

We first do not impose any constraints on the significance of the function f' and determine the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ of this function and the operation $\otimes$ such that $\mathbf{F} = \mathbf{G}$, where $\mathbf{G}$ is the vector of values of the formula g,

$$g = [\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i, \quad x_i \in \mathbf{X}, \quad x_i \notin \mathbf{X}'.^{34} \tag{13}$$

Procedure 2. Design of the initial (precise) formula of form (13) for the function f defined by the representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$:

(1) let the variable x_i of significance k_i be the last variable in the $\mathbf{X}$ and the variable x_j of significance k_j be the next to last one, that is, $\mathbf{X} = [\ldots\ x_j\ x_i]$ and $\mathbf{K} = [\ldots\ k_j\ k_i]$;[35]

(2) the matrix of the operation $\otimes$ is defined as follows:

$$\otimes = [p_{st} \mid\ p_{st} = f_q, \quad q = s + tK_i, \quad s = \overline{0,\, K_i - 1}, \quad t = \overline{0,\, k_i - 1}], \tag{14}$$

where f_q is the qth element of the vector $\mathbf{F}$, $K_i = m \operatorname{div} k_i$ and m is the length of the $\mathbf{F}$;

(3) for the desired function f' we determine the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ where

$$\mathbf{F}' = [f'_t \mid f'_t = t;\ t = \overline{0,\ K_i - 1}],$$
$$\mathbf{K}' = [\ldots\ k_j],$$
$$\mathbf{X}' = [\ldots\ x_j];$$

(4) we return the resulting formula $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$. ♦

Theorem 3. *The vector of values* $\mathbf{G}$ *of formula* (13) *designed using Procedure 2 is equal to the vector of the initial function* $\mathbf{F}$.

Validity of Theorem 3 is demonstrated by way of example.[36]

Example 9. Let it be required to determine the initial formula for the function of Example 1 with representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$:

$$\mathbf{F} = [0\,1\,1\,1\,2\,0\,2\,0\,2\,0\,2\,2\,0\,1\,0\,0\,2\,0\,\underline{2}\,1\,0\,2\,0\,2];$$
$$\mathbf{K} = [2\ 2\ 2\ 3];$$
$$\mathbf{X} = [x_0\ x_1\ x_2\ x_3].$$

---

[34] In what follows, instead of a formula like $x_i \otimes \psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ we use its equivalent formula $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes^T x_i$ where the matrix of the operation $\otimes$ is transposed, and instead of the formula $\psi[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$, the representation $[\mathbf{F}'\ \mathbf{K}'\mathbf{X}']$ which is not detailed to a particular expression.

[35] Since the order of variables in the representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ is changed later on, it may be that an arbitrary variable x_i occupies the last place in the vector $\mathbf{X}$ instead of x_{n-1}. The same concerns the next to last variable in this vector.

[36] Propositions whose truth is demonstrated by purely semantic means through informal reasoning are called the assertions in contrast to the theorems which are formulated and proved by strictly syntactic means expressed, possibly, in a natural language. It is believed that for every semantic proof one can determine its syntactic counterpart if the corresponding informal reasoning is formalized (Hilbert thesis [34, p. 49]). It is assumed that in this case the set theory based on the Zermelo–Fraenkel axioms [27] represented in terms of the classical predicate calculus of the first order [35] is used as the syntactic theory.

By applying Procedure 2 to this representation, we obtain

$$\begin{bmatrix} \mathbf{F}' = [0\ 1\ \underline{2}\ 3\ 4\ 5\ 6\ 7] \\ \mathbf{K}' = [2\,2\,2] \\ \mathbf{X}' = [x_0 x_1 x_2] \end{bmatrix} \underset{\otimes}{\begin{bmatrix} 0 & 2 & 2 \\ 1 & 0 & 0 \\ 1 & 2 & \underline{2} \\ 1 & 2 & 1 \\ 2 & 0 & 0 \\ 0 & 1 & 2 \\ 2 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix}} x_3,$$

or $f = [\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_3$ where the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ is used to define some function f'. By checking we make sure that calculation of the vector of values of the resulting formula provides the initial vector $\mathbf{F}$.

For example, for values of the variables $\mathbf{X}$ equal to $\boldsymbol{\lambda} = [0\ 1\ 0\ 2]$, we get $\mathbf{F}(\boldsymbol{\lambda}) = f_q = 2$ (underlined) where q is the ordinal number of the element $\mathbf{F}$ as determined using (4),

$$q = x_0 + k_0 x_1 + k_0 k_1 x_2 + k_0 k_1 k_2 x_3 = \lambda_0 + 2\lambda_1 + 4\lambda_2 + 8\lambda_3 = 18.$$

In its turn, for the same vector $\boldsymbol{\lambda}$ we determine value 2 (underlined) of the function f' from its representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ for $\lambda' = [0\ 1\ 0]$ and the value of the operation $\otimes$ for $x_3 = 2$ which is also equal to 2 (underlined). ⧫

4.3. Reduction of Formulas

The function f' determined using Procedure 2 usually has a greater significance than k appearing in the description of Procedure 1.

Definition 15. By the reduction of a formula like (13) is meant its transformation such that the vector of values of the reduced formula is away from the vector of values of the initial formula at the least distance and the significance of the function f' does not exceed certain number k. The reduction is precise if the final formula is equivalent to the initial one; otherwise, it is imprecise.

Procedure 3. Reduction of the significance k of a formula like (13) defined by the function f' with the representation $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ and operation $\otimes$:

(1) determine the set C from s classes of equivalence of the rows of the matrix $\otimes$, $C = \{C(\mathbf{r}_0),\ C(\mathbf{r}_1),\ \ldots,\ C(\mathbf{r}_{s-1})\}$ relative to the equivalence relation $\equiv$,[37] where $\mathbf{r}_j$ $(j = \overline{0,\ s-1})$ are the exemplary rows decomposing the set of rows of the matrix $\otimes$ into disjoint subsets $C(\mathbf{r}_j)$;[38]

(2) create of C a vector of equivalence classes of the row numbers $\mathbf{C}$ for which purpose:

—transform the classes $C(\mathbf{r}_j)$ in the sets C_j $(j = \overline{0,\ s-1})$ consisting not of the equivalent rows but of their numbers;

—sort the sets C_j in the nonascending order of the numbers of their elements and obtain the vector $\mathbf{C}$, $\mathbf{C} = [C_0\ C_1 \ldots C_{s-1}]$;

[37] By the class of equivalence $C(\mathbf{r})$ of the row $\mathbf{r}$ here is meant the subset of rows of the matrix of the operation $\otimes$ that are equivalent to $\mathbf{r}$, where the row equivalence relation is defined as a reflexive, symmetrical, and transitive relation of the elementwise equivalence of rows considered in their turn as ordered sets. At that, equivalence of the elements is defined as above by the relation $\equiv$.

[38] The equivalence relation is known to decompose the set into a family of disjoint subsets [28, p. 70].

(3) reduce the operation $\otimes$, that is, transform its matrix to a form having one row from each of the k first classes $\mathbf{C}$ and modify the vector $\mathbf{F}'$ as follows:

—the tth row of the matrix $\otimes$ is a row $\mathbf{r}_t$ with the number from the equivalence class C_t ($t = \overline{0,\ k-1}$);

—each element $e < k$ of the vector $\mathbf{F}'$ is replaced by the number of the equivalence class t such that $e \in C_t$;

—all elements $e \geq k$ of the vector $\mathbf{F}'$ whose equivalence classes C_j ($j = \overline{k,\ s-1}$) are disregarded are replaced by the sets of the numbers $\{t \mid t < k\}$ of the equivalence classes C_t ($t = \overline{0, k-1}$) whose rows $\mathbf{r}_t$ are away at the least distance from the rows $\mathbf{r}_j$ with the numbers of the disregarded classes C_j such that $e \in C_j$.[39] ♦

Example 10. We reduce the formula of Example 9 using Procedure 3. Let the reduction significance k be an arbitrarily large number. We establish that the number of equivalence classes of the rows of the matrix of the operation $\otimes$ is 7, their total number being 8:

$$C_0 = \{0\}, \quad C_1 = \{1\}, \quad C_2 = \{2\}, \quad C_3 = \{3\}, \quad C_4 = \{4,\ 6\}, \quad C_5 = \{5\}, \quad C_6 = \{7\}.$$

Sorting of the classes generates the vector $\mathbf{C} = [C_4\ C_0\ C_1\ C_2\ C_3\ C_5\ C_6]$ of seven elements. Since the number k is not limited, all elements of $\mathbf{C}$ are used for modification of the operation matrix $\otimes$ for which purpose the following replacement of the elements of the matrix $\otimes$ is carried out in the order of succession of the sets in $\mathbf{C}$: the elements $\{4,\ 6\}$ are replaced by 0 or $\{4,\ 6\} \to 0$; $0 \to 1$; $1 \to 2$; $2 \to 3$; $3 \to 4$; $5 \to 5$; $7 \to 6$. As the result,

$$\begin{bmatrix} \mathbf{F}' = [1\ 2\ 3\ 4\ 0\ 5\ 0\ 6] \\ \mathbf{K}' = [2\ 2\ 2] \\ \mathbf{X}' = [x_0\ x_1\ x_2] \end{bmatrix} \underset{\otimes}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 2 \\ 1 & 0 & 0 \\ 1 & 2 & 2 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 2 \end{bmatrix}} x_3. \quad ♦$$

If in the course of reduction the number of rows in the operation matrix is at most k, then the reduction is precise; otherwise, it is imprecise. In the last example reduction turned out to be precise because of the lack of constraints on k. Now we consider an example of imprecise reduction.

Example 11. We reduce the function of Example 10 for limited significance of reduction. Let $k = 3$. Transformation of the operation $\otimes$ provides the formula

$$\begin{bmatrix} \mathbf{F}' = [1\ 2\ \{1\}\ \{12\}\ 0\ \{1\}\ 0\ \{1\}] \\ \mathbf{K}' = [2\ 2\ 2] \\ \mathbf{X}' = [x_0\ x_1\ x_2] \end{bmatrix} \underset{\otimes}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 2 \\ 1 & 0 & 0 \end{bmatrix}} x_3,$$

[39] Replacement of the elements which have no equivalence classes by the sets of numbers of the selected classes is explained by the fact that there may be more than one equivalence class which is away at the least distance from the row of the replaced element. In its turn, at subsequent reduction only those elements of these sets are used which minimize the number of equivalence classes of the rows of the matrix of operation.

Table 4. Incompletely defined function given with the precision 0.5

x_0	x_1	x_2	f
0.2 (0)	−1.7 (0)	0.1 (0)	0.1 (1)
0.7 (1)	−1.8 (0)	0.0 (0)	0.9 (2)
0.3 (0)	−1.6 (0)	0.5 (1)	−0.4 (0)
0.1 (0)	−1.3 (1)	0.9 (1)	−0.1 (0)
0.6 (1)	−1.4 (1)	0.7 (1)	0.7 (2)
0.9 (1)	−1.9 (0)	1.1 (2)	0.3 (1)
0.4 (0)	−1.0 (1)	1.0 (2)	0.8 (2)
0.5 (1)	−1.1 (1)	1.4 (2)	−0.3 (0)

where the brackets embrace the function values minimizing identically the distance between the vectors of the initial and reduced formulas. Selection of an alternative value of the elements of the vector $\mathbf{F}'$ provides

$$\begin{bmatrix} \mathbf{F}' = [1\ 2\ 1\ 1\ 0\ 1\ 0\ 1] \\ \mathbf{K}' = [2\,2\,2] \\ \mathbf{X}' = [x_0 x_1 x_2] \end{bmatrix} \underset{\otimes}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 2 \\ 1 & 0 & 0 \end{bmatrix}} x_3.$$

Calculation of this formula for every possible vector of values of the variables in the order defined by (4) provides the vector

$$\mathbf{G} = [0\,1\,0\,0\,2\,0\,2\,0\,2\,0\,2\,0\,2\,2\,0\,2\,0\,2\,2\,0\,2\,2\,0\,2\,0\,2],$$

which already is not equal to the vector of the initial function $\mathbf{F}$. ♦

If the initial table of the function is defined incompletely, the reduction replaces the missing values of the function by the sign * comparable with any other value including * as well. If the analytical identification of a function must be performed with the assigned precision ε, then the reduction is preceded by decomposing the definition domain of the initial function and the definition domains of the variables into a finite family of disjoint subdomains determined on the basis of the value of ε, and the values of the function and variables are replaced by the numbers of the subdomains which they hit.

Example 12. We reduce the significance $k = 3$ of the function incompletely defined in Table 4 to within $\varepsilon = 0.5$. As can be seen from the table, the function f varies within the range from -0.5 to 1.0, the variable x_0, from 0.0 to 1.0, the variable x_1, from -2.0 to -1.0, and the variable x_2, from 0.0 to 1.5. After passing from the real values of the function and its variables to the numbers of subdomains, we obtain the discrete function shown in Table 4 in parentheses. Next, we obtain its representation [**F K X**]:

$$\mathbf{F} = [1\,2 * * 0 * 0\,2 * 1\,2\,0];$$
$$\mathbf{K} = [2\ 2\ 3];$$
$$\mathbf{X} = [x_0\ x_1\ x_2].$$

Let us determine the initial approximation of the obtained function:

$$\begin{bmatrix} \mathbf{F}' = [0\ 1\ 2\ 3] \\ \mathbf{K}' = [2\ 2] \\ \mathbf{X}' = [x_0\ x_1] \end{bmatrix} \begin{bmatrix} 1 & 0 & * \\ 2 & * & 1 \\ * & 0 & 2 \\ * & 2 & 0 \end{bmatrix} x_2$$

and perform reduction in the variable x_2:

$$\begin{bmatrix} \mathbf{F}' = [0\ 1\ 0\ 2] \\ \mathbf{K}' = [2\ 2] \\ \mathbf{X}' = [x_0\ x_1] \end{bmatrix} \begin{bmatrix} 1 & 0 & 2 \\ 2 & * & 1 \\ * & 2 & 0 \end{bmatrix} x_2.$$

If the representations $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}']$ are reduced and the incompletely defined operation are redefined, then we get the final formula[40]

$$\begin{bmatrix} -0.25 \\ 0.25 \\ 0.75 \end{bmatrix} \left(\left(x_0 \begin{bmatrix} 0 & 0 \\ 1 & 2 \end{bmatrix} x_1 \right) \begin{bmatrix} 1 & 0 & 2 \\ 2 & 0 & 1 \\ 0 & 2 & 0 \end{bmatrix} x_2 \right),$$

where a special unary operation transforming the numbers of the subdomains into the mean values of their corresponding intervals was used to restore the initial function. It goes without saying that the calculation of this formula is preceded by a similar inverse transformation of the initial values of the object parameters expressed by the corresponding variables. ♦

4.4. Permutation of Variables

The number of the equivalence classes of the rows of the operation $\otimes$ in Procedure 3 depends on the last variable in the initial representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$. That is why the reduction is preceded by seeking a representation $[\tilde{\mathbf{F}}\ \tilde{\mathbf{K}}\ \tilde{\mathbf{X}}]$ such that it is equivalent to the initial one but leads to minimization of the number of equivalence classes of the rows of the desired operation $\otimes$. For that we change the order of variables by permuting the elements in the vectors $\mathbf{X}$ and $\mathbf{K}$. To retain equivalence of the representations $[\tilde{\mathbf{F}}\ \tilde{\mathbf{K}}\ \tilde{\mathbf{X}}]$ and $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$, the vector $\mathbf{F}$ of values itself must be modified in a way.

Procedure 4. Permutation of the variables in the representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$:

(1) let given be the permutation $\mathbf{S}$ of variables,

$$\mathbf{S} = [s_i \mid s_i \in \mathbf{N}_n;\ s_i \not\equiv s_j \text{ for } i \neq j;\ i = \overline{0,\ n-1}];$$

(2) we apply the permutation $\mathbf{S}$ to the vectors $\mathbf{K}$ and $\mathbf{X}$ and as a result obtain the vectors $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{X}}$ with identical permutation of their elements;

(3) for each element f_q of the vector of values $\mathbf{F} = [f_q \mid q = \overline{0, m-1}]$, perform the following actions:

—determine for q a corresponding vector of values of the variables $\boldsymbol{\lambda} = [\lambda_i \mid i = \overline{0, n-1}]$ for which purpose make use of the recurrent rule (5) for the initial vector of the significances of the variables $\mathbf{K}$;

—apply the permutation $\mathbf{S}$ to the vector $\boldsymbol{\lambda}$ and obtain a new vector $\tilde{\boldsymbol{\lambda}}$;

—calculate for $\tilde{\boldsymbol{\lambda}}$ the new value of the index $\tilde{q}$ with the use of (4) yet for the vector of significances of the variables $\tilde{\mathbf{K}}$;

—set down the element f_q in the position $\tilde{q}$ of the desired vector $\tilde{\mathbf{F}}$;

(4) the procedure is completed with the new representation $[\tilde{\mathbf{F}}\ \tilde{\mathbf{K}}\ \tilde{\mathbf{X}}]$. ♦

[40] We notice that in the course of formula design the initial function was redefined so as to enable its determination at the points of the definition domain that were not given by the initial table. Approximation of the function that is done at that is based on the requirement for getting the simplest expression relating the value of the function with the values of its variable. It deserves noting that the continuous functions are approximated in a similar manner. In this case, restoration of the missed values is based on the assumption of function smoothness, which also is a simplification of its formulaic expression.

Table 5. Intermediate results of Example 14

X	**F**	**G**	$d(\mathbf{F}, \mathbf{G})$
$[x_0\ x_1\ x_2\ x_3]$	[011120202022010020210202]	$[0\underline{2}112020\underline{1}022010\underline{1}20\underline{1}10202]$	4
$[x_3\ x_0\ x_1\ x_2]$	[022100122121200012200002]	$[022\underline{2}00122\underline{2}212000\underline{2}2200002]$	3
$[x_2\ x_3\ x_0\ x_1]$	[022020100102122020102012]	$[022020\underline{0}0\underline{2}\underline{0}02122020102012]$	3
$[x_1\ x_2\ x_3\ x_0]$	[012222002200110002100122]	$[0\underline{0}22220\underline{2}22001100\underline{0}0100\underline{0}22]$	4

Example 13. For the function of Example 9, determine its representation after a cyclic right shift of the vector of variables and their significances, that is, for $\mathbf{S} = [1\ 2\ 3\ 0]$. Application of Procedure 4 to the initial representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$, where

$$\mathbf{F} = [0\,1\,1\,1\,2\,0\,2\,0\,2\,0\,2\,2\,0\,1\,0\,0\,2\,0\,2\,1\,0\,2\,0\,2];$$
$$\mathbf{K} = [2\ 2\ 2\ 3];$$
$$\mathbf{X} = [x_0\ x_1\ x_2\ x_3],$$

provides a new representation of the function $[\tilde{\mathbf{F}}\ \tilde{\mathbf{K}}\ \tilde{\mathbf{X}}]$:

$$\tilde{\mathbf{F}} = [0\,2\,2\,1\,0\,0\,1\,2\,2\,1\,2\,1\,2\,0\,0\,0\,1\,2\,2\,0\,0\,0\,0\,2];$$
$$\tilde{\mathbf{K}} = [3\ 2\ 2\ 2];$$
$$\tilde{\mathbf{X}} = [x_3\ x_0\ x_1\ x_2]. \quad \blacklozenge$$

Theorem 4. *The representation* $[\tilde{\mathbf{F}}\ \tilde{\mathbf{K}}\ \tilde{\mathbf{X}}]$ *obtained from the representation* $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ *with the use of Procedure 4 and arbitrary permutation of the variables* $\mathbf{S}$ *is equivalent to* $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$.

Theorem 4 is proved in the Appendix.

Let us consider now the determination of the best permutation of the variables $\mathbf{S}$ in the representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ minimizing the number of the equivalence classes of the rows of operation $\otimes$ at reduction of the formula $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$ by Procedure 3.

Procedure 5. Search of the closest formula of significance k and form $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$ for the function f with representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$:

(1) use Procedure 2 to determine the equivalent formula g for the function f, $g = [\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_t$, where x_t is the last variable in the vector $\mathbf{X}$;

(2) reduce the formula g for the given significance k with the use of Procedure 3;

(3) calculate and store the distance $d(\mathbf{F},\ \mathbf{G})$ between the vector $\mathbf{F}$ and the vector of values $\mathbf{G}$ of the formula g;

(4) if all variables were at the last position of the vector $\mathbf{X}$, then go to Step 5; otherwise, use Procedure 4 to calculate a new representation $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ under the cyclic right shift of the variable and repeat the calculation from Step 1;

(5) take a permutation of variables with the least value of $d(\mathbf{F},\ \mathbf{G})$ and complete the procedure with the formula $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$ corresponding to this permutation. ♦

Example 14. We determine the best formula for the function of Example 13 by applying Procedure 5 to the initial representation of the function and compile the intermediate results in Table 5 where nonequivalent elements of the vectors $\mathbf{G}$ and $\mathbf{F}$ are underlined.

We restore the second formula for the result with the distance 3:

$$\left[\begin{array}{l}\mathbf{F}' = [2\ 0\ 0\ 1\ 0\ 1\ \{2\}\ 1\ \{0\,1\,2\}\ \{1\}\ 2\ 0] \\ \mathbf{K}' = [2\ 2\ 3] \\ \mathbf{X}' = [x_2\ x_3\ x_0]\end{array}\right] \underset{\otimes}{\begin{bmatrix} 2 & 2 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}} x_1. \quad \blacklozenge$$

Since there are $n!$ permutations of the n-element vector $\mathbf{X}$ and Procedure 5 analyzes only n cyclic permutations of variables, the following theorem seems to be topical.

Theorem 5. *Let given be the formula* $[\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$ *obtained from the representation* $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ *with the use of Procedure 2. Then, the number of row equivalence classes of the operation* $\otimes$ *does not vary at permutations of the variables in* $[\mathbf{F}\ \mathbf{K}\ \mathbf{X}]$ *leaving the variable* x_i *in its place.*

Theorem 5 is proved in the Appendix.

We notice that the use only of the cyclic permutations of variables in Procedure 5 is not obligatory. As can be seen from the proof of Theorem 5, it is possible to use any n permutations which successively move to the last position of the vector $\mathbf{X}$ each of its n variables.

4.5. Calculation of the Residue

If it turns out that at Step 4 of Procedure 1 a formula g was obtained which is not equivalent to the identified function f, that is, the vector of values $\mathbf{G}$ of the formula $g = [\mathbf{F}'\ \mathbf{K}'\ \mathbf{X}'] \otimes x_i$ is not equal to the vector of values $\mathbf{F}$ of the function f, then Step 5 is executed at which the binary operation $\oplus$ and the residue vector $\mathbf{F}''$ are determined such that $\mathbf{G} \oplus \mathbf{F}'' = \mathbf{F}$.[41] The following procedure demonstrates how this can be done.

Procedure 6. Construction of the operation $\oplus$ and the residue vector $\mathbf{F}''$ from the vector $\mathbf{G}$ and $\mathbf{F}$:

(1) determine the number of entries e_{st} of the ordered pairs $[s\ t]$ $(s,\ t = \overline{0, k-1})$ into the set $\{[f_i\ g_i] \mid i = \overline{0,\ m-1}\}$ generated by the vectors $\mathbf{F}$ and $\mathbf{G}$ and construct the matrix $\tilde{\oplus} = [e_{st} \mid s,\ t = \overline{0, k-1}]$;

(2) each row $\mathbf{r}_s = [e_{st} \mid t = \overline{0, k-1}]$ of the matrix $\tilde{\oplus}$ is replaced by the permutation of the numbers of its columns driving the elements of this row into a monotonically nonincreasing sequence; as the result we obtain the matrix of the desired operation $\oplus$;

(3) generate m equations like $g_i \oplus f_i'' = f_i$ $(i = \overline{0, m-1})$ and after solving them[42] in f_i'' determine the elements of the residue vector $\mathbf{F}''$, $\mathbf{F}'' = [f_0''\ f_1''\ \ldots f_{m-1}'']$. $\blacklozenge$

Example 15. Determine the operation $\oplus$ and the residue vector $\mathbf{F}''$ for imprecisely reduced formula of Example 14. The source data for Procedure 6 are as follows:

$$\mathbf{F} = [0\,2\,2\,0\,2\,0\,1\,0\,0\,1\,0\,2\,1\,2\,2\,0\,2\,0\,1\,0\,2\,0\,1\,2];$$
$$\mathbf{G} = [0\,2\,2\,0\,2\,0\,0\,0\,2\,0\,0\,2\,1\,2\,2\,0\,2\,0\,1\,0\,2\,0\,1\,2].$$

We calculate the number of entries of each pair of elements occurring at the identical positions of the vectors $\mathbf{F}$ and $\mathbf{G}$ and represent the results as the matrix $\tilde{\oplus}$, and obtain the desired operation $\oplus$

[41] We assume that the binary operation $\circ$ defined for the elements a_i and b_i $(i = \overline{0, m-1})$ of the vectors $\mathbf{A}$ and $\mathbf{B}$ is applicable to the vectors themselves. In this case, a notation like $\mathbf{A} \circ \mathbf{B}$ denotes a vector $\mathbf{C}$ such that $\mathbf{C} = [c_i \mid i = \overline{0,\ m-1};\ c_i = a_i \circ b_i]$.

[42] Solutions of these equations always exist and are unique (see Theorem 6).

by replacing the rows of this matrix by the permutations of its elements driving the initial rows in a monotonically nonincreasing sequences:

$$\tilde{\oplus} = \begin{bmatrix} 9 & 2 & 0 \\ 0 & 3 & 0 \\ 1 & 0 & 9 \end{bmatrix}; \quad \oplus = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 0 \\ 2 & 0 & 1 \end{bmatrix}.$$

As can be seen from the matrix $\tilde{\oplus}$, the pair $[i\ j]$ with the values [0 0] is repeated nine times, [0 1] 2 times, [0 2] 0 times (the first row of $\tilde{\oplus}$), and so on, and that with the value [2 2] 9 times (the last row of $\tilde{\oplus}$). We also notice that none of the rows of the matrix $\oplus$ has two identical elements. Consequently, for arbitrary a and b the equation $a \oplus c = b$ has a single solution in c.

Now, we set down the equation $\mathbf{G} \oplus \mathbf{F}'' = \mathbf{F}$ for calculation of the elements of the residual vector $\mathbf{F}''$. Its solution provides

$$\mathbf{F}'' = [0\,0\,0\,0\,0\,0\,1\,0\,2\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0].$$

Analysis of the vector $\mathbf{F}''$ shows that element 0 occurs 21 time, element 1 twice, and element 2, once, that is, the vector of the number of entries of the elements [21 2 1] is monotonically nonincreasing. ♦

Theorem 6. *Let given be two vectors* $\mathbf{F}$ *and* $\mathbf{G}$ *of length* m *and significance* k. *Then, the following statements are valid for the groupoid*[43] $[\mathbf{N}_k\ \oplus]$ *and the residual vector* $\mathbf{F}''$ *obtained in Procedure 6.*

(a) *for any elements* a *and* c *of the groupoid* $[\mathbf{N}_k\ \oplus]$, *there exists a single its element* b *such that* $a \oplus b = c$;

(b) *the vector* $[e_s \mid s = \overline{0, k-1}]$ *composed of the numbers of entries* $e_s \in \mathbf{N}_{m+1}$ *of the elements* $s \in \mathbf{N}_k$ *in the residual vector* $\mathbf{F}''$, *is monotonically nonincreasing, that is,* $e_t \geq e_s$ *for all* $t \leq s$.

Theorem 6 is proved in the Appendix.

In the constructed groupoid, therefore, equations like $g_i \oplus f_i'' = f_i$ have solutions in the unknowns f_i'', where g_i, f_i'', and f_i $(i = \overline{0, m-1})$ are the elements of the vectors $\mathbf{G}$, $\mathbf{F}''$, and $\mathbf{F}$, respectively. Additionally, at calculation of the residue the frequently occurring elements of the residual vector $\mathbf{F}''$ are coded by smaller numbers, and the rarely occurring ones, by larger numbers. If an element did not appear even once, it will be coded by the largest number, which is equivalent to reduction in the function significance as expressed by the vector $\mathbf{F}''$. All this results finally in reduced complexity of the formula f''.

4.6. Demonstration Example

Without repeating again the explanations of the steps of Procedure 1 given in the above examples, we illustrate the final result of the analytical identification of the discrete functions.

Example 16. Let the function of Example 1 be given. We take the significance of the desired formula equal to 3. Application of Procedure 1 to the function with the representation

$$\mathbf{F} = [0\,1\,1\,1\,2\,0\,2\,0\,2\,0\,2\,2\,0\,1\,0\,0\,2\,0\,2\,1\,0\,2\,0\,2],$$
$$\mathbf{K} = [2\ 2\ 2\ 3],$$
$$\mathbf{X} = [x_0\ x_1\ x_2\ x_3],$$

provides

$$((x_1 \otimes_0 (x_2 \otimes_1 (x_0 \otimes_2 x_3))) \oplus (x_1 \otimes_4 (x_0 \otimes_5 (x_3 \otimes_6 x_2)))),$$

[43] By the groupoid is meant the set with binary operation over it.

where

$$\otimes_0 = \begin{bmatrix} 2 & 0 & 0 \\ 2 & 0 & 1 \end{bmatrix}, \quad \otimes_1 = \begin{bmatrix} 0 & 2 & 2 \\ 1 & 0 & 1 \end{bmatrix}, \quad \otimes_2 = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 1 \end{bmatrix},$$

$$\otimes_4 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \otimes_5 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \otimes_6 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}, \quad \oplus = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}. \quad \blacklozenge$$

If one interprets the analytical identification as the design of a combinatorial automaton, then in the above example sought is the description of this automaton in the binary and ternary logic ($k \leq 3$). The formula obtained at that can be transformed into a diagram of the established logic elements (operations). However, at analytical identification operations may occur that are not realizable by the available assembly of logic elements.

Therefore, compactness of the formulas provided by analytical identification is attained through rejection of a fixed basis of operations and rigid analytical construction of the formulas.

5. COMPLEXITY ESTIMATION

Let us consider a canonical analytical construction (12) and determine the number of applications of the productions $\Psi \to ((\mathrm{X}\,\Theta\,\Psi)\Theta\,\Psi)$ for identification of a complex discrete function.[44] By a $(t-1)$-fold application of the production $\Psi \to ((\mathrm{X} \otimes \Psi) \oplus \Psi)$ and single application of the production $\Psi \to (\mathrm{X} \otimes \Psi)$ to the last entry Ψ to the resulting formula, we obtain

$$\Psi \to ((\mathrm{X}_1 \otimes_1 \Psi_1) \oplus_1 ((\mathrm{X}_2 \otimes_2 \Psi_2) \oplus_2 (\ldots \oplus_{t-1} (\mathrm{X}_t \otimes_t \Psi_t) \ldots))), \tag{15}$$

where different entries of the operations $\otimes$ and $\oplus$, variables X, and subformulas Ψ are supplied with subscripts and t is the number of identification steps. To simplify calculations, we assume in what follows that the identified function, its variables, and the designed formula have identical significance equal to k.

Theorem 7. *The maximal number of steps of identification $t_{\max}$ of an arbitrary discrete function of significance k in n variables of significance k is at most as follows:*

$$k\frac{n-2}{n-1}.\text{[45]} \tag{16}$$

Theorem 7 is proved in the Appendix.

Using now (16), we estimate the length of formulas designed using Procedure 1.

Theorem 8. *The maximal length of the formulas $L_n(k)$ obtained at identification of an arbitrary discrete function of significance k in n variables of significance k is at most as follows:*

$$\frac{2(k-1)^2+k}{k(k-1)^2}\frac{k^{n-1}}{n-1} - \frac{k}{(k-1)^2}\frac{1}{n-1} - \frac{1}{(k-1)}\frac{n-2}{n-1} - 1. \tag{17}$$

Theorem 8 is proved in the Appendix.

[44] By the complex discrete function is meant here a function that indescribable by formulas like Δ, X and ΞX.

[45] The fractional value of the number of identification steps must be interpreted as the mean number of steps required to identify complex functions of the given number of variables.

Table 6. Analytical identification of the Boolean functions of three variables

Function	Operations
$\mathbf{F} = [00000001]$, $(x_2 \otimes_0 (x_1 \otimes_0 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
$\mathbf{F} = [00000010]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00000011]$, $(x_2 \otimes_0 x_1)$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
$\mathbf{F} = [00000100]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$
$\mathbf{F} = [00000101]$, $(x_2 \otimes_0 x_0)$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
$\mathbf{F} = [00000110]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00000111]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$
$\mathbf{F} = [00001000]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$
$\mathbf{F} = [00001001]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00001010]$, $(x_2 \otimes_0 x_0)$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00001011]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$
$\mathbf{F} = [00001100]$, $(x_2 \otimes_0 x_1)$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00001101]$, $(x_2 \otimes_0 (x_1 \otimes_0 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$
$\mathbf{F} = [00001110]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
$\mathbf{F} = [00001111]$, x_2	
$\mathbf{F} = [00010000]$, $(x_2 \otimes_0 (x_1 \otimes_1 x_0))$	$\otimes_0 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $\otimes_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$

In particular, for $n = 3$ and $k = 2$ of (17) we get $L_3(2) = 2$, that is, for the analytical identification of a Boolean function of three variables[46] two operation suffice, which is corroborated by the experimental data compiled in Table 6.[47]

[46] For large n, computational experiment over all functions entails insurmountable difficulties. However, it is seen from (16) that with increase in n the maximal number of identification steps $t_{\max}$ tends rapidly to k. Consequently, the asymptotic behavior of $L_n(k)$ is defined at large by the length of formulas at small number of variables.

[47] To consider the entire set of the Boolean functions of three variable, it suffices to consider the functions defined to within inversion of variables and inversion of the function itself. The inversion of variables gives rise to permutation of rows (columns) of the corresponding operations, and inversion of the function, permutation of values of the last calculated operation (see Example 8).

Complexity of identification of the Boolean functions in the finite and asymptotic domains in terms of the number of variables n can be obtained from (17) for $k = 2$.

$$L_n(2) = \frac{2^n - n - 1}{n - 1}, \quad L_n(2) \sim \frac{2^n}{n}.$$

The length of the formula $L_n(2)$ of complex Boolean functions of n variables is known to satisfy the following estimates [36–38]:[48]

$$\frac{1}{\log_2 3}\frac{2^n}{n} < L_n(2) < 2(1 + \log_2 n)\frac{2^n}{n},^{49} \quad L_n(2) \sim \frac{2^n}{n}.$$

Consequently, the developed theory of analytical design of formulas is the best at least for the Boolean functions.

6. CONCLUSIONS

The considered problem of analytical identification of the discrete objects is claimed for by practice. Its formulation is distinct from that of other, allied problems such as nonparametric and parametric identification, functional constructions in the finite algebras, or data analysis. The distinctive feature of the developed method of identification lies in using a new mathematical apparatus relying on the reduction of a discrete function and its successive approximations, which enables one to calculate in the course of identification not only the operations in the function formula, but also to determine the order of variables and the places of bracket arrangement.

This mathematical apparatus also enables one to calculate the output parameters of the object for those values of the input parameter that were not yet observed. In this case, the object's behavior is forecasted by approximation (interpolation and extrapolation) of incompletely defined discrete function subject to the condition of getting the simplest formula relating the function's values with those of its variables.

The developed procedure of analytical identification of the discrete objects is supported by a simple effective algorithm to design the formulas of discrete functions with the best complexity estimates both in the finite and asymptotic domains. Additionally, since the resulting formulas consist of a sequence of discrete operations over discrete variables, they are well suited for realization on the existing discrete computing facilities.

It deserves noting that the described approach is applicable not only to identification of the discrete memoryless objects. Its union with the finite-automaton formalism enables one to identify objects operating in time and, thus, varying their states. In this case, the process of identification lies in designing a transition function describing variations of the object state and the output functions depending on the current state and providing the values of the model output variables.

Within the framework of the finite-automaton approach, the problem of analytical identification lies in determining the set of automaton states and the tables of the transition and output functions. It is assumed that the input and output alphabets of the automaton are given. As was proved in [39], the behavior of a q-state automaton is restored by a repeated experiment of length $2q - 1$. Stated differently, for complete and reliable identification of the finite automaton, one must feed to its

[48] $L_n(2)$ is called the Shannon function defined as $\max \min \omega_i$, where ω_i is the number of operations required to express the ith function and the subscript i runs the numbers of all functions.

[49] Roughness of the upper estimate from [37] is due to the fact that its derivation made use of the analytical construction with only two binary Boolean operations—conjunction and inequivalence—of 10 possible, as well as to the use of constant 1 together with inequivalence to express the unary negation which, as was shown above, is redundant for function identification. Additionally, the analytical construction of formulas in [37] makes use of the rigid order of bracket arrangement, bracket embeddedness being unlimited.

input every possible input sequences of length $2q-1$. There are precisely p^{2q-1} such sequences for p symbols in the input alphabet.

However, if the number of automaton states is unknown, then its identification is possible even in the limit, that is, on the countable set of the input rows. Nevertheless, if an automaton with q' states is determined whose description is not contradicted by the subsequent tests of length greater than $2q'-1$, then one can assume with high degree of confidence that the automaton is identified.

It deserves noting that the applied problems of analytical identification have limited complexity and in practice one succeeds to do without lasting experiments. For example, it was shown in [32] that almost for all automata the degree of their restoration by behavior has the order of $\log q$, which is much smaller than $2q-1$. In its turn, the fraction of automata for which the above estimate is not satisfied vanishes rapidly with q.

APPENDIX

Proof of Theorem 1. A. We prove that any formula of the construction Φ like (7) is identically transformed into a formula with construction K like (11). For that, we

—determine the variants of the entries of the constants Δ and the unary operations Ξ in the formulas generated by the analytical construction Φ;

—establish equivalence of these entries to other combinations of symbols;

—derive a new form of analytical constructions of the formulas obtained from the identical transformations of the initial formulas with regard for the established equivalences and compare it with K.

(1) Equivalences for the entries of the constants Δ in the formula of the construction Φ:

(a) $\Xi\,\Delta \sim \Delta$;[50]

(b) $\Delta\,\Theta\,\Phi \sim \Xi\,\Phi$;

(c) $\Phi\,\Theta\,\Delta \sim \Xi\,\Phi$;

(d) $\Delta\,\Theta\,\Delta \sim \Delta$.

The equivalences (a) and (d) are trivial. We prove equivalence of (b). Let the constant Δ be equal to δ, and the operation Θ be defined by the matrix $\bullet$. Then, δ is equal to the number of the row of matrix $\bullet$ used in the calculations $\delta \bullet \Phi$. This calculation amounts to applying to the result of calculating the subformula Φ the unary operation $\neg$ with a vector equal to the row δ of the matrix $\bullet$, that is, $\delta \bullet \Phi = \neg\,\Phi$. Whence it follows that $\Delta\,\Theta\,\Phi \sim \Xi\,\Phi$ in virtue of arbitrary choice of the constant δ and matrix $\bullet$, which is what we set out to prove. Equivalence of (c) is proved along the same lines.

It follows from the analysis of the equivalences (a)–(d) that the formula with the analytical construction Φ is transformed identically into a formula without entries of the constants Δ, that is, has the analytical construction Φ':

$$\Phi' \to \mathrm{X}; \quad \Phi' \to \Xi\,\Phi'; \quad \Phi' \to (\Phi'\,\Theta\,\Phi').$$

(2) Equivalences for the entry of the unary operations Ξ in the construction Φ':

(a) $\Xi\,\Xi\Phi' \sim \Xi\,\Phi'$;

(b) $\Xi(\Phi'\,\Theta\,\Phi') \sim \Phi'\,\Theta\,\Phi'$;

(c) $\Xi\,\Phi'\,\Theta\,\Phi' \sim \Phi'\,\Theta\,\Phi'$;

[50] By equivalence of two rows is meant here the equality of the sets of formulas derived from them using the grammar productions with allowance for the possible identical transformations of these formulas.

(d) $\Phi' \, \Theta \, \Xi \, \Phi' \sim \Phi' \, \Theta \, \Phi'$;

(e) $\Xi \, \Phi' \, \Theta \, \Xi \, \Phi' \sim \Phi' \, \Theta \, \Phi'$.

We prove equivalence (c). Let the operations Ξ and Θ be defined by the vector $\neg$ and the matrix $\bullet$. We carry out the identical transformations $\neg \, \Phi' \bullet \Phi'$ by copying the rows of the matrix $\bullet$ to the matrix of the new operation $\circ$ in the order defined by the elements of the $\neg$. As the result of such transformations, we obtain $\neg \, \Phi' \bullet \Phi' = \Phi' \circ \Phi'$ and $\Xi \, \Phi' \, \Theta \, \Phi' \sim \Phi' \, \Theta \, \Phi'$. The rest of the equivalences are proved along the same lines.

It follows from the analysis of the equivalences (a)–(e) that the formula with analytical construction from subitem (1) is transformed identically into the formula with construction Φ'':

$$\Phi'' \to \mathrm{X}; \quad \Phi'' \to (\Phi'' \, \Theta \, \Phi'').$$

Identification of Φ'' and K provides (11), which proves item (A).

(B) Each of the above identical transformations does not increase the formula length. Consequently, the resulting formula has a length shorter than or equal to the length of the initial formula. ♦

Proof of Theorem 2. To prove the theorem, it suffices to demonstrate that the grammars differing in the sets of productions $\Psi \to \mathrm{X} \mid (\Psi \, \Theta \, \Psi)$ and $\Psi \to \mathrm{X} \mid (\mathrm{X} \, \Theta \, \Psi) \mid ((\mathrm{X} \, \Theta \, \Psi) \Theta \, \Psi)$ generate the same formulas. The proof is done by induction on the number of operations Θ in the formula. First, we notice that in both grammars there exists a single production $\Psi \to \mathrm{X}$ generating the terminal entry of the variable and the application of any other production can increase by 1 the number of formula operations in the first grammar and by 2 in the second grammar. Consequently, the inductive reasoning must be done for two successive values of the numbers of operations n in the formula.

We directly make sure that the sets of formulas generated by the grammars are $\{\mathrm{X}\}$ for $n = 0$, $\{(\mathrm{X} \, \Theta \, \mathrm{X})\}$ for $n = 1$, and $\{((\mathrm{X} \, \Theta \, \mathrm{X}) \Theta \, \mathrm{X}), \ (\mathrm{X} \, \Theta \, (\mathrm{X} \, \Theta \, \mathrm{X}))\}$ for $n = 2$.

Let the sets of formulas of two these grammars be equal for some n. We prove that these sets are equal also for $n + 1$ and $n + 2$. For $n + 1$, at the last step the first grammar needs instead of the substitution $\Psi \to \mathrm{X}$ the single possible sequence of substitutions $\Psi \to (\Psi \, \Theta \, \Psi) \to (\mathrm{X} \, \Theta \, \Psi) \to (\mathrm{X} \, \Theta \, \mathrm{X})$. In the second grammar, we use $\Psi \to (\mathrm{X} \, \Theta \, \Psi) \to (\mathrm{X} \, \Theta \, \mathrm{X})$ instead of $\Psi \to \mathrm{X}$.

For $n+2$, in the first grammar at the last step of inference we make one of the following feasible substitutions:

$$\Psi \to (\Psi \, \Theta \, \Psi) \to ((\Psi \, \Theta \, \Psi) \Theta \, \Psi) \to \ldots \to ((\mathrm{X} \, \Theta \, \mathrm{X}) \Theta \, \mathrm{X});$$
$$\Psi \to (\Psi \, \Theta \, \Psi) \to (\Psi \, \Theta (\Psi \, \Theta \, \Psi)) \to \ldots \to (\mathrm{X} \, \Theta (\mathrm{X} \, \Theta \, \mathrm{X})),$$

and in the second grammar we use

$$\Psi \to ((\mathrm{X} \, \Theta \, \Psi) \Theta \, \Psi) \to \ldots \to ((\mathrm{X} \, \Theta \, \mathrm{X}) \Theta \, \mathrm{X});$$
$$\Psi \to (\mathrm{X} \, \Theta \, \Psi) \to (\mathrm{X} \, \Theta \, (\mathrm{X} \, \Theta \, \Psi)) \to (\mathrm{X} \, \Theta (\mathrm{X} \, \Theta \, \mathrm{X})).$$

The performed inductive transition proves the theorem. ♦

Proof of Theorem 4. It follows from the construction that the equality $\mathbf{F}(\boldsymbol{\lambda}) = \tilde{\mathbf{F}}(\tilde{\boldsymbol{\lambda}})$, where $\tilde{\mathbf{F}}$ and $\tilde{\boldsymbol{\lambda}}$ are, respectively, the vectors of functions and values of the variable as determined using Procedure 4 with a certain permutation of variable $\mathbf{S}$, is satisfied for any vector of values of the variables $\boldsymbol{\lambda}$. The equality $\mathbf{F}(\boldsymbol{\lambda}) = \tilde{\mathbf{F}}(\tilde{\boldsymbol{\lambda}})$ is interpreted here as that of the elements of the vectors $\mathbf{F}$ and $\tilde{\mathbf{F}}$ obtained for arbitrary but corresponding to each other vectors of values of the variable $\boldsymbol{\lambda}$ and $\tilde{\boldsymbol{\lambda}}$, that is, for identical values of the variable but different orders of their arrangements in the vectors $\boldsymbol{\lambda}$ and $\tilde{\boldsymbol{\lambda}}$. Whence it follows that the representations $[\mathbf{F} \ \mathbf{K} \ \mathbf{X}]$ and $[\tilde{\mathbf{F}} \ \tilde{\mathbf{K}} \ \tilde{\mathbf{X}}]$ are equivalent. ♦

Proof of Theorem 5. We represent without loss of generality $\mathbf{X} = [x_0\ x_1 \ldots x_{n-2}\ x_{n-1}]$ as $[X'\ x_{n-1}]$, where X' is a new variable replacing the variables $x_0,\ x_1,\ \ldots,\ x_{n-2}$ and having the significance $K' = k_0 k_1 \ldots k_{n-2}$. Then, according to (14) we obtain $p_{st} = f_q$ for the elements p_{st} of the matrix of operation $\otimes$, where q is the subscript of an element from the vector of values of the function ($q = s + K't$), s is the subscript of the row of this element in the matrix $\otimes$ equal to the value of the variable X' ($s = \overline{0,\ K'-1}$), and t is the subscript of its column equal to the value of the variable x_n ($t = \overline{0,\ k_n - 1}$).

Whence it follows that any permutation $\mathbf{S}$ not affecting the variable x_n changes (permutes) the rows of elements without changing their columns. This fact implies that the number of row equivalence classes is independent of the considered set of permutations, which is what we set out to prove. ♦

Proof of Theorem 6. Indeed, the groupoid $[\mathbf{N}_k\ \oplus]$ whose operation matrix consists of the rows equal to the permutations of the elements of the set $\mathbf{N}_k$ is constructed at step (2) of Procedure 6. Then, all elements of the set $\mathbf{N}_k$ occur in each row of the operation matrix. Consequently, no matter what is the left operand a (index of the row of the operation matrix), there is always exactly one column (right operand b) containing any preassigned value of the operation c (element of row a), which proves statement (a) of the theorem.

However, there exists a set of operations corresponding to the permutations of the elements of the set $\mathbf{N}_k$ among which an operation is selected at step (2) of Procedure 6 such that it has smaller values for more frequently appearing pairs of operands and greater values for more rare ones. Consequently, the vector composed of the numbers of entries of the elements $\mathbf{N}_k$ in the resulting vector is monotone nonincreasing, which proves statement (b). ♦

Proof of Theorem 7. Let the significances of all operations in (15) be equal to k. Then, the length of the function vector represented by the formula Ψ is k^n, and by the formulas Ψ_i ($i = \overline{1,\ t}$), k^{n-1}. We calculate the number of functions N_t of n variables expressed by formula (15) for various t.

For $t = 1$, the formula has the form $\Psi \to (\mathrm{X}_1 \otimes_1 \Psi_1)$, the operation $\otimes_1$ consists of k rows of length k, and the matrix of the function Ψ, of k^{n-1} rows of length k. Then the number of functions expressed by formulas like $(\mathrm{X}_1 \otimes_1 \Psi_1)$ is equal to the number of the variants of compositions of the matrix Ψ of k rows belonging to the operation matrix $\otimes_1$ or

$$k^{k^{n-1}} - N_0, \tag{A.1}$$

where $N_0 = k + n(k^k - k)$ is the initial condition or the number of functions expressed by simple formulas like Δ, X, and $\Xi\,$X.[51]

For $t = 2$, the formula is given by $\Psi \to ((\mathrm{X}_1 \otimes_1 \Psi_1) \oplus_1 (\mathrm{X}_2 \otimes_2 \Psi_2))$, and together with the operation $\oplus_1$ the operations $\otimes_1$ and $\otimes_2$ generate k^2 different rows from which one can compile the matrix of the function Ψ. Then, the number of functions expressed by such formulas is as follows:

$$(k^2)^{k^{n-1}} - N_1,$$

with the exception of the functions N_1 expressed by the formulas like $(\mathrm{X}_1 \otimes_1 \Psi_1)$.

By continuing in this way, we obtain for any $t > 1$ that

$$N_t = k^{tk^{n-1}} - N_{t-1}.$$

[51] The number of simple formulas may be established as follows. There are exactly k (the number of constants) formulas Δ. The formula X is a special case of the formulas ΞX whose number in each variable is $k^k - k$ (the number of nonconstant vectors of length k).

The maximal number of identification steps $t_{\max}$ is determined from the condition of expression by a formula like (15) of all complex functions whose number is $k^{k^n} - N_0$:

$$\sum_{t=1}^{t_{\max}} N_t = k^{k^n} - N_0,$$

whence we obtain $t_{\max} k^{n-1} \leq k^n$ and $t_{\max} \leq k$.

Convergence of the procedure of analytical identification follows from the obtained estimate $t_{\max}$. However, the permutations of variables in Procedure 5 were disregarded at derivation of this estimate.

To allow for the impact of the permutations of variables, we consider a formula like $\Psi \to (\mathrm{X}_1 \otimes_1 \Psi_1)$ and seek the number of functions expressed by it in the following form:

$$\tilde{k}^{k^{n-1}} - N_0, \tag{A.2}$$

where $\tilde{k}$ is some expression, $\tilde{k} > k$.[52] It follows from the description of Procedure 5 that each shift of variables takes out by turn one variable from the set of variables of the formula Ψ_1 and substitutes it by another variable. Then, the effect obtained by shifting the variables is representable as a fictitious dependency of the formula Ψ_1 on $n-2$, and not on $n-1$, variables with retention of the length of the vector of the function which it expresses, that is, $\tilde{k}^{n-2} = k^{n-1}$. As the result:

$$\tilde{k} = \sqrt[n-2]{k^{n-1}} = k^{\frac{n-1}{n-2}}, \quad N_1 = k^{\frac{n-1}{n-2} k^{n-1}} - N_0.$$

By repeating the same reasoning for the rest of the identification steps, we get the desired formula:

$$t_{\max} = \frac{n-2}{n-1} k. \quad \blacklozenge$$

Proof of Theorem 8. The next recurrent equation follows from the structure of formula (15) and expression (16):

$$L_n(k) = \frac{n-2}{n-1} k \left(L_{n-1}(k) + 2 \right) - 1.^{53} \tag{A.3}$$

As can be seen from (15), at each step of identification two binary operations and the subformula Ψ_i of length $L_{n-1}(k)$ are determined. Since there are at most $t_{\max}$ such steps, $L_n(k)$ is equal to $t_{\max}(L_{n-1}(k) + 2) - 1$, where consideration is given to the operations $\otimes$ and $\oplus$ calculated at each step, except for one operation $\oplus$ which is not calculated at the last step. Substitution for $t_{\max}$ of its value from (16) provides Eq. (A.3).

From the evident initial condition $L_2 = 1$ from (A.3) we establish that

$$L_n(k) = \frac{1}{k(n-1)} \left(2k^{n-1} + \sum_{i=2}^{n-1} (i-1) k^{n-i} \right) - 1,$$

whence the desired estimate (17) for $L_n(k)$ is derived directly. $\blacklozenge$

[52] It may turn out that the number of complex functions expressed by the formula Ψ_1 cannot exceed (A.1). However, one has to take into consideration that the permutations of variables enable one to generate a set of different vectors for the same function, which is expressed as a matter of fact by formula (A.2).

[53] If a fractional number of operations is obtained as the result of calculating formula (A.3), it should be interpreted as the mean number of operations required to identify complex functions.

REFERENCES

1. Myshkis, A.D., *Elementy teorii matematicheskikh modelei* (Elements of the Theory of Mathematical Models), Moscow: Komkniga, 2007.

2. Samarskii, A.A. and Mikhailov, A.P., *Matematicheskoe modelirovanie. Idei. Metody. Primery* (Mathematical Modeling. Ideas. Methods. Examples), Moscow: Fizmatlit, 2001.

3. Karabutov, N.N., *Strukturnaya identifikatsiya sistem: analiz informatsionnykh struktur* (Structural System Identification: Analysis of the Information Structures), Moscow: MGIU, 2008.

4. Perel'man, I.I., Methodology of Selecting the Model Structure for Plant Identification, *Autom. Remote Control*, 1983, vol. 44, no. 11, part 1, pp. 1389–1407.

5. *Spravochnik po teorii avtomaticheskogo upravleniya* (Handbook of the Theory of Automatic Control), Krasovskii, A.A., Ed., Moscow: Nauka, 1987.

6. Graupe, D., *Identification of Systems*, New York: Van Nostrand, 1972. Translated under the title *Metody identifikatsii sistem*, Moscow: Mir, 1979.

7. Vapnik, V.N., *Vosstanovlenie zavisimostei po empiricheskim dannym* (Restoration of Dependencies from Empirical Data), Moscow: Nauka, 1979.

8. Zagoruiko, N.G., *Prikladnye metody analiza dannykh i znanii* (Applied Methods of Analysis of Data and Knowledge), Novosibirsk: Inst. Mat., 1999.

9. Katkovnik, V.Ya., *Neparametricheskaya identifikatsiya i sglazhivanie dannykh. Metod lokal'noi approksimatsii* (Nonparametric Identification and Data Smoothing. Method of Local Approximation), Moscow: Nauka, 1985.

10. Kurzhanskii, A.B., The Identification Problem—Theory of Guaranteed Estimates, *Autom. Remote Control*, 1991, vol. 52, no. 4, part 1, pp. 447–465.

11. Ikonen, E. and Najim, K., *Advanced Process Identification and Control*, New York: Marcel Dekker, 2002.

12. Sage, A.P. and Melsa, J.L., *System Identification*, New York: Academic, 1971. Translated under the title *Identifikatsiya sistem upravleniya*, Moscow: Nauka, 1974.

13. Heuberger, P., Van den Hof, P., and Wahlberg, B., *Modelling and Identification with Rational Orthogonal Basis Functions*, London: Springer, 2005.

14. Greblicky, W. and Pawlak, M., *Nonparametric System Identification*, Cambridge: Cambridge Univ. Press, 2008.

15. Rastrigin, L.A. and Madzharov, N.E., *Vvedenie v identifikatsiyu ob"ektov upravleniya* (Introduction to Identification of the Control Plants), Moscow: Energiya, 1977.

16. Furasov, V.D., *Zadachi garantirovannoi identifikatsii* (Problems of Guaranteed Identification), Moscow: BINOM, Lab. Znanii, 2005.

17. Cassandras, C.G. and Lafortune, S., *Introduction to Discrete Event Systems*, New York: Springer, 2008.

18. Masami Hagiya, Discrete State Transition Systems on Continuous Space-Time: A Theoretical Model for Amorphous Computing, in *Proc. 7 Int. Conf. "Unconventional Computation" (UC 2008)*, Vienna, 2008, pp. 117–129.

19. Kuzovkov, N.T., Karabanov, S.V., and Salychev, O.S., *Nepreryvnye i diskretnye sistemy upravleniya i metody identifikatsii* (Continuous and Discrete Control Systems and Identification Methods), Moscow: Mashinostroenie, 1978.

20. Garnier, H. and Wang, L., *Identification of Continuous-time Models from Sampled Data*, London: Springer, 2008.

21. Yablonskii, S.V., *Vvedenie v diskretnuyu matematiku* (Introduction to Discrete Mathematics), Moscow: Nauka, 1986.

22. Bibilo, P.N., Decomposition of Boolean Functions (Review), in *Proektirovanie ustroistv logicheskogo upravleniya* (Design of Logic Control Devices), Moscow: Nauka, 1985, pp. 106–126.

23. Yablonskii, S.V., Functional Constructions in the k-valued Logic, *Tr. Mat. Inst.* 1958, vol. 51, pp. 5–142.

24. Yablonskii, S.V., On Algorithmic Difficulties of Designing the Minimal Contact Circuits, *Probl. Kibern.*, Moscow: Fizmatgiz, 1959, no. 2, pp. 75–122.

25. Vykhovanets, V.S., Spectral Methods in Logical Data Analysis, *Autom. Remote Control*, 2001, vol. 62, no. 10, pp. 1565–1587.

26. Vykhovanets, V.S., Polunomial Factorization of Spectral Bases, *Autom. Remote Control*, 2004, vol. 65, no. 12, pp. 1861–1871.

27. Kuratowski, K. and Mostowski, A., *Set Theory*, Amsterdam: North-Holland, 1967. Translated under the title *Teoriya mnozhestv*, Moscow: Mir, 1970.

28. Belousov, A.I. and Tkachev, S.B., *Diskretnaya matematika* (Discrete Mathematics), Moscow: MGTU, 2004.

29. Vykhovanets, V.S., Algebraic Decomposition of Discrete Functions, *Autom. Remote Control*, 2006, vol. 67, no. 3, pp. 361–392.

30. Pentus, A.E. and Pentus, M.R., *Teoriya formal'nykh yazykov* (Theory of Formal Languages), Moscow: Mosk. Gos. Univ., 2004.

31. Reyward-Smith, V.J., *A First Course in Formal Language Theory*, London: Blackwell, 1983. Translated under the title *Teoriya formal'nykh yazykov. Vvodnyi kurs*, Moscow: Radio i Svyaz', 1988.

32. Trakhtenbrot, B.A. and Bardzin', Ya.M., *Konechnye avtomaty (povedenie i sintez)* (Finite Automata (Behavior and Design)), Moscow: Nauka, 1970.

33. *Selecta Mathematica II*, Jacobs, K., Ed., New York: Springer, 1970. Translated under the title *Mashiny T'yuringa i rekursivnye funktsii*, Ebbinghauz, G.-D., Yakobs, K., Man, F.-K., and Hermes, H., Eds., Moscow: Mir, 1972.

34. *Handbook of Mathematical Logic*, Barwise, J., Ed., Amsterdam: Horth-Holland, 1977. Translated under the title *Vvedenie v logiku pervogo poryadka. Spravochnaya kniga po matematicheskoi logike*, Moscow: Nauka, 1982.

35. Edel'man, S.L., *Matematicheskaya logika* (Mathematical Logic), Moscow: Nauka, 1975.

36. Even, S., Kohavi, I., and Paz, A., On Minimal Modulo 2 Sums of Products for Switching Functions, *IEEE Trans. Elect. Comput.*, 1967, pp. 671–674.

37. Kirichenko, K.D., Upper Estimate of Complexity of the Polynomial Normal Forms of Boolean Functions, *Diskret. Mat.*, 2005, vol. 17, no. 3, pp. 80–88.

38. Lupanov, O.B., On Designing Some Classes of the Control Systems, *Probl. Kibern.*, 1963, no. 10, pp. 63–97.

39. Moore, E.E., *Sequential Machines*, Reading: Addison-Wesley, 1964. Translated under the title *Umozritel'nyi experiment s posledovatel'nostnymi mashinami. Avtomaty*, Moscow: Inostrannaya Literatura, 1966.

This paper was recommended for publication by P.Yu. Chebotarev, a member of the Editorial Board