fractal and fractional

MDPI

Article

Finiteness of One-Valued Function Classes in Many-Valued Logic

Elmira Yu. Kalimulina

Institute for Information Transmission Problems of the Russian Academy of Sciences (Kharkevich Institute), 127051 Moscow, Russia; eyk@iitp.ru

Abstract: This paper addresses the theoretical issues in k-valued logic, which are crucial for developing solutions in various fields of science and technology. One of the fundamental issues is a complete description of the closed classes of functions of three-valued logic. The explicit description of closed classes in multivalued logic is an open problem. In this study, we consider a special case of the finite generation of all closed classes of three-valued logic through the operation of superposition. Previously, we considered the issue of the finite generation of classes containing a subset of single-variable functions. We have also provided a description of superlattices (lattices of lattices) containing a precomplete class of unary functions. The finite generation of these superlattices is proved. On the basis of these results, in this paper, we have proven that any class containing any of the precomplete classes from the set of single-valued functions is also finitely generated. The main result of this paper consists of three theorems on the finite generation of classes containing precomplete classes of single-valued functions and classes including all monotone unary functions. Thus, the obtained theoretical result provides easily verifiable criteria for the finiteness of classes of multivalued logic functions. It allows you to use simple procedures instead of cumbersome explicit constructs. The finite generation of overlattices allows the development of digital computing circuits that are crucial for practical applications. The proofs are based on an explicit description of these classes by an induction in the number of variables and essentially use the properties of functionally closed (Burle) classes of functions.

Keywords: k-valued logic; finiteness in many-valued logic; closed classes; three-valued logic functions; superposition operation; precomplete classes; unary functions; majority function; monotone unary functions

check for updates

Citation: Kalimulina, E.Y. Finiteness of One-Valued Function Classes in Many-Valued Logic. *Fractal Fract.* **2024**, *8*, 29. https://doi.org/10.3390/fractalfract8010029

Academic Editor: Carlo Cattani

Received: 15 November 2023
Revised: 26 December 2023
Accepted: 27 December 2023
Published: 29 December 2023

1. Introduction

In recent times, the significance of computation schemes based on k-valued logic, particularly ternary logic, has become relevant again. The rapid growth in supercomputing and quantum computing and their extensive usage in the field of artificial intelligence has resulted in a resurgence in interest in k-valued logic [1]. In particular, the research and development of algorithms based on k-valued logic are important for machine learning and neural networks [2,3]. In particular, machine learning and neural network algorithms based on k-valued logic significantly reduce the computational time required to solve problems compared with algorithms based on binary logic [2,3]. In addition, k-valued logic is highly relevant in the modeling of complex dynamical systems, where causal relations play a crucial role [4]. It is also used in logical dynamical systems [5], the modern control and stabilization of unmanned systems [6], cyber-physical systems [7], data transmission [8,9], optimal network routing [10,11], and quantum cryptography [12]. A comprehensive analysis of the use of k-valued logic in various applications has been provided in the research paper [9]. Among the wide range of applications, data aggregation schemes stand out because the development of three-valued logic algorithms has proven to be drastically more efficient and cost-effective than two-valued logic [13,14]. In particular, it is effective in providing a stable Internet connection for users in high-speed trains outside areas with stable cellular network coverage. By using a nonbinary number system to encode, for example, the states of an LTE modem, we can significantly reduce the complexity and number of internal computations. The transition from two-valued logic to multivalued logic reduces the number of computational operations and increases the speed of data processing. Fuzzy sets in this case are approximated by finite discrete sets. The general state of the system is determined by the complex predicate $Y(\cdot)$ as a superposition of other predicates on the original set $\{P_k\}$. The complete construction of the predicate and a detailed description of traffic aggregation are given in [14].

We also note several theoretical papers devoted to topics connecting *k*-valued logic with algebraic problems and matrix group/semigroup representation. In [15], using the semitensor product form of matrices, the authors suggested expressing *k*-valued logical function into its algebraic form, which establishes an equivalent relation between unary logical operators and their structure matrices. In [16], conjunctive and disjunctive normal forms of *k*-valued logic have been introduced, and the completeness of *k*-valued logic by constructing an adequate set of connectives is discussed, which was then extremely compressed by constructing a proper set of generators. The paper in [17] presents a construction for the classification of subalgebras for all algebras of finite-valued functions. In addition, it is shown that the classes of this classification are disjoint.

Another interesting area of theoretical research explores the interaction of many-valued logic, Markov processes with jumps, and the stability of dynamical systems. In the scientific paper in [18], the authors provided a comprehensive study of the stability aspects of *k*-valued logic networks, considering the influence of Markov jumps. To facilitate the analysis, they introduce an equivalent Markov jump system using a semitensor product of matrices combined with multivalued logic functions. As part of their study, the authors proposed two deterministic criteria for assessing the finite-time stability of distributions of *k*-valued logic networks subject to Markov processes with jumps. The study in [19] investigates the optimal control problem of switched *k*-valued logic control networks with state-dependent switching signals. It provides a mathematical representation of the switched *k*-valued logic control networks and presents a condition that is both necessary and sufficient for their stabilization.

A complete description of all closed classes of functions of *k*-valued logic is the most important problem in the application of *k*-valued logic [20]. The finite generation of all closed classes ensures that digital circuits can be implemented with the desired function and algorithms [21].

Emil Post presented a complete description of closed classes of Boolean functions concerning superposition in his famous papers [22,23]. This result allowed us to solve numerous problems in the field of binary logic and in the computer industry. It was also demonstrated that all classes of closed binary logic are finitely generated with respect to superposition operations. However, a detailed description of *k*-valued logic is impossible because the transition to *k*-valued logic leads to the emergence of a continuum of closed classes with respect to superposition operations.

In fact, in the case of *k*-valued logic, no finitely generated classes exist, as evidenced by the example of Yanov and Muchnik [24]. Therefore, the problem of describing all finitely generated classes of *k*-valued logic remains open.

The completeness problem has a solution if we consider some special operators. To describe a sublattice of closed classes, we can start with the lattice of all functions and define subsets that are closed, for example, under the superposition operator. The superposition operator combines two functions to form a new function by applying each function in the pair to the same input data and then combining the results. The description of a sublattice of closed classes may include identifying the various closed classes and defining the closure properties of their elements under the action of the superposition operator. This can be achieved by studying the properties of the functions of each class and their composition.

The finite generation of all closed classes of two-valued logic with respect to the superposition operation is a special case considered in [25]. In [26], the operation of binary superposition is considered and defined by functions of k-valued logic based on their representation in the binary number system. The authors of [27] investigated the requirement of implicit completeness in three-valued logic from the perspective of precomplete classes.

The sublattice of closed classes itself forms a lattice structure, the elements of which are closed classes, and the ordering relation is based on inclusion. Each closed class in this lattice is a set of functions that can be composed of other functions of the same class to create new functions belonging to the same class. Several sufficient conditions for finite generation are known. The most famous examples include all unary functions, choice functions, and majority functions in the class. The majority function takes a set of inputs and returns true (or 1) if most inputs are true (1) and false (or 0) otherwise. The choice function takes only one element from a nonempty set. Unary functions operate on a single input. The existence of majority, choice, and unary functions is important not only in terms of class finiteness in many-valued logic theory but also in computational complexity theory, formal languages, automata theory, algorithm design, voting systems, and data processing. Thus, although this paper is rather theoretical, all the results obtained here are significant in their respective fields of study and have been widely researched and studied in the field of computer science. In [9,28], a description of superlattices containing some precomplete class of unary functions is also given, and the problem of checking the finite generation of classes containing some subclass of a function of one variable is studied. It has been proven that overlattices can be finitely generated. Any class consisting of monotonic functions and containing each monotonic function of one variable is finitely generated. These classes do not include a choice function, a majority function, all single-valued functions, or a set of all single-valued functions precompleted for all single-valued functions. This paper continues the study of the theoretical problems of k-valued (in particular, ternary) logic [9,28]. As mentioned above, in two-valued logic, there are several sufficient conditions for finitely generated classes [29]. Similar conditions can be found for multivalued logic. If they are satisfied, then the class is finitely generated. From a practical point of view, these conditions mean that we do not need to provide a complete description of a class to test its finiteness. However, it is sufficient to check the conditions for some special functions. These conditions must be such that they can be easily verified. One of these conditions is the presence in the class (multivalued logic functions) of one-valued functions. In this paper, based on the previously proven results [9,25,28], we show that any class containing any of the precomplete classes of the set of unary functions is finitely generated. These conditions must be such that they can be easily verified. One of these conditions is the presence in the class (multivalued logic functions) of single-valued functions.

In this paper, based on the previously proven results of [9,25,28], we show that any class containing any of the prerecomplete classes of the set of unary functions is finitely generated.

1.1. Paper Structure

This paper consists of the following sections: Section 2, Section 3 (auxiliary results), Section 4 (main results), and the conclusion. Section 2 includes the necessary notations, the definition of k-valued logic functions, and preliminary results. This section provides the fundamental Burle's result, which describes all functionally closed classes of functions of k-valued logic that are called Burle's classes [29]. This theorem plays a vital role in k-valued logic theory. The proofs presented in this paper significantly use the following properties of Burle's classes: $P_k(1)$ is a precomplete class in $LQ_k \cup P_k(1)$, $LQ_k \cup P_k(1)$ is a precomplete class in $P_k^{(2)} \cup P_k(1)$, $P_k^{(l)} \cup P_k(1)$ is a precomplete class in $P_k^{(l+1)} \cup P_k(1)$ for $2 \leqslant l \leqslant k-2$, $P_k^{(k-1)} \cup P_k(1)$ is a precomplete class in P_k.

Relationship between Lemmas and Proof Scheme

The concept of the proof is based on the explicit construction of the corresponding classes of functions. Preliminary results prove the existence of closed classes of functions that are monotonic under linear order operations (functions that have the necessary properties) and give their explicit construction. Then, it is shown that the functions have an explicit representation in the form of elementary functions (a specific type of expansion is given) from the finite-generated classes. We consider the narrowing of this class of functions to a class of functions with exactly two variables and three values.

The main part shows that if a monotonic function of many variables takes three values, then there is a set composed of such functions, containing monotonic functions that essentially depend on exactly two variables and take three values. These are exactly the functions discussed in the section with preliminary results, and they have the necessary properties.

There are many technical lemmas that provide intermediate steps in our proofs.

The first auxiliary result is given by Lemmas 1 and 2. We consider linear order and monotonicity in relation to the two operations of maximum and minimum (symbols). These operations are defined below, and the corresponding subsets of the function are denoted by the letters D and K. Accordingly, the classes of monotone functions considered with respect to these operations are denoted as f^K and f^D. In the intermediate results, the lemmas regarding these two operations are proved. They will be considered separately, but the evidence is the same; therefore, it will be given only for one of the classes. Lemma 1 and Lemma 2 provide a general description of the set of all monotone functions with respect to linear order operation (not only for one-valued functions). These lemmas only establish that a closed class is exactly the join of all monotone functions from the finitely generated sets. Lemma 1 and Lemma 2 are lemmas about the existence of closed classes for monotone functions. They guarantee some property for a certain class of function but do not provide the explicit construction of such a class.

Lemma 3 explicitly describes the system that generates such a class and establishes that this class exists and is not empty. This class is generated by applying the expansion (1) to all subfunctions from Lemma 3. Note that these results apply to all monotone functions and define the widest possible class of functions, which in our case is redundant because the class of arbitrary functions may not be finite. Therefore, we further prove Lemma 4, which specifically describes the class of functions of interest, i.e., monotone functions for three-valued logic. Lemma 5 proves that the classes of monotone functions with respect to the operations of maximum and minimum coincide and constitute the closed finite class of monotone functions of interest, i.e., M. Lemmas 4 and 5 directly describe the class of monotone functions consisting of only two- and one-variable functions.

Lemma 6 is based on Lemmas 4 and 5 and gives the explicit representation of monotone functions exactly in the form of functions from $M^{(2)}$ (the set of all monotone functions that take at most two values) to the set $M_3(1)$ (one-place functions of 3-valued logic). Thus, the overall result is narrowed to a set of specific functions with s representation in the form of matrices. It establishes that any monotone function consisting of only two and one variables can be composed of other functions within the same class to produce new functions within the same class. Any class consisting of monotonic functions and containing each monotonic function of one variable is finitely generated.

Thus, taken together, Lemmas 1–6 provide an explicit description (construction) of the class of functions that are monotonic with respect to linear order. These results guarantee not only the existence of such a class but also provide an explicit design for its construction.

It is shown that for any monotonic function from this class that satisfies the monotonicity condition (condition (2), the function depends significantly on more than one variable. In Lemma 6, it is already shown that these two classes exhaust the entire set of all monotone functions that take no more than two values.

In the main part of the paper, Lemma 7 establishes the property of an arbitrary monotone function from class P_3: if linear-order conditions are satisfied for a monotone function, then the function depends significantly on more than one variable. Lemmas 8 and 9 establish that for such functions, it is possible to construct an identical function that will depend on $(n-1)$ variables and will have the same properties. This is a general result for functions of many variables.

Lemmas 10 and 11 are based on Lemmas 7–9. They restrict the result for functions of many variables to the case of functions that essentially depend on two variables. Thus, any monotonic function can be associated with a function of two variables and is contained in a finitely generated class (Lemma 12).

Thus, Lemmas 7–12 from the main results section establish that the functions of 3-valued logic (the description of which is completely determined by Lemmas 1–6) generate sets of functions that satisfy the basic properties defining a closed class of monotone functions. Thus, we obtain a comparison of the class of monotonic functions with a certain class of functions $\{g\}$, which are specified explicitly and have the necessary properties.

Theorem 2 proves the finite generation of classes containing all monotone unary functions. In Theorem 3, we proved that the constructed set of monotone functions essentially depends on exactly two variables (but it is a 3-valued logic function).

As a result, we established that the set of monotone functions $f(x_1, \ldots, x_n), n > 1$, takes three values if it is combined with the class $M_3(1)$ and contains a monotone function that essentially depends on exactly two variables and takes three values. In other words, F contains exactly the class of functions and properties considered in the lemmas. On the basis of the properties that have been proved on these Lemmas, we can conclude that class F is finitely generated.

Theorem 4 provides the criteria for F to be a finite-generated class. In this way, we reduce the problem to checking criteria that must be satisfied.

2. Preliminary Results and Definitions on k-Valued Functions

Let us begin with some recall and definitions of k-valued logic functions. Let E_k be the set $\{0, 1, \ldots, k-1\}$. For any $n \in \mathbb{N}$ the set $E_k^n = \underbrace{E_k \times \ldots \times E_k}_{n}$ is a Cartesian power, and the mapping $f : E_k^n \to E_k$ is a single-valued function of k-valued logic. The set of all functions of k-valued logic is denoted by P_k. Furthermore, we assume that $k = 3$ in most cases, so the set P_3 will be considered.

Let F be a closed class of P_k, then $F(n)$ is the set of all functions in F that depend on the variables $x_1, \ldots, x_n$. $F^{(n)}$ is the set of all functions F that take at most n values. $CR(F)$ is the set of all precompleted classes in the closed class $F \subseteq P_k$. PS_k is the set of all unary functions that take exactly k values. A complete description of the prior completeness superlattice in the $P_k(1)$ class for $k = 3$ can be found in [9,28]. Just recall a few important definitions and one theorem.

Definition 1. *A function $f(x_1, \ldots, x_n)$ that takes two or fewer values is sublinear if $(1 \leqslant i \leqslant n)$ for any number of variables i, where $(1 \leqslant i \leqslant n)$, and for any two elements $\alpha, \beta \in E_k$, one of the following relationships holds:*

- *either for any $\gamma_1, \ldots, \gamma_{i-1}, \gamma_{i+1}, \ldots, \gamma_n \in E_k$*

$$f(\gamma_1, \ldots, \gamma_{i-1}, \alpha, \gamma_{i+1}, \ldots, \gamma_n) = f(\gamma_1, \ldots, \gamma_{i-1}, \beta, \gamma_{i+1}, \ldots, \gamma_n),$$

- *or for any $\gamma_1, \ldots, \gamma_{i-1}, \gamma_{i+1}, \ldots, \gamma_n \in E_k$*

$$f(\gamma_1, \ldots, \gamma_{i-1}, \alpha, \gamma_{i+1}, \ldots, \gamma_n) \neq f(\gamma_1, \ldots, \gamma_{i-1}, \beta, \gamma_{i+1}, \ldots, \gamma_n).$$

The set of all quasi-linear functions in P_k is denoted by LQ_k. The problem of enumerating all closed classes of k-valued logic containing all functions of one variable was solved by G.A. Burle [29].

Theorem 1 (see [29])**.** *The following and only these classes, which contain all k-valued logic functions of one variable, are functionally closed classes: $P_k(1)$, $LQ_k \cup P_k(1)$, $P_k^{(2)} \cup P_k(1), \ldots, P_k^{(k-1)} \cup P_k(1)$, P_k.*

The k-valued classes $P_k(1)$, $LQ_k \cup P_k(1)$, $P_k^{(2)} \cup P_k(1), \ldots, P_k^{(k-1)} \cup P_k(1)$, P_k are Burle's classes. An overlattice of a class G is the set of all classes $F \subseteq P_k$ such that $G \subseteq F$.

3. Auxiliary Theorems and Proofs of Technical Lemmas

3.1. Finite Generation of Classes Containing Precomplete Classes of One-Valued Functions

Theorem 2. *Let $F \subseteq P_3$, $G \in CR(P_3(1))$, $G \subseteq F$. Then, class F is finitely generated.*

Proof. Let us consider an overlattice of a class G as the set of all classes $F \subseteq P_k$ such that $G \subseteq F$. The overlattice description for all $G \in CR(P_3(1))$ (accordingly, $F \in P_3(1)$) is given in [30]. Therefore, it is sufficient to prove the finite generation of the already described classes. If $G = L_3(1)$, then the overlattice G consists of the following closed classes: $L_3(1)$, L_3, $P_3(1)$, $LQ_3 \cup P_3(1)$, $P_3^{(2)} \cup P_3(1)$, P_3.

Let V be a precomplete class in PS_3. If $G = V \cup P_3^{(2)}(1)$, then its overlattice consists of the classes $V \cup P_3^{(2)}(1)$, $P_3(1)$, $V \cup P_3^{(2)}(1) \cup LQ_3$, $P_3(1) \cup LQ_3$, $V \cup P_3^{(2)}(1) \cup P_3^{(2)}$, $P_3(1) \cup P_3^{(2)}$, P_3.

Because for all precomplete classes in PS_3, the proof is the same as in [28], we set the class V to be fixed. Then, it is sufficient to prove that the following nine classes are finitely generated: $L_3(1)$, $P_3(1)$, $V \cup P_3^{(2)}(1)$, $LQ_3 \cup P_3(1)$, $P_3^{(2)} \cup P_3(1)$, P_3, L_3, $V \cup P_3^{(2)}(1) \cup LQ_3$, $V \cup P_3^{(2)}(1) \cup P_3^{(2)}$.

For the convenience of the proof, we split them into the following groups:

1. $L_3(1)$, $P_3(1)$, $V \cup P_3^{(2)}(1)$ consist of a finite number of functions.
2. $LQ_3 \cup P_3(1)$, $P_3^{(2)} \cup P_3(1)$, P_3 are Burle's classes (see [29]), where a finite generation has already been proved.
3. $L_3 = \{\{x + y(\mathrm{mod} 3), 1\}\}$.
4. $V \cup P_3^{(2)}(1) \cup LQ_3$. Burle [29] proved that for any function $f \in LQ_3$, the following function representation holds:

$$f(x_1, \ldots, x_n) = \varphi(\{\varphi_1(x_1) + \ldots + \varphi_n(x_n)\}(\mathrm{mod} 2)),$$

   where $\varphi, \varphi_1, \ldots, \varphi_n$ are functions from $P_3(1)$. Using this representation, Burle showed that $P_3(1)$ is precomplete in $LQ_3 \cup P_3(1)$. Because this representation includes addition by *mod* 2, it is easy to see that only functions from $P_3^{(2)}(1)$ are used in this expansion. Therefore, $V \cup P_3^{(2)}(1)$ is precomplete in $V \cup P_3^{(2)}(1) \cup LQ_3$. This proves that class $V \cup P_3^{(2)}(1) \cup LQ_3$ is finitely generated.
5. $V \cup P_3^{(2)}(1) \cup P_3^{(2)}$.

Burle's proof of precompleteness for $LQ_3 \cup P_3(1) \in P_3^{(2)} \cup P_3(1)$ [29] is completely the same for this case, because it is completely based only on the unary functions that take no more than two values.

Therefore, $V \cup P_3^{(2)}(1) \cup LQ_3$ is precompleted in $V \cup P_3^{(2)}(1) \cup P_3^{(2)}$. This proves the finite generation of the class $V \cup P_3^{(2)}(1) \cup P_3^{(2)}$, and the finite generation of $V \cup P_3^{(2)}(1) \cup LQ_3$ already been proved. □

3.2. Finite Generation of Classes Containing All Monotone Unary Functions

We already proved the finite generation of classes containing all unary functions or some precomplete class of unary functions.

A similar problem may be developed for classes of one-place functions satisfying the monotonic condition with respect to linear order.

Definitions and Auxiliary Results

For simplicity, let $x_1 \vee x_2 \vee \ldots \vee x_n$ denote the maximum, and let $x_1 x_2 \ldots x_n$ and $x_1 \& x_2 \ldots \& x_n$ denote the minimum of the set of variables $x_1, x_2, \ldots, x_n$.

Let $\circ \in \{\vee, \&\}$. Then, for any subset of $\tilde{\alpha}, \tilde{\beta} \in E_3^n$, the notation $\tilde{\alpha} \circ \tilde{\beta}$ means $\tilde{\gamma} = (\alpha_1 \circ \beta_1, \ldots, \alpha_n \circ \beta_n)$, where for any $i = 0, 1, \ldots, n$ $\alpha_i \circ \beta_i$ is the maximum of the set α_i, β_i, if $\circ = \vee$, then – $\alpha_i \circ \beta_i$ is the minimum from α_i, β_i.

We define the following sets of monotone functions of 3-valued logic:

- D is the set of all monotone functions such that for any collections $\tilde{\alpha}_1, \tilde{\alpha}_2 \in E_3^n$, the following inequality holds $f(\tilde{\alpha}_1 \vee \tilde{\alpha}_2) \leqslant f(\tilde{\alpha}_1) \vee f(\tilde{\alpha}_2)$,
- K is the set of all monotone functions such that for any collections $\tilde{\alpha}_1, \tilde{\alpha}_2 \in E_3^n$, the following inequality holds $f(\tilde{\alpha}_1 \& \tilde{\alpha}_2) \leqslant f(\tilde{\alpha}_1) \& f(\tilde{\alpha}_2)$,
- $M^{(2)}$ is the set of all monotone functions that take at most two values and functions from the set $M_3(1)$,

where M is the set of all monotone functions with respect to the linear order $(0 < 1 < 2)$ from P_3, and $M_3(1)$ is the set of all monotone functions that essentially depend on at most one variable.

Furthermore, we will be interested in the finite generation of closed classes $F \subset P_3$ that satisfy the following conditions: $M_3(1) \subseteq F$, $F \subseteq M$.

Lemmas 1 and 2 provide intermediate auxiliary results. They describe the set of all monotone functions with respect to the linear-order operation (not only one-valued).

Lemma 1. *The set D is a closed class and exactly is the join of all monotone functions from the finitely generated class F (of monotone functions of one variable) with the maximum of a binary set $\{x, y\}$: $D = \big\{\{x \vee y\} \cup M_3(1)\big\}$.*

Proof. First, let us show that D is a closed class. Since class D contains the function x, it is sufficient to show that the function $g = f_0(f_1, \ldots, f_q)$ belongs to D if $f_0, f_1, \ldots, f_q \in D$. Let $g(x_1, \ldots, x_n) = f_0(f_1(x_1, \ldots, x_n), \ldots, f_q(x_1, \ldots, x_n))$. It is clear that $g \in M$. Consider two arbitrary sets $\tilde{\alpha}_1, \tilde{\alpha}_2 \in E_3^n$. Then, $g(\tilde{\alpha}_1 \vee \tilde{\alpha}_2) = f_0(f_1(\tilde{\alpha}_1 \vee \tilde{\alpha}_2), \ldots, f_q(\tilde{\alpha}_1 \vee \tilde{\alpha}_2)) \leqslant f_0(f_1(\tilde{\alpha}_1) \vee f_1(\tilde{\alpha}_2), \ldots, f_q(\tilde{\alpha}_1) \vee f_q(\tilde{\alpha}_2)) = f_0(\tilde{\beta}_1 \vee \tilde{\beta}_2) \leqslant f_0(\tilde{\beta}_1) \vee f_0(\tilde{\beta}_2) = g(\tilde{\alpha}_1) \vee g(\tilde{\alpha}_2)$, where $\tilde{\beta}_i = (f_1(\tilde{\alpha}_i), \ldots, f_q(\tilde{\alpha}_i))$, $i = 1, 2$.

Second, we show that $\big\{\{x \vee y\} \cup M_3(1)\big\} \subseteq D$. For any sets $\tilde{\alpha}, \tilde{\beta} \in E_3^2$ the following relation holds: $\max(\tilde{\alpha} \vee \tilde{\beta}) = \max\{\alpha_1 \vee \beta_1, \alpha_2 \vee \beta_2\} = \max\{\alpha_1, \alpha_2, \beta_1, \beta_2\} = \max\{\max(\tilde{\alpha}), \max(\tilde{\beta})\}$. Therefore, $\max(x, y) = x \vee y \in D$. Let $f(x)$ be an arbitrary function from $M_3(1)$. Considering two cases $a \leq b$ and $a \geqslant b$, we obtain $f(a \vee b) = f(a) \vee f(b)$, and, therefore, $M_3(1) \subseteq D$. Because D is a closed class, we obtain superpositions of functions from $M_3(1)$, and functions $x \vee y$ are contained in D.

Third, we must show that $D \subseteq \big\{\{x \vee y\} \cup M_3(1)\big\}$. For a collection $\tilde{\alpha}$, we denote $\tilde{\alpha}_i^0$ as the collection that coincides with the collection $\tilde{\alpha}$ on all components, except, possibly, the ith, which is equal to zero. Let $f(x_1, \ldots, x_n)$, $n > 1$ be an arbitrary function from D. Since f is monotone (by the Definition of the class D), then $f(\ ta) \geqslant f(\ ta_i^0)$ for any $i = 1, \ldots, n$, and hence, $f(\tilde{\alpha}) \geqslant f(\ ta_1^0) \vee \ldots \vee f(\ ta_n^0)$. At the same time, since $\tilde{\alpha} = \max\limits_{i=1,\ldots,n} \{\tilde{\alpha}_i^0\}$, then by virtue of the property defining the class D, we have $f(\tilde{\alpha}) \leqslant f(\tilde{\alpha}_1^0) \vee \ldots \vee f(\tilde{\alpha}_n^0)$. Therefore, the decomposition $f(x_1, \ldots, x_n) = f(0, x_2, \ldots, x_n) \vee f(x_1, 0, \ldots, x_n) \vee \ldots \vee f(x_1, x_2, \ldots, 0)$ holds for any function $f(x_1, \ldots, x_n) \in D$. Therefore, by applying this decomposition to all subfunctions, we obtain $f \in \big\{\{x \vee y\} \cup M_3(1)\big\}$ for any function $f \in D$. Therefore, $D = \big\{\{x \vee y\} \cup M_3(1)\big\}$. □

The next statement is similar to the one proved above, so its proof is omitted.

Lemma 2. *The set K is a closed class and exactly is the join of all monotone functions from the finitely generated class F (of monotone functions of one variable) with the minimum from the set of variables x, y: $K = \big\{\{x \& y\} \cup M_3(1)\big\}$.*

Lemmas 1 and 2 show that the set of all monotone functions is closed. They guarantee some property for a certain class of function. The next question is, does this class exist? The next Lemma 3 stands for a system that generates such a class.

Lemma 3. *(on the generating system of the set of monotone functions)*

$$\{\max(x,y), \min(x,y), J_0(x), J_1(x), \ldots, J_{k-1}(x), 0, \ldots, k-1\} = M. \tag{1}$$

Proof. It is obvious that $\max(x,y), \min(x,y), J_0(x), J_1(x), \ldots, J_{k-1}(x)$ are contained in M. Let $f \in M$. Therefore, $f(x_1, \ldots, x_n) = J_{k-1}(x_1)f(k-1, x_2, \ldots, x_n) \vee \ldots \vee J_1(x_1)f(1, x_2, \ldots, x_n) \vee f(0, x_2, \ldots, x_n)$. This equality can be verified directly by substituting the values of the variable x_1 and using the definition of monotonicity of the function f. □

In accordance with Lemmas 1 and 2, we can assert that a closed class exists, is not empty, and is generated by some system. By applying the expansion (1) to all subfunctions, we obtain the system generated from the set of monotone functions.

Remark 1. *The function f of two variables is represented as a matrix 3×3, where $a_{ij} = f(i,j)$, and the one-valued function g can be represented as a column vector of height three, where $a_i = f(a_i)$.*

Lemmas 4 and 5 directly describe the class of monotone functions consisting of only two and one variable functions. Thus, they narrow the general set to a set of concrete functions. Lemma 6 based on these results gives the representation of monotone functions exactly for 3-valued logic.

Lemma 4. $x\&y \in \{\{x \vee y\} \cup M_3(1) \cup \{T(x,y)\}\}$, *where*

$$T(x,y) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Proof. Let us define the functions

$$f_1(x,y) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad f_2(x,y) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

It is easy to see that the following equality holds: $x\&y = f_1(x,y) \vee f_2(x,y)$.

Moreover, it is clear that $f_1(x,y) = J_1(T(x,y))$, $f_2(x,y) = T(J_1(x), J_2(y))$. Hence, $x\&y \in \{\{x \vee y\} \cup M_3(1) \cup \{T(x,y)\}\}$. □

Lemma 5.

$$\{M^{(2)} \cup \{\max\}\} = \{M^{(2)} \cup \{\min\}\} = M.$$

Proof. Prove that $\{M^{(2)} \cup \{\max\}\} = \{M^{(2)} \cup \{\min\}\}$ holds Let us define the functions

$$f_1(x,y) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad f_2(x,y) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Then, it is obvious that equality $\min(x,y) = \max(f_1(x,y), f_2(x,y))$ holds.

Let us define the functions

$$f_3(x,y) = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 1 & 2 \\ 2 & 2 & 2 \end{pmatrix}, \quad f_4(x,y) = \begin{pmatrix} 0 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{pmatrix}$$

Then, it is easy to check the following equality: $\max(x,y) = \min(f_3(x,y), f_4(x,y))$.

Well, then $\max(x,y) \in \{M^{(2)} \cup \{\min(x,y)\}\}$, $\min(x,y) \in \{M^{(2)} \cup \{\max(x,y)\}\}$.

Then, by Lemma 3 $\{M^{(2)} \cup \{\max(x,y)\}\} = M = \{M^{(2)} \cup \{\min(x,y)\}\}$. □

Definition 2. *For any $n \geqslant 2$, the set of functions denoted by $\bigwedge_{a,b}^{n}$ (accordingly $\bigvee_{a,b}^{n}$), where $a \neq b$, $a, b \in \{0, 1, 2\}$, consisting of all functions depending on n variables, taking only the values a, b and on tuples consisting of a and b that coincide with $\min(x_1, \ldots, x_n)$ (accordingly with $\max(x_1, \ldots, x_n)$), is called the set of conjunctions of length n on a, b (accordingly, the set of conjunctions of length n to a, b).*

An arbitrary function from the set $\bigwedge_{a,b}^{n}$ (accordingly, $\bigvee_{a,b}^{n}$) is called a conjunction of length n by a, b (accordingly, a disjunction of length n by a, b).

Definition 3. *A set $\bigwedge_{a,b} = \bigcup\limits_{n\geqslant 2} \bigwedge_{a,b}^{n}$ (accordingly $\bigvee_{a,b} = \bigcup\limits_{n\geqslant 2} \bigvee_{a,b}^{n}$), where $a \neq b$, $a, b \in \{0, 1, 2\}$ will be called the set of all conjunctions on a, b (i.e., the set of all disjunctions on a, b).*

Note that, for any $n \geqslant 2$, by identifying variables from a function belonging to the set $\bigwedge_{a,b}^{n}$ (accordingly, $\bigvee_{a,b}^{n}$), we can obtain a function from the set $\bigwedge_{a,b}^{2}$ (accordingly, $\bigvee_{a,b}^{2}$), and using the superposition operation, we can make the reverse transition. Therefore, for brevity, we often omit the enumeration of variables on which functions from the sets $\bigwedge_{a,b}$ and $\bigvee_{a,b}$ depend.

Lemma 6. *Let $\vee_{0,2} \in \bigvee_{a,b}$, $\&_{0,2} \in \bigwedge_{a,b}$. Then $M^{(2)} = \big\{\{\vee_{0,2}\} \cup \{\&_{0,2}\} \cup M_3(1)\big\}$.*

Proof. For all monotone functions, the expansion holds

$$f(x_1, \ldots, x_n) = J_2(x_1) f(2, x_2, \ldots, x_n) \vee J_1(x_1) f(1, x_2, \ldots, x_n) \vee f(0, x_2, \ldots, x_n)$$

(see Lemma 3).

Note that for functions that take only the values zero and two, instead of $x \vee y$, you can use any function that matches $x \vee y$ on sets consisting of zeros and twos. Likewise, for xy.

Hence, for all functions taking values of zero and two, the following expansion holds:

$$f(x_1, \ldots, x_n) = J_2(x_1) \&_{0,2} f(2, x_2, \ldots, x_n) \vee_{0,2} J_1(x_1) \&_{0,2} f(1, x_2, \ldots, x_n) \vee_{0,2} f(0, x_2, \ldots, x_n).$$

Let $f(x_1, \ldots, x_n)$ take values of zero and one. Then $f(x_1, \ldots, x_n) = j_1(g(x_1, \ldots, x_n))$, where $g(x_1, \ldots, x_n)$ is obtained from $f(x_1, ..., x_n)$ by replacing value one with value two in the value table.

Because $g(x_1, \ldots, x_n)$ takes values of one and two, then, as proved above, we obtain $g \in \big\{\{\vee_{0,2}\} \cup \{\&_{0,2}\} \cup M_3(1)\big\}$; hence, $f(x_1, \ldots, x_n)$, taking values of zero and one, is contained in $\big\{\{\vee_{0,2}\} \cup \{\&_{0,2}\} \cup M_3(1)\big\}$.

The same scheme is true for functions that take values of one and two. Hence

$$M^{(2)} \subseteq \big\{\{\vee_{0,2}\} \cup \{\&_{0,2}\} \cup M_3(1)\big\}.$$

The inverse inclusion is obvious. □

Denote by f^D, f^K, $f^{M^{(2)}}$, $f^{DM^{(2)}}$, $f^{KM^{(2)}}$ monotone functions of 3-valued logic such that they do not belong to the sets D, K, $M^{(2)}$, $DM^{(2)} = D \cap M^{(2)}$, $KM^{(2)} = K \cap M^{(2)}$ accordingly. It follows from the definition of class D that if sets (combinations) $\tilde{\alpha}, \tilde{\beta} \in E_3^n$ for a function $f(x_1, \ldots, x_n)$, $n > 1$ exist such that the following condition holds:

$$f(\tilde{\alpha} \vee \tilde{\beta}) > f(\tilde{\alpha}) \vee f(\tilde{\beta}) \tag{2}$$

then a function $f(x_1, \ldots, x_n)$, $n > 1$ does not belong to the class D.

In this case, we say that the condition (2) holds on sets $\tilde{\alpha}, \tilde{\beta}$.

4. Main Results

Lemma 7. *Let $f(x_1,\ldots,x_n)$, $n \geqslant 1$ be an arbitrary monotone function from P_3 which satisfies condition (2). Then, $f(x_1,\ldots,x_n)$ significantly depends on more than one variable.*

Proof. (by contradiction). Let $f(x_1,\ldots,x_n) = j(x_k)$ for some x_k, $k = 1,\ldots,n$, and for some function $j(x) \in M_3(1)$.

Let us consider two arbitrary sets $\tilde{\alpha}, \tilde{\beta} \in E_3^n$. By our assumption, $f(\tilde{\alpha} \vee \tilde{\beta}) = j(\alpha_k \vee \beta_k)$.

Without loss of generality, we can assume that $\alpha_k > \beta_k$. Then, $f(\tilde{\alpha} \vee \tilde{\beta}) = j(\alpha_k \vee \beta_k) = j(\alpha_k) = f(\tilde{\alpha})$ and $f(\tilde{\alpha}) \vee f(\tilde{\beta}) = j(\alpha_k) \vee j(\beta_k) = j(\alpha_k) = f(\tilde{\alpha})$.

Hence, for any pair of sets $\tilde{\alpha}, \tilde{\beta} \in E_3^n$, the following equality holds:

$$f(\tilde{\alpha} \vee \tilde{\beta}) = f(\tilde{\alpha}) \vee f(\tilde{\beta}),$$

but this contradicts the condition (2). □

Lemma 8. *Let $f(x_1,\ldots,x_n)$, $n > 1$ be an arbitrary monotone function from P_3 which satisfies condition (2) on such subsets $\tilde{\alpha}, \tilde{\beta}$ from E_3^n that j $(1 \leqslant j \leqslant n)$, and a $(a \in E_3)$ exist, and $\alpha_j = \beta_j = a$. Then, function $g(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n) = f(x_1,\ldots,x_{j-1},x_i,x_{j+1},\ldots,x_n)$ satisfies condition (2).*

Proof. Set $\tilde{\gamma} = (\alpha_1,\ldots,\alpha_{j-1},\alpha_{j+1},\ldots,\alpha_n)$, $\tilde{\delta} = (\beta_1,\ldots,\beta_{j-1},\beta_{j+1},\ldots,\beta_n)$. Then, $g(\tilde{\gamma} \vee \tilde{\delta}) = f(\tilde{\alpha} \vee \tilde{\beta}) > f(\tilde{\alpha}) \vee f(\tilde{\beta}) = g(\tilde{\gamma}) \vee g(\tilde{\delta})$, that is, for the function g, the condition (2) holds on sets $\tilde{\gamma}, \tilde{\delta}$. □

The next statement is similar to the one proved above, so its proof is omitted.

Lemma 9. *Let $f(x_1,\ldots,x_n)$, $n > 1$ be an arbitrary monotone function from P_3 that satisfies condition (2) on such sets $\tilde{\alpha}, \tilde{\beta}$ from E_3^n that $1 \leqslant i, j \leqslant n$, $i < j$ exist and $\alpha_i = \alpha_j$, $\beta_i = \beta_j$. Then, a function $g(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n) = f(x_1,\ldots,x_{j-1},x_i,x_{j+1},\ldots,x_n)$ satisfies the condition (2).*

Lemma 10. *For any monotone function $f^D(x_1,\ldots,x_n)$, $n \geqslant 2$, there exists a monotone function $g^D(x_1,\ldots,x_n) \in \left\{\{f^D(x_1,\ldots,x_n)\} \cup M_3(1)\right\}$ such that the condition (2) holds on such sets $\tilde{\alpha}, \tilde{\beta} \in E_3^n$ that $g^D(\tilde{\alpha}\&\tilde{\beta}) = g^D(\tilde{\alpha}) = g^D(\tilde{\beta}) = 0$, $g^D(\tilde{\alpha} \vee \tilde{\beta}) = 1$.*

Proof. Since $f^D(x_1,\ldots,x_n) \in D$, the condition (2) holds for this. Let us consider two sets $\tilde{\alpha}, \tilde{\beta}$ such that $f^D(\tilde{\alpha} \vee \tilde{\beta}) > f^D(\tilde{\alpha}) \vee f^D(\tilde{\beta})$. The sets $\tilde{\alpha}$ and $\tilde{\beta}$ are incomparable; otherwise, there is a contradiction with the condition (2).

Consider four sets: $\Gamma = \{\tilde{\alpha}\&\tilde{\beta}, \tilde{\alpha}, \tilde{\beta}, \tilde{\alpha} \vee \tilde{\beta}\}$. Consider also the set

$$S_\Gamma = \{f^D(\tilde{\alpha}\&\tilde{\beta}), f^D(\tilde{\alpha}), f^D(\tilde{\beta}), f^D(\tilde{\alpha} \vee \tilde{\beta})\}.$$

It is clear that $|S_\Gamma| \geqslant 2$. Otherwise, we have a contradiction with the condition (2).

Let $|S_\Gamma| = 2$. Denote by a the minimum element of S_Γ, and by b—the maximum element. Due to the inequality $f^D(\tilde{\alpha} \vee \tilde{\beta}) > f^D(\tilde{\alpha}) \vee f^D(\tilde{\beta})$ it is clear that $f^D(\tilde{\alpha} \vee \tilde{\beta}) = b$, $f^D(\tilde{\alpha}) = a$, $f^D(\tilde{\beta}) = a$. Then, we have $f^D(\tilde{\alpha}\&\tilde{\beta}) = a$ by the monotonicity of the function f^D.

Denote by $j(x)$ a function $M_3(1)$ such that $j(a) = 0$ and $j(b) = 1$. Consider a function $g(x_1,\ldots,x_n) = j(f^D(x_1,\ldots,x_n))$ that satisfies the relation $g(\tilde{\alpha}\&\tilde{\beta}) = j(f^D(\tilde{\alpha}\&\tilde{\beta})) = j(a) = 0$. Analogically, $g(\tilde{\alpha}) = 0$, $g(\tilde{\beta}) = 0$, $g(\tilde{\alpha} \vee \tilde{\beta}) = 1$.

By these equalities, the function g satisfies the condition $(*)$, namely $g(\tilde{\alpha} \vee \tilde{\beta}) > g(\tilde{\alpha}) \vee g(\tilde{\beta})$. Hence, $g \notin D$ holds according to the definition of class D.

Let $|S_\Gamma| = 3$. Then, $f^D(\tilde{\alpha} \vee \tilde{\beta}) = 2$ holds surely, since f^D is monotone. By the condition (2) no other element from S_Γ cannot be equal to 2.

Consider a function $h(x_1,\ldots,x_n) = j_2(f^D(x_1,\ldots,x_n))$. For this $h(\tilde{\alpha} \vee \tilde{\beta}) = j_2(f^D(\tilde{\alpha} \vee \tilde{\beta})) = j(2) = 1$. Analogically, $h(\tilde{\alpha}\&\tilde{\beta}) = h(\tilde{\alpha}) = h(\tilde{\beta}) = 0$, since $f^D(\tilde{\alpha}), f^D(\tilde{\beta}), f^D(\tilde{\alpha}\&\tilde{\beta}) < 2$.

By these equalities, the function h satisfies the condition (2), specifically, $h(\tilde{\alpha} \vee \tilde{\beta}) > h(\tilde{\alpha}) \vee h(\tilde{\beta})$. Hence, $h \notin D$ holds according to the definition of the class D. □

Lemma 11. *For any monotone function $f^D(x_1, \ldots, x_n)$, $n \geqslant 2$ there is a monotone function $g^D(x, y)$, which depends essentially on two variables such that $g^D(x, y) \in \{\{f^D(x_1, \ldots, x_n)\} \cup M_3(1)\}$.*

Proof. The condition (2) holds by definition of the class D for function $f^D(x_1, \ldots, x_n)$, $n \geqslant 2$: there are sets $\tilde{\alpha}, \tilde{\beta} \in E_3^n$ such that $f(\tilde{\alpha} \vee \tilde{\beta}) > f(\tilde{\alpha}) \vee f(\tilde{\beta})$.

If j $(1 \leqslant j \leqslant n)$ exists such that $\alpha_j = \beta_j$, then by Lemma 8 we have a function that depends on a smaller number of variables for which the condition (2) holds. Therefore, this function does not belong to the class D.

If i, j $(1 \leqslant i, j \leqslant n, i < j)$ exists such that $\alpha_i = \alpha_j$, $\beta_i = \beta_j$, then by Lemma 9 we obtain a function depending on a smaller number of variables, and this function satisfies the condition (2), and therefore does not belong to class D.

Let $\tilde{\alpha} = (\alpha_1, \ldots, \alpha_n)$, $\tilde{\beta} = (\beta_1, \ldots, \beta_n)$ such that for any j $(1 \leqslant j \leqslant n)$ $\alpha_j \neq \beta_j$ and for any i, j $(1 \leqslant i, j \leqslant n, i < j)$ either $\alpha_i \neq \alpha_j$ or $\beta_i \neq \beta_j$.

The set $\{(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)\}$ is contained in the set $\{(0,1), (1,0), (2,0), (0,2), (1,2), (2,1)\}$. There, $n \leqslant 6$, but by Lemma 7, $n \geqslant 2$.

Let at least one of the sets $\tilde{\alpha}, \tilde{\beta}$ include one. By our assumption for any j $(1 \leqslant j \leqslant n)$ $\alpha_j \neq \beta_j$. Then, there is i $(1 \leqslant i \leqslant n)$ such that either $\alpha_i = 0$, $\beta_i = 1$ or $\alpha_i = 1$, $\beta_i = 2$ (without loss of generality, we can assume that $\alpha_i < \beta_i$).

In addition, without loss of generality, assume that $i = 1$. Then, we can consider a function $g(x_1, \ldots, x_n) = f(j(x_1), \ldots, x_n)$, where

$$j(x_1) = \begin{cases} \alpha_1, & x_1 = 0; \\ \beta_1, & x_1 > 0. \end{cases}$$

The condition (2) for a function g holds on sets $\tilde{\gamma} = (0, \alpha_2, \ldots, \alpha_n)$, $\tilde{\delta} = (2, \beta_2, \ldots, \beta_n)$. If one is included in at least one of sets $\tilde{\gamma}$, $\tilde{\delta}$, then we continue to follow a similar pattern. We can then assume that a condition (2) for the function $f^D(x_1, \ldots, x_n)$ holds on sets of zeros and twos.

That is, the sets $\tilde{\alpha}$, $\tilde{\beta}$ consist of zeros and twos. Because for any j $(1 \leqslant j \leqslant n)$ $\alpha_j = \beta_j$ and for any i, j $(1 \leqslant i, j \leqslant n, i < j)$ either $\alpha_i \neq \alpha_j$ or $\beta_i \neq \beta_j$ the set $\{(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)\}$ is contained in the set $\{(2,0), (0,2)\}$. Therefore, $n \geqslant 2$, but by Lemma 7 $n \geqslant 2$. Hence, $n = 2$. □

A similar statement is true for class K. This is proven in a similar way.

Lemma 12. *For any monotone function $f^K(x_1, \ldots, x_n)$, $n \geqslant 2$, a monotone function $g^K(x, y)$ depends essentially on two variables such that $g^K(x, y) \in \{\{f^K(x_1, \ldots, x_n)\} \cup M_3(1)\}$.*

Theorem 3. *Let the monotone function $f(x_1, \ldots, x_n)$, $n \geqslant 2$ take three values. Then, the set $\{\{f(x_1, \ldots, x_n)\} \cup M_3(1)\}$ contains a monotone function that essentially depends on exactly two variables and takes three values.*

Proof. (induction by the number of variables).

$n = 3$ **(induction base)**

Let $f = f(x_1, x_2, x_3)$.

(i) If variables x_i, x_j $(i \neq j, i, j = 1, 2, 3)$ of function f exist such that when they are identified, we obtain a function that depends essentially on more than one variable, and there is a set $\tilde{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ such that $\alpha_i = \alpha_j$ and $f(\tilde{\alpha}) = 1$, then it is clear that, by identifying the variables x_i and x_j, we obtain the desired monotone function that essentially depends on exactly two variables and takes three values, since $f(0,0,0) = 0$ and $f(2,2,2) = 2$.

(ii) Suppose that there is no such pair of variables, i.e., for any identification of two variables of the function f, we obtain a function that depends on less than two variables. By $f(0,0,0) = 0$ and $f(2,2,2) = 2$ for each identification, we can obtain only functions from $M_3(1) \cap T_0 \cap T_2$. There are only three possible ways to choose a pair of variables for identification.

Suppose the functions $f(x,x,y)$, $f(x,y,x)$, $f(y,x,x)$ are known. Then, the values of the initial function $f(x_!, x_2, x_3)$ are defined on all sets except six which all have three different elements. Because the value of $f(1,1,1)$ is uniquely defined, it is clear that if $f(x,x,y) = r(x)$ or $f(x,x,y) = r(y)$, then $f(x,y,x), f(y,x,x) \in \{r(x), r(y)\}$ for any function r from set $M_3(1) \cap T_0 \cap T_2$. For every function r from $M_3(1) \cap T_0 \cap T_2$, namely for J_2, J_1, and function e_1^1, we must consider the following eight cases:

$$\begin{aligned}
&1.\ \ f(x,x,y) = r(x),\ \ f(x,y,x) = r(x),\ \ f(y,x,x) = r(x),\\
&2.\ \ f(x,x,y) = r(x),\ \ f(x,y,x) = r(x),\ \ f(y,x,x) = r(y),\\
&3.\ \ f(x,x,y) = r(x),\ \ f(x,y,x) = r(y),\ \ f(y,x,x) = r(x),\\
&4.\ \ f(x,x,y) = r(x),\ \ f(x,y,x) = r(y),\ \ f(y,x,x) = r(y),\\
&5.\ \ f(x,x,y) = r(y),\ \ f(x,y,x) = r(x),\ \ f(y,x,x) = r(y),\\
&6.\ \ f(x,x,y) = r(y),\ \ f(x,y,x) = r(x),\ \ f(y,x,x) = r(x),\\
&7.\ \ f(x,x,y) = r(y),\ \ f(x,y,x) = r(y),\ \ f(y,x,x) = r(y),\\
&8.\ \ f(x,x,y) = r(y),\ \ f(x,y,x) = r(y),\ \ f(y,x,x) = r(x).
\end{aligned}$$

Consider the following cases:

1. If $r = J_2$ or $r = J_1$, then the function $f(x_1, x_2, x_3)$ takes two values. However, this is a contradiction with the initial condition. There is only one case remaining: when the function $f(x_1, x_2, x_3)$ is defined as follows: $f(x,x,z) = x$, $f(x,y,y) = y$, $f(z,y,z) = z$, i.e., f is a majority function of three variables that demonstrates the constant substitution operation. By substituting one for the first variable, we obtain g from $M \setminus M^{(2)}$, which depends essentially on two variables. Specifically,

$$g(x,y) = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 2 \end{pmatrix}.$$

2. It is easy to see that $f(x_1, x_2, x_3) = r(x_1)$.
3. It is easy to see that $f(x_1, x_2, x_3) = r(x_2)$.
4. For any function r from $M_3(1) \cap T_0 \cap T_2$, the equality $f(0,2,0) = 2$ and $f(0,2,2) = 0$ hold. However, this contradicts the monotonicity of the function $f(x_1, x_2, x_3)$.
5. For any function r from $M_3(1) \cap T_0 \cap T_2$ the equality $f(2,0,0) = 2$ and $f(2,2,0) = 0$ hold. However, this contradicts the monotonicity of the function $f(x_1, x_2, x_3)$.
6. It is easy to see that $f(x_1, x_2, x_3) = r(x_3)$.
7. For any function r from $M_3(1) \cap T_0 \cap T_2$ the equality $f(2,0,0) = 2$, $f(2,2,0) = 0$ hold. However, this contradicts the monotonicity of the function $f(x_1, x_2, x_3)$.
8. For any function r from $M_3(1) \cap T_0 \cap T_2$ the equality $f(0,2,0) = 2$, $f(2,2,0) = 0$ hold. However, this contradicts the monotonicity of the function $f(x_1, x_2, x_3)$.

(iii) Suppose we have variables x_i, x_j $(i \neq j, i, j = 1,2,3)$ of function f, whose identification gives a function that essentially depends on two variables.

(a) By identifying variables x_k and x_l $(k \neq l, k, l = 1,2,3)$, we obtain a function that takes only three values. If $k = i$ and $l = j$, then the theorem is proven. Let $(i,j) \neq (k,l)$, then by identification of x_k, x_l, we can obtain a function that takes three values but depends essentially on no more than one variable. This function can only be a selector function e_i^3, where $i = 0,1,3$. Then, it is obvious that $f(0,0,0) = 0$, $f(1,1,1) = 1$, $f(2,2,2) = 2$. Then, the identification of any pair of variables x_i, x_j, gives the desired function from $M \setminus M^{(2)}$.

(b) By identifying any pair of variables, we obtain a function that takes at most two values. Consider a set $\tilde{\alpha}$ such that $f(\tilde{\alpha}) = 1$. Since identification of any pair of variables gives a function that takes at most two values, then $\alpha_i \neq \alpha_j$ is necessary to fulfill with $i \neq j, i, j = 1,2,3$.

Without loss of generality, we can assume that $\alpha_1 = 2$, $\alpha_2 = 0$, $\alpha_3 = 1$. Consider a set $\tilde{\beta}$ such that $\beta_1 = 2$, $\beta_2 = \beta_3 = 0$, and a set $\tilde{\gamma}$ such that $\gamma_1 = 2$, $\gamma_2 = \gamma_3 = 1$. By the monotonicity of the function $f(x,y,z)$, we obtain $f(\tilde{\beta}) \leqslant f(\tilde{\alpha})$.

Therefore, we have either $f(\tilde{\beta}) = 0$ or $f(\tilde{\beta}) = 1$. Let $f(\tilde{\beta}) = 1$, then by identifying the variables x_2, x_3, we have a function taking three values. However, this contradicts the assumption. Consequently, we have $f(\tilde{\beta}) = 0$. Similarly, we find that $f(\tilde{\gamma}) = 2$. Consider a function $g(x_2, x_3)$ derived from $f(x,y,z)$ by substituting the constant 2 instead of the variable x_1.

It is clear that $g(\beta_2, \beta_3) = g(0,0) = 0$, $g(\alpha_2, \alpha_3) = g(0,1) = 1$, $g(\gamma_2, \gamma_3) = g(1,1) = 2$. From these equalities and from the fact that $\beta_3 \neq \alpha_3$, $\beta_2 = \alpha_2$ and $\gamma_3 = \alpha_3$, $\gamma_2 \neq \alpha_2$ follows that variables x_2 and x_3 are essential for $g(x_2, x_3)$. Hence, $g(x_2, x_3)$ is the required function that takes three values and depends essentially on exactly two variables.

We have shown how to obtain from $M \setminus M^{(2)}$ a function g that depends essentially on two variables. This can be performed by identifying the variables and substituting constants into an arbitrary function f from $M \setminus M^{(2)}$, where f depends essentially on three variables.

$n > 3$ (by the inductive assumption)

Consider a set $\tilde{\alpha}$ such that $f(\tilde{\alpha}) = 1$. Because $n > 3$, then $i, j : \alpha_i = \alpha_j$ exist, $i \neq j$.

(i) If a set $\tilde{\alpha}$ consists of zeros and twos only and is such that $f(\tilde{\alpha}) = 1$, then we identify the variables x_i and x_j. Suppose that we obtain a function r that essentially depends on only one variable by the identity of variables x_i, x_j. Then, from one hand, we have $r(0) = 0$ and $r(2) = 2$ since $f(\tilde{0}) = 0$ and $f(\tilde{2}) = 2$, and by $f(\tilde{\alpha}) = 1$, we have either $r(0) = 1$, or $r(2) = 1$ from the other hand. We obtained a contradiction.

(ii) Let all tuples $\tilde{\alpha}$ such that $f(\tilde{\alpha}) = 1$ contain one. Let us take one of these sets. Because $n > 3$ there are $i, j such that \alpha_i = \alpha_j$, $i \neq j$. Let $\alpha_{i_1} = \ldots = \alpha_{i_m} = 1$, $m > 0$. Suppose that by identifying the numbers x_i and x_j, we obtain a function r that depends essentially only on one variable. Then, it is clear that $r \in \{e^n_{i_1}, \ldots, e^n_{i_m}\}$, since otherwise we obtain that either $r(0) = 1$, or $r(2) = 1$. Hence, we have $f(\tilde{1}) = 1$. Then, by identifying at least one pair of variables of the function $f(x_1, \ldots, x_n)$, we obtain a function that depends on more than one variable. The inductive step has been performed. Suppose the identification of any two variables results in a function that essentially depends on one variable. Let us prove that this is not true if f essentially depends on n variables.

(a) If, for any identification of two numbers, we obtain the function e^n_i (the number i is the same for all identifications), then we obtain a contradiction with the fact that f depends essentially on n variables, since we obtain $f = e^n_i$.

(b) Suppose variables x_{i_1}, x_{j_1} exist such that we obtain $e^n_{k_1}$ by their identification. Also, suppose variables x_{i_2}, x_{j_2} exist such that we obtain $e^n_{k_2}$ by their identification; moreover, we have $e^n_{k_1} \neq e^n_{k_2}$. Therefore, we have $k_1 \neq k_2$.

(1) $k_1 \notin \{i_1, j_1\}$.
Consider a set $\tilde{\alpha}$ such that $\alpha_{k_1} = 2$, $\alpha_j = 1$ for any $j \neq k_1$, and a set $\tilde{\beta}$ such that $\beta_{k_2} = 1$, $\beta_j = 2$ for any $j \neq k_2$. Then $\tilde{\alpha} \leqslant \tilde{\beta}$, $f(\tilde{\alpha}) = 2$, $f(\tilde{\beta}) = 1$, but this contradicts the monotonicity of the function f.

(2) $k_1 = i_1$. Note that in this case $k_2 \neq i_1$.

(2.1) $k_2 \notin \{i_2, j_2\}$.

(2.1.1) $k_2 \neq j_1$. Consider a set $\tilde{\alpha}$ such that $\alpha_{k_1} = 2$, $\alpha_{j_1} = 2$, $\alpha_j = 1$ for any $j \neq k_1, j_1$, and a set $\tilde{\beta}$ such that $\beta_{k_2} = 1$, $\beta_j = 2$ for any $j \neq k_2$. Then $\tilde{\alpha} \leqslant \tilde{\beta}$, $f(\tilde{\alpha}) = 2$, $f(\tilde{\beta}) = 1$, but this contradicts the monotonicity of the function f.

(2.1.2) $k_2 = j_1$. Then, $e^n_{k_2} = e^n_{j_1} = e^n_{i_1} = e^n_{k_1}$, this contradicts our assumption.

(2.2) $k_2 = i_2$.

(2.2.1) $k_2 = j_1$. Then, $e^n_{k_2} = e^n_{j_1} = e^n_{i_1} = e^n_{k_1}$, this contradicts our assumption.

(2.2.2) $k_2 \neq j_1$.

(2.2.2.1) $j_2 \in \{i_1, j_1\}$. Then, $e^n_{k_2} = e^n_{j_1} = e^n_{i_1} = e^n_{k_1}$, this contradicts our assumption.

(2.2.2.2) $j_2 \notin \{i_1, j_1\}$. Consider a set $\tilde{\alpha}$ such that $\alpha_{k_1} = 2$, $\alpha_{j_1} = 2$, $\alpha_j = 1$ for any $j \neq k_1, j_1$, and a set $\tilde{\beta}$ such that $\beta_{k_2} = 1$, $\beta_{j_2} = 1$, $\beta_j = 2$ for any $j \neq k_2, j_2$. Then, $\tilde{\alpha} \leqslant \tilde{\beta}$, $f(\tilde{\alpha}) = 2$, $f(\tilde{\beta}) = 1$, but this contradicts the monotonicity of the function f.

(2.3) $k_2 = j_2$. In the same way (2.2).

(3) $k_1 = j_1$. In the same way (2).

Thus, for any $i_1, j_1, i_2, j_2, k_1, k_2$, there is either a contradiction with the monotonicity of the function f or with the fact that it depends essentially on n variables.

Consequently, there are variables of the function f identifying which one obtains a function that depends essentially on more than one variable. □

Now, we are ready to formulate and prove the final result.

Theorem 4 (on finite generation of classes of monotone functions containing all unary monotone functions)**.** *Let $F \subseteq M$, $M_3(1) \subseteq F$. Then F is a finitely generated class.*

Proof. We consider four possible cases to prove this theorem:

1. Case I. $x \vee y \in F$, $x\&y \in F$.
2. Case II. $x \vee y \in F$, $x\&y \notin F$.
3. Case III. $x \vee y \notin F$, $x\&y \in F$.
4. Case IV. $x \vee y \notin F$, $x\&y \notin F$.

In Case I, we obtain a class that coincides with M. In Case II and Case III, we obtain one class that satisfies satisfies the conditions of the theorem and differs from M and $M_3(1)$. In Case IV, we have two subclasses in $M^{(2)}$.

Case II. Let $x \vee y \in F$, $x\&y \notin F$.

(1) For any function $f \in F$ the following condition holds: $f \in \{\{x \vee y\} \cup M_3(1)\}$. In this case, a class $F = D = \{\{x \vee y\} \cup M_3(1)\}$ is finitely generated.

(2) The function $f^D \in F$ exists.

By Lemma 11, it is sufficient to consider $f^D(x, y)$ (similarly, we can prove the existence $f^K(x, y)$ by Lemma 12). By Lemma 10 we can assume $f^D(\tilde{a} \vee \tilde{b}) = 1$, $f^D(\tilde{a}) = f^D(\tilde{b}) = f^D(\tilde{a}\&\tilde{b}) = 0$ for some sets $\tilde{a}$, $\tilde{b}$. Hence, for function $f^D(x, y)$ on sets $\tilde{a}$, $\tilde{b}$, the condition $(*)$ holds; therefore, sets $\tilde{a} = (\alpha_1, \alpha_2)$ and $\tilde{b} = (\beta_1, \beta_2)$ are incomparable. Suppose that $\alpha_1 < \beta_1$, $\alpha_2 < \beta_2$. Then,

$$f^D(p(x), q(y)) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \text{where } p(x) = \begin{pmatrix} \alpha_1 \\ \alpha_1 \\ \beta_1 \end{pmatrix}, q(y) = \begin{pmatrix} \beta_2 \\ \beta_2 \\ \alpha_2 \end{pmatrix}.$$

Application of Lemma 4 yields $x\&y \in \{\{x \vee y\} \cup M_3(1) \cup \{f^D(x,y)\}\}$, and consequently, by Lemma 3, we obtain that $\{\{x \vee y\} \cup M_3(1) \cup \{f^D(x,y)\}\} = M$. Hence, $\{D \cup \{f^D(x,y)\}\} = M$.

Then, on the one hand, we have $\{\{x \vee y\} \cup M_3(1)\} = D \subset F$, and on the other hand, $f^D \in F$. It has been proved above that the set $D = \{\{x \vee y\} \cup M_3(1)\}$ is precomplete in M. From this, it follows that $F = M$. Case II has been considered and the theorem has been proved. □

5. Conclusions

In this paper, two theorems on the finiteness generation of classes containing precomplete classes of one-valued functions, and classes containing all monotone unary functions are proved. These results enable us to explicitly describe a class that contains a subclass of one-variable functions and provide a description of a supergroup of classes from P_k that contains the full class of unary functions. Similar conditions were previously described for binary logic. In this paper, we present their counterpart for many-valued logic. The main challenge with many-valued logic is that we cannot explicitly describe closed classes. The only thing that can be performed is to apply certain criteria to determine whether the class is complete or not. If these criteria are met, then the class is considered to be finitely generated. It is important that these conditions can be easily verified. Specifically, this condition is expressed in terms of one-valued functions. It has been proven that all classes containing one of the precomplete classes of the set of unary functions are finitely generated. The main result of this paper consists of two theorems on the finite generation of classes containing precomplete classes of one-valued functions and classes containing all monotone unary functions. Thus, the theoretical result provides a criterion for checking the finiteness of classes of functions in multivalued logic.

The finite generation of overlattices means that we can develop consistent computation schemes that are quite useful for applications. Although this paper is rather theoretical, the results obtained here are all significant in their respective areas of study and may be applied extensively in computer science. In practice, the proved theorems mean that we can construct circuit schemes or algorithms explicitly for general cases.

Some problems remain open. On the basis of the considerations above, it is possible to construct a diagram of the inclusions of monotone classes that contain all monotone functions. It can also be proven that this inclusion diagram of monotone classes includes all unary monotone functions. For further research, we plan to consider finitely generated classes that do not contain the majority functions and a choice function. In addition, there is an example of a class whose finite generation cannot be proven via the one sufficient condition described in the first part of this paper. However, the method of modeling constants allows proving their finite generation based on the fact that a class $\{P_k\}$ is finitely generated.

Funding: This publication was prepared with the support of the Russian Foundation for Basic Research according to the research project No. 20-01-00575_A.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Bykovsky, A.Y. Heterogeneous Network Architecture for Integration of AI and Quantum Optics by Means of Multiple-Valued Logic. *Quantum Rep.* **2020**, *2*, 126–165. [CrossRef]
2. Aizenberg, I. Multiple-Valued Logic and Complex-Valued Neural Networks. In *Claudio Moraga: A Passion for Multi-Valued Logic and Soft Computing*; Seising, R., Allende-Cid, H., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 153–171. [CrossRef]
3. Kunze, H. *Engineering Mathematics and Artificial Intelligence: Foundations, Methods, and Applications*; Mathematics and Its Applications: Modelling, Engineering, and Social Sciences; CRC Press: Boca Raton, FL, USA, 2024.

4. Luo, C.; Wang, H.; Zheng, Y. Controllability of k-Valued Fuzzy Cognitive Maps. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 1694–1707. [CrossRef]
5. Kong, X.; Sun, Q.; Li, H. Survey on Mathematical Models and Methods of Complex Logical Dynamical Systems. *Mathematics* **2022**, *10*, 3722. [CrossRef]
6. Xiaojie, W.; Chao, L.; Chen, L. The feedback stabilization of finite-state fuzzy cognitive maps. *Trans. Inst. Meas. Control* **2022**, *44*, 2485–2499. [CrossRef]
7. Pfeifer, A.; Lohweg, V. Classification of Faults in Cyber-Physical Systems with Complex-Valued Neural Networks. In Proceedings of the 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vasteras, Sweden, 7–10 September 2021; pp. 1–7. [CrossRef]
8. Bykovsky, A.Y. Multiple-Valued Logic Modelling for Agents Controlled via Optical Networks. *Appl. Sci.* **2022**, *12*, 1263. [CrossRef]
9. Kalimulina, E.Y. Lattice Structure of Some Closed Classes for Three-Valued Logic and Its Applications. *Mathematics* **2022**, *10*, 94. [CrossRef]
10. Kalimulina, E.Y. On Exponential Convergence of Dynamic Queueing Network and Its Applications. In *Distributed Computer and Communication Networks*; Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V., Eds.; Springer: Cham, Switzerland, 2020; pp. 463–474.
11. Kalimulina, E.Y. Analysis of Unreliable Open Queueing Network with Dynamic Routing. In *Distributed Computer and Communication Networks*; Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V., Eds.; Springer: Cham, Switzerland, 2017; pp. 355–367.
12. Bykovsky, A.Y.; Vasiliev, N.A. Data Verification in the Agent, Combining Blockchain and Quantum Keys by Means of Multiple-Valued Logic. *Appl. Syst. Innov.* **2023**, *6*, 51. [CrossRef]
13. Kalimulina, E.Y. Lattice Structure of Some Closed Classes for Non-binary Logic and Its Applications. In *Mathematical Methods for Engineering Applications*; Yilmaz, F., Queiruga-Dios, A., Santos Sánchez, M.J., Rasteiro, D., Gayoso Martínez, V., Martín Vaquero, J., Eds.; Springer: Cham, Switzerland, 2022; pp. 25–34.
14. Kalimulina, E.Y. Application of Multi-Valued Logic Models in Traffic Aggregation Problems in Mobile Networks. In Proceedings of the 2021 IEEE 15th International Conference on Application of Information and Communication Technologies (AICT), Baku, Azerbaijan, 13–15 October 2021; pp. 1–6. [CrossRef]
15. Cheng, D.; Feng, J.e.; Zhao, J.; Fu, S. On adequate sets of multi-valued logic. *J. Frankl. Inst.* **2021**, *358*, 6705–6722. [CrossRef]
16. Cheng, D.; Liu, Z.; Qi, H. Completeness and normal form of multi-valued logical functions. *J. Frankl. Inst.* **2020**, *357*, 9871–9884. [CrossRef]
17. Malkov, M.A. Algebra of Finite-Valued Functions: Classification of Functions and Subalgebras, Essential and Fictitious Subalgebras. *Pure Appl. Math. J.* **2019**, *8*, 30–36. [CrossRef]
18. Ding, X.; Li, H. Stability analysis of multi-valued logical networks with Markov jump disturbances. *Int. J. Control* **2022**, *95*, 554–561. [CrossRef]
19. Xu, N.; Chen, Y.; Xue, A.; Zhao, X.; Sun, P. Set stabilization and optimal control of switched multi-valued logical control networks with state-dependent switching signals. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *68*, 1952–1956. [CrossRef]
20. Yablonskiy, S.; Gavrilov, G.; Kudryavtsev, V. *Logical Algebra Functions And Post Classes*; Nauka: Moscow, Russia, 1966.
21. Parhami, B.; McKeown, M. Arithmetic with binary-encoded balanced ternary numbers. In Proceedings of the 2013 Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 3–6 November 2013; pp. 1130–1133. [CrossRef]
22. Post, E.L. Introduction to a General Theory of Elementary Propositions. *Am. J. Math.* **1921**, *43*, 163–185. [CrossRef]
23. Post, E.L. *The Two-Valued Iterative Systems of Mathematical Logic. (AM-5)*; Princeton University Press: Princeton, NJ, USA, 1941.
24. Yanov, Y.I.; Muchnik, A. The Existence of a k-Valued Closed Class that Has No Finite Basis. *Doklady Akad. Nauk SSSR* **1959**, *127*, 44.
25. Esin, A.; Yavorskiy, R.; Zemtsov, N. Brief Announcement Monitoring of Linear Distributed Computations. In *Distributed Computing*; Dolev, S., Ed.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 566–568.
26. Podol'ko, D.K. A family of classes of functions closed with respect to a strengthened superposition operation. *Mosc. Univ. Math. Bull.* **2015**, *70*, 79–83. [CrossRef]
27. Starostin, M.V. Implicitly Maximal Classes and Implicit Completeness Criterion in the Three-Valued Logic. *Mosc. Univ. Math. Bull.* **2018**, *73*, 82–84. [CrossRef]
28. Kalimulina, E.Y. Mutual Generation of the Choice and Majority Functions. In *Mathematical Methods for Engineering Applications*; Yilmaz, F., Queiruga-Dios, A., Martín Vaquero, J., Mierluş-Mazilu, I., Rasteiro, D., Gayoso Martínez, V., Eds.; Springer: Cham, Switzerland, 2023; pp. 49–57.
29. Burle, G. Classes ofk-valued logic containing all functions of a certain variable. *Diskretn. Anal.* **1967**, *10*, 3–7.
30. Posypkin, M.A. On closed classes containing precomplete classes of the set of all one-place functions. *Mosc. Univ. Math. Bull.* **1997**, *52*, 58–59.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.