
Clones of Self-Dual and Self-K-Al Functions in K-valued Logic

M. A. Malkov

Russian Research Center for Artificial Intelligence, Moscow, Russia

Email address:

mamalkov@gmail.com

To cite this article:

M. A. Malkov. Clones of Self-Dual and Self-K-Al Functions in K-valued Logic. *Pure and Applied Mathematics Journal*.

Vol. 6, No. 2, 2017, pp. 59-70. doi: 10.11648/j.pamj.20170602.11

Received: February 17, 2017; **Accepted:** February 14, 2017; **Published:** March 10, 2017

Abstract: We give a classification of dual functions, they are m -al functions. We call a function m -al with respect to an operator if the operator lives any function unchanged after m times of using the operator. And $2 \leq m \leq k$. Functions with different m have very different properties. We give theoretical results for clones of self-dual ($m = 2$) and self- k -al ($m = k$) functions in k -valued logic at $k \leq 3$. And we give numerical results for clones of self-dual and self-3-al functions in 3-valued logic. In particular, the inclusion graphs of clones of self-dual and of self-3-al functions are not a lattice.

Keywords: Discrete Mathematics, K-Valued Function Algebra, Selfdual Functions

1. Introduction

Multi-valued logic attracts intensive attention because of the connection with computer technology. But study of the logic is very difficult since connections of clones are very tangled and sometimes a graph of inclusions of clones is not a lattice. Some classification of all clones was given in [1].

More essential results are got for 3-valued logic. In particular, all maximal clones were found by S. V. Yablonskiy [2], all sub-maximal clones were found by D. Lau [3], all minimal clones were found by B. Csahány [4], all subclones of the linear clone were found by D. Lau [3], some self-3-al clones were found by S. S. Marchenkov, J. Demetrovics, and L. Hannak [5]. More complete set of self-3-al clones is given in the article. The set of monotone clones was investigated by Machida [6]. The other types of clones were investigated very little.

The article consists of 2 sections.

Section 1 contains theoretical results. We select classes of functions from clones of functions since the classes are disjoint but clones are not disjoint. Some theorems are true only for the classes. We define conditions satisfied by functions and find their properties. The conditions are an extension of Post's conditions and they are used to prove some important theorems. We introduce an m -al operator which lives any function unchanged after using the operator m times. Then we give some properties of dual and k -al

operators. We find the numbers of relations preserved by dual and k -al functions.

Section 2 contains numerical results. We build the inclusion graphs of clones of self-dual and 3-al functions and find properties of the functions.

We use the next notation. The letter k is used for values of logics only. The letter n is used for numbers of variables of functions only. The letters i and j are used as any or all values of subscripts. The letter b is used as a value of variables. The letter l is used as values of functions. The sign $+$ is used as addition by modulo k . We use $\{\}$ for empty sets. And we use \bar{b} for any value which does not equal b .

We say that a relation generates a set of functions if the functions preserve the relation. We use Boolean projections of functions if values of functions are 0 or 1 whenever values of variables are 0 or 1. Any Boolean projection is a Boolean function and can be presented by disjunctive normal form (DNF). We use sign “,” to separate values of variables from values of functions in lines of tables of functions. We call a sequence of values of variables a *tuple* and we can present a line of a table of a function as a *tuple* l , where the tuple contains values of variables and l is a value of the function. We call the values of variables *components of the tuple*.

2. Theoretical Results in K-valued Logic

2.1. Classes of Compositions of Function

Clones of functions are called sometimes classes. This is not well since classes must be disjoint. But clones are not disjoint, they can contain other clones. And a clone can belong to several clones.

Below we give a definition of disjoint classes. We use them since some theorems will true for classes only. And some properties of clones are in classes only.

Definition A class of a clone is a set of all functions to be generators of the clone.

Further we call classes of functions briefly classes. A name of a class is a name of a clone containing the class.

Theorem. Classes are disjoint.

Proof. Any function generates a clone set. Every function of a class generates its clone X . This function cannot belong to a class of a clone contained in X since, by definition, the function does not belong to any clone contained in X . And the function cannot belong to a class of another clone since the function generates only the clone X . \square

Corollary. Every function belongs to one of classes.

Proof. Every function generating a clone belongs only to the class of the clone. \square

Hence, classes give a classification of functions.

If we remove all clones contained in any clone then we get a class of the clone. A function generating a clone is a member of one-membered basis. If a clone has no one-membered basis then we get an empty class after deleting other clones in the clone, since every function of a multi-membered basis belongs to one of deleted clones. But empty classes must be absent in any classification. Then clones with multi-valued bases are fictitious since they are useless for the classification of functions.

2.2. Family of Functions with the Same Diagonal

We will call families of functions with the same diagonal briefly families.

Families were introduced by Post. There are 4 families in two-valued logic: α , β , γ , δ . Post used the families to build his classification of Boolean functions.

Families of functions exist in all k -valued logics too. The number of these families is 2^k . We will use them to classify some sets of clones.

The family α is defined standardly: $f(x, \dots, x) = x$. The family δ is defined standardly, too: $f(x, \dots, x) \neq x$. Definitions of the other families are special in k -valued logics but they are an extension of Post's definitions.

We denote all types of families by $\{j_1, \dots, j_m\}$, where $j_i \in \{0, \dots, k-1\}$ and where $0 \leq m \leq k-1$.

Definition A function f belongs to a family $\{j_1, \dots, j_m\}$ if $f(x, \dots, x)$ is equal to x at $x = j_1 \vee \dots \vee x = j_m$, and is not equal to x at $x \neq j_1 \wedge \dots \wedge x \neq j_m$.

Hence, the α family is denoted by $\{0, 1, \dots, k-1\}$ and the δ family is denoted by $\{\}$. A constant j belongs to the family $\{j\}$ because a diagonal of a constant is this constant. The β family of two-valued logic is denoted by $\{1\}$ and the γ family is denoted by $\{0\}$.

The denotation $\{j_1, \dots, j_m\}$ means that diagonal of a function preserves values j_1, \dots, j_m , i.e., if all variables of a function equal a j_i then the value of the function equals the j_i . But the function does not preserve the other values.

And sets of functions of families with the same m are isomorphic with respect to compositions.

The next theorem is very useful in k -valued logic.

Theorem. Any function of a family $X = \{j_1, \dots, j_m\}$ generates functions of a family $Y \supseteq X$.

Proof. The operators of permutation and identification do not change a family of a function. The operator of substitution can change a family of a function.

Let functions $f_1^{n_1}, f_2^{n_2}$ be of the same family X , $f(x_1, \dots, x_{n_1+n_2-1}) = f_1^{n_1}(f_2^{n_2}(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})$, and denotation of family of f be Y .

We have $f(j, \dots, j) = j$ for every $j \in X$. In this case $j \in Y$ for all $j \in X$. Then $Y \supseteq X$ because $f(j, \dots, j)$ can equal j for some $j \notin X$. But if $f(j, \dots, j) \neq j$ for all $j \notin X$ then $Y = X$. The proof is finished. \square

Corollary. A function of the family α generates only function of this family. A function of the family δ may generate functions of any family.

Proof. The denotation of family α is the set $X = \{0, \dots, k-1\}$ and there is no family with denotation Y such that $Y \supset X$. The family δ has the denotation $X = \{\}$ and $Y \supseteq X$ for every family with denotation Y . \square

Families realize a natural classification of functions because they are disjoint.

2.3. Classes and Families

Functions of a clone can belong to several families. But functions of a class belong to one family only, as it follows from the next theorem.

Theorem. If a function of a class belongs to a family then all functions of the class belong to the family.

Proof. Let $k = 3$ and let a family be $\{j_1, \dots, j_m\}$ for $0 \leq m \leq 3$.

It is enough to use only families with $m = 3$ since the proof of the other families is analogous. Families with $m = 3$ are α , $\{0, 1, 2\}$, $\{0, 1, \bar{2}\}$, $\{0, \bar{1}, 2\}$, $\{\bar{0}, 1, 2\}$, $\{0, \bar{1}, \bar{2}\}$, $\{\bar{0}, 1, \bar{2}\}$, δ , where \bar{b} has any value except b .

The operators of permutation and identification of variables do not change a family.

Then we use the operator of substitution of the first variable

in a function f^{n_1} by a function f^{n_2} . Both functions belong to the same family.

If f^{n_1} and f^{n_2} belong to α family then the function $f^{n_1}(f^{n_2}(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})$ belongs to α family since the function equals x when all x_i equal x .

If f^{n_1} and f^{n_2} belong to $\{0, 1, \bar{2}\}$ family then $f^{n_1}(f^{n_2}(0, \dots, 0), 0, \dots, 0) = 0$, $f^{n_1}(f^{n_2}(1, \dots, 1), 1, \dots, 1) = 1$, $f^{n_1}(f^{n_2}(2, \dots, 2), 2, \dots, 2) = b$, where b has any value. If $b = 2$ then we have α family. If $b = \bar{2}$ then we have $\{0, 1, \bar{2}\}$ family. Hence, if a function of a class belongs to the family $\{0, 1, \bar{2}\}$ then all functions of the class belong to the family, since α functions belong to another class.

We will get the same result for families $\{0, \bar{1}, 2\}$ and $\{\bar{0}, 1, 2\}$.

If f^{n_1} and f^{n_2} belong to $\{0, \bar{1}, \bar{2}\}$ family then $f^{n_1}(f^{n_2}(0, \dots, 0), 0, \dots, 0) = 0$, $f^{n_1}(f^{n_2}(1, \dots, 1), 1, \dots, 1) = b_1$, $f^{n_1}(f^{n_2}(2, \dots, 2), 2, \dots, 2) = b_2$, where b_1 and b_2 have any value. If $b_1 = 1 \wedge b_2 = 2$ then we have α family. If $b_1 = 0 \wedge b_2 = \bar{2}$ or $b_1 = \bar{1} \wedge b_2 = 2$ then we have $\{0, 1, \bar{2}\}$ or $\{0, \bar{1}, 2\}$ family. If $b_1 = \bar{0} \wedge b_2 = \bar{2}$ then we have $\{0, \bar{1}, \bar{2}\}$ family. Hence, if a function of a class belongs to the family $\{0, \bar{1}, \bar{2}\}$ then all functions of the class belong to the family, since otherwise the function belongs to another class.

We will get the same result for families $\{\bar{0}, 1, \bar{2}\}$ and $\{\bar{0}, \bar{1}, 2\}$.

Let f^{n_1} and f^{n_2} belong to δ family. In this case if a function of a class belongs to δ family then all functions of the class belong to the family, since otherwise a function belongs to another class.

We will get the same result for $k > 3$. □

2.4. Conditions of Functions

Any function must satisfy some conditions.

There are 2 conditions for any function in 2-valued logic. They are a^μ and A^μ [7]. We will denote them by a_0^μ and a_1^μ . But there are k conditions in k -valued logic. They are $a_0^\mu, \dots, a_{k-1}^\mu$. This holds for all k including $k = \omega$.

Definition A function satisfies a condition

- o a_l^μ ($0 \leq \mu < \omega$) if any μ tuples l have a column with value l only, and if there are $\mu + 1$ tuples l which have not a column with value l only,

- o $a_l^{\omega+m}$ ($0 \leq m < \omega$) if we have m tuples l and all they have a column with value l .

If a function does not satisfy a condition of the first part of

definition then the function satisfies a condition of the second part of definition. Indeed, let a function do not satisfy a_l^μ . Then all tuples l are absent or m tuples l exist but $m + 1$ tuples l are absent. In the former case the function satisfies the condition a_l^ω . In the latter case the function satisfies the condition $a_l^{\omega+m}$.

By definition, if one of tuples l of a function does not contain value l of variables then the function satisfies the condition a_l^0 . And by definition, if a function takes not all values then the function satisfies the condition a_l^ω for absent l .

A constant j satisfies conditions a_j^0 and a_i^ω for $i \neq j$. Any delta function f satisfies a_i^0 for all i since $f(x, \dots, x) \neq x$.

The next lemma is very useful.

Lemma. If a function satisfies a condition a_l^μ for some l at $\mu < \omega$ then there are $\mu + 1$ tuples l and $\mu + 1$ columns in them such that every column contains only one value \bar{l} , and every tuple l has only one value \bar{l} in the columns.

Proof. By definition, if a function satisfies a condition a_l^μ for $\mu < \omega$ then there are $\mu + 1$ different tuples l not satisfying the condition. We take only columns of the tuples l such that every of the columns has only one value equal to \bar{l} . Such columns exist since otherwise some μ of $\mu + 1$ tuples l have no column with value l only. If the number of different columns with only one value \bar{l} is less than $\mu + 1$ then some μ of $\mu + 1$ tuples l have no column with value l only. But every μ of $\mu + 1$ tuples l have a column with value l only, iff there are $\mu + 1$ different columns with only one value \bar{l} . Every tuple l has only one value \bar{l} in the columns since the columns are different.

If $\mu < \omega$ then we can take lines and columns of any table such that values not equaling l form a diagonal:

$$\begin{array}{ccccccc}
 b_1 & l & l & \dots & l & & \\
 l & b_2 & l & \dots & l & & \\
 l & l & b_3 & \dots & l & & \\
 \cdot & \cdot & \cdot & \cdot & \cdot & & \\
 l & l & l & \dots & b_{\mu+1} & &
 \end{array}$$

where any b_i does not equal l . □

We will call the subset of tuples l sub-tuples l .

2.5. Classes and Conditions

Functions of a clone can satisfy several conditions. But functions of a class satisfy one condition only, as it follows

from the next theorem. For simplification of the proof we say that a function has μ for some l , if it satisfies a condition a_l^μ .

Theorem. *If a function of a class satisfies a condition $a_l^{\mu_0}$ at $0 \leq \mu_0 < \omega$ for some l then all functions of the class satisfy the condition.*

Proof. Let a function $f_0^{\mu_0}$ belong to a class and satisfy a condition $a_l^{\mu_0}$ at $0 \leq \mu_0 < \omega$ for some l . We must prove that $f_0^{\mu_0}$ generates the class of functions f with $\mu = \mu_0$. For that we use Mal'cev's operators of compositions [8].

The operator of permutation of variables does not change the condition since a permutation of variables is the permutation of columns, but the permutation does not change values into columns.

The operator of identification of variables does not change the conditions, too. Indeed, we can have the next $\mu_0 + 1$ sub-tuples l of tuples l of the function $f_0^{\mu_0}$:

$$\begin{array}{cccccc} b_1 & l & l & \dots & l & \\ l & b_2 & l & \dots & l & \\ l & l & b_3 & \dots & l & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \\ l & l & l & \dots & b_{\mu_0+1} & \end{array}$$

where any b_i has any value except l . If we identify some variables then a part of the sub-tuples l can be deleted. If no sub-tuple l is deleted then $\mu = \mu_0$, where μ belongs to the f and μ_0 belongs to the $f_0^{\mu_0}$. Otherwise we have one or more columns with value l only. Then $\mu > \omega$ but functions with the μ belong to another classes since functions with the μ cannot generate functions with $\mu < \omega$ (functions of a class must generate one to other). So the theorem holds in this case.

Now we must prove that the operator of substitution of the first variable in a function $f_1^{\mu_0}$ with μ_0 by the function $f_2^{\mu_0}$ does not change the condition. The operator gives

$$f_1^{\mu_0+n_2-1}(x_1, \dots, x_{n_1+n_2-1}) = f_1^{\mu_0}(f_2^{\mu_0}(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})$$

We must prove that the function $f_1^{\mu_0+n_2-1}$ has $\mu = \mu_0$. For that we will use tuples l of $f_1^{\mu_0+n_2-1}$. If we replace $f_2^{\mu_0}(x_1, \dots, x_{n_2})$ in the tuples l by values of $f_2^{\mu_0}$ then we have the next sub-tuples l of $f_1^{\mu_0+n_2-1}$:

$$\begin{array}{cccccc} b_1 & l & l & \dots & l & l \\ l & b_2 & l & \dots & l & l \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ l & l & l & \dots & l & b_{\mu_0+1} \end{array}$$

where any b_i has any value except l .

It cannot be $\mu < \mu_0$ since every μ_0 tuples l must have a column with values l only, then the μ tuples l have a column with values l only.

Hence, $\mu \geq \mu_0$. In reality, $\mu = \mu_0$. Indeed, let functions f_1 with μ_1 and f_2 with μ_2 belong to the same class. We have $\mu_2 \geq \mu_1$ since f_1 generates f_2 . And we have $\mu_1 \geq \mu_2$ since f_2 generates f_1 . Hence, $\mu_1 = \mu_2$. \square

Corollary. *If a function of a class satisfies a condition $a_l^{\mu_0}$ at $0 \leq \mu_0 < \omega$ for some l then all functions of the family in the clone of the class satisfy the condition with $\mu \geq \mu_0$ for the l .*

Proof. By previous proof a function with μ_0 generates functions with $\mu \geq \mu_0$. \square

Hence, if functions satisfy different a_l^μ with $\mu < \omega$ then the functions generate different clones. We will use the property below.

2.6. Boolean Projections

Boolean projections are used at classification of clones.

Definition A function f has a Boolean projection if $f(x_1, \dots, x_n) \in \{0, 1\}$ when all $x_i \in \{0, 1\}$. We denote a Boolean projection by prf . A clone of functions has a Boolean projection if every function of the clone has some Boolean projection.

Lemma. If a function has a Boolean projection then a clone generated by the function has a Boolean projection.

Proof. It is enough to prove that a Boolean projection of a function and a Boolean projection of any composition of the function belong to the same clone.

If we permute or identify variables of a function then we permute or identify variables in the Boolean projection of the function. Then the initial Boolean projection and the new Boolean projection belong to the same clone.

Let functions f^{μ_1} , f^{μ_2} belong to the same clone and have Boolean projections. The operation of substitution gives the function $f^{\mu_1+n_2-1}$ that equals $f^{\mu_1}(f^{\mu_2}(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})$. The Boolean projection of the function is a result of substitution of the Boolean projections. Indeed, $f^{\mu_2} \in \{0, 1\}$ if its $x_i \in \{0, 1\}$. And $f^{\mu_1+n_2-1} \in \{0, 1\}$ if $x_1, \dots, x_{n_1+n_2-1}$ are members of $\{0, 1\}$. Hence, the new Boolean projection and the initial Boolean projection belong to the same clone. \square

The set of Boolean functions is countable. We can suppose

that any set of clones is not continual if these clones have Boolean projections and $k > 2$. But this is not proved now.

2.7. Dual and M-Al Operators

The operators are used to build clones of functions.

Definition A unary operator O over functions is m -al if using the operator m times, we leave any function unchanged. We call a function f_1 m -al to f_2 , if $f_1 = Of_2$ and O is m -al. We call a set X_1 of functions m -al to a set X_2 of functions if every function of X_1 is m -al to a function of X_2 .

Dual operators are particular cases of m -al operators. But now one calls any m -al operator “dual”. It is not well since m -al operators with different m have very different properties.

Some operators can be functions. For example, the identity function $f(x) = x$ is 1-al operator and the negation (in Post logic) $f(x) = x + 1$ is k -al operator. Here values of x are functions. We denote the operators by $s_i^m(x)$, where i means an ordinal number of m -al operator.

Operators s_i^m are not used since they do not preserve compositions of functions.

Indeed, we say that an operator O preserves compositions of functions if a function f is m -al to a function g and the clone generated by f is m -al to the clone generated by g .

The next lemma states that operators s_i^m do not preserve compositions of functions.

Lemma. Let a function f be m -al to a function g . Then it can be that the clone generated by f is not m -al to the clone generated by g .

Proof. Compositions of functions use operators of permutation and identification of variables in functions and the operator of substitution of the first variable into a function by a function. It is enough to use only the last operator for the proof.

Let functions f_1 and f_2 be m -al to functions g_1 and g_2 with respect to an operator s_i^m :

$$f_1(x_1, \dots, x_{n_1}) = s_i^m(g_1(x_1, \dots, x_{n_1})), \quad f_2(x_1, \dots, x_{n_2}) = s_i^m(g_2(x_1, \dots, x_{n_2})) \quad (1)$$

and let functions f_0 and g_0 be

$$f_0(x_1, \dots, x_{n_1+n_2-1}) = f_1(f_2(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1}) \quad (2)$$

$$g_0(x_1, \dots, x_{n_1+n_2-1}) = g_1(g_2(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1}) \quad (3)$$

We shall prove that it can be

$$f_0(x_1, \dots, x_{n_1+n_2-1}) \neq s_i^m(g_0(x_1, \dots, x_{n_1+n_2-1})) \quad (4)$$

Using (1) and (2), we have

$$f_0(x_1, \dots, x_{n_1+n_2-1}) = s_i^m(g_1(f_2(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})) \quad (5)$$

Using (3), we have

$$s_i^m(g_0(x_1, \dots, x_{n_1+n_2-1})) = s_i^m(g_1(g_2(x_1, \dots, x_{n_2}), x_{n_2+1}, \dots, x_{n_1+n_2-1})) \quad (6)$$

Comparing (5) and (6), we see that if $f_2 \neq g_2$ then (4) holds.

Some operators are not functions and can preserve compositions of functions. This property of the operators is named the duality (more correctly, m -ality) principle. These operators are not s_i^m but they use s_i^m (below we take s_i instead of s_i^m and we use superscripts as powers, for example, $s_i^{-1} = s_i^{k-1}$):

$$Of(x_1, \dots, x_n) = s_i^{-1}(f(s_i(x_1), \dots, s_i(x_n)))$$

Further we use only these operators to create clones since only they preserve compositions.

The functions

$$s_i = \begin{pmatrix} 0 & 1 & \dots & k-1 \\ b_1 & b_2 & \dots & b_{k-1} \end{pmatrix}$$

are used as two-ary relations

$$\rho_i = \left\{ \begin{matrix} 0 & 1 & \dots & k-1 \\ b_1 & b_2 & \dots & b_{k-1} \end{matrix} \right\},$$

and the relations generate clones, i.e., every s_i has corresponding clone. We will say conditionally that $\rho_i = s_i$.

There are m clones, one X_1 of them is m -al to an X_2 , where X_2 is m -al to an X_3 , and so on. Some clones can be equal. And there are m sets of the clones. All of the sets are isomorphic with respect to inclusions of clones. This means that we can use only one of the sets to build all inclusion diagrams of their clones. The dual and the k -al functions are presented more interest.

2.8. Dual Operators

A dual operator can be a function. Let the function be $s(x)$.

There are 3 functions in 3-valued logic:

$$s_1 = \begin{pmatrix} 012 \\ 021 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 012 \\ 102 \end{pmatrix}, \quad s_3 = \begin{pmatrix} 012 \\ 210 \end{pmatrix},$$

where s_1 is presented by cycles (0)(1,2), s_2 is presented by cycles (0,1)(2), and s_3 is presented by cycles (0,2)(1).

There are 9 functions in 4-valued logic:

$$s_1 = \begin{pmatrix} 0123 \\ 0132 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 0123 \\ 0213 \end{pmatrix}, \quad s_3 = \begin{pmatrix} 0123 \\ 0321 \end{pmatrix},$$

$$s_4 = \begin{pmatrix} 0123 \\ 1023 \end{pmatrix}, \quad s_5 = \begin{pmatrix} 0123 \\ 2103 \end{pmatrix}, \quad s_6 = \begin{pmatrix} 0123 \\ 3120 \end{pmatrix},$$

$$s_7 = \begin{pmatrix} 0123 \\ 2301 \end{pmatrix}, \quad s_8 = \begin{pmatrix} 0123 \\ 3210 \end{pmatrix}, \quad s_9 = \begin{pmatrix} 0123 \\ 1032 \end{pmatrix}.$$

where all s_i are presented by cycles of lengths 1 and 2 or of length 2 only. And relations generating clones are $\rho_i = s_i$. Clones generated by $\rho_1 - \rho_6$ are isomorphic with respect to compositions. And clones generated by $\rho_7 - \rho_9$ are isomorphic too.

The next characteristics of the operators exist for all k :

- $s_i^{-1}(x) = s_i(x)$,
- relations ρ_i generating clones equal s_i ,
- the functions $s_i(x)$ are presented by cycles of lengths 1 and 2 or of length 2 only, and inverse is true: 1-ary functions with the cycles are $s_i(x)$,
- if some s_i have cycles only of length 2 then the number of the s_i is $1*3*5*...*(k-1)$ and then clones generated by their ρ_i are maximal, but clones generated by the other ρ_i are not maximal ([1]),
- different ρ_i generate different clones, i.e., ρ_i generating the same clones are absent,
- if some s_i have the same number of cycles and the cycles have the same lengths, then their ρ_i generate isomorphic clones,

◦ the number of s_i is $\binom{k}{2} + \binom{k}{2} \binom{k-2}{2} / 2! + \binom{k}{2} \binom{k-2}{2} \binom{k-4}{2} / 3! + \dots$, where $\binom{m_1}{m_2} = 0$ if $m_1 < m_2$.

The next theorem proves the last property.

Theorem. The number of functions $s_i(x)$ in k -valued logic is

$$\binom{k}{2} + \binom{k}{2} \binom{k-2}{2} / 2! + \binom{k}{2} \binom{k-2}{2} \binom{k-4}{2} / 3! + \dots$$

Proof. The number of 1-ary functions with one cycle (i_1, i_2) of length 2 (all other cycles have length 1) is $\binom{k}{2}$. The number of 1-ary functions with two cycles $(i_1, i_2)(i_3, i_4)$ ($i_1, i_2 \neq i_3, i_4$) is $\binom{k}{2} \binom{k-2}{2}$. But cycles $(i_1, i_2)(i_3, i_4)$ and $(i_3, i_4)(i_1, i_2)$ are the same. The number of 1-ary functions with 3 cycles $(i_1, i_2)(i_3, i_4)(i_5, i_6)$ is $\binom{k}{2} \binom{k-2}{2} \binom{k-4}{2}$. But there are $3!$ permutations of the pairs. And so on. \square

2.9. Self-Dual Functions

We will call clones of self-dual (auto-dual) functions briefly *clones*. We will prove that the number of clones is infinite if $k \geq 3$. But this number is finite if $k = 2$ [7].

Definition We call a function self-dual with respect to a dual operator if the function is dual to itself, i.e., it remains unchanged after one application of the dual operator.

A clone of self-dual functions can be presented as $\text{Pol}(\rho_i)$, where $\rho_i = s_i$. So the number of clones generated by the relations equals the number of $s_i(x)$.

The other property is given by the next theorem.

Theorem. Clones of self-dual functions satisfy conditions a_i^μ at $k \geq 3$ for any finite μ and for any l not presented in cycles of length 1.

Proof. It is enough to prove the theorem in 3-valued logic because in k -valued logic the proof is analogous.

There are 3 dual operators and 3 $\text{Pol}(\rho_i)$. Corresponding clones are isomorphic and we use only the relation

$$\rho = \left\{ \begin{array}{ccc} 0 & 1 & 2 \\ 0 & 2 & 1 \end{array} \right\}. \text{ The relation equals } s_1.$$

Let $f(x_1, \dots, x_n)$ be functions of $n \geq 3$. A function of some n is presented in table 1.

Table 1. Values of function $f(x_1, \dots, x_n)$. Absent values are 0.

x_1	x_2	x_3	...	x_n	f
2	1	1	...	1	1
1	2	1	...	1	1
1	1	2	...	1	1
.
1	1	1	...	2	1
1	2	2	...	2	2
2	1	2	...	2	2
2	2	1	...	2	2
.
2	2	2	...	1	2

We must prove the next characteristics:

- the functions preserve the relation $\rho = \left\{ \begin{array}{ccc} 0 & 1 & 2 \\ 0 & 2 & 1 \end{array} \right\}$.
- the functions satisfy the conditions $\{a_0^0, a_1^{n-1}, a_2^{n-1}\}$,
- the functions of different n generate different clones. Indeed, the function preserves ρ because
 - $\left\{ \begin{array}{l} f(1, \dots, 1, 2, 1, \dots, 1) = 1 \\ f(2, \dots, 2, 1, 2, \dots, 2) = 2 \end{array} \right\}$,
 - $\left\{ \begin{array}{l} f(2, \dots, 2, 1, 2, \dots, 2) = 2 \\ f(1, \dots, 1, 2, 1, \dots, 1) = 1 \end{array} \right\}$,
 - the other pairs of lines contain the pair of function's values $\left\{ \begin{array}{l} 0 \\ 0 \end{array} \right\}$ or they are absent because they contain the pair $\left\{ \begin{array}{l} 1 \\ 1 \end{array} \right\}$ or the pair $\left\{ \begin{array}{l} 2 \\ 2 \end{array} \right\}$ of values of variables.

The functions satisfy the condition a_0^0 because $f(1, \dots, 1) = 0$. They satisfy conditions a_1^{n-1} and a_2^{n-1}

because every $n-1$ lines have a column with values 1 whenever $f=1$ and every $n-1$ lines have a column with values 2 whenever $f=2$.

By theorem in 2.5, the function generates different clones for different n .

Hence, the number of the clones is countable. \square

2.10. On K -al Operators

There are 2 functions to be k -al operators in 3-valued logic:

$$s_1 = \begin{pmatrix} 012 \\ 120 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 012 \\ 201 \end{pmatrix}$$

Relations $\rho_1 = s_1$ and $\rho_2 = s_2$ generate the same clones.

There are 6 functions to be k -al operators in 4-valued logic:

$$s_1 = \begin{pmatrix} 0123 \\ 1230 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 0123 \\ 1302 \end{pmatrix}, \quad s_3 = \begin{pmatrix} 0123 \\ 2031 \end{pmatrix}$$

$$s_4 = \begin{pmatrix} 0123 \\ 2310 \end{pmatrix}, \quad s_5 = \begin{pmatrix} 0123 \\ 3012 \end{pmatrix}, \quad s_6 = \begin{pmatrix} 0123 \\ 3201 \end{pmatrix}$$

Every pair of relations $(\rho_1 = s_1, \rho_5 = s_5)$, $(\rho_2 = s_2, \rho_3 = s_3)$, or $(\rho_4 = s_4, \rho_6 = s_6)$ generates the same clones. Clones generated by these pairs are isomorphic with respect to compositions.

All operators s_i of any k -valued logic have the next characteristics:

- $s_i^{-1}(x) = s_i^{k-1}(x)$;
- relations ρ_i generating clones equal s_i ,
- clones generated by any ρ_i are not maximal,
- if some s_i have the same number of cycles, the cycles have the same lengths, and the ρ_i generate different clones then the ρ_i generate isomorphic clones,
- if $k = 2^m$ then $k/2$ of ρ_i generate the same clones;
- if $k = p^m \wedge p \geq 3$, where p is prime number and $m \geq 1$, then all ρ_i generate the same clones,
- if $k = p_1^{m_1} * \dots * p_j^{m_j} \wedge j \geq 2$ then all ρ_i generate different clones,
- there are $(k-1)!$ functions s_i with one cycle,
- if $k = p^m$ then every s_i has one cycle,
- if $k = p_1^{m_1} \dots p_l^{m_l}$ then we have s_i with one cycle and s_i with several cycles such that l_j of the cycles have length $p_j^{m_j}$ and l_0 have length 1, this is true for all l_j and l_0 which defined by $l_0 + l_1 * p_1^{m_1} + \dots + l_l * p_l^{m_l} = k$.

We shall prove three last properties.

Theorem. There are $(k-1)!$ functions s_i with one cycle.

Proof. Let s_i have one cycle. Then

$$s_i = \begin{pmatrix} 0 & b_1 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & \dots & b_{k-1} & 0 \end{pmatrix}$$

where the first row is a sequence $(0, b_1, b_2, \dots, b_{k-1})$, all b_i are mutually unequal, and all b_i do not equal 0. Then b_1 can have $k-1$ different values, b_2 can have $k-2$ different values, ..., b_{k-1} has only one value. Hence, there are $(k-1)!$ different values and $(k-1)!$ different s_i . \square

Theorem If $k = p^m$ then every s_i has one cycle.

Proof by contradiction. We must prove that $s_i^m = s_i$ only if $m \neq k$.

Let s_i have 2 cycles of lengths m_1 and m_2 ($m_1 + m_2 \leq k$) and let the other cycles have length 1:

$$s_i = \begin{pmatrix} 0 & b_1 & \dots & b_{m_1-2} & b_{m_1-1} & b_{m_1} & \dots & b_{m_1+m_2-2} & b_{m_1+m_2-1} & \dots \\ b_1 & b_2 & \dots & b_{m_1-1} & 0 & b_{m_1}+1 & \dots & b_{m_1+m_2-1} & b_{m_1} & \dots \end{pmatrix}$$

The first m_1 components become without change after $s_i^{m_1}$. And the next m_2 component become without change after $s_i^{m_2}$. If $m_1 = m_2$ then $s_i = s_i^{m_1}$ but $m_1 < k$.

If m_1 and m_2 are relatively prime then $s_i \neq s_i^{m_1} \wedge s_i \neq s_i^{m_2}$ but $s_i = s_i^{m_1 * m_2}$ and $k \neq m_1 * m_2$.

If $m_1 \neq m_2$ and m_1 and m_2 are not relatively prime then $s_i = s_i^{m_1 * m_2 / m_0^2}$ where m_0 is the common deviser of m_1 and m_2 . But $m_1 * m_2 / m_0^2 \neq k$.

We will get the same result if s_i has 3 or more cycles. \square

Theorem. If $k = p_1^{m_1} \dots p_l^{m_l}$ then we have s_i with one cycle and s_i with several cycles such that l_j of the cycles have length $p_j^{m_j}$ and l_0 have length 1, this is true for all l_j and l_0 which defined by $l_0 + l_1 * p_1^{m_1} + \dots + l_l * p_l^{m_l} = k$.

Proof. If $s_i = \begin{pmatrix} 0 & b_1 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & \dots & b_{k-1} & 0 \end{pmatrix}$ then $s_i^k = s$, i.e., the s_i have one cycle of length k .

We have $s_i \neq s_i^{p_j^{m_j}}$ for $1 \leq j \leq l$ but $s_i = s_i^{p_1^{m_1} * \dots * p_l^{m_l}}$. Hence, the s_i have l_j cycles of length $p_j^{m_j}$ for all l_j and l_0 defined by $l_0 + l_1 * p_1^{m_1} + \dots + l_l * p_l^{m_l} = k$. \square

For example, if $k = 2 * 3$ then we have

$$s_1 = \begin{pmatrix} 012345 \\ 123450 \end{pmatrix}, \dots, s_{121} = \begin{pmatrix} 012345 \\ 021453 \end{pmatrix}, \dots$$

where $s_1 - s_{120}$ have one cycle of length 6, s_{121} and all

other s_i have 3 cycles of lengths 1, 2, 3.

If $k = 2^2 * 3$ then we have s_i with: (1) one cycle of length k ; (2) 7 cycles such that one of them has length 4, one has length 3, and 5 have length 1; (3) 4 cycles such that two of them have length 4, one has length 3, and one has length 1; (4) 5 cycles such that one of them has length 4, two have length 3 and two have length 1.

2.11. Self-K-AL Functions

The number of clones of self- k -al functions is finite at $k = 2$ (k -al and dual operators are the same if $k = 2$). But the number of the clones is not finite at $k > 2$.

Definition We call a function k -al to a function if the former function is obtained from the latter functions by using some k -al operator $k - 1$ times. We call a function self- k -al if the function equals its k -al function.

So a function f is self- k -al if $f(x_1, \dots, x_n) = s_i^{-1}(f(s_i(x_1), \dots, s_i(x_n)))$.

A clone of self- k -al functions can be presented as $\text{Pol}(\rho_i)$, where $\rho_i = s_i$.

The number of clones of self- k -al functions can be finite. We shall prove this but before we shall prove the next lemmas.

Lemma 1. If s_i has one cycle then the numbers of tuples l of any self- k -al function preserving $\rho_i = s_i$ are the same for all l .

Proof. Let a self- k -al function f preserve the relation

$$\left\{ \begin{array}{cccccc} 0 & b_1 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & \dots & b_{k-1} & 0 \end{array} \right\}$$

Then every tuple 0 has corresponding tuple b_1 . Hence, the number of tuples 0 equals the number of tuples b_1 . Further, every tuple b_1 has corresponding tuple b_2 . Hence, the number of tuples b_1 equals the number of tuples b_2 . And so on. \square

Corollary. If s_i has one cycle then self- k -al functions generated by $\rho_i = s_i$ have all values.

Proof. If tuples l of a function are absent for some l then the function has not the value l . By the lemma, all values of the function are absent. But such the function does not exist. \square

Lemma 2. Let s_i have one cycle and let f be a self- k -al function preserving $\rho_i = s_i$. Then values of $f(0, x_2, \dots, x_n)$ are arbitrary but values of $f(\bar{0}, x_2, \dots, x_n)$ depend on values of $f(0, x_2, \dots, x_n)$.

Proof. Let

$$\rho = \left\{ \begin{array}{cccccc} 0 & b_1 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & \dots & b_{k-1} & 0 \end{array} \right\}$$

then

$$f(b_1, b_{j_1}, b_{j_2}, \dots, b_{j_{k-1}}) = f(0, b_{j_1-1}, b_{j_2-1}, \dots, b_{j_{k-1}-1})$$

$$f(b_2, b_{j_1}, b_{j_2}, \dots, b_{j_{k-1}}) = f(0, b_{j_1-2}, b_{j_2-2}, \dots, b_{j_{k-1}-2})$$

and so on. \square

Lemma 3. If s_i has one cycle then self- k -al functions preserving $\rho_i = s_i$ belong only to α and δ families.

Proof. Let

$$\rho_i = \begin{pmatrix} 0 & b_1 & b_2 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & b_3 & \dots & b_{k-1} & 0 \end{pmatrix}$$

Any self- k -al function f has $f(0, \dots, 0) = 0$ or $f(0, \dots, 0) = \bar{0}$.

If $f(0, \dots, 0) = 0$ then the function is α . Indeed, by the first column in ρ_i , $f(0, \dots, 0) = 0$ implies $f(b_1, \dots, b_1) = b_1$. By the second column in ρ_i , $f(b_1, \dots, b_1) = b_1$ implies $f(b_2, \dots, b_2) = b_2$. And so on. But every b_i runs through all values.

If $f(0, \dots, 0) = \bar{0}$ then the function is δ since $f(0, \dots, 0) = \bar{0}$ implies $f(b_1, \dots, b_1) = \bar{b}_1$, $f(b_1, \dots, b_1) = \bar{b}_1$ implies $f(b_2, \dots, b_2) = \bar{b}_2$, and so on.

The same families of functions are generated by compositions.

Indeed, the operations of permutations and identification of variables do not change families. The operation of substitution does not change α families. But the operation can change δ families.

Let f be a δ function. Then $f(f(0, \dots, 0), 0, \dots, 0) = 0$ or $f(f(0, \dots, 0), 0, \dots, 0) = \bar{0}$. If $f(f(0, \dots, 0), 0, \dots, 0) = 0$ then the function is α else the functions is δ . \square

Theorem. If s_i has one cycle then all self- k -al functions generated by $\rho_i = s_i$ satisfy the same conditions for all l .

These conditions are a_l^0 , a_l^1 , or $a_l^{\omega+k^{n-1}}$.

Proof. The proof consists of 2 parts.

• We will prove that all self- k -al functions satisfy the same conditions for all l .

Let a self- k -al function preserve a relation $\left\{ \begin{array}{cccccc} 0 & 1 & \dots & k-2 & k-1 \\ 1 & 2 & \dots & k-1 & 0 \end{array} \right\}$ and let the function satisfy a condition a_l^μ for some l .

Then we have the next sub-tuples l after permuting columns:

$$\begin{array}{cccccc} b_1 & l & l & \dots & l \\ l & b_2 & l & \dots & l \\ l & l & b_3 & \dots & l \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ l & l & l & \dots & b_{\mu+1} \end{array}$$

where b_i does not equal l .

Using the relation, we have

$$\begin{matrix} b_1+1 & l+1 & l+1 & \dots & l+1 \\ l+1 & b_2+1 & l+1 & \dots & l+1 \\ l+1 & l+1 & b_3+1 & \dots & l+1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l+1 & l+1 & l+1 & \dots & b_{\mu+1}+1 \end{matrix}$$

where b_i+1 does not equal $l+1$.

Hence, the function satisfies the same condition for $l+1$ and for other l too.

We have the same result if we use the other relations for self- k -al functions, because every tuple i_1 has corresponding tuple i_2 .

• We will prove that any self- k -al functions f^n can satisfy only conditions a_i^μ with $\mu=0$, with $\mu=1$, or with $\mu=\omega+k^{n-1}$. If all tuples o have a column with values 0 only then $\mu=\omega+k^{n-1}$ since the number of the tuples o is k^{n-1} . Let $\mu < \omega$.

A table of any self- k -al function has a line $(b_1, \dots, b_n; 0)$ for some b_i . The table has also a line $(b'_1, \dots, b'_n; l)$ with all b'_i such that $b'_i = l \wedge b_i \neq 0$ and with all b'_i such that $b'_i \neq l \wedge b_i = 0$.

Let $l=0$. If all $b_i \neq 0$ or all $b'_j \neq 0$ then $\mu=0$ otherwise $\mu=1$ since all $b_i = b'_i = 0$ are absent, i.e., a column with 2 zeros is absent but a column with 1 zero exists.

Let $l \neq 0$. Then there is an l_0 such that $l+l_0=0$. And there is a line $(b'_1+l_0, \dots, b'_n+l_0; 0)$ with $b'_i+l_0=0$ if $b'_i \neq b_i$ and with $b'_i+l_0 \neq 0$ if $b'_i = b_i$. Then $\mu=0$ or $\mu=1$.

The theorem is proved. □

We have used the relation ρ_1 . But we have the same result for relations

$$\left\{ \begin{matrix} 0 & b_1 & \dots & b_{k-2} & b_{k-1} \\ b_1 & b_2 & \dots & b_{k-1} & 0 \end{matrix} \right\}$$

Lemma. A function has a Boolean projection if $\mu \neq 0$.

Proof. Let $\mu \neq 0$. Then a value of $f(x_1, \dots, x_n)$ equals some of the x_i for every function since otherwise $\mu=0$. Hence, if $x_i \in \{0,1\}$ then $f(x_1, \dots, x_n) \in \{0,1\}$. □

3. Numerical Results in 3-Valued Logic

3.1. Classification of Self-Dual Functions

Any classification must use disjoint classes. But any class belongs only to one clone. Therefore classification with classes and classification with clones are the same.

We call clones of self-dual functions briefly clones.

We must give the graph of inclusions of clones. The graph begins at the clone containing all other clones. Then the graph has 3 branches because there are 3 dual operators:

$$s_i(f(s_i(x_1), \dots, s_i(x_n)))$$

where

$$s_1 = \begin{pmatrix} 012 \\ 021 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 012 \\ 102 \end{pmatrix}, \quad s_3 = \begin{pmatrix} 012 \\ 210 \end{pmatrix}.$$

and $s_i^{-1} = s_i$.

The branches are isomorphic with respect to inclusions. Therefore we can give the graph of only one branch created by s_1 . And we call clones created by s_1 just clones.

The branch begins with the clone containing all the other clones. The clone is the set of functions preserving the relation

$$\rho_1 = \left\{ \begin{matrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{matrix} \right\}.$$

The inclusion graph is presented in figure 1.

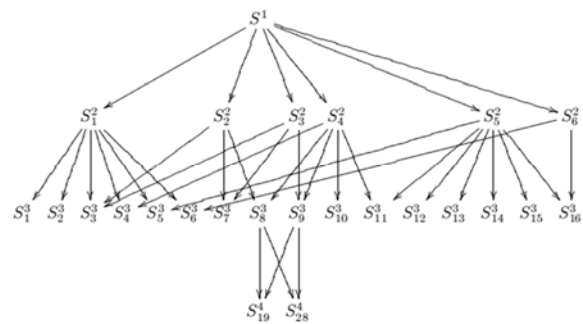


Figure 1. Three levels of inclusion graph of self-dual clones, fourth level has 39 points and web of edges. Some points (19 and 28) of fourth level form non-lattice part of graph.

3.2. Self-Dual Clones and Their Preserving Relations

We give the set of clones and their preserved relations.

$$S = \text{Pol} \left\{ \begin{matrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{matrix} \right\}$$

$$S^2_1 = \text{Pol} \{S, \{0\}\}$$

$$S^2_2 = \text{Pol} \{S, \{0\ 1\}\}$$

$$S^2_3 = \text{Pol} \{S, \{0\ 2\}\}$$

$$S^2_4 = \text{Pol} \{S, \{1\}\}$$

$$S^2_5 = \text{Pol} \{S, \{1\ 2\}\}$$

$$S^2_6 = \text{Pol} \left\{ S, \begin{matrix} 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 \end{matrix} \right\}$$

$$S_1^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 & 0 & 0 & 1 & 2 \\ 0 & 1 & 2 & 0 & 0 \end{Bmatrix}\}$$

$$S_2^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{Bmatrix}\}$$

$$S_3^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 & 0 \\ 0 & 1 \end{Bmatrix}\}, \text{Pol}\{S_2^2, \{0\}\}, \text{Pol}\{S_3^2, \{0\}\}$$

$$S_4^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}\}, \text{Pol}\{S_4^2, \{0\}\}$$

$$S_5^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 & 0 \\ 1 & 2 \end{Bmatrix}\}, \text{Pol}\{S_5^2, \{0\}\}$$

$$S_6^3 = \text{Pol}\{S_1^2, \begin{Bmatrix} 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 \end{Bmatrix}\}, \text{Pol}\{S_6^2, \{0\}\}$$

$$S_7^3 = \text{Pol}\{S_2^2, \begin{Bmatrix} 1 & 1 \\ 1 & 2 \end{Bmatrix}\}, \text{Pol}\{S_3^2, \begin{Bmatrix} 1 & 1 \\ 1 & 2 \end{Bmatrix}\}$$

$$S_8^3 = \text{Pol}\{S_2^2, \begin{Bmatrix} 2 \\ 1 \end{Bmatrix}\}, \text{Pol}\{S_3^2, \{0 \ 1\}\}$$

$$S_9^3 = \text{Pol}\{S_3^2, \{1\}\}, \text{Pol}\{S_4^2, \{0 \ 2\}\}$$

$$S_{10}^3 = \text{Pol}\{S_4^2, \begin{Bmatrix} 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{Bmatrix}\}$$

$$S_{11}^3 = \text{Pol}\{S_4^2, \begin{Bmatrix} 1 & 1 \\ 1 & 2 \end{Bmatrix}\}, \text{Pol}\{S_5^2, \{1\}\}$$

$$S_{12}^3 = \text{Pol}\{S_5^2, \begin{Bmatrix} 0 & 0 & 0 & 1 & 2 \\ 0 & 1 & 2 & 0 & 0 \end{Bmatrix}\}$$

$$S_{13}^3 = \text{Pol}\{S_5^2, \begin{Bmatrix} 0 & 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 1 & 2 \end{Bmatrix}\}$$

$$S_{14}^3 = \text{Pol}\{S_5^2, \begin{Bmatrix} 0 & 0 & 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 1 & 2 \end{Bmatrix}\}$$

$$S_{15}^3 = \text{Pol}\{S_5^2, \begin{Bmatrix} 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 \end{Bmatrix}\}, \text{Pol}\{S_6^2, \{1 \ 2\}\}$$

$$S_{16}^3 = \text{Pol}\{S_5^2, \begin{Bmatrix} 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{Bmatrix}\}$$

3.3. Classification of Self-3-AI Functions

The classification is presented in figure 2 by an inclusion graph.

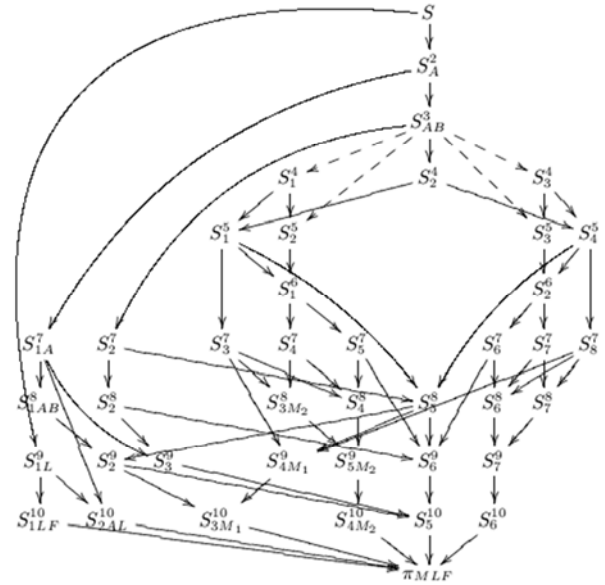


Figure 2. Inclusion graph of clones of self-3-ai functions. Subscripts: SA means clones of α -functions, SB means clones with Boolean projections (all chains of SS_3 have clones of α -functions with Boolean projections, implicitly), MS means clones of monotonic functions with all types of order, SM_1 means clones of monotonic functions with order $S1 < 2 < 0$, SM_2 means clones of monotonic functions with order $S2 < 0 < 1$, SL means clones of linear functions. Dashed arrows mean infinite chains of clones, SS^3 has Boolean projection SC_4 (in Post's notation), SS^4_1 and SS^4_2 have Boolean projections SF^{∞}_1 and SF^{∞}_5 respectively; SS^5_1 and SS^5_3 have Boolean projections SF^{∞}_2 and SF^{∞}_6 respectively; π is clone of projective (selective) functions.

The created graph is not a lattice as it follows from figure 3.



Figure 3. Non-lattice parts of inclusion graph.

3.4. Clones of Functions Preserving Relations

We give clones and relations preserved by functions of the clones.

$$S = \text{Pol}\left\{\begin{Bmatrix} 0 & 1 & 2 \\ 1 & 2 & 0 \end{Bmatrix}\right\}$$

$$S^2 = \text{Pol}\{S, \{0\}\}$$

$$S^3 = \text{Pol}\left\{S^2, \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}\right\}$$

$$S_1^4 = \text{Pol}\left\{S^3, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\right\}$$

$$S_2^4 = \text{Pol}\{S^3, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}\}$$

$$S_8^7 = \text{Pol}\{S^2, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$S_3^4 = \text{Pol}\{S^3, \begin{Bmatrix} 0 & 0 & 2 \\ 0 & 1 & 1 \end{Bmatrix}\}$$

$$S_1^8 = \text{Pol}\{S_1^7, \begin{Bmatrix} 0 & 0 \\ 0 & 1 \end{Bmatrix}\}$$

$$S_1^5 = \text{Pol}\{S_1^4, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}\} = \text{Pol}\{S_2^4, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_2^8 = \text{Pol}\{S_2^7, \begin{Bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_2^5 = \text{Pol}\{S^3 \vee S_1^4, \begin{Bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_3^8 = \text{Pol}\{S_3^7 \vee S_4^7, \begin{Bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\}$$

$$S_3^5 = \text{Pol}\{S^3, \begin{Bmatrix} 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_2^4, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_4^8 = \text{Pol}\{S_3^7 \vee S_4^7, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_5^7, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$S_4^5 = \text{Pol}\{S_2^4, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\} = \text{Pol}\{S_3^4, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}\}$$

$$S_5^8 = \text{Pol}\{S_1^5 \vee S_4^5, \begin{Bmatrix} 0 & 1 \\ 0 & 2 \end{Bmatrix}\} = \text{Pol}\{S_2^7, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_1^6 = \text{Pol}\{S_1^5, \begin{Bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{Bmatrix}\} = \text{Pol}\{S_2^5, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}\}$$

$$S_6^8 = \text{Pol}\{S_5^7, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\} = \text{Pol}\{S_7^7 \vee S_8^7, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 0 \end{Bmatrix}\}$$

$$S_2^6 = \text{Pol}\{S_3^5, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}\} = \text{Pol}\{S_4^5, \begin{Bmatrix} 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{Bmatrix}\}$$

$$S_7^8 = \text{Pol}\{S_7^7 \vee S_8^7, \begin{Bmatrix} 0 & 0 & 0 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\}$$

$$S_1^7 = \text{Pol}\{S^2, \begin{Bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{Bmatrix}\}$$

$$S_1^9 = \text{Pol}\{S, \begin{Bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\ 0 & 2 & 1 & 2 & 1 & 0 & 1 & 0 & 2 \end{Bmatrix}\}$$

$$S_2^7 = \text{Pol}\{S^3, \begin{Bmatrix} 0 & 1 \\ 0 & 2 \end{Bmatrix}\}$$

$$S_2^9 = \text{Pol}\{S_1^8, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\} = \text{Pol}\{S_5^8, \begin{Bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{Bmatrix}\}$$

$$S_3^7 = \text{Pol}\{S_1^5, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$S_3^9 = \text{Pol}\{S_1^7, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\} = \text{Pol}\{S_2^8, \begin{Bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{Bmatrix}\}$$

$$S_4^7 = \text{Pol}\{S_1^6, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$S_4^9 = \text{Pol}\{S_3^7 \vee S_8^7, \begin{Bmatrix} 0 & 1 \\ 0 & 2 \end{Bmatrix}\} = \text{Pol}\{S_5^8, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$S_5^7 = \text{Pol}\{S_1^6, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{Bmatrix}\}$$

$$S_5^9 = \text{Pol}\{S_3^8, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_4^8, \begin{Bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\}$$

$$S_6^7 = \text{Pol}\{S^2, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 0 \end{Bmatrix}\}$$

$$S_6^9 = \text{Pol}\{S_5^7, \begin{Bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_6^7, \begin{Bmatrix} 0 & 0 & 0 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\} =$$

$$S_7^7 = \text{Pol}\{S^2, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\}$$

$$= \text{Pol}\{S_2^8, \begin{Bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\} = \text{Pol}\{S_5^8, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{Bmatrix}\} =$$

$$S_7^9 = \text{Pol}\{S_6^8, \begin{Bmatrix} 0 & 0 & 0 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_7^8, \begin{Bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 0 \end{Bmatrix}\}$$

$$S_1^{10} = \text{Pol}\{S_1^9, \begin{Bmatrix} 0 & 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 1 & 2 & 0 & 2 \end{Bmatrix}\}$$

$$S_2^{10} = \text{Pol}\{S_1^7, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{Bmatrix}\} = \text{Pol}\{S_1^9, \{0\}\}$$

$$S_3^{10} = \text{Pol}\{S_2^9, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\} = \text{Pol}\{S_4^9, \begin{Bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{Bmatrix}\}$$

$$S_4^{10} = \text{Pol}\{S_5^9, \begin{Bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 2 & 0 & 1 \end{Bmatrix}\}$$

$$S_5^{10} = \text{Pol}\{S_2^9, \begin{Bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{Bmatrix}\} = \text{Pol}\{S_3^9, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{Bmatrix}\} =$$

$$= \text{Pol}\{S_6^9, \begin{Bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \end{Bmatrix}\}$$

$$S_6^{10} = \text{Pol}\{S_7^9, \begin{Bmatrix} 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 1 & 2 & 0 & 2 \end{Bmatrix}\}$$

$$\pi = \text{Pol}\{S_1^{10}, 0\} = \text{Pol}\{S_2^{10}, \begin{Bmatrix} 0 & 0 \\ 0 & 1 \end{Bmatrix}\} = \text{Pol}\{S_3^{10}, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 2 \end{Bmatrix}\} =$$

$$= \text{Pol}\{S_4^{10}, \begin{Bmatrix} 0 & 0 & 1 \\ 0 & 1 & 2 \end{Bmatrix}\} = \text{Pol}\{S_5^{10}, \begin{Bmatrix} 0 & 1 \\ 0 & 2 \end{Bmatrix}\} = \text{Pol}\{S_6^{10}, \begin{Bmatrix} 0 & 1 \\ 0 & 2 \end{Bmatrix}\}$$

4. Conclusion

The main problem of any theory, including the theory of multi-valued logic, is a classification of objects of this theory.

Partial classification of clones of functions in multi-valued logic was given in the works of S. Jablonski [2] and A. Mal'cev [8].

More detailed classification of clones was given by I. Rosenberg [9]. Top level of this classification contains a clone of all functions. The next level has 6 groups of clones, but the main group contains clones of self-m-al functions. This group was named *clones of self-dual functions*.

In the first time, we gave full classification of the group. Number of classes of the classification is $k-1$. We named them self-m-al at $2 \leq m \leq k$. We built the theory of self-k-al functions and proved 10 important theorems. We get numerical results for 3-valued logic and found that inclusion graphs of clones of self-dual and of self-3-al functions are not lattices.

References

- [1] Malkov M. A., Classifications of closed sets of functions in multi-valued logic, *SOP transactions on applied math.* 1:3, 96-105 (2014), <http://www.scipublish.com/journals/AM/papers/download/2102-970.pdf>.
- [2] Yablonskiy S. V.: Functional constructions in many-valued logics (Russian). *Tr. Mat. Inst. Steklova*, 515-142 (1958).
- [3] Lau D., *Functions algebra on finite sets*, Springer (2006).
- [4] Csákány B., All minimal clones on the three-element set, *Acta Cybernet.*, 6, 227-238 (1983).
- [5] Marchenkov S. S., Demetrovics J., Hannak L., On closed classes of self-dual functions in P3. (Russian) *Metody Diskrem. Anal.* 34, 38-73 (1980).
- [6] Machida H., On closed sets of three-valued monotone logical functions. In: *Colloquia Mathematica Societatis Janos Bolyai 28, Finite Algebra and multiple-valued logic*, Szeged (Hungary), 441-467 (1979).
- [7] Post E. L., *The two-valued iterative systems of mathematical logic*. Princeton Univ. Press, Princeton (1941).
- [8] Mal'cev A. I., *Iterative Post algebras*, NGU, Novosibirsk, (Russian) (1976).
- [9] Rosenberg, I. G., Über die Verschiedenheit maximaler Klassen in Pk. *Rev. Roumaine Math. Pures Appl.* 14, 431-438 (1969).
- [10] Malkov M. A. Classifications of Boolean Functions and Their Closed Sets, *SOP transactions on applied math.* 1:2, 172-193 (2014), <http://www.scipublish.com/journals/AM/papers/download/2102-522.pdf>.