A PLANAR GEOMETRICAL MODEL FOR REPRESENTING MULTIDIMENSIONAL DISCRETE SPACES AND MULTIPLE-VALUED LOGIC FUNCTIONS

by

Ryszard S. Michalski

Report No. 897, Department of Computer Science, University of Illinois,
Urbana, Illinois, January 1978.

Report No. UIUCDCS-R-78-897

UILU-ENG 78 1707

A PLANAR GEOMETRICAL MODEL FOR REPRESENTING MULTIDIMENSIONAL DISCRETE SPACES AND MULTIPLE-VALUED LOGIC FUNCTIONS

by

Ryszard S. Michalski

January 1978

Report No. UIUCDCS-R-78-897

A PLANAR GEOMETRICAL MODEL FOR REPRESENTING MULTIDIMENSIONAL DISCRETE SPACES AND MULTIPLE-VALUED LOGIC FUNCTIONS

by

Ryszard S. Michalski

January 1978

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

This work was supported in part by the National Science Founadation under grant NSF MCS 76-22940.

ABSTRACT

A general planar model of multidimensional discrete spaces (a diagram) is described which can be used for geometrically representing binary, multiple-valued, discrete and variable-valued logic functions. It is essentially a multiple-valued extension of Marquand's binary diagram, with an additional feature of varying thickness of lines representing axes of variables.

The diagram has been used extensively over the years by the author and his collaborators, and has proved to be useful for such tasks as: logic design and development of efficient algorithms for optimization of switching circuits of many variables, detection of symmetry in binary or multiple-valued logic functions, fast conversion of normal forms of switching functions to exclusive-or polynomial forms, design of algorithms for inductive inference and pattern recognition, optimization of decision tables and their conversion to optimal decision trees.

A method for recognizing in the diagram certain constructs (cartesian complexes), which are important for various applications, is presented and illustrated by two examples: one, involving a determination of a classification rule, and another, involving a synthesis of a variable-valued logic expression.

Key words and phrases: Logic diagram, Marquand diagram, Karnaugh map, Veitch diagram, discrete spaces, multiple-valued logic, variable-valued logic, logic design.

CR categories: 3.61, 3.63, 5.20, 5.21, 6.1

Motto: A picture is worth a thousand words. Especially when it is a right one.

1. INTRODUCTION

There exists a large number of problems in which there is a need to deal with functions of the general form:

$$f: D_1 \times D_2 \times \ldots \times D_n \rightarrow D \qquad (1)$$

where

$D_1, D_2, \ldots, D_n$ are finite non-empty sets

and

D is a finite or infinite non-empty set.

For example, a binary switching function is a special case of f when $D_1 = D_2 = \ldots = D_n = \{0,1\}$ and $D = \{0,1,*\}$, where * denotes a 'DON'T CARE' value. A k-valued switching function is a special case of f when $D_1 = D_2 = \ldots = D_n = \{0,1, \ldots, k-1\}$ and $D = \{0,1, \ldots, k-1,*\}$. Often used in operations research are the so-called psuedo-Boolean functions, which are a special case of f, when

$D_1 = D_2 = \ldots = D_n = \{0,1\}$ and $D = [0,1]$ (the closed interval).

When sets D_i and D are finite sets of integers, $\{0,1,2,\ldots\}$, then f represents a discrete function (e.g., Deschamps and Thayse [2]).

In pattern recognition and decision theory many problems can be phrased as a search for an efficient expression of a function f, where D_i are domains of values of certain features or descriptors which are used to characterize objects or situations, and D is either a finite set of decision classes to which the objects belong or a set of 'degrees of membership' of an object in a class (usually the closed interval [0,1]). Considered mainly from this point of view, functions f, in which D_i and D are arbitrary finite sets with or without any order, were studied by Michalski and his collaborators (e.g., [12]-[14]) under the name of variable-valued logic functions.

When analyzing a function f (which often is only partially defined), or designing algorithms involving such a function, it can be very useful to represent it in a form of a geometrical pattern.

The advantage of such representation, as compared to a symbolic representation, is that it is easier for humans to perceive in this form the global properties of the function, i.e., to "see" the function as a whole. Also, it is easier to detect various regularities in the function, if it is done by comparing geometrical configurations rather than strings of symbols or numbers. The above explains the popularity of Venn diagrams, Karnaugh maps, histograms or other graphical aids for representing functions.

This paper describes a general planar geometrical model (a diagram) for representing functions type (1) and gives a method for easily recognizing constructs in the diagram which are important for applications. The presented diagram has been extensively used over the years by the author and his collaborators, and has proved to be very useful for various tasks such as:

- logic design and a development of efficient algorithms for optimization of switching circuits of many variables (Michalski [8], Michalski and Kulpa [17])
- detection of symmetry in binary and multiple-valued logic functions (Michalski [9], Jensen [3])
- fast conversion of normal forms of switching functions to exclusive-or polynomial forms (Michalski [10])
- design of algorithms for inductive inference and pattern recognition (Michalski [12]-[15], Larson and Michalski [6])
- optimization of decision tables and their conversion to optimal decision trees (Michalski [16]).

Let us define $\mathbf{E}(d_1, d_2, \ldots, d_n)$, written also as $\mathbf{E}$, as the cartesian product of sets D_i, $i = 1, 2, \ldots n$:

$$\mathbf{E}(d_1, d_2, \ldots, d_n) = \mathbf{E} = D_1 \times D_2 \times \ldots \times D_n \qquad (2)$$

where d_i - the number of elements in D_i, and call it the <u>universe of events</u> (or <u>event space</u>). Let us assume, for simplicity, that the domains D_i are sets of positive integers:

$$D_i = \{0, 1, \ldots, d_i\}, \ i = 1, 2, \ldots, n \qquad (3)$$

where $d = d_i - 1$. This assumption causes no loss in generality because any finite set can be isomorphically mapped into D_i given by (3). Let us also assume that $D_1, D_2, \ldots, D_n$ are value sets (domains) of certain variables $x_1, x_2, \ldots, x_n$, respectively (i.e., x_1 can take values only from D_1, x_2 - only from D_2, and so on).

A discrete-Euclidean representation of the space **E** would be in the form of an n-dimensional 'grid', spanned from the $d_1, d_2, \ldots, d_n$ points on axes associated with variables $x_1, x_2, \ldots, x_n$, respectively (Fig. 1). The above geometrical model of **E** is, however, difficult to visualize when $n > 3$, and therefore it is not practical for a larger number of variables.

In the past, for the case when x_i are binary variables, many different planar representations of the space **E** have been proposed. Among well known early constructions are Euler's circles (1768) [1] and Venn diagrams (1880) [20]. The earliest known constructions for representing logical operations were developed in XIII century by Raymond Lull [1].

A diagram of rectangular shape, which is divided into cells corresponding to single combinations of binary logic values was first proposed by Marquand in 1881 [7]. The regularity and simplicity of such a diagram made it useful for a larger number of variables than the Venn diagram. It has not become popular, however, because there was no pressing need at that time for representing complex logical functions. Such a need arose many years later when Shannon discovered the applicability of Boolean algebra for describing switching systems. And then a Marquand-type diagram was independently rediscovered by Veitch in 1952 [19] (a diagram of this type was also developed by Svoboda [18]). Soon afterward in 1953, Karnaugh [4] reorganized the Veitch diagram, assigning variable values to rows and columns according to the Grey's code rather than to the regular binary code, used in Marquand and Veitch diagrams. In such an arrangement, any two neighboring cells in the diagram

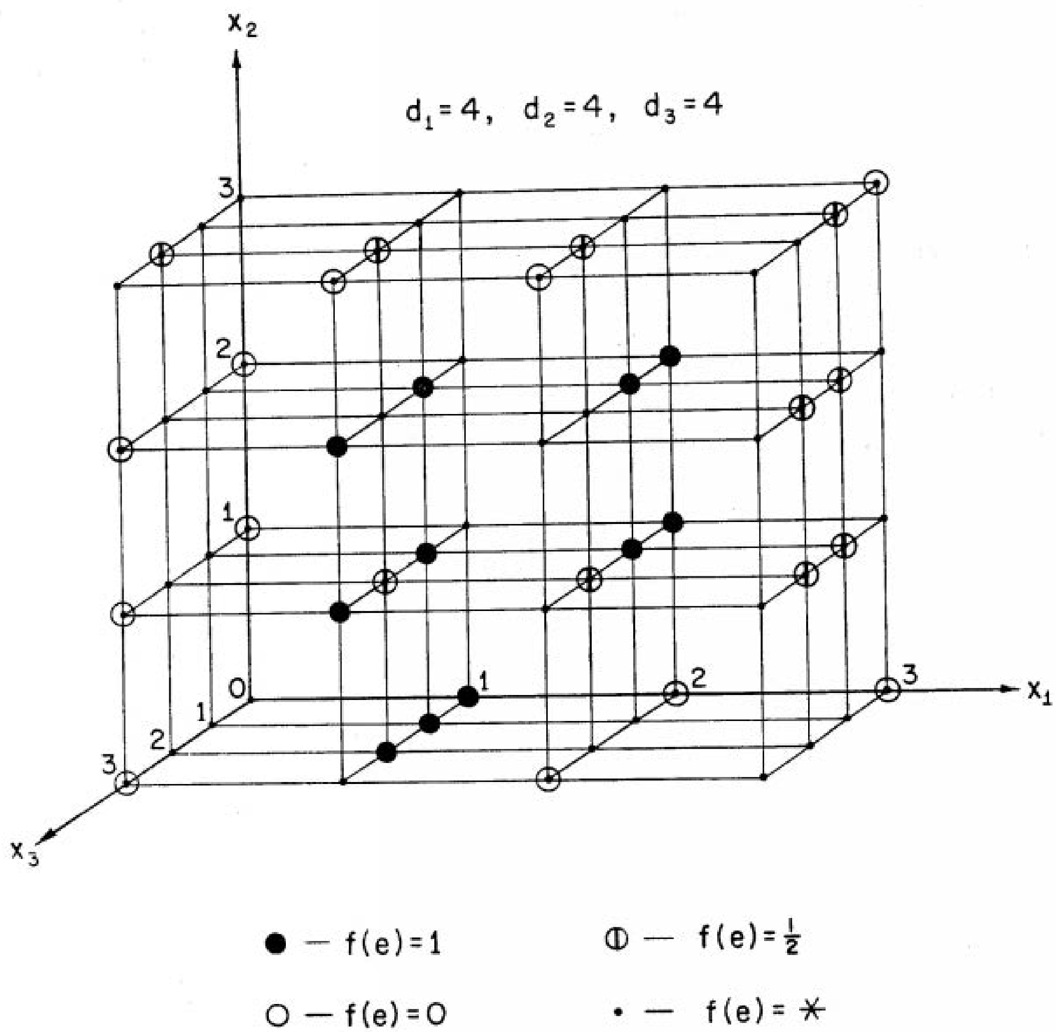


Fig. 1. Discrete-Euclidean representation of the space $\mathbb{E}(4,4,4)$ and a mapping $f\colon \mathbb{E}(4,4,4) \to \{0, \tfrac{1}{2}, 1, *\}$.

correspond to adjacent conjunctions (i.e., conjunctions which differ only in one literal (variable or its negation) and, therefore, can be reduced to one conjunction). In this form, the diagram (called the Karnaugh's map) became very popular as an aid in logical design of electronic circuitry. It is quite convenient for minimization of Boolean functions up to 4 variables. When there are more than 4 variables, however, the rules of using the map change and quickly become rather complicated.

This paper describes a diagram for representing discrete spaces spanned over not only binary variables, but variables with any discrete values. The rules for constructing and using the diagram are the same for any number of variables and any number of values which variables can take. Consequently, the diagram provides a general geometrical model of discrete spaces.

The diagram is essentially a multiple-valued extension of Marquand's binary diagram (although the author was not aware of this when he developed it). In addition, it has a new feature which is a varying thickness of lines representing axes of variables. This feature greatly improves the clarity of the diagram, especially when there are many variables. The diagram was originally described in a departmental report by Michalski [11] which is out of print. The binary form of this diagram was described earlier (Michalski [8]). The intention of this paper is to make available written information about the diagram and describe various new results not yet published.

2. CONSTRUCTION OF THE DIAGRAM

Suppose a task is to represent the space $\mathbf{E}(d_1, d_2, \ldots, d_n)$, that is a space spanned over n variables, which take $d_1, d_2, \ldots, d_n$ values, respectively. First, determine v which is the maximal number satisfying relation:

$$d_1 \cdot d_2 \cdots d_v \leq d_{v+1} \cdot d_{v+2} \cdots d_n \quad (4)$$

This is an arbitrary step, but it leads to a unique diagram for a given **E**. Also, the diagram is visually more satisfying if the products on both sides of (4) are approximately equal. (To achieve this a permutation of the d_i -s may be helpful.) Next, draw an arbitrary rectangle and divide it into $d_1 \cdot d_2 \ldots d_v$ rows and $d_{v+1} \cdot d_{v+2} \ldots d_n$ columns according to the following rules:

(i) In the first step, divide the rectangle by horizontal lines into d_1 rows and assign to the rows values 0, 1, ..., $\not{d}_1$ (values of variable x_1) in order from the top to the bottom. In the step i, $1 < i \leq v$, divide each row generated in step i-1 into d_i rows, and assign to them values 0, 1, ..., $\not{d}_i$, in order from the top to the bottom ($\not{d}_i = d_i - 1$).

(ii) Do the steps v+1, v+2, ..., n, in the similar way as above, but divide the rectangle by vertical lines into columns.

(iii) The lines generated in step i, i = 1, 2, ..., n, are called <u>axes of x_i</u>. Vary the thickness of axes of different variables: the thinnest should be axes of x_v and x_n, next thinnest - the axes of x_{v-1} and x_{n-1}, and so on, until exhausting all the axes. (Thus, if n is even and v = n/2, the thickest will be the axes x_1 and x_{v+1}).

Figure 2 presents a general form of the diagram for n = 4. A unique vector $(x_1, x_2, \ldots, x_v)$ corresponds to each row and a unique vector $(x_{v+1}, x_{v+2}, \ldots, x_n)$ corresponds to each column of the diagram. The intersection of any row with any column is called a <u>cell</u>. We will assume that the

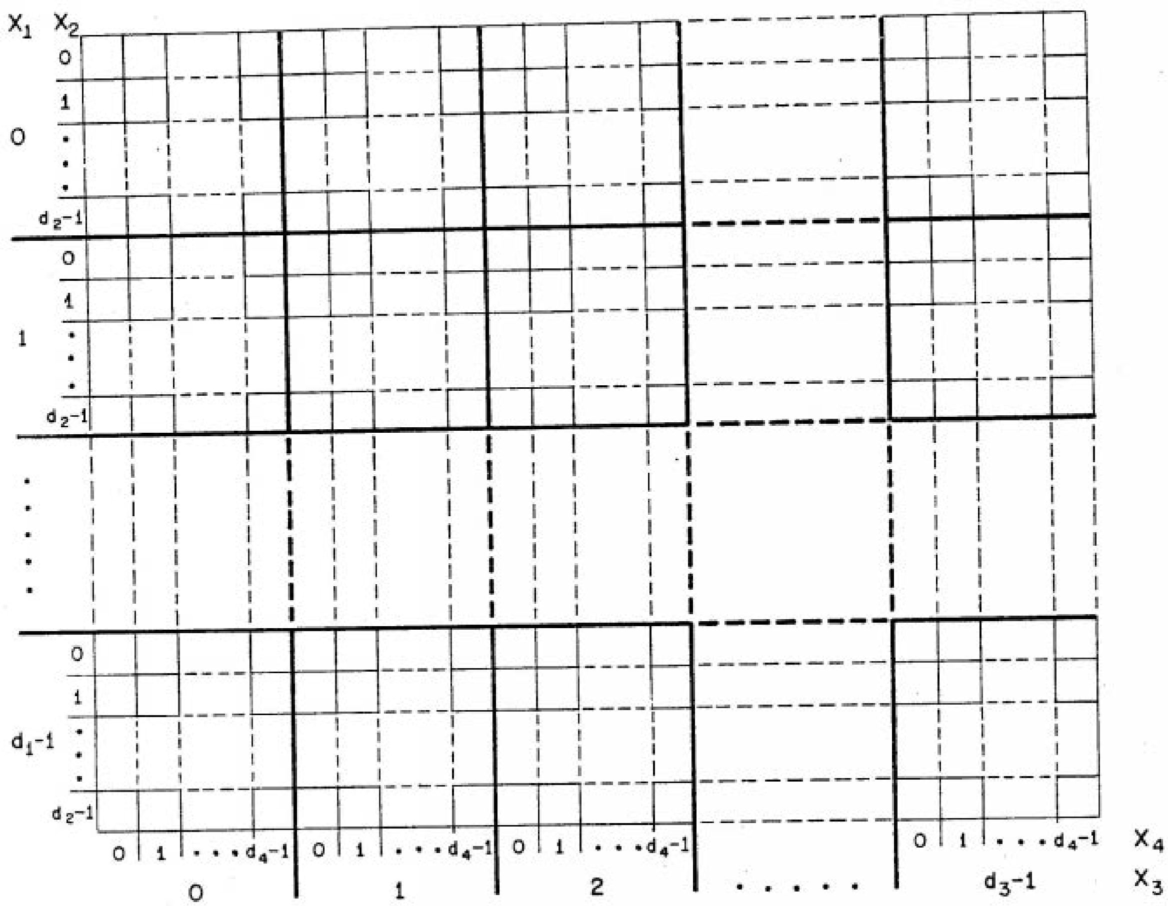


Fig. 2. A diagram representing $\mathbf{E}(d_1,d_2,d_3,d_4)$.

cells do not include points belonging to any axis nor to the perimeter of the rectangle. Each cell represents an element of **E** (<u>event</u>) determined by concatenating vectors $(x_1, x_2, \ldots, x_v)$ and $(x_{v+1}, x_{v+2}, \ldots, x_n)$, corresponding to the row and the column, respectively, whose intersection produces the given cell. The diagram comprises $d=d_1 \cdot d_2 \ldots d_n$ cells (that is the number of events in **E**).

Each event $e = (x_1, x_2, \ldots, x_n)$ from **E** can be assigned a unique number $\gamma(e)$ according to the formula:

$$\gamma(e) = x_n + \sum_{i=n-1}^{1} x_i \prod_{k=n}^{i+1} d_k \tag{5}$$

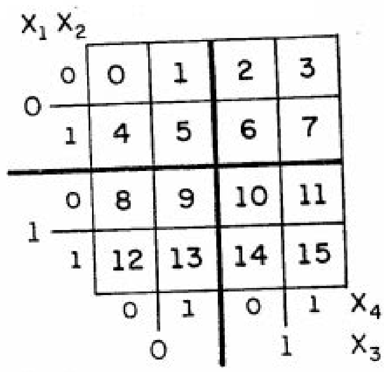
For example, in the space **E**(5,6,4,3) the value of the function $\gamma(e)$ (γ-<u>number</u>) for the event $e = (3,4,1,2)$ is

$$\gamma(e) = 2 + 1 \cdot 3 + 4 \cdot 4 \cdot 3 + 3 \cdot 6 \cdot 4 \cdot 3 = 269$$

It is easy to see that from a given value $\gamma(e)$ and values $d_1, d_2, \ldots, d_n$, one can determine the corresponding event $e=(x_1, x_2, \ldots, x_n)$. Namely, first divide $\gamma(e)$ by d_n; the remainder is the value of x_n. Next, divide the result by d_{n-1}; the remainder is the value of x_{n-1}, and so on, until the value of x_1 is obtained. It can be easily verified that if one assigns the corresponding γ-number to each cell, the order of the numbers in any diagram will be lexicographical (i.e., from left to right and from top to bottom), no matter what is the number of variables or the number of values the variables take.

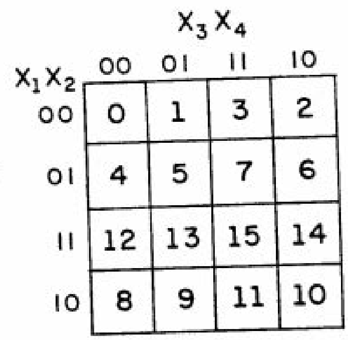
Figures 3 and 4 present diagrams for two different event spaces and the distribution of γ-numbers in them.

a.



A diagram representing binary space **E**(2,2,2,2)

b.



An equivalent Karnaugh map

Fig. 3. Distribution of Υ-numbers in a diagram and an equivalent Karnaugh map.

X_1	X_2												
0	0	0	1	2	3	4	5	6	7	8	9	10	11
	1	12	13	•	•	•	•	•					
1	0												
	1												
2	0						•	•	•	•	•	58	59
	1	60	61	62	63	64	65	66	67	68	69	70	71
	X_4	0	1	2	0	1	2	0	1	2	0	1	2
	X_3	0			1			2			3		

Fig. 4. Diagram for space $\mathbb{E}(3,2,4,3)$.

3. RECOGNITION OF INTERVAL AND CARTESIAN COMPLEXES IN A DIAGRAM

3.1 Definitions

We will introduce now certain concepts which play an important role in various applications of the diagram.

Let $\{x_i = \alpha_i\}$, $i \in 1,2,\ldots,n$, where α_i is a subset of D_i, denote the set of all events from an $\mathbf{E}$, whose x_i component takes a value from α_i:

$$\{x_i = \alpha_i\} = \{e = (x_1, x_2, \ldots, x_n) | x_i \in \alpha_i\} \tag{6}$$

Such an event set is called a cartesian literal. If the subset α_i is a sequence of consecutive integers, a+1, ..., b, then the cartesian literal is called an interval literal and denoted $\{x_i = a..b\}$. If α_i consists of only one element, then $\{x_i = \alpha_i\}$ is called an elementary literal.

A set-theoretical product of cartesian (interval) literals is called a cartesian complex (interval complex):

$$L = \bigcap_{i \in I} \{x_i = \alpha_i\}, \; I \subseteq \{1,2,\ldots,n\} \tag{7}$$

A cartesian complex which is a product of elementary literals is called an elementary complex.

Set-theoretical operations on cartesian complexes: complement, product and sum are equivalent to the complement, intersection and union of the corresponding sets of cells in a diagram* (Fig. 5, 6, 7).

In the binary case (i.e., when $d_1 = d_2 = \ldots = d_n = 2$) cartesian complexes reduce to sets of events corresponding to single conjunctions of binary literals. For example, complex $\{x_1=0\}\{x_3=1\}\{x_5=0\}$ corresponds to conjunction $\bar{x}_1 x_3 \bar{x}_5$.

3.2 Supporting concepts

A function

$$f: \; \mathbf{E} \to D \tag{8}$$

---

* In the sequel, whenever it does not lead to confusion, a set of cells corresponding to a cartesian complex will be called simply a cartesian complex.

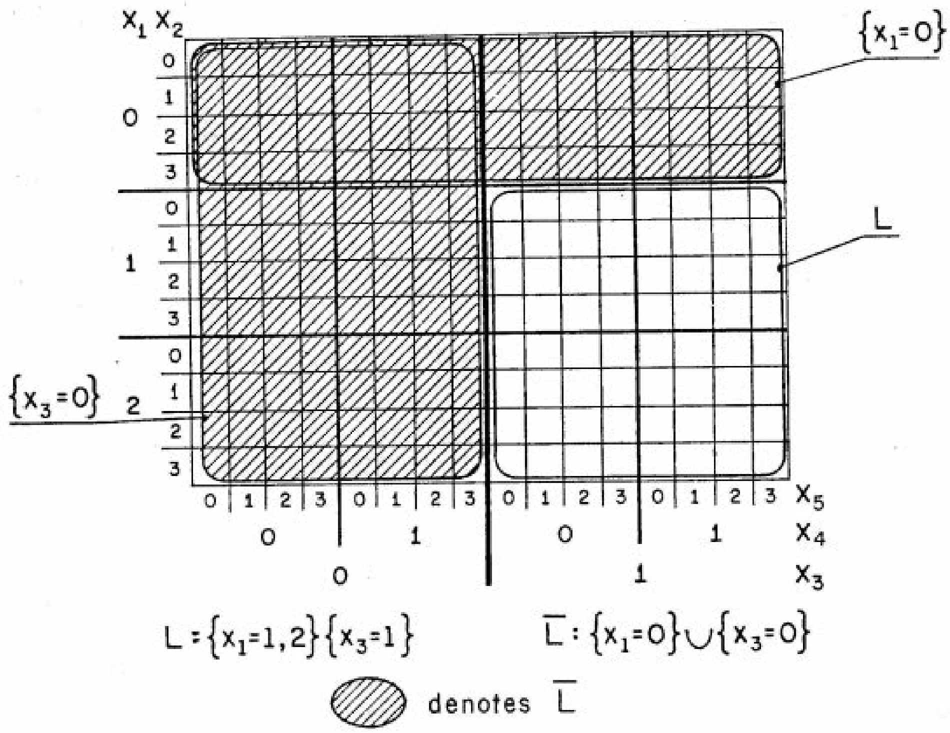


Fig. 5. Complement of a cartesian complex.

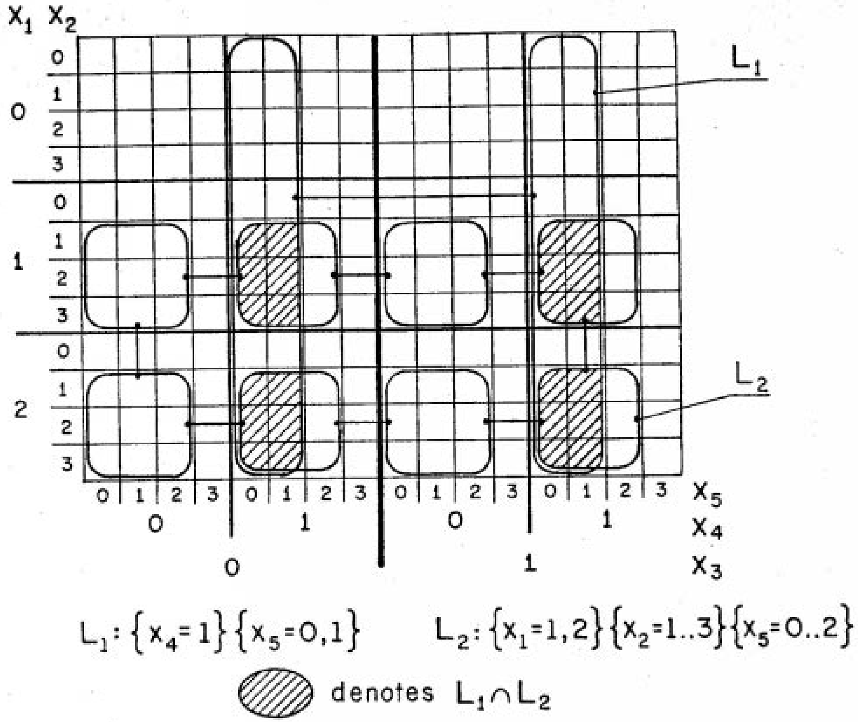


Fig. 6. An intersection of cartesian complexes.

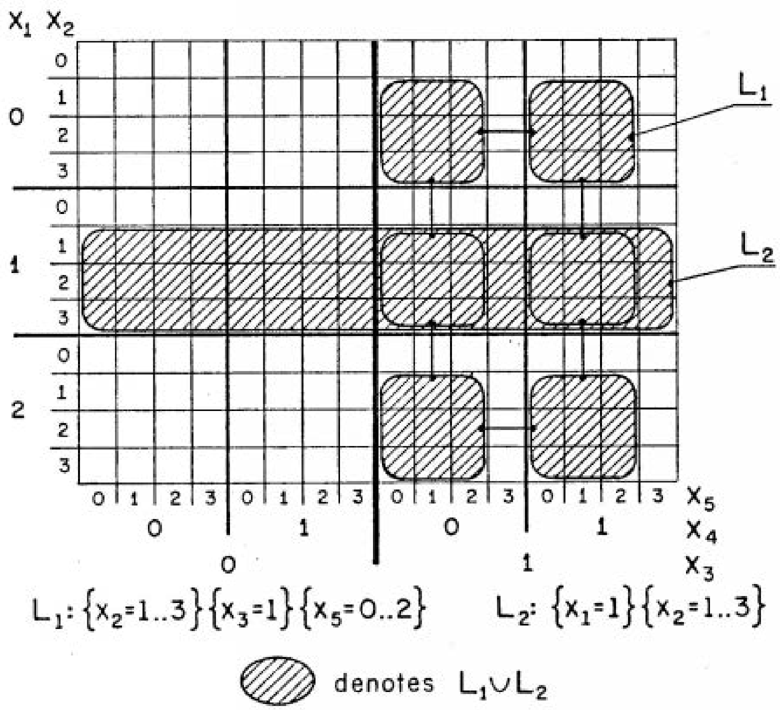


Fig. 7. A union of cartesian complexes.

can be represented in a diagram by marking cells representing events $e \in \mathbf{E}$ by corresponding value f(e). For example, Fig. 8 presents a diagramatic representation of the function from Fig. 1:

$$f: \mathbf{E}(4, 4, 4, 4) \rightarrow \{0, \tfrac{1}{2}, 1, *\} \tag{9}$$

In solving various logical or combinatorial problems occurring, e.g., in logic design, pattern classification, decision theory, etc., there is often a need for expressing a function f (8) in terms of concepts which are special cases of cartesian or interval complexes. For example, in logical design such concepts are prime implicants; in pattern classification and artificial intelligence - property lists or logical products of conditions: 'does feature x have value a' or 'does feature x have value in the set A' (Michalski [12]). In using diagrams for manual solving or illustrating such problems, or as an aid in designing and testing algorithms for a computer solution, a question arises of how to visually recognize in a diagram the configurations of cells which correspond to cartesian or interval complexes. In order to develop such a rule, we will first introduce some geometrical concepts which are very easy to recognize in the diagram, and then express the rule in terms of these concepts.

Definition 1: A set of cells included in one row {column} or in two or more adjacent rows {columns} generated in step $i = 1, 2, \ldots, v$ $\{i = v+1, v+2, \ldots, n\}$ and, if $i \neq 1$ $\{i \neq v+1\}$, contained in a single row {column} generated in step $i-1$, is called a regular row {regular column} (Fig. 9).

Definition 2: The intersection of any regular row and any regular column is called a regular rectangle (Fig. 10).

Definitions 1 and 2 imply that any regular rectangle can be expressed as a single product of interval literals, that is as an interval complex. A regular rectangle can be considered a diagram itself, representing

X_2	0				1				2				3			
X_1 \ X_3	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
0				0	0			0	0			0			1/2	
1	1	1	1			1	1/2	1		1		1			1/2	0
2	0			0	1	1	1/2		1	1					1/2	0
3	0					1/2	1/2			1/2	1/2		0	1/2		

Fig. 8. Diagramatic representation of the mapping f from Fig. 1. The empty cells have value *.

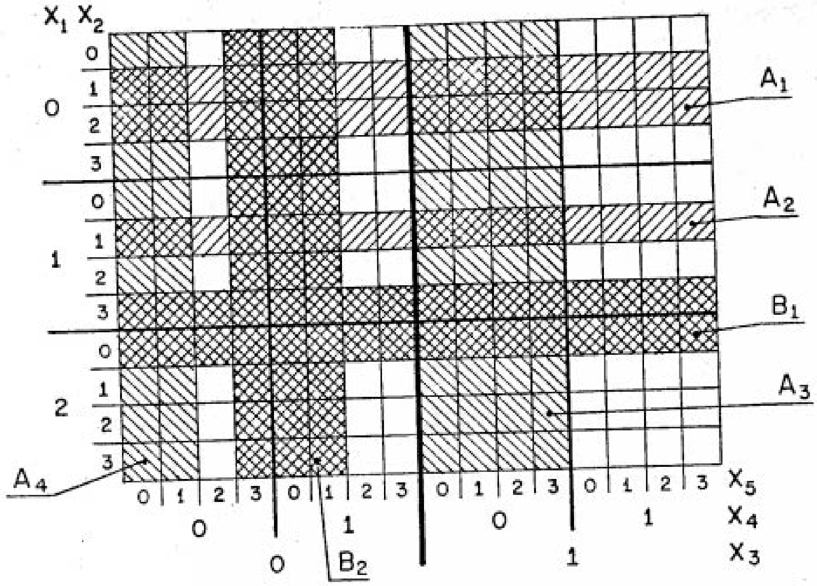


Fig. 9. A_1, A_2 - regular rows, A_3, A_4 - regular columns
B_1 - not a regular row, B_2 - not a regular column.

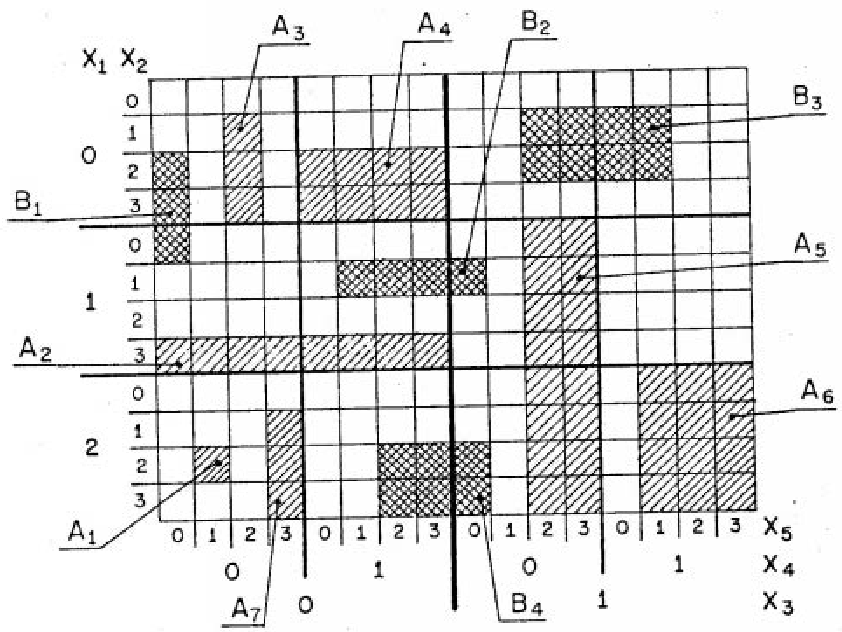


Fig. 10. $A_1, A_2, \dots, A_7$ - regular rectangles
B_1, B_2, B_3, B_4 - not regular rectangles.

a subspace of the total event space, spanned over the axes which cross the rectangle. If we assume that the perimeter of a diagram is made of the thickest line, then it is easy to observe that the thickest axis crossing any regular rectangle is never thicker than the thickest, parallel to it, borderline axis.

From now on, whenever we use the name rectangle, we will mean a regular rectangle.

Definition 3: A regular partition of a rectangle is a set of rectangles obtained by splitting the original rectangle along the thickest horizontal or vertical axes crossing the rectangle (Fig. 11).

It follows from the definition 3 that expressions of rectangles in a partition differ only in one literal (Fig. 11).

Definition 4: Let E be a set of cells. The minimal-under-inclusion regular rectangle which includes E (i.e. the regular rectangle contained in every other rectangle which includes E), is called the covering rectangle of E (Fig. 12).

Suppose a configuration of cells, E, corresponds to a simple cartesian complex L(E), i.e., a product of literals involving variables from the set $\{x_1,\ldots,x_n\}$.

Lemma 1 If E is a proper subset of rectangle R, then L(E) can be expressed as

$$L(E) = L(R) \cap L \tag{10}$$

where L(R) is a product of literals expressing rectangle R, and L a product of literals called an expression of E in context of R.

Proof Since an intersection of sets of cells corresponds to a product of expressions representing sets and $E \subset R$, therefore the expression L(R) must be a part of the product L(E); the remaining part is L. ∎

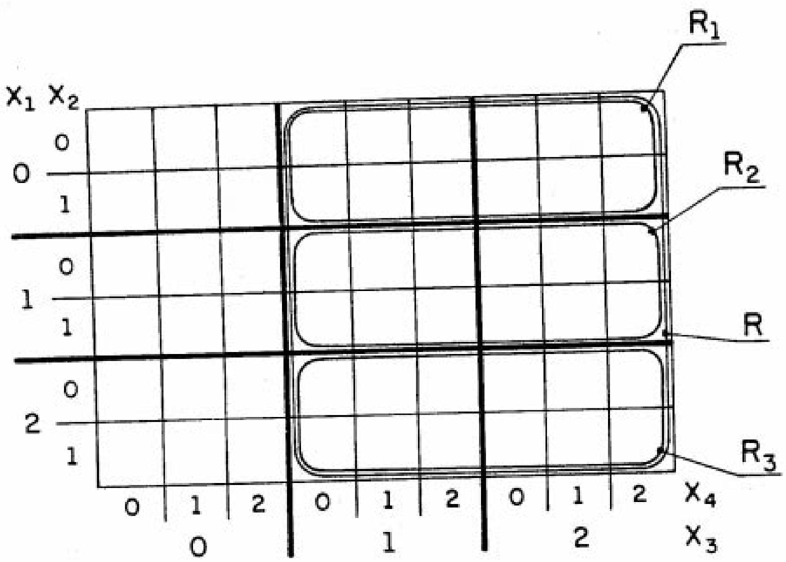


Fig. 11. R_1 : $\{x_1=0\}\{x_3=1,2\}$ R_2 : $\{x_1=1\}\{x_3=1,2\}$

R_3 : $\{x_1=2\}\{x_3=1,2\}$

The set $\{R_1, R_2, R_3\}$ is a regular partition of rectangle R.

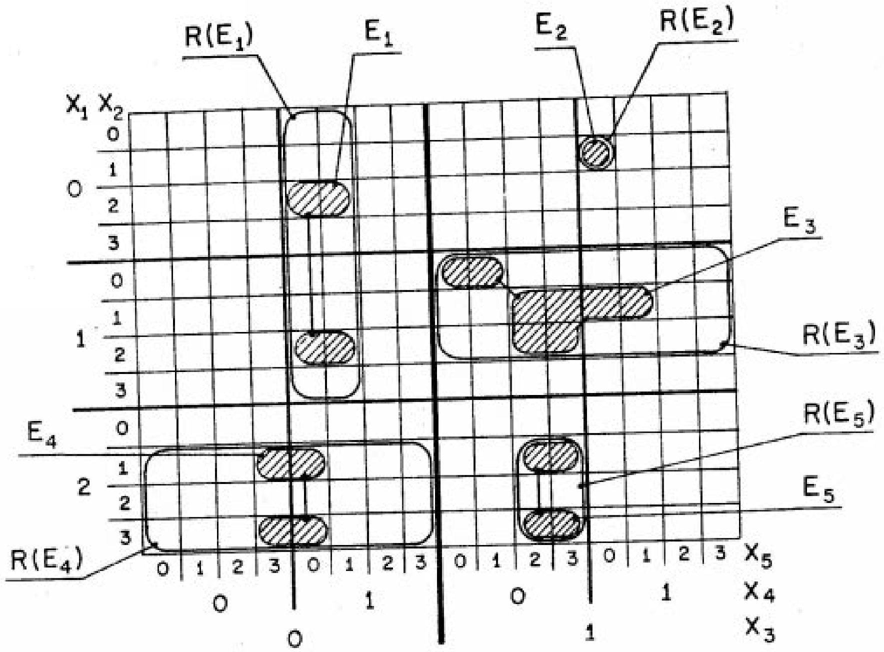


Fig. 12. Event sets E_i and their covering rectangles $R(E_i)$, $i = 1,2,\ldots,5$.

The product L is a cartesian complex in the subspace of event space **E**, determined by rectangle R, i.e., spanned over the axes crossing the rectangle. Lemma 1 implies that in order to determine whether a set E corresponds to a cartesian complex, it is sufficient to determine whether E is a cartesian complex in the context of a rectangle which includes E (rather than in the context of the whole diagram).

Let E be a subset of cells of some regular rectangle R. E can be represented in R by marking cells which belong to E and leaving the remaining cells of R unmarked.

Definition 5: A regular rectangle R with marked cells belonging to a set of cells E is called an image of E and denoted I(R,E). If E is a cartesian complex in the context of R, then R is called an image of a cartesian complex. For completeness, a rectangle with no marked cells is alternatively called an empty image.

Definition 6: Two or more images are congruent if they can be superimposed by translation and, when superimposed, the corresponding cells in the images are all marked or all empty (Fig. 13).

3.3 A cartesian complex recognition theorem and a recognition rule

Suppose E_1, E_2, ... are configurations of cells corresponding to cartesian complexes.

Lemma 2 If and only if images $I(R_1,E_1)$, $I(R_2,E_2)$, ... are congruent then expressions of $E_1,E_2 \ldots$ in the context of $R_1,R_2,\ldots$ are identical.

Proof According to lemma 1, a cartesian complex $L(E_j)$ corresponding to E_j, $j = 1, 2, \ldots$, can be expressed as

$$L(E_j) = L(R_j) \cap L_j \quad \text{(i)}$$

where $L(R_j)$ is the expression of the rectangle R_j, and L_j is the expression of E_j in the context of R_j. Since images $I(R_j,E_j)$ are congruent, then each E_j, $j = 1, 2, \ldots$, has an identical location with regard to the borderlines of the

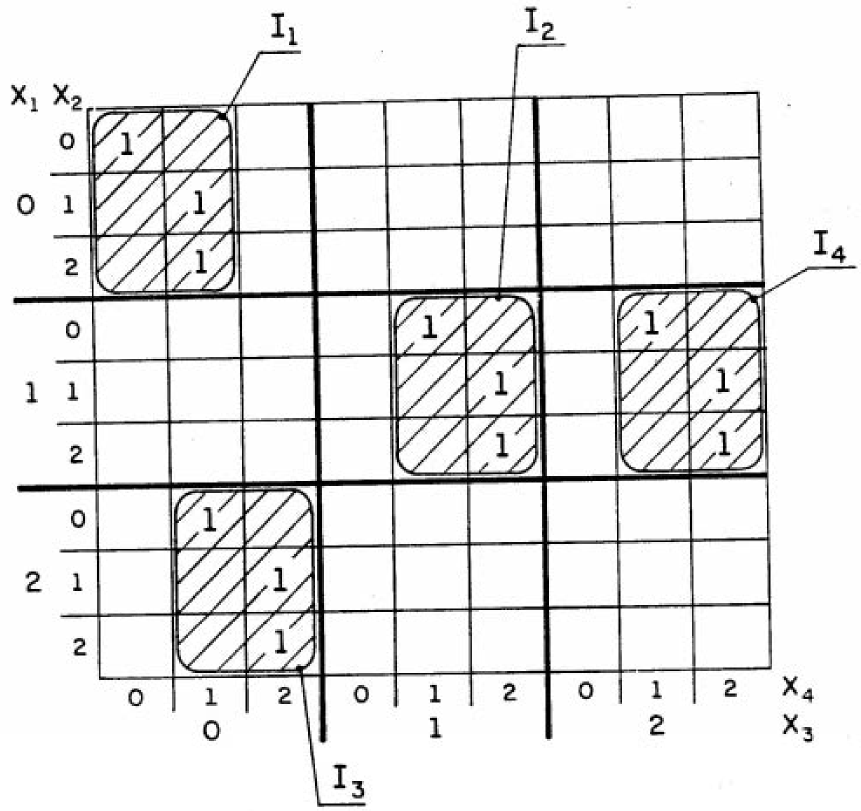


Fig. 13. Images $I_1, .., I_n$ are congruent.

corresponding rectangle and, consequently, the differences between expressions $L(E_j)$, $j = 1, 2, \ldots$, can only be due to the differences in expressions of rectangles, i.e., $L(R_j)$. Thus, all L_j must be identical.

To prove the reverse implication, observe, that if all L_j are identical, then expressions $L(E_j)$, given by (i), can differ only in expressions of rectangles, i.e., $L(R_j)$, and, consequently, images $I(R_j, E_j)$ have to be congruent. ∎

Let L_j, $j = 1, 2, \ldots$ be cartesian complexes which differ only in one literal:

$$L_j = T\{x_k = a_j\}, \; j = 1, 2, \ldots$$

where T is the common part. Using the distributive property of the set-theoretical union over the intersection and the definition of literal, we can write:

$$\bigcup_j L_j = \bigcup_j T\{x_k = a_j\} = T\{x_k = \bigcup_j a_j\} \tag{11}$$

Consequently, the union of such cartesian complexes is also a cartesian complex. The equation (11) is called the <u>combining rule</u>. If $\bigcup_j a_j = D_k$, where D_k is the domain of x_k, then $\{x_k = \cup a_j\}$ is equivalent to the event space $\mathbb{E}$ and rule (11) reduces to the <u>simplification rule</u>:

$$\bigcup_j T\{x_k = a_j\} = T \tag{12}$$

We can now formulate a theorem which leads to a recursive recognition rule for cartesian and interval complexes in an arbitrary diagram.

<u>Theorem 1</u> A configuration E of cells in a diagram is a cartesian complex if and only if:

(1) E is a regular rectangle (in this case E is also an interval complex), or

(2) A regular partition of the covering rectangle of E consists of congruent images of cartesian complexes

and possibly some empty images. If the partition has no empty images, and consists of congruent images of interval complexes, then E is also an interval complex.

Proof: Condition 1: The definition of a regular rectangle (definition 2) implies that a rectangle can be expressed as an interval complex.

Condition 2: Definition 3 implies that a regular partition of a rectangle consists of rectangles whose expressions differ only in one literal. Let R_1, R_2, ... denote nonempty rectangles and E_1, E_2, ... event sets contained in them, respectively. Because images $I(R_j,E_j)$, $j = 1, 2, \ldots,$ are congruent, therefore, according to lemma 2, expressions of E_j in the context of R_j are identical, and, therefore, complete expressions of E_j, $L(E_j)$, differ only in one literal. We have

$$E = E_1 \cup E_2 \cup \cdots \qquad (i)$$

therefore, the expression of E, L(E), is the union of $L(E_j)$ and, according to the combining rule (11), is a cartesian complex.

If a partition of the covering rectangle does not include empty images, that means that in the expressions

$$L(E_j) = T\{x_i = a_j\},\ j = 1, 2, \ldots$$

a_j-s form a set of consecutive integers. Consequently, the union $\bigcup_j L(E_j) = T\{x_i=\alpha_i\}$, where $\alpha_i = \bigcup_j a_j$, is an interval complex, if T is an interval complex, i.e., if rectangles of the partition are images of interval complexes.

Now we have to prove that if E corresponds to a cartesian complex then it satisfies condition 2 (if it satisfies condition 1 it also satisfies condition 2). Let L(E) denote a cartesian complex corresponding to E:

$$L(E) = \bigcap_{i=1}^{n} \{x_i=\alpha_i\},\ \alpha_i \subseteq D_i$$

Suppose sets α_i are transformed to new sets $\alpha_i^1 \supseteq \alpha_i$, by filling 'gaps' in α_i or substituting D_i for α_i, such that

$$R(E) = \bigcap_{i=1}^{n} \{x_i = \alpha_i^1\},$$

defines the covering rectangle of E. If k is selected in such a way that x_k are the thickest vertical or horizontal axes crossing rectangle R(E), then rectangles R_a defined by complexes

$$L(R_a) = L^1 \{x_k = a\}, \; a \in \alpha_k^1$$

where

$$L^1 = \bigcap_{\substack{i \neq k \\ i=1}}^{n} \{x_i = \alpha_i^1\}$$

constitute a regular partition of R(E). A rectangle defined by $L(R_a)$, contains for a ε α_k, $\alpha_k \subseteq \alpha_k^1$, a set of cells, $E_a \subseteq E$, defined by the cartesian complex

$$L(E_a) = L\{x_k = a\}$$

where

$$L = \bigcap_{\substack{i=1 \\ i \neq k}}^{n} \{x_i = \alpha_i\}$$

and for a ε $\alpha_k^1 \setminus \alpha_k$, where $\setminus$ denotes set subtraction, no cells of E. Because all $L(E_a)$, a ε α_i, differ only in one literal (involving x_k), therefore the expressions of E_a in the context of R_a (which are parts of L) are identical, and, according to lemma 2, images $I(R_a, E_a)$, for a ε $\alpha_i^1 \setminus \alpha_i$ are empty. If E is an interval complex then $\alpha_k^1 = \alpha_k$, there are no empty images $I(R_a, E_a)$, and all images are images of interval complexes. ∎

Based on the above theorem, the following rule can be suggested for determining whether a given set E of cells in a diagram represents a cartesian complex:

Rule 1:

1. Find the covering rectangle, R(E), of E. If R(E) is identical with E then go to YES.
2. Determine a vertical or horizontal partition of R(E) which produces the most square-like images.
3. Check if the nonempty images of the partition are congruent. If answer is no, go to NO.
4. Select any nonempty image and treat the subset of cells from E contained in the image as a new set E. Go to 1.

YES: E is a cartesian complex.

NO : E is not a cartesian complex.

If an application of the rule terminates with YES and there were no empty images at any step of the rule, then E is an interval complex.

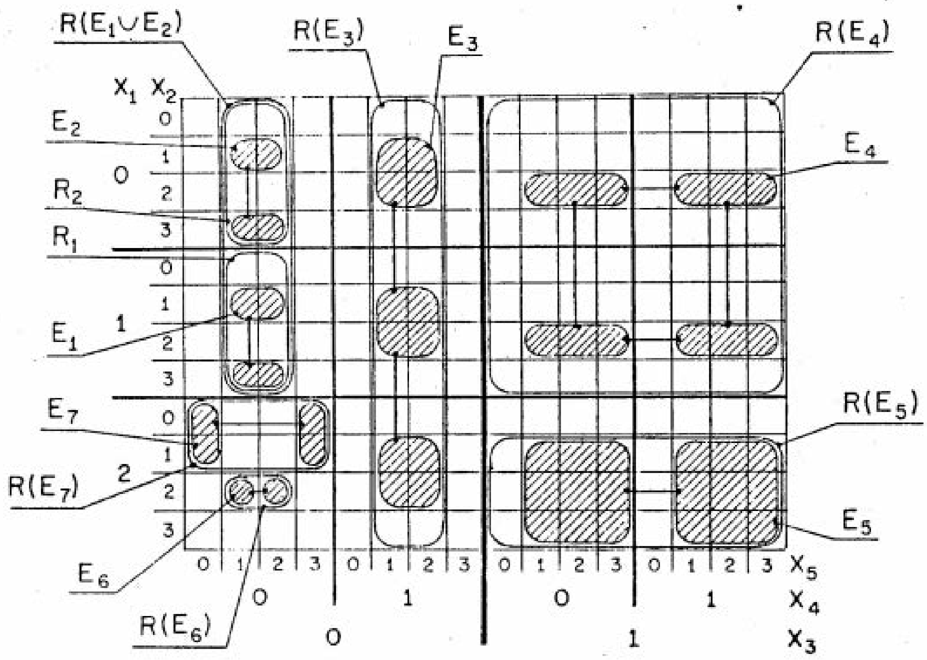
The rule can be used for 'building up' complexes or determining maximal possible complexes, by starting with any set of cells which can be directly recognized as representing a complex (in the worst case, an individual cell), then adding to it more cells and applying the rule.

Fig. 14 illustrates the rule by a few examples of cartesian complexes.

3.4 Recognition rule for elementary complexes

In many applications, for example, in optimizing decision tables (Michalski [16]), one deals with a special case of cartesian complexes, which are elementary complexes (i.e., complexes $L_i = \bigcap\{x_i = L_i\}$, where L_i is just a single value (see sec. 3.1)). For such applications, rather than general rule 1, one needs a rule for recognizing elementary complexes. Such a rule can be easily obtained by appropriately modifying rule 1.

To do so, we will first introduce concepts of an elementary row, elementary column and an elementary rectangle (which are special cases of the



$E_1 \cup E_2, E_3, E_4, E_5, E_6$, and E_7 are cartesian complexes

(E_3, ..., E_6 are interval complexes)
Illustration of Theorem 1.

Fig. 14

regular row, regular column and regular rectangle, respectively - see definition 1 and 2).

Definition 1a: A set of cells included in a single row {column} created at any step of constructing a given diagram, is called an elementary row {column}.

Definition 2a: The intersection of any elementary row with any elementary column is called an elementary rectangle.

Next we define:

Definition 4a: The minimal-under-inclusion elementary rectangle which contains a given set of cells, E, is called an elementary covering rectangle of the set E.

It follows from definitions 1a and 2a, that an elementary rectangle is an elementary complex. Therefore it is easy to see, that if in theorem 1 every regular rectangle is restricted to be an elementary rectangle, then the theorem describes conditions for recognizing elementary complexes. Thus, we have:

Theorem 1a: A configuration of cells, E, is an elementary cartesian complex, iff:

(1) E is an elementary rectangle, or

(2) a regular partition of the elementary covering rectangle of E consists of congruent images of elementary cartesian complexes.

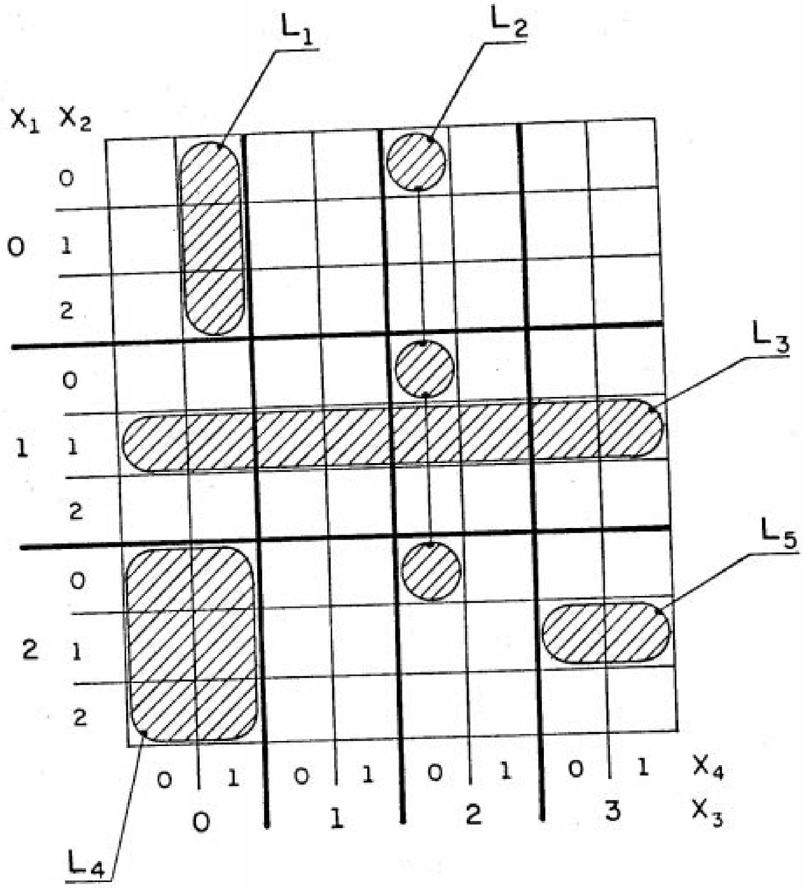
The proof is left to the reader. Using theorem 1a, it is easy now to appropriately modify rule 1 to obtain a recognition rule for elementary complexes.

Figure 15 gives a few examples of elementary complexes.

4. REMARKS ON APPLICATIONS AND EXAMPLES

The diagram presented here is a general model of discrete spaces and therefore can be useful as a geometrical representation in any situation involving such spaces. In particular, it can be used as an educational aid in various areas of computer science, logic, discrete mathematics, etc.

For research purposes, it can be useful in design and testing computer algorithms involving discrete spaces and functions determined on them (e.g., binary, many-valued, discrete and variable-valued logic functions).



L_1: $\{X_1=0\}\{X_3=0\}\{X_4=1\}$

L_2: $\{X_2=0\}\{X_3=2\}\{X_4=0\}$

L_3: $\{X_1=1\}\{X_2=1\}$

L_4: $\{X_1=2\}\{X_3=0\}$

L_5: $\{X_1=2\}\{X_2=1\}\{X_3=3\}$

Fig. 15. Examples of elementary cartesian complexes.

An example of such an application is the development of a series of AQVAL programs for computer-aided inductive inference (Michalski [15]). The diagram was extensively used there for developing algorithms, solving test examples, illustrating obtained solutions, etc.

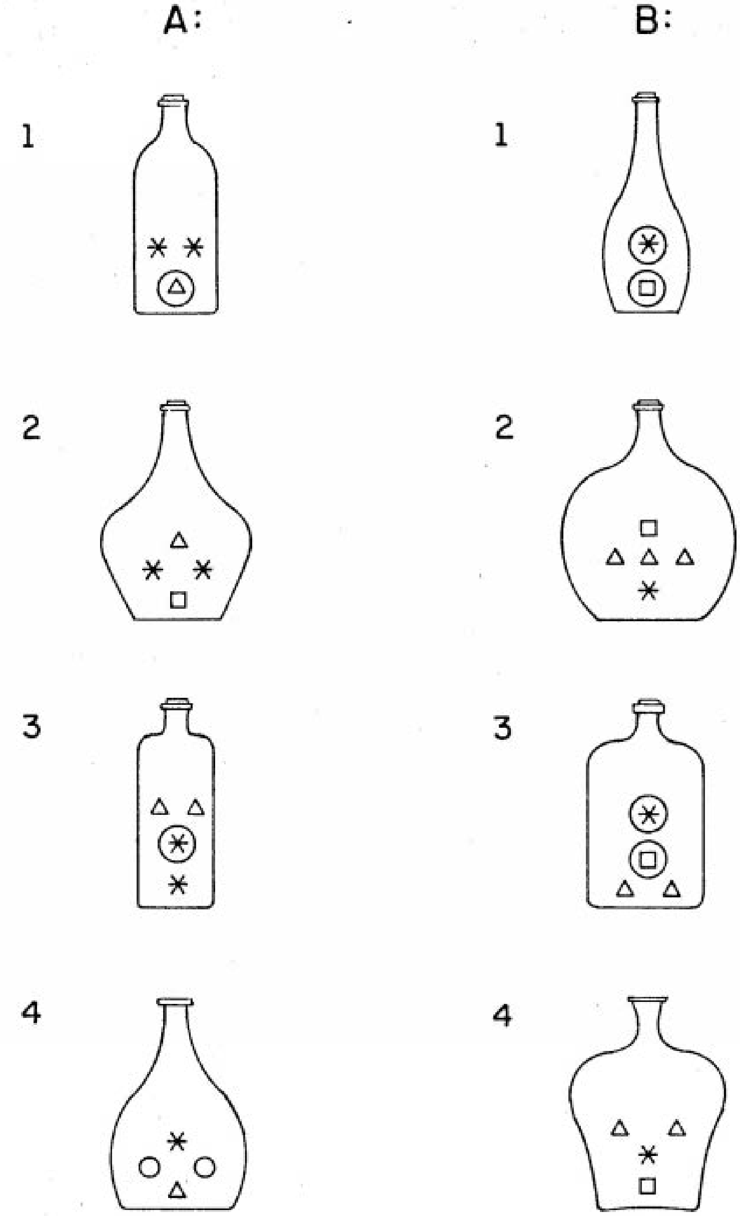
In binary logic design the diagram can be used similarly to the Karnaugh map (it can be especially useful in the case of more than 4 variables and up to 8-10 variables). And, of course, the diagram can also be used in many-valued logic design.

Another possible application is to designing decision tables and decision trees, in particular for quick testing decision tables for redundancy, consistency and completeness; for reduction of decision tables and for converting them into decision trees. This application is described in detail in Michalski [16].

In conclusion, to give the reader an opportunity to get more practice in using the diagram we will consider two examples.

Example 1: Find a classification rule.

This, somewhat whimsical, example gives an illustration of a less-traditional use of the diagram. Suppose we were given 4 different bottles of wine produced by company A and 4 different bottles of wine produced by company B (Fig. 16). Suppose, for the sake of the example, that the wines did not have labels with company names but each bottle had a special geometrical pattern on it, as shown in Fig. 16. The problem is to determine the 'simplest' rule(s) which will permit one to distinguish the wines of each company based on the geometrical pattern. A way to solve the problem is to determine first a set of possible relevant characteristics (descriptors) of patterns, and then to construct the simplest (according to some criteria) 'discriminant' description of each class of wines in terms of these descriptors (or their subset). Although determining a set of possibly relevant descriptors is, in general, a difficult problem itself, we will assume here that it can be done, and will consider only the problem of determining the final rule.



x_1 - # of squares $\quad D(x_1)$ - {0,1}

x_2 - # of triangles $\quad D(x_2)$ - {0,1,2,3}

x_3 - # of cirles $\quad D(x_3)$ - {0,1,2}

x_4 - # of asterisks $\quad D(x_4)$ - {0,1,2}

A: $\{x_3 = 1\} \cup \{x_2 = 1\}$

B: $\{x_1 = 1\}\{x_4 = 1\}$

Fig. 16. Bottles of wine produced by company A and B.

Suppose the following descriptors were selected as a possible try:

x_1 - number of squares in the pattern,

x_2 - number of triangles,

x_3 - number of circles,

x_4 - number of asterisks.

From Fig. 16, we can determine that the domains $D(x_i)$ of these descriptors, sufficient for describing each bottle, are: $D(x_1) = \{0, 1\}$, $D(x_2) = \{0, 1, 2, 3\}$, $D(x_3) = \{0, 1, 2\}$ and $D(x_4) = \{0, 1, 2\}$.

Figure 17 shows the diagram for the space $D(x_1) x \ldots x\, D(x_4)$ with marks indicating the correspondence between the cells and individual bottles. If the number of conjunctive statements is accepted as a measure of complexity of a description, then the simplest discriminant description of bottles of company, say A, can be obtained by grouping cells marked A into the fewest elementary cartesian complexes, whose union includes every cell A but does not include any of the cells B. Such a grouping is shown in Fig. 17.

The union of complexes L_1 and L_2 gives us a description of the bottles of company A:

$$A: \ \{x_2 = 1\} \cup \{x_3 = 1\}$$

It can be interpreted: 'if a bottle <u>has 1 triangle or 1 circle</u> then the wine was made by company A.' In a similar way, we can obtain a description of wines of company B:

$$\{x_1 = 1\}\{x_4 = 1\}$$

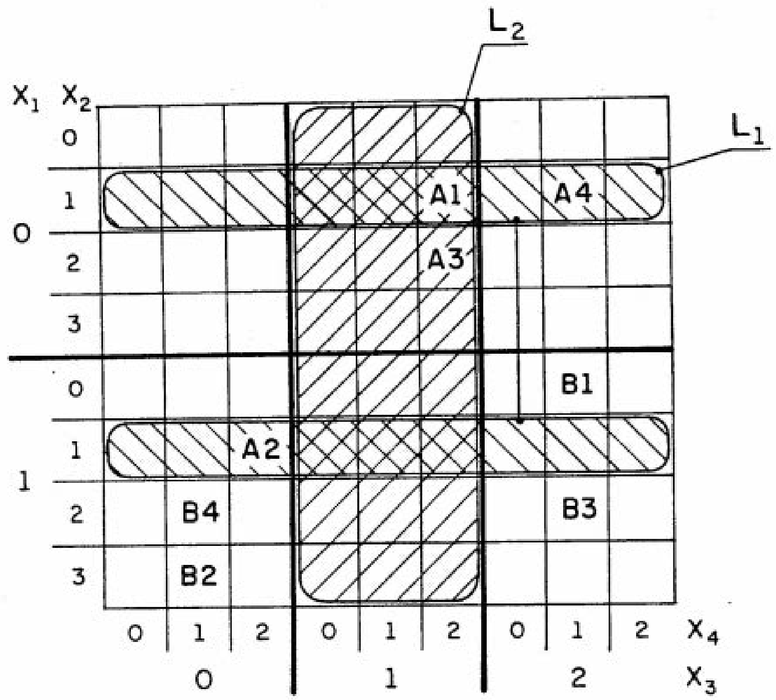
That is: 'if a bottle <u>has 1 square and 1 asterisk</u> then the wine was made by company B.'

By grouping cells differently, we can obtain alternative descriptions, for example:

$$A: \ \{x_1 = 0\} \cup \{x_2 = 1\}$$

or

$$A: \ \{x_1 = 0\} \cup \{x_4 = 2\}$$



A: $L_1 \cup L_2$

where L_1: $\{x_2 = 1\}$, L_2: $\{x_3 = 1\}$

Fig. 17. Diagram representing a discriminant description of bottles A.

B: $\{x_1 = 1\}\{x_2 \neq 1\}$

or

B: $\{x_1 = 1\}\{x_4 \neq 2\}$

The interpretation of these descriptions is left to the reader.

The above example gave an elementary and informal introduction to the kind of problems which occur in the application of the variable-valued logic system VL_1 to problems of computer induction and pattern recognition (Michalski[12-15] , Larson and Michalski[6]).

It is obvious that when the number of descriptors is large, descriptions of this kind should be synthesized using a computer. (Paper [14] presents synthesis algorithms and papers [6], [15] describe various computer implementations in the logic system VL_1, and, also, much richer system VL_2[15]).

Example 2: Determine a DVL_1 expression of the function represented in Fig. 1 (and Fig. 8).

Any function f: $\mathbf{E} \to D$, where D is a linearly ordered set, can be expressed as a disjunctive normal expression in the variable-valued logic system VL_1, i.e., a $\underline{DVL_1}$ expression (Michalski [13]).

Assuming that $D = \{0,...,d\}$, a DVL_1 expression is a disjunction (maximum function) of terms, where term is a product (minimum function) of selectors. A selector is either a constant from D or a function $[x_i=\alpha_i]$, from $\mathbf{E}$ into endpoints of D, defined:

$$[x_i=\alpha_i](e) = \begin{cases} d, & \text{if } e \in \{x_i=\alpha_i\} \\ 0, & \text{otherwise} \end{cases}$$

To obtain a DVL_1 expression of the function f from Fig. 8 one groups the cells marked by 1 into cartesian complexes (including cells marked by *, whenever it is convenient).

Next, all cells marked by 1 obtain mark *; and cells marked by 1/2 are grouped into cartesian complexes (Fig. 18). Expressing complexes as terms with appropriate constants one obtains the following DVL_1 formula:

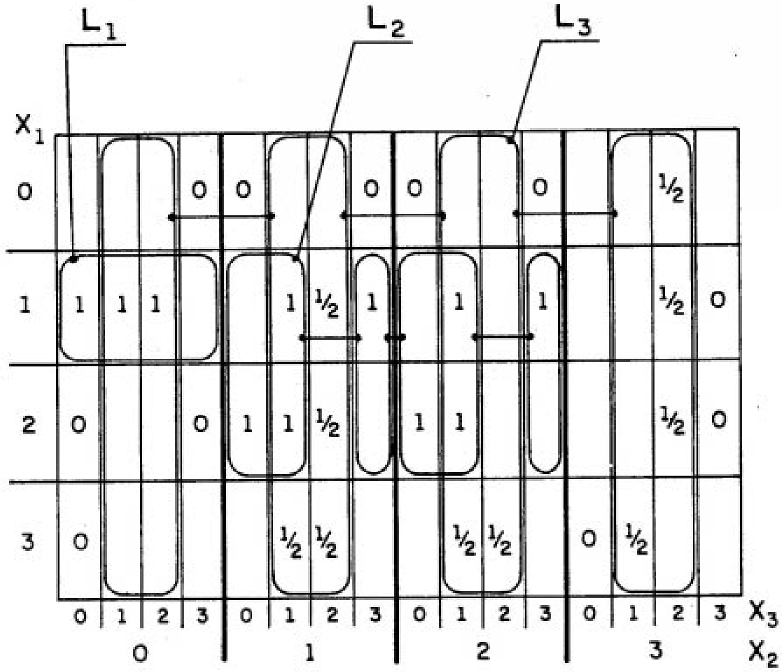


Fig. 18. Cartesian complexes for determining a DVL_1, expression of function from Fig. 8.

$$f:\ \underbrace{1[x_1=1][x_2=0]}_{L_1} \vee \underbrace{1[x_1=1,2][x_2=1,2][x_3\neq 2]}_{L_2} \vee \underbrace{\tfrac{1}{2}[x_3=1,2]}_{L_3}$$

An equivalent way of expressing f is by an ordered set of production rules (i.e., rules which consist of a condition part and an action part; the action part is evoked if the condition part is satisfied). Rules in an ordered set are applied sequentially, the first rule **satisfied** produces the action:

$$[x_1=1][x_2=0] \Rightarrow [f=1]$$
$$[x_1=1,2][x_2=1,2][x_3\neq 2] \Rightarrow [f=1]$$
$$[x_3=1,2] \Rightarrow [f=\tfrac{1}{2}]$$

Sometimes an unordered set of rules is preferable (i.e., rules can be applied in any order). In this case, each set of cells with the same mark is treated independently. For example, function f from Fig. 8 can be expressed by the following unordered set of rules:

$$[x_1=1][x_2=0] \Rightarrow [f=1]$$
$$[x_1=1,2][x_2=1,2][x_3\neq 2] \Rightarrow [f=1]$$
$$[x_3=2] \Rightarrow [f=\tfrac{1}{2}]$$
$$[x_1=3][x_3=1,2] \Rightarrow [f=\tfrac{1}{2}]$$

To see clearly the difference between the two sets of rules, the reader is advised to represent them in the diagram.

5. SUMMARY

We have presented here a general geometrical model of multidimensional discrete spaces, and introduced several concepts associated with the model, such as regular row, column, rectangle, a regular partition, etc. These concepts were subsequently used for determining a rule for recognizing constructs, specifically cartesian and interval complexes, that are important in various applications of the model.

The model can serve as an aid in designing, testing, describing, and, also, in many practical problems, in hand executing algorithms involving discrete, many valued or variable-valued logic functions. It can also be useful in education, for teaching concepts and algorithms involving such functions.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation Grant NSF MCS- 76-22940.

The author owes many thanks to his collaborators and students who shared with him the experiences of using the diagram over the years and contributed to the final form of the recognition rule of cartesian complexes. Thanks are also due to the former CACM Technical Department Editor Professor Glenn Manacher for his comments and suggestions.

REFERENCES

1. Bochenski, J. M. Formal Logic. Frelburg-Munchen, 1956

2. Deschamp, J. P. and Thayse A. Representation of Discrete Functions. Proceedings of the 1975 International Symposium on Multiple-Valued Logic, Indiana University, Bloomington, Indiana, May 13-16, 1975, pp. 99-111.

3. Jensen, G. M. The Determination of Symmetric VL_1 Formulas. Master's Thesis, Department of Computer Science, University of Illinois, Urbana, 1975.

4. Karnough, M. The Map Method for Synthesis of Combinational Logic Circuits, Commun. Electron. November 1953, pp. 593-599.

5. Kuzichev, A. S. Diagramy Venna. Izd. Nauka, 1968 (in Russian).

6. Larson, J. and Michalski, R. S. AQVAL/1 (AQ7) User's Guide and Program Description. Report No. 731. Department of Computer Science, University of Illinois, June 1975.

7. Marquand, A. On Logical Diagrams for n Terms. The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science. Vol. XII, 5th Series, No. 75, 1881, 266-270.

8. Michalski, R. S. Graphical Minimization of Switching Functions in Class of Disjunctive Normal Forms. Journal of the Institute of Automatic Control. Polish Academy of Sciences. No. 52, Warsaw, 1967 (in Polish).

9. Michalski, R. S. Recognition of Total or Partial Symmetry in a Switching Function Completely or Incompletely Specified. Proceedings of the IV Congress of International Federation on Automatic Control (IFAC), Vol. 27 (Finite Automata and Switching Systems), Warsaw, June 16-21, 1969, pp. 109-129.

10. Michalski, R. S. Conversion of Normal Forms of Switching Functions into Exclusive-Or-Polynomial Forms. Archiwum Automatyki i Telemechaniki, Polish Academy of Sciences, No. 3, 1971, 263-278 (in Polish).

11. Michalski, R. S. A Geometrical Model for the Synthesis of Interval Covers. Report No. 461, Department of Computer Science, University of Illinois, Urbana, Illinois, June 24, 1971.

12. Michalski, R. S. A Variable-Valued Logic System as Applied to Picture Description and Recognition," Chapter in the book, GRAPHIC LANGUAGES, eds. F. Nake and A. Rosenfeld, North-Holland Publishers, 1972.

13. Michalski, R. S. Variable-Valued Logic and Its Applications to Pattern Recognition and Machine Learning. Chapter in the monograph: Multiple-Valued Logic and Computer Science, edt. David Rine, North-Holland Publishers, 1975.

14. Michalski, R. S. Synthesis of Optimal and Quasi-Optimal Variable-Valued Logic Formulas. Proceedings of the 5th International Symposium on Multiple-Valued Logic, Bloomington, Indiana, May 13-16, 1975, pp. 76-87.

15. Michalski, R. S. A System of Programs for Computer-aided Induction: A Summary. Proceedings of the Fifth Intern. Joint Conf. on Artificial Intelligence, MIT, 22-26 August, 1977.

16. Michalski, R. S. Designing Extended Entry Decision Tables and Optimal Decision Trees Using Decision Diagrams. Report No. UIUCDCS-R-78-898, Department of Computer Science, University of Illinois, Urbana, IL., March 1978.

17. Michalski, R. S., Kulpa A. A system of programs for the synthesis of switching circuits using the method of disjoint stars, Foundations of Information Processing, IFIP Congress, Ljubliana, August 1971.

18. Svoboda, A. Grafico-mechanicke pomucky uzirane pri analise a synthese kontaktowych obvodu. Stroje Zpracov. Inf. No. 4, 1956 (in Chech).

19. Veitch, E. W. A Chart Method for Simplifying Truth Functions. Proc. Assoc. Comput. Machine Conf., May 2-3, 1952, pp. 127-133.

20. Venn, J. On the Diagrammatic and Mechanical Representations of Propositions and Reasoning. The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science, vol. 10, 1880.

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUCDCS-R-78-897	2.	3. Recipient's Accession No.
4. Title and Subtitle: A PLANAR GEOMETRICAL MODEL FOR REPRESENTING MULTIDIMENSIONAL DISCRETE SPACES AND MULTIPLE-VALUED LOGIC FUNCTIONS			5. Report Date: January 1978
			6.
7. Author(s): Ryszard S. Michalski			8. Performing Organization Rept. No. UIUCDCS-R-78-897
9. Performing Organization Name and Address: University of Illinois at Urbana-Champaign, Department of Computer Science, Urbana, Illinois 61801			10. Project/Task/Work Unit No.
			11. Contract/Grant No.
12. Sponsoring Organization Name and Address			13. Type of Report & Period Covered
			14.

15. Supplementary Notes

16. Abstracts A general planar model of multidimensional discrete spaces (a diagram) is described which can be used for geometrically representing binary, multiple-valued, discrete and variable-valued logic functions. It is essentially a multiple-valued extension of Marquand's binary diagram, with an additional feature of varying thickness of lines representing axes of variables.

The diagram has been used extensively over the years by the author and his collaborators, and has proved to be useful for such tasks as: logic design and development of efficient algorithms for optimization of switching circuits of many variables, detection of symmetry in binary or multiple-valued logic functions, fast conversion of normal forms of switching functions to exclusive-or polynomial forms, design of algorithms for inductive inference and pattern recognition, optimization of decision tables and their conversion to optimal decision trees.

A method for recognizing in the diagram certain constructs (cartesian complexes), which are important for various applications, is presented and illustrated by two examples: one, involving a determination of a classification rule, and another, involving a synthesis of a variable-valued logic expression.

17. Key Words and Document Analysis. 17a. Descriptors

Logic diagram, Marquand diagram, Karnaugh map, Veitch diagram, discrete spaces, multiple-valued logic, variable-valued logic, logic design.

CR categories: 3.61, 3.63, 5.20, 5.21, 6.1

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 42
Release Unlimited	20. Security Class (This Page UNCLASSIFIED	22. Price

FORM NTIS-35 (10-70) USCOMM-DC 40329-P71