A GEOMETRIC MODEL FOR THE SYNTHESIS OF INTERVAL COVERS

by

Ryszard S. Michalski

Report No. 461, Department of Computer Science, University of Illinois,
Urbana, Illinois, June 24, 1971.

COO-2118-0013

Report No. 461

A GEOMETRICAL MODEL FOR THE SYNTHESIS OF INTERVAL COVERS

by

R. S. Michalski

June 24, 1971

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

This work was supported by Contract AT(11-1)-2118 with the U.S. Atomic Energy Commission.

A GEOMETRICAL MODEL FOR THE SYNTHESIS OF INTERVAL COVERS

R. S. Michalski
Department of Computer Science
University of Illinois

ABSTRACT

The paper presents a planar geometrical model of a discrete finite vector space $\mathbf{E}$, called a generalized logical diagram (GLD), and then uses the GLD to interpret and illustrate different concepts and algorithms described in [1] (the concept of an interval complex, of a mapping f: $\mathbf{E} \rightarrow \{[0,1],*\}$, of an interval cover of f, of the extension operation $\surd$, algorithm G for star generation and algorithm A^Q for interval cover synthesis).

In simple cases, the GLD can also be used for the direct (graphical) synthesis of interval covers using a simple rule for the recognition of interval complexes, given in the paper.

ACKNOWLEDGMENT

The author would like to acknowledge his gratitude to Professor B. H. McCormick for the valuable discussions pertaining to this paper, his reading the manuscript and comments toward its improvement.

The author wishes also to thank Mrs. Roberta Andre' for the accurate typing and Mr. Stanley Zundo for the excellent drawings.

1. INTRODUCTION

The interval generalization of switching theory described in [1] discussed a Boolean algebra of event sets in a discrete finite vector space $\mathbf{E}$ and mappings f from E into $\{[0,1],*\}$, where * represents an unspecified value. The Boolean algebra and Boolean functions considered in switching theory are a special case of the above, in which $\mathbf{E}$ is a space of binary vectors and f maps E into the endpoints of the interval [0,1] and *. The basic problem investigated in [1] was how to find a minimal collection of multidimensional intervals, whose union covers a set $F^{1\lambda}$, defined as $\{e \in \mathbf{E} | f(e) \geq \lambda\}$, where λ is an assumed threshold value $(0 \leq \lambda \leq 1)$, and does not cover any element of the set $F^{0\lambda} = \{e \in E | f(e) < \lambda\}$. A set satisfying the above conditions was called a minimal exact unordered interval cover of the set $F^{1\lambda}$ against $F^{0\lambda}$ (the names 'exact,unordered' distinguish this cover from free and ordered covers, also introduced in [1]).

The purpose of the present paper is to present a planar geometrical model of the space $\mathbf{E}$ which can serve as a useful visual aid for representing interval covers of mappings f and geometrically interpreting algorithms for their synthesis. In simple cases this model can also be used for the direct graphical synthesis of interval covers.

To permit one to understand this paper without prior, though desirable, knowledge of [1], those concepts from [1] which are relevant to this paper have been included (in most cases in their equivalent geometrical form).

2. THE FINITE DISCRETE VECTOR SPACE $\mathbf{E}$

Assume that some objects or processes are described by a set of n parameters (discrete or continuous) and each parameter is mapped into a discrete variable x_i, $x_i \in \{ 0, 1, 2, \ldots, h_i-1 \}$, $i = 1, 2, \ldots, n$. Values h_i represent the number of discrete units into which the range of variability of each parameter is resolved to achieve desired accuracy.

Variables $x_1, x_2, \ldots, x_n$ are grouped together into vectors $e^j = (x_1, x_2, \ldots, x_n)$, called <u>events.</u> The finite vector space consisting of all possible events e^j is denoted by $\mathbf{E}(h_1, h_2, \ldots, h_n)$ or briefly by $\mathbf{E}$:

$$\mathbf{E} = \{ (x_1, x_2, \ldots, x_n) \} = \{ e^j \}_{j=0}^{H-1} \tag{1}$$

where $H = h_1 h_2 \cdots h_n$ and j is the value of a function γ: $\mathbf{E} \rightarrow \{ 0, 1, 2, \ldots, H-1 \}$:

$$j = \gamma(e) = x_n + \sum_{j=n-1}^{1} \left(x_j \prod_{k=n}^{j+1} h_k \right) \tag{2}$$

$\gamma(e)$ is called the <u>number</u> of an <u>event</u> e. For example, if in the space $\mathbf{E}(5, 6, 4, 3)$ an event $e = (3, 4, 1, 2)$ then $\gamma(e) = 2 + 1 \cdot 3 + 4 \cdot 4 \cdot 3 + 3 \cdot 6 \cdot 4 \cdot 3 = 269$ and the event e is denoted by e^{269}.

3. A MAPPING f: $\mathbf{E} \rightarrow \{ [0,1], * \}$ AND A COVER OF THE MAPPING f

Let f be a mapping from the space $\mathbf{E}$ into $\{ [0,1], * \}$, where * denotes an unspecified value:

$$f: \quad \mathbf{E} \rightarrow \{ [0,1], * \} \tag{3}$$

The mapping f can be interpreted, for example, in the following way. Assume we are given two classes of objects, class 1 and 0. By measuring some assumed parameters and specifying values for $x_1, x_2, \ldots, x_n$, each object can be characterized by an event $e \in E$. Assume that samples are taken from class 1 and 0 and the sets E^1 and E^0 are formed from distinguished events characterizing the objects of class 1 and 0, respectively. The sets E^1 and E^0, in general, may not be disjoint. Thus, if an event $e \in E^1 \cap E^0$ is given, we cannot infer from this alone to which class the object characterized by e belongs.

Assume that in the above case we are able to estimate the probability f(e) that the object belongs to class 1. Thus, the example can be formally described by a mapping f defined by (3), if:

$$F^1 = \{ e \in E \mid \quad f(e) = 1 \} = E^1 \setminus E^0 \tag{4}$$

$$F^\phi = \{ e \in E \mid 0 < f(e) < 1 \} = E^1 \cap E^0 \tag{5}$$

$$F^0 = \{ e \in E \mid \quad f(e) = 0 \} = E^0 \setminus E^1 \tag{6}$$

$$F^* = \{ e \in E \mid \quad f(e) = * \} = E \setminus (E^1 \cup E^0) \quad . \tag{7}$$

Events from the set F^* correspond to no objects in the sample sets; they are called <u>unspecified events</u>. Events from the set F^ϕ are called <u>mixed</u> events. By assuming some threshold λ, $0 \leq \lambda \leq 1$, we can refer the objects characterized by mixed events to class 1 if $f(e) \geq \lambda$, and to class 0, otherwise. We define:

$$F^{1\lambda} = \{ e \mid f(e) \geq \lambda \} \tag{8}$$

$$F^{0\lambda} = \{ e \mid f(e) < \lambda \} \tag{9}$$

An important concept defined in [1] is that of a literal ${}^{a_i}X_i^{b_i}$. A literal ${}^{a_i}X_i^{b_i}$ is the set of all events $e^j = (x_1, x_2, \ldots, x_n) \in E$, whose x_ith component takes values between a_i and b_i:

$$ {}^{a_i}X_i^{b_i} = \{ (x_1, x_2, \ldots, x_n) \mid a_i \leq x_i \leq b_i \} \tag{10} $$

When $a_i = b_i$ the literal is denoted briefly by $X_i^{a_i}$ and called an elementary literal. A set-theoretic product of literals $L = \bigcap_{i \in I} {}^{a_i}X_i^{b_i}$, $I \subseteq \{ 1, 2, \ldots, n \}$ is called an interval. It constitutes in the space $\mathbf{E}$ an n-dimensional interval, i.e. a set of events lying between two arbitrary vectors. For example, L is a set of events lying between vectors $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_n)$ where, for $i \notin I$, $a_i = 0$ and $b_i = h_i - 1$:

$$ L = \{ e \in E \mid \mathbf{a} \leq e \leq \mathbf{b} \} = \{ (x_1, x_2, \ldots, x_n) \mid \forall i, a_i \leq x_i \leq b_i \} \tag{11} $$

The basic concept of the paper [1] is that of an interval cover of a mapping f. A set $D(f|\lambda) = \{ L_j \}$ of intervals is called an unordered exact interval cover of f under λ (or an interval cover of $F^{1\lambda}$ against $F^{0\lambda}$) if the set-theoretic union of intervals L_j covers every element of $F^{1\lambda}$ and does not cover any element of $F^{0\lambda}$:

$$ F^{1\lambda} \subseteq \bigcup L_j \subseteq F^{1\lambda} \cup F^{*} \tag{12} $$

An interval cover $D(f|\lambda)$ which has the minimum number of intervals is called a minimal* cover of f^{λ} (there can be many equivalent minimal covers). Free and ordered interval covers of a mapping f were also defined in [1]. The present paper, however, will only consider unordered exact interval covers.

---

* In the case when a cost functional is specified for intervals, a minimal cover means a cover of minimal cost.

Sets $F^{1\lambda}$ and $F^{0\lambda}$ define a mapping f^{λ}:

$$f^{\lambda}: \quad E \longrightarrow \{0,1,*\} \tag{13}$$

where: $\{e \mid f^{\lambda}(e) = 1,0,*\} = F^{1\lambda}, F^{0\lambda}, F^{*}$ respectively.

If $F^{*} = \phi$, the mapping F^{λ} can be expressed in a notation similar to that used to represent a Boolean function in switching theory:

$$f^{\lambda} = \bigvee_{L_j \in D(f)} |L_j| \tag{14}$$

where:

$|L_j|$ denotes the characteristic function of the interval L_j in $\mathbf{E}$, i.e.

$$|L_j|(e) = \begin{cases} 1, \text{ if } e \in L_j \\ 0, \text{ otherwise} \end{cases}$$

$D(f)$ is an interval cover of $F^{1\lambda}$ against $F^{0\lambda}$,

$\bigvee$ denotes disjunction (logical sum).

Expression (14) parallels the disjunctive normal form (sum-of-products) of a Boolean function. It will therefore be called the <u>interval disjunctive normal form</u> of f^{λ} (alternatively, of f <u>under</u> λ).

Similarly as in switching theory we can express f^{λ} in another form, corresponding to the conjunctive normal form of a Boolean function:

$$f^{\lambda} = \bigwedge_{L_j \in K(f)} S(L_j)$$

where $S(L_j)$ is the disjunction of the characteristic functions of literals obtained by applying de Morgan's laws to the complement of the interval L_j. For example, if $L_j = \bigcap_{i \in I} {}^{a_i}X_i^{b_i}$, $I \subseteq \{1,2,\ldots,n\}$, then, $S(L_j) = |\overline{L}_j| = \left| \bigcup_{i \in I} \overline{{}^{a_i}X_i^{b_i}} \right| = \bigvee_{i \in I} \left(|{}^{0}X_i^{a_i-1}| \vee |{}^{b_i+1}X_i^{n_i-1}| \right)$

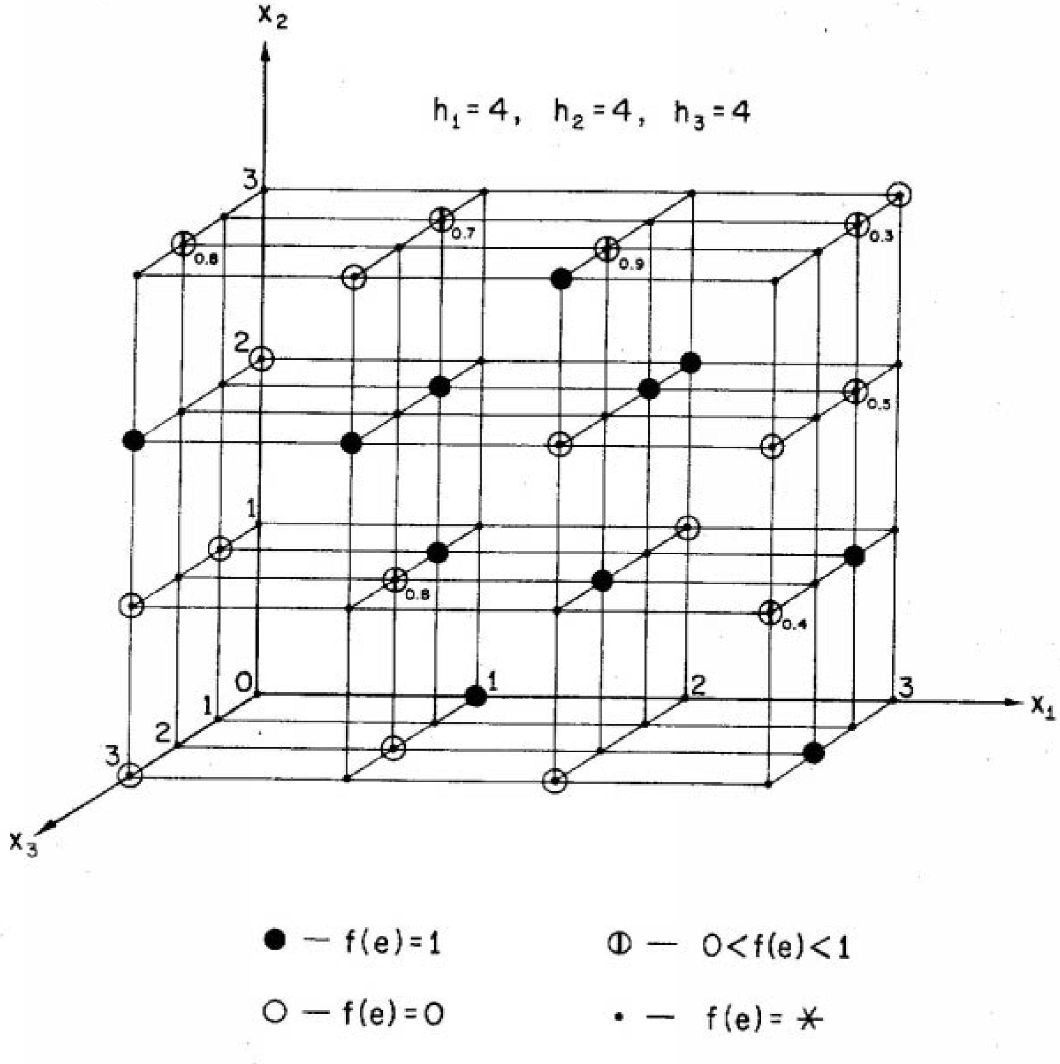


Fig. 1. Space E(4,4,4) and a mapping f: E(4,4,4) → {[0,1],*}

(for a detailed explanation of the rules of the transformation see [1]).

$K(f)$ is a cover of $F^{0\lambda}$ against $F^{1\lambda}$.

The above expression, called a <u>interval conjunctive normal expression</u> of f^{λ} is formed by applying de Morgan's laws to the equation $F^{1\lambda} = \overline{F^{0\lambda}}$.

4. GENERALIZED LOGICAL DIAGRAM (GLD) AND A FUNCTION IMAGE T(f)

A discrete-euclidean geometrical representation of the space E would be in the form of an n-dimensional 'grid', spanned from the $h_1, h_2, \cdots, h_n$ points on axes $x_1, x_2, \ldots, x_n$, respectively. A mapping f could then be represented by assigning values $f(e^j)$ to the nodes corresponding to vectors e^j (fig. 1).

The above geometrical model of the space E is, however, not easily visualized when $n > 3$. We therefore introduce another representation, a planar one, which may be extended to any value of n.

Let us divide an arbitrary rectangle into $h_1 h_2 \cdots h_v$ rows and $h_{v+1} h_{v+2} \cdots h_n$ columns, where v is the maximal value for which $h_1 h_2 \cdots h_v \leq H/2$, and assign vectors $(x_1, x_2, \ldots, x_v)$, $x_i \in \{0, 1, \ldots, h_i\}$, $i \leq v$, to the rows, and vectors $(x_{v+1}, x_{v+2}, \ldots, x_n)$, $x_i \in \{0, 1, \ldots, h_i\}$, $v < i \leq n$ to the columns, according to the rules:

1. In the first step the rectangle is divided by horizontal lines into h_1 rows which, in order from top to bottom, are assigned values $0, 1, \ldots, h_1-1$, respectively (values of the component x_1 of vectors $e \in E$). In step i each row generated in step i-1 is divided into h_i rows. The rows, in order from top to bottom, within each row generated in step i-1, are assigned values $0, 1, \ldots, h_i-1$ respectively (the values of the component x_i).

In toto, v steps are executed.

2. Steps v+1, v+2, ..., n are executed analogously to steps 1, 2, ..., v, but now the rectangle is divided by vertical lines into columns and the columns are assigned values in order from left to right.

The lines which divide the rectangle in step i, $i \in \{1,2,\dots,n\}$ are called axes of the component x_i (axes of different components are graphically distinguished by different thicknesses of the lines).

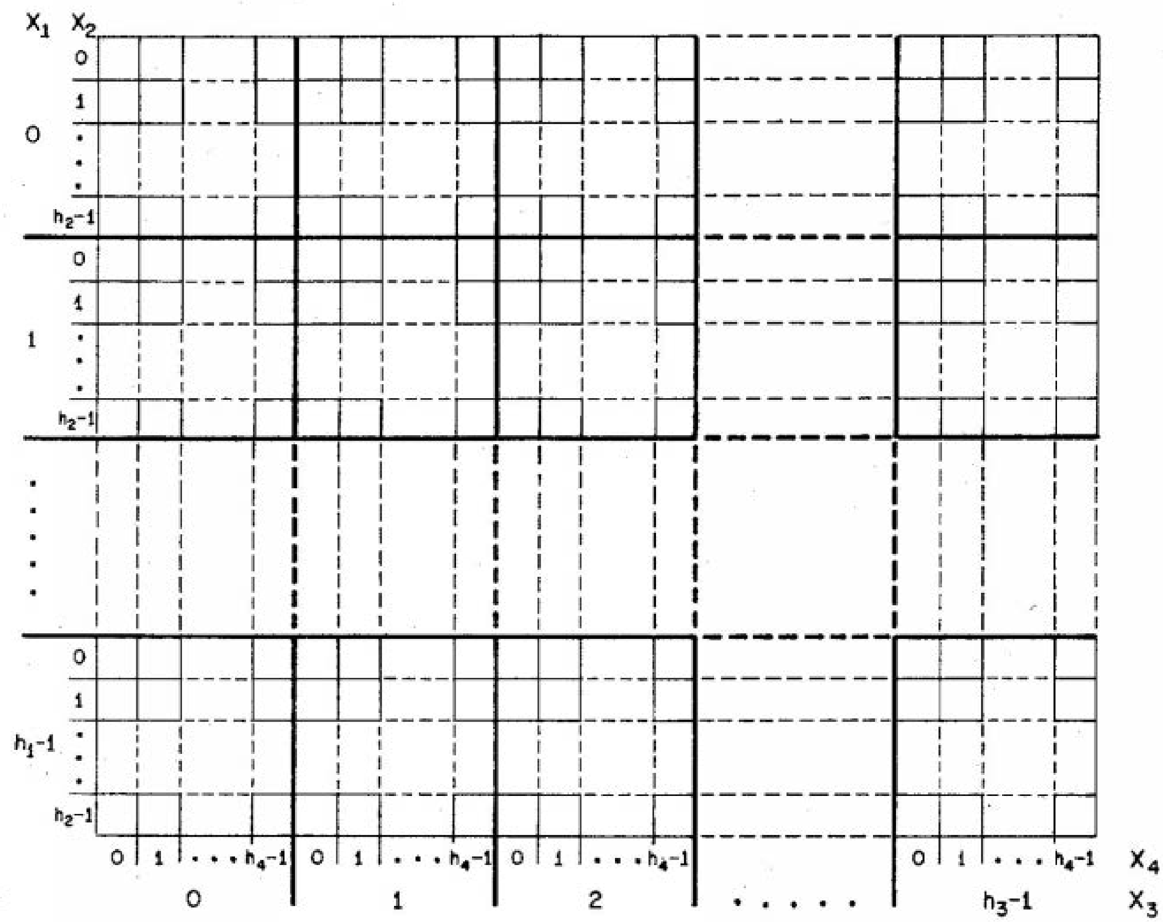
The diagram so obtained, is called a generalized logical diagram (GLD). Fig. 2. illustrates the GLD for n = 4.

A unique vector $(x_1,x_2,\dots,x_v)$ corresponds to each row in the GLD and a unique vector $(x_{v+1},x_{v+2},\dots,x_n)$ corresponds to each column. The intersection of any row with any column is called a cell of the diagram. To be precise, we will assume that the cells do not include points belonging to any axis nor to the perimeter of the rectange. The obtained diagram comprises $H = h_1h_2\cdots h_2$ cells (number of events in **E**). Each cell of the diagram represents a unique event e of **E**, determined by concatenating vectors $(x_1,x_2,\dots,x_v)$ and $(x_{v+1},x_{v+2},\dots,x_n)$ which correspond to row and column respectively, such that their intersection is the given cell.

Thus the GLD is a geometrical model of the space **E**.

Cells of the GLD will also be denoted by e^j and it will be clear from the context if e^j denotes an event or a cell. If e^j denotes a cell e, then the index $j = \gamma(e)$ will be called the number of the cell e.

It can easily be verified that the numbers of the cells are distributed in a GLD, in lexicographical order, i.e. from left to right and from top to bottom, as is shown for **E**(4,3,4,2) in fig. 3. If **E** is the



$\max_i \; h_1 h_2 \cdots h_i \leq h_{i+1} h_{i+2} \cdots h_n$

Fig. 2. Generalized Logical Diagram representing $E(h_1, h_2, h_3, h_4)$

$h_1 = 4$, $h_2 = 3$, $h_3 = 4$, $h_4 = 2$

X_3		0	0	1	1	2	2	3	3
X_1	X_2 \ X_4	0	1	0	1	0	1	0	1
0	0	0	1	2	3	4	5	6	7
	1	8	9	10	11	12	13	14	15
	2	16	17	18	19	•	•	•	•
1	0	•	•	•	•				
	1								
	2								
2	0								
	1								
	2					•	•	•	•
3	0	•	•	•	•	76	77	78	79
	1	80	81	82	83	84	85	86	87
	2	88	89	90	91	92	93	94	95

Fig. 3. Distribution of cell numbers in the GLD for E(4,3,4,2)

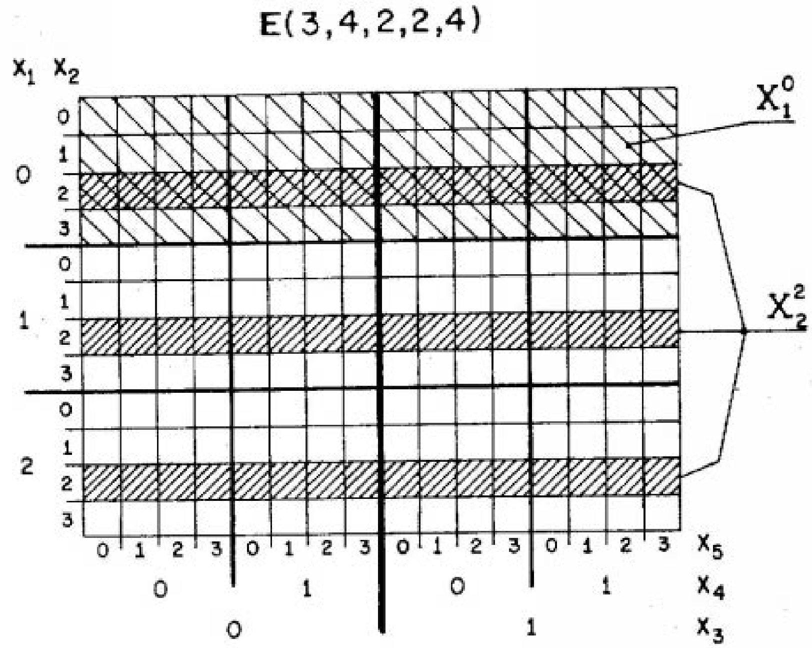


Fig. 4. The GLD representation of X_1^0 and X_2^2

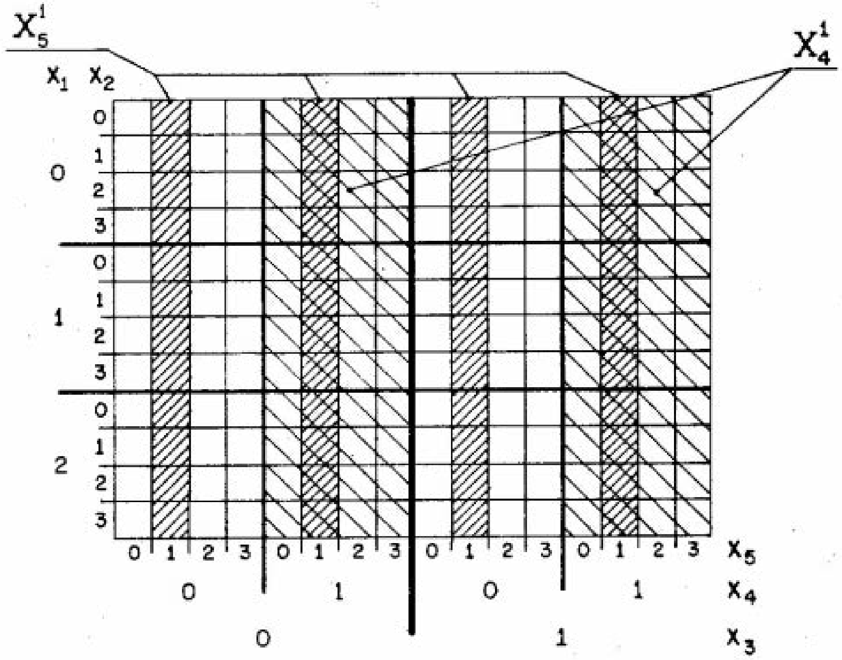


Fig. 5. The GLD representation of X_5^1 and X_4^1

space of binary vectors, i.e. $E(2,2,\dots,2)$, the GLD becomes the <u>logical diagram</u> defined in [2]. The logical diagram of [2] is similar to a diagram for the solution of logical problems first constructed by Alan Marquand in 1881 [3]*. The GLD can therefore be viewed as an extension of Marquand's idea.

It can easily be verified that an elementary literal X_1^k $k \in \{0,1,\dots,h_1-1\}$ is represented in the GLD by a set of cells contained in the row generated in step 1 and assigned the value k. An elementary literal X_i^k, $i \in \{2,3,\dots,v\}$, $k \in \{0,1,2,\dots,h_i-1\}$ is represented by the union of cell sets comprising cells from the rows generated in step i and assigned the value k (fig. 4). The literals X_i^k, $i = v+1,v+2,\dots,n$, $k = 0,1,\dots,h_i-1$, are represented analogously (fig. 5).

Because

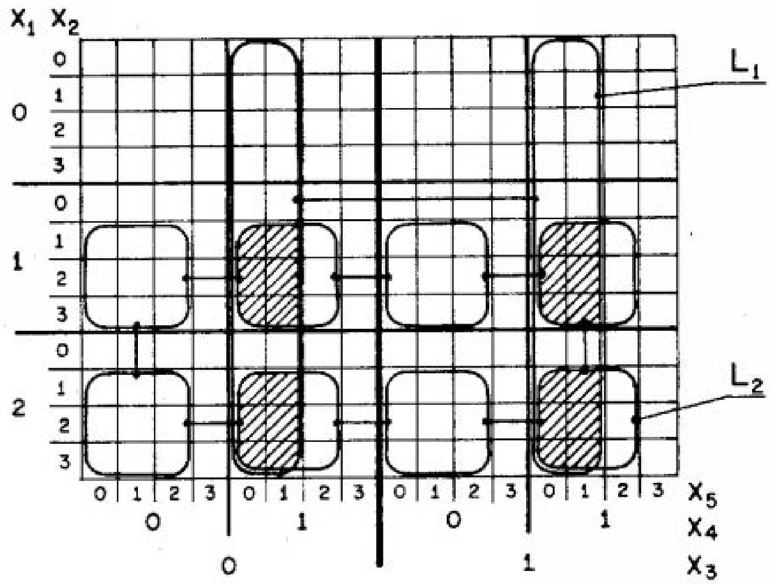
$$ {}^{a_i}X_i^{b_i} = \bigcup_{k=a_i}^{b_i} X_i^k, \tag{15}$$

a non-elementary literal ${}^{a_i}X_i^{b_i}$ is represented by the union of cell sets corresponding to the elementary literals $X_i^{a_i}$, $X_i^{a_i+1}$, ..., $X_i^{b_i}$.

Generally, the set-theoretic operations on any event sets (thus also on literals and intervals, i.e. products of literals) are equivalent

---

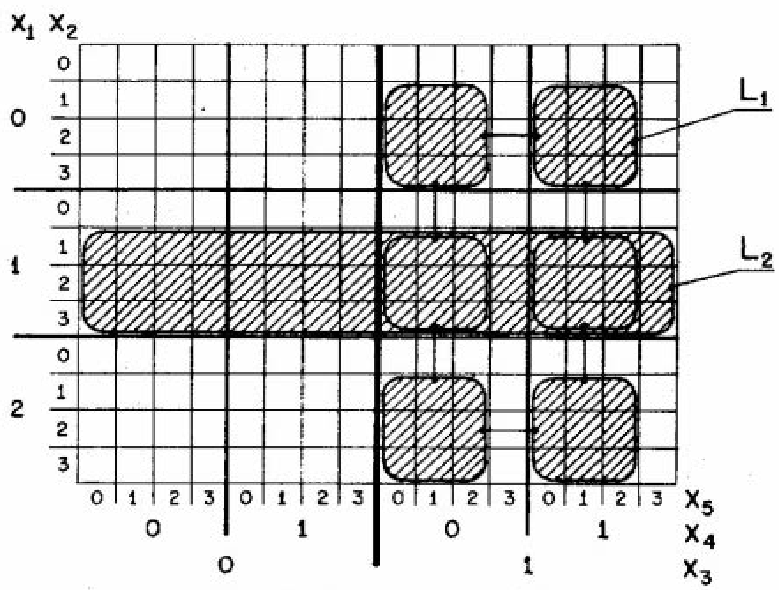
* 1) 'Charts' of E. W. Veitch [4], ~~discovered or~~ rediscovered in 1952, are diagrams of the same kind, constructed for simplifying switching function expressions. They have not become as popular as Karnaugh maps (in which rows and columns are ordered according to Grey's code instead of the natural binary code) because simple rules for detecting sets of cells corresponding to prime implicants were lacking. However, if such rules are formulated [6, see also section 5] and axes of different variables are distinguished by differing their thicknesses (as in the diagrams of [6,2] and in the GLD), the diagrams become highly useful for simplifying switching function expressions as well as for detecting switching function symmetry [2] and for converting normal Boolean expressions into exclusive-or-polynomial expressions and vice versa [7]. Important properties of these diagrams are that all the rules remain the same for any number of variables and that the cell numbers have lexicographical order (which Karnaugh's maps do not have).



$L_1 = X_4^1\,{}^0X_5^3 \qquad L_2 = {}^1X_1^2\,{}^1X_2^3\,{}^0X_5^2$

denotes $L_1 \cap L_2$

Fig. 6. An intersection of interval complexes



$L_1 = {}^1X_2^3\,X_3^1\,{}^0X_5^2 \qquad L_2 = X_1^1\,{}^1X_2^3$

denotes $L_1 \cup L_2$

Fig. 7. A union of interval complexes

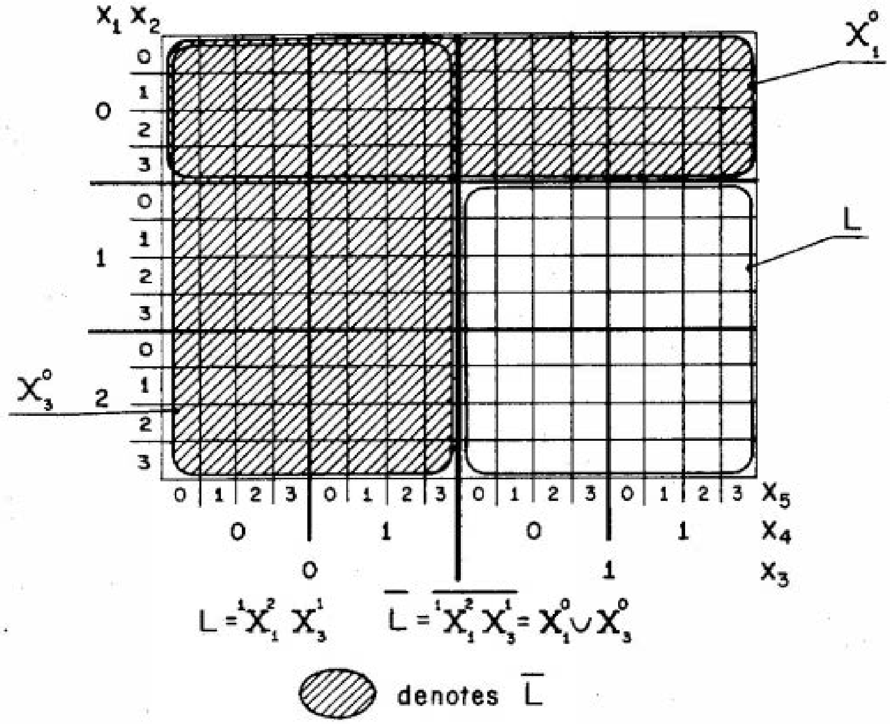


Fig. 8. Complement of an interval complex

to the set-theoretic operations on the cell sets representing them. For convenience, we will preserve the same notation for intervals and the cell sets which represent them. However, to distinguish the intervals from their geometrical equivalents we will call the latter _interval complexes_. Fig. 6, 7 and 8 give examples of an intersection, a union and a complement of interval complexes, respectively.

To represent a mapping $f: E \rightarrow \{ [0,1], * \}$, for each event $e \in E$, the value $f(e)$ is assigned to the cell representing e. The set of cells of the GLD with their assigned values is called a _function image_ of f and denoted by $T(f)$. The function image $T(f)$ which corresponds to the 'grid' representation of f in fig. 1, is shown in fig. 9. A cover $D(f|\lambda)$ of a mapping f under λ can be represented by marking in $T(f)$ the interval complexes corresponding to the intervals of the cover.

Using $T(f)$, we can in simple cases (when the GLD is not too large) visually determine a cover $D(f|\lambda)$. First, the set F^{ϕ} of cells with values $f(e)$, $0 < f(e) < 1$, is partitioned into the sets $F^{\phi 1} = \{ e | \lambda \leq f(e) < 1 \}$ and $F^{\phi 0} = \{ 0 \mid 0 < f(e) < \lambda \}$ and the sets $F^{1\lambda} = F^{1} \cup F^{\phi 1}$ and $F^{0\lambda} = F^{0} \cup F^{\phi 0}$ are determined (e.g., by changing the values of the cells of $F^{\phi 1}$ to 1 and those of $F^{\phi 0}$ to 0). Then we visually find a set of interval complexes whose union covers every cell of $F^{1\lambda}$ and does not cover any cell of $F^{0\lambda}$ (it may, however, cover cells of F^{*}). Section 5 gives a rule for the easy recognition of cell sets which are interval complexes. The above procedure for determining a cover, based on the intuitive grouping of cells into interval complexes, parallels the graphical way of finding a simplified disjunctive normal form of a Boolean function in switching theory. It does not guarantee that the obtained cover is minimal; in general it gives an approximately minimal one, without, however, any estimation of the distance

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			1			0.8	
1	1		0			1	0.8			1		1		0.7		0
2				0	0		1		1	1		0			0.9	1
3			1			1		0.4		0.5		0	0	0.3		

Fig. 9. T(f) of the mapping f from Fig. 1. The empty cells have value *

(in terms of the number of intervals) to the minimal cover. The minimal, or approximately minimal cover with an estimate of the maximal possible distance to the minimum can be found by applying the algorithm A^Q (see section 7).

5. RECOGNITION OF INTERVAL COMPLEXES IN THE GLD

The synthesis of a minimal interval cover is computationally a very complex problem, particularly if the number of dimensions n and the number of distinct values for each dimension h_i, are not small numbers, e.g. $n \geq 8$ and $h_i \geq 4$. Therefore, the synthesis of interval covers normally has to be performed by a computer. However, for checking the results and understanding the synthesis algorithms, a geometrical representation of a mapping f by a function image T(f) and of its interval cover, provided by the GLD, is very useful. Also, in simple cases, the image T(f) can be directly used for (graphical) synthesis of an interval cover. This can be simplified by having rules for easy recognition of interval complexes in the GLD.

In order to formulate such rules, we shall first define some simple geometrical objects in the GLD.

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. 10).

The intersection of any regular row and any regular column is called a regular rectangle (fig. 11).

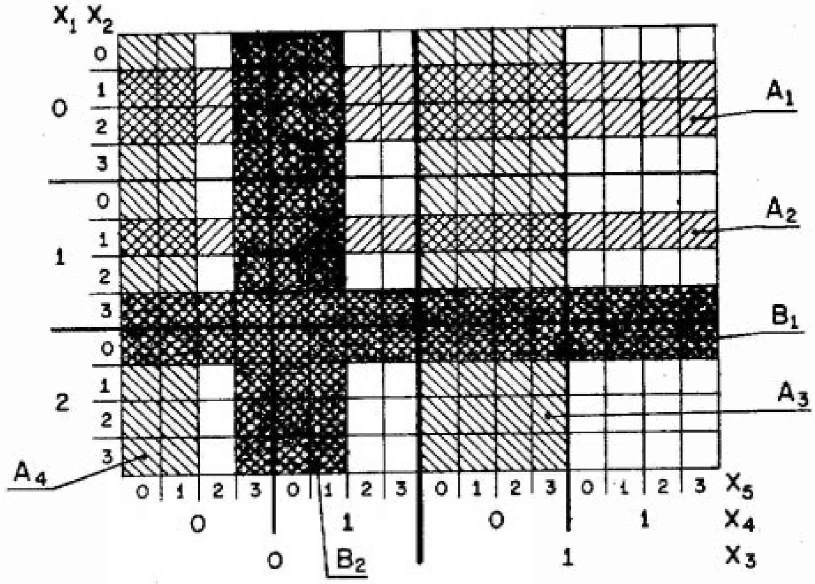


Fig. 10. A_1,A_2 - regular rows, A_3,A_4 - regular columns
B_1 - not a regular row, B_2 - not a regular column

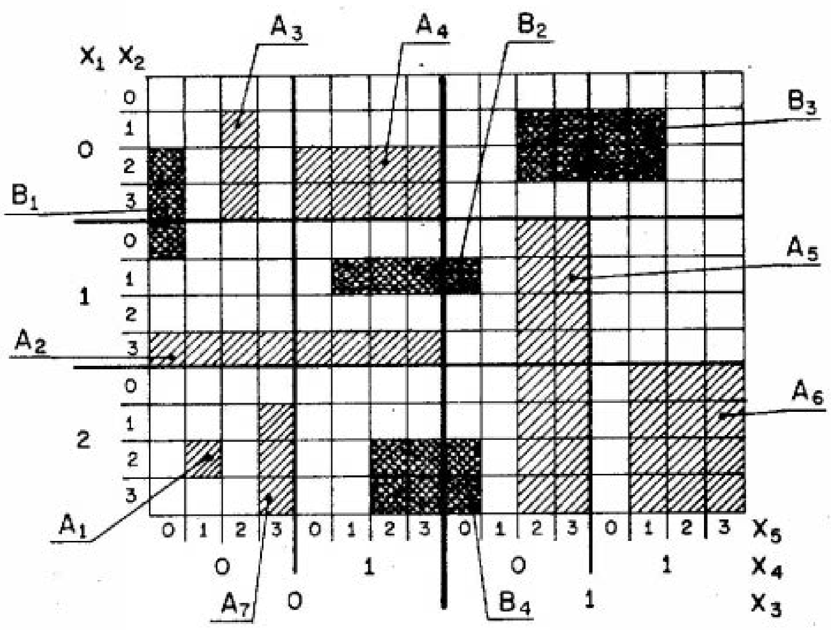


Fig. 11. A_1,A_2,...,A_7 - regular rectangles
B_1,B_2,B_3,B_4 - not regular rectangles

Regular rectangles which can be made to cover each other by translation are called identical.

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 regular rectangle which includes E), is called a covering rectangle for E and is denoted by R(E) (fig. 12).

Let R_1 and R_2 be identical regular rectangles containing event sets E_1 and E_2, respectively. E_1 is said to have the same placement in R_1 as E_2 in R_2, if, when R_1 and R_2 are superimposed E_1 and E_2 cover each other. E_1 and E_2 having the same placement in R_1 and R_2 respectively, can be expressed by:

$$E_1 = R_1 \wedge L \text{ and } E_2 = R_2 \wedge L \tag{16}$$

where L is an interval complex, called an interval complex addressing E_1 in R_1 (notation: $L(E_1,R_1)$), or E_2 in R_2 $(L(E_2,R_2))$. For example, in fig.13 the set E_1 has the same placement in R_1 as E_2 in R_2 and we have:

$$E_1 = \overbrace{X_1^1\ X_3^0\ X_4^0\ {}^1X_5^2}^{R_1}\ \overbrace{X_2^1}^{L\,=\,L(E_1,R_1)}$$

$$E_2 = \underbrace{X_1^0\ X_3^0\ X_4^0\ {}^1X_5^2}_{R_2}\ \underbrace{X_2^1}_{L\,=\,L(E_2,R_2)}$$

Now we will formulate a theorem which gives a rule for recognizing interval complexes in the GLD. Let $E_1, E_2, \ldots, E_k$ be some event sets.

Theorem 1:

The union $E_1 \vee E_2 \vee \ldots \vee E_k$ is an interval complex, if $E_1, E_2, \ldots, E_k$ are interval complexes and the covering rectangle $R(E_1 \vee E_2 \vee \ldots \vee E_k)$ can

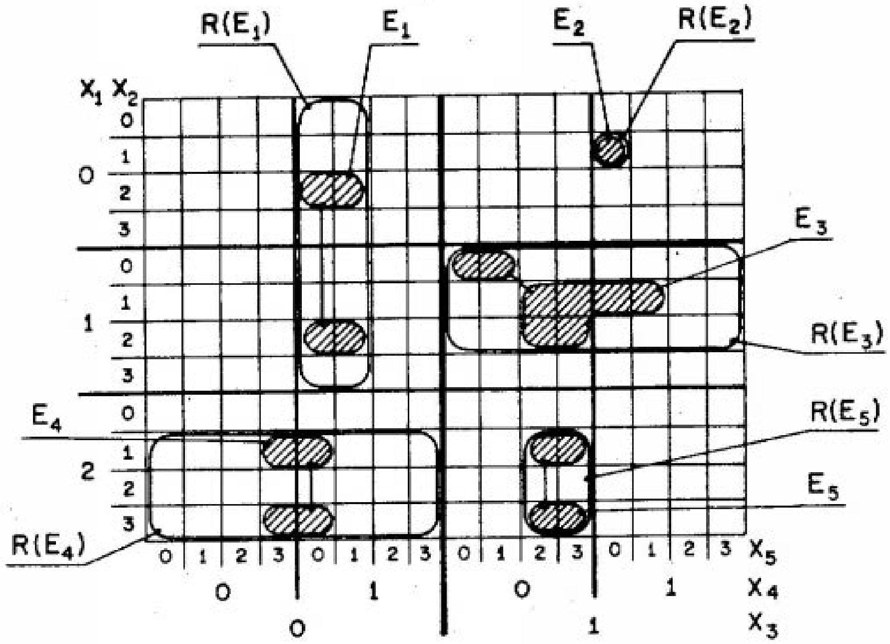


Fig. 12. Event sets E_i and their covering rectangles $R(E_i)$, $i = 1,2,\ldots,5$

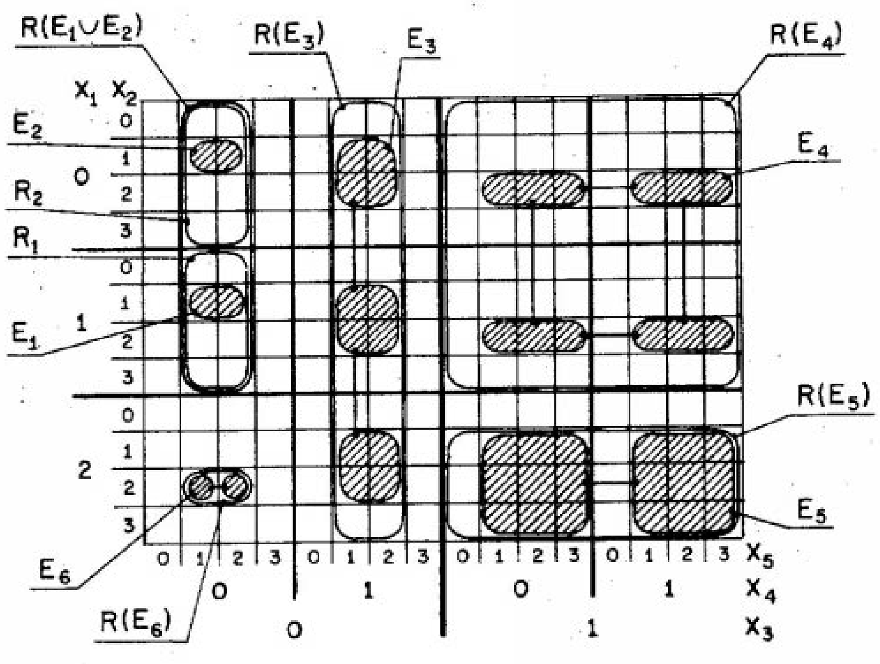


Fig. 13. $E_1 \cup E_2, E_3, E_4, E_5$ are interval complexes (Theorem 1)

be partitioned into k identical regular rectangles $R_1, R_2, \dots, R_k$ in which $E_1, E_2, \dots, E_k$ have the same placement, respectively.

Proof

If $E_1, E_2, \dots, E_k$ are interval complexes and have the same placement in $R_1, R_2, \dots, R_k$, respectively, then we have:

$$E_1 = R_1 \cap L,\ E_2 = R_2 \cap L,\ \dots,\ E_k = R_k \cap L$$

where :

$$L = L(E_1, R_1) = L(E_2, R_2) = \dots = L(E_k, R_k).$$

$$\bigcup_{i=1}^{k} E_i = (R_1 \cap L) \cup (R_2 \cap L) \cup \dots \cup (R_k \cap L) = L \cap (R_1 \cup R_2 \cup \dots \cup R_k)$$

But $(R_1 \cup R_2 \cup \dots \cup R_k)$ is the covering rectangle $R(E_1 \cup E_2 \cup \dots \cup E_k)$, therefore

$$\bigcup_{i=1}^{k} E_i = L \cap R(E_1 \cup E_2 \cup \dots \cup E_n)$$

The intersection of interval complexes is also an interval complex, thus $\bigcup_{i=1}^{k} E_i$ is an interval complex. Q.E.D.

Theorem 1 is illustrated in fig. 13.

6. THE EXTENSION OPERATION $\surd$ AND A STAR $G(e|\lambda)$

An important concept introduced in [1] is that of the extension operation $\surd$ on event sets. Here we will give the GLD interpretation of this operation.

Recall from [1] that $\{E_1, E_2, \dots\}^{\vee}$ denotes the union of event sets E_i, $i = 1,2,\dots$ and $\sqrt[\vee]{E}$ the set of all maximal (under inclusion) intervals contained in E:

$$\{E_1, E_2, \ldots\}^{\cup} = \bigcup_i E_i \tag{17}$$

$$\sqrt[\cup]{E} = \{L_j \mid L_j \subseteq E \text{ and } \nexists L'_j \subseteq E, L_j \subset L'_j\} \tag{18}$$

Let E_1 and E_2 be event sets. The <u>extension</u> <u>operation</u> $\vee\!\!\!-$ <u>on</u> E_1 <u>relative</u> to E_2 (or the <u>interval</u> <u>extension</u> of E_1 <u>in</u> E_2) is defined:

$$E_1 \vee\!\!- E_2 = \{L_j \mid L_j \subseteq E_2 \text{ and } L_j \cap E_1 \neq \phi\}^{\cup} \tag{19}$$

Clearly, $E_1 \vee\!\!- E_2 \subseteq E_2$ (if $E_1 \cap E_2 = \phi$, $E_1 \vee\!\!- E_2 = \phi$).

If in (19) the set $\{L_j\}$ includes only maximal (under inclusion) intervals which are contained in E_2 and have a non-empty intersection with E_1, the union $\{L_j\}^{\cup}$ will be the same. Therefore, we have the equivalent definition:

$$E_1 \vee\!\!- E_2 = \Lambda^{\cup} \tag{20}$$

where $\Lambda = \{L_j \mid L_j \in \sqrt[\cup]{E_2} \text{ and } L_j \cap E_1 \neq \phi\}$

Thus we have:

$$\sqrt[\cup]{E_1 \vee\!\!- E_2} = \Lambda \tag{21}$$

To form $E_1 \vee\!\!- E_2$ using the GLD, mark the intersection $E_1 \cap E_2$ and then, by applying theorem 1, find all the maximal interval complexes which cover at least one event $e \in E_1 \cap E_2$ and are contained in E_2. The union of these interval complexes is $E_1 \vee\!\!- E_2$ (fig. 14).

If E_2 is an interval complex L, then clearly:

$$E_1 \vee\!\!- L = \begin{cases} L, & \text{if } E_1 \cap L \neq \phi \\ \phi, & \text{otherwise} \end{cases} \tag{22}$$

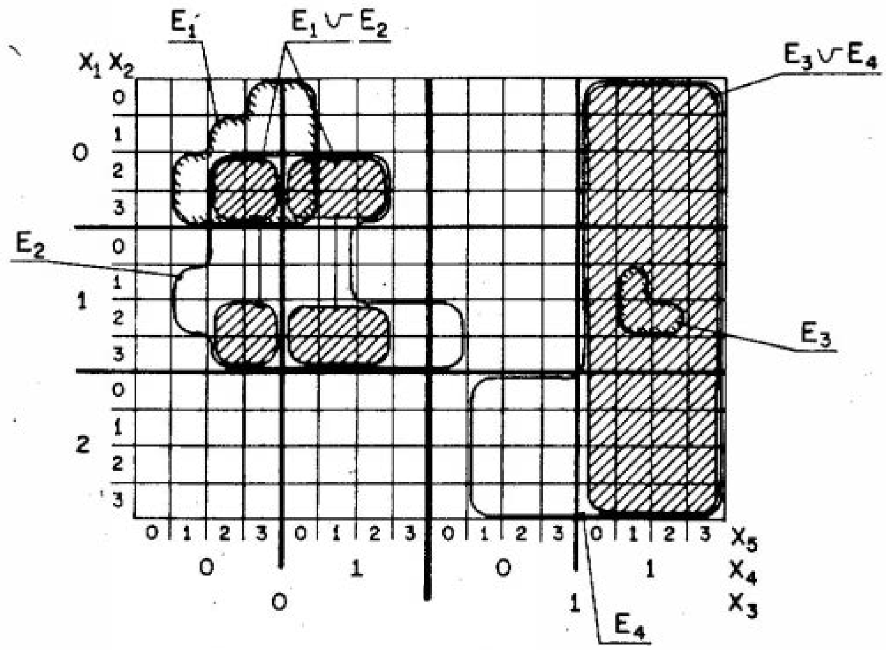


Fig. 14. Extension operation $\vee$

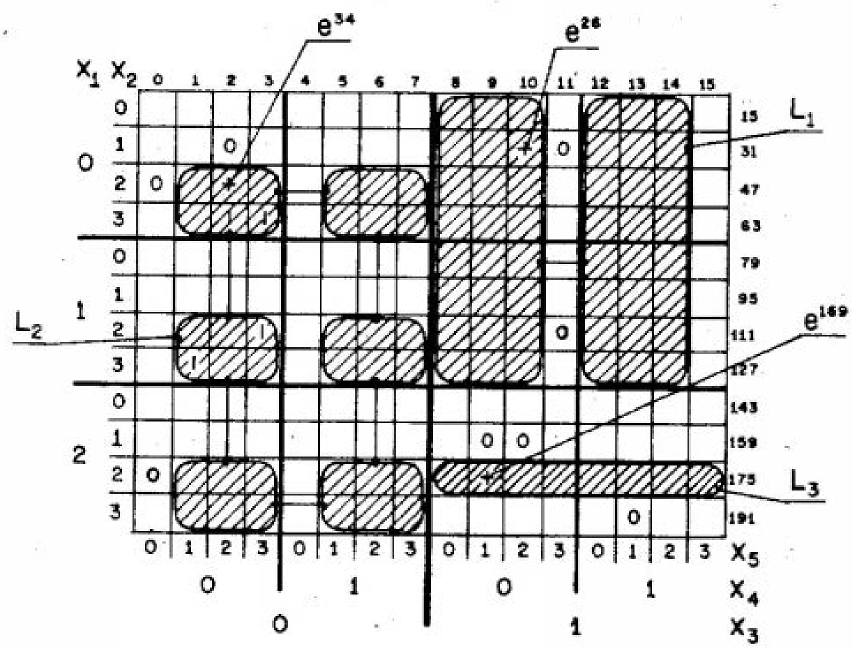


Fig. 15. Examples of maximal interval complexes.
The empty cells have value *

Interval complexes

$$L_j^{\lambda} \in \sqrt[\cup]{F^{1\lambda} \smile \overline{F^{0\lambda}}} \tag{23}$$

are called <u>maximal</u> <u>interval</u> <u>complexes</u> <u>in</u> T(f) <u>under</u> λ (or in $T(f^{\lambda})$).
In other words, maximal interval complexes L_j^{λ} are maximal under inclusion interval complexes which cover some events from $F^{1\lambda}$ and do not cover any event from $F^{0\lambda}$, i.e. they are contained in $F^1 \cup F^*$. The L_i (i = 1,2,3) in fig. 15 are examples of maximal interval complexes in a function image $T(f^{\lambda})$. They were generated by starting with some arbitrary cell $e_i \in F^{1\lambda}$ (where $e_1 = e^{26}, e_2 = e^{34}, e_3 = e^{169}$) and then adding more cells from $F^1 \cup F^*$, as long as their union is still an interval complex according to theorem 1.

In general, there can be many different maximal interval complexes in a $T(f^{\lambda})$ which cover the same event $e \in F^{1\lambda}$. The set of all maximal interval complexes in $T(f^{\lambda})$ covering a given event $e \in F^{1\lambda}$ is called the <u>interval</u> <u>star</u> $G(e|\lambda)$ <u>under</u> λ of that event. Clearly:

$$G(e|\lambda) = \sqrt[\cup]{\{e\} \smile \overline{F^{0\lambda}}} \tag{24}$$

The concept of an interval star is fundamental in our approach to the synthesis of quasi-minimal interval covers described in Section 7. We will here briefly explain the algorithm G of [1] for generating a star $G(e|\lambda)$ and illustrate it by a simple example.

Denoting $(G(e|\lambda))^{\cup}$ by $G^{\cup}(e|\lambda)$, we have from (24):

$$G^{\cup}(e|\lambda) = \{e\} \smile \overline{F^{0\lambda}} \tag{25}$$

which is equivalent (see theorem 13 of [1]) to:

$$G^{\cup}(e|\lambda) = \bigcap_{e_k \in F^{0\lambda}} (\{e\} \smile \{e_k\}) \tag{26}$$

Equation (26) is the theoretical basis for algorithm G.

First, the second de Morgan's law is applied to $\{\overline{e_k}\}$, $e_k \in F^{0\lambda}$:

$$\{\overline{e_k}\} = \overline{X_1^{a_1} X_2^{a_2} \cdots X_n^{a_n}} = \overline{X_1^{a_1}} \cup \overline{X_2^{a_2}} \cup \cdots \cup \overline{X_n^{a_n}} \qquad (27)$$

and then the $\overline{X_i^{a_i}}$ are expressed as:

$$\overline{X_i^{a_i}} = {}^{0}X_i^{a_i-1} \cup {}^{a_i+1}X_i^{h_i} \qquad (28)$$

By applying the distribution property of $\bar{\cup}$ over $\cup$ ([1]) and the rule (22), each $(\{e\} \bar{\cup} \{\overline{e_k}\})$ in (26) is transformed into a sum of literals, denoted by S_k. Then, by multiplying the S_k by each other and applying the absorption laws:

$$L(L \cup L_a \cup L_b \cup \cdots) = L \qquad (29)$$

$$L \cup (L\, L_a\, L_b \cdots) = L \qquad (30)$$

(where $L, L_a, L_b, \cdots$ are any literals or products of literals), the $G^{\cup}(e|\lambda)$ is expressed as the irredundant sum of intervals:

$$G^{\cup}(e|\lambda) = L_1 \cup L_2 \cup L_3 \cup \cdots . \qquad (31)$$

The star $G(e|\lambda)$ is then:

$$G(e|\lambda) = \{L_j\},\ j = 1,2,\ldots \qquad (32)$$

<u>Example 1</u>

Generate the star $G(e|\lambda)$ of the event $e = e^{39} \in \mathbf{E}(3,3,3,3)$ if $F^{0\lambda} = \{e_k\}_{k=1}^{5} = \{e^{21}, e^{37}, e^{44}, e^{67}, e^{75}\}$.

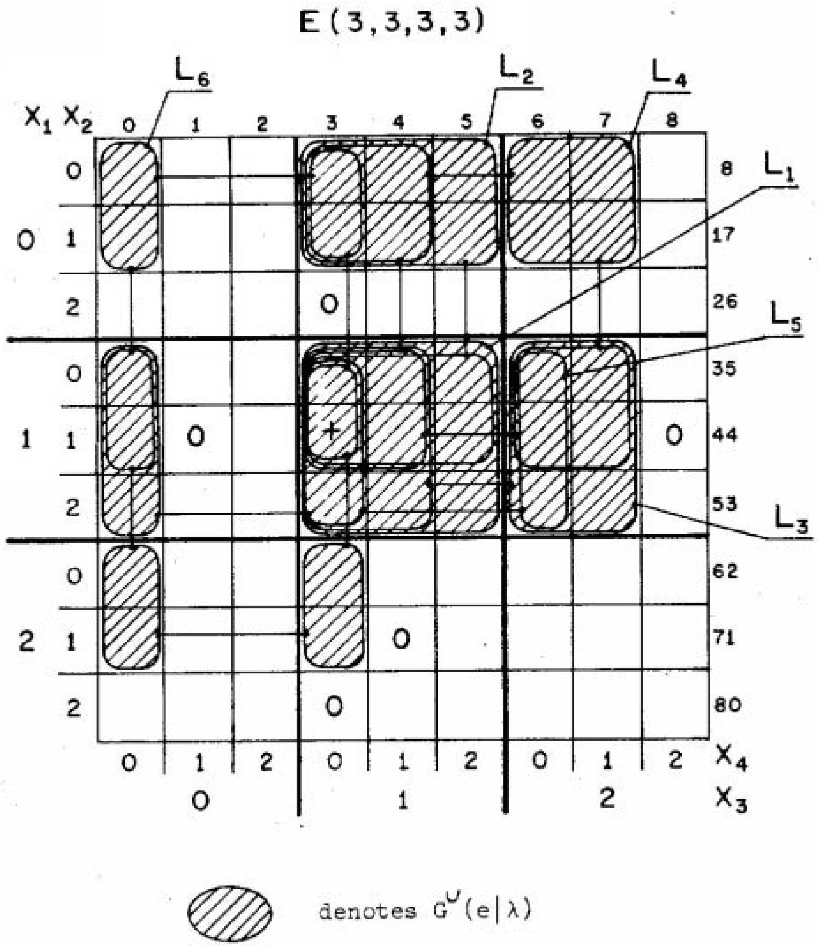


Fig. 16. The star $G(e|\lambda)$ of event $e = e^{39}$ (Example 1)

From the GLD for $\mathbf{E}(3,3,3,3)$ the products of literals corresponding to the events e and e_k, $k = 1,2,\ldots,5$, are determined, and then the sums S_k are formed:

$$\{e^{39}\} = X_1^1\ X_2^1\ X_3^1\ X_4^0$$

$$
\begin{array}{lll}
\{e_1\} = X_1^0\ X_2^2\ X_3^1\ X_4^0\ , & S_1 = \{e\}\cup\overline{\{e_1\}} & = {}^1X_1^2 \cup {}^0X_2^1 \\
\{e_2\} = X_1^1\ X_2^1\ X_3^0\ X_4^1\ , & S_2 = \{e\}\cup\overline{\{e_2\}} & = {}^1X_3^2 \cup \ X_4^0 \\
\{e_3\} = X_1^1\ X_2^1\ X_3^2\ X_4^2\ , & S_3 = \{e\}\cup\overline{\{e_3\}} & = {}^0X_3^1 \cup {}^0X_4^1 \\
\{e_4\} = X_1^2\ X_2^1\ X_3^1\ X_4^1\ , & S_4 = \{e\}\cup\overline{\{e_4\}} & = {}^0X_1^1 \cup \ X_4^0 \\
\{e_5\} = X_1^2\ X_2^2\ X_3^1\ X_4^0\ , & S_5 = \{e\}\cup\overline{\{e_5\}} & = {}^0X_1^1 \cup {}^0X_2^1
\end{array}
$$

According to (26):

$$G^{\vee}(e|\lambda) = \bigcap_{k=1}^{5} S_k = ({}^1X_1^2 \cup {}^0X_2^1)({}^1X_3^2 \cup X_4^0)({}^0X_3^1 \cup {}^0X_4^1)({}^0X_1^1 \cup X_4^0)({}^0X_1^1 \cup {}^0X_2^1) \qquad (33)$$

After multiplication with the use of the rule:

$$
{}^{a_1}X_{i_1}^{b_1}\ {}^{a_2}X_{i_2}^{b_2} = \begin{cases} {}^{a}X_i^{b} & \text{, if } i_1 = i_2 = i \text{ and } a \leq b \\ \phi & \text{, if } i_1 = i_2 = i \text{ and } a > b \\ {}^{a_1}X_{i_1}^{b_1}\ {}^{a_2}X_{i_2}^{b_2} & \text{, if } i_1 \neq i_2 \end{cases} \qquad (34)
$$

where $a = \max(a_1,a_2)$ and $b = \min(b_1,b_2)$, and the absorption laws, we have:

$$G^{\vee}(e|\lambda) = \bigcup_{j=1}^{5} L_j \qquad (35)$$

where

$$L_1 = X_1^1\ X_3^1 \quad , \quad L_2 = {}^0X_1^1\ {}^0X_2^1\ X_3^1 \ , \quad L_3 = X_1^1\ {}^1X_3^2\ {}^0X_4^1$$

$$L_4 = {}^0X_1^1\ {}^0X_2^1\ {}^1X_3^2\ {}^0X_4^1, \quad L_5 = X_1^1\ X_4^0 \quad , \quad L_6 = {}^0X_2^1\ X_4^0$$

Thus the star $G(e|\lambda) = \{L_j\}_{j=1}^{6}$ (see fig. 16).

Algorithm G was developed for the machine synthesis of interval covers. The most tedious part of the algorithm, the transformation from $G^{\cup}(e|\lambda) = (S_1 S_2 \cdots)$ to (31) (in the example above from (33) to (35)), can be simplified by using the special rules described in [8]. When n and h_i are small, a star $G(e|\lambda)$ usually consists of only a few elements and therefore can be determined graphically without using the algorithm, simply by visual inspection of the function image T(f).

7. GRAPHICAL SYNTHESIS OF QUASI-MINIMAL INTERVAL COVERS

Paper [1] described an adaptation of the algorithm A^q [9](which provides a solution for the general covering problem) to the synthesis of quasi-minimal interval covers. From now on we will denote interval adaptations of A^q by A^q-INTERCOVER or, briefly, by A^Q.

In this section we will geometrically describe a version of A^Q, using the GLD and will give examples of the graphical synthesis of interval covers. The version described here differs slightly from the one in [1] in that we assume the criterion of cover minimality be not only the number of intervals (as in [1]), but also, with secondary priority, the number of literals.

The flow-diagram of our version of A^Q is given in fig. 17.

Input to the algorithm is a function image $T(f)$ and an assumed value λ. The output is a quasi-minimal interval cover $M^q(f|\lambda)$ (equal to the last value of the variable M^q), and parameters Δ and δ which estimate the maximal possible difference between the number of intervals and literals, respectively, in the cover $M^q(f|\lambda)$ and in the minimal cover $M(f|\lambda)$:

$$c(M^q(f|\lambda)) - c(M(f|\lambda)) \leq \Delta \tag{36}$$

$$z(M^q(f|\lambda)) - z(M(f|\lambda)) \leq \delta \tag{37}$$

By $c(M)$, $z(M)$ is denoted the number of intervals and number of literals respectively, in the cover M.

$:=$	denotes the assignment statement,
F^p	is auxiliary variable, whose initial value is the set $F^{1\lambda}$ defined by (8),
$O(F^p;e_1)$	denotes the operation of choosing the cell with the smallest number from the set specified by the current value of F^p and naming it e_1,
L^q	is an interval complex in $G(e_1|\lambda)$, called a <u>quasi-extremal</u>, which covers the maximum number of events from the current set $F^{1\lambda}$ (in flow-diagram $F^{1\lambda}$ designates a variable whose initial value is the set $F^{1\lambda}$ given by (8) and following values are the sets which remain to be covered after each step of the algorithm, similarly F^* is also a variable whose values are sets; its initial value is given by (7)),

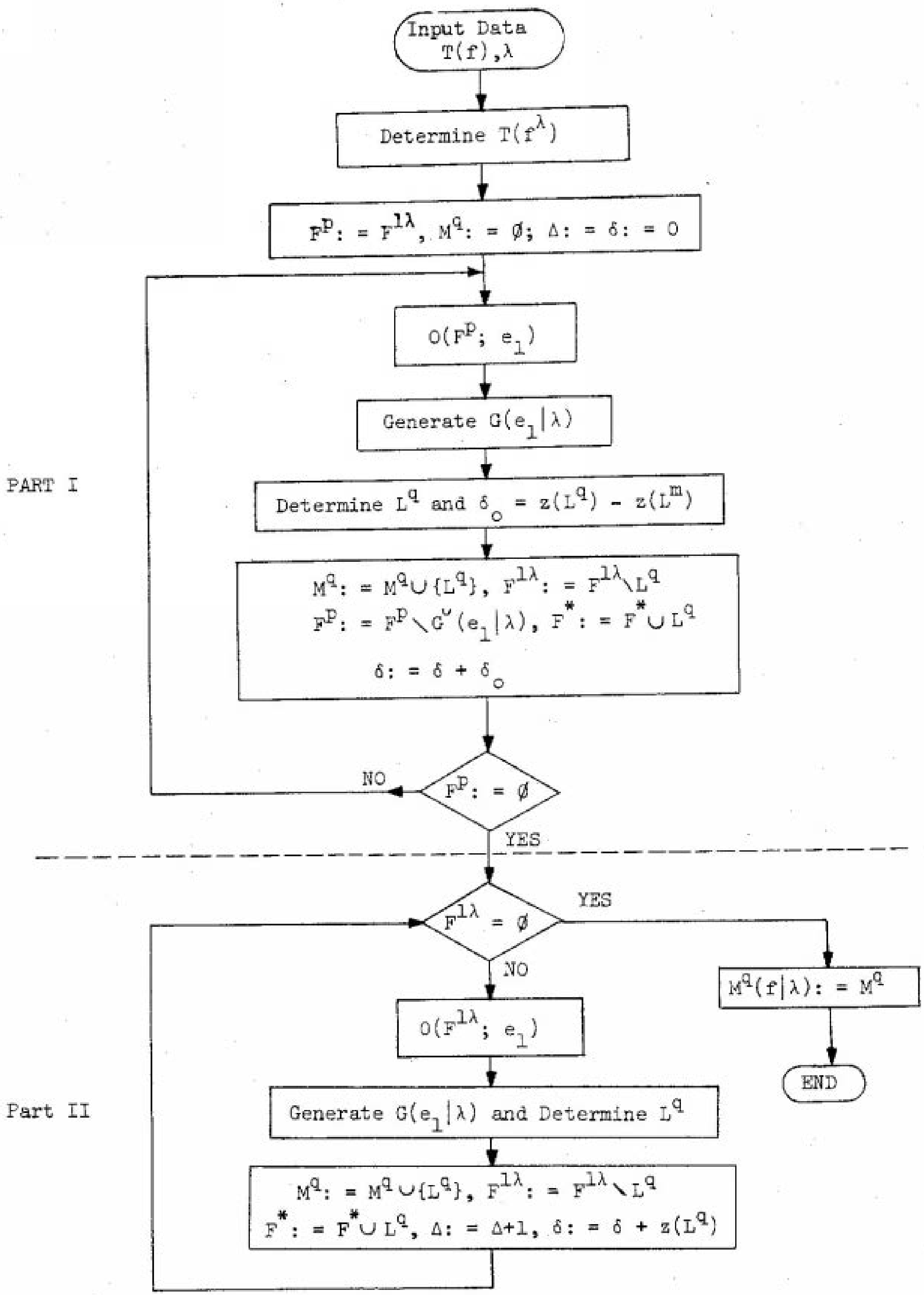


Fig. 17. Algorithm A^Q (a graphical version)

L^m is an interval complex in $G(e_1|\lambda)$ which has the minimum number of literals,

$z(L^q), z(L^m)$ are numbers of literals in L^q and L^m, respectively,

M^q is the variable which stores the set of determined quasi-extremals. Its final value is called a quasi-minimal cover of f^λ and denoted by $M^q(f|\lambda)$.

The algorithm consists of 2 parts.

Part I.

After $T(f^\lambda)$ is determined, stars $G(e|\lambda)$ are successively generated for events which have the smallest number in the current set F^p, and L^q, L^m and $\delta_o = z(L^q) - z(L^m)$ are found for each star. If a star contains more than one L^q, the one with the smaller number of literals is preferable. Since F^p in each step includes only those events from $F^{1\lambda}$ which were not covered by an interval complex from any star generated in previous steps (due to the operation $F^p := F^p \setminus G^\vee(e|\lambda)$), any two stars are disjoint, i.e. they have no common interval complexes.

Part I is terminated when $F^p = \emptyset$ and therefore the family of stars which were generated (we will denote this family by $\mathbf{G}^r$) is a maximal under inclusion family of disjoint stars (i.e. there does not exist an $e \in F^{1\lambda}$ such that $G(e|\lambda) \notin \mathbf{G}^r$ and is disjoint with all its members).

Part II.

This part (executed if $F^{1\lambda} \neq \emptyset$ and $F^p = \emptyset$) determines additional interval complexes which cover the remaining cells of $F^{1\lambda}$. Their number is the value Δ in (36).

To prove (36) let F^r denote the set of cells e_k for which stars were generated in Part I (note that $F^r \subseteq \{e_k | G(e_k) \in \mathbf{G}^r\}$). Since the stars of any two events of F^r have no common interval complexes, every cover of F^r against $F^{0\lambda}$ has to include at least $c(G^r)^*$ elements. But $F^r \subsetneq F^{1\lambda}$, and therefore any cover of $F^{1\lambda}$ against $F^{0\lambda}$, thus also a minimal cover $M(f|\lambda)$ (which here means a cover with the minimum number of elements and the minimum number of literals for that number of elements) cannot have less elements than $c(G^r)$. Thus, the number Δ of interval complexes which have to be used to cover the remaining set $F^{1\lambda}$ in part II of the algorithm can be viewed as an estimate of the maximal possible difference in number of elements between the cover $M^q(f|\lambda)$ and the minimal cover $M(f|\lambda)$. The estimate δ in (37) is obtained by summing the differences δ_o determined in part I and then adding the number of literals in intervals determined in part II. If $\Delta = 0$ and $\delta = 0$, $M^q(f|\lambda) = M(f|\lambda)$.

If after the first execution of the algorithm, Δ and δ are considered to be too large, a better estimate (and/or solution) may be obtained by performing other iterations. A simple and often successful method for realizing further iterations is described in [8].

Example 2

Determine a quasi-minimal interval cover $M^q((f|\lambda)$ for the mapping f^λ defined by $T(f^\lambda)$ in fig. 18.

In the first step, the star $G(e^{18}|\lambda)$ is determined. It consists of interval complexes L_1, L_1', L_2'' (fig. 19). Since L_1 covers the maximum number of cells of $F^{1\lambda}$ (8 cells) it is chosen to be the quasi-extremal. The operation $F^p := F^p \setminus G^\vee(e^{18}|\lambda)$ is realized by marking the cells of $G(e^{18}|\lambda) \cap F^{1\lambda}$ with a $\cdot$. Part I terminates after determining quasi-extremal L_5 in the star $G(e^{185}|\lambda)$.

---

*c(K) denotes the cardinality of K.

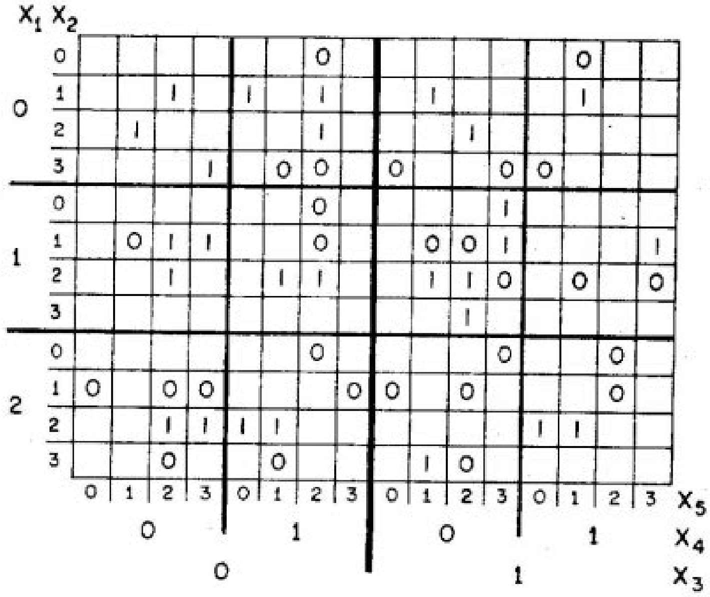
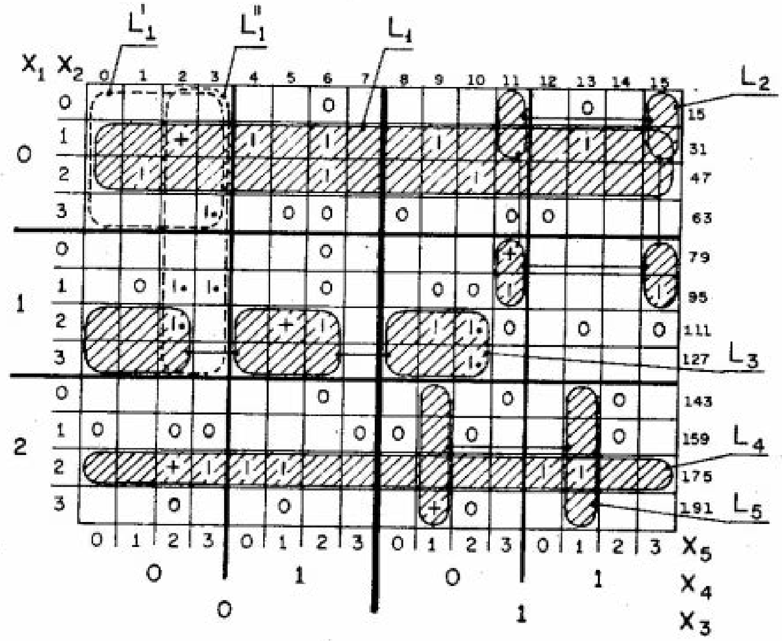


Fig. 18. $T(f^\lambda)$ for example 2



$\Delta = 1$
$\delta = 3$

$$M^q\ (f|\lambda) = \{L_1, L_2, \ldots, L_5, L_1''\}$$

Fig. 19. Interval cover $M^q(f|\lambda)$ of $T(f^\lambda)$ (fig. 18)

In part II, the remaining cells e^{51}, e^{82}, and e^{83} of $F^{1\lambda}$ are covered by L_1''. Thus $\Delta = 1$ and $\delta = z(L_1) = 3$ (fig. 19).

It can easily be verified that if the algorithm is started with the cell e^{33}, the obtained cover will be the same but $\Delta = \delta = 0$. Thus the above determined quasi-minimal cover $M^q(f|\lambda)$ is, in fact, the minimal cover.

Example 3

Assume that objects of class 1 are characterized by set $F^{1\lambda}$ of events $e = (x_1, x_2, x_3, x_4) \in \mathbf{E}(4,4,4,4)$:

$$
\begin{aligned}
F^{1\lambda} = \{&(0,3,3,0),\ (0,3,2,1),\ (1,2,3,0),\ (1,3,3,1),\\
&(0,2,3,1),\ (1,3,2,1),\ (1,2,2,0),\ (1,2,3,1),\\
&(3,0,0,3),\ (2,1,0,3),\ (2,0,1,2),\ (3,1,1,2),\\
&(2,1,1,2),\ (3,1,0,4),\ (2,0,0,2),\ (3,0,1,2)\}
\end{aligned}
$$

and objects of class 0 by set $F^{0\lambda}$:

$$
\begin{aligned}
F^{0\lambda} = \{&(0,0,1,0),\ (1,2,1,2),\ (0,2,0,1),\ (1,1,1,0),\\
&(3,3,3,2),\ (2,3,3,2),\ (2,3,3,0),\ (3,3,2,3),\\
&(3,0,3,0),\ (2,1,3,1),\ (0,3,1,3),\ (0,2,0,3),\\
&(3,3,3,0),\ (2,0,3,2),\ (0,1,1,1),\ (1,1,0,1)\}
\end{aligned}
$$

The above events can be represented graphically as small 'pictures' (fig. 20 a) by interpreting the variables x_i, $i = 1,2,3,4$ as elements of a square:

x_1	x_2
x_3	x_4

and giving them different shades corresponding to the values of x_i ($x_i \in \{0,1,2,3\}$).

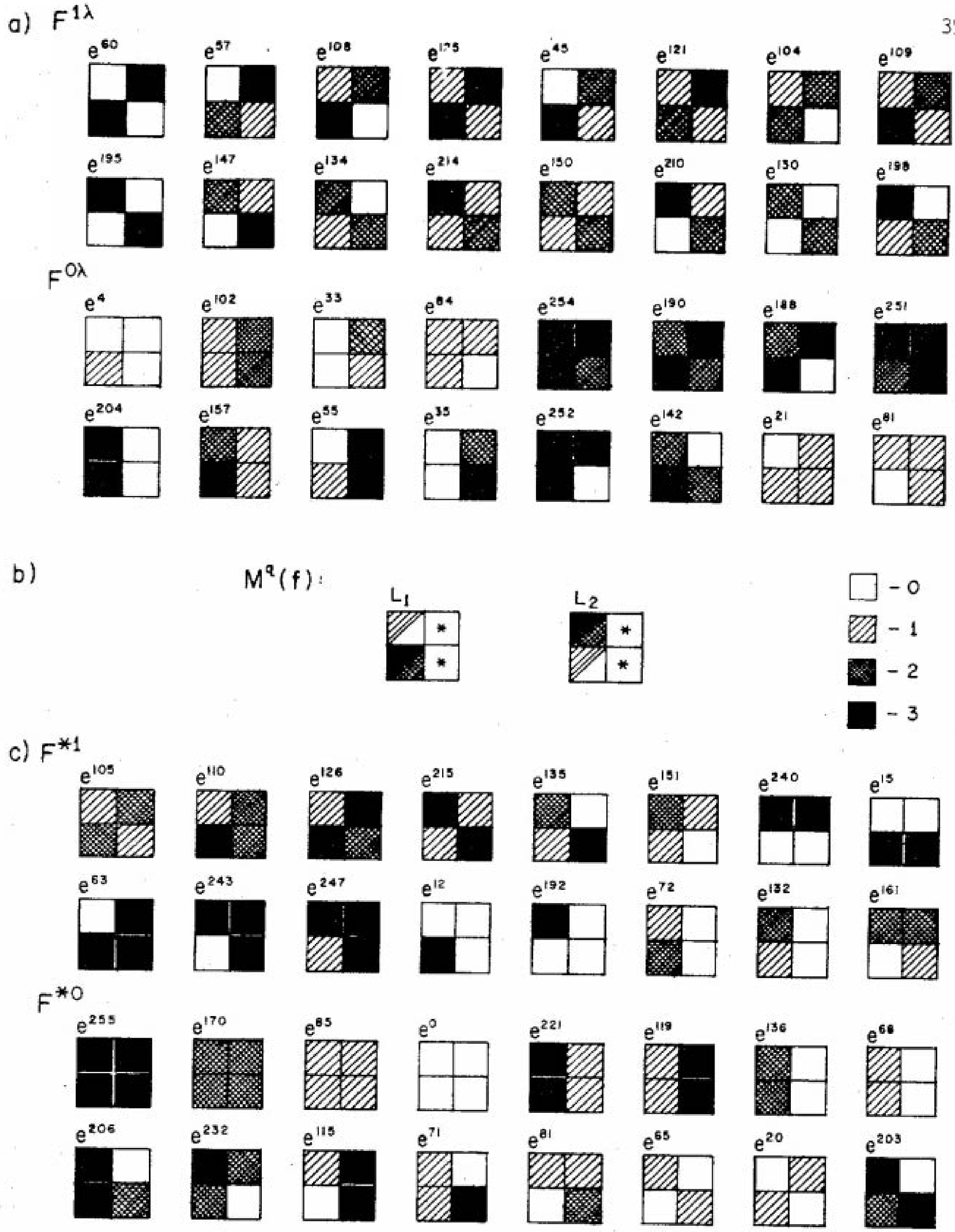
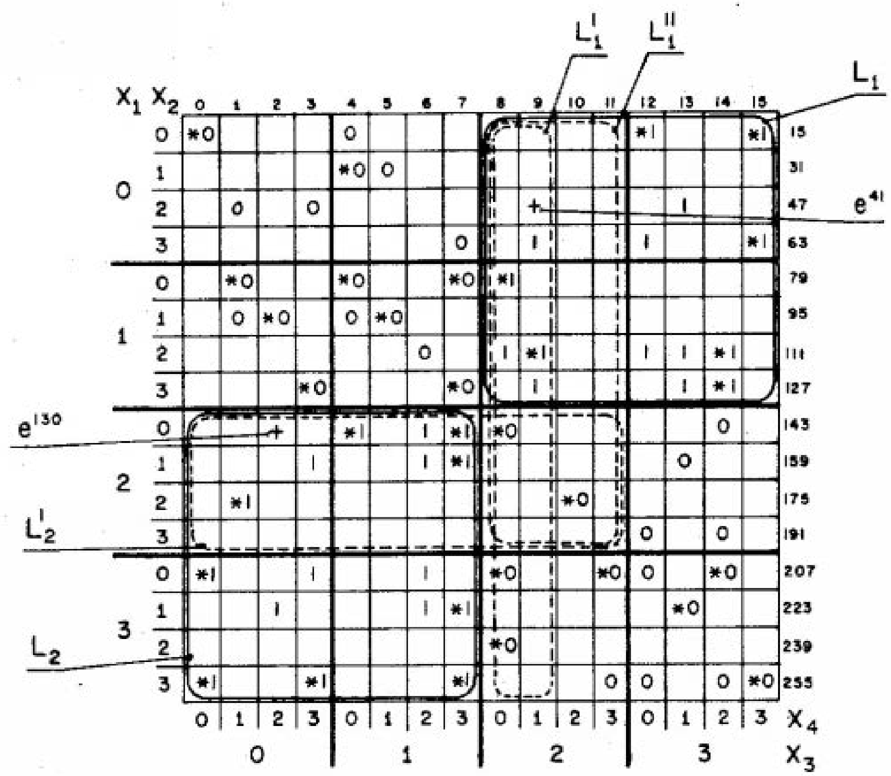


Fig. 20. a. 'Pictures' respresenting events from sets $F^{1\lambda}$ and $F^{0\lambda}$ for example 3.

b. Cover $M^q(f|\lambda)$

c. Samples of 'pictures' representing sets F^{*1} and F^{*0}



$$M^q(f|\lambda) = \{L_1, L_2\}$$

$\Delta = 0$
$\delta = 0$

Fig. 21. Cover $M^q(f|\lambda)$ for example 3

Find the minimal set of features, in the form of intervals in $E(4,4,4,4)$, which permit classification of any event from $F^{1\lambda} \cup F^{0\lambda}$ into the set $F^{1\lambda}$ or $F^{0\lambda}$.

Fig. 21 shows the function image $T(f^{\lambda})$ defined by $F^{1\lambda}$ and $F^{0\lambda}$ and the cover $M^{q}(f|\lambda)$ found by realizing the algorithm A^{Q} . First the star for e^{41} was generated (L_1, L_1' and L_1 in fig. 21 are 3 of 5 elements of $G(e^{41}|\lambda)$), a second star was generated for e^{130} (L_2, L_2' are 2 of 5 elements of $(e^{130}|\lambda)$. Since $\Delta = \delta = 0$, $M^{q}(f|\lambda) = M(f|\lambda)$.

Fig. 20 b shows the intervals L_1 and L_2 of $M^{q}(f|\lambda)$ as features for distinguishing events from $F^{1\lambda}$ and $F^{0\lambda}$. The cover $M^{q}(f|\lambda)$ partitions all events of $E(4,4,4,4)$ into class F^{1} of events which are covered by $L_1 \cup L_2$, and class F^{0} of events which are not covered by $L_1 \cup L_2$. Clearly, $F^{1\lambda} \leqslant F^{1}$ and $F^{0\lambda} \leqslant F^{0\lambda}$. Thus, the events previously unspecified (events of F^{*}) are now included in either the set $F^{*1} = F^{1} \setminus F^{1\lambda}$ or $F^{*0} = F^{*} \setminus F^{*1}$. Assuming that events $e \in F^{*1}$ are classified as events of class 1 and events $e \in F^{*0}$ as events of class 0, the sets F^{1} and F^{0} can be viewed as <u>generalized</u> sample sets $F^{1\lambda}$ and $F^{0\lambda}$. To illustrate the above generalization, samples of events from sets F^{*1} and F^{*0} were shown in fig. 20c. (In fig. 21 these events are marked by *1 and *0, respectively.)

8. CONCLUSION

The GLD model of the space $\mathbb{E}$ is a useful visual aid for geometrically representing mappings f and their interval covers, and for interpreting algorithms for interval covering synthesis. It has proved to be particularly useful for developing and verifying computer implementations of algorithm A^Q [8].

If the number n of dimensions of $\mathbb{E}$ and the values h_i are small, e.g. $n \leq 6$, $h_i \leq 4$, the GLD can easily be used for the direct graphical synthesis of interval covers.

This geometric model can also be applied for representing functions and algorithms of many-valued logic.

9. REFERENCES

1. R. S. Michalski, B. H. McCormick, "Interval Generalization of Switching Theory," Proceedings of the Third Annual Houston Conference on Computer and System Science, Houston (Texas), April 26-27, 1971, (an extended version in Report No. 442, Department of Computer Science, University of Illinois, Urbana, Illinois, May 3, 1971).

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

3. A. Marquand, "On Logical Diagrams for n terms," Edinburgh and Dublin Philosophical Magazine and Journal of Science, Vol. XII, 5th Series, No. 75, pp. 266-270, 1881.

4. E. W. Veitch, "A Chart Method for Simplifying Truth Functions," Proceedings of the Association for Computing Machinery, Pittsburgh, Pa., pp. 127-133, May 2-3, 1952.

5. A. Svoboda, "Grafico - Mechanicke Pomucky Uzirane Pri Analise a Synthese Kontaktowych Obvodu," Stroje Zpracov. Inf., No. 4, 1956 (in Czechoslovakian)

6. R. S. Michalski, "Synthesis of Minimal Forms and Recognition of Symmetry of Switching Functions", Proceedings of the Institute of Automatic Control, Polish Academy of Sciences, No. 91, Warsaw, 1971 (in Polish).

7. R. S. Michalski, "Conversion of Normal Forms of Switching Functions into Exclusive-or-Polynomial Forms," to appear in Archivum Automatyki i Telemechaniki, Polska Akademia Nauk, 1971 (in Polish).

8. R. S. Michalski, V. G. Tareski, "Interval Covers Synthesis: Computer Implementation of and Experiments with Algorithm A^q ", to appear as a report of the Department of Computer Science, University of Illinois.

9. R. S. Michalski, "On the Quasi-Minimal Solution of the General Covering Problem," Proceedings of the 5th Internation Symposium on Information Processing (FCIP 69), Vol. A3, pp. 125-128, Yugoslavia, Bled, October 8-11, 1969.

Form AEC–427
(6/68)
AECM 3201

U.S. ATOMIC ENERGY COMMISSION
UNIVERSITY–TYPE CONTRACTOR'S RECOMMENDATION FOR DISPOSITION OF SCIENTIFIC AND TECHNICAL DOCUMENT

(See Instructions on Reverse Side)

1. AEC REPORT NO. 461 COO-2118-0013	2. TITLE A GEOMETRICAL MODEL FOR THE SYNTHESIS OF INTERVAL COVERS

3. TYPE OF DOCUMENT (Check one):

- [x] a. Scientific and technical report
- [ ] b. Conference paper not to be published in a journal:
  - Title of conference
  - Date of conference
  - Exact location of conference
  - Sponsoring organization
- [ ] c. Other (Specify)

4. RECOMMENDED ANNOUNCEMENT AND DISTRIBUTION (Check one):

- [x] a. AEC's normal announcement and distribution procedures may be followed.
- [ ] b. Make available only within AEC and to AEC contractors and other U.S. Government agencies and their contractors.
- [ ] c. Make no announcement or distrubution.

5. REASON FOR RECOMMENDED RESTRICTIONS:

6. SUBMITTED BY: NAME AND POSITION (Please print or type)

R. S. Michalski
Visiting Asst. Professor of Computer Science

Organization
University of Illinois
Department of Computer Science
Urbana, Illinois

Signature R. S. Michalski	Date June 24, 1971

FOR AEC USE ONLY

7. AEC CONTRACT ADMINISTRATOR'S COMMENTS, IF ANY, ON ABOVE ANNOUNCEMENT AND DISTRIBUTION RECOMMENDATION:

8. PATENT CLEARANCE:

- [ ] a. AEC patent clearance has been granted by responsible AEC patent group.
- [ ] b. Report has been sent to responsible AEC patent group for clearance.
- [ ] c. Patent clearance not required.