PECULIARITY OF GRAPH COLORING IN DECOMPOSITION OF A SYSTEM OF INCOMPLETELY SPECIFIED BOOLEAN FUNCTIONS

Yu. Pottosin* and E. Shestakov

Institute of Engineering Cybernetics of National Academy of Sciences of Belarus, Surganov str., 6, 220012, Minsk, BELARUS, pott@newman.bas-net.by

ABSTRACT

A method for decomposition of a system of incompletely specified Boolean functions is proposed that includes coloring the graph of orthogonality of the decomposition map rows. The method takes into account the dependence of the complexity of Boolean functions resulting from the decomposition on the choice of a variant of the graph coloring.

1. INTRODUCTION

The problem of decomposition of a system of Boolean functions takes an important place in the logic design of discrete devices based on VLSI circuits [1-3]. Here, we consider this problem in the following statement [1-7]. A system of incompletely specified (or partial) Boolean functions is given as a vector function $y=f(x)$ is given where $x=(x_1 x_2 \dots x_n)$, $f(x)=(f_1(x) f_2(x) \dots f_m(x))$, and $y=(y_1 y_2 \dots y_m)$. Subsets $W=\{w_1, w_2, \dots, w_r\}$ and $Z=\{z_1, z_2, \dots, z_s\}$ of the set $X=\{x_1, x_2, \dots, x_n\}$ of arguments are given as well and $X=W \cup Z$. It is necessary to represent the given system in the form of superposition of systems of partial Boolean functions $h(w, g(z))$, with $w=(w_1 w_2 \dots w_r)$, $z=(z_1 z_2 \dots z_s)$.

Let a system of incompletely specified Boolean functions $y=f(x)$ be defined by two ternary matrices whose elements take values from the set $\{0,1,-\}$: $(l \times n)$-matrix U and $(l \times m)$-matrix V. The value "–" is called undefined. In matrix U, ith row, $1 \leq i \leq l$, defines an interval c_i of Boolean space of arguments [1,2]. An interval represented by a ternary vector c is a set of all Boolean vectors that can be obtained by substitution symbols "–" in c by 0s and 1s. The function $y_j = f_j(x)$, $1 \leq j \leq m$, takes the value $\delta \in \{0,1\}$ for a value x^* of the vector variable x if and only if there is an interval c_i, that is defined by i-th row of matrix U and contains x^*, and element v_{ij} of V is equal to δ [1]. The columns of U are marked with $x_1, x_2, \dots, x_n$ and the columns of V are marked with $y_1, y_2, \dots, y_m$. The specification domain of system $y=f(x)$ is defined by the set D_f of values of x where $D_f = c_1 \cup c_2 \cup \dots \cup c_l$.

Let the numbers of components of ternary vectors y^* and y'^* be equal. We say that y^* absorbs y'^* ($y^* \leq y'^*$) if and only if the values of all components of y^* different from "–" are equal to the values of related components of y'^*.

Since vector variables w, z are formed from the components of vector variable x, the components of their values w^*, z^* are the same as the related components of x^*.

Superposition of systems of partial Boolean functions $h(w, g(z))$ realizes a system of partial Boolean functions $y=f(x)$ ($f(x) \leq h(w, g(z))$) if and only if $z^* \in D_g$ and $f(x^*) \leq h(w^*, g(z^*))$ for any $x^* \in D_f$.

In this paper we consider the following problem of decomposition.

Given a system of partial Boolean functions $y=f(x)$ and sets W, Z, where $W \cup Z = X$, find a superposition $h(w, g(z)) = h(w, u)$ of systems of partial Boolean functions realizing f, the sum of the components of w and u (or the number of Boolean arguments of h) being minimum or close to minimum. The systems h and g should be "good" for minimization, i.e. the numbers of different terms in their minimal systems of disjunctive normal forms should be as small as possible.

2. TECHNIQUE OF DECOMPOSITION

Let subsets W, Z of X be given where $W \cup Z = X$. We construct table D whose rows correspond to different values w^* of the vector variable w that are parts of $x^* \in D_f$. The columns of D correspond to different values z^* of the vector variable z that are parts of $x^* \in D_f$. For every $x^* \in D_f$, the value $f(x^*)$ is the entry of the table D at the row corresponding to w^* and at the column corresponding to z^*. If no function of the system $f(x)$ is specified for some value of the vector variable x the corresponding entry of D is "–". The table D may be regarded as the decomposition map introduced in [4,5] and

---

* The author is supported by ISTC project B-104-98.

generalized for a system of partial Boolean functions. The table D defines the system of partial Boolean functions that realizes the system $f(x)$.

For example, let the system of Boolean functions be given by the following matrices, where $x=(x_1, x_2, x_3, x_4, x_5)$, $y=(y_1, y_2)$.

$$U=\begin{array}{|ccccc|} x_1 & x_2 & x_3 & x_4 & x_5 \\ \hline 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & - & - \\ 1 & - & 0 & 0 & 1 \\ - & 1 & - & 1 & 0 \\ 1 & 1 & 1 & 0 & - \\ 0 & - & 0 & - & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & - & 0 & 1 & 1 \end{array}, \qquad V=\begin{array}{|cc|} y_1 & y_2 \\ \hline 0 & 1 \\ 1 & 1 \\ - & 0 \\ - & 1 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ - & 1 \end{array}$$

Assume that $W=\{x_1, x_2, x_3\}$, $Z=\{x_3, x_4, x_5\}$, $w=(x_1x_2x_3)$, $z=(x_3x_4x_5)$. The corresponding table D is shown by tabl. 1.

Table 1. Decomposition map

	$x_3 x_4 x_5$						
$x_1 x_2 x_3$	101	100	111	110	001	011	010
001	01	–	10	–	–	–	–
011	11	11	11	11	–	–	–
100	–	–	–	–	–0	–1	–
110	–	–	–	–	–0	–1	–1
010	–	–	–	–	00	00	–1
111	01	01	–	–1	–	–	–
000	–	–	–	–	00	00	–

The table D, as a decomposition map, defines a given function $f(x)$ in the form of superposition $h(w, g(z))$. The minimization of the number of Boolean arguments of the function h is expressed by the minimization of the number of components of the vector $u=g(z)$. This problem is reduced to coloring a graph [2, 8].

To show this we regard each column of D as a ternary vector divided into sections where every section is a corresponding value of the vector variable y. We define the orthogonality relation between columns of table D as the orthogonality relation between ternary vectors defined in [1]. Two vectors of the same dimension $a=(a_1 a_2 \ldots a_n)$ and $b=(b_1 b_2 \ldots b_n)$ are orthogonal if there is $i\in\{1, 2, \ldots, n\}$ such that both a_i and b_i aren't "–" and $a_i=\overline{b_i}$. We call the values w^* and z^* the codes of corresponding row and column.

In [1] the intersection operation is defined on mutually non-orthogonal ternary vectors. The result of intersection of non-orthogonal ternary vectors is a vector whose *i-th* component has the value $\delta\in\{0,1\}$ if the *i-th* component of any of the vectors taking part in the operation has this value. If both *i-th* components of that vectors are undefined the *i-th* component of the resulting vector is undefined. Substitution of columns of D by their intersection (merging columns) reduces their number. Every column of the new table D is assigned with a set of vectors z^* that were assigned to the columns being merged to the mentioned one. We specify the function $g(z)$ to have the same value on all vectors assigned to one column.

Let us consider a variant of merging columns of the table D resulting in that each column corresponds to some set of values of z on which the function $g(z)$ has the same value. The number of different values of $g(z)$ is equal to the number of columns of D. Evidently, it is minimum if we have managed to get the minimum number of columns in their merging. It can be done having obtained the minimum coloring of the vertices of the graph of column orthogonality. If the minimum coloring has k colors then the number of components of the vector function $g(z)$ is $\lceil \log_2 k \rceil$ where $\lceil a \rceil$ is the closest integer which isn't less then a.

To obtain the superposition $h(w, g(z))$ we assign binary codes of minimal length to columns of table D. Assume that the code u_i^* assigned to *i-th* column of D is a value of a vector variable $u=g(z)$ at all values of z assigned to this column. The complexity of the system of functions $g(z)$ expressed by the number of different terms in their DNFs depends to a considerable extent on the choice of coloring of the graph of column orthogonality of D. We suggest the heuristic method for coloring the graph G of column orthogonality of D that is described below.

We introduce the integer function $w(v_i, v_j)$ on the pairs of vertices of G. This function is of the form

$$w(v_i, v_j)=w'(v_i, v_j)-w''(v_i, v_j) \qquad (1)$$

where $w'(v_i, v_j)$ is the Hamming's distance between the codes of the columns corresponding to vertices v_i and v_j, $w''(v_i, v_j)$ is the number of the components of the same name with the same value 1 in the columns corresponding to vertices v_i and v_j.

Two non-adjacent vertices v_i and v_j of graph G are desirable to be colored in the same color if the value $w(v_i, v_j)$ at these vertices is small. In other words, the less value of $w(v_i, v_j)$, the more desirable for v_i, and v_j to have the same color.

We suggest the following algorithm for coloring the vertices of graph G.

1. Find a maximal complete subgraph in G and color all vertices of it arbitrary in different

colors. The number of colors is equal to the number of vertices in the subgraph.

2. For every uncolored vertex v, define the set B_v of colors that it may have. If there is a vertex u for which B_u is empty, introduce a new color and put it to vertex u, then go to 2. Otherwise, go to 3.
3. Choose a vertex v from uncolored ones for which B_v has the minimal size. If there are several such vertices, then for every such a vertex v and color c from B_v calculate $\sum w(v, u_c)$ where the sum is taken over vertices of color c. Choose the vertex v and the color c for which this sum is minimal, and put color c to v.

Repeat steps 2 and 3 until all the vertices become colored.

To encode the colors we define the function ψ on the set of the pairs of the colors as

$$\psi(i,j)= \sum_{u\in c_i, v\in c_j} w(u,v)/(|c_i|\cdot|c_j|) \qquad (2)$$

where c_i is the set of vertices having color i. The main strategy in encoding colors is: the less value of $\psi(i,j)$ for i and j, the more desirable for i and j to be closer by Hamming's distance.

To keep this strategy one may use the technique similar to that of [9] called "assembling a Boolean hypercube".

Let C be a set of colors of the vertices of the column orthogonality graph G and $\psi(i,j)$ be a real-valued function specified on the set of pairs of colors belonging to C. At the start of the process, the vertices of the hypercube are the vertices of an empty graph (without edges) and related to those colors.

The input data for constructing the k-dimensional hypercube are the values of $\psi(i,j)$ and the number of colors γ of the vertices of G. If γ is not an integer power of two, it should be increased to 2^k where $k=\lceil \log_2\gamma \rceil$, and virtual colors should be introduced respectively. It is regarded that $\psi(i,j)$ is maximum if one of c_i and c_j is such a virtual color.

The process of constructing the Boolean hypercube can be represented as the sequence of k steps. At the sth step, the set of $(s-1)$-dimensional hypercubes are considered. They join into pairs, and s-dimensional hypercube is obtained from each pair by adding edges properly. As far as it is possible, those vertices i and j are chosen for being connected with an edge, which have the smallest value of corresponding $\psi(i,j)$. For every two $(s-1)$-dimensional hypercubes, the sum $\Sigma\psi(i,j)$ is calculated where summing is performed over all pairs i, j of indices of vertices that can be introduced as new edges. The variant with minimal value of this sum is chosen.

After k steps a k-dimensional Boolean hypercube will be obtained. The k-component Boolean vectors are assigned to the vertices of the hypercube where the neighborhood relation between the vectors should be represented by the edges of the hypercube.

At the first step of this process 1-dimensional hypercubes in the form of $\gamma/2$ nonadjacent edges are composed of 0-dimensional hypercubes represented by γ isolated vertices. At the last, kth, step an k-dimensional hypercube is assembled from two $(k-1)$-dimensional ones by adding 2^{k-1} edges.

More details of constructing a hypercube are described in [9].

3. EXAMPLE

Let the system of Boolean functions be given specified by matrices U and V above. The decomposition map D is at tabl. 1. The adjacency matrix of the column orthogonality graph G for table D is as follows.

v_2	v_3	v_4	v_5	v_6	v_7	
0	1	0	0	0	0	v_1
	0	0	0	0	0	v_2
		0	0	0	0	v_3
			0	0	0	v_4
				1	1	v_5
					1	v_6

The values of function w calculated by formula (1) are shown in tabl. 2.

The maximal complete subgraph of G is induced by the set of vertices $\{v_5, v_6, v_7\}$. We put them the following colors: v_5–1, v_6–2, v_7–3. Any other vertex of G may be colored in any color, therefore $B_v=\{1, 2, 3\}$ for $v\in\{v_1, v_2, v_3, v_4\}$.

At the first execution of step 3 of the algorithm above the sum $\sum w(v, u_c)$ has only one term that can be taken from tabl. 2. Here we put color 1 to vertex v_1. The set B_3 for vertex v_3 becomes {2, 3}. tabl. 3 shows the values of $\sum w(v, u_c)$ calculated at the next iteration of step 3 of the algorithm.

After coloring vertex v_2 in color 1 we have $u_1=\{v_1, v_2, v_5\}$, $u_2=\{v_6\}$, $u_3=\{v_7\}$ and two uncolored vertices, v_3 and v_4. The values of $\sum w(v_3, u_c)$ for v_3 don't change. For v_4 we have $\sum w(v_4, u_1)=0$. Finally we have $u_1=\{v_1, v_2, v_4\ v_5\}$. $u_2=\{v_3\ v_6\}$, $u_3=\{v_7\}$.

Table 2. Values of function $w(v_i, v_j)$

v_2	v_3	v_4	v_5	v_6	v_7	
–2	–1	–1	1	2	3	v_1
	0	–2	2	3	2	v_2
		–1	2	1	2	v_3
			3	2	1	v_4
				1	2	v_5
					0	v_6

Table 3. Values of $\sum w(v, u_c)$

	Vertex		
Color	v_2	v_3	v_4
1	0	–	2
2	3	1	2
3	2	2	1

To construct the Boolean graph we introduce a virtual color 4 and calculate the values of function ψ by formula (2): $\psi(1,2)=1$, $\psi(1,3)=2$, $\psi(2,3)=1$. The codes of colors 1 and 2 must be neighbor ones. So must be the codes of 2 and 3. If we put codes 00, 01, and 11 for colors 1, 2, and 3 respectively, we obtain two systems of functions that are specified by the following matrices:

$$\begin{array}{ccc} x_3 & x_4 & x_5 \\ \hline 1 & 0 & - \\ - & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{array}\quad , \quad \begin{array}{cc} g_1 & g_2 \\ \hline 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{array}$$

and

$$\begin{array}{ccccc} x_1 & x_2 & x_3 & u_1 & u_2 \\ \hline 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & - & 0 & 0 & 0 \\ 1 & - & 0 & 0 & 1 \\ - & 1 & 0 & 1 & 1 \\ 0 & - & 0 & 0 & - \\ 1 & 1 & 1 & 0 & - \end{array}\quad , \quad \begin{array}{cc} y_1 & y_2 \\ \hline 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ - & 0 \\ - & 1 \\ - & 1 \\ 0 & 0 \\ 0 & 1 \end{array}\quad .$$

In this example the number of arguments of h_1 and h_2 isn't less than that of f_1 and f_2. This example is given only to illustrate the technique for decomposition with overlapping W and Z.

After minimization of these systems incompletely specified functions we obtain:

$$u_1=g_1(x_3, x_4, x_5)=\bar{x}_3\bar{x}_5;$$
$$u_2=g_2(x_3, x_4, x_5)=\bar{x}_3\bar{x}_5 \vee x_4x_5;$$
$$y_1=h_1(x_1, x_2, x_3, u_1, u_2)=x_2x_3u_2 \vee \bar{x}_1x_2x_3;$$
$$y_2=h_2(x_1, x_2, x_3, u_1, u_2)=x_3\bar{u}_2 \vee x_1u_2 \vee u_1;$$

4. CONCLUSION

The problem of decomposition of a system of incompletely specified Boolean functions doesn't lose its importance. In this paper we suggest a method for decomposition using special graph coloring and color encoding oriented to obtaining comparatively simple systems of functions.

REFERENCES

[1].Zakrevskij A.D. Logical Design of Cascade Circuits. - Moscow: *Nauka*, 1981 (in Russian).

[2].Bibilo P.N., Enin S.V. Synthesis of Combinational Circuits by the Functional Decomposition Method. - Minsk: *Nauka i Tekhnika*, 1987 (in Russian).

[3].Luba T., Selvaraj H. A general approach to Boolean function decomposition and its application in FPGA-Based Synthesis. *VLSI Design*, **3**, N 3–4, 1995, pp. 289–300.

[4].Ashenhurst R. L. The decomposition of switching functions. Proc. of *International Symp. Theory of Switching Functions*, 1959.

[5].Curtis H. A. A New Approach to the Design of Switching Circuits. - Princeton, NJ: D. Van Nostrand Co. Inc., 1962.

[6].Brzozowski J.A., Luba T. Decomposition of Boolean functions specified by cubes (ftp://cs-archive.uwaterloo.ca/cs-archive/CS-97-01/CS-97-01.ps.Z), 1997, pp. 36.

[7]. Pottosin Yu.V., Shestakov E.A. On the decomposition of a system of incompletely specified Boolean functions. Proc. of the *Third International Conf. "New Information Technologies in Education"* (NITE'98), **3**, Minsk, State Economic University, 1998, pp. 65-71.

[8].Zakrevskij A.D. Graph coloring by Boolean function decomposition. *Logical Design*, issue 5, Minsk, Institute of Engineering Cybernetics of NAS of Belarus, 2000, pp.83-97 (in Russian).

[9].Pottosin Yu.V. "Assembling" a Boolean hypercube: an approach to state assignment of finite state machines. Proc. of the *Second International Conf. on Computer-Aided Design of Discrete Devices* (CAD DD'97), **1**, Minsk, Institute of Engineering Cybernetics of NAS of Belarus, 1997, pp. 54-59.