Proceedings of the 3rd International Conference on Applications of Computer Systems – ACS'96, Poland, 1996, pp.65-74

Exact Irredundant Searching for a Minimal Reed-Muller Expansion for an Incompletely Specified MVL Function

S. Jaroszewicz, V. Szmerko, S. Januszkiewicz

Institute of Computer Science and Information Systems, Technical University of Szczecin
ul. Żołnierska 49, 71-210 Szczecin, Poland, fax: (4891) 764-54
e-mail: shmerko@beta.ii.tuniv.szczecin.pl

Abstract - A power method for finding an optimal polynomial representation of a weakly specified many-inputs one-output Multiple-Valued Logic (MVL) function is suggested. The algorithm proposed uses a matrix method instead of over-defining the don't cares.

Index Terms - Multiple-valued logic, incompletely specified logic function, Reed-Muller fixed and mixed polarity expansions

1. Introduction

MVL functions as models of multi-level signals are used together with Boolean representations in modern digital systems [Sasao93]. Canonical forms, especially Reed-Muller ones are of special interest, for instance, when designing logic arrays from MIN, MOD-k-SUM gates.

Incompletely specified functions are more difficult to handle in logic design than completely specified functions, because the don't-care values introduce an additional degree of freedom to the process of logic minimization [Brayton84, McKenzie93, Miller94, Sasao93, Chen95]. The problem considered in this paper is: how to find an optimal Reed-Muller canonical form of a given incompletely specified MVL function? First we must adopt a criterion of the quality of representation. The criterion we use is the number of nonzero products of the polynomial and the number of literals, which should be minimal. The previous solutions of this problem simply over-defined the don't care term of functions, thus being ineffective for weakly specified functions [Green87, Varma91]. Some new approaches to reduce the computational complexity were proposed in [Csansky93, Falkowski92, Zilic95]. An approach to heuristic solution based on genetic algorithms was proposed for Boolean functions [Miller94] and for MVL functions [Kalg96].

A new approach to such problems for many-inputs one-output Boolean functions has been proposed by A.Zakrevskij [ZakTor95]. It uses the so called staircase search. The main new idea was not to consider the unspecified values of the function.

In this paper we generalize the algorithm proposed by Zakrevskij onto Multiple-Valued Logic (MVL) functions. The algorithm is irredundant because we also do not care about don't cares, and the search area includes the specified values only.

2. Generalized Reed-Muller Canonical Forms for Multiple-Valued Functions

Let us have a k-valued (k is a prime number) logic function. The following equation gives the generalized Reed-Muller (GRM) canonical form of fixed polarity c for a k-valued logic function (k is a prime number):

$$F_c(X) = \sum_{j=0}^{k^n-1} f_c^{(j)}(x_1+c_1)^{j_1}\dots(x_n+c_n)^{j_n} \quad \text{over GF}(k),$$

where: $c=0,1,\dots,k^n-1$; $c_1\, c_2 \dots c_n$ and $j_1 j_2 \dots j_n$ are correspondingly k-valued representations of integers c and j: $c_i, j_i = 0,1,\dots,k-1$; $x_i + c_i$ denotes a literal, a so called „c_i-order cyclic inversion of x_i"; by that $(x_i+c_i)^0=1$; $f_c^{(j)}$ are the polynomial's coefficients [Green 89, Yan95].

There are k^n GRM canonical forms of fixed polarity for any k-valued function. There is also a more general class of GRM forms with mixed polarity where each variable is allowed to be with a cyclic inversion or without inversion.

<u>**Example:**</u> All 3^2 GRM canonical forms of fixed polarities for the 3-valued function of two variables given by its truth vector X=[2 0 1 0 0 0 1 0 2] are:

$F_0(x) = 2x_1 + x_1^2 + x_1x_2 + 2x_1^2x_2$ over GF(3)

$F_1(x) = 2 + 2x_1 + 2x_1^2 + 2\hat{x}_2 + 2x_1\hat{x}_2 + 2x_1^2\hat{x}_2$ over GF(3)

$F_2(x) = 2x_1\hat{\hat{x}}_2 + 2x_1^2\hat{\hat{x}}_2 + 2\hat{\hat{x}}_2$ over GF(3)

$F_3(x) = 1 + 2\hat{x}_1 + \hat{x}_1^2 + 2x_2 + \hat{x}_1x_2 + 2\hat{x}_1^2x_2$ over GF(3)

$F_4(x) = 2 + \hat{x}_1 + 2\hat{x}_1^2 + 2\hat{x}_2 + \hat{x}_1\hat{x}_2 + 2\hat{x}_1^2\hat{x}_2$ over GF(3)

$F_5(x) = 2\hat{x}_1\hat{\hat{x}}_2 + \hat{x}_1^2\hat{\hat{x}}_2 + 2\hat{\hat{x}}_2$ over GF(3)

$F_6(x) = 2\hat{\hat{x}}_1^2x_2 + \hat{\hat{x}}_1$ over GF(3)

$F_7(x) = 2\hat{\hat{x}}_1^2x_2 + 2\hat{\hat{x}}_1$ over GF(3)

$F_8(x) = 2\hat{\hat{x}}_1^2\hat{\hat{x}}_2$ over GF(3)

It can be seen that the number of terms in the representation heavily depends on the polarity.

A minimal canonical form is the form so that, at first, the number of zero coefficients $f_c^{(j)}$ is maximal, and the second, the common number of literals in products is minimal.

3. Exact RM minimization algorithm for MVL functions

A quite different minimization is required for incompletely specified MVL functions. Suppose we have a k-valued (k is a prime number) logic function specified on some definition area. This area is represented by a matrix $\boldsymbol{B}$ with n columns corresponding to k-valued variables $x_1,\dots,x_n$, and with m rows corresponding to different inputs from the area. The values of the function are given by its incomplete truth column vector X.

Example 2. Let us a k=3-valued function f is given on m=5 values of its 3 variables

$$B=\begin{array}{c}\begin{array}{ccc} x_1 & x_2 & x_3 \end{array}\\ \begin{bmatrix}1 & 0 & 2\\0 & 0 & 0\\2 & 1 & 2\\2 & 2 & 1\\0 & 0 & 1\\1 & 2 & 0\end{bmatrix}\end{array}, \qquad X=\begin{bmatrix}1\\1\\2\\0\\2\\1\end{bmatrix}$$

Our aim is to find the optimal polynomial for the given incompletely specified MVL function. We propose the algorithm based on so-called staircase search proposed by A.Zakrevsky for Boolean functions [ZakrTor95].

If we have the complete truth vector X^*, we would realise so-called orthogonal transform

$$F^*=K_kn\, X^* \text{ over GF}(k)$$

to find coefficient column vector F^* ; spectral transform matrix K_kn is defined as in [Yan95].

Having the incompletely specified function given by its truth vector X, we meet with the problem of solving the following linear matrix equation with k-valued variables

$$X = E\ F \text{ over GF}(k)\,,$$

where the matrix E is the Extended Definition Matrix; F is the incomplete Reed-Muller coefficient vector. Matrix E is formed from the vector-columns, representing all possible results (without repetitions) of componentwise and multiplace conjunctions of columns from the matrix B and also their 1,2,..,(k-1)-powers and cyclic inverted variables in accordance with polarity, i.e. all different projections of potentially useful terms for Reed-Muller polynomial implementation of a given polarity. In other words, the columns of matrix E represent all possible terms $(x_1+c_1)^{j1}...(x_n+c_n)^{jn}$ for the Reed-Muller polynomial forms. Note that E always contains the first column consisting of 1 generated by the conjunction of zero number of columns.

Example 3. For the matrix B given in Example 1 and c=0 the matrix E is the following

$$B=\begin{array}{c}\begin{array}{ccc} x_1 & x_2 & x_3 \end{array}\\ \begin{bmatrix}0 & 0 & 1\\0 & 2 & 0\\0 & 2 & 2\\1 & 0 & 2\\2 & 0 & 0\end{bmatrix}\end{array}, \qquad E=\begin{array}{c}\begin{array}{cccccccc} 1 & x_1 & x_2 & x_3 & x_2x_3 & x_3^2 & x_1x_3 & x_1^2 \end{array}\\ \begin{bmatrix}1 & 0 & 0 & 1 & 0 & 1 & 0 & 0\\1 & 0 & 2 & 0 & 0 & 0 & 0 & 0\\1 & 0 & 2 & 2 & 1 & 1 & 0 & 0\\1 & 1 & 0 & 2 & 0 & 1 & 2 & 1\\1 & 2 & 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}\end{array}$$

(we write one of the repeated terms only).

To find a minimum polynomial implementation of an incompletely specified MVL function, it is necessary to find a minimum subset of columns from the Extended Definition matrix, such that componentwise modulo k sum (or usual sum when we concern arithmetical form) of these columns (with 1,2,...,k-1 coefficients) have to be equal to the specified values

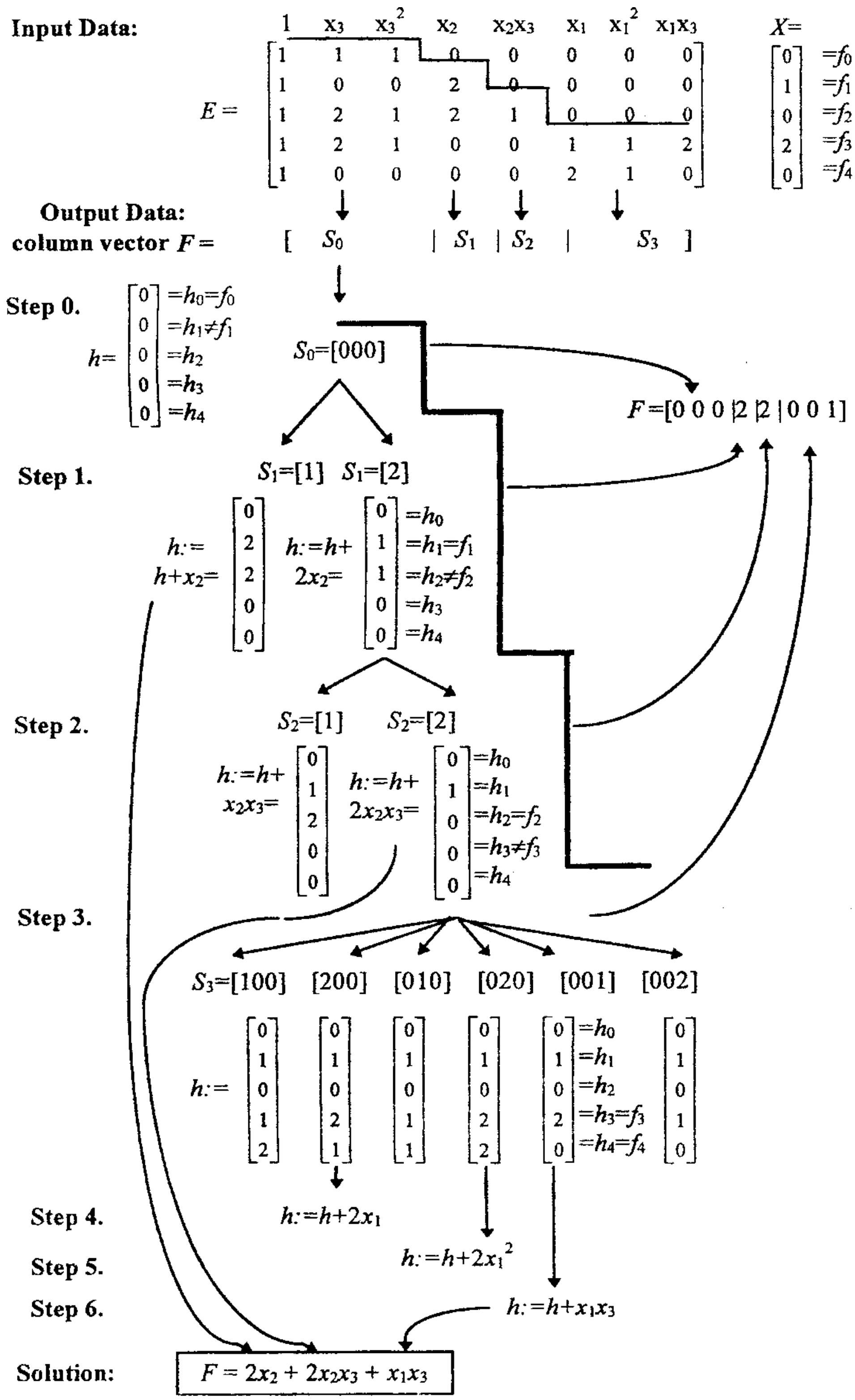

Fig.1. Illustration for Example 5.

of the initial MVL function, i.e. vector X. The chosen set of columns from E is indicated by vector F - its elements shows what columns from E were chosen and with what coefficients. the time needed for the appropriate search depends powerfully upon the number of columns in E, since problem is NP-hard, however this number can be considerably less than the number of all possible products, especially for weakly defined functions.

To implement the algorithm to derive the minimal polynomial form for the MVL function given in such way, we have to use the staircase form of matrix E, i.e. the matrix where the columns are ordered in the following way: at the beginning the columns having the upper 1s (2s,3s,...,(k-1)s) in the first row, then the columns having the upper 0s and the next 1s (2s,3s,...,(k-1)s) in the second row, and so on.
In Boolean case we are guaranteed to obtain a staircase matrix if the rows of the input area matrix B are sorted so that they satisfy the following condition: do not cover any rows by preceding ones. The MVL case is more difficult. In some cases the rows cannot be sorted so, that they satisfy the above condition. So we will obtain stairs of zero width. However, we can still obtain the missing stairs. We can do this by addition of the rows of the matrix E. The rows containing zero-width stairs should be added to each other and the first row above them. Adding the rows, we should also add the corresponding values of the vector X.

Example:
Having the E matrix given below the fifth stair is of zero width:

$$E = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 2 & 0 & 1 \end{bmatrix}, \qquad X = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 0 \\ 1 \end{bmatrix}$$

We can obtain a staircase matrix by adding the fifth row multiplied by two to the fourth row:

$$E = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 2 & 2 & 0 \\ 1 & 0 & 0 & 0 & 0 & 2 & 0 & 1 \end{bmatrix}, \qquad X = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

Table 1 shows the relationship between the values of n and m and the number L of columns in matrix E for $k=3$ and 5. Note that the columns of E correspond to the classes of conjunctive terms; terms of the same class have equal projections onto the definition area, and this projection is presented by the corresponding column. That is why the number L of columns in E can, for small m, be far less than the number k^n of all conjunctive terms, and this fact facilitates the finding of a solution of the regarded problem.

Table 1, 2. The number L of columns in the matrix E (the conjunctive closure of the definition area matrix B), for 3 and 5 valued logic.

Number of spec. terms	Number of variables n, *3-valued logic*						
	6	7	8	9	10	11	12
6	55	79	71	99	104	138	137
7	57	101	100	137	225	237	233
8	82	102	155	189	260	335	377
9	98	161	227	273	303	466	506
10	113	154	234	300	517	506	741
11	125	207	298	438	578	672	1078
12	155	224	384	554	571	941	1226

Number of spec. terms	Number of variables n, *5-valued logic*			
	4	5	6	7
4	33	55	66	143
5	88	115	186	
6	104	171	286	

Example 4. For the matrix E given in Example 3 the staircase form is the following:

$$\begin{array}{cc} & \begin{array}{cccccccc} 1 & x_1 & x_3^2 & x_2 & x_2x_3 & x_1 & x_1x_3 & x_1^2 \end{array} \\ E= & \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 2 & 2 & 0 \\ 1 & 0 & 0 & 0 & 0 & 2 & 0 & 1 \end{bmatrix} \end{array}$$

The staircase algorithm consists in the reduced tree searching, using columns of the matrix E. Corresponding fragments (stairs) of the matrix are fixed for every tree level and that facilities looking for an optimal solution of the problem. It should be noted that solution can include $1,2,\ldots,m-1$ or none columns from each stair.

Suppose a set H_t is the current set (at the step t) of some columns from E, and its cost (the number of these columns) denotes by $cost(H_t)$. We carefully derive the stairs from S_0 to the last one. The best solution obtained at the moment is H. The vector S_t indicates which column from the stair can be chosen. So, the exact algorithm includes the following steps:

The algorithm for finding a minimal Reed-Muller representation of an incompletely specified MVL-function

Input Data: definition area matrix **B**, function values vector **X**

Output Data: coefficient vector **F**

step 1. Suppose the current value of F (the current modulo k sum of the columns F_j from E) is $h=[h_0\ h_1\ \ldots\ h_m]=[0\ldots0]$.

step 2. If the t-th element of the vector h is not equal to the t-th element of truth vector $X=[f_0\ f_1\ldots f_m]$: $h_t \neq f_t$, and $cost(H_t)<cost(H)-1$, then we include one (2,..,(k-1)) column F_j or none from the stair into our solution. $h:=h+F_j$, $H:=\{F_j\}$.

If $h_t=f_t$ and $cost(H_t) < cost\ t(H)$, then $H:=H_t$.

If $cost(H_t)\geq cost(H)-1$, then we change the value of the stair S_t so that the $cost(H)$ does not decrease.

step 3. The searching terminates when $H_t=H$ with minimum cost, i.e. minimal polynomial F realizing the function f defined on B, is obtained.

Example 5. Let us execute the staircase searching algorithm step by step for the function considered in the example above (Fig. 1).

Step 0. h=[00000]. Since f_0=h_0=0, S_0=[F_0 F_1 F_2]=[000].

Step 1. Since f_1=1 ≠ h_1=0, then S_1=[F_3]=[1] or [2] (Fig.3). We choose the column x_2 with co-factor (coefficient) 2. Then h:=h+2x_2. H={2x_2}.

Step 2. Since f_2=0 ≠ h_2=1, then S_2=[F_4]=[1] or [2] (Fig.3). We choose the column x_2x_3 with coefficient 2. Then h:=h+2x_2x_3=2x_2+2x_2x_3. H={2x_2,+ 2x_2x_3}.

Step 3. Since f_3=2 ≠ h_3=0, then S_3=[F_5 F_6 F_7]=[100], [200], [010], [020], [001] or [002] (Fig.3). We choose S_3=[200], i.e. column x_1 with co-factor 2. Then h=[01021]. But h≠f since their last elements are differed. Then we continue the search for this stair.

Step 4. We choose column x_1^2 with co-factor 2, S_3=[020]. Then h=[01022] . But h≠f. Then we continue the search for this stair.

Step 5. Choosing column x_1x_3 with co-factor 2, we S_3=[001] and h=[01020]. Then h:= h+ x_1 x_3=f.

So, we obtain the following:

$$F = [\mathbf{00022001}],$$

$$F = 2x_2 + 2x_2x_3 + x_1x_3 \text{ (over GF(3))}.$$

The obtained vector F corresponds to the following complete coefficient vector

F^* = **[000220000010000000000000000]**.

It is clear, that the inversion transform of the coefficient vector gives a restored completely defined MVL function f^* .

4. Experimental Results

The exact algorithm was tested in the range of k=3,5; n ≥ 5, m. ≤ 40 and for polynomials of both the fixed and mixed polarity. The experiments have been conducted on a PC486/100Mhz and OS LINUX. The value of each runtime shown below is an average time of calculation over 10 randomly generated samples of the specified size. The estimation of the runtime, and the number of terms in the constructed optimised system of Reed-Muller expansions representing the initial MVL functions can be evaluated using Fig. 2 - 5.

The estimations and runtime (Fig.2) show that the method is highly efficient for weakly specified functions (for small m) but as the problem is NP-hard, the time of computations depends exponentially on the number of specified terms.

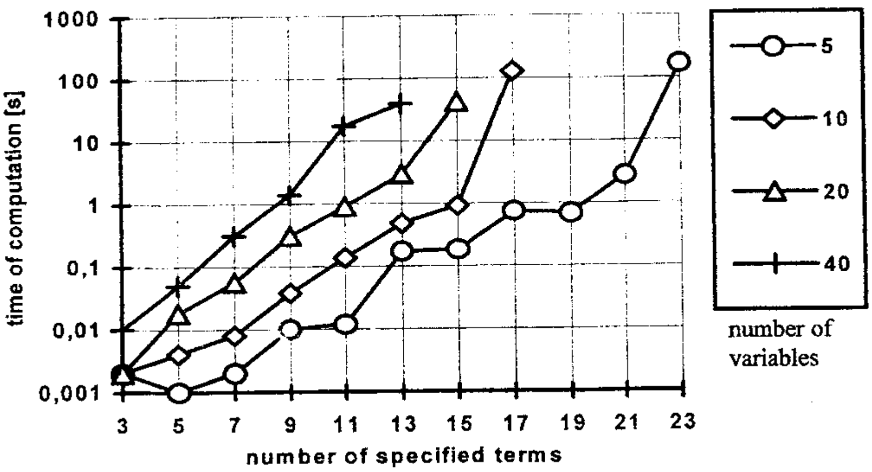

Fig.2. The time of computation depending on the number of specified terms for 5, 10, 20, 40 variables; 3-valued logic. The matrix **B** contains 50% of nonzero elements. The time is shown in logarithmic scale.

The rate of the minimization characterised by the number *nx* of the non-zero terms in the constructed Reed-Muller polynomial represented the initial 3-valued logic function (Fig.3) shows that these numbers are comparable, but the rate is better for smaller number of specified values of the functions.

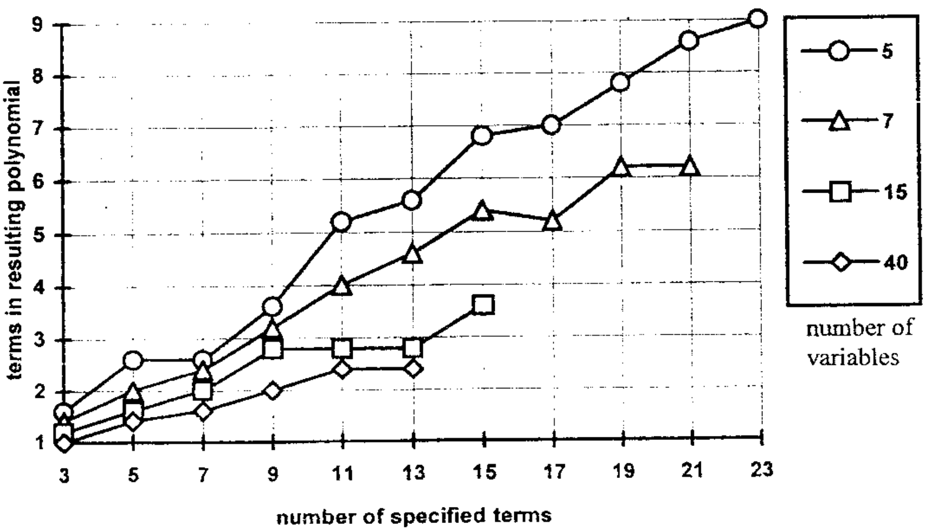

Fig.3. The number of terms in the resulting polynomial depending on the number of specified terms for 5,7,15,40 variables. The matrix **B** contains 50% of nonzero elements; 3-valued logic

It can be seen (fig. 4 and 5), that the time of computations and the number of terms in the resulting polynomial depend heavily on the number of nonzero elements in the definition area matrix **B**.

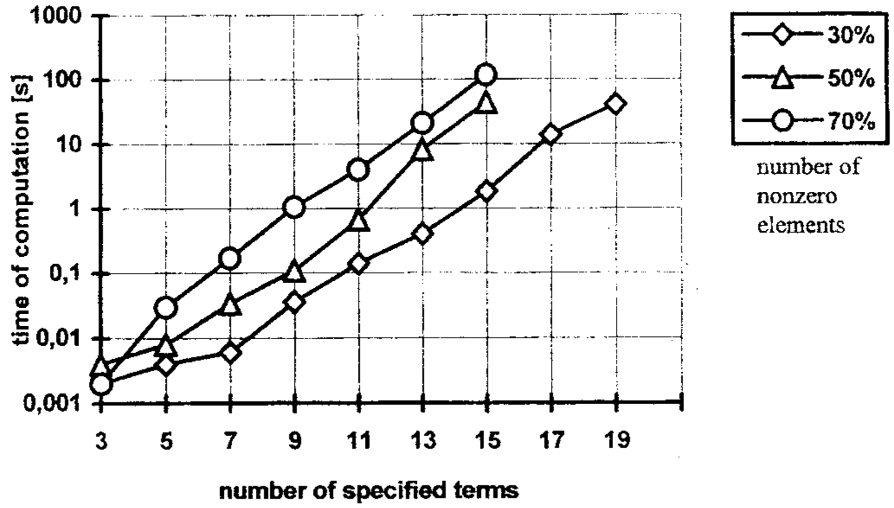

Fig.4. The time of computation depending on the number of nonzero elements in the matrix **B** and the number of specified terms for 15 variables, 3-valued logic. The time is shown in logarithmic scale.

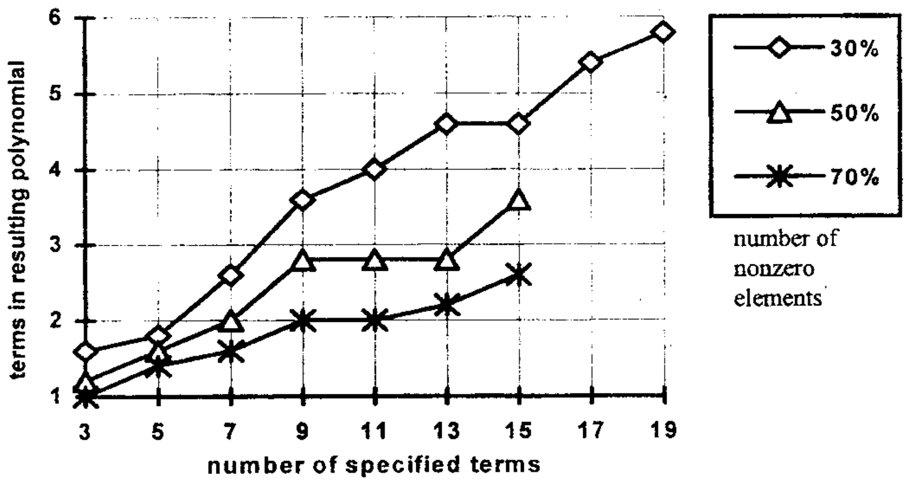

Fig.5. The number of terms in the resulting polynomial depending on the number of nonzero elements in the matrix **B** and the number of specified terms for 15 variables, 3-valued logic

5. Conclusion

The proposed approach to solve the problem of minimization of an incompletely specified MVL function is based on generalized exact staircase algorithms. It should be noted that the algorithm is much more powerful for strongly unspecified functions (the less the number of defined values, the less the time of computations and non-zero products in the solution polynomial). Also the algorithm result characteristics are better when the number of zero elements of the definition area matrix is greater. It should be noted, that this exact algorithm obviously has restrictions on the numbers of variables n and specified values m. That's why some approaches to reduce the algorithm are developed.

So, the range of applying the algorithm (on the view of the number of variables) can be extended by using element of heuristic searching [ZakTor95]. Such an approach is developed in [Hol96]. The exact algorithm for multiple-output MVL functions is proposed in [Zakr96].

References

1. Brayton R.K., Hachtel G.D., McMullen C.T., Sangiovanni-Vincentelly, Logic Minimization Algorithms for VLSI Synthesis, Boston, MA, Kluwer, 1984
2. Chen L-C., and H.T.Twu, Synthesis of Multilevel NAND Gate Circuits for Incompletely Specified Multi-Output Boolean Functions and CAD Using Permissible Cubes and PCRM Graphs. Int.J. Electronics, vol.78,no.2, 1995, pp.303-316
3. Csansky, L., Perkowski, M., Schafer, I., Canonical restricted Mixed-polarity Exclusive-OR Sums of Products and the efficient Algorithm to their Minimisation, IEE Proceedings-E, Vol.140, No.1, 1993, pp.69-77
4. Falkowski B.J., I.Schafer and M.A.Perkowski, Generation of Adding and Arithmetic Multi-Polarity Transforms for Incompletely Specified Boolean Functions, Int.J. Electronics, vol.73,no.2, 1992, pp.321-331
5. Green D.H., Reed-Muller Expansion of Incompletely Specified Functions. IEE Proceedings, vol.134, Pt.E, no.5, 1987, pp.228-236
6. Green D.H., Ternary Reed-Muller switching functions with fixed and mixed polarities. Int. Journal od Electronics, Vol.67, no.5, 1989, pp.761-775
7. Hołowiński G., Yanushkevich S., Fast Heuristic Minimization of MVL functions in Generalized Reed-Muller Domain (this issue).
8. Kalganova T., Kochergov E., Zaitseva E., Yanushkevich S., A Genetic Approach to Optimise Polynomial Forms of Incompletely Specified MVL Functions, Proc.of the Int. Workshop on Evolutionary Programming AISB'96, UK, 1996 (accepted)
9. McKenzie L. Almani A. , Miller I.F., Thomson P., Optimization of Reed-Muller Logic Functions .Int. Journal of Electronics, 1993, Vol. 75, No.3, pp 451-466
10. Miller J.F., Luchian H., Bradbeer P.G.,and Barclay P.J., Using a Genetic Algorithm for Optimizing Fixed Polarity Reed-Muller Expansions of Boolean Functions. Int. J. Electronics, Vol.76, No.4, 1994, p.601-609
11. Miller, J.F., Thomson, P., Highly Efficient Exchaustive Search Algorithm to Optimizing Reed-Muller Expansions of Boolean functions, Int.Journal of Electronics, 1994, Vol.76, No.1, pp.37-56
12. Perkowski, M., Chrzanowska-Jeske, M., An Exact Algorithm to Minimize Mixed-radix Exclusive Sums of Produsts for Incompletely Specified Boolean functions, Proc. on ISCAS Int. Symposium on Circuits and Systems, New Orlean, USA, 1990, pp.1652-1655
13. Riege M.W. and Ph.W.Besslich, Low-Complexity Synthesis of Incompletely Specified Multiple-Output Mod2 Sums. IEE Proceedings-E, vol139, no.4, 1992, pp.355-362
14. Sasao T., AND_EXOR Expressions and their Optimization, In: Logic Synthesis and Optimization, Ed. by T.Sasao, Kluwer Academic Publishers, 1993
15. Varma D.,Trachtenberg E., Computation of Reed-Muller Expansions of Incompletely Specified Boolean Functions from Reduced Representations, IEE Proc.-E,Vol.138, No.2,1991,pp.85-92
16. Yanushkevich S. Spectral and Differential Methods to Synthesize Polynomial Forms of MVL functions on Systolic Arrays. Proc. of the Fifth Int. Workshop on Spectral Techniques, Ed.: C.Moraga, Z.Gongli, etc., Beijing, China, 1994, p.78-93
17. Zakrevskij A., Minimum Polynomial Implementation of Systems of Incompletely Specified Boolean Functions, Proc. of IFIP WG 10.5 Workshop on Application of the Reed-Muller Canonical Expansion in Circuit Design, Japan, 1995, pp.250-256
18. Zakrevskij A.,Yanushkevich S., Jaroszewicz S., Minimization of Reed-Muller Expansions for System of Incompletely Specified MVL Functions, Proc. of the Third International Symposium on MMAR, Miedzyzdroje 1996, pp. 1085-1090
19. Zakrevskij A., Toropov N., Optimizing Polynomial Implementation of Incompletely Specified Boolean Functions, Proc. of the Int. Conference on CAD DD, Minsk, 1995, pp. 93-98
20. Zilic Z. and Z.Vranesic. A Multiple Valued Reed-Muller Transform for Incompletely Specified Functions. IEEE Trans. Computers, vol.C-44, no.8, 1995, pp.1012-1020
21. Zilic Z. and Z.Vranesic. New Interpolation Algorithms for Multiple-Valued Reed-Muller Forms. Proc. of ISMVL96, pp. 16-23