FACTA UNIVERSITATIS (NIŠ)
SERIES: ELECTRONICS AND ENERGETICS VOL.13, NO.1, APRIL 2000, 11–25

AND/EXOR MINIMIZATION OF SWITCHING FUNCTIONS BASED ON INFORMATION THEORETIC APPROACH

Vladimir Shmerko, Denis Popel, Radomir Stanković, Vasily Cheushev and Svetlana Yanushkevich

Abstract. In modern circuit design, concept of Shannon decomposition of switching functions is widely used. On the other hand, in Information Theory, Shannon entropy as a quantitative measure of information is a key notion. In this paper, we relate these two concepts, belonging to different areas, into an approach to the minimization of switching functions in Exclusive-or Sum-Of-Products (AND/EXOR) form. The Shannon decomposition and Davio decomposition for AND/EXOR expressions are investigated and interpreted in Information Theory terms. Thank to that, we have proposed an entropy-based strategy for minimization of switching function. We have provided a comparison and an experimental verification of this strategy with some known heuristic minimization strategies using benchmarks. In some cases our program InfoEXOR have shown extremely better results. Moreover, information theory notation of classical decomposition of switching functions gives new point of view to the existing design styles.

Key words: switching functions, minimization, decomposition, entropy, AND/EXOR expressions.

1. Introduction

In 1938 Shannon introduced the method for decomposition of switching functions, well known as *Shannon expansion*. In 1948 he suggested a measure

Manuscript received March 2, 2000. A version of this paper was presented at the fourth IEEE Conference on Telecommunications in Modern Satellite, Cables and Broadcasting Services, TELSIKS'99, October 1999, Niš, Serbia.

Prof. V. Shmerko, D. Popel and Dr. hab. S. Yanushkevich are with Faculty of Computer Science and Information Theory Systems, Zolnierska 49, 71-210 Szczecin, Poland, e-mails: `[vladimir.shmerko; denis.popel; svetlana.yanushkevich]@wi.ps.pl`. Prof. R. Stanković is with Faculty of Electronic Engineering, 18000 Niš, Serbia, Beogradska 14, e-mail: `stanko@503c1.elfak.ni.ac.yu`. Ph.D. V. Cheushev is with Faculty of Applied Mathematics, Sorina ave. 4, Minsk, Belarus, 220050, e-mail: `cheushev@bsu.unibel.by` .

to represent an information in numerical value, so called *Shannon entropy*. We summarized here an approach to merge both notions and direct it toward one of circuit design problems, namely minimization of AND/EXOR forms of switching functions.

The study of entropy based strategy for minimization of switching functions is based on previously obtained results in AND/OR Decision Trees (DTs) design [2], [5], [7], [8], [13].[1] In our investigation we focus on the minimization of AND/EXOR expressions. Recently, there is a growing interest in CAD of AND/EXOR circuits. Implementation of AND/EXOR circuits often results in a more economical realization of the circuit and is often more easily tested. This is particularly true for applications like error control, arithmetic circuits, and encrypting schemes [3], [9], [10]. However, the known optimization strategies on AND/OR DTs cannot be directly used to optimize of AND/EXOR expressions.

Motivated by these reasons, we address to design of DT with nodes of three types: Shannon (S) positive Davio (pD) and negative Davio (nD) based on information theoretical approach. We consider the following problem: given a switching function, find a quasi-minimal AND/EXOR expression using S, pD or nD expansion.

Our approach revolves around choosing the "best" variable and "best" expansion type for any node of DT. It means that in any step of DT design, we have an opportunity to choose a variable and a type of expansion based on entropy criterion. The result of the DT design is so-called *Free Pseudo Kronecker DT*. The term 'free' means that at every DT level different variables and type of nodes can occur. For some functions free DTs and DDs allow an exponential reduction with respect to the number of nodes compared to ordered ones [16], [4], [6], [11], [17]. So, using entropy-based minimization criterion we solve at the same time two problems: variable ordering and choice of expansion for every node of tree. Optimized DT may be represented in the analytical form: each path from the root to a terminal node corresponds to a term in the AND/EXOR expression.

The rest of paper is organized as follows. In Section 2. we summarize the necessary definitions. Section 3. introduces the information model of Shannon and Davio decomposition. Section 4. describes an algorithm to minimize AND/EXOR expression. In Section 5. we present results to validate our approach. Currently we have InfoEXOR package to realize entropy-based strategy for S, pD, nD expansion and their combinations.

[1] There have been developed many applications in circuit design based on Shannon' information theory. As an examples of successful results and non-solved problems, we refer the reader to [18] and [14].

Section 6. concludes the paper.

2. Basics

Let us formulate the task to be solved as follows. Given a switching function f of n variables from $X = \{x_1, x_2, \cdots, x_n\}$ in the form of truth table (the example of truth table for function Misex24 from LGSynth91-benchmarks is given in Table 1). Let us find a quasi-minimal AND/EXOR expression for given function. We solve this task through conversion of the truth table into *Free Pseudo Kronecker DT*, which corresponds to AND/EXOR expression for f.

Table 1
Truth table of function Misex24

x_1	x_2	x_3	x_4	f	x_1	x_2	x_3	x_4	f
0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	0	0	0
0	1	0	1	1	1	1	0	1	1
0	1	1	0	0	1	1	1	0	0
0	1	1	1	1	1	1	1	1	0

2.1 Decision trees

We consider DT as directed acyclic graph $Tree = \{N, V\}$ with nodes set N and edges set V. Each node is labeled by possible expansion ω with respect to arbitrary variable x. To design DT, we consider tree types of expansion (types of nodes): *Shannon*, *Positive Davio* and *Negative Davio*. We denote a set of expansion types by $\Omega = \{S, pD, nD\}$ and assign a pair (x, ω) to a node, where $x \in X$ and $\omega \in \Omega$. Each node has one incoming edge and two outgoing edges, which correspond to decomposition step of switching function f into cofactors with respect to type of expansion. A terminal edge is labeled with leaf value and has no successors, a nonterminal edge has two successors.

Definition 1: Free DT is designed if each variable is encountered at most once on each path from the root to a terminal node, and variables order may be different for each path.

So, unlike to *ordered DTs*, in *free DTs* any variable may be chosen for any node of tree.

Definition 2: Pseudo Kronecker DT is designed by arbitrary choosing any of S, pD or nD expansion for each node.

In this paper we consider *Free Pseudo Kronecker DTs* as general class. In experiments we investigated with our package InfoEXOR some proper subsets of this class, namely Shannon DTs, fixed polarity Reed-Muller DTs, and Pseudo Reed-Muller DT. For example, *Free Pseudo Reed-Muller DT* is designed by arbitrary choosing of any variable and any of positive Davio or negative Davio expansion for each node. For more detailed definitions see, for example, [16].

2.2 Information measures

In our DT design strategy we use some basic concepts of information theory, namely, *entropy, conditional entropy* and *mutual information* (see, for example, [1]). Here we are focused on mathematical aspects of these concepts.

In order to quantify the content of information revealed by the outcome for finite field of events $A = \{a_1, a_2, \cdots, a_n\}$ with the probabilities distribution $\{p(a_1), p(a_2), \cdots, p(a_n)\}$, Shannon introduced the concept of entropy. *Entropy* of finite field A is given by

$$H(A) = -\sum_{i=1}^{n} p(a_i) \cdot \log p(a_i), \tag{1}$$

where logarithm is in base 2. Note that entropy $H(A)$ never be negative and is equal to zero if and only if A contains one event only.

Let A and B are finite fields of events with probabilities distribution $\{p(a_i)\}, i = 1, 2, \cdots, n$, and $\{p(b_j)\}, j = 1, 2, \cdots, m$, respectively.

Conditional entropy of A with respect to B is defined by

$$H(A|B) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p(a_i, b_j) \cdot \log \frac{p(a_i, b_j)}{p(b_j)}. \tag{2}$$

Mutual information between two finite fields A and B is

$$I(A; B) = H(A) - H(A|B). \tag{3}$$

In our task we deal with two finite fields: set of values of switching function f for different combinations of variables values (we name such combination as *pattern*) and set of values of arbitrary variable x. We use (1)

- (3) to calculate information estimations with respect to function and its variables.

We calculate the probability, for example, $p(f = 0)$ as follows:

$$p(f = 0) = k|_{f=0}/k,$$

where $k_{f=0}$ is the number of patterns, for which the switching function takes the value 0 and k is total number of patterns (for completely specified switching function $k = 2^n$). Other probabilities are calculated in the similar way.

Example 1: Consider the Misex24 function (Table 1). The entropy $H(f) = -{}^4/_{16} \cdot \log {}^4/_{16} - {}^{12}/_{16} \cdot \log {}^{12}/_{16} = 0.81$ bit/pattern. The conditional entropy of function with respect to variable x_2 is $H(f|x_2) = -{}^7/_{16} \cdot \log {}^7/_8 - {}^1/_{16} \cdot \log {}^1/_8 - {}^5/_{16} \cdot \log {}^5/_8 - {}^3/_{16} \cdot \log {}^3/_8 = 0.75$ bit/pattern. Thus, mutual information between f and x_2 (on the other words - information, which is carried out by x_2 about f) is 0.81-0.75 = 0.06 bit/pattern.

3. Information Model of Recursive Decomposition of Switching Functions

Instead giving the information-theoretic interpretation of S, pD, and nD decomposition in a formal way, we cover the main points in an simplified and unified way in order to help the reader to get an impression and quick understanding of the entropy based strategy in DT design.

Consider the designing of DT for switching function f from information theory point of view (Fig. 1). We use two information measures, conditional entropy $H(f|Tree)$ and mutual information $I(f;Tree)$ to describe DT design process.

Initial state of this process when we have not DT is characterized by maximum value for conditional entropy

$$H(f|Tree) = H(f), Tree = \{\emptyset, \emptyset\}. \tag{4}$$

Nodes are recursively attached to DT by using the top-to-down strategy. In this strategy entropy $H(f|Tree)$ of function reduces, information $I(f;Tree)$ increase because variables carry out information about the function. Any *intermediate state* can be described in information-theoretical terms by equation

$$I(f;Tree) = H(f) - H(f|Tree). \tag{5}$$

We maximize the information $I(f;Tree)$ (or minimize the entropy $H(f|Tree)$ that the same) on each step of decision tree design as describe in subsection 3.4.

Final state of decision tree is characterized by

$$H(f|Tree) = 0 \Longleftrightarrow I(f;Tree) = H(f), \tag{6}$$

i.e. $Tree$ represents switching function f (see Fig. 1).

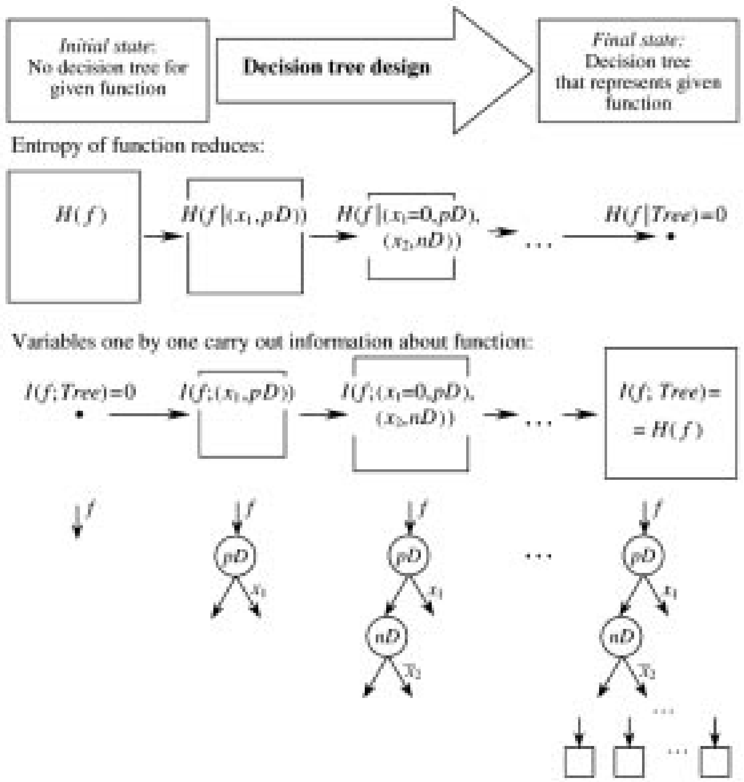

Fig. 1. Information-theoretical model of iterative function decomposition trough DT design

3.1 Information model of expansion of a switching function

We consider the DT design process as *recursive decomposition* of switching function. A step of this recursive decomposition corresponds to the expansion of switching function f with respect to variable x. From information-theoretical position, we say that pair (x, ω) conveys information about function f.

Initial state of the expansion $\omega \in \{S, pD, nD\}$ can be characterized by entropy $H(f)$ of function f, final state - by conditional entropy $H^{\omega}(f|x)$. Likely (5), the ω-expansion of function f with respect to variable x is described as follows

$$I^{\omega}(f;x) = H(f) - H^{\omega}(f|x). \tag{7}$$

Note that if for some node $H^{\omega}(f|x) = 0$ then it means that outgoing branches of this nodes point to leaves.

Now, let us discuss in details the information measures for S, pD and nD expansion.

3.2 Information notation of S expansion

AND/OR trees (also known as *Shannon trees*) are used as appropriate data structures for a wide variety of CAD problems (logic synthesis, verification, diagnosis, testing). Entropy based strategy for Sum-Of-Product minimization that based on AND/OR tree design considered in [7], [8]. This strategy based on S-expansion

$$f = \overline{x} \cdot f_{|x=0} \vee x \cdot f_{|x=1}, \tag{8}$$

where $f_{|x=a}$ is *cofactor* of f, i.e. f with x replaced by $a \in \{0,1\}$.

In this paper we use S-expansion in the form of Exclusive-or Sum-Of-Products

$$f = \overline{x} \cdot f_{|x=0} \oplus x \cdot f_{|x=1}. \tag{9}$$

The designed Shannon tree is mapped into SOP or ESOP expression as follows: a leaf with value 0 is mapped into $f = 0$ and with value 1 - into $f = 1$; a non-terminal node is mapped into f according to (8) or (9) respectively.

Definition 3: Equation

$$H^{S}(f|x) = p_{|x=0} \cdot H(f_{|x=0}) + p_{|x=1} \cdot H(f_{|x=1}) \tag{10}$$

represents an information measure of Shannon expansion for a switching function f with respect to variable x.

Lemma 1: Information measure of S-expansion is equal to conditional entropy $H(f|x)$

$$H^{S}(f|x) = H(f|x). \tag{11}$$

For proofs of lemma and following theorem see [12].

3.3 Information notation of pD and nD expansion

As well known, a switching function f can be represented as pD expansion

$$f = f_{|x=0} \oplus x \cdot (f_{|x=0} \oplus f_{|x=1})$$

or nD expansion

$$f = f_{|x=1} \oplus \overline{x} \cdot (f_{|x=0} \oplus f_{|x=1}).$$

Those follow from S expansion because $x \cdot f_{|x=0}$ and $\overline{x} \cdot f_{|x=1}$ are disjoint.

Definition 4: Equations

$$H^{pD}(f|x) = p_{|x=0} \cdot H(f_{|x=0}) + p_{|x=1} \cdot H(f_{|x=0} \oplus f_{|x=1}), \tag{12}$$
$$H^{nD}(f|x) = p_{|x=1} \cdot H(f_{|x=1}) + p_{|x=0} \cdot H(f_{|x=0} \oplus f_{|x=1}) \tag{13}$$

represent information measures of Positive Davio expansion and Negative Davio expansion of a function f with respect to variable x.

Theorem 1: Information merit (efficiency) to choose pD or nD nodes for DT design in comparison to S expansion are calculated as

$$\triangle I^{pD} = p_{|x=1} \cdot (H(f_{|x=1}) - H(f_{|x=0} \oplus f_{|x=1})) \tag{14}$$

and

$$\triangle I^{nD} = p_{|x=0} \cdot (H(f_{|x=0}) - H(f_{|x=0} \oplus f_{|x=1})) \tag{15}$$

respectively.

3.4 Information criterion for DT design

The main properties of information measures introduced above are (a) recursive character like S, pD and nD expansion in classical notation, and (b) possibility to choose decomposable variable and expansion type based on their information measure. We will show that recursively process allow to order variables in DT based on information criterion which can be near optimal order.

Entropy based optimization of DT design can be described as optimal (with respect to information criterion) node selection process. The path of DT starts from the node and finished by terminal node. Each path corresponded to the term in final expression.

Definition 5: The criterion to choose decomposition variable x and expansion type ω is that the conditional entropy of the function with respect to this variable has to be minimal:

$$H^{\omega}(f|x) \rightarrow \min, \forall (x, \omega) \in X \times \Omega. \tag{16}$$

Note, for completely specified switching function we use the property that $p(x=0)=p(x=1)={}^1/_2$. Thus, we can rewrite information measures defined above as follows: $H^S(f|x) = {}^1/_2(H(f_0)+H(f_1))$, $H^{pD}(f|x) = {}^1/_2(H(f_0)+H(f_2))$, and $H^{nD}(f|x) = {}^1/_2(H(f_1)+H(f_2))$, where $f_0 = f_{|x=0}$, $f_1 = f_{|x=1}$ and $f_2 = f_0 \oplus f_1$.

In order to build Free Pseudo Kronecker DT we choose for each node the argument (i.e. pair (x,ω)) of minimum from $(H(f_0)+H(f_1), H(f_0)+H(f_2), H(f_1)+H(f_2))$ as shown in Example 2.

Fig. 2 illustrates recursive process of decision tree design.

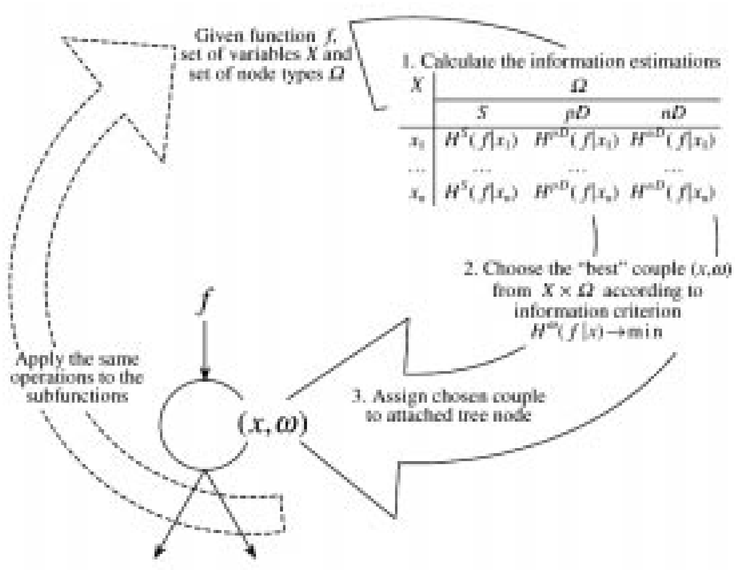

Fig. 2. Entropy based minimization of switching function via recursive DT design: subsequent node attaching

Now, based on introduced formal models we able to describe our algorithm and program InfoEXOR.

4. Algorithm

In this section entropy based algorithm for minimization of AND/EXOR expressions is described. A sketch of algorithm is given in Fig. 4. In this algorithm the ordering restriction is relaxed. This means that (i) each variable appear once on each path and (ii) the order of variables along each path may be different.

Input Switching function $f = f(x_1, x_2, ..., x_n)$, set of expansion types Ω
Output *Tree*: Free DT, AND/EXOR expression
Comments
(i) Function f and its cofactors are given by truth tables
(ii) Notation 'Tree $\leftarrow$' means that tree is attached by node or leaf
(iii) X denotes a set of variables which are not included in current path
(iv) For Pseudo Kronecker and Kroneker DT: $\Omega = \{S, pD, nD\}$,
for pseudo Reed-Muller and Reed-Muller DT: $\Omega = \{pD, nD\}$,
for AND/OR DT: $\Omega = \{S\}$

```
InfoEXOR(f)
{
  if(f = c, where c = const) then {
    Tree ← leaf(c);
    return;
}
for (∀x ∈ X) {
  Determine the functions f|x=0, f|x=1 and f|x=0 ⊕ f|x=1;
  Compute the entropy H(f|x=0), H(f|x=1) and H(f|x=0 ⊕ f|x=1);
  Compute information estimations for S, pD and nD expansion
  accordingly (10), (12) and (13) respectively;
}
Choose variable x and expansion type ω ∈ Ω where H^ω(f|x) → min;
Attach node assigned by couple (x,ω) to Tree
Tree ← node(x,ω);

Determine the functions f_l and f_r for left and right edges
according to chosen expansion:
```

S	pD	nD
$f_l = f_{\|x=0}$ $f_r = f_{\|x=1}$	$f_l = f_{\|x=0}$ $f_r = f_{\|x=0} \oplus f_{\|x=1}$	$f_l = f_{\|x=1}$ $f_r = f_{\|x=0} \oplus f_{\|x=1}$

```
Recursively construct sub-trees for functions f_l and f_r:
Tree_l = InfoEXOR (f_l);
Tree_r = InfoEXOR (f_r);
return; }
```

Fig. 3. Sketch of the algorithm for entropy based switching function minimization.

Example 2: Consider design of Free Pseudo Kronecker Tree for function Misex24 (Table 1). Tree design process and information measures are illustrated in Fig. 4.

Step 1. Choose variable x_4 and S expansion for root node according to minimal entropy $H^S(f|x_4) = {}^1/_2(H(f_0) + H(f_1)) = 0.5$ bit/pattern. Select function $f_l = f_{|x_4=0}$ as current one. This function is constant 0. Select function $f_r = f_{|x_4=1}$.

Step 2. Choose variable x_3 and pD expansion for next node according to minimal entropy $H^{pD}(f|x_3) = {}^1/_2(H(f_0) + H(f_2)) = 0.405$ bit/pattern. Select function $f_l = f_{|x_3=0}$ as current one.

Step 3. Choose variable x_1 and S expansion for next node according to minimal entropy $H^S(f|x_1) = {}^1/_2(H(f_0) + H(f_1)) = 0.5$ bit/pattern. For the last node we do not calculate an information measure. All the others edges point to leaves.

The corresponded to constructed DT Pseudo Reed-Muller expression is: $f = x_4x_2 \oplus x_4x_1x_3$.

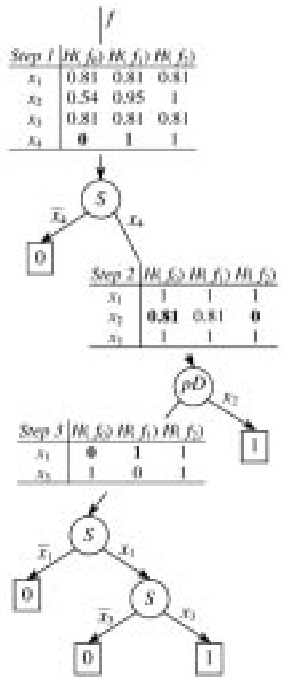

Fig. 4. Step by step Free Pseudo Kronecker Tree design for function misex24 using information measures (in bit/pattern)

5. Experiments

Our *InfoEXOR* program in C++ implements the described above algorithm to design Free Pseudo Kronecker DTs and any it's subset namely Shannon, Reed-Muller and Pseudo Reed-Muller trees.

In experiments we study on several LGSynth91[2] benchmark functions three types of DTs:

[2] http://www.cbl.ncsu.edu/pub/Benchmark_dirs/LGSynth91

(i) RM DTs based on assigning the pD or nD expansion for nodes and fixing the expansion type for each variable;

(ii) Pseudo RM DTs based on an arbitrary choosing any of the pD or nD expansion for each node;

(iii) Pseudo Kronecker DTs based on an arbitrary choosing any of S, pD or nD expansion for each node.

All experiments have been performed on Pentium 100 MHz with 48 MBytes of main memory. In all Tables with experimental results $T/L/t$ denotes the number of terms T, number of literals L and run time t in CPU seconds.

5.1 Fixed polarity RM expressions

In the first series of experiments we studied the criterion $H^{\omega}(f|x) \rightarrow \min$ for $\omega \in \{pD, nD\}$ and fixed expansion type for each variable. The results have been compared with *FDD* - functional decision diagram approach for minimization of fixed polarity RM expressions, reported in [3] (*HP Apollo series* 700 workstation). The fragments of this results are listed for single-output benchmarks in Table 2 (**I/O no.** is the number of input variables/name of output) and for multi-output benchmarks in Table 3. The experiments show that we can obtain the result for the functions much faster in comparison to *FDD*. Our program run times takes into account pre-processing time.

Table 2
Comparison of Drechsler et al. results for FPRM minimization
and the results of *InfoEXOR* (FPRM) for single-output benchmarks

	I/O no.	*FDD* [3] T/t	*InfoEXOR* (FPRM) T/t
rd53	5/2	5/0.4	5/0.001
5xp1	7/1	12/1.8	12/0.001
z4	7/2	9/1.8	9/0.009
rd73	7/2	7/1.8	7/0.001
f51m	8/4	7/3.6	7/0.051
9sym	9/1	173/8.1	173/0.12
life	9/1	100/9.2	100/1.06
sao2	10/2	52/16.3	52/3.83
Total		365/43.0	365/**5.073**

5.2 Pseudo Kronecker expressions

In a final series of experiments we studied the criterion $H^{\omega}(f|x) \rightarrow \min$ for $\omega \in \{S, pD, nD\}$. We compared the minimization results with

EXORCISM-MV3 [15] for minimization of ESOP expressions on benchmarks (Table 4). Our method is much faster and the average number of literals is fewer, in contradiction to the number of terms. It should be pointed out that Pseudo Kronecker expressions are the proper subset of ESOP expressions.

Table 3
Comparison of Drechsler et al. results for FPRM minimization and the results of *InfoEXOR* (FPRM) for multi-output benchmarks

	I/O	*FDD* [3] T/t	*InfoEXOR* (FPRM) T/t
rd53	5/3	20/0.5	20/0.001
rd73	7/3	63/2.3	63/0.001
rd84	8/4	107/5.5	107/0.096
dist	8/5	185/12.5	185/5.32
log8mod	8/5	53/6.5	53/0.310
sao2	10/4	100/48.1	100/1.270
Total		528/75.4	528/**6.988**

Table 4
Comparison of Song and Perkowski results obtained via *EXORCISM-MV3* program for ESOP minimization and *InfoEXOR* for minimization of Pseudo-Kronecker expressions

	I/O	*EXORCISM-MV3* [15] $T/L/t$	*InfoEXOR* (PSDKRO) $T/L/t$
bw	5/28	22/319/1.1	22/65/0.002
rd53	5/3	14/57/0.4	20/45/0.002
squar5	5/8	19/87/0.8	23/56/0.002
con1	7/2	9/37/0.1	14/42/0.81
inc	7/9	26/176/1.7	41/158/0.94
5xp1	7/10	32/170/2.2	45/189/2.08
rd73	7/4	35/188/4.2	63/189/0.02
adr4	8/5	31/144/1.7	34/106/0.04
misex1	8/7	12/82/0.2	15/59/1.32
mlp4	8/8	60/395/16.1	97/466/0.18
rd84	8/4	59/322/20.2	107/352/0.15
9sym	9/1	51/426/4.3	173/636/0.17
Total		401/2559/55.2	654/**2357**/**5.716**

6. Concluding Remarks

The contribution reported here is the development of information theoretical approach to find quasi-minimal AND/EXOR expressions of switching functions. The minimization is maintained via free DTs. In order to identify this results, we investigated the problem from the side of variables ordering and choosing the "best" node type from S, pD or nD for variables, with respect to the minimum entropy criterion. The experiments show that in many cases our program produces results, which are comparable with the results produced by the programs based on other methods, but it works much faster than analogs.

Acknowledgment

Denis Popel acknowledges the support from the State University of Informatics and Radioelectronics, Minsk, Belarus, and the Fond of Fundamental Researches, Belarus.

REFERENCES

1. R.B. Ash: *Information Theory.* John Wiley and Sons, 1967.
2. V. Cheushev, V. Shmerko, D. Simovici, S. Yanushkevich: *Functional entropy and decision trees.* In: Proc. IEEE Int. Symp. on Multiple-Valued Logic, Fukuoka, Japan, 1998, pp. 257-263.
3. R. Drechsler et.al.: *Fast OFDD minimization of fixed polarity Reed-Muller expressions.* IEEE Trans. on Computers, vol.45, No.11, 1996, pp. 1294-1299.
4. J. Gergov, C. Meinel: *Efficient Boolean manipulation with OBDDs can be extended to FBDDs.* IEEE Trans. on Computers, vol.C-43, No.10, 1994, pp. 1197-1209.
5. R. M. Goodman, P. Smith: *Decision tree design from a communication theory standpoint.* IEEE Trans. on Information Theory, vol.IT-34, No.5, 1988, pp. 979-994.
6. W. Gunther, R. Drechsler: *Minimization of free BDDs.* In: Asia and South Pacific Design Automation Conference (ASP-DAC'99), Hong Kong, 1999, pp. 323-326.
7. A.M. Kabakcioglu, P.K. Varshney, C.R.P. Hartman: *Application of information theory to qwitching function minimization.* IEE Proc., Pt E, vol.137, 1990, pp. 389-393.
8. A. Lloris, J. F. Gomez, R. Roman: *Using decision trees for the minimization of multiple-valued functions.* Int. J. Electronics, vol.75, No.6, 1993, pp. 1035-1041.
9. M. Perkowski, L. Jozwiak, R. Drechsler: *New hierarchies of AND/EXOR trees, decision diagrams, lattice diagrams, canonical forms, and regular layouts.* In: Proc. Workshop on Application of Reed-Muller Expansion in Circuit Design, 1997, pp. 115-132.
10. T. Sasao: *Switching Theory for Logic Synthesis.* Kluwer Academic Publishers, Norwell, MA, 1999.
11. T. Sasao, J.T. Butler: *A design method for look-up table type FPGA by Pseudo-Kronecker expansions.* In: Proc. IEEE Int. Symp. on Multiple-Valued Logic, 1994, pp. 97-106.

12. V. Shmerko D. Popel R.S. Stanković, V. Cheushev, S. Yanushkevich: *Information theoretical approach to minimization of And/Exor expressions of switching functions.* In: Proc. of 4-th IEEE Int. Conf. on Telecommunications in Modern Satellite, Cable and Broadcasting Services, Nis, Yugoslavia, vol. 2, 1999, pp. 444-451.
13. D. Simovici, V. Shmerko, V. Cheushev, S. Yanushkevich: *Information estimations of logic functions.* In: Proc. Int. Conf. on Applications of Computer Systems, Szczecin, Poland, 1997, pp. 287-300.
14. D. Simovici, S. Yanushkevich: *Information theory approach in logic design: results, trends and non-solved problems.* In: Booklet of the 8th Int. Workshop on Post-Binary Ultra-Large Scale Integration (ULSI'99), Germany, 1999, pp. 26-29.
15. N. Song, M. Perkowski: *New fast approach to approximate ESOP minimization for incompletely specified multi-output functions.* In: Proc. Workshop on Application of Reed-Muller Expansion in Circuit Design, 1997, pp. 61-72.
16. R. Stanković: *Spectral Transform Decision Diagrams in Simple Questions and Simple Answers.* IP NAUKA, Belgrad,Yugoslavia, 1998, ISBN 86-7621-065-9.
17. R. Stanković, T. Sasao, C. Moraga: *Spectral Transforms Decision Diagrams.* In: T. Sasao, M. Fujita, (eds.), Representations of Discrete Functions, Kluwer Academic Publishers, 1996, pp. 55-92.
18. S.N. Yanushkevich: *Logic Differential Calculus in Multi-Valued Logic Design.* Techn. University of Szczecin Academic Publishers, Poland, 1998, ISBN 83-87423-16-5.