The Boolean Differential Calculus -
a Compact Introduction and Selected Applications

Bernd Steinbach
Institute of Computer Science
Freiberg University of Mining and Technology
Freiberg, Germany
Email: steinb@informatik.tu-freiberg.de

Abstract—The Boolean Differential Calculus is a valuable supplement of the Boolean Algebra and the switching theory. This calculus is based upon the Boolean Algebra and explores the changing behavior of Boolean functions. The definitions of derivative and differential operations of the Boolean Differential Calculus extends the aspects of the problems to solve. Important new solutions, not only in circuit design, could be found in this way by utilizing the Boolean Differential Calculus.

This paper gives a compact introduction into the Boolean Differential Calculus for both single Boolean functions and lattices of Boolean functions. Definitions, interpretations, and theorems provide the basic knowledge. Together with the given selected applications, this paper should motivate the reader to apply the Boolean Differential Calculus to solve further tasks.

Index Terms—Boolean Differential Calculus, derivative operations, differential of a Boolean variable, differential operations, lattice of Boolean functions, applications.

I. INTRODUCTION

The solution of practical problems to test and to synthesize switching circuits forced the scientists Reed [12], Huffman [7] and Akers [1] in the 1970es to develop the roots of the Boolean Differential Calculus (BDC). Basic comprehensive presentations were the books of Thayse and his colleagues [5], [33].

The importance of the BDC for a wide field of applications was noticed by a research group of the University of Chemnitz where the BDC was significantly extended. In 1981 a monograph [3] of this extended calculus was published in German language. In the following period of time many applications were found and published, see, e.g., [2], [8], [13], [18], [19], [24], [26], [27], [28], [30], [31], [32].

Due to the growing number of applications one section of [9] summarizes the theory of the BDC which is used in other sections of this book. Complementary to the textbook [9] the book [25] is available that contains many examples and exercises together with their solutions. Prepared as part of a course about digital circuits and systems [20] contains a section with the main definitions and some applications of the BDC. A comprehensive paper about the BDC [21] was published in the Journal of Computational and Theoretical Nanoscience.

It is also very important to mention that in parallel a comprehensive software package XBOOLE [17] with several versions has been developed where the derivative operations of the Boolean Differential Calculus have been implemented in such a way that they can be used in different areas. This restricts the efforts of researchers to the modeling of the problem using the BDC, and the software is doing, as usual, the rest of the work.

Sometimes, it is necessary to calculate derivative operations of all Boolean functions of a certain lattice. Recently it has been found that the result of each derivative operation for all Boolean functions of a lattice results in another lattice, and both the mark functions and possible restrictions of such a new lattice can be calculated using the known derivative operations of the BDC [15, 23]. First applications of this extended BDC have been published in [16].

The rest of this paper is organized as follows. Section II provides the definitions and interpretations of derivative operations for single Boolean functions. Section III explains some of the theorems for these derivative operations. Section IV introduces the differential of Boolean variables as well as differential operations of Boolean functions. Section V extends the BDC by derivative operations for lattices of Boolean functions. Section VI describes selected applications of the BDC, and Section VII shows that derivative operations can efficiently calculated. Section VIII concludes the paper.

II. DEFINITIONS AND INTERPRETATIONS OF DERIVATIVE OPERATIONS FOR SINGLE BOOLEAN FUNCTIONS

A. Basic Facts

Derivative operations of the BDC [4, 21] can be calculated for Boolean functions. These calculations utilize the known operations of the Boolean Algebra. The results of the derivative operations are again Boolean functions which describe the properties of the given Boolean function regarding a selected direction of change and an explored change of function values.

Derivative operations facilitate the analysis of a given Boolean function regarding the behavior in a selected direction of change. The direction of change is specified by the applied derivative operation. The direction of change can be specified by

1) a single variable,
2) a set of variables which have to be changed simultaneously, or
3) a set of variables where a sequential change of single included variables is assumed.
There are three possible types of value pairs of the given function for the selected direction of change:

1) the function changes the value,
2) the function value remains equal to 1, or
3) the function value remains equal to 0.

It is important to mention, from a practical point of view, that the results of all derivative operation are simpler than the given function. Due to this property, solution algorithms of many practical tasks prefer derivative operations of the BDC. The algorithms to calculate the derivative operation also take advantage of this simplification.

**B. Simple Derivative Operations**

The direction of change is fixed for all simple derivative operations to the selected variable $x_i$. The term *simple* associates that only the change of a single variable is evaluated. The number of variables which are changed for the calculation of a derivative operation is only 1 in the case of simple derivative operations. Therefore the term *simple* can be avoided.

**Definition 1:** Let $f(x) = f(x_i, x_1)$ be a Boolean (logic) function of $n$ variables. Then

$$\frac{\partial f(x_i)}{\partial x_i} = f(x_i, x_1) \oplus f(\overline{x_i}, x_1)$$  

is the (simple) derivative of this function with regard to $x_i$, with $x_1$ being a constant value.

**Definition 2:**

1) the (simple) *minimum* of this function with regard to $x_i$, and

$$\min_{x_i} f(x) = f(x_i, x_1) \land f(\overline{x_i}, x_1)$$  

2) the (simple) *maximum* of this function with regard to $x_i$, and

$$\max_{x_i} f(x) = f(x_i, x_1) \lor f(\overline{x_i}, x_1)$$

the difference between the simple derivative and the simple minimum is that the EXOR-operation $\oplus$ is replaced by the AND-operation $\land$:

$$g_1(c_i, c_1) = \frac{\partial f(x_i, c_1)}{\partial x_i} = f(c_i, c_1) \oplus f(\overline{c_i}, c_1).$$

The derivative $g_1(c_i, c_1) = 1$ only in the case that the function value of $f(c_i, c_1)$ is different from $f(\overline{c_i}, c_1)$; or in other words, the function $f(x_i, c_1)$ changes their value when $x_i$ is changed.

The difference between the simple derivative and the simple minimum is that the EXOR-operation $\oplus$ is replaced by the AND-operation $\land$:

$$g_2(c_i, c_1) = \min_{x_i} f(x_i, c_1) = f(c_i, c_1) \land f(\overline{c_i}, c_1).$$

The minimum $g_2(c_i, c_1) = 1$ only in the case that the function values of both $f(c_i, c_1)$ and $f(\overline{c_i}, c_1)$ are equal to 1; or in other words, for all values of $x_i$ the function value of $f(c_i, c_1)$ is equal to 1 (and does not change to 0).

In the third derivative operation, the simple maximum, the OR-operation $\lor$ is used to compare the function values of $f(c_i, c_1)$ and $f(\overline{c_i}, c_1)$:

$$g_3(c_1) = \max_{x_i} f(x_i, c_1) = f(c_i, c_1) \lor f(\overline{c_i}, c_1).$$

The maximum $g_3(c_i, c_1) = 0$ only in the case that the function values of both $f(c_i, c_1)$ and $f(\overline{c_i}, c_1)$ are equal to 0. Vice versa, the maximum $g_3(c_i, c_1) = 1$ if there exists at least one value 1 for one of these function values.

The constant $c_i$ can be either equal to 0 or equal to 1. It is easy to see that the results of the simple derivative operations are equal for both of these constant values. Hence, we have

$$g_1(c_i, c_1) = g_1(\overline{c_i}, c_1) = g_1(c_1),$$

$$g_2(c_i, c_1) = g_2(\overline{c_i}, c_1) = g_2(c_1),$$

and we can conclude that the results of all simple derivative operations are independent of $x_i$.

**C. Vectorial Derivative Operations**

The change of the value at one point in time is not restricted to a single variable. Vectorial derivative operations describe properties when all variables of the subset $x_0 \subseteq x$ change their values simultaneously. The term *vectorial* emphasizes that all variables of the subset of variables $x_0$ change their values at the same point in time. The same subset of variables $x_0$ is also used to characterize the $m$-fold derivative operations. Hence, the term *vectorial* can not be avoided in a specification of these derivative operations.

**Definition 2:** Let $f(x) = f(x_0, x_1)$ be a Boolean (logic) function of $n$ variables. Then

$$\frac{\partial f(x_0, x_1)}{\partial x_0} = f(x_0, x_1) \oplus f(\overline{x_0}, x_1)$$

is the vectorial derivative of this function with regard to $x_0$, with $x_1$ being a constant value.

$$\min_{x_0} f(x_0, x_1) = f(x_0, x_1) \land f(\overline{x_0}, x_1)$$

the vectorial minimum of this function with regard to $x_0$, and

$$\max_{x_0} f(x_0, x_1) = f(x_0, x_1) \lor f(\overline{x_0}, x_1)$$

the vectorial maximum of this function with regard to $x_0$.

The comparison of Definition 1 and 2 shows that $f(x_0 = 0, x_1)$ of the simple derivative operations is replaced by $f(x_0, x_1)$ for the vectorial derivative operations and $f(x_i = 1, x_1)$ for the simple derivative operations is replaced by $f(x_0, x_1)$ for the vectorial derivative operations, respectively. Hence, both groups of these derivative operations evaluate pairs of functions values of the given function in order to specify the result. The difference between these groups of derivative operations is that the value of more than one variable will change at one point in time in the case of the vectorial derivative operations.

The local analysis of vectorial derivative operations for constant sets of variables $c_0$ and $c_1$ which replace the sets of variables $x_0$ and $x_1$ in Definition 2 emphasizes the meaning of the vectorial derivative operations.

For the vectorial derivative we get

$$g_4(c_0, c_1) = \frac{\partial f(c_0, c_1)}{\partial x_0} = f(c_0, c_1) \oplus f(\overline{c_0}, c_1).$$
The vectorial derivative $g_4(c_0, c_1) = 1$ holds only in the case that the function value of $f(c_0, c_1)$ is different from $f(\bar{c}_0, c_1)$; or in other words, the function $f(c_0, c_1)$ changes their value when $c_0$ changes.

The difference between the vectorial derivative and the vectorial minimum is that the EXOR-operation $\oplus$ is replaced by the AND-operation $\wedge$:

$$g_5(c_0, c_1) = \min_{x_0} f(c_0, c_1) = f(c_0, c_1) \wedge f(\bar{c}_0, c_1) .$$

The vectorial minimum $g_5(c_0, c_1) = 1$ only in the case that the function values of both $f(c_0, c_1)$ and $f(\bar{c}_0, c_1)$ are equal to 1; or in other words, all function values of the compared pair are equal to 1 (and do not change to 0).

In the third vectorial derivative operation, the vectorial maximum, the OR-operation $\vee$ is used to compare the function values of $f(c_0, c_1)$ and $f(\bar{c}_0, c_1)$:

$$g_6(c_0, c_1) = \max_{x_0} f(c_0, c_1) = f(c_0, c_1) \vee f(\bar{c}_0, c_1) .$$

The vectorial maximum $g_6(c_0, c_1) = 0$ only in the case that the function values of both $f(c_0, c_1)$ and $f(\bar{c}_0, c_1)$ are equal to 0. Vice versa, the vectorial maximum $g_6(c_0, c_1) = 1$ if there exists at least one value 1 for one of these function values.

In general, the results of all vectorial derivative operations depend on the same variables as the given function. However, it is easy to see that $g_4(c_0, c_1) = g_4(\bar{c}_0, c_1)$, $g_5(c_0, c_1) = g_5(\bar{c}_0, c_1)$, and $g_6(c_0, c_1) = g_6(\bar{c}_0, c_1)$.

### D. M-fold Derivative Operations

The result of each simple derivative operation is again a Boolean function. Hence, the same simple derivative operation with regard to another variable can be executed for this new function. In this way, we can define the iterative calculation of the following m-fold derivative operations.

**Definition 3:** Let $f(x) = f(x_0, x_1)$ be a Boolean (logic) function of $n$ variables, where $x_0 = (x_1, x_2, \ldots, x_m)$. Then

$$\frac{\partial^m f(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} = \frac{\partial}{\partial x_m} \left( \ldots \left( \frac{\partial}{\partial x_2} \left( \frac{\partial f(x_0, x_1)}{\partial x_1} \right) \right) \ldots \right)$$

is the m-fold derivative of the logic function with regard to $x_0$.

$$\min_{x_0}^m f(x_0, x_1) = \min_{x_0} \left( \ldots \left( \min_{x_2} \left( \min_{x_1} f(x_0, x_1) \right) \right) \ldots \right)$$

the m-fold minimum of the logic function with regard to $x_0$.

$$\max_{x_0}^m f(x_0, x_1) = \max_{x_0} \left( \ldots \left( \max_{x_2} \left( \max_{x_1} f(x_0, x_1) \right) \right) \ldots \right)$$

the m-fold maximum of the logic function with regard to $x_0$, and

$$\Delta_{x_0} f(x_0, x_1) = \min_{x_0}^m f(x_0, x_1) \oplus \max_{x_0}^m f(x_0, x_1)$$

the $\Delta$-operation of the logic function with regard to $x_0$.

The general meaning of all m-fold derivative operations can be found by means of the following considerations. The simple derivative operation with regard to $x_1$ compares pairs of function values and results in identical values for these pairs. The subsequent simple derivative operation with regard to $x_2$ compares again pairs of function values and results in identical values for these pairs. Hence, subspaces of four values carry the same value in the intermediate result of the m-fold derivative operations at this point of the calculation. Finally, the m-fold derivative operations specify a single function value for all $2^m$ positions in the subspaces $x_1 = \text{constant}$. The value in these subspaces depends on the type of the m-fold derivative operation.

- The m-fold derivative with regard to $x_0$ is equal to 1 if an odd number of function values in a subspaces $x_1 = \text{constant}$ is equal to 1.
- The m-fold minimum with regard to $x_0$ is equal to 1 if all $2^m$ function values in a subspaces $x_1 = \text{constant}$ is equal to 1.
- The m-fold maximum with regard to $x_0$ is equal to 0 if all $2^m$ function values in a subspaces $x_1 = \text{constant}$ is equal to 0. Vice versa, the m-fold minimum with regard to $x_0$ is equal to 1 if there exists at least one function value 1 in a subspaces $x_1 = \text{constant}$.
- The $\Delta$-operation combines the m-fold minimum and the m-fold maximum with regard to $x_0$ by an EXOR-operation. Hence, the $\Delta$-operation is equal to 1 if the subspaces $x_1 = \text{constant}$ contain different function values.

Knowing that all simple derivative operations do not depend on $x_1$, it directly follows from Definition 3 that all m-fold derivative operations with regard to $x_0$ are independent of all variables $x_i \in b x_0$.

$$g_7(x_1) = \frac{\partial^m f(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} ,$$

$$g_8(x_1) = \min_{x_0}^m f(x_0, x_1) ,$$

$$g_9(x_1) = \max_{x_0}^m f(x_0, x_1) .$$

### III. Theorems for Derivative Operations of Single Boolean Functions

The Shannon decomposition [14]

$$f(x_i, x_1) = \bar{x}_i f(x_i = 0, x_1) \oplus x_i f(x_i = 1, x_1)$$

$$f(x_i, x_1) = \bar{x}_i f(x_i = 0, x_1) \vee x_i f(x_i = 1, x_1)$$

is valid for all Boolean functions $f(x_i, x_1)$. The two forms (11) and (12) of this decomposition follow from the orthogonality theorem: if

$$C_i \wedge C_j = 0 \text{ , } \forall i \neq j$$

then

$$\bigoplus_{i=1}^{k} C_i = \bigvee_{i=1}^{k} C_i .$$
Applying the Shannon decomposition (11) to the definition of the simple derivative (1) we get:

\[
\frac{\partial f(x)}{\partial x_i} = f(x_i, x_1) \oplus f(x_1, x_i) = x_i f(x_i = 0, x_1) \oplus x_i f(x_i = 1, x_1) = (x_i \oplus x_i) f(x_i = 0, x_1) \oplus (x_i \oplus x_i) f(x_i = 1, x_1) = f(x_i = 0, x_1) \oplus f(x_i = 1, x_1). \tag{15}
\]

Applying the Shannon decomposition (12) to the definition of the simple minimum (2) and utilizing the commutative and distributive law of the Boolean Algebra we get:

\[
\min_{x_i} f(x) = f(x_i, x_1) \lor f(x, x_i) = x_i (f(x_i = 0, x_1) \lor f(x_i = 1, x_1)) \lor x_i (f(x_i = 1, x_1) \lor f(x_i = 0, x_1)) = (x_i \lor x_i) f(x_i = 0, x_1) \lor (x_i \lor x_i) f(x_i = 1, x_1) = f(x_i = 0, x_1) \lor f(x_i = 1, x_1). \tag{16}
\]

Similarly by utilizing the Shannon decomposition (12) for the definition of the simple maximum (3) we get:

\[
\max_{x_i} f(x) = f(x_i, x_1) \lor f(x, x_i) = x_i (f(x_i = 0, x_1) \lor f(x_i = 1, x_1)) \lor x_i (f(x_i = 1, x_1) \lor f(x_i = 0, x_1)) = (x_i \lor x_i) f(x_i = 0, x_1) \lor (x_i \lor x_i) f(x_i = 1, x_1) = f(x_i = 0, x_1) \lor f(x_i = 1, x_1) \tag{17}
\]

Obviously, a Boolean function \( f(x_i, x_1) \) is independent of \( x_i \) if

\[
f(x_i = 0, x_1) = f(x_i = 1, x_1). \tag{18}
\]

This property can be expressed by means of a simple derivative with regard to \( x_i \). The following transformation shows that the Boolean function \( f(x_i, x_1) \) is independent of \( x_i \) if its simple derivative with regard to \( x_i \) is equal to 0:

\[
\frac{\partial f(x_i, x_1)}{\partial x_i} = 0. \tag{19}
\]

Consequently, the independency of \( x_i \) for all simple derivative operations of \( f(x_i, x_1) \) with regard to \( x_i \) can be expressed by:

\[
\frac{\partial}{\partial x_i} \left( \frac{\partial f(x_i, x_1)}{\partial x_i} \right) = 0, \tag{20}
\]

\[
\frac{\partial}{\partial x_i} \left( \min_{x_i} f(x_i, x_1) \right) = 0, \tag{21}
\]

\[
\frac{\partial}{\partial x_i} \left( \max_{x_i} f(x_i, x_1) \right) = 0. \tag{22}
\]

We mentioned already that the vectorial derivatives generally depend on the same variables as the given function. However, there is also a simplification for the vectorial derivative operations. This simplification becomes visible by the following formulas which show that all vectorial derivative operations with regard to \( x_i \) are independent of the common change of all variables of \( x_0 \):

\[
\frac{\partial}{\partial x_i} \left( \frac{\partial f(x_0, x_i)}{\partial x_0} \right) = 0, \tag{23}
\]

\[
\frac{\partial}{\partial x_i} \left( \min_{x_0} f(x_0, x_1) \right) = 0, \tag{24}
\]

\[
\frac{\partial}{\partial x_i} \left( \max_{x_0} f(x_0, x_1) \right) = 0. \tag{25}
\]

The theorems (23), (24), and (25) follow directly from Definition 2.

The \( m \)-fold derivative operations are defined as sequence of simple derivative operations. Hence, the result of each \( m \)-fold derivative operation does not depend on all variables \( x_i \in x_0 \) which belong to their direction of change:

\[
\frac{\partial}{\partial x_i} \left( \frac{\partial^m f(x_0, x_1)}{\partial x_i \partial x_2 \ldots \partial x_m} \right) = 0, \tag{26}
\]

\[
\frac{\partial}{\partial x_i} \left( \min_{x_0}^m f(x_0, x_1) \right) = 0, \tag{27}
\]

\[
\frac{\partial}{\partial x_i} \left( \max_{x_0}^m f(x_0, x_1) \right) = 0, \tag{28}
\]

\[
\frac{\partial}{\partial x_i} (\Delta x_0 f(x_0, x_1)) = 0. \tag{29}
\]

The term minimum is used as name of \( \min_{x_i} f(x) \) due the inequality

\[
\min_{x_i} f(x_i, x_1) \leq f(x_i, x_1) \tag{30}
\]

which is equivalent to the restrictive equation

\[
\min_{x_i} f(x_i, x_1) \lor \overline{f(x_i, x_1)} = 0. \tag{31}
\]

The validity of (30) and (31) follows from (2) and the law of the Boolean Algebra that \( f \lor \overline{f} = 0 \):

\[
f(x_i, x_1) \lor f(x_i, x_1) \lor \overline{f(x_i, x_1)} = 0
\]

\[
0 = 0 .
\]

Similarly, the term maximum is used as name for

\[
\max_{x_i} f(x) \tag{32}
\]

due the inequality

\[
\max_{x_i} f(x_i, x_1) \geq f(x_i, x_1) \tag{33}
\]

which is equivalent to the restrictive equation

\[
\max_{x_i} f(x_i, x_1) \land f(x_i, x_1) = 0. \tag{34}
\]

The validity of (32) and (33) follows from (3) and the laws of the Boolean Algebra that \( f \lor g = f \land g \) (de Morgan) and \( f \lor g = 0 \):

\[
f(x_i, x_1) \lor f(\overline{x_i}, x_1) \land f(x_i, x_1) = 0
\]

\[
f(x_i, x_1) \land f(\overline{x_i}, x_1) \land f(x_i, x_1) = 0
\]

\[
0 = 0 .
\]
The extension of the number of variables from one to more than one in (30) and (32) leads to similar theorems for the vectorial derivative operations

$$\min_{x_0} f(x_0, x_1) \leq f(x_0, x_1)$$ and $$\max_{x_0} f(x_0, x_1) \geq f(x_0, x_1).$$ (34)

Due to Definition 3 and Theorems (30) and (32), for $$f(x) = f(x_0, x_1)$$ and $$x_0 = (x_1, x_2, \ldots, x_m)$$ follow the more general theorems

$$\min_{x_0}^{m} f(x_0, x_1) \leq \ldots \leq \min_{(x_1, x_2)}^{2} f(x) \leq \min_{x_1} f(x) \leq f(x)$$

and

$$f(x) \leq \max_{x_1} f(x) \leq \max_{(x_1, x_2)}^{\infty} f(x) \leq \ldots \leq \max_{x_0}^{m} f(x_0, x_1).$$ (36)

The $$\leq$$ relations of Theorems (36) and (37) combine $$<$$ and $$=.$$ The split of these relations leads to new theorems. In these theorems we use the implication $$a \rightarrow b$$ which is equal to 1 if the premise $$a$$ is equal to 0 or if the premise $$a$$ is equal to 1 and the conclusion $$b$$ is equal to 1. Hence, we have the meaning: if $$a$$ then $$b.$$

Let $$f(x) = f(x_0, x_1, x_1)$$ be a Boolean function, $$x_i \notin x_0$$ and $$x_i \notin x_1.$$ Then

$$\left( \min_{(x_0, x_1)}^{m+1} f(x) < \min_{x_0}^{m} f(x) \right) \rightarrow \partial \min_{x_0}^{m} f(x) \partial x_i = 1.$$ (38)

Using the rules of the Boolean Algebra $$a \rightarrow b = \overline{a} \lor b$$ and $$a \land b = a \lor b,$$ Theorem (38) is equivalent to

$$\left( \min_{(x_0, x_1)}^{m+1} f(x) \land \min_{x_0}^{m} f(x) \right) \lor \partial \min_{x_0}^{m} f(x) \partial x_i = 1$$

and finally to

$$\min_{x_0}^{m} f(x_0, x_1) \lor \min_{x_0}^{m} f(x_0, x_1) \lor \min_{x_0}^{m} f(x_0, x_1) = 1,$$

$$\min_{x_0}^{m+1} f(x) \land \min_{x_0}^{m+1} f(x) = 1.$$ (42)

In the other case where $$\min_{(x_0, x_1)}^{m+1} f(x) = \min_{x_0}^{m} f(x)$$ we have

$$\min_{(x_0, x_1)}^{m+1} f(x) \lor \max_{x_1} \left( \min_{x_0}^{m} f(x) \right) = 1$$

and the following theorem is satisfied.

Let $$f(x) = f(x_0, x_1, x_1)$$ be Boolean function, $$x_i \notin x_0$$ and $$x_i \notin x_1.$$ Then

$$\left( \min_{(x_0, x_1)}^{m+1} f(x) \lor \max_{x_1} \left( \min_{x_0}^{m} f(x) \right) \right) \rightarrow \partial \min_{x_0}^{m} f(x) \partial x_i = 1.$$ (43)

Using the rules of the Boolean Algebra $$a \rightarrow b = \overline{a} \lor b$$ and $$a \land b = a \lor b,$$ Theorem (43) is equivalent to

$$\left( \min_{(x_0, x_1)}^{m+1} f(x) \lor \max_{x_1} \left( \min_{x_0}^{m} f(x) \right) \right) \lor \partial \min_{x_0}^{m} f(x) \partial x_i = 1.$$ (44)

The calculation of the minimum and the maximum with regard to $$x_i$$ in (44) leads to

$$\left( \left( \min_{x_0}^{m} f(x_0, x_1, x_1) \land \min_{x_0}^{m} f(x_0, x_1, x_1) \right) \lor \min_{x_0}^{m} f(x_0, x_1, x_1) \lor \partial \min_{x_0}^{m} f(x) \partial x_i = 1 \right)$$

that can be simplified using the rules $$a \lor b = a \lor b$$ as well as $$a \lor a = 0$$ as follows:

$$\left( \left( \min_{x_0}^{m} f(x_0, x_1, x_1) \land \min_{x_0}^{m} f(x_0, x_1, x_1) \right) \lor \min_{x_0}^{m} f(x_0, x_1, x_1) \lor \partial \min_{x_0}^{m} f(x) \partial x_i = 1 \right)$$

and finally to

$$\min_{x_0}^{m} f(x_0, x_1, x_1) \lor \min_{x_0}^{m} f(x_0, x_1, x_1) \lor \min_{x_0}^{m} f(x_0, x_1, x_1) = 1,$$

$$\min_{x_0}^{m+1} f(x) \land \min_{x_0}^{m+1} f(x) = 1.$$ (42)
Similarly, the \(\geq\) - relations of Theorems (32) and (37) can be split into the relations \(>\) and \(=\). Hence, there are equivalent theorems. Let \(f(x) = f(x_0, x_i, x_1)\) be a Boolean function, \(x_i \notin x_0\) and \(x_i \notin x_1\). Then

\[
\left( \max_{x_0}^m f(x) < \max_{(x_0, x_i)} f(x) \right) \rightarrow \frac{\partial \max_{x_0}^m f(x)}{\partial x_i} = 1 .
\]

(47)

In the other case where

\[
\max_{x_0}^m f(x) = \max_{(x_0, x_i)} f(x)
\]

we have

\[
\min_{x_i} \left( \max_{x_0}^m f(x) \right) \cap \max_{(x_0, x_i)} f(x) = 1 ,
\]

and the following theorem will hold.

Let \(f(x) = f(x_0, x_i, x_1)\) be a Boolean function, \(x_i \notin x_0\) and \(x_i \notin x_1\). Then

\[
\left( \min_{x_i} \left( \max_{x_0}^m f(x) \right) \cap \max_{(x_0, x_i)} f(x) \right) \rightarrow \frac{\partial \max_{x_0}^m f(x)}{\partial x_i} = 1 .
\]

(48)

The proofs of these theorems use transformations which are analogous to the proofs of Theorems (38) and (43).

Sometimes it is helpful to replace an expression with two derivatives by a single other derivative. The simple derivative operations satisfy the restrictive equation

\[
\frac{\partial f(x_0, x_1)}{\partial x_i} + \min_{x_i} f(x_i, x_1) + \max_{x_i} f(x_i, x_1) = 0 .
\]

(49)

This theorem follows directly from Definition 1 and the rules

\(a \lor b = a \oplus b \oplus (a \land b)\) and \(a \oplus a = 0: \)

\[
\begin{align*}
(f(x_i, x_1) \oplus f(x_i, x_1)) & \quad \oplus \\
(f(x_0, x_1) \land f(x_0, x_1)) & \quad \oplus \\
(f(x_0, x_1) \lor f(x_0, x_1)) & = 0 , \\
(f(x_0, x_1) \lor f(x_0, x_1)) & \quad \oplus \\
(f(x_0, x_1) \land f(x_0, x_1)) & \quad \oplus \\
f(x_0, x_1) & \quad \oplus \\
(f(x_0, x_1) \lor f(x_0, x_1)) & \quad \oplus \\
f(x_0, x_1) & = 0 , \\
0 & = 0 .
\end{align*}
\]

Due to the linearity of the \(\oplus\)-operation, Equation (49) can be transformed into

\[
\begin{align*}
\frac{\partial f(x_0, x_1)}{\partial x_i} & = \min_{x_i} f(x_i, x) \oplus \max_{x_i} f(x_i, x) , \\
\min_{x_i} f(x_i, x) & = \frac{\partial f(x_i, x)}{\partial x_i} \oplus \max_{x_i} f(x_i, x) , \\
\max_{x_i} f(x_i, x) & = \frac{\partial f(x_i, x)}{\partial x_i} \oplus \min_{x_i} f(x_i, x) ,
\end{align*}
\]

(50) \hspace{1cm} (51) \hspace{1cm} (52)

so that one simple derivative operation can be expressed by the two other ones.

Due to the similar definition, the restrictive equation (49) and Theorems (50), (51), and (52) can be applied to the vectorial derivative operations:

\[
\begin{align*}
\min_{x_0} f(x_0, x_1) & = \frac{\partial f(x_0, x_1)}{\partial x_0} \oplus \min_{x_0} f(x_0, x_1) \oplus \max_{x_0} f(x_0, x_1) = 0 , \quad (53) \\
\min_{x_0} f(x_0, x_1) & = \frac{\partial f(x_0, x_1)}{\partial x_0} \oplus \max_{x_0} f(x_0, x_1) , \quad (54) \\
\min_{x_0} f(x_0, x_1) & = \frac{\partial f(x_0, x_1)}{\partial x_0} \oplus \max_{x_0} f(x_0, x_1) , \quad (55) \\
\min_{x_0} f(x_0, x_1) & = \frac{\partial f(x_0, x_1)}{\partial x_0} \oplus \max_{x_0} f(x_0, x_1) . \quad (56)
\end{align*}
\]

An equation of the type (49) or (53) follows in the case of the \(m\)-fold derivative operations from the definition of the Delta-operation:

\[
\Delta_{x_0} f(x_0, x_1) \oplus \min_{x_0}^{m} f(x_0, x_1) \oplus \max_{x_0}^{m} f(x_0, x_1) = 0 . \quad (57)
\]

One \(m\)-fold derivative operation can be calculated using the other two \(m\)-fold derivative operations by

\[
\begin{align*}
\Delta_{x_0} f(x_0, x_1) & = \min_{x_0}^{m} f(x_0, x_1) \oplus \max_{x_0}^{m} f(x_0, x_1) , \quad (58) \\
\min_{x_0}^{m} f(x_0, x_1) & = \Delta_{x_0} f(x_0, x_1) \oplus \max_{x_0}^{m} f(x_0, x_1) , \quad (59) \\
\max_{x_0}^{m} f(x_0, x_1) & = \Delta_{x_0} f(x_0, x_1) \oplus \min_{x_0}^{m} f(x_0, x_1) . \quad (60)
\end{align*}
\]

Simplifications in expressions with derivative operations can be reached because logic operations between certain derivative operations are equal to 0. The combination of (30) and (32) is as follows:

\[
\min_{x_i} f(x_i, x_1) \leq f(x_i, x_1) \leq \max_{x_i} f(x_i, x_1) . \quad (61)
\]

From (61) it can be concluded additionally to (31) and (33) also that

\[
\min_{x_i} f(x_i, x_1) \land \max_{x_i} f(x_i, x_1) = 0 . \quad (62)
\]

Furthermore, we have

\[
\frac{\partial f(x_i, x_1)}{\partial x_0} \land \max_{x_i} f(x_i, x_1) = 0 . \quad (63)
\]

The proof of Theorem (63) uses the alternative calculation (53) of the simple derivative and Theorem (62):

\[
\begin{align*}
\frac{\partial f(x_i, x_1)}{\partial x_0} \land \max_{x_i} f(x_i, x_1) & = \\
& = \left( \min_{x_i} f(x_i, x_1) \oplus \max_{x_i} f(x_i, x_1) \right) \land \max_{x_i} f(x_i, x_1) \\
& = \min_{x_i} f(x_i, x_1) \land \max_{x_i} f(x_i, x_1) \\
& = 0 .
\end{align*}
\]

There is a similar relation between the simple minimum and the simple derivative:

\[
\min_{x_i} f(x_i, x_1) \land \frac{\partial f(x_i, x_1)}{\partial x_i} = 0 . \quad (64)
\]
The proof of Theorem (64) uses again the alternative calculation (53) of the simple derivative, several rules of the Boolean Algebra, and finally Theorem (62):

\[
\min_{x_i} f(x_i, x_1) \land \frac{\partial f(x_i, x_1)}{\partial x_i} = \\
= \min_{x_i} f(x_i, x_1) \land \left( \min_{x_i} f(x_i, x_1) \lor \max_{x_i} f(x_i, x_1) \right) \\
= \min_{x_i} f(x_i, x_1) \lor \left( \min_{x_i} f(x_i, x_1) \land \max_{x_i} f(x_i, x_1) \right) \\
= \min_{x_i} f(x_i, x_1) \land \left( 1 \lor \max_{x_i} f(x_i, x_1) \right) \\
= \min_{x_i} f(x_i, x_1) \land \max_{x_i} f(x_i, x_1) \\
= 0.
\]

IV. DIFFERENTIALS OF BOOLEAN VARIABLES AND DIFFERENTIAL OPERATIONS

A. Basic Facts

Differential operations of the BDC [4, 21] can be also calculated for Boolean functions, and these calculations utilize the known operations of the Boolean Algebra. The results of the differential operations are again Boolean functions which depend on the variables of the given function and additionally on differentials of these variables. Hence, the number of Boolean variables of a differential operation can be twice the number of variables of the given function.

The higher number of variables increases the information in the result of a differential operation in comparison to the given function. The additional information reflects the change behavior of the given function with regard to several directions of change. The kind of the evaluated change is specified by the type of the differential operation.

Due to the included differentials of variables, the result of differential operations can be visualized by graphs. This opens an additional large field of applications. The additional Boolean variables extend the effort for practical calculations. However, differential operations have some benefits for the compact representation of a certain changing behavior of a Boolean function. This can be helpful for theoretical explorations.

B. Differentials of Boolean Variables

Boolean variables can carry the values of the Boolean space \( \mathbb{B} = \{0, 1\} \). Hence, they have a static character. The gap of missing dynamic descriptions is filled by the differentials of Boolean variables.

**Definition 4:** Let \( x_i \) be a Boolean variable, then

\[
dx_i = \begin{cases} 
1, & \text{if } x_i \text{ changes its value} \\
0, & \text{if } x_i \text{ does not change its value} 
\end{cases}
\]

(65)
is the differential of the Boolean variable \( x_i \). Change of the value means the transition from \( x_i = 0 \) to \( x_i = 1 \) or vice versa.

There are two important aspects in the relation between the differential \( dx_i \) and the associated Boolean variable \( x_i \).

1) The differential \( dx_i \) is an independent Boolean variable. However, the differential \( dx_i \) is associated with the Boolean variable \( x_i \) because the differential \( dx_i \) describes the change of the value of \( x_i \). Hence, the space for modeling is extended by the differential \( dx_i \).

2) The differential \( dx_i \) is also a Boolean variable. Hence, all laws of the Boolean Algebra remain valid for these differentials which simplifies their applications.

C. Total Differential Operations

Total Differential Operations evaluate the change behavior of a given function \( f(x) \) with regard to all directions of change. Hence, these operations summarize the information of all simple and vectorial derivatives.

**Definition 5:** Let \( f(x) \) be a Boolean (logic) function of \( n \) variables. Then

\[
d_x f(x) = f(x) \lor f(x \oplus dx) \tag{66}
\]
is the (total) differential of \( f(x) \) with regard to \( x \),

\[
\text{Min}_x f(x) = f(x) \land f(x \oplus dx) \tag{67}
\]
the (total) differential minimum of \( f(x) \) with regard to \( x \),

\[
\text{Max}_x f(x) = f(x) \lor f(x \oplus dx) \tag{68}
\]
the (total) differential maximum of \( f(x) \) with regard to \( x \), and

\[
F(x, dx) = f(x) \land \bigwedge_{i=1}^n (dx_i \lor \overline{dx_i}) \tag{69}
\]
the (total) differential expansion of \( f(x) \) with regard to \( x \).

Definition 5 shows that the total differential operations evaluate the relationship between the function \( f(x) \) and the same function under consideration of a certain change \( dx \). Due to the used operation for comparison,

- the result of the total differential describes edges in a graph which connects nodes where the function \( f(x) \) has different values,
- the result of the total differential minimum describes edges in a graph which connects nodes where the function \( f(x) \) is equal to 1, and
- the result of the total differential maximum describes edges in a graph which connects nodes where the function \( f(x) \) carries at least once the value 1.

The total differential expansion is an embedding of the Boolean function \( f(x) \) into the Boolean space \( \mathbb{B}^2n \) of the variables \( x \) and \( dx \). This expansion can be used for comparisons between the total differential operations. The term total emphasizes that all variables \( x_i \) are included into the calculation of the differential operations. This is the general case; hence, the term total can be avoided.
The following theorems show how the simple and vectorial derivative operations are summarized within the total differential operations.

\[
d_x f(x) = \frac{\partial f(x)}{\partial x_1} dx_1 \oplus \ldots \oplus \frac{\partial f(x)}{\partial x_n} dx_n
\]

Vice versa, the vectorial derivative operations can be derived from the total differential operations.

\[
\begin{align*}
\min_{x_0} f(x_0, x_1) &= \frac{\partial f(x_0, x_1)}{\partial x_1} |_{\delta x_0=1, \delta x_1=0} \\
\max_{x_0} f(x_0, x_1) &= \frac{\partial f(x_0, x_1)}{\partial x_1} |_{\delta x_0=1, \delta x_1=0}
\end{align*}
\]

If the vector \( x_0 \) includes only a single variable, the formulas (73), (74) and (75) are usable for simple derivative operations.

\[
\begin{align*}
\frac{\partial f(x_0, x_1)}{\partial x_0} &= d_x f(x_0, x_1) |_{\delta x_0=1, \delta x_1=1} \\
\min_{x_0} f(x_0, x_1) &= \frac{\partial f(x_0, x_1)}{\partial x_0} |_{\delta x_0=1, \delta x_1=1} \\
\max_{x_0} f(x_0, x_1) &= \frac{\partial f(x_0, x_1)}{\partial x_0} |_{\delta x_0=1, \delta x_1=1}
\end{align*}
\]

D. Partial Differential Operations

Sometimes, the change behavior is only needed for a subset of directions of change. The restriction to the evaluation with regard to \( x_0 = (x_1, x_2, \ldots, x_m) \subseteq x \) reduces the effort for the calculation of partial differential operations.

Definition 6: Let \( f(x) = f(x_0, x_1) \) be a Boolean (logic) function of \( n \) variables and \( x_0 = (x_1, x_2, \ldots, x_m) \). Then

\[
d_{x_0} f(x_0, x_1) = f(x_0, x_1) \oplus f(x_0 \oplus \delta x_0, x_1)
\]

is the partial differential of \( f(x) \) with regard to \( x_0 \),

\[
\begin{align*}
\min_{x_0} f(x_0, x_1) &= f(x_0, x_1) \land f(x_0 \oplus \delta x_0, x_1) \\
\max_{x_0} f(x_0, x_1) &= f(x_0, x_1) \lor f(x_0 \oplus \delta x_0, x_1)
\end{align*}
\]

the partial differential minimum of \( f(x) \) with regard to \( x_0 \),

\[
\begin{align*}
\min_{x_0} f(x_0, x_1) &= f(x_0, x_1) \land f(x_0 \oplus \delta x_0, x_1) \\
\max_{x_0} f(x_0, x_1) &= f(x_0, x_1) \lor f(x_0 \oplus \delta x_0, x_1)
\end{align*}
\]

the partial differential maximum of \( f(x) \) with regard to \( x_0 \),

\[
F(x_0, x_1, \delta x_0) = f(x_0, x_1) \land \bigwedge_{i=1}^{m} (d x_i \lor \delta x_i)
\]

the partial differential expansion of \( f(x) \) with regard to \( x_0 \).

The partial differential operations describe the same change behavior as the total differential operations but take only changes in the directions covered by \( \delta x_0 \) into account. The following formulas show for \( x_0 = (x_1, x_2) \) how the simple and vectorial derivative operations are summarized within the partial differential operations.

\[
\begin{align*}
\frac{\partial f(x_1, x_2, x_1)}{\partial x_1} &= \frac{\partial f(x_1, x_2, x_1)}{\partial x_1} dx_1 \oplus \frac{\partial f(x_1, x_2, x_1)}{\partial x_2} dx_2 \\
\min_{x_1} f(x_1, x_2, x_1) &= \frac{\partial f(x_1, x_2, x_1)}{\partial x_1} dx_1 \oplus \frac{\partial f(x_1, x_2, x_1)}{\partial x_2} dx_2 \\
\max_{x_1} f(x_1, x_2, x_1) &= \frac{\partial f(x_1, x_2, x_1)}{\partial x_1} dx_1 \oplus \frac{\partial f(x_1, x_2, x_1)}{\partial x_2} dx_2
\end{align*}
\]

\[
\begin{align*}
\min_{(x_1, x_2)} f(x_1, x_2, x_1) &= f(x_1, x_2, x_1) \oplus f(x_1, x_2, x_1) dx_1 dx_2 \\
\max_{(x_1, x_2)} f(x_1, x_2, x_1) &= f(x_1, x_2, x_1) \oplus f(x_1, x_2, x_1) dx_1 dx_2
\end{align*}
\]

The vectorial derivative operations with regard to \( x_0 = (x_1, x_2) \) can be derived from these partial differential operations as follows:

\[
\begin{align*}
\frac{\partial f(x_1, x_2, x_1)}{\partial (x_1, x_2)} &= d_{(x_1, x_2)} f(x_0, x_1) \bigg|_{dx_1=1, dx_2=1} \\
\min_{(x_1, x_2)} f(x_1, x_2, x_1) &= \frac{\partial f(x_1, x_2, x_1)}{\partial (x_1, x_2)} \bigg|_{dx_1=1, dx_2=1} \\
\max_{(x_1, x_2)} f(x_1, x_2, x_1) &= \frac{\partial f(x_1, x_2, x_1)}{\partial (x_1, x_2)} \bigg|_{dx_1=1, dx_2=1}
\end{align*}
\]
E. M-fold Differential Operations

Two generally valid properties of the partial differential operations allow us the definition of m-fold differential operations:

1) the set of variables \( x_0 \) can be restricted to a single variable \( x_i \),
2) a partial differential operation with regard to \( x_i \) is again Boolean function so that same partial differential operation with regard to another variable \( x_j \) can be calculated for this intermediate function.

**Definition 7:** Let \( f(x) = f(x_0, x_1) \) be a Boolean (logic) function of \( n \) variables and \( x_0 = (x_1, x_2, \ldots , x_m) \). Then

\[
d^m_{x_0} f(x_0, x_1) = d_{x_m} \ldots \left( d_{x_2} \left( d_{x_1} f(x_0, x_1) \right) \right) \ldots \tag{89}
\]

is the m-fold differential of \( f(x) \) with regard to \( x_0 \),

\[
\text{Min}^m_{x_0} f(x_0, x_1) = \text{Min}_{x_m} \ldots \left( \text{Min}_{x_2} \left( \text{Min}_{x_1} f(x_0, x_1) \right) \right) \ldots \tag{90}
\]

the m-fold differential minimum of \( f(x) \) with regard to \( x_0 \),

\[
\text{Max}^m_{x_0} f(x_0, x_1) = \text{Max}_{x_m} \ldots \left( \text{Max}_{x_2} \left( \text{Max}_{x_1} f(x_0, x_1) \right) \right) \ldots \tag{91}
\]

the m-fold differential maximum of \( f(x) \) with regard to \( x_0 \),

and

\[
\partial_{x_0} f(x_0, x_1) = \text{Min}^m_{x_0} f(x_0, x_1) \oplus \text{Max}^m_{x_0} f(x_0, x_1) \tag{92}
\]

the \( \partial \)-operation of \( f(x) \) with regard to \( x_0 \).

A detailed analysis reveals that the m-fold differential can be expressed by

\[
d^m_{x_0} f(x_0, x_1) = \frac{\partial^m f(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} \land dx_1 dx_2 \ldots dx_m \tag{93}
\]

The other m-fold differential operations cover all associated m-fold derivative operations with regard to all subsets of \( x_0 \).

\[
\partial_{x_0} f(x_0, x_1) = \Delta_{x_1} f(x) \ dx_1 \Delta_{x_2} \ldots \Delta_{x_m} + \ldots + \tag{94}
\]

\[
\Delta_{x_m} f(x) \ dx_1 \Delta_{x_2} \ldots dx_m + \ldots + \tag{95}
\]

\[
\Delta_{(x_1, x_2)} f(x) \ dx_1 dx_2 \ldots \Delta_{x_m} + \ldots + \tag{96}
\]

\[
\Delta_{x_0} f(x) \ dx_1 dx_2 \ldots dx_m \tag{97}
\]

\[
\text{Min}_{x_0}^m f(x_0, x_1) = f(x) \overline{dx_1} \overline{dx_2} \ldots \overline{dx_m} \lor \ldots \lor \tag{98}
\]

\[
\min f(x) \overline{dx_1} \overline{dx_2} \ldots dx_m \lor \ldots \lor \tag{99}
\]

\[
\min^2 f(x) \overline{dx_1} dx_2 \ldots \overline{dx_m} \lor \ldots \lor \tag{100}
\]

\[
\min_{x_0}^m f(x) \ dx_1 dx_2 \ldots dx_m \tag{101}
\]

\[
\text{Max}_{x_0}^m f(x_0, x_1) = f(x) \overline{dx_1} \overline{dx_2} \ldots \overline{dx_m} \lor \ldots \lor \tag{102}
\]

\[
\max f(x) \overline{dx_1} \overline{dx_2} \ldots \overline{dx_m} \lor \ldots \lor \tag{103}
\]

\[
\max^2 f(x) \overline{dx_1} dx_2 \ldots \overline{dx_m} \lor \ldots \lor \tag{104}
\]

\[
\max_{x_0}^m f(x) \ dx_1 dx_2 \ldots dx_m \tag{105}
\]

\[
\text{Max}_{x_0}^m f(x_0, x_1) = f(x) \overline{dx_1} \overline{dx_2} \ldots \overline{dx_m} \lor \ldots \lor \tag{106}
\]

\[
\max f(x) \overline{dx_1} \overline{dx_2} \ldots \overline{dx_m} \lor \ldots \lor \tag{107}
\]

\[
\max^2 f(x) \overline{dx_1} dx_2 \ldots \overline{dx_m} \lor \ldots \lor \tag{108}
\]

\[
\max_{x_0}^m f(x) \ dx_1 dx_2 \ldots dx_m \tag{109}
\]

Vice versa, the m-fold derivative operations can be extracted from the m-fold differential operations.

\[
\frac{\partial^m f(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} = d^m_{x_0} f(x_0, x_1) \mid_{dx_0 = 1} \tag{97}
\]

\[
\Delta_{x_0} f(x_0, x_1) = \partial_x f(x_0, x_1) \mid_{dx_0 = 1, dx_1 = 0} \tag{98}
\]

\[
\text{Min}_{x_0}^m f(x_0, x_1) = \text{Min}_x^m f(x_0, x_1) \mid_{dx_0 = 1, dx_1 = 0} \tag{99}
\]

\[
\max_{x_0}^m f(x_0, x_1) = \text{Max}_x^m f(x_0, x_1) \mid_{dx_0 = 1, dx_1 = 0} \tag{100}
\]

If the vector \( x_0 \) includes only a single variable, the formulas (97), (99), and (100) are equivalent to (76), (77), and (78) which can be used for the simple derivative operations.

V. DEFINITIONS AND INTERPRETATIONS OF DERIVATIVE OPERATIONS FOR LATTICES OF BOOLEAN FUNCTIONS

A. Basic Facts

A lattice of Boolean functions is a set of functions which satisfies the definition of a lattice. Lattices of Boolean functions occur, e.g., in circuit design where each function of the lattice can be chosen as a function to realize the circuit structure. Hence, lattices of Boolean functions provide a possibility for optimization in circuit design.

Widely used are lattices which can be modeled as incompletely specified function (ISF). Such an incompletely specified Boolean function divides the \( 2^n \) patterns \( x \) of the Boolean space \( \mathbb{B}^n \) into three disjoint sets:

- \( x \in \text{don’t-care-set} \)
  \[
  \Leftrightarrow f_\varphi(x_1, \ldots , x_n) = 1
  \]
  \[
  \Rightarrow \text{it is allowed to choose the function value of } f(x) \text{ without any restrictions ,}
  \]

- \( x \in \text{ON-set} \)
  \[
  \Leftrightarrow f_\varphi(x_1, \ldots , x_n) = 1
  \]
  \[
  \Leftrightarrow (f_\varphi(x_1, \ldots , x_n) = 0) \land (f(x_1, \ldots , x_n) = 1) ,
  \]

- \( x \in \text{OFF-set} \)
  \[
  \Leftrightarrow f_r(x_1, \ldots , x_n) = 1
  \]
  \[
  \Leftrightarrow (f_r(x_1, \ldots , x_n) = 0) \land (f(x_1, \ldots , x_n) = 0) .
  \]

These mark functions \( f_q(x) \), \( f_r(x) \) and \( f_\varphi(x) \) cover the whole Boolean space

\[
\max^m f(x) \ dx_1 dx_2 \ldots dx_m .
\]

for all vectors \( x \), and they are also mutually disjoint:

\[
f_q(x) \land f_r(x) = 0 ,
\]

\[
f_\varphi(x) \land f_r(x) = 0 ,
\]

\[
f_\varphi(x) \land f_r(x) = 0 .
\]
Generally, a lattice of Boolean functions has the following properties:
- it is closed regarding the AND-operation ($\land$),
- it is closed regarding the OR-operation ($\lor$),
- it has a smallest functions, the infimum $f_q(x)$,
- it has a largest function, the supremum $f_r(x)$,
- it is complementary, i.e., for each function $f_l(x)$ of the lattice exists a complementary function $f_l(x)$ in the lattice that satisfies:
  \[ f_l(x) \land f_l(x) = f_q(x) \text{ and } f_l(x) \lor f_l(x) = f_r(x). \]

[23] showed that lattices which satisfy (108) and (109) are only a subset of all lattices. Selected functions of such a lattice which are independent on certain directions of change satisfy also the rules of a lattice. These directions of change can be stored within an independency matrix.

**Definition 8:** The independency matrix $\text{IDM}$ of a Boolean function $f(x_1, x_2, \ldots, x_n)$ is a Boolean matrix of $n$ rows and $n$ columns. The columns of the independency matrix are associated to the $n$ variables of the Boolean space in the fixed order $(x_1, x_2, \ldots, x_n)$. The independency matrix has the shape of an echelon; all elements below the main diagonal are equal to 0, and the elements above the main diagonal can be equal to 1 only if the element in the main diagonal of the same row is equal to 1. The independency matrix of a Boolean function $f(x_1, x_2, \ldots, x_n)$ must describe all independent directions of change in a unique manner.

A Boolean function of $n$ variables has at most $n$ independent directions of change. The actual number of independent directions of change is implicitly specified by the independency matrix IDM.

**Definition 9:** The rank of an independency matrix $\text{IDM}(f)$ describes the number of independent directions of change of the Boolean function $f(x_1, x_2, \ldots, x_n)$. The rank($\text{IDM}(f)$) is equal to the number of elements 1 in the main diagonal of the unique echelon shape of $\text{IDM}(f)$.

A lattice can be restricted to functions which do not change their values in the case of the simultaneous change of several variables or even of a set of such changes of direction. These directions of change can be expressed by a disjunction of appropriate vectorial derivatives which are uniquely indicated in the independency matrix. For a short notation we define an independency function $f^{id}(x)$.

**Definition 10:** The independency function $f^{id}(x)$ of a Boolean function corresponds to the independency matrix $\text{IDM}(f)$ such that
\[
f^{id}(x) = \bigvee_{i=1}^{n} \frac{\partial f(x)}{\partial x_{0i}}.
\] (110)

if all elements of the row $i$ in $\text{IDM}(f)$ are equal to 0, and
\[ x_j \in x_{0i} \text{ if } \text{IDM}(f)[i,j] = 1. \] (112)

In this way, the Boolean Differential Calculus facilitates the more general definition of lattices of Boolean functions of the lattice functions $f_l(x)$:
\[
f_q(x) \land f_l(x) \lor f_r(x) \lor f^{id}(x) = 0.
\] (113)

It is known from [23] that the result of all derivative operations of a lattice of Boolean functions is again a lattice of Boolean functions of the more general characteristic (113). Hence, it is not necessary to calculate the needed derivative operations of all functions of the lattice separately, but the calculation of the mark functions of the new lattice and the adjustment of the independency matrix IDM of the result lattice are sufficient.

It is important to know whether a lattice contains functions which depend on the explored direction of change. [23] provides for this analysis [23] the algorithm
\[
s_{min} = \text{MIDC}(\text{IDM}(f), x_0)\] (114)

which calculates the minimal independent direction of change. Another algorithm
\[
\text{IDM}(g) = \text{UM}(\text{IDM}(f), bx_0)\] (115)

realizes the unique merge of the new independent direction of change into the independency matrix IDM.

**B. Simple Derivative Operations**

It is known that the results of all simple derivative operations of a single Boolean function with regard to $x_i$ are independent of $x_i$: see (20), (21), and (22). Hence, the results of all simple derivative operations of a lattice of Boolean functions must be a lattice of Boolean functions which do not depend on $x_i$.

**Theorem 1:** Let $f_l(x) = f_l(x_1, x_2, \ldots, x_n)$ be a Boolean function of $n$ variables that belongs to a lattice defined by (113) where $f_q(x)$ and $f_r(x)$ satisfy (105), and $f_l(x)$ depends on $x_i$:
\[
\text{MIDC}(\text{IDM}(f_l), x_i) > 0.
\] (116)

Then all simple derivatives
\[
g_l(x) = \frac{\partial f_l(x)}{\partial x_i}
\] (117)
belong to a Boolean lattice defined by
\[
f_q^{\partial x_i}(x_1) \land g_l(x) \lor f_r^{\partial x_i}(x_1) \lor g_{l_i}^{id}(x) = 0\] (118)
with the mark functions of the simple derivative of the lattice with regard to $x_i$
\[
f_q^{\partial x_i}(x_1) = \max_{x_i} f_q(x, x_1) \land \max_{x_i} f_r(x, x_1),
\] (119)
\[
f_r^{\partial x_i}(x_1) = \min_{x_i} f_q(x, x_1) \lor \min_{x_i} f_r(x, x_1),
\] (120)
and the independency function $g_{l_i}^{id}(x)$ associated to
\[
\text{IDM}(g_l) = \text{UM}(\text{IDM}(f_l), x_i),
\] (121)
all simple minimums
\[
g_2(x) = \min_{x_i} f_l(x)
\] (122)
belong to a Boolean lattice defined by
\[ f_q^{\text{min}_{x_i}}(x_1) \land g_2(x) \lor g_2(x) \land f_r^{\text{min}_{x_i}}(x_1) \lor g_2^{id}(x) = 0 \] (123)
with the mark functions of the simple minimum of the lattice with regard to \( x_i \)
\[ f_q^{\text{min}_{x_i}}(x_1) = \min_{x_i} f_q(x_1, x_1) , \] (124)
\[ f_r^{\text{min}_{x_i}}(x_1) = \max_{x_i} f_r(x_1, x_1) , \] (125)
and the independency function \( g_2^{id}(x) \) associated to
\[ \text{IDM}(g_2) = \text{UM}(\text{IDM}(f_i), x_i) , \] (126)
and all simple maximums
\[ g_3(x) = \max_{x_i} f_i(x) \] (127)
belong to a Boolean lattice defined by
\[ f_q^{\text{max}_{x_i}}(x_1) \land g_3(x) \lor g_3(x) \land f_r^{\text{max}_{x_i}}(x_1) \lor g_3^{id}(x) = 0 \] (128)
with the mark functions of the simple maximum of the lattice with regard to \( x_i \)
\[ f_q^{\text{max}_{x_i}}(x_1) = \max_{x_i} f_q(x_1, x_1) , \] (129)
\[ f_r^{\text{max}_{x_i}}(x_1) = \min_{x_i} f_r(x_1, x_1) , \] (130)
and the independency function \( g_3^{id}(x) \) associated to
\[ \text{IDM}(g_3) = \text{UM}(\text{IDM}(f_i), x_i) . \] (131)
The three independency functions are equal to each other:
\[ g_1^{id}(x) = g_2^{id}(x) = g_3^{id}(x) , \] (132)
with
\[ \text{IDM}(g) = \text{UM}(\text{IDM}(f_i), x_i) . \] (133)
The variable \( x_i \) does not occur in the resulting mark functions of all simple derivative operations of a lattice of Boolean functions with regard to \( x_i \). Hence, these mark functions and consequently all functions of the lattice do not depend on \( x_i \). This property of the lattice will be emphasized by the derivative \( \frac{\partial g(x)}{\partial x_i} \) in the definition (113) of a lattice of Boolean functions.

C. **Vectorial Derivative Operations**

It is known that the results of all vectorial derivative operations of a single Boolean function with regard to \( x_0 \) are independent of the simultaneous change of all variables of \( x_0 \): see (23), (24), and (25). Hence, the results of all vectorial derivative operations of a lattice of Boolean functions must be a lattice of Boolean functions which do not depend on \( x_0 \).

**Theorem 2:** Let \( f_i(x) = f_i(x_1, x_2, \ldots, x_n) \) be a Boolean function of \( n \) variables that belongs to a lattice defined by the equation (113) where \( f_q(x) \) and \( f_r(x) \) hold (105), and \( f_1(x) \) depend on the simultaneous change of all variables of \( x_0 \):
\[ \text{MIDC}(\text{IDM}(f_i), x_0) > 0 . \] (134)
Then all vectorial derivatives
\[ g_1(x) = \frac{\partial f_i(x_0, x_1)}{\partial x_0} \] (135)
belong to a Boolean lattice defined by
\[ f_q^{\partial x_0}(x_0, x_1) \land g_1(x) \lor g_1(x) \land f_r^{\partial x_0}(x_0, x_1) \lor g_1^{id}(x) = 0 \] (136)
with the mark functions of the vectorial derivative of the lattice with regard to \( x_0 \)
\[ f_q^{\partial x_0}(x_0, x_1) = \max_{x_0} f_q(x_0, x_1) \land \max_{x_0} f_r(x_0, x_1) , \] (137)
\[ f_r^{\partial x_0}(x_0, x_1) = \min_{x_0} f_q(x_0, x_1) \lor \min_{x_0} f_r(x_0, x_1) , \] (138)
and the independency function \( g_1^{id}(x) \) associated to
\[ \text{IDM}(g_1) = \text{UM}(\text{IDM}(f_i), x_0) , \] (139)
all vectorial minimums
\[ g_2(x) = \min_{x_0} f_i(x) \] (140)
belong to a Boolean lattice defined by
\[ f_q^{\text{min}_{x_0}}(x_0, x_1) \land g_2(x) \lor g_2(x) \land f_r^{\text{min}_{x_0}}(x_0, x_1) \lor g_2^{id}(x) = 0 \] (141)
with the mark functions of the vectorial minimum of the lattice with regard to \( x_0 \)
\[ f_q^{\text{min}_{x_0}}(x_0, x_1) = \min_{x_0} f_q(x_0, x_1) , \] (142)
\[ f_r^{\text{min}_{x_0}}(x_0, x_1) = \max_{x_0} f_r(x_0, x_1) , \] (143)
and the independency function \( g_2^{id}(x) \) associated to
\[ \text{IDM}(g_2) = \text{UM}(\text{IDM}(f_i), x_0) . \] (144)
and all vectorial maximums
\[ g_3(x) = \max_{x_0} f_i(x) \] (145)
belong to a Boolean lattice defined by
\[ f_q^{\text{max}_{x_0}}(x_0, x_1) \land g_3(x) \lor g_3(x) \land f_r^{\text{max}_{x_0}}(x_0, x_1) \lor g_3^{id}(x) = 0 \] (146)
with the mark functions of the simple maximum of the lattice with regard to \( x_0 \)
\[ f_q^{\text{max}_{x_0}}(x_0, x_1) = \max_{x_0} f_q(x_0, x_1) , \] (147)
\[ f_r^{\text{max}_{x_0}}(x_0, x_1) = \min_{x_0} f_r(x_0, x_1) , \] (148)
and the independency function \( g_3^{id}(x) \) associated to
\[ \text{IDM}(g_3) = \text{UM}(\text{IDM}(f_i), x_0) . \] (149)
The three independency functions are equal to each other:
\[ g_1^{id}(x) = g_2^{id}(x) = g_3^{id}(x) = g^{id}(x) \tag{150} \]
with
\[ \text{IDM}(g) = \text{UM}(\text{IDM}(f_i), x_0) \tag{151} \]

Due to (151), all lattice functions belonging to the results of a vectorial derivative operation are simpler than the functions of the given lattice.

D. M-fold Derivative Operations

It is known that the results of all m-fold derivative operations of a single Boolean function with regard to \( x_0 \) are independent of all \( x_i \in x_0 \): see (26), (27), (28), and (29). Hence, the results of all m-fold derivative operations of a lattice of Boolean functions must be a lattice of Boolean functions which do not depend on all \( x_i \in x_0 \).

**Theorem 3:** Let \( f_i(x) = f_i(x_1, x_2, \ldots, x_n) \) be a Boolean function of \( n \) variables that belongs to the lattice defined by (113) where \( f_q(x) \) and \( f_r(x) \) satisfy (105), and \( f_i(x) \) is not independent on all variable \( x_i \in x_0 \):
\[ \exists x_i \in x_0 : \quad \text{MIDC}(\text{IDM}(f_i), x_i) > 0 \tag{152} \]

Then all m-fold derivatives
\[ g_1(x) = \frac{\partial^m f_i(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} \tag{153} \]
belong to a Boolean lattice defined by
\[ f_q^{\partial x_1, \partial x_2 \ldots \partial x_m}(x_1) \land g_1(x) \land f_r^{\partial x_1, \partial x_2 \ldots \partial x_m}(x_1) \lor g_1^{id}(x) = 0 \tag{154} \]
with the mark functions of the m-fold derivatives with regard to \( x_0 \)
\[ f_q^{\partial x_1, \partial x_2 \ldots \partial x_m}(x_1) = \frac{\partial^m f_q(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} \land \min_{x_0}^m(f_q(x) \lor f_r(x)) \tag{155} \]
\[ f_r^{\partial x_1, \partial x_2 \ldots \partial x_m}(x_1) = \frac{\partial^m f_r(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} \land \min_{x_0}^m(f_q(x) \lor f_r(x)) \tag{156} \]
and the independency function \( g_1^{id}(x) \) associated to IDM(\( g_1 \)) satisfies
\[ \forall x_i \in x_0 : \quad \text{MIDC}(\text{IDM}(g_1), x_i) = 0 \tag{157} \]
all m-fold minimums of the lattice (113) with regard to \( x_0 \)
\[ g_2(x) = \min_{x_0}^m f_i(x_0, x_1) \tag{158} \]
belong to a Boolean lattice defined by
\[ f_q^{\min_{x_0}^m}(x_1) \land g_2(x) \land f_r^{\min_{x_0}^m}(x_1) \lor g_2^{id}(x) = 0 \tag{159} \]
with the mark functions of the m-fold minimums with regard to \( x_0 \)
\[ f_q^{\min_{x_0}^m}(x_1) = \min_{x_0}^m f_q(x_0, x_1) \tag{160} \]
\[ f_r^{\min_{x_0}^m}(x_1) = \max_{x_0}^m f_r(x_0, x_1) \tag{161} \]
and the independency function \( g_2^{id}(x) \) associated to IDM(\( g_2 \)) satisfies
\[ \forall x_i \in x_0 : \quad \text{MIDC}(\text{IDM}(g_2), x_i) = 0 \tag{162} \]
al all m-fold maximums of the lattice
\[ g_3(x) = \max_{x_0}^m f(x_0, x_1) \tag{163} \]
belong to a Boolean lattice defined by
\[ f_q^{\max_{x_0}^m}(x_1) \land g_3(x) \land f_r^{\max_{x_0}^m}(x_1) \lor g_3^{id}(x) = 0 \tag{164} \]
with the mark functions of the m-fold maximums with regard to \( x_0 \)
\[ f_q^{\max_{x_0}^m}(x_1) = \max_{x_0}^m f_q(x_0, x_1) \tag{165} \]
\[ f_r^{\max_{x_0}^m}(x_1) = \min_{x_0}^m f_r(x_0, x_1) \tag{166} \]
and the independency function \( g_3^{id}(x) \) associated to IDM(\( g_3 \)) satisfies
\[ \forall x_i \in x_0 : \quad \text{MIDC}(\text{IDM}(g_3), x_i) = 0 \tag{167} \]
and all \( \Delta \)-operations with regard to \( x_0 \)
\[ g_4(x) = \Delta_{x_0} f_i(x_0, x_1) \tag{168} \]
belong to a Boolean lattice defined by
\[ f_q^{\Delta_{x_0}}(x_1) \land g_4(x) \land f_r^{\Delta_{x_0}}(x_1) \lor g_4^{id}(x) = 0 \tag{169} \]
with the mark functions of the \( \Delta \)-operations with regard to \( x_0 \)
\[ f_q^{\Delta_{x_0}}(x_1) = \max_{x_0}^m f_q(x_0, x_1) \land \max_{x_0}^m f_r(x_0, x_1) \tag{170} \]
\[ f_r^{\Delta_{x_0}}(x_1) = \min_{x_0}^m f_q(x_0, x_1) \land \min_{x_0}^m f_r(x_0, x_1) \tag{171} \]
and the independency function \( g_4^{id}(x) \) associated to IDM(\( g_4 \)) satisfies
\[ \forall x_i \in x_0 : \quad \text{MIDC}(\text{IDM}(g_4), x_i) = 0 \tag{172} \]
The four independency functions are equal to each other:
\[ g_1^{id}(x) = g_2^{id}(x) = g_3^{id}(x) = g_4^{id}(x) = g^{id}(x) \tag{173} \]
with
\[ \text{IDM}(g) = \text{UM}(\ldots \text{UM}(\text{IDM}(f_i), x_1), \ldots, x_m) \tag{174} \]
All variable \( x_i \in x_0 \) do not occur in the result mark functions of all m-fold derivative operations of a lattice of
functions which are linear in \( x \). Hence, these mark functions and consequently all functions of the lattice do not depend on all \( x_i \in x \). This property of the lattice will be emphasized by the associated derivatives \( \frac{\partial f(x)}{\partial x_i} \) in the definition (113) of a lattice of Boolean functions.

VI. APPLICATIONS OF THE BOOLEAN DIFFERENTIAL CALCULUS

A. Properties of Boolean Functions

Constant values of the simple derivative indicate special properties of the explored function. It has already been shown (19) that a Boolean function \( f(x_i, x_1) \) is independent of \( x_i \) if

\[
\frac{\partial f(x)}{\partial x_i} = 0 .
\]

A Boolean function \( f(x_i, x_1) \) is linear in \( x_i \) if the variable \( x_i \) can be separated using an EXOR-operation:

\[
 f(x_i, x_1) = x_i \oplus g(x_1)
\]

(175)

Functions which are linear in \( x_i \) satisfy

\[
 \frac{\partial f(x_i, x_1)}{\partial x_i} = 1 ,
\]

(176)

because the substitution of (175) into (176) leads to

\[
 \frac{\partial f(x)}{\partial x_i} = (x_i \oplus g(x_1)) + (\overline{x_i} \oplus g(x_1))
\]

\[
 = x_i \oplus \overline{x_i} \oplus g(x_1) + g(x_1)
\]

\[
 = 1 .
\]

A Boolean function \( f(x) \) is self-dual if

\[
 f(x) = f(\overline{x}) .
\]

(177)

Functions which are self-dual satisfy

\[
 \frac{\partial f(x)}{\partial x} = 1 ,
\]

(178)

because the substitution of (177) into (178) leads to

\[
 \frac{\partial f(x)}{\partial x} = f(x) \oplus f(\overline{x})
\]

\[
 = f(\overline{x}) \oplus f(x)
\]

\[
 = 1 \oplus f(\overline{x}) + f(x)
\]

\[
 = 1 .
\]

A Boolean function \( f(x_i, x_j, x_1) \) is symmetric with regard to \( (x_i, x_j) \) if the exchange of the variables \( x_i \) and \( x_j \) does not change the function:

\[
 f(x_i, x_j, x_1) = f(x_j, x_i, x_1) ,
\]

(179)

which can be transformed into

\[
 f(x_i = 0, x_j = 1, x_1) = f(x_i = 1, x_j = 0, x_1) .
\]

(180)

Functions which are symmetric with regard to \( (x_i, x_j) \) satisfy

\[
 (x_i \oplus x_j) \wedge \frac{\partial f(x_i, x_j, x_1)}{\partial (x_i, x_j)} = 0 ,
\]

(181)

because the application of the Shannon decomposition and with the substitution of (181) into (181) leads to

\[
 (x_i \oplus x_j) \wedge \frac{\partial f(x_i, x_j, x_1)}{\partial (x_i, x_j)} = (x_i \overline{x_j} \vee \overline{x_i} x_j) \wedge
\]

\[
 (f(x_i, x_j, x_1) \oplus f(x_i, \overline{x_j} x_1))
\]

\[
 = (x_i \overline{x_j} \vee \overline{x_i} x_j) \wedge (\overline{x_i} \overline{x_j} [f(0, 0, x_1) \oplus f(1, 1, x_1)] \vee
\]

\[
 x_i \overline{x_j} [f(0, 1, x_1) \oplus f(1, 0, x_1)] \vee
\]

\[
 x_i x_j [f(1, 0, x_1) \oplus f(0, 1, x_1)] \vee
\]

\[
 x_i x_j [f(1, 0, x_1) \oplus f(0, 0, x_1)]
\]

\[
 = \overline{x_i} x_j [f(0, 1, x_1) \oplus f(1, 0, x_1)] \vee
\]

\[
 x_i \overline{x_j} [f(1, 0, x_1) \oplus f(0, 1, x_1)] \vee
\]

\[
 x_i \overline{x_j} [f(0, 1, x_1) \oplus f(0, 1, x_1)]
\]

\[
 = 0 .
\]

More generally, a Boolean function

\[
 f(x) = f(x_1, x_2, \ldots, x_n)
\]

is symmetric with regard to all variables \( x_i, i = 1, \ldots, n \), if

\[
 \bigvee_{i=2}^n (x_i \oplus x_i) \wedge \frac{\partial f(x)}{\partial x_i} = 0 .
\]

(182)

B. Graph Equations Differentials of Boolean Variables

The differential \((d)x_i\) describes the change of the value of the Boolean variable \( x_i \). Problems, where changes of certain variables are relevant, can be modeled and solved using these additional variables. A differential \((d)x_i\) can describe, e.g., the change of a signal in a digital circuit. Here, we solve a very simple constraint problem.

A ferryman \( f \), a wolf \( w \), a goat \( g \), and a cabbage \( c \) have to cross a river. However, there is only a very small boat and special interests of a meal. Therefore, the following constraints:

1) Neither the wolf and the goat nor the wolf and the cabbage can stay without the ferryman on one bank.
2) Only the ferryman is able to control the boat.
3) The remaining space in the boat suffices only for one of the other three items.
4) The state of item 1 is also forbidden as result of a change.
5) The stay of all four “traveler” on one bank is forbidden, because it prevents that the other bank is reached.

This problem can be modeled and solved using a system of graph equations. The value of the Boolean variables \( f, w, g \), and \( c \) specify the bank where the travelers stay:

left bank: \( f = w = g = c = 0 \),

right bank: \( f = w = g = c = 1 \).

The differentials of these variables describe whether the travelers stay on one bank or cross the river (change the bank):

stay on one bank: \( df = dw = dg = dc = 0 \),

change the bank: \( df = dw = dg = dc = 1 \).
one output pattern $y$ for each input pattern $x$. In the second case, the system function $F(x,y)$ can permit several output patterns for the same input pattern as source of optimizations, but missing permitted output patterns for some input patterns prohibit the synthesis. Hence, it must be checked whether the system function $F(x,y)$ can be realized by a combinational circuit.

The condition for the realization of a given system function $F(x,y)$ by a combinational circuit is that the characteristic system equation

$$F(x,y) = 1$$

is solvable with regard to all $y_i \in y$. This condition is satisfied if

$$\max_y^m F(x,y) = 1.$$  \hspace{1cm} (186)\)

If (186) is not satisfied, the input patterns for the missing behavior are determined by

$$f_{mis}(x) = \max_y^m F(x,y).$$

Otherwise, the solution functions

$$y_1 = f_1(x),$$

$$y_2 = f_2(x),$$

$$\vdots,$$

$$y_m = f_m(x)$$

(188)

can be calculated. Each of these solution functions can either be uniquely determined or chosen from a lattice of Boolean functions. The system equation (185) is uniquely solvable with regard to $y_i$ if

$$\frac{\partial}{\partial y_i} \left( \max_{y \neq y_i}^{m-1} F(x,y) \right) = 1.$$  \hspace{1cm} (189)\)

If (189) is satisfied the function $f_i(x)$ of (188) can be calculated by

$$f_i(x) = \max_y^m (y_i \land F(x,y)).$$

(190)

If the system equation (185) is solvable with regard to $y$ (the condition (186) is satisfied), but it is not uniquely solvable (the condition (189) is not satisfied) then the solution function $f_i(x)$ can be chosen out of a lattice of Boolean functions with the don’t care function

$$f_{\phi i}(x) = \min_{y_i} \left( \max_{y \neq y_i}^{m-1} F(x,y) \right)$$

(191)

and the mark functions $f_{qi}(x)$ and $f_{ri}(x)$:

$$f_{qi}(x) = \max_y^m (y_i \land F(x,y)) \land f_{\phi i}(x),$$

(192)

$$f_{ri}(x) = \max_y^m (\overline{y_i} \land F(x,y)) \land f_{\phi i}(x).$$

(193)

The mark functions $f_{qi}(x)$ and $f_{ri}(x)$ must often be computed as subtask in the synthesis of digital circuits. An
alternative but also expensive possibility to calculate these mark functions is:

\[ f_{qi}(x) = \max_y (y_i \land F(x, y)) \land \max_y (\overline{y_i} \land F(x, y)) , \]

(194)

\[ f_{ri}(x) = \max_y (y_i \land \overline{F(x, y)}) \land \max_y (\overline{y_i} \land F(x, y)) . \]

(195)

The theorems of the BDC help to restrict this effort. We demonstrate this simplification for the infimum of the lattice \( f_{qi}(x) \). The function \( f_{qi}(x) \) (194) will not change when the expression is extended by an OR-operation and a constant value 0:

\[ f_{qi}(x) = \max_y (y_i \land F(x, y)) \land \max_y (\overline{y_i} \land F(x, y)) \lor 0 . \]

(196)

The value 0 can be expressed by the conjunction of a function and the complement of the same function:

\[ f_{qi}(x) = \left( \max_y (y_i \land F(x, y)) \land \max_y (\overline{y_i} \land F(x, y)) \right) \lor \left( \max_y (y_i \land F(x, y)) \lor \max_y (\overline{y_i} \land F(x, y)) \right) . \]

(197)

The application of the distributive law leads to

\[ f_{qi}(x) = \max_y (\overline{y_i} \land F(x, y)) \land \left( \max_y (y_i \land F(x, y)) \lor \max_y (\overline{y_i} \land F(x, y)) \right) . \]

(198)

The disjunction of the \( m \)-fold maximum with regard to the same variables of two different functions is equal to the \( m \)-fold maximum of the disjunction of these functions:

\[ f_{qi}(x) = \max_y (\overline{y_i} \land F(x, y)) \land \max_y (y_i \land F(x, y)) \lor \overline{y_i} \land F(x, y) . \]

(199)

The application of the distributive law within the expression in the second \( m \)-fold maximum leads to a disjunction \( y_i \lor \overline{y_i} \) which is equal to 1.

\[ f_{qi}(x) = \max_y (\overline{y_i} \land F(x, y)) \land \max_y (F(x, y) \lor (y_i \lor \overline{y_i})) \]

\[ = \max_y (\overline{y_i} \land F(x, y)) \land \max_y (F(x, y)) . \]

(200)

The second \( m \)-fold maximum in (200) is equal to 1 due to the satisfied condition (186) to solve system equation (185) with regard to all output variables \( y_i \in y \). Hence, we get the simpler formula

\[ f_{qi}(x) = \max_y (\overline{y_i} \land F(x, y)) \]

(201)

to calculate the mark function \( f_{qi}(x) \) of the lattice. In a similar manner the simpler formula

\[ f_{ri}(x) = \max_y (y_i \land F(x, y)) \]

(202)

can be derived for the other mark function \( f_{ri}(x) \) of the lattice.

A necessary condition in quantum computing is that the function to be implemented is reversible [6]. The number of inputs and the number of outputs are the same for reversible Boolean functions. A system function \( F(x, y) \) is reversible if and only if it is uniquely solvable with regard to all \( x_i \in x \):

\[ \forall x_i \in x : \quad \frac{\partial}{\partial x_i} \left( \max^{m-1} F(x, y) \right) = 1 , \quad (203) \]

and uniquely solvable with regard to all \( y_i \in y \):

\[ \forall y_i \in y : \quad \frac{\partial}{\partial y_i} \left( \max^{m-1} F(x, y) \right) = 1 . \quad (204) \]

**D. Decomposition of Boolean Functions**

Povarov suggested in [11] a simple, but powerful decomposition of a given Boolean function. Figure 2 shows the associated circuit structure.

The function \( h(g(x_a), x_b) \) can be represented by

\[ h(g(x_a), x_b) = h_1(x_b) \oplus g \land h_2(x_b) . \]

(205)

One possibility to calculate the function \( h_1(x_b) \) for \( x_a = (x_{a1} \cdot x_{a2} \cdot \ldots \cdot x_{am}) \) is

\[ h_1(x_b) = \max^{m} (x_{a1} \cdot x_{a2} \cdot \ldots \cdot x_{am} \land f(x_a, x_b)) . \]

(206)

Using the circuit structure of Figure 2 and (205), a simplified intermediate function \( f'(x_a, x_b) \) can be constructed:

\[ f'(x_a, x_b) = f(x_a, x_b) \oplus h_1(x_b) \]

\[ = h(g(x_a), x_b) \oplus h_1(x_b) \]

\[ = h_1(x_b) \oplus g(x_a) \land h_2(x_b) \oplus h_1(x_b) \]

\[ = g(x_a) \land h_2(x_b) . \]

(207)

This intermediate function must be strongly AND-bi-decomposable. Hence, a Povarov decomposition exists if

\[ \overline{f'(x_a, x_b)} \land \max^{m} f'(x_a, x_b) \land \max^{m} f'(x_a, x_b) = 0 . \]

(208)

If (208) is satisfied, the decomposition function

\[ g(x_a) = \max^{m} f'(x_a, x_b) , \]

(209)

\[ h_2(x_b) = \max^{m} f(x_a, x_b) . \]

(210)

The Povarov decomposition splits the set of input variables into two disjoint sets. The decomposition functions \( g(x_a) \) (209) and \( h(g(x_a), x_b) \) (205), specified by \( h_1(x_b) \) (206) and \( h_2(x_b) \) (206), are much simpler than the given function.
Unfortunately, only few functions satisfy the (208) of the Povarov decomposition.

Another possibility to decompose a Boolean function is the strong bi-decomposition which is basically not disjoint. Hence, the possibility that functions satisfy the condition of a strong bi-decomposition is much higher than in the case of the Povarov decomposition.

The strong bi-decomposition utilizes the behavior of the simple logic gates (OR, AND, or XOR) that the function on the output of the gate combines the logic functions of the two associated inputs. The requirement for the simplification is that the decomposition functions on the inputs of the gate depend on less variables than the function on the output. The strong bi-decomposition reaches this requirement by dividing the set of all variables $x$ into three disjoint subsets $x_a$, $x_b$, and $x_c$ where $x_a$ is only used in the decomposition function $g(x_a, x_c)$, $x_b$ is only used in the decomposition function $h(x_b, x_c)$, and the remaining variables $x_c$ are commonly used by both decomposition functions.

**Definition 11 (Strong Bi-Decompositions):** A strong OR-bi-decomposition

$$f(x_a, x_b, x_c) = g(x_a, x_c) \lor h(x_b, x_c),$$

(a strong AND-bi-decomposition

$$f(x_a, x_b, x_c) = g(x_a, x_c) \land h(x_b, x_c),$$

or a strong XOR-bi-decomposition

$$f(x_a, x_b, x_c) = g(x_a, x_c) \oplus h(x_b, x_c)$$

(213)

decomposes a logic function $f(x_a, x_b, x_c)$ into the decomposition functions $g(x_a, x_c)$ and $h(x_b, x_c)$.

Figure 3 shows the circuit structures of these three types of strong bi-decompositions.

The existence of a strong bi-decomposition of a logic function $f(x)$ with regard to dedicated sets of variables $x_a \subseteq x$ and $x_b \subseteq x$ for each of the three gates OR, AND, or XOR is a property of the given function $f(x)$.

**Theorem 4 (Conditions for Strong Bi-Decompositions):** A lattice of Boolean functions with the mark functions $f_q(x_a, x_b, x_c)$, $f_r(x_a, x_b, x_c)$, and $f_p(x_a, x_b, x_c)$ contains at least one functions $f(x_a, x_b, x_c)$ that is strongly bi-decomposable with regard to the dedicated sets of variables $x_a$ and $x_b$ for an OR-gate if and only if

$$f_q(x_a, x_b, x_c) \land \max_{x_a} f_r(x_a, x_b, x_c)$$

$$\land \max_{x_b} f_p(x_a, x_b, x_c) = 0,$$  

(214)

for an AND-gate if and only if

$$f_r(x_a, x_b, x_c) \land \max_{x_a} f_q(x_a, x_b, x_c)$$

$$\land \max_{x_b} f_p(x_a, x_b, x_c) = 0,$$  

(215)

or for an XOR-gate with $x_a = a$ and $x_b$ if and only if

$$\max_{x_b} f_q^{\alpha_a}(x_b, x_c) \land f_r^{\alpha_a}(x_b, x_c) = 0,$$  

(216)

where $f_q^{\alpha_a}(x_b, x_c)$ and $f_r^{\alpha_a}(x_b, x_c)$ are the mark functions of the simple derivative of the given lattice with regard to $a$, see (119) and (120).

The lattices of the decomposition functions can be calculated for an OR-bi-decomposition by

$$g_q(x_a, x_c) = \max_{x_b} (f_q(x_a, x_b, x_c) \land \max_{x_a} f_q(x_a, x_b, x_c)),$$  

(217)

$$g_r(x_a, x_c) = \max_{x_b} f_r(x_a, x_b, x_c),$$  

(218)

$$h_q(x_b, x_c) = \max_{x_a} (f_q(x_a, x_b, x_c) \land g(x_a, x_c)),$$  

(219)

$$h_r(x_b, x_c) = \max_{x_a} f_r(x_a, x_b, x_c),$$  

(220)

where $g(x_a, x_c)$ in (219) is the chosen decomposition function of the lattice with the mark functions (217) and (218).

The lattices of the decomposition functions can be calculated for an AND-bi-decomposition by

$$g_q(x_a, x_c) = \max_{x_b} f_q(x_a, x_b, x_c),$$  

(221)

$$g_r(x_a, x_c) = \max_{x_b} (f_r(x_a, x_b, x_c) \land \max_{x_a} f_q(x_a, x_b, x_c)),$$  

(222)

$$h_q(x_b, x_c) = \max_{x_a} f_q(x_a, x_b, x_c),$$  

(223)

$$h_r(x_b, x_c) = \max_{x_a} (f_r(x_a, x_b, x_c) \land g(x_a, x_c)),$$  

(224)

where $g(x_a, x_c)$ in (224) is the chosen decomposition function of the lattice with the mark functions (221) and (222).

The decomposition function $g(a, x_c)$ of an EX-OR-bi-decomposition is uniquely specified by

$$g(a, x_c) = a \land \max_{x_b} f_q^{\alpha a}(x_b, x_c),$$  

(225)
and the associated decomposition function \( h(x_0, x_c) \) can be chosen from the lattice with the mark functions

\[
h_q(x_0, x_c) = \max_a \left( (g(a, x_c) \land f_q(a, x_0, x_c)) \lor (g(a, x_c) \land f_r(a, x_0, x_c)) \right),
\]

\[
h_r(x_0, x_c) = \max_a \left( (g(a, x_c) \land f_q(a, x_0, x_c)) \lor (g(a, x_c) \land f_r(a, x_0, x_c)) \right).
\]

More details about these strong bi-decompositions are given in [10] and [21]. The additional use of the weak bi-decomposition established a complete synthesis method for all combinational functions.

E. Test of Combinational Circuits

In microelectronic circuits typically only the inputs and the outputs, but not the internal wires of the circuit are available. Widely used is the network model of a sensible path, if these conditions become true and a change of \( f \) through several gates to the output \( f(x) \). Such a sensible path can be a physical path that satisfies several conditions [10].

1) If the sensible path goes through an AND-gate, the values of the other inputs of this gate must be constant equal to 1.

2) If the sensible path goes through an OR-gate, the values of the other inputs of this gate must be constant equal to 0.

3) If the sensible path goes through an EXOR-gate, the values of the other inputs of this gate must be constant.

It can be concluded that there is no stuck-at error on the sensible path if these conditions become true and a change of the input value \( x_i \) of the sensible path causes the change of its output \( f(x) \). Formula (228) emphasizes that there are very strong constraints for a sensible path:

\[
\bigwedge_j \left( g_j^{\text{and}}(x) \frac{\partial g_j^{\text{and}}(x)}{\partial x_i} \right) \land \\
\bigwedge_j \left( g_j^{\text{or}}(x) \frac{\partial g_j^{\text{or}}(x)}{\partial x_i} \right) \land \\
\bigwedge_j \left( g_j^{\text{exor}}(x) \frac{\partial g_j^{\text{exor}}(x)}{\partial x_i} \right)
\]

The method can be implicitly used in parallel to the synthesis by bi-decomposition [29]. All functions \( g_j^{\text{and}}, g_j^{\text{or}}, g_j^{\text{exor}} \) are calculated as necessary part of the design process. In this way the \( NP \)-hard task of test pattern generation needs only about 10 percent additional time within the synthesis process.

VII. EFFICIENT ALGORITHMS FOR DERIVATIVE OPERATIONS

The examples of Section VI show the benefits of the Boolean Differential Calculus in order to create a model of the problem, to derive a solution method, and to simplify the necessary calculations. However, it remains the computation of derivative operations as necessary link in the chain to the final practical solution. Efficient algorithms for derivative operations provide this link.

The following algorithms assume that the functions are given as a ternary vector list (TVL) in orthogonal disjunctive or antivalence form (ODA), and the dedicated variables of the derivative operations are specified as a set of variables (SV).

**Algorithm 1** \( m \)-fold derivative of \( f(x_0, x_1) \) with regard to \( x_0 = (x_1, x_2, \ldots, x_m) \)

**Require:** TVL of \( f = f(x_0, x_1) \) in ODA-form

**Require:** SV of \( x_0 = (x_1, x_2, \ldots, x_m) \)

**Ensure:** TVL of \( g(x_1) = \frac{\partial^m f(x_0, x_1)}{\partial x_1 \partial x_2 \ldots \partial x_m} \)

1: \( h \leftarrow \text{DRCD}(f, x_0) \)

2: \( h \leftarrow \text{SFORM}({\text{DC}}(h, x_0), \text{A-form}) \)

3: \( g \leftarrow \text{ORTH}(h) \)

Algorithm 1 specifies a very simple procedure to calculate the \( m \)-fold derivative. It shows that it is not necessary to execute a sequence of simple derivatives as defined in (7). The \( m \)-fold derivative is equal to 1 in subspaces where an odd number of function values 1 exists.

A ternary vector that contains a dash (−) in one of the columns \( x_0 \) describes an even number of function values. Hence, these vectors do not contribute to the solution and are deleted by the function \( \text{DRCD}(f, x_0) \) delete rows containing dashes in columns \( x_0 \) in line 1 of Algorithm 1.

The solution function \( g(x_1) \) does not depend on \( x_0 \). Hence, the function \( \text{DC}(h, x_0) \) deletes the columns \( x_0 \). The XBOOLE-operation \( \text{SFORM} \) assigns the antivalence form to the result TVL in line 2 of Algorithm 1. The orthogonalization of this antivalence form using the XBOOLE-operation \( \text{ORTH} \) in line 2 of Algorithm 1 results in the wanted \( m \)-fold derivative.

**Example 1:** The 2-fold derivative of the function

\[
f(x_1, x_2, x_3, x_4, x_5) = x_1x_2x_3x_5 \lor x_3x_4x_5 \lor x_1x_3x_4x_5 \lor x_1x_2x_4x_5 \lor x_1x_2x_3x_4x_5 \lor x_1x_2x_3x_4x_5 = x_1x_2x_3x_5 \oplus x_3x_4x_5
\]
with regard to \(x_0 = (x_1, x_3)\) has to be calculated.

\[
x_1 \ x_2 \ x_3 \ x_4 \ x_5 \quad (229)
\]

The XBOOLE-operation ORTH of line 3 of Algorithm 1 deletes the identical rows two and three and removes the common binary vector \((100)\) from row number one and four. Hence, the result of this calculation is

\[
g(x_2, x_4, x_5) = \begin{pmatrix}
x_2 & x_4 & x_5 \\
1 & 0 & 0
\end{pmatrix},
\]

and we get the 2-fold derivative

\[
g(x_2, x_4, x_5) = \frac{\partial^2 f(x_1, x_2, x_3, x_4, x_5)}{\partial x_1 \partial x_3} = x_2 x_4 x_5.
\]

The simple derivative is a special case of the \(m\)-fold derivative with \(m = 1\). Hence, Algorithm 1 can also be used to calculate a simple derivative.

**Algorithm 2** \(m\)-fold maximum of \(f(x_0, x_1)\) with regard to \(x_0 = (x_1, x_2, \ldots, x_m)\)

**Require**: TVL of \(f = f(x_0, x_1)\) in ODA-form

**Require**: SV of \(x_0 = (x_1, x_2, \ldots, x_m)\)

**Ensure**: TVL of \(g(x_1) = \max_{x_0} m f(x_0, x_1)\)

1: \(h \leftarrow \text{SFORM}(\text{DC}(f, x_0), \text{D-form})\)

2: \(g \leftarrow \text{ORTH}(h)\)

Algorithm 2 is again simpler than Algorithm 1. It is also not necessary to execute a sequence of simple maxima as defined in (9). The \(m\)-fold maximum is equal to 1 in subspaces where at least one function value 1 exists. Hence, all columns of \(x_0\) can be removed. The remaining rows indicate subspaces for which the \(m\)-fold maximum is equal to 1.

**Example 2**: We take the same function (229) of Example 1 and calculate the 2-fold maximum with regard to \(x_0 = (x_1, x_3)\). Due to line 1 of Algorithm 2 the shadowed columns are deleted in

\[
x_1 \ x_2 \ x_3 \ x_4 \ x_5
\]

and together with the assignment of the disjunctive form we get

\[
x_2 \ x_4 \ x_5
\]

The result of the orthogonalization in line 2 of Algorithm 2 is

\[
x_2 \ x_4 \ x_5
\]

and a simplification using the XBOOLE-operation OBB (orthogonal block building) leads to

\[
x_2 \ x_4 \ x_5
\]

Hence, 2-fold maximum of \(f(x_1, x_2, x_3, x_4, x_5)\) (229) is

\[
g(x_2, x_4, x_5) = \max_{(x_1, x_3)}^2 f(x_1, x_2, x_3, x_4, x_5) = x_2 \lor x_5.
\]

The simple maximum is a special case of the \(m\)-fold maximum with \(m = 1\). Hence, Algorithm 2 can also be used to calculate a simple maximum.

It is possible to calculate the \(m\)-fold minimum by means of the \(m\)-fold maximum:

\[
\min_{x_0}^m f(x_0, x_1) = \max_{x_0}^m f(x_0, x_1).
\]

However, the inner complement operation is time-consuming because all variables \(x = (x_0, x_1)\) must be computed. The calculation of a sequence of simple minima finds the solution of the \(m\)-fold minimum faster, because the number of variables is reduced in each swap of the iteration.

**VIII. CONCLUSIONS**

This compact introduction into the Boolean Differential Calculus (BDC) gives an overview of this extension to the
Boolean Algebra. Initialized by Reed, Huffman, and Akers to solve a special practical problem, the comprehensive theory of the BDC was developed by some groups of researches. An important milestone in this development is the monograph [4] by Bochmann and Posthoff.

The BDC deals with Boolean functions and is well structured. Generally, the BDC explores the behavior of changes of Boolean functions.

Derivative operations of the BDC describe these properties of Boolean functions when the direction of change is fixed. Depending on the number of variables which are involved in this change we distinguish between simple and vectorial derivative operations. The behavior of change within whole subspaces is explored by the $m$-fold derivative operations. In each of these three groups the derivative, the minimum, and the maximum describes a special change behavior. It is an important property that the results of all derivative operations are simpler than the given function.

Several directions of change are commonly explored by differential operations of Boolean functions. Here, we distinguish also among groups: the total differential operations, the partial differential operations, and the $m$-fold differential operations. These differential operations summarize the information of several associated derivative operations. Due to these summaries, the differential operations are more complex than the given function. Hence, the main value of the differential operations is the compact representation of properties of the change of Boolean function within theoretical explorations.

Based on the ten derivative operations and the nine differential operations, the BDC provides a huge number of theorems which allow us to transform found expression containing such operations and to find new relationships or simpler possibilities for needed calculations. Sections III and IV refer only a subset of these theorems.

Practical applications deal often not only with single Boolean functions but with many Boolean functions of a lattice. Recently, it was found that all derivative operations transform a given lattice into another lattice. The application of the BDC was the key to the description of a more general lattice of Boolean functions. Section V summarizes the new and so far only partially utilized knowledge about derivatives of lattices of Boolean functions.

There is a very wide field of applications of the BDC. Section VI enumerates some applications about properties of Boolean functions, the graph equations with differentials of variables, the utilization of the system function of digital circuits, several types of decompositions, and the test of combinational circuits. There are many further applications. We completely excluded, for instance, the concept of a Boolean differential equation [22] for this introduction into the BDC.

The performance of the BDC is based upon both the comprehensive theory and the very efficient algorithm to calculate differential operations. Some of them are explored in Section VII.

References


