

# Foundations for Applications of Gibbs Derivatives in Logic Design and VLSI

RADOMIR S. STANKOVIĆ<sup>a,\*</sup>, MILENA STANKOVIĆ<sup>a</sup> and REINER CREUTZBURG<sup>b</sup>

<sup>a</sup>*Faculty of Electronics, Department Computer Science, Beogradska 14, 18 000 Niš, Yugoslavia;* <sup>b</sup>*Fachhochschule Brandenburg, Department of Computer Science, University of Applied Sciences, P. O. Box 2132, Brandenburg an der Havel D-14737, Germany and Institute of Digital and Computer Systems, Tampere University of Technology, Tampere, FIN-33101, Finland*

(Received 20 January 2000; In final form 4 October 2000)

New technologies and increased requirements for performances of digital systems require new mathematical theories and tools as a basis for future VLSI CAD systems. New or alternative mathematical approaches and concepts must be suitable to solve some concrete problems in VLSI and efficient algorithms for their efficient application should be provided. This paper is an attempt in this direction and relates with the recently renewed interest in arithmetic expressions for switching functions, instead representations in Boolean structures, and spectral techniques and differential operators in switching theory and applications.

Logic derivatives are efficiently used in solving different tasks in logic design, as for example, fault detection, functional decomposition, detection of symmetries and co-symmetries of logic functions, etc. Their application is based on the property that by differential operators, we can measure the rate of change of a logic function. However, by logic derivatives, we can hardly distinguish the direction of the change of the function, since they are defined in finite algebraic structures.

Gibbs derivatives are a class of differential operators on groups, which applied to logic functions, permit to overcome this disadvantage of logic derivatives. Therefore, they may be useful in logic design in the same areas where the logic derivatives have been already using. For such applications, it is important to provide fast algorithms for calculation of Gibbs derivatives on finite groups efficiently in terms of space and time.

In this paper, we discuss the methods for efficient calculation of Gibbs derivatives. These methods should represent a basis for further applications of these and related operators in VLSI CAD systems.

**Keywords:** Logic design; Spectral transforms; Walsh transform; Vilenkin–Chrestenson transform; Differential operators; Gibbs derivatives

## INTRODUCTION AND MOTIVATION

With increased complexity and more demanding practical applications, a continuous advent and improvement of VLSI CAD systems become imperative. The performance of these systems greatly depends on the suitability of the used mathematical theories and efficiency of the data structures for representing discrete functions.

The present state-of-art are VLSI CAD systems based on Boolean algebraic structures and related theories. However, alternative approaches are continuously presented from earlier times until now [17,18,20,30,31,33,35,37,48,75], and challenges of new technologies renewed interest in such research attempts. For more details, we refer to a discussion of the development of switching theory in Ref. [46].

Many tasks in logic design and VLSI can be performed through series of manipulations and calculations with different analytic expressions for switching functions efficiently in terms of space and time. In this setting, different AND/EXOR expressions [45] are considered as analogues of polynomial or polynomial-like expressions for real variable functions. The coefficients in these expressions are taken from the union of the set of function values and the set of the Boolean differences of all orders and with respect to all the variables. More closer analogy with real variable functions is achieved when the arithmetic expressions for switching functions are used [20,32,37,74]. In these expressions, the coefficients are function values, with logic values 0 and 1 interpreted as the integers 0 and 1, and the partial Gibbs derivatives [23].

\*Corresponding author. E-mail: [rstankovic@elfak.ni.ac.yu](mailto:rstankovic@elfak.ni.ac.yu)

Further applications of such expressions and related derivatives as an alternative for improving the performance of Boolean difference and other logic derivatives are ultimately related with efficient methods for their calculations. That offers a motivation for further considerations in this paper.

## Differential Operators in Logic Design and VLSI

Differential operators are an important tool in signal processing. They are used to measure the rate of change of a signal, as well as to determine the direction of the change of it. The logic differential operators are used in a similar context in logic design and VLSI design regarded as subareas of signal processing devoted to the processing of a special class of signals, the logic signals [30,31]. In these areas, differential operators are efficiently used to solve many important problems. For example, they have proved efficient in fault detection, functional decomposition, symmetry and co-symmetry properties detection. For more details, see for example Refs. [12,46]. Moreover, there is some renewed interest in application of logic differential operators [1,2,27,36,49,73,74].

However, in switching and multiple-valued (MV) functions theory, we are using finite algebraic structures, as for example, the finite (Galois) fields  $GF(2)$  and  $GF(p)$ , respectively. A problem in dealing with differential operators in such structures is that some properties of these operators vanish for the properties of the used algebraic structures. For example, the Boolean difference is considered as proper counterpart of differential operators for switching functions considered as functions in  $GF(2)$  [2,66]. However, we cannot differentiate the direction of change of a switching function by the Boolean difference, since the addition and subtraction performed modulo 2 coincide, and a change from 0 to 1, and vice versa, produce the identical result. A similar problem appears with logic differential operators for MV functions.

Gibbs derivatives are a broad family of differential operators on groups, see Ref. [5] and references given there. For a review of more recent results, we refer to Refs. [55,72]. Gibbs derivatives on finite groups are a subclass of these operators restricted to applications on discrete and digital signals defined at a finite number of points [54]. They are somewhere denoted as hybrid operators [24], since are applied to functions whose arguments are taken in finite algebraic structures, and function values are allowed to be the integers, the real numbers, or the complex numbers. From this property, the Gibbs derivatives applied to switching and MV functions permit to overcome the above discussed problem in application of logic derivatives. Thanks to that, the application of Gibbs derivatives in logic design was suggested probably for the first time in Refs. [13,14,30].

In these papers, the Gibbs derivatives on finite dyadic groups were used as auxiliary operators to determine all the possible Boolean differences of a given switching functions  $f$  efficiently in terms of space and time. Related

applications of Gibbs derivatives in detection of symmetry properties of switching functions were suggested in Ref. [30]. Recently, applications of Gibbs derivatives in logic design were reviewed and further extended in Refs. [43, 64].

For these reasons, we have found interesting to consider methods for efficient calculations of Gibbs derivatives through Decision diagrams (DDs), which are nowadays a standard part of CAD systems for logic design [12,46], and also are used in MV design [39,56].

The interest in Gibbs derivatives on different, not necessarily Abelian groups, has been found in recent works on different topologies for circuit design. The topologies derived from Fibonacci sequences are suggested to overcome some particular disadvantages of Boolean topologies [15,16,29]. Different groups may generate different shift operators, which may be useful in definition of new topologies for circuit synthesis. Therefore, for such and related applications, it may be useful to have developed efficient methods for calculation of Gibbs derivatives on finite not necessarily Abelian groups.

The paper is organized as follows. In the second section, we briefly introduce some basic definitions and notations. In the third section, we give some basic definitions of the Gibbs derivatives on finite groups. In the fourth section, we give the matrix interpretation of these operators, which is used in their calculation by different methods FFT-like briefly reviewed in the fifth section. In the sixth section, we discuss the calculation of Gibbs derivatives on Abelian groups, and in the seventh section on non-Abelian groups through DDs. Some basic concepts in group representations, which may be useful in reading this paper are given in the Addendum.

## DEFINITIONS AND NOTATIONS

### Discrete Functions

Discrete signals defined on a finite set of a given number of  $g$  points are conveniently represented by vectors  $\mathbf{f} = [f(0)\dots f(g-1)]^T$ .

Some algebraic structure should be imposed on the index set  $\{0, 1, \dots, g-1\}$ , as well as to the range of  $f$ , to get a mathematically tractable model for  $f$ .

The structure of a group  $G$  of order  $g$  is the weakest structure still permitting Fourier analysis for  $f$  useful in many practical applications [69]. The structure of a field  $P$ , which may be the real field  $R$ , the complex field  $C$  or a finite (Galois) field  $GF(p)$  is usually assumed for the range of  $f$ . We denote by  $P(G)$  the space of functions on  $G$  into  $P$ . It is often assumed, to provide the existence of the Fourier transform in  $P(G)$ , that

1.  $\text{char } P = 0$ , or  $\text{char } P$  does not divide  $g$ ,
2.  $P$  is a so-called splitting field for  $G$ , where  $\text{char } P$  is the characteristic of  $P$ .

TABLE I Assignment of matrices to the levels for calculation of partial Gibbs derivatives

	$\mathbf{D}_1$	$\mathbf{D}_2$	$\mathbf{D}_3$
$q_1$	$\mathbf{D}_{G_2}$	$\mathbf{I}_2$	$\mathbf{I}_2$
$q_2$	$\mathbf{I}_3$	$\mathbf{D}_{G_3}$	$\mathbf{I}_3$
$q_3$	$\mathbf{I}_3$	$\mathbf{I}_3$	$\mathbf{D}_{G_3}$

We assume that  $G$  is a decomposable group,

$$G = X_{i=1}^n G_i, \quad g = \prod_{i=1}^n g_i, \quad g_1 \leq g_2 \leq \dots \leq g_n. \quad (1)$$

where  $g_i$  is order of the constituent subgroup  $G_i$ .

Each  $x \in G$  can be uniquely represented as

$$x = \sum_{i=1}^n a_i x_i, \quad x_i \in G_i, \quad x \in G,$$

with

$$a_i = \begin{cases} \prod_{j=i+1}^n g_j, & i = 1, \dots, n-1 \\ 1, & i = n. \end{cases} \quad (2)$$

where  $g_i$  is the order of  $G_i$ .

The group operation  $\delta$  of  $G$  can be expressed in terms of the group operation  $\delta_i$  of the subgroups  $G_i$ ,  $i = 1, \dots, n$ , by:

$$x\delta y = (x_1 \overset{1}{\delta} y_1, x_2 \overset{2}{\delta} y_2, \dots, x_n \overset{n}{\delta} y_n),$$

In that case,  $f \in P(G)$  can be considered as an  $n$  variable function  $f(x_1, \dots, x_n)$ ,  $x_i \in G_i$ .

## Group Representations

Denote by  $K$  the number of equivalence classes of irreducible representations of  $G$  over  $P$ . Each such equivalence class contains just one unitary representation.

Denote the  $K$  unitary irreducible representations of  $G$  in some fixed order by  $\mathbf{R}_0, \mathbf{R}_1, \dots, \mathbf{R}_{K-1}$ . We denote by  $\mathbf{R}_w(x)$  the values of  $\mathbf{R}_w$  at  $x \in G$ .  $\mathbf{R}_w(x)$  stands for a non-singular  $(r_w \times r_w)$  matrix with elements  $R_w^{(i,j)}(x)$ ,  $i, j = 1, \dots, r_w$  in  $P$ .

If  $G$  is representable in the form (1), then its unitary irreducible representations can be obtained as the Kronecker product of the unitary irreducible representations of subgroups  $G_i$ ,  $i = 1, \dots, n$ . Therefore, the number  $K$  of unitary irreducible representations of  $G$  can be expressed as

$$K = \prod_{i=1}^n K_i,$$

where  $K_i$  is the number of unitary irreducible representations of the  $i$ -th subgroup  $G_i$ .

The index  $w$  of each unitary irreducible representation  $\mathbf{R}_w$  can be written as  $w = \sum_{i=1}^n b_i w_i$ , for  $w_i \in \{0, \dots, K_i - 1\}$ , and  $w \in \{0, \dots, K - 1\}$ , where

$$b_i = \begin{cases} \prod_{j=i+1}^n K_j, & i = 1, \dots, n-1, \\ 1, & i = n. \end{cases} \quad (3)$$

The functions  $\mathbf{R}_w^{(i,j)}(x)$ ,  $w = 0, 1, \dots, K - 1$ ,  $x \in G$ ,  $i, j = 1, \dots, r_w$  form an orthogonal system in  $P(G)$ . This system is used in definition of the Fourier transform on finite, not necessarily Abelian, groups [64,70]. For Abelian groups, it reduces to the group characters.

*Example 1 (Abelian group)*

Consider a function  $f$  given by the vector

$$\mathbf{F} = [0, 1, 1, 2, 2, 2, 1, 1, 0, 2, 2, 2, 1, 2, 2, 2]^T.$$

This function can be considered as a function on a group  $G = (G_{18}, \Delta)$ , of order 18 into a field  $P = C$ . This group can be represented as  $G_{18} = C_2 \times C_3 \times C_3$ , where  $C_2 = (\{0, \frac{1}{2}\}, \overset{2}{\Delta})$ ,  $\Delta$ -modulo 2 addition,  $C_3 = (\{0, 1, 2\}, \Delta)$ ,  $\Delta$ -modulo 3 addition. The group operation and group representations of  $G_{18}$  are given in the Addendum.

*Example 2 (Non-Abelian group)*

Consider a function  $f$  given by the vector

$$\mathbf{f} = [0, 6, 2, 1, 0, 0, 2, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2]^T.$$

Since  $f$  is defined in 24 points, it can be considered as a function on a group  $G_{24} = C_2 \times C_2 \times S_3$ , where  $C_2$  is the cyclic group of order 2, and  $S_3$  is the symmetric group of permutations of order three. Thus,  $C_2 = (\{0, 1\}, \oplus)$ , where  $\oplus$  is the addition modulo 2, and  $S_3 = (0, (132), (123), (12), (13), (23), \cdot)$ . If the group elements of  $S_3$  are denoted by 0, 1, 2, 3, 4, 5, respectively, then the group operation of  $S_3$  is shown in Table IV.  $S_3$  is a non-Abelian group, and thus,  $G_{24}$  is a non-Abelian group.

Since the values of  $f$  are all smaller than 11, we can assume that  $f$  takes values in a finite Galois field. In this paper, we chose the Galois field  $GF(11)$ . Group representations of  $C_2$ , and the group operation and group representations of  $S_3$ , over  $GF(11)$  are given in the Addendum.

## GIBBS DERIVATIVES

Differential operators denoted as the Gibbs derivatives were introduced first by Gibbs [23] for functions on finite dyadic groups under the name logical or finite dyadic derivative. This derivative was defined as a linear operator having the discrete Walsh functions [71] as the eigenfunctions. The concept was accepted by a number of authors and enlarged into the theory of Gibbs derivatives extending the concept of differentiation to

arbitrary, not necessarily Abelian, groups. A general characterization of Gibbs derivatives can be given as follows [55].

**DEFINITION 1** Denote by  $K(G)$  the space of functions  $f : G \rightarrow K$ , where  $G$  is a group of order  $g$  and  $K$  a field admitting the existence of a Fourier transform.

Gibbs derivative of order  $k$ ,  $D^k f$  of  $f \in K(G)$  is considered as a linear operator in  $K(G)$  such that

$$(F(D^k f))(w) = \varphi(w, k)(F(f))(w),$$

where  $F$  denotes the Fourier transform operator in  $K(G)$ , and  $\varphi$  is a function in  $K(G)$  permitting the above relation between the Fourier transform of  $f$  and of its derivative.

Uniqueness of the Gibbs derivatives is assured by the requirement that the eigenfunctions of a Gibbs derivative are the group characters (for Abelian groups) or the elements of unitary irreducible representations for finite non-Abelian groups, i.e.

$$D^k(\chi_w) = a(w, k)\chi_w,$$

for Abelian groups, and

$$D^k R_w^{(i,j)} = a(w, k)R_w^{(i,j)},$$

where  $\chi_w$  is the  $w$ -th group character of an Abelian group and  $R_w^{(i,j)}$  is the  $(i,j)$ -th element of the  $w$ -th unitary irreducible representation  $R_w$  of a non-Abelian group.

**Example 3** The finite dyadic group  $C_2^n$  of order  $2^n$  consists of the set of binary  $n$ -tuples  $(x_1, \dots, x_n)$  under the componentwise addition modulo 2. The discrete Walsh functions are the group characters of  $C_2^n$  [22]. Therefore,

$$D^k \text{wal}(w, x) = w^k \text{wal}(w, x).$$

### Properties of Gibbs Derivatives

The chief properties of Gibbs derivatives on finite groups which permits their application in engineering practice, may be expressed as follows [55].

1.  $D$  is a closed operator in  $K(G)$ .
2. The derivative of a constant

$$Df = 0 \in K(G) \quad \text{iff } f \text{ is a constant function.}$$

3. Convolution property: for each  $f_1, f_2 \in K(G)$

$$D(f_1 * f_2) = (Df_1) * f_2 = f_1 * (Df_2),$$

where  $*$  denotes the convolution in  $K(G)$ .

4. The group characters  $\chi_w$  for Abelian groups and the functions  $R_w^{(i,j)}(\cdot)$  for finite non-Abelian groups are infinitely many times Gibbs-differentiable functions.
5. The Gibbs derivatives do not obey the product rule, i.e. for any  $f_1, f_2 \in K(G)$ ,

$$D(f_1 f_2) \neq f_1 (Df_2) + (Df_1) f_2.$$

6. Shift invariance: for each  $f \in K(G)$

$$D(T_a f) = T_a(Df),$$

where  $T_a$  denotes the shift operator on  $G$  defined by  $(T_a f)(x) = f(x\Delta a)$ , where  $\Delta$  denotes the group operation in  $G$ .

7. Haar integral of derivative

$$\int_G Df = 0 \in K.$$

For a review of Gibbs derivatives up to 1989, see Ref. [5] and the bibliography given there [25]. A review of recent developments in the area is given in Ref. [55,72].

### MATRIX INTERPRETATION OF GIBBS DERIVATIVES ON FINITE ABELIAN GROUPS

Matrix notation proved very useful in formulation procedures for calculation of Gibbs derivatives. These procedures are based upon the matrix factorization permitting derivation of fast calculation algorithms resembling those in FFT theory [11,40,53].

**DEFINITION 2** The Gibbs derivative  $\mathbf{D}_G$  on a finite non-cyclic Abelian group of order  $g$  is defined as:

$$\mathbf{D}_G = g^{-1} \mathbf{X} \mathbf{G} \mathbf{X}^{-1},$$

where  $\mathbf{X}$  is the  $(g \times g)$  matrix of group characters and  $\mathbf{G} = \text{diag}(0, 1, \dots, g - 1)$ .

### Matrix Interpretation of Partial Gibbs Derivatives

For a function  $f(x) = f(x_1, \dots, x_n) \in K$ , we define the partial Gibbs derivative with respect to the variable  $x_i \in G_i$  as a restriction on  $G_i$  of the previously introduced Gibbs derivative on  $G$ .

**DEFINITION 3** Let  $G$  be representable in the form (1). The partial Gibbs derivative  $\mathbf{D}_i$  with respect to the variable  $x_i$  is defined as:

$$\mathbf{D}_i = \bigotimes_{j=1}^n \mathbf{A}_j,$$

where

$$\mathbf{A}_j = \begin{cases} \mathbf{D}_{G_j}, & j = i, \\ \mathbf{I}_j, & j \neq i, \end{cases}$$

where  $\otimes$  denotes the Kronecker product,  $\mathbf{D}_{G_j}$  is the matrix representing the Gibbs derivative on  $G_j$  and  $\mathbf{I}_j$  is  $(g_j \times g_j)$  identity matrix.

### Gibbs Derivative in Terms of Partial Gibbs Derivatives

The matrix  $\mathbf{D}_G$  representing the Gibbs derivative on a finite group  $G$  of order  $g$  can be expressed in terms of

partial Gibbs derivatives as [53]:

$$D_G = \sum_{i=1}^n a_i D_i, \quad (4)$$

where the coefficients  $a_i$  are defined by Eq. (3).

**EXAMPLE 4** Thanks to that factorization, the Gibbs derivative on  $G_{18}$  described in Example 1, can be represented in terms of the partial Gibbs derivatives on  $C_2$  and  $C_3$  as

$$D_{G_{18}} = 9D_1 + 3D_2 + D_3$$

where

$$\begin{aligned} \mathbf{D}_1 &= \mathbf{D}_{G_2} \otimes \mathbf{I}_3 \otimes \mathbf{I}_3, & \mathbf{D}_2 &= \mathbf{I}_2 \otimes \mathbf{D}_{G_3} \otimes \mathbf{I}_3, \\ \mathbf{D}_3 &= \mathbf{I}_2 \otimes \mathbf{I}_3 \otimes \mathbf{D}_{G_3}, \end{aligned}$$

where

$$\mathbf{D}_{G_2} = -\frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{D}_{G_3} = \begin{bmatrix} 1 & a & b \\ b & 1 & a \\ a & b & 1 \end{bmatrix},$$

where

$$a = \frac{1}{3}(e_1 - 1), \quad b = \frac{1}{3}(e_2 - 1),$$

and  $\mathbf{I}_i$  are the identity matrices of the order  $i$ .

### Matrix Interpretation of Gibbs Derivatives on Finite Non-Abelian Groups

Matrix interpretation of Gibbs derivatives on finite Abelian groups can be extended to the non-Abelian groups by using the generalized matrix multiplications permitting calculations with matrices whose elements are matrices [34]. These matrix operations are defined in the Addendum.

**DEFINITION 4** The Gibbs derivative  $\mathbf{D}_G$  on a finite non-cyclic Abelian group of order  $g$  is defined as:

$$\mathbf{D}_G = g^{-1} [\mathbf{R}] \odot \mathbf{G} \odot [\mathbf{R}]^{-1}, \quad \mathbf{G} = \text{diag}(0, 1, \dots, g - 1).$$

where  $\odot$  and  $\odot$  denote the generalized matrix multiplications permitting calculations whose elements are matrices.

### Matrix Interpretation of Partial Gibbs Derivatives on Finite Non-Abelian Groups

For  $f \in P(G)$ , we define the partial Gibbs derivative with respect to the variable  $x_i \in G_i$  as a restriction on  $G_i$  of the previously introduced Gibbs derivative on  $G$ .

**DEFINITION 5** Let  $G$  be representable in the form (1). The partial Gibbs derivative  $\mathbf{D}_i$  of  $f \in P(G)$  with respect to the variable  $x_i$  is defined as:

$$\mathbf{D}_i = \bigotimes_{j=1}^n \mathbf{A}_j, \quad \mathbf{A}_j = \begin{cases} \mathbf{D}_{G_j}, & j = i, \\ \mathbf{I}_j, & j \neq i, \end{cases}$$

where  $\mathbf{D}_{G_j}$  is the matrix representing the Gibbs derivative on  $G_j$  and  $\mathbf{I}_j$  is  $(g_j \times g_j)$  identity matrix for  $j < i$ , and  $(K_j \times K_j)$  identity matrix for  $j > i$ .

### Gibbs Derivative on Non-Abelian Groups in Terms of Partial Gibbs Derivatives

The matrix  $\mathbf{D}_G$  representing the Gibbs derivative on a finite group  $G$  of order  $g$  can be expressed in terms of partial Gibbs derivatives  $D_i$  as [53]:

$$D_G = \sum_{i=1}^n b_i D_i, \quad (5)$$

where the coefficients  $b_i$  are defined by Eq. (2).

**EXAMPLE 5** The Gibbs derivative for functions on the group  $G_{24}$  in Example 2 is defined by

$$\mathbf{D}_f = 6\mathbf{D}_1 + 3\mathbf{D}_2 + \mathbf{D}_3,$$

where the partial Gibbs derivatives are given by

$$\mathbf{D}_1 = \mathbf{D}_{C_2} \otimes \mathbf{I}_{(2 \times 2)} \otimes \mathbf{I}_{(6 \times 6)}, \quad \mathbf{D}_2 = \mathbf{I}_{(2 \times 2)} \otimes \mathbf{D}_{C_2} \otimes \mathbf{I}_{(6 \times 6)},$$

$$\mathbf{D}_3 = \mathbf{I}_{(2 \times 2)} \otimes \mathbf{I}_{(2 \times 2)} \otimes \mathbf{D}_{S_3},$$

with

$$\mathbf{D}_{C_2} = \begin{bmatrix} 6 & 5 \\ 5 & 6 \end{bmatrix}, \quad \mathbf{D}_{S_3} = \begin{bmatrix} 7 & 5 & 5 & 9 & 9 & 9 \\ 5 & 7 & 5 & 9 & 9 & 9 \\ 5 & 5 & 7 & 9 & 9 & 9 \\ 9 & 9 & 9 & 7 & 5 & 5 \\ 9 & 9 & 9 & 5 & 7 & 5 \\ 9 & 9 & 9 & 5 & 5 & 7 \end{bmatrix}.$$

### CALCULATION OF GIBBS DERIVATIVES

There are different approaches to efficiently calculate the Gibbs derivatives on finite groups. In this section, we discuss three of them.

### Convolution Algorithms

From Definition 1, the Gibbs derivatives can be regarded as convolution operators and, thanks to that, can be calculated through convolution algorithms [40]. These algorithms can be derived in analogy to the convolution

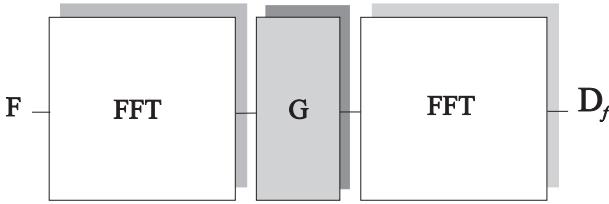


FIGURE 1 Structure of the convolution algorithms for Gibbs derivatives.

algorithms defined in terms of FFT [4,10,40,41]. For the Gibbs derivative on a given group  $G$ , the algorithm consists of one direct Fourier transform on  $G$ , component-wise multiplication by the vector  $G$  and one inverse Fourier transform. The structure of these algorithms is shown in Fig. 1.

The complexity of the convolution algorithms for Gibbs derivatives is similar to that of the convolution algorithms in Fourier analysis. Time and space complexity approximate  $O(n + 1 + n)$  and  $O(g)$ , respectively, if the in-place computation [65] is assumed.

### FFT-like Algorithms

FFT-like algorithms for calculation of Gibbs derivatives are derived through the application of the Good–Thomas factorization [26] to the matrix representing the Gibbs derivative on a given group  $G$  [53]. Unlike the algorithms for calculation of Fourier transform on groups, the steps in FFT-like algorithms for Gibbs derivatives can be performed simultaneously. That approach reduces the time complexity at the price of the space complexity. Gibbs derivative on any finite group can be calculated in two steps. However, the space complexity approximates  $O(ng + g)$ . The general structure of these algorithms is shown in Fig. 2.

*Example 6* FFT-like algorithms for the partial Gibbs derivatives on  $Z_{18}$  are shown in Fig. 3, where  $d_{i,j}$  is the  $j$ -th element in the vector representing the partial Gibbs derivative with respect to the  $i$ -th variable  $D_f$ . The vectors of partial Gibbs derivatives multiplied by 9, 3 and 1, respectively, and added componentwise determine the vector representing the values of the Gibbs derivative of  $f$ .

Similar algorithms can be defined for the Gibbs derivatives on finite non-Abelian groups [53].

### DECISION DIAGRAMS

Decision diagrams are a data structure permitting compact representations of functions on finite groups [47]. In this paper, we use Multi-terminal DDs (MTDDs) [8] to represent discrete functions on finite, not necessarily Abelian groups. MTDDs are derived by the reduction of Multi-terminal decision trees (MTDTs), which are graphical representation of the procedure of enumeration of function values for  $f$  on a group  $G$  of the form (1).

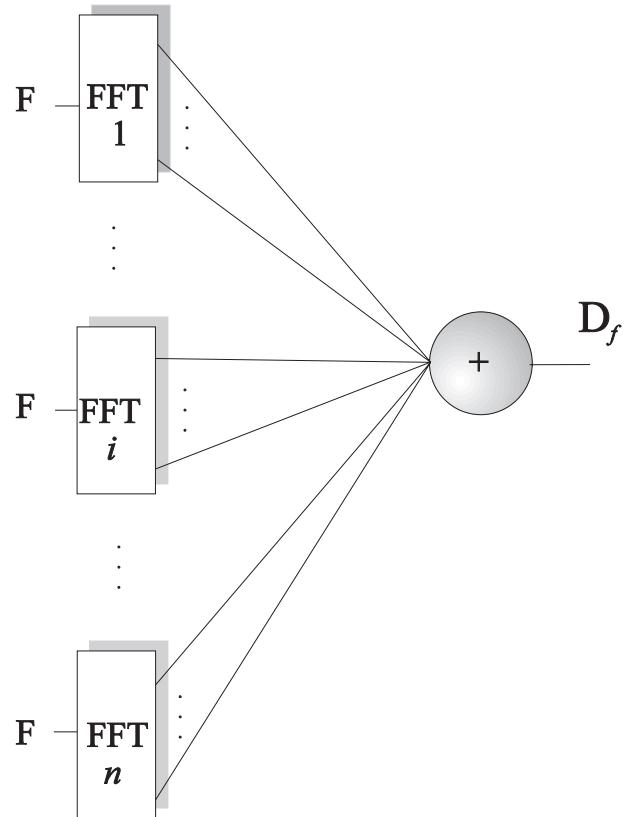


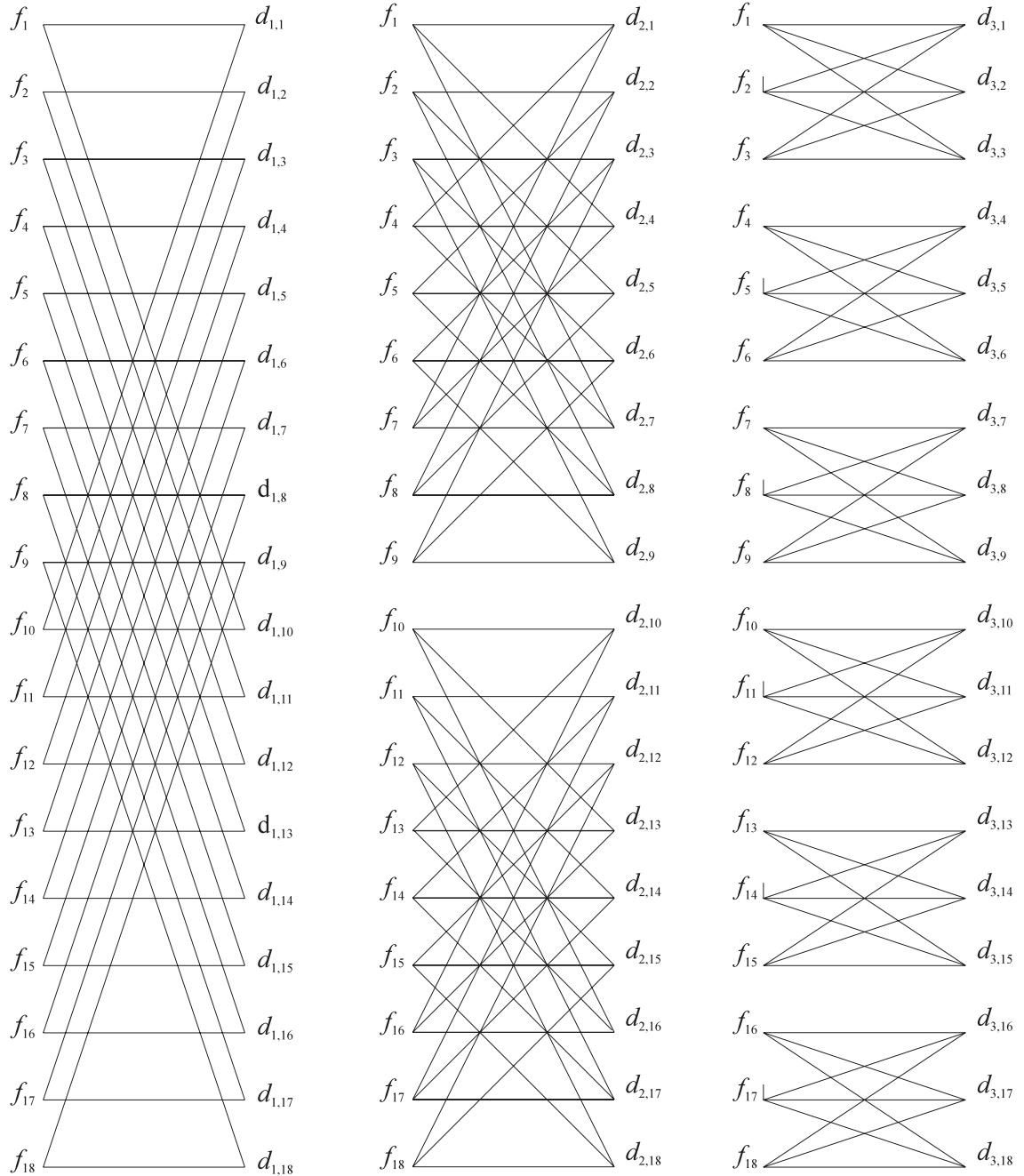
FIGURE 2 Structure of FFT-like algorithms for Gibbs derivatives.

Elements of such a group can be identified with the lexicographically ordered set of  $n$ -tuples  $(x_1, \dots, x_n)$ ,  $x_i \in G_i$ . These  $n$ -tuples are generated through the recursive assignment of values for variables in  $f$ . This recursive assignment means that for  $x_i = s_j$ ,  $s_j$  a particular value in  $G_i$ , the variable  $x_{i-1}$  takes all the values in  $G_{i-1}$ , before  $x_i$  take the value  $s_{j+1}$  in  $G_i$ . Each variable corresponds to a level in the MTDT. The  $i$ -th level consists of nodes with  $g_i$  outgoing edges. Each edge corresponds to a value for  $x_i$ , and is denoted by the corresponding literal for  $x_i$ .

The variable  $x_1$  is assigned to the root node representing  $f$ . The outgoing edges of nodes at the level corresponding to the variable  $x_n$  point to the constant nodes representing values for  $f$  at the points whose coordinates are labels at the edges in the paths from the root node to the constant nodes.

A MTDD for a given  $f$  is derived from the MTDT by sharing isomorphic subtrees and deleting the redundant information in the DT. Formally, a MTDD can be defined as follows.

**DEFINITION 6** A MTDD for representation of  $f \in P(G)$  is a root-directed acyclic graph  $D(V,E)$  with the node set  $V$  consisting of non-terminal nodes and terminal or constant nodes. A non-terminal node is labeled with a variable  $x_i$  of  $f$  and has  $g_i$  successors denoted by  $\text{sucess}_k(v) \in V$  with  $k \in G_i$ . A constant node  $v$  is labeled with an element from  $P$  and has no successors.

FIGURE 3 FFT-like algorithms for partial Gibbs derivatives on  $G_{18}$ .

In a MTDD, edges connecting nodes at non-successive levels may appear. Cross points are points where such an edge crosses levels in the MTDD. Through cross points, the impact of the deleted nodes from the MDDT is taken into account. The concept of MTDD is explained and illustrated by the following examples.

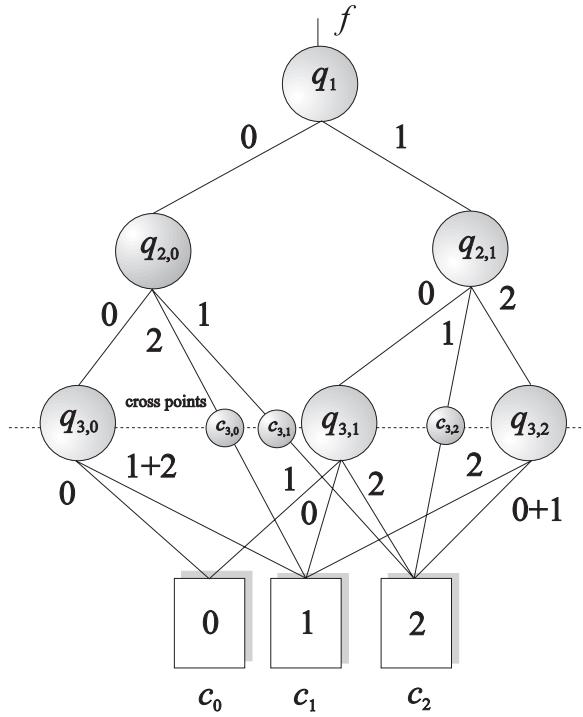
*Example 7* Consider the function  $f$  on  $G_{18}$  into the Galois field  $GF(3)$  used in Example 1. This function is represented by the decision diagram in Fig. 4.

EXAMPLE 8 Figure 5 shows MTDD for  $f$  in Example 2.

## CALCULATION OF GIBBS DERIVATIVES THROUGH DDS

After the publication of Ref. [8], DDS are efficiently used in calculation of different spectral transforms and related Kronecker product representable linear operators, see for example, Refs. [9,6,7,17,19,20,21,28,38,44,50,57,58,60, 61,62,68].

Thanks to the recursive structure of matrices describing Gibbs derivatives, which originates in their representation in terms of Fourier transform on groups, it is possible to extend DDS methods to calculation of Gibbs derivatives.

FIGURE 4 Decision diagram for function  $f$  in Example 1.

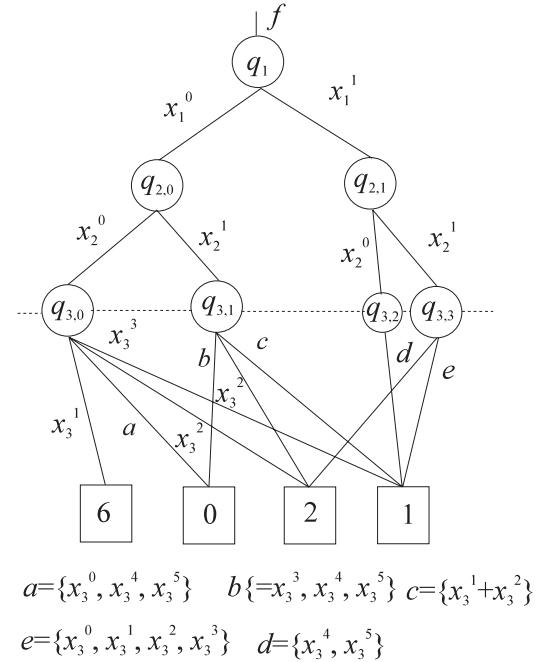
Both convolution and FFT-like algorithms for Gibbs derivatives are based upon the truth-vector representation of a given function  $f$  on  $G$ . Therefore, their complexity is determined by the order  $g$  of  $G$ . In practical applications, this limits the use of these algorithms to functions of a relatively small number of variables. Algorithms based on MTDDs permit calculation of Gibbs derivatives of functions of a considerable number of variables.

A procedure to calculate Gibbs derivatives is based on decomposition of Gibbs derivative into the linear combination of partial Gibbs derivatives in Eq. (3). It is derived as a generalization of the procedure for calculation of the Fourier transform on non-Abelian groups and as a modification of the procedure for calculation of Gibbs derivatives on finite Abelian groups through DDs [59]. The procedure consists of the following steps.

#### Procedure for calculation of Gibbs derivatives

1. Represent  $f$  by the MTDD.
2. Determine partial Gibbs derivatives.
3. Determine the Gibbs derivative as the linear combination of partial Gibbs derivatives.

The partial Gibbs derivatives are calculated through MTDD for  $f$  and represented again by MTDDs. The Gibbs derivative is determined by adding MTDDs representing the partial Gibbs derivatives.

FIGURE 5 MTDD for  $f$  in Example 2.

## Calculation of Partial Gibbs Derivatives

The procedure for calculation of partial Gibbs derivatives is similar to that for FFT. The difference is in the processing rules applied at the nodes and cross points in the MTDD for  $f$ . For the partial Gibbs derivative with respect to  $x_i$ , the nodes and cross points at the  $i$ -th level are processed by the rule determined by  $\mathbf{D}_i$ . The nodes and cross points at the other levels are processed by the rules determined by the identity matrices of the corresponding orders as determined in Definition 5.

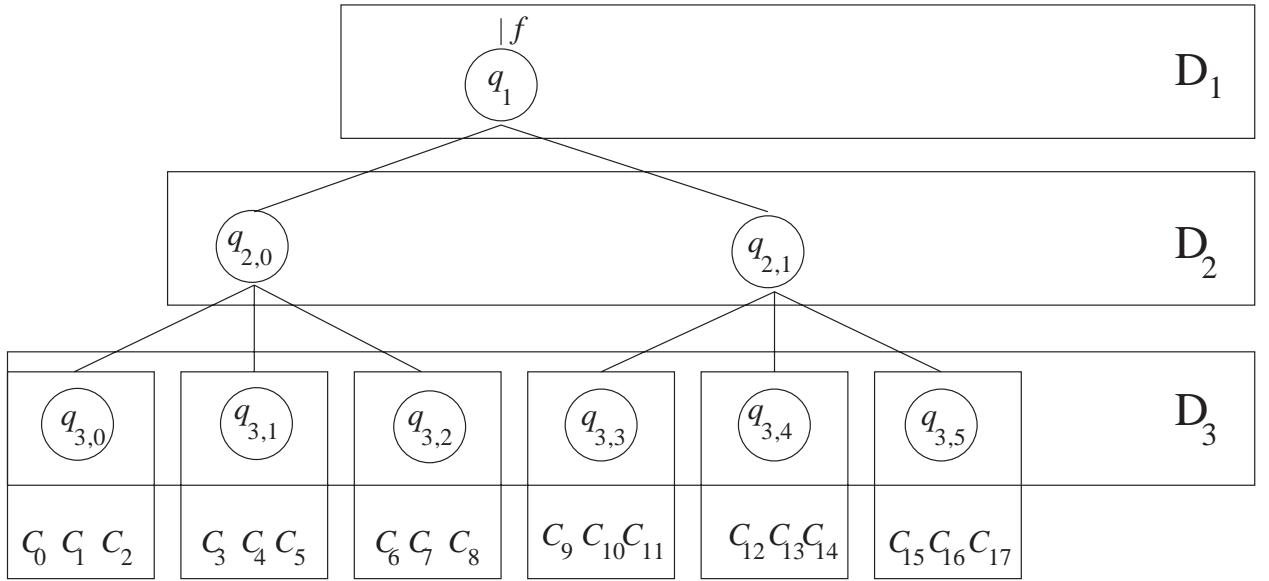
Procedure for calculation of partial Gibbs derivative  $D_i$

Given a function  $f$  on the decomposable group  $G$  of the form (1).

1. Represent  $f$  by the MTDD.
2. Descend the MTDD in a recursive way level by level starting from the nodes at the level to which  $x_n$  is assigned up to the root node.
3. For  $j = n$  to 1, process the nodes and the cross points at the  $j$ -th level by using the rule determined by  $\mathbf{D}_{G_j}$  if  $j = i$ ,  $\mathbf{I}_{(g_j \times g_j)}$  if  $j < i$ , and  $\mathbf{I}_{(K_j \times K_j)}$  if  $j > i$ . The output from the processing of the root node is the partial Gibbs derivative of  $f$  with respect to the variable  $x_i$ .

## Abelian Groups

We derive the following procedure for calculation of the partial Gibbs derivatives of the function  $f$  given by a decision diagram.

FIGURE 6 Calculation of Gibbs derivative on  $G_{18}$  for  $f$  over MTDT.

Procedure for calculation of  $D_i$

for  $i := 1$  to  $n$  do

    for  $k := n$  downto 1 do

        for  $r := 0$  to  $n_k - 1$  do

            if  $k = i$  then

$$D_i(z_k, j_r) = \diamond^{g_k-1} (\oplus_{p=0}^{g_k-1} d_{s,p}^i D_i(z_{k+1}, j_p)$$

            else

$$D_i(z_k, j_r) = \diamond^{g_k-1} D_i(z_{k+1}, j_p)$$

$$z_{k,j_r} \in \{q_{k,j_r}, c_{k,j_r}\}$$

        end.

where  $\diamond$  denotes the concatenation of vectors,  $D_i(z_k, j_r)$  is the value of the  $i$ -th partial Gibbs derivative in the  $r$ -th node or cross point at the  $k$ -th level of the DD.  $D_i(z_{n+1}, j_p) = C_p$  are the values of the constant nodes of the DD. The value of the partial Gibbs derivative  $D_i$  is obtained as  $D_i(q_1, 0)$ .

The calculation procedure is performed through the manipulation of decision diagrams, which ensures its efficiency. All operations are performed over subtrees in the MTDT for  $f$ . Figure 6 shows the subtrees used in calculation of Gibbs derivatives on  $G_{18}$ . Table I shows assignment of matrices to the levels in the MTDT for calculation of particular partial Gibbs derivatives. In practical implementations, calculations are performed

over DD for  $f$  and advantages are taken from the compactness of the decision diagrams. The method will be further explained by the following example.

In the matrix notation, the calculation procedure for  $D_1$  of  $f$  in Example 1 goes as follows.

First we process the nodes  $q_{3,0}$ ,  $q_{3,1}$ , and  $q_{3,2}$ , and the cross points by the matrix  $I_3$ . Thus, the calculations reduce to the concatenations.

1.  $D_1(q_{3,0}) = [C_0] \diamond [C_1] \diamond [C_1] = [0] \diamond [1] \diamond [1] = [0 \ 1 \ 1]^T$
2.  $D_1(q_{3,1}) = [C_1] \diamond [C_0] \diamond [C_2] = [1] \diamond [0] \diamond [2] = [1 \ 0 \ 2]^T$
3.  $D_1(q_{3,2}) = [C_2] \diamond [C_2] \diamond [C_1] = [2] \diamond [2] \diamond [1] = [2 \ 2 \ 1]^T$
4.  $D_1(c_{3,0}) = [C_1] \diamond [C_1] \diamond [C_1] = [1] \diamond [1] \diamond [1] = [1 \ 1 \ 1]^T$
5.  $D_1(c_{3,1}) = D_1(c_{3,2}) = [C_2] \diamond [C_2] \diamond [C_2] = [20] \diamond [2] \diamond [2] = [2 \ 2 \ 2]^T$

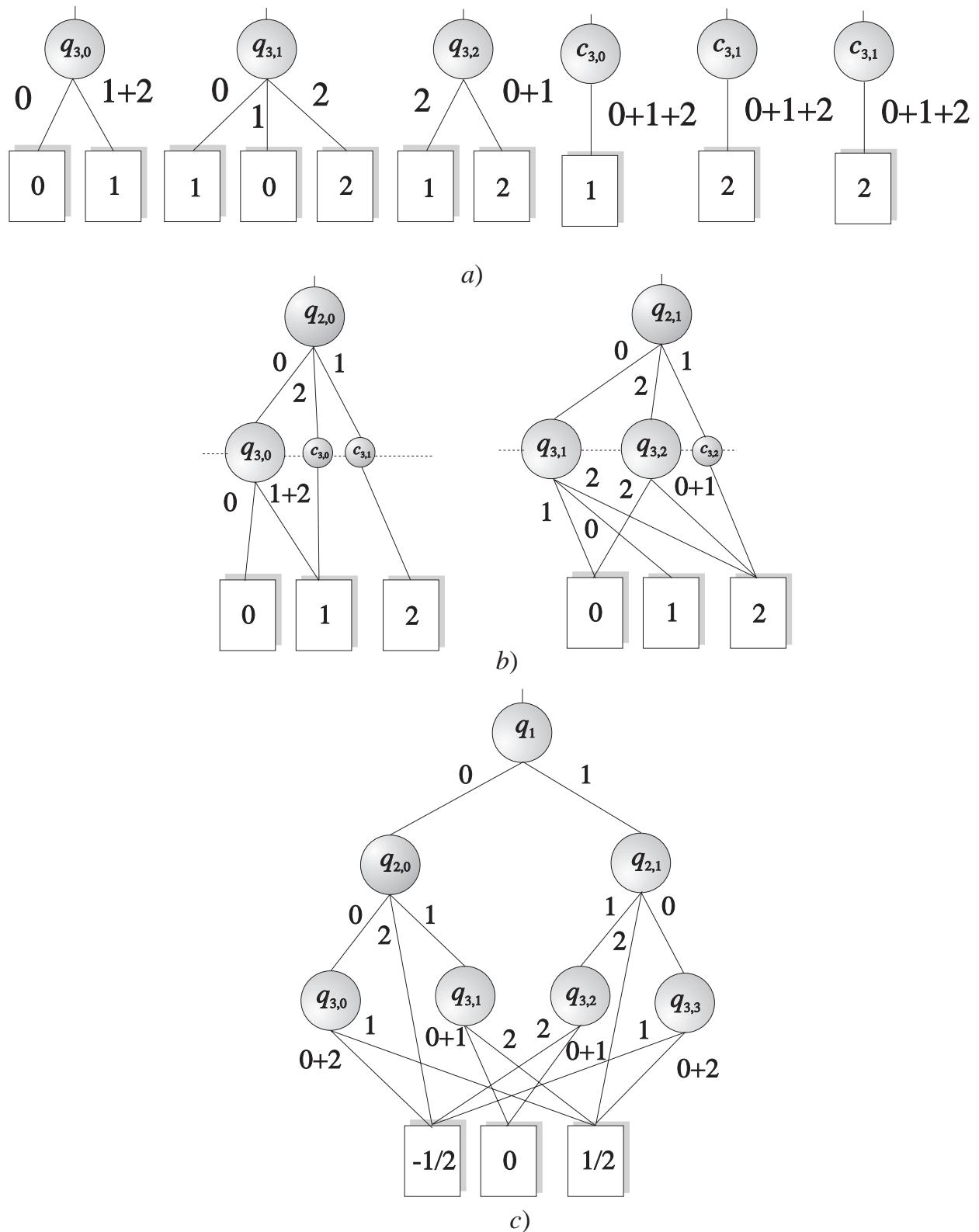
The calculation procedure is shown on the decision diagrams in Fig. 7a

6. Then, the nodes at level 2 in the decision diagram in Fig. 4 are processed by the matrix  $I_3$  as is described by the following matrix relations.

$$D_1(q_{2,0}) = D_1(q_{3,0}) \diamond D_1(c_{3,1}) \diamond D_1(c_{3,0})$$

$$= \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \diamond \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} \diamond \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= [0 \ 1 \ 1 \ 2 \ 2 \ 2 \ 1 \ 1 \ 1]^T,$$

FIGURE 7 MTDDs in calculation of  $D_1$  for  $f$  in Example 1.

7.

$$D_1(q_{2,1}) = D_1(q_{3,1}) \diamond D_1(q_{3,2}) \diamond D_1(c_{3,2})$$

$$= \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \diamond \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} \diamond \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

$$= [1 \ 0 \ 2 \ 2 \ 2 \ 1 \ 2 \ 2 \ 2]^T$$

Calculation procedure is shown on the decision diagrams in Fig. 7b.

8. The value of the partial Gibbs derivative  $D_1$  of  $f$  is determined by processing the root node in the decision diagram in Fig. 4 by the matrix  $\mathbf{Q}_1$  as is described by the matrix relation

$$D_1(q_1) = -\frac{1}{2}[-D_1(q_{2,0}) + D_1(q_{2,1})] \diamond -\frac{1}{2}[D_1(q_{2,0})$$

$$- D_1(q_{2,1})]$$

$$= -\frac{1}{2} \left[ \begin{array}{cc|cc} & \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 2 \\ 2 & 2 \end{array} & \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 2 \\ 2 & 2 \end{bmatrix} & \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 2 \\ 2 & 2 \end{bmatrix} \\ \begin{bmatrix} 2 & 2 \\ 2 & 1 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} & \diamond & \begin{bmatrix} 2 & 2 \\ 2 & 1 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} & \end{array} \right]$$

$$= -\frac{1}{2} \left[ \begin{array}{cc|cc} & \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 0 & 0 \\ 0 & 0 \\ -1 & 1 \\ 1 & -1 \\ 1 & -1 \\ 1 & -1 \end{bmatrix} & \begin{bmatrix} q_{3,0} \\ q_{3,1} \end{bmatrix} & \begin{bmatrix} q_{3,0} \\ q_{3,1} \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} & \diamond & \begin{bmatrix} q_{3,2} \\ q_{3,3} \end{bmatrix} & \end{array} \right]$$

Partial Gibbs derivative  $D_1 f$  is represented by the decision diagram in Fig. 7c.

The calculation of the partial Gibbs derivatives with respect to  $x_2$  and  $x_3$  is performed in the similar way. The determination of the Gibbs derivative is done by using Eq. (4) which is performed over the decision trees representing the partial Gibbs derivatives in the same way as is suggested in Ref. [8].

### Non-Abelian Groups

The procedure for calculation of the Gibbs derivatives on non-Abelian groups through MTDDs is explained and illustrated by the following example.

**EXAMPLE 9** For  $f$  in Example 2 calculation of the Gibbs derivative on  $G_{24}$  goes as follows.

To calculate the partial Gibbs derivative with respect to  $x_3$ , we perform calculations determined by definition of  $\mathbf{D}_{S_3}$  at the nodes  $q_{3,0}$ ,  $q_{3,1}$ ,  $q_{3,3}$  and the cross point  $q_{3,2}$ . In the nodes at the levels corresponding to  $x_2$  and  $x_1$ , we perform the identical mapping defined by  $\mathbf{I}_{(2 \times 2)}$ .

To calculate the partial Gibbs derivative with respect to  $x_2$ , we perform the identical mapping determined by  $\mathbf{I}_{(6 \times 6)}$  at the nodes and cross point at the level for  $x_3$ , the calculations determined by  $\mathbf{D}_{C_2}$  at the nodes for  $x_2$  and the identical mapping determined by  $\mathbf{I}_{(2 \times 2)}$  at the root node.

Similarly, to calculate partial Gibbs derivative with respect to  $x_1$ , we perform  $\mathbf{D}_{C_2}$  at the root node, while at the other nodes and the cross points the identical mappings  $\mathbf{I}_{(2 \times 2)}$  are performed.

1. Partial Gibbs derivative with respect to  $x_3$ :

$$q_{3,0} = \mathbf{D}_{S_3}[0, 6, 2, 1, 0, 0]^T = [5, 6, 9, 2, 0, 0]^T,$$

$$q_{3,1} = \mathbf{D}_{S_3}[2, 1, 1, 0, 0, 0]^T = [2, 0, 0, 3, 3, 3]^T,$$

$$q_{3,2} = \mathbf{D}_{S_3}[1, 1, 1, 1, 1, 1]^T = [0, 0, 0, 0, 0, 0]^T,$$

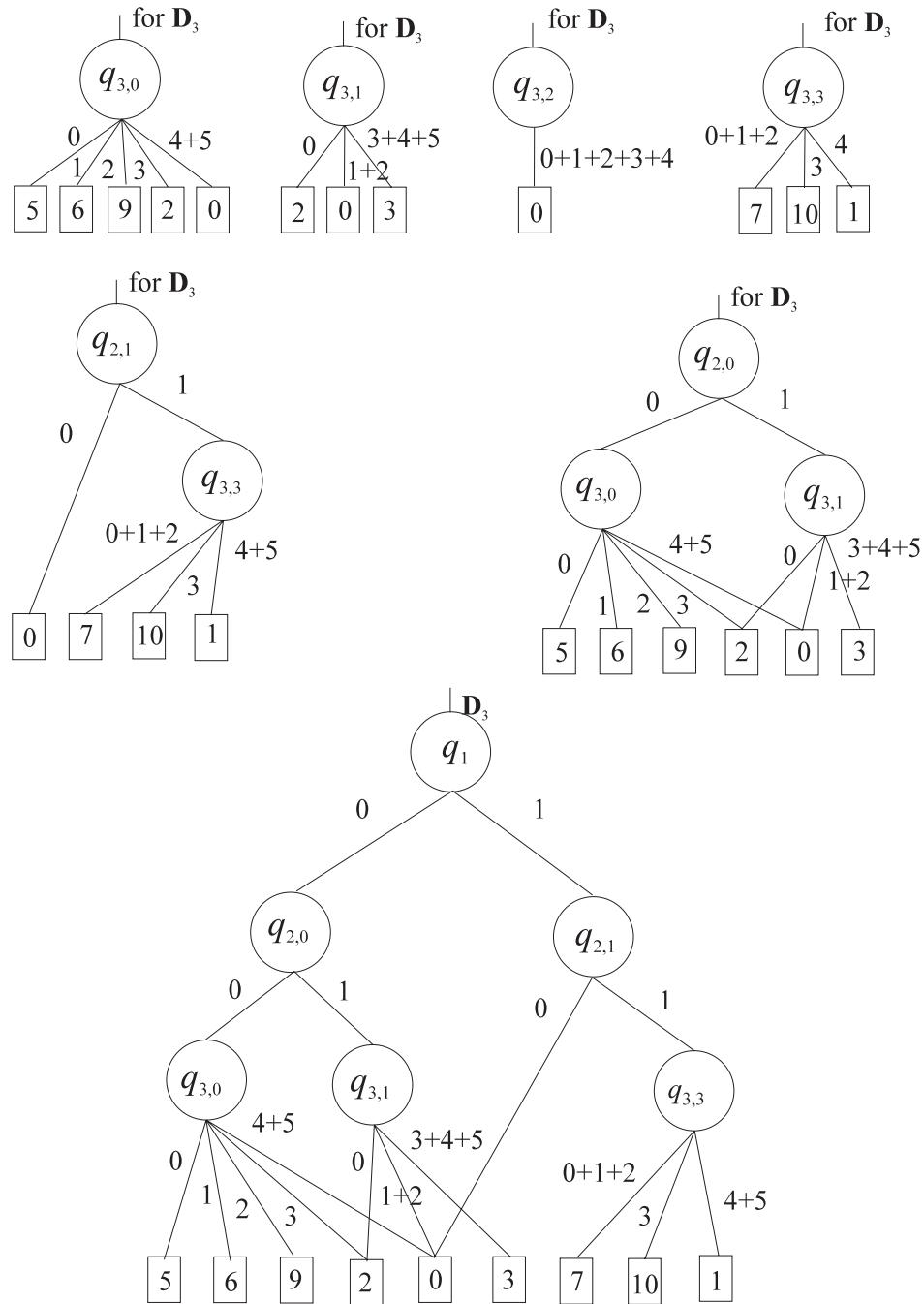
$$q_{3,3} = \mathbf{D}_{S_3}[1, 1, 1, 1, 2, 2]^T = [7, 7, 7, 10, 1, 1]^T.$$

$$q_{2,0} = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{3,0} \\ q_{3,1} \end{bmatrix} = [5, 6, 9, 2, 0, 0, 2, 0, 0, 3, 3, 3]^T,$$

$$q_{2,1} = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{3,2} \\ q_{3,3} \end{bmatrix} = [0, 0, 0, 0, 0, 0, 7, 7, 7, 10, 3, 3]^T.$$

$$\mathbf{D}_3 = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{2,0} \\ q_{2,1} \end{bmatrix}$$

$$= [5, 6, 9, 2, 0, 0, 2, 0, 0, 3, 3, 3, 0, 0, 0, 0, 0, 0, 7, 7, 7, 10, 1, 1]^T.$$

FIGURE 8 Calculation of the partial Gibbs derivative with respect to  $x_3$  for  $f$  in Example 2.

2. Partial Gibbs derivative with respect to  $x_2$ :

$$q_{3,0} = \mathbf{I}_{(6 \times 6)}[0, 6, 2, 1, 0, 0]^T = [0, 6, 2, 1, 0, 0]^T,$$

$$q_{3,1} = \mathbf{I}_{(6 \times 6)}[2, 1, 1, 0, 0, 0]^T = [2, 1, 1, 0, 0, 0]^T,$$

$$q_{3,2} = \mathbf{I}_{(6 \times 6)}[1, 1, 1, 1, 1, 1]^T = [1, 1, 1, 1, 1, 1]^T,$$

$$q_{3,3} = \mathbf{I}_{(6 \times 6)}[1, 1, 1, 1, 2, 2]^T = [1, 1, 1, 1, 2, 2]^T.$$

$$q_{2,0} = \mathbf{W}(1) \begin{bmatrix} q_{3,0} \\ q_{3,1} \end{bmatrix} = \begin{bmatrix} 6q_{3,0} + 5q_{3,1} \\ 5q_{3,0} + 6q_{3,1} \end{bmatrix}$$

$$= [10, 8, 6, 6, 0, 0, 1, 3, 5, 5, 0, 0]^T,$$

$$q_{2,1} = \mathbf{W}(1) \begin{bmatrix} q_{3,2} \\ q_{3,3} \end{bmatrix} = \begin{bmatrix} 6q_{3,2} + 5q_{3,3} \\ 5q_{3,2} + 6q_{3,3} \end{bmatrix}$$

$$= [0, 0, 0, 0, 5, 5, 0, 0, 0, 0, 6, 6]^T.$$

$$\mathbf{D}_2 = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{2,0} \\ q_{2,1} \end{bmatrix}$$

$$= [10, 8, 6, 6, 0, 0, 1, 3, 5, 5, 0, 0, 0, 0, 0, 0, 5, 5, 0, 0, 0, 0, 6, 6]^T.$$

3. Partial Gibbs derivative with respect to  $x_1$ :

$q_{3,0}$ ,  $q_{3,1}$ ,  $q_{3,2}$ , and  $q_{3,3}$  are as in calculation of  $D_2$ .

$$q_{2,0} = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{3,0} \\ q_{3,1} \end{bmatrix} = [0, 6, 2, 1, 0, 0, 2, 1, 1, 0, 0, 0]^T,$$

$$q_{2,1} = \mathbf{I}_{(2 \times 2)} \begin{bmatrix} q_{3,2} \\ q_{3,3} \end{bmatrix}$$

$$= [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2]^T.$$

$$\mathbf{D}_1 = \mathbf{W}(1) \begin{bmatrix} q_{2,0} \\ q_{2,1} \end{bmatrix} = \begin{bmatrix} 6q_{2,0} + 5q_{2,1} \\ 5q_{2,0} + 6q_{2,1} \end{bmatrix}$$

$$= [5, 8, 6, 0, 5, 5, 6, 0, 0, 5, 10, 10, 6, 3, 5, 0, 6, 6, 5, 0, 0, 6, 1, 1]^T.$$

Therefore,

$$\mathbf{D}_{24} = 6\mathbf{D}_1 + 3\mathbf{D}_2 + \mathbf{D}_3$$

$$= [10, 1, 8, 9, 8, 8, 9, 4, 4, 8, 8, 3, 7, 8, 0, 7, 7, 4, 7, 7, 2, 3, 3]^T.$$

Each step of calculation can be represented through MTDDs. For example, Fig. 8 shows calculation of the partial Gibbs derivative with respect to  $x_3$  for  $f$ .

## CLOSING REMARKS

In this paper, we discussed different methods for calculation of Gibbs derivatives on finite groups. By

combining powerful FFT-like algorithms together with new data structures for discrete functions, we were able to develop algorithms for efficient calculation of Gibbs derivatives.

Gibbs derivatives share useful properties of logic differential operators and spectral methods. Especially, compact matrix representations for functions on non-Abelian groups and their efficient representations through decision diagrams, offer a way for study of properties of switching functions through their Gibbs derivatives and relations in terms of Gibbs derivatives. In that way, the presented methods raise applicability of Gibbs derivatives in engineering practice.

It is believed that these algorithms are a suitable basis for further research in applications of Gibbs derivatives in the areas where the logic derivatives and spectral techniques are already used, as for example, logic design, verification and testing of logic networks.

## Acknowledgements

The authors thank Prof. Claudio Moraga for many comments and suggestions that considerably improved the presentation in this paper.

This paper was prepared during the authors' stay at International Center for Signal Processing of Tampere University for Technology (TICSP). The support and facilities provided by TICSP are appreciated and gladly acknowledged.

## References

- [1] Aizenberg, N.N. and Trofimljuk, O.T. (1981) "Conjunctive transforms for discrete signals and their applications of tests and the detection of monotone functions", *Kibernetika* **1K**, 128–139, in Russian.
- [2] Agaian, S., Astola, J.T. and Egiazarian, K. (1995) *Binary Polynomial Transforms and Nonlinear Digital Filters* (Marcel Dekker, New York).
- [3] Bryant, R.E. (1986) "Graph-based algorithms for Boolean functions manipulation", *IEEE Trans. Comput.* **C-35**(8), 667–691.
- [4] Burrus, C.S. and Eschenbacher, R.W. (1981) "An in-place, in-order prime factor FFT algorithm", *IEEE Trans. Acoust., Speech, Signal Process.* **29**, 806–816.
- [5] Stanković, R.S., Stojić, M.R., Stanković, M.S., eds, (1990) *Theory and Applications of Gibbs Derivatives* (Matematički institut, Beograd).
- [6] Chang, C.H. and Falkowski, B.J. (1997) "Efficient symbolic computation of generalized spectra", *IEE Electron. Lett.*, UK **33**(22), 1837–1838.
- [7] Chang, C.H. and Falkowski, B.J. (1999) "Haar spectra based entropy approach to quasi-minimization of FBDDs", *IEE Proc. Comput., Digit. Techn.*, UK **146**(1), 41–49.
- [8] Clarke, E.M., McMillan, K.L., Zhao, X. and Fujita, M. (1993) "Spectral transforms for extremely large Boolean functions", In: Kehschull, U., Schubert, E. and Rosenstiel, W., eds, *Proc. IFIP WG 10.5 Workshop on Applications of the Reed–Muller Expansion in Circuit Design, 16–17.9.1993, Hamburg, Germany*, pp 86–90.
- [9] Clarke, E.M., Fujita, M. and Zhao, X. (1996) "Multi-terminal decision diagrams and hybrid decision diagrams", In: Sasao, T. and Fujita, M., eds, *Representations of Discrete Functions* (Kluwer Academic Publishers, Dordrecht), pp 93–108.
- [10] Cooley, J.W. and Tukey, J.W. (1965) "An algorithm for the machine calculation of complex Fourier series", *Math. Comput.* **19**, 297–301.

- [11] Creutzburg, R. and Tasche, M. (1985) "F-Transformation und Faltung in kommutativen Ringen", *Elektron. Informationsverarb. Kybern.* **EIK-21**(3), 129–149.
- [12] De Michelli, G. (1994) *Synthesis and Optimization of Digital Circuits* (McGraw-Hill, New York).
- [13] Edwards, C.R. (1977) "The design of easily tested circuits using mapping and symmetry methods", *Radio Electron. Eng.* **47**, 321–342.
- [14] Edwards, C.R. (1978) "The generalized dyadic differentiator and its application to 2-valued functions defined on an  $n$ -space", *Proc. IEE, Comput. Digit. Techn.* **1**(4), 137–142.
- [15] Egiazarian, K., Gevorkian, D. and Astola, J.T. (1997) "Time-varying filter banks and multiresolution transforms based on generalized Fibonacci topology", *Proc. 5th IEEE Int. Workshop on Intelligent Signal Proc. and Communication Systems, Kuala Lumpur, Malaysia, 11–13 Nov.*, S1651–S1654.
- [16] Egiazarian, K. and Astola, J.T. (1997) "On generalized Fibonacci cubes and unitary transforms", *Appl. Algebra Eng., Commun. Comput. AAECC* **8**, 371–377.
- [17] Falkowski, B.J. and Chang, C.H. (1997) "Forward and inverse transformations between Haar spectra and ordered binary decision diagrams of Boolean functions", *IEEE Trans. Comput.* **46**(11), 1272–1279.
- [18] Falkowski, B.J. and Chang, C.H. (1997) "Properties and methods of calculating generalized arithmetic and adding transforms", *IEE Proc. Circuits Devices Syst.* **144**(5), 249–258.
- [19] Falkowski, B.J. and Chang, C.H. (1998) "Calculation of gray code ordered Walsh spectra through algebraic decision diagrams", *IEE Electron. Lett.* **UK 34**(9), 848–850.
- [20] Falkowski, B.J. and Chang, C.H. (1998) "Mutual conversions between generalized arithmetic expansions and free binary decision diagrams", *IEE Proc. Circuits, Devices Syst., UK* **145**(4), 219–228.
- [21] Falkowski, B.J. and Rahardja, S. (1996) "Complex decision diagrams", *IEE Electron. Lett.* **UK 32**(4), 290–291.
- [22] Fine, N.J. (1949) "On the Walsh functions", *Trans. Am. Math. Soc.* **3**, 372–414.
- [23] Gibbs, J.E. (1967) "Walsh spectrometry, a form of spectral analysis well suited to binary digital computation," NPL DES Rept., Teddington, Middlesex, England, January.
- [24] Gibbs, J.E. and Simpson, J. (1974) "Differentiation on finite Abelian groups," *National Physical Lab.*, Teddington, England, DES Rept. No. 14.
- [25] Gibbs, J.E. and Stanković, R.S. (1990) "Why IWGD-89? A look at the bibliography of Gibbs derivative", In: Butzer, P.L. and Stanković, R.S., eds, *Theory and Applications of Gibbs Derivatives* (Matematički institut, Beograd), pp xi–xxiv.
- [26] Good, I.J. (1958) "The interaction algorithm and practical Fourier analysis", *J. Roy. Statist. Soc., Ser. B* **20**, 361–372, Addendum, Vol. 22, 1960, 372–375.
- [27] Guima, T.A. and Tapia, M.A. (1987) "Differential calculus for fault detections in multi-valued logic networks", *Proc. ISMVL-87*, 99–108.
- [28] Hansen, J.P. and Sekine, M. (1997) "Decision diagrams based techniques for the Haar wavelet transform", *Proc. IEEE Int. Conf. on Information, Communications and Signal Processing (1st ICICS), Singapore* **1**, 59–63.
- [29] Hsu, W.-J. (1993) "Fibonacci cubes—a new interconnection topology", *IEEE Trans. Parallel Distrib. Syst.* **4**, 3–12.
- [30] Hurst, S.L. (1978) *The Logical Processing of Digital Signals* (Crane Russak/Edvard Arnold, Basel/Bristol).
- [31] Hurst, S.L., Miller, D.M. and Muzio, J.C. (1985) *Spectral Techniques in Digital Logic* (Academic Press, New York).
- [32] Jain, J. "Arithmetic transform of Boolean functions," in [47], 133–161.
- [33] Karpovsky, M.G. (1976) *Finite Orthogonal Series in the Design of Digital Devices* (Wiley/JUP, New York/Jerusalem).
- [34] Karpovsky, M.G. and Trachtenberg, E.A. (1977) "Some optimization problems for convolution systems over finite groups", *Inform. Control.* **34**(3), 227–247.
- [35] Komamiya, Y. (1951) "Theory of relay networks for the transformation between the decimal and binary system", *Bull. E.T.L.* **15**(8), 188–197.
- [36] Kukharev, G.A., Shmerko, V.P. and Yanushkevich, S.N. (1991) *Technique of Binary Data Parallel Processing for VLSI* (Vyshey-shaja shcola, Minsk, Belarus).
- [37] Malyugin, V.D. (1997) *Paralleled Calculations by Means Arithmetic Polynomials* (Physical and Mathematical Publishing Company, Russian Academy of Sciences, Moscow), in Russian.
- [38] Miller, D.M. (1994) "Spectral transformation of multiple-valued decision diagrams", *Proc. 24th Int. Symp. on Multiple-valued Logic, Boston, Massachusetts, USA, May 22–25*, 89–96.
- [39] Miller, D.M. and Drechsler, R. (1998) "Implementing a multiple-valued decision diagram package", *Proc. 28th Int. Symp. on Multiple-Valued Logic, Fukuoka, Japan, May*, 52–57.
- [40] Nussbaumer, H.J. (1981) *Fast Fourier Transform and Convolution Algorithms* (Springer, Berlin).
- [41] Pichler, F. (1980) In: Lainiotis, D.G. and Tzannes, N.C., eds, *Fast Linear Methods For Image Filtering* (Reidel, Den Haag), pp 3–11.
- [42] Pichler, F. (1990) "Experiments with 1-D and 2-D signals using Gibbs derivatives", In: Butzer, P.L. and Stanković, R.S., eds, *Theory and Applications of Gibbs Derivatives* (Matematički institut, Beograd), pp 181–196.
- [43] Pichler, F. (1992) In: Trappl, R. and Horn, W., eds, *Realization of Prigogine's  $\Lambda$ -transform by dyadic convolution* (Austrian Society for Cybernetics Studies, ISBN 3 85206 127X).
- [44] Rahardja, S. and Falkowski, B.J. (1997) "Classifications and graph based representations of switching functions using a novel spectral technique", *Int. J. Electron., UK* **83**(6), 731–742.
- [45] Sasao, T. "Representations of logic functions by using EXOR operators," in [47], 29–54.
- [46] Sasao, T. (1999) *Switching Theory for Logic Synthesis* (Kluwer Academic Publishers, Dordrecht).
- [47] Sasao, T. and Fujita, M., eds (1996) *Representations of Discrete Functions* (Kluwer Academic Publishers, Dordrecht).
- [48] *Synthesis of electronic calculation and control networks*, translation from English under supervision by B.I. Shestakov, Izv. Inostranoj Literaturi, Moscow, 1954.
- [49] Shmerko, V.P., Yanushkevich, S., Lavashenko, V. and Bondar, I. (1996) "Technique of computing logic derivatives for MVL-functions", *Proc. 26th Int. Symp. on Multiple-Valued Logic, Santiago de Compostela, Spain, May 24–31*, 267–272.
- [50] Srinivasan, A., Kam, T., Malik, Sh. and Brayant, R.K. (1990) "Algorithms for discrete function manipulation", *Proc. Inf. Conf. CAD*, pp 92–95.
- [51] Stanković, M. (1990) "Fast algorithms for calculation of Gibbs derivatives on finite Abelian groups", In: Butzer, P.L. and Stanković, R.S., eds, *Theory and Applications of Gibbs Derivatives* (Matematički institut, Beograd).
- [52] Stanković, R.S. (1990) "Matrix-interpretation of fast Fourier transforms on finite non-Abelian groups", *Proc. Int. Conf. on Signal Processing, Beijing90, October 22–26*, 1187–1190.
- [53] Stanković, R.S. (1991) "Fast algorithms for calculation of Gibbs derivatives on finite groups", *Approx. Theory Appl.* **7**(2), 1–19.
- [54] Stanković, R.S. (1992) "Some remarks on Fourier transforms and differential operators for digital functions", *Proc. 22nd Int. Symp. on Multiple-Valued Logic*, 365–370.
- [55] Stanković, R.S. (1994) "Gibbs derivatives", *Numer. Funct. Anal. Optimiz.* **15**(1–2), 169–181.
- [56] Stanković, R.S. and Drechsler, R. (1997) "Circuit design from Kronecker Galois field decision diagrams for multiple-valued logic functions", *Proc. IEEE Int. Symp. on Multiple-Valued Logic, Antigonish, Nova Scotia, Canada, May*, 275–280.
- [57] Stanković, M., Janković, D. and Stanković, R.S. (1996) "Efficient algorithm for Haar spectrum calculation", *Sci. Rev.* **21–22**, 171–182.
- [58] Stanković, R.S. and Moraga, C. (1999) "Calculation of Fourier transform on finite non-Abelian groups through decision diagrams", *Proc. Second International Workshop on Spectral Transforms and Filter Banks, Brandenburg, Germany, (March 5–7)*.
- [59] Stanković, R.S. and Stanković, M. (1998) "Calculation of the Gibbs derivatives on finite Abelian groups through the decision diagrams", *Approx. Theory Appl.* **14**(4), 12–25.
- [60] Stanković, R.S., Stanković, M. and Janković, D. (1998) *Spectral Transforms in Switching Theory, Definitions and Calculations* (Nauka, Belgrade).
- [61] Stanković, R.S., Stanković, M., Moraga, C. and Sasao, T. (1994) "Calculation of Vilenkin–Chrestenson transform coefficients of multiple-valued functions through multiple-place decision diagrams", *Proc. 5th Int. Workshop on Spectral Techniques, March 15–17*, 107–116.
- [62] Stanković, R.S., Stanković, M., Moraga, C. and Sasao, T. (1994). "Calculation of Reed–Muller–Fourier coefficients of multiple-valued functions through multiple-place decision diagrams," *Proc. 24th Int. Symp. on Multiple-Valued Logic, Boston, Massachusetts, USA, May*.

- [63] Stanković, R.S., Stojić, M.R. and Bogdanović, S.M. (1989) *Fourier Representation of Signals* (Naučna knjiga, Beograd), in Serbian.
- [64] Stanković, R.S., Stojić, M.R., Stanković, M.S., eds, (1996) “Recent Developments in Abstract Harmonic Analysis with Applications in Signal Processing”, (Nauka/Elektronski fakultet, Belgrade/Niš).
- [65] Stojić, M.R., Stanković, M.S. and Stanković, R.S. (1993) *Discrete Transforms in Application*, 2nd Ed. (Nauka, Beograd), in Serbian.
- [66] Thayse, A. (1981) *Boolean Calculus of Differences* (Springer, Berlin).
- [67] Thomas L.H. (1963). “Using a computer to solve problems in physics,” *Application of Digital Computers*, Boston, Mass., Ginn.
- [68] Thornton, M.A. (1997) “Modified Haar transform calculation using digital circuit output probabilities”, *Proc. IEEE Int. Conf. on Information, Communications and Signal Processing (1st ICICS)*, Singapore 1, 52–58.
- [69] Trachtenberg, E.A. (1988) “SVD of Frobenius matrices for approximate and multiobjective signal processing tasks”, In: Deprettere, E.F., eds, *SVD and Signal Processing* (Elsevier/North-Holland, Amsterdam/New York), pp 331–345.
- [70] Trachtenberg, E.A. “Applications of Fourier Analysis on Groups in Engineering Practices,” in: [64], 331–403.
- [71] Walsh, J.L. (1923) “A closed set of orthogonal functions”, *Am. J. Math.* **45**, 5–24.
- [72] Su Weiyi, (1991). “Gibbs derivatives and their applications,” Rept. of the Institute of Mathematics, Nanjing University, 91-7, Nanjing, P.R. China, 1–20.
- [73] Whitney, M. and Muzio, J.C. (1988) “Decisive differences and partial differences for stuck-at-fault detection in MVL circuits”, *Proc. ISMVL-88*, 321–328.
- [74] Yanushkevich, S. (1994) “Spectral and differential methods to synthesize polynomial forms of MVL-functions on systolic arrays”, *Proc. 5th Int. Workshop on Spectral Techniques, Beijing, China, March 15–17*, 78–93.
- [75] Yanushkevich, S.N. (1997) “Matrix method for solving multivalued logic differential equations”, *IEE Proc. Comput., Digit. Tech.* **144**(5), 667–772.

## ADDENDUM

**EXAMPLE 10 (Abelian group)**  $G = (G_{18}, \Delta)$ , of order 18 and assume  $K = C$ . For convenience, the elements of this group will be identified with first 18 non-negative integers. Thus,

$$\begin{aligned} G &= Z_{18} \\ &= (\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17\}, \Delta). \end{aligned}$$

The group operation  $\Delta$  of  $G_{18}$  is given in Table II. Group representations over  $C$  are the Vilenkin–Chrestenson functions [63,65], shown in Table III.

### EXAMPLE 11 (Non-Abelian group)

Group representations of  $C_2$  over  $GF(11)$  are given by the columns of the matrix

$$\mathbf{W}(1) = \begin{bmatrix} 1 & 1 \\ 1 & 10 \end{bmatrix}.$$

This matrix is self-inverse up to a multiplicative constant, and therefore, the Fourier transform on  $C_2$  is defined by the transform matrix

$$\mathbf{W}(1) = 6 \begin{bmatrix} 1 & 1 \\ 1 & 10 \end{bmatrix}.$$

The group operation for  $S_3$  is given by Table IV.

In the matrix notation, the group representations of  $S_3$  over  $GF(11)$  are given by the columns of the matrix

$$\begin{aligned} [\mathbf{S}_3](1) &= \begin{bmatrix} 1 & 1 & \mathbf{I} \\ 1 & 1 & \mathbf{A} \\ 1 & 1 & \mathbf{B} \\ 1 & 10 & \mathbf{C} \\ 1 & 10 & \mathbf{D} \\ 1 & 1 & \mathbf{E} \end{bmatrix}, \\ \mathbf{I} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 5 & 8 \\ 3 & 5 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 5 & 3 \\ 8 & 5 \end{bmatrix}, \\ \mathbf{C} &= \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 5 & 8 \\ 8 & 6 \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} 5 & 3 \\ 3 & 6 \end{bmatrix}. \end{aligned}$$

TABLE II Group operation of  $G_{18}$

$\Delta$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	0	4	5	3	7	8	6	10	11	9	13	14	12	16	17	15
2	2	0	1	5	3	4	8	6	7	11	9	10	14	12	13	17	15	16
3	3	4	5	6	7	8	0	1	2	12	13	14	15	16	17	9	10	11
4	4	5	3	7	8	6	1	2	0	13	14	12	16	17	15	10	11	9
5	5	3	4	8	6	7	2	0	1	14	12	13	17	15	16	11	9	10
6	6	7	8	0	1	2	3	4	5	15	16	17	9	10	11	12	13	14
7	7	8	6	1	2	0	4	5	3	16	17	15	10	11	9	13	14	12
8	8	6	7	2	0	1	5	3	4	17	15	16	11	9	10	14	12	13
9	9	10	11	12	13	14	15	16	17	0	1	2	3	4	5	6	7	8
10	10	11	9	13	14	12	16	17	15	1	2	0	4	5	3	7	8	6
11	11	9	10	14	12	13	17	15	16	2	0	1	5	3	4	8	6	7
12	12	13	14	15	16	17	9	10	11	3	4	5	6	7	8	0	1	2
13	13	14	12	16	17	15	10	11	9	4	5	3	7	8	6	1	2	0
14	14	12	13	17	15	16	11	9	10	5	3	4	8	6	7	2	0	1
15	15	16	17	9	10	11	12	13	14	6	7	8	0	1	2	3	4	5
16	16	17	15	10	11	9	13	14	12	7	8	6	1	2	0	4	5	3
17	17	15	16	11	9	10	14	12	13	8	6	7	2	0	1	5	3	4

TABLE III Vilenkin–Chrestenson functions  $\{\chi_w(x)\}, w, x \in G_{18}$ 

X =	$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & e_1 & e_2 & 1 & e_1 & e_2 \\ 1 & e_2 & e_1 & 1 & e_2 & e_1 \\ 1 & 1 & 1 & e_1 & e_1 & e_1 & e_2 & e_2 & e_2 & 1 & 1 & 1 & e_1 & e_1 & e_1 & e_2 & e_2 & e_2 \\ 1 & e_1 & e_1 & e_1 & e_2 & 1 & e_2 & 1 & e_1 & 1 & e_1 & e_1 & e_1 & e_2 & 1 & e_2 & 1 & e_1 \\ 1 & e_2 & e_1 & e_1 & 1 & e_2 & e_2 & e_1 & 1 & 1 & e_2 & e_1 & e_1 & 1 & e_2 & e_2 & e_1 & 1 \\ 1 & 1 & 1 & e_2 & e_2 & e_2 & e_1 & e_1 & e_1 & 1 & 1 & 1 & e_2 & e_2 & e_2 & e_1 & e_1 & e_1 \\ 1 & e_1 & e_2 & e_2 & 1 & e_1 & e_1 & e_2 & 1 & 1 & e_1 & e_2 & e_2 & 1 & e_1 & e_1 & e_2 & 1 \\ 1 & e_2 & e_1 & e_2 & e_1 & 1 & e_1 & 1 & e_2 & 1 & e_2 & e_1 & e_2 & 1 & e_1 & 1 & e_1 & e_2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & e_1 & e_2 & 1 & e_1 & e_2 & 1 & e_1 & e_2 & -1 & -e_1 & -e_2 & -1 & -e_1 & -e_2 & -1 & -e_1 & -e_2 \\ 1 & e_2 & e_1 & 1 & e_2 & e_1 & 1 & e_2 & e_1 & -1 & -e_2 & -e_1 & -1 & -e_2 & -e_1 & -1 & -e_2 & -e_1 \\ 1 & 1 & 1 & e_1 & e_1 & e_1 & e_2 & e_2 & e_2 & -1 & -1 & -1 & -e_1 & -e_1 & -e_1 & -e_2 & -e_2 & -e_2 \\ 1 & e_1 & e_1 & e_1 & e_2 & 1 & e_2 & 1 & e_1 & -1 & -e_1 & -e_1 & -e_1 & -e_2 & -1 & -e_2 & -1 & -e_1 \\ 1 & e_2 & e_1 & e_1 & 1 & e_2 & e_2 & e_1 & 1 & -1 & -e_2 & -e_1 & -e_1 & -1 & -e_2 & -e_2 & -e_1 & -1 \\ 1 & 1 & 1 & e_2 & e_2 & e_2 & e_1 & e_1 & e_1 & -1 & -1 & -1 & -e_2 & -e_2 & -e_1 & -e_1 & -e_1 & -e_1 \\ 1 & e_1 & e_2 & e_2 & 1 & e_1 & e_1 & e_2 & 1 & -1 & -e_1 & -e_2 & -e_2 & -1 & -e_1 & -e_1 & -e_2 & -1 \\ 1 & e_2 & e_1 & e_2 & e_1 & 1 & e_1 & 1 & e_2 & -1 & -e_2 & -e_1 & -e_1 & -1 & -e_1 & -e_1 & -e_2 & -1 \end{bmatrix}$
-----	---

### Generalized Matrix Multiplications

For matrix formulation of Fourier transform on finite non-Abelian groups we need the generalized matrix multiplications defined as follows [52].

**DEFINITION 7** Let  $\mathbf{A}$  be an  $(m \times n)$  matrix with elements  $a_{ij} \in P$ ,  $i \in \{0, 1, \dots, m-1\}$ ,  $j \in \{0, 1, \dots, n-1\}$ . Let  $[\mathbf{B}]$  be an  $(n \times r)$  matrix whose elements  $\mathbf{b}_{jk}$ ,  $j \in \{0, \dots, n-1\}$ ,  $k \in \{0, 1, \dots, r-1\}$  are  $(p \times p)$  matrices of not necessarily mutually equal orders with elements in  $P$ . We define the product  $\mathbf{A} \odot [\mathbf{B}]$  as an  $(m \times r)$  matrix  $[\mathbf{Y}]$  whose elements  $y_{ik}$ ,  $i \in \{0, 1, \dots, m-1\}$ ,  $k \in \{0, 1, \dots, r-1\}$  are  $(p \times q)$  matrices with elements in  $P$  given by

$$y_{ik} = \sum_{j=0}^{n-1} a_{ij} \mathbf{b}_{jk}.$$

The product  $[\mathbf{B}] \odot \mathbf{A}$  is defined similarly.

**DEFINITION 8** Let  $[\mathbf{Z}]$  be an  $(m \times n)$  matrix whose elements  $\mathbf{z}_{ij}$ ,  $i \in \{0, 1, \dots, m-1\}$ ,  $j \in \{0, 1, \dots, n-1\}$  are the square matrices of not necessarily mutually equal orders with elements in  $P$ .

Let  $[\mathbf{B}]$  be an  $(n \times r)$  matrix whose elements  $\mathbf{b}_{jk}$ ,  $j \in \{0, 1, \dots, n-1\}$ ,  $k \in \{0, 1, \dots, r-1\}$  are square matrices of not necessarily mutually equal orders with elements in  $P$ . Under the condition that the matrices  $\mathbf{z}_{ij}$  and  $\mathbf{b}_{jk}$  are of the same order or, if not, that one of them is of the order 1, the product of matrices  $[\mathbf{Z}]$  and  $[\mathbf{B}]$  is defined as an  $(m \times r)$  matrix  $\mathbf{Y} = [\mathbf{Z}] \odot [\mathbf{B}]$  whose elements  $y_{ik} \in P$  are given by

$$y_{ik} = \sum_{j=0}^{n-1} \text{Tr}(\mathbf{z}_{ij} \mathbf{b}_{jk}).$$

**DEFINITION 9** Let  $[\mathbf{Z}]$  be an  $(m \times n)$  matrix whose elements  $\mathbf{z}_{ij}$ ,  $i \in \{0, 1, \dots, m-1\}$ ,  $j \in \{0, 1, \dots, n-1\}$  are  $(p \times q)$  matrices of not necessarily mutually equal orders with elements in  $P$ . Let  $[\mathbf{B}]$  be an  $(n \times r)$  matrix whose elements  $\mathbf{b}_{jk}$ ,  $j \in \{0, 1, \dots, n-1\}$ ,  $k \in \{0, 1, \dots, r-1\}$  are  $(s \times t)$  matrices of not necessarily mutually equal orders with elements in  $P$ . The elementwise Kronecker product of matrices  $[\mathbf{Z}]$  and  $[\mathbf{B}]$  is defined as an  $(m \times r)$  matrix  $[\mathbf{V}] = [\mathbf{Z}] \otimes [\mathbf{B}]$  whose elements  $v_{ik}$  are given by

$$v_{ik} = \sum_{j=0}^{n-1} \mathbf{z}_{ij} \otimes \mathbf{b}_{jk},$$

where  $\otimes$  denotes the ordinary Kronecker product.

### Authors' Biographies

**Radomir S. Stanković** received B.E. degree in Electronic Engineering from Faculty of Electronics, University of

TABLE IV Group operation of  $S_3$ 

0	1	2	3	4	5
0	0	1	2	3	4
1	1	2	0	5	3
2	2	0	1	4	5
3	3	4	5	0	1
4	4	5	3	2	0
5	5	3	4	1	2

Niš, in 1976, and M.Sc., and Ph.D. degrees in Applied Mathematics from Faculty of Electrical Engineering, University of Belgrade, in 1984, and 1986, respectively. He was with High School of Electrotechnic, Niš, from 1976 to 1987. From 1987 to date he is with Faculty of Electronic, Niš. Presently, he is a Professor teaching Logic Design. His research interests include switching theory and multiple-valued logic, signal processing and spectral techniques. He served as a co-editor and editor for two editorials and the author for a couple of monographs in spectral techniques.

**Milena Stanković** received B.E. degree in Electronic Engineering in 1976, and M.Sc., and Ph.D. degrees in Computer Science in 1982 and 1988 from Faculty of Electronics, University of Niš. She was with High School of Electrotechnic, Niš, from 1976 to 1978. From 1978 to date she is with Faculty of Electronics, Niš. Presently, she is a Professor teaching Programming Languages and Compilers. Her research interests include programming languages, spectral techniques, and signal processing. She was an author and co-author for a couple of monographs and textbooks in computer science.

**Reiner Creutzburg** has received his B.Sc. degree and Ph.D. degree at University of Rostock, Department of Mathematics, Germany, in 1976 and 1985, respectively. Since 1992, he is a full time Professor in Computer Science, Fachhochschule Brandenburg, University of Applied Sciences, Brandenburg, Germany. In 1992–1995, he was a Member of the Founding Senate, Fachhochschule Brandenburg. Since 1994, Prof. Creutzburg is the Head and Founding Director of the INM—Institute for Networking and Multimedia Brandenburg e.V. In 1997/98, he was a Visiting Professor at Tampere International Center for Signal Processing, Tampere University of Technology, Finland. Research interest of Prof. Creutzburg include digital signal and image processing, fast algorithms and transforms, fractals, wavelet transforms, data and image compression, algebraic approaches to signal processing, networked multimedia technology and education in electronic commerce, computer architecture (parallel memory organization, memory conflicts, skewing schemes, data access) consulting for companies and governmental agencies (multimedia, image processing, internet consulting), innovation and technology transfer. Prof. Creutzburg is the author of three books, two patents, approx. 80 articles, and a number of preprints and conference papers.

