Utilization of Eight-Variable Karnaugh Maps in the Exploration of Problems of Qualitative Comparative Analysis

Ali Muhammad Ali Rushdi\(^1\) and Raid Salih Badawi\(^1\)

\(^1\)Department of Electrical and Computer Engineering, King Abdulaziz University, P.O.Box 80200, Jeddah 21589, Saudi Arabia.

**Authors' contributions**

This work was carried out in collaboration between both authors. Author AMAR designed the study, performed the analysis, solved the example, deliberated over the QCA paradigm, contributed to the literature search, and wrote the preliminary manuscript. Author RSB managed the literature search, deliberated over the QCA paradigm, and drew the figures. Both authors read and approved the final manuscript.

**ABSTRACT**

Qualitative Comparative Analysis (QCA) is an emergent methodology of diverse applications in many disciplines. However, its premises and techniques are continuously subject to discussion, debate, and (even) dispute. We use a regular and modular Karnaugh map to explore a prominent recently-posed eight-variable QCA problem. This problem involves a partially-defined Boolean function (PDBF), that is dominantly unspecified. Without using the algorithmic integer-programming approach, we devise a simple heuristic map procedure to discover minimal sets of supporting variables. The eight-variable problem studied herein is shown to have at least two distinct such sets, with cardinalities of 4 and 3, respectively. For these two sets, the pertinent function is still a partially-defined Boolean function (PDBF), equivalent to \(2^8 = 1024\) completely-specified Boolean functions (CSBFs) in the first case, and to four CSBFs only in the second case. We obtained formulas for the four functions of the second case, and a formula for a sample fifth function in the

*Corresponding author: E-mail: arushdi@yahoo.com;*
1. INTRODUCTION

Boolean analysis (BA) is a kind of analysis utilizing (two-valued) Boolean functions (also known as switching functions). It was introduced by Flament [1] for application to questionnaire data. This analysis is a partial-order generalization of scalogram analysis, and is particularly useful for hierarchical data, and generally any type of data that can be dichotomized, i.e., reduced to optimal binary data. Boolean analysis is the core essence of Qualitative Comparative Analysis (QCA) first introduced in [2]. Comparative Analysis has now branched into three variants, namely crisp-set Qualitative Comparative Analysis (csQCA), multi-value Qualitative Comparative Analysis (mvQCA), and fuzzy-set Qualitative Comparative Analysis (fsQCA) [3,4]. Despite the diversified use of QCA in many disciplines, it is the subject of many kinds of criticism [5-15].

We note that a switching (two-valued Boolean) function has many types of formulas (See the Appendices). Of these, we mention the irredundant disjunctive forms (IDFs), the minimal sums (MSs), the complete sum (CS), together with their duals (the irredundant conjunctive forms (ICFs), the minimal products (MPs), the complete product (CP), as well as the probability-ready expressions (PREs). We stress that, among these formulas, only the two dual formulas of the complete sum and the complete product are guaranteed to be unique and canonical, and hence they are referred to as the Blake canonical form and the dual Blake canonical form, respectively [16].

There are many scientific disciplines which are well established and highly robust in their utilization of Boolean functions. These include digital design and testing, artificial intelligence, deductive inference, Boolean reasoning, and system reliability. Digital design and testing is usually interested in all the afore-mentioned types of formulas [17-21]. Deductive inference and Boolean reasoning usually deal with the complete sum or the complete product [21-29]. System reliability starts by constructing minimal sums (called minimal paths and minimal cutsets), but then it necessitates the construction of probability-ready expressions or simply disjoint sum-of-products expressions [30-40]. By contrast, QCA is a relatively new user of Boolean functions, which still has some serious un-answered challenges and an associated ongoing debate on the nature of Boolean formulas that it should use. Table 1 compares the fields of digital circuits and QCA as two prominent users of Boolean Analysis.

Crama et al. [41] addressed a problem of eight presumed variables, of a nature quite similar to that addressed by QCA. Out of 256 configurations in their problem, only 11 configurations were observable, and hence they dealt with a Boolean function that is partially defined [42-48]. They found that the problem has eight minimal sets of supporting variables, and for each of these minimal sets, they obtained (usually) several irredundant disjunctive forms (IDFs). Any of these IDFs could be the correct solution. Mainstream QCA typically seeks a more powerful, highly appealing and less ambiguous (albeit unjustified) result. It prefers the minimal sum among the set of all possible resulting IDFs, and hence it arbitrarily selects a set of supporting variables (occasionally forgetting about missing or extraneous variables) and it disregards the remaining prime implicants (if any) within the complete sum [11,13].

Thiem et al. [14] demonstrate that 'the uncritical import of Boolean optimization algorithms from electrical engineering into some areas of the social sciences in the late 1980s has induced algorithmic bias on a considerable scale over the last quarter century.' They caution 'scientists against letting methods and algorithms travel too easily across disparate disciplines without sufficient prior evaluation of their suitability for
Boolean functions (See Appendix E). The above exhaust this requirement is the class of unate class of Boolean functions that satisfy (but do not conjecture has only essential prime implicants and complete sum. A Boolean function satisfying this would be trusted only if it produces only a single IDF that is both a unique minimal sum and the complete result of a unique theory they aspired to have. Obviously since it does not supply the strong researches were aware of (and frequently cited) the work in [41], they practically ignored it, as demonstrating concepts, proving theorems, and achieving other pedagogical purposes such as providing pictorial insight (often unjustified) feature. Such a model is obtained at the expense of any competing solutions, which are eliminated on the grounds that they are dominated by the model selected. Thiem et al. [14] hold that the too strong results of QCA ‘have often neither been the upshot of the collection of high-quality data material nor the skillful use of pertinent theories but simply an algorithmic corollary of the uncritical import of the Quine-McCluskey (QMC) algorithm from electrical engineering.’ For digital circuits, this algorithm eliminates redundant disjunctive forms (IDFs) that cannot outperform one of them characterized as a minimal sum.

Thiem et al. [14] assert that QCA differs from methods based on counterfactual, interventionist, mechanistic or probabilistic theories of causation, since it relies on the regularity theory of INUS causation, i.e., causation pertaining to an Insufficient, but Necessary part of an Unnecessary but Sufficient condition. Under the INUS theory of causation, any minimally necessary disjunction of minimally sufficient conjunctions is a potential candidate for a causal explanation [49]. For the purposes of causal inference, the QMC algorithm must produce all redundant disjunctive forms, and not only a minimal sum. In fact, this is the technique that is employed by Crama et al. [41] to tackle other layers of ambiguity in their analysis, as they consider a Boolean function that is only partially defined, and whose minimal set of supporting variables is not unique. Their mathematically sound solution encompasses potentially several IDFs for each of the possible minimal sets of supporting variables. Though the QCA researchers were aware of (and frequently cited) the work in [41], they practically ignored it, obviously since it does not supply the strong result of a unique theory they aspired to have.

Rushdi and Zagzoog [16] conjecture that QCA would be trusted only if it produces only a single IDF that is both a unique minimal sum and the complete sum. A Boolean function satisfying this conjecture has only essential prime implicants and does not have any non-essential ones. A large class of Boolean functions that satisfy (but do not exhaust) this requirement is the class of unate Boolean functions (See Appendix E). The above conjecture reconciles the aspiration for minimality by mainstream QCA researchers, the search for IDFs by Crama et al. [41] and Thiem et al. [14], and the insistence on including the complete sum by Rushdi [11] and Rushdi & Badawi [13]. The search for a unate Boolean function in a QCA solution by Rushdi & Rushdi [12] conforms to this conjecture. The running example considered herein (taken from Delreux [50]) is also in favor for the validity of this conjecture.

This paper argues that the Boolean Analysis necessitated by QCA (whether it be supporting-variable identification, complete-sum derivation or enumeration of all IDFs) is preferably achieved by the manual pictorial tool of the Karnaugh map rather than by automated algorithms. It seems to us that QCA implementation of QMC via computer programs is a fast way of obtaining fast results without questioning or scrutinizing the underlying recipes, which might be black boxes for the users. Admittedly, the Karnaugh map is useful only for a small number of input variables. Nevertheless, the map plays an indispensable role in logic design, although typical real-life problems therein are large (and even extremely large). The map role in digital design is essentially limited to providing pictorial insight and achieving other pedagogical purposes such as providing pictorial insight, illustrating procedures, and exploring modular, repetitive or symmetric structures. The size-limitation of the Karnaugh map, however, is of no concern in the case of Boolean Analysis needed for QCA, where the typical number of input variables of ‘real-life’ problems seems to be eight to ten at most [8,12]. Some limited use of Karnaugh maps (of four or fewer variables) has been already made in QCA applications [8,9,11,12,51-53].

In this paper, we introduce a novel regular and modular eight-variable Karnaugh map, first introduced by Halder [54] and Motil [55], and efficiently used by Rushdi and Zagzoog [16] for the study of four-bit magnitude digital comparators. We employ this map for analyzing one of the largest QCA problems ever studied [50], being one of a few QCA problems involving eight or more explanatory variables (see, e.g., [50,56-64]). We compress the eight input binary variables as two hexadecimal variables, so as to facilitate the conversion of the original truth table into a an eight-variable Karnaugh map. Only observed configurations are explicitly shown in map cells entered as asserted high (1) or asserted low (0). Non-observed configurations
corresponding to logical remainders (don’t cares, in our language) are represented with blank cells (as usual). We observe that the solution in [50] (obtained by automated means) uses different sets of supporting variables for the two literals of one of the two outputs, and further makes different assignments of logical remainders for this output and its purported complement, as it performs individual and separate minimization for each of them. As a result, the disjunction of this output and its purported complement fails to equal 1, as required if complementarity is preserved. We use an eight-variable Karnaugh map to explore the problem in [50], which involves a partially-defined Boolean function (PDBF), that is dominantly unspecified. Without using the integer-programming approach of Crama et al. [41], we devise a simple map procedure to discover minimal sets of supporting variables. The problem taken from [50] is shown to have at least two distinct such sets, with cardinalities of 4 and 3, respectively. For these two sets, the pertinent function is still a partially-defined Boolean function (PDBF), equivalent to \(2^{10} = 1024\) completely-specified Boolean functions (CSBFs) in the first case, and to four CSBFs in the second case. We obtained formulas for the four functions of the second case, and a formula for a sample fifth function in the first case. Although only this fifth function is unate, each of the five functions studied does not have a non-essential prime implicant, and hence each of them enjoys the desirable feature of having a single IDF that is both a unique minimal sum and the complete sum.

The organization of the rest of this paper is as follows. Section 2 introduces the regular and modular version of the Karnaugh map used herein. This map version can be (theoretically) extended to an arbitrary large number of variables, and includes all maps of smaller sizes as special cases. Section 3 presents the QCA problem of Delreux [50] used as a current example herein. Section 4 revisits this problem from a map perspective, and contributes several improved solutions. Section 5 concludes the paper. To make the paper self-contained, it is supplemented with five appendices. Appendix A explains basic concepts of Boolean minimization. Appendix B is about the complete sum. Appendix C defines probability-ready expressions. Appendix D briefly introduces the Boole-Shannon expansion. Appendix E deals with unate Boolean functions.

2. ON THE REGULAR EIGHT-VARIABLE KARNAUGH MAP

The Karnaugh map is an enhanced form of the truth table [65], in which two dimensions (rather than one) are used, and in which reflected binary ordering or grey ordering (as opposed to direct binary ordering) is employed. The \(n\)-variable map consists of \(2^n\) cells, such that every cell has \(n\) neighboring cells or logically adjacent cells. Two cells are (first) neighbors or (immediately) adjacent if their variable values except one are exactly the same. Such two cells are said to have a Hamming distance [66-75] of one or to differ in exactly one-bit position. The map is constructed such that any two logically adjacent cells are made also as visually adjacent as possible. In general, two logically adjacent cells appear as the mirror images with respect to boundary lines separating the internal and external domains of the variable in whose value the two cells differ (See Fig. 1).

Typically, the Karnaugh map is conveniently used up to six variables [17-20,65]. There are occasions in which Karnaugh maps of eight variables are used, in which the rectangular shape of cells is abandoned to a triangular shape [76-80]. In this paper, however, we will use the aforementioned regular and modular form of the Karnaugh map that appeared earlier in [16,54,55,81-84], and is such that

a) The rectangular shape of the cell is retained.

b) The internal domain of the \((n + 1)\)st variable is introduced to be centered around the boundary lines of the \((n - 1)\)st variable (See Fig. 2).

We note that the process outlined in (b) above can be, in theory, indefinitely continued. Hence, there is no theoretical upper bound on the size of the Karnaugh map constructed this way. However, as the number of variables increases, the size of the map increases exponentially, and its utility diminishes very quickly due to prohibitively increasing difficulty. As a demonstration of the usefulness of the aforementioned version of the Karnaugh map, we present its case of eight variables herein. This map suffices to handle most problems of QCA. In fact, the largest number of variables we came across in a QCA paper was just ten [62].
Table 1. Comparison of variants of Boolean analysis encountered in digital circuits and in Qualitative Comparative Analysis (QCA)

|                                | Digital Circuits                              | Qualitative Comparative Analysis (QCA)                                                                 |
|--------------------------------|-----------------------------------------------|-------------------------------------------------------------------------------------------------------|
| Function behavior when it is specified or defined | Prescribed                                   | Known or observed                                                                                   |
| Function behavior when it is not specified or defined | Arbitrary, irrelevant, and cannot be obtained | Unknown but relevant, and might be obtained at the cost of additional observations that might require prohibitively long times |
| Can undefined configurations be defined later | No                                            | Yes                                                                                                   |
| Goal of study                  | Selection of an optimal function among the family of extensions for a partially defined Boolean function (PDBF) | Realistic Goal: Gaining an overview of the whole family of PDBFs and noting common features among members of this family Claimed Goal: Identifying one particular formula for a member of this family |
| Objective function of the optimization algorithm | Sum minimality (minimal cost solution)         | Conventionally: sum minimality                                                                        |
| Minimal sets of supporting variables | Usually, one set of full cardinality, viz., the set of all variables, i.e., the function cannot be made independent of any variable | Should be: sum irredundancy [14]                                                                       |
| Names given to undefined points or configurations | Don’t cares (cases that never happen, or in which the output is indeterminate) | Frequently, several sets of various cardinalities, i.e., the function can be made independent of some variables in different ways Logical remainders |
| Variable symbols               | Simple short abstract symbols                  | Lengthy mnemonic symbols                                                                             |
| Designation of a complemented variable | Using an overhead bar (as opposed to not using one) | A lower-case letter (as opposed to an upper-case one)                                                |
Fig. 1. The general layout of the eight-variable Karnaugh map used herein. The cell colored in green (column 15 and row 0) represents the minterm $x_1\overline{x}_2\overline{x}_3\overline{x}_4\overline{x}_5\overline{x}_6\overline{x}_7\overline{x}_8$ or the bit sequence 10101010. Its eight logically adjacent or neighboring cells are highlighted in yellow. Only four of these cells are visually adjacent to the original cell when the map is viewed to lie on a torus.

Fig. 2. The eight-variable Karnaugh map of Fig. 1. There are two borders of the variable $X_3$ (separating its internal domain ($X_3 = 1$) and external domain ($X_3 = 0$)), which are highlighted in bold. There are two internal regions for the variable $X_5$ (colored) which are centered around these borders.
The construction of a Karnaugh map is based on the Boole-Shannon Expansion (See Appendix D), and its intimately-related concept of a subfunction [85-88] or Boolean quotient [21,34]. Various types of map folding allow the replacement of an $n$-variable map by two $(n-1)$-variable maps [8,9,12,18,89-91]. Conversely, the $n$-variable map might be viewed as a map-entered map [92-95] with a new map variable, say $Z_n$, and two major cells, each of which having the size of an $(n-1)$-variable map. Such a view might be repeated recursively, so as to construct a map of any desirable size. Various map relations and properties allow the map to provide instructive and pedagogical exposition of many important concepts in Boolean algebra and switching theory including those of implicants, prime implicants, essential prime implicants, minimal sum, complete sum (Blake Canonical Form) [96,97].

If a Karnaugh map is used to represent a Boolean function $f(X)$, then the map can be split into two halves (with respect to the borders of the variable $X_k$) representing the internal and external domains of this variable. These half maps depict, respectively, the two subfunctions or Boolean quotients $f(X)/\bar{X}_k$ and $f(X)/X_k$, which are functions of the $(n-1)$ variables of $X$ other than $X_k$. We say that $f(X)$ is vacuous in (independent of) $X_k$ if the following relation is identically satisfied [8,17,74,75].

$$f(X)/\bar{X}_k = f(X)/X_k$$  \hspace{1cm} (1)

According to (D1), if (1) is satisfied then $f(X)$ becomes equal to each of $f(X)/\bar{X}_k$ and $f(X)/X_k$, and hence it becomes a function of the $(n-1)$ variables of $X$ other than $X_k$. This means that $X_k$ is now guaranteed to be not a supporting variable of $f(X)$. The variable $X_k$ must be included in a set of supporting variables of $f(X)$ if at least a cell in the half map $f(X)/\bar{X}_k$ is found to differ in its entry from its image cell in the half map $f(X)/X_k$ (obtained through reflection with respect to the nearest border of $X_k$). Note that only the 0 and 1 entries are viewed as different, opposing or contradictory, while a don’t-care entry (d) can be made to match either 0 or 1. The variable $X_k$ might be excluded from a set of supporting variables of $f(X)$ if (a) no case of the above-mentioned pair of contradictory image cells can be found, and (b) appropriate conditions are placed on don’t-care cells in either the $f(X)/\bar{X}_k$ or $f(X)/X_k$ half maps that are images of 0-entered or 1-entered cells in the other half so as to ensure that (1) is satisfied.

3. A PROMINENT QCA PROBLEM USED AS A RUNNING EXAMPLE

In Table 2, we present the running example to be used throughout this paper, which is taken from Delreux [50]. Table 2 is a truth table with a single ternary endogenous factor (outcome) $Z$ and a set of eight binary exogenous (input) factors $(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8)$. All variables of Delreux [50] have lengthy mnemonic names conveying technical meanings (Table 3), which are renamed herein with simple abstract symbols, so as to facilitate the reconstruction of the truth table as a Karnaugh map. The set of eight input factors is split into two distinct sets of Variables $X_H = (X_1, X_2, X_5, X_6)$ and $X_L = (X_3, X_4, X_7, X_8)$, which serve as the horizontal and vertical variables of the Karnaugh map, respectively. Each row in Table 2 represents a unique configuration that is characterized by a binary number $(X_1X_2X_3X_4X_5X_6X_7X_8)_2$ or a hexadecimal number $(X_HX_L)_{16}$ (Table 4), where

$$X_H = 2^3X_1 + 2^2X_2 + 2^1X_5 + 2^0X_6 = 8X_1 + 4X_2 + 2X_5 + X_6$$  \hspace{1cm} (2)

$$X_L = 2^3X_5 + 2^2X_6 + 2^1X_7 + 2^0X_8 = 8X_5 + 4X_6 + 2X_7 + X_8$$  \hspace{1cm} (3)

Note that the hexadecimal number system uses 16 symbols, the first 10 of which (0,1,2,...,9) are borrowed (with the same meaning) from the conventional decimal system, while the remaining 6 symbols are the beginning uppercase letters of the alphabet A, B, C, D, E, and F used to designate the values 10, 11, 12, 13, 14, and 15, respectively, in decimal, which correspond to 1010,1011,1100,1101,1110 and 1111, respectively in binary (Table 4). Designation of a configuration in Table 2 by a hexadecimal number $X_HX_L$ considerably facilitates the conversion of the truth table in Table 2 to the Karnaugh map in Fig.3. The configuration $X_HX_L$ is simply located at the horizontal coordinate $X_H$ and the vertical coordinate $X_L$. The output variable $Z$ is a ternary variable of a value belonging to [0,1,2], encoded as two dummy binary variables $Z_u$ and $Z_l$. The binary variable $Z_u$ is an indicator that is effective when $Z_L = 0$. It selects $Z = 1$ when $Z_u = 0$ and $Z = 2$ when $Z_u = 1$. The binary variable $Z_l$ is an indicator for $Z = 0$ ($Z_l = 1$ means $Z = 0$ and $Z_l = 0$ means either $Z = 1$ or $Z = 2$, depending on the value of $Z_u$).
Table 2. Truth table of the running example, adapted from Delreux [50]. Only 13 out of 256 lines or configurations are specified

| $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_H$ | $X_L$ | $Z$ | $Z_H$ | $Z_L$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----|-------|-------|
| 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0   | d     | 1     |
| 0     | 0     | 0     | 0     | 0     | 0     | 1     | 0     | 1     | 0     | d   | 1     | 1     |
| 0     | 1     | 0     | 0     | 0     | 0     | 0     | 0     | 4     | 0     | 0   | d     | 1     |
| 0     | 1     | 0     | 0     | 0     | 0     | 1     | 4     | 1     | 0     | d   | 1     | 1     |
| 1     | 1     | 0     | 0     | 0     | 0     | 1     | C     | 1     | 0     | d   | 1     | 1     |
| 0     | 1     | 0     | 0     | 0     | 0     | 1     | 0     | 4     | 2     | 1   | 0     | 0     |
| 0     | 1     | 0     | 0     | 0     | 0     | 1     | 1     | 4     | 3     | 1   | 0     | 0     |
| 1     | 1     | 1     | 0     | 1     | 1     | 1     | D     | 7     | 1   | 0   | 0     | 0     |
| 1     | 1     | 1     | 0     | 1     | 1     | 0     | D     | 6     | 1   | 0   | 0     | 0     |
| 0     | 0     | 1     | 0     | 1     | 1     | 1     | 2     | F     | 2   | 1   | 0     | 0     |
| 0     | 0     | 1     | 1     | 1     | 0     | 1     | 3     | B     | 2   | 1   | 0     | 0     |
| 1     | 1     | 0     | 0     | 0     | 1     | 1     | 0     | C     | 6   | 2   | 1     | 0     |
| 1     | 1     | 1     | 1     | 0     | 1     | 0     | F     | 6     | 2   | 1   | 0     | 0     |

Table 3. Translation of the notation of Delreux [50] to our simpler notation

| Notation of Delreux [50] | Meaning |
|--------------------------|---------|
| $X_1$ prefprin           | Preference homogeneity among the principals |
| $X_2$ prefpa             | The degree of preference homogeneity between principals |
| $X_3$ polit              | Level of politicization |
| $X_4$ infag              | Information asymmetry in favor of the agent |
| $X_5$ infprin            | Asymmetrically divide information in favor of the principals |
| $X_6$ inst               | The degree of institutional density |
| $X_7$ extcomp            | Pressure of the compellingness of the external environment |
| $X_8$ agrprin            | A subset of the principals |
| $Z$ disc                 | Ternary outcome variable, encoded as two dummy binary variables $Z_H$ and $Z_L$ |
| $Z_H$ disc_high          | Binary indicator that is effective when $Z_L = 0$. It selects $Z = 1$ when $Z_H = 0$ and $Z = 2$ when $Z_H = 1$ |
| $Z_L$ disc_low           | Binary indicator for $Z = 0$ |
### Table 4. Hexadecimal ‘digits’ in terms of decimal digits and binary bits

| Hexadecimal ‘Digits’ | Decimal Digits | Binary Bits |
|----------------------|----------------|-------------|
| 0                    | 0              | 0000        |
| 1                    | 1              | 0001        |
| 2                    | 2              | 0010        |
| 3                    | 3              | 0011        |
| 4                    | 4              | 0100        |
| 5                    | 5              | 0101        |
| 6                    | 6              | 0110        |
| 7                    | 7              | 0111        |
| 8                    | 8              | 1000        |
| 9                    | 9              | 1001        |
| A                    | 10             | 1010        |
| B                    | 11             | 1011        |
| C                    | 12             | 1100        |
| D                    | 13             | 1101        |
| E                    | 14             | 1110        |
| F                    | 15             | 1111        |
The truth table in Fig. 3 is notorious for its extremely few specifications. Out of $2^8 = 256$ possible configurations, data is available for only 13 configurations. The fact is vividly illustrated by Fig. 3, in which the Karnaugh map has 256 cells, with only 8 of them of assigned asserted entries of 00 or 10, plus 5 cells having a semi asserted entry of d1. The remaining 243 cells are left blank, indicating that they are logical remainders (in QCA terminology) or don’t-cares (in the digital-design terminology).

Delreux [50] obtained the following solutions for his problem

\[ Z_L = \overline{X}_7, \]  
\[ \overline{Z}_L = X_7, \]  
\[ Z_H = X_6 (\overline{X}_3 \lor \overline{X}_1 X_4) \lor X_3 X_4, \]  
\[ \overline{Z}_H = \overline{X}_4 (\overline{X}_6 \lor X_3 \overline{X}_5). \]

While the results of $Z_L$ and $\overline{Z}_L$ in (4) and (5) are consistent, those of $Z_H$ and $\overline{Z}_H$ in (6) and (7) are problematic. These latter purported complements in (6) and (7) are obviously obtained via independent automated minimizations that allowed contradictory assignments of logical remainders (don’t-cares). We have two obvious objections on the validity of the set $Z_H$ and $\overline{Z}_H$ in (6) and (7), namely

(a) The purported complements $Z_H$ and $\overline{Z}_H$ have different sets of supporting variables. The uncomplemented literal $Z_H$ has a set $\{X_1, X_3, X_4, X_5, X_6\}$ of supporting variables, which is a strict superset of the set $\{X_2, X_4, X_5, X_6\}$ of supporting variables for the complemented literal $\overline{Z}_H$. Anyhow, there is no guarantee that either set is the true set of supporting variables for the output variable $Z_H$.

(b) The disjunction of the uncomplemented literal $Z_H$ and its purported complement $\overline{Z}_H$ is not identically equal to 1 as it ought to be. Fortunately, the conjunction of these purported complements is 0 as required.

These two objections are not pertaining to the results in [50] only, but are expected to arise frequently in general QCA practice. We raised similar objections [8,98] about handling complements in a much smaller problem of just four input variables that appeared first in the area of peace research [99]. We presented in [13] detailed mathematical investigation for the
necessity of consistent handling of complementary literals. Despite our repeated criticism of the way of handling logical remainders in QCA circles, no adequate response from the QCA community is obtained so far, and the mainstream QCA journals have never addressed the issue at all, as if it were of no genuine scientific concern.

4. MAP EXPLORATION OF THE RUNNING EXAMPLE

Our running example is henceforth represented by the 8-variable Karnaugh map in Fig. 3 for \( Z_LZ_H \). We first consider the low binary dummy variable \( Z_L \) represented by the 8-variable Karnaugh map in Fig. 4. We immediately see that this variable has a single minimal supporting set of variables \( \{X_1\} \), which contains the single variable \( X_7 \). Fig. 5 demonstrates that \( X_7 \) cannot be excluded from any minimal set of supporting variables for \( Z_L \) since \( Z_L \) cannot be made independent of \( X_7 \). In fact, the relation \( Z_L/X_7 = Z_L/X_7 \) cannot be identically satisfied (the 1’s in the cells 40 and 41 (shown blue) contradict the 0’s in their image cells 42 and 43 (shown red), respectively). Fig. 6 demonstrates that \( Z_H \) can be made independent of \( X_6 \) since (a) no pair of contradictory image cells can be found in \( Z_L/X_6 \) and \( Z_L/X_6 \), and (b) appropriate conditions are placed on don’t-care cells in either the \( Z_L/X_6 \) or \( Z_L/X_6 \) half maps that are images of 0-entered or 1-entered cells in the other half so as to ensure that \( Z_L/X_6 = Z_L/X_6 \) is identically satisfied. In fact, we can add further conditions to Fig. 6 to make \( Z_L \) further independent of \( X_2, X_3, X_4, X_5, X_6, X_7, X_8 \). This happens when the region of \( Z_L \) is filled with 0’s, while that of \( Z_L \) is filled with 1’s. Fig. 7 demonstrates that \( Z_H \) can be made independent simultaneously of the seven variables \( X_1, X_2, X_3, X_4, X_5, X_6, X_8 \), thereby asserting that all these seven variables can be simultaneously excluded from a set of supporting variables of \( Z_L \), and hence verifying correctness of expressions (4) and (5).

We now consider the high binary dummy variable \( Z_H \) represented by the 8-variable Karnaugh map in Fig. 8. Fig. 9 shows that both \( X_4 \) and \( X_6 \) must be included in a supporting set for \( Z_H \). Fig. 9 also adds extra conditions (shown in red) to exclude \( X_1 \) from such a set, upon which \( X_6 \) must join this set. Fig. 10 adds further conditions to Fig. 9 to exclude \( X_2 \) also from the set of supporting variables for \( Z_H \), upon which \( X_5 \) must join this set, which is currently the set \( \{X_2, X_3, X_5, X_6\} \). Fig. 11 asserts that this set is indeed a minimal supporting set of variables for \( Z_H \), as this figure adds further conditions to Fig. 10 to ensure that \( Z_H \) can be made further independent of \( X_7 \) and then of \( X_8 \). Fig. 12 summarizes our findings by reducing Fig. 11 to a 4-variable map with members of the set \( \{X_3, X_4, X_5, X_6\} \) as map variables. We will shortly see the great advantage attained by this reduction.

Fig. 13 mimics what is done by most automated minimization procedures by attacking the 8-variable problem as it is. The figure shows our initial attempt at obtaining an IDF for the high binary dummy variable \( Z_H \). Our result is an obvious improvement over the result (6) of Delreux [50]. This result amounts to selecting (as an afterthought) the set \( \{X_3, X_4, X_5, X_6\} \) (referred to above) as a minimal set of supporting variables for \( Z_H \). We will demonstrate now that this set is not the only such set, and that it is surpassed by a set of a smaller cardinality.

\[
Z_H = \overline{X}_3 X_6 \lor X_4 \lor X_5 \tag{8}
\]

Fig. 14 again asserts that both \( X_3 \) and \( X_4 \) must be included in a supporting set for \( Z_H \), but it now adds different extra conditions to exclude \( X_2, X_5, X_6, X_8 \) from such a set, upon which \( X_1 \) must join this set. This means that the high binary dummy variable \( Z_H \) has a (minimal) supporting set of three variables \( \{X_1, X_3, X_4\} \). Fig. 15 is a s-variable map obtained as a reduction of the one in Fig. 14. The ratio of specified configurations/cells increased dramatically from 8/256 in Fig. 8 to 6/8 in Fig. 15, which contains only two don’t-care values, denoted by \( d_2 \) and \( d_3 \). The partially–defined Boolean function (PDBF) in Fig. 15 is actually four genuine (completely specified) functions, and should be treated as such Fig. 16 shows the first CSBF (\( d_2 = d_3 = 1 \)) for the PDBF of supporting variables \( \{X_1, X_3, X_4\} \). Each of the functions \( Z_H \) and \( \overline{Z}_H \) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown in the figure (which is both a minimal sum and the complete sum), namely.

\[
Z_H = X_4 \lor \overline{X}_1 X_3 \lor X_1 \overline{X}_3 \tag{9}
\]
\[
\overline{Z}_H = \overline{X}_1 \overline{X}_3 \overline{X}_4 \lor X_1 X_3 X_4 \tag{10}
\]
Fig. 4. An 8-variable Karnaugh map for the low binary dummy variable

Fig. 5. Demonstration that $X_7$ cannot be excluded from any minimal set of supporting variables for $Z_L$ since $Z_L$ cannot be made independent of $X_7$. The relation $Z_L/X_7 = Z_L/X_7$ cannot be identically satisfied
Fig. 6. Demonstration that $Z_L$ can be made independent of $X_6$

Fig. 7. Demonstration that $Z_L$ can be made independent simultaneously of $X_1, X_2, X_3, X_4, X_5, X_6,$ and $X_8$. This happens when the region of $Z_L$ is filled with 0’s, while that of $\bar{Z}_L$ is filled with 1’s. Original entries are highlighted in yellow
Fig. 8. An 8-variable Karnaugh map for the high binary dummy variable

Fig. 9. Both $X_5$ and $X_6$ must be included in a supporting set for $Z_H$, and extra conditions (shown in red) exclude $X_1$ from such a set. Now, $X_6$ joins this set
Fig. 10. Further conditions added to Fig. 9 to exclude $X_2$ from the supporting set of variables for $Z_H$. Now, $X_3$ joins this set.

Fig. 11. The high binary dummy variable with a supporting set of four variables $\{X_3, X_4, X_5, X_6\}$.
Fig. 12. The high binary dummy variable with a supporting set of four variables \( \{X_2, X_4, X_5, X_6\} \), a reduction of Fig. 11

\[ Z_H = X_3 X_6 \lor X_4 \lor X_5 \]

Fig. 13. Our initial attempt at obtaining an IDF for the high binary dummy variable

Fig. 17 shows the first CSBF (\( d_2 = 0, \ d_3 = 1 \)) for the PDBF of supporting variables \( \{X_1, X_3, X_4\} \). Again, each of the functions \( Z_{H_1} \) and \( Z_{H_2} \) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown in the figure (which is both a minimal sum and the complete sum), namely.

\[
Z_{H_1} = X_3 X_4 \lor \overline{X_1} X_3 \lor X_1 \overline{X_3}, \quad (11)
\]

\[
Z_{H_2} = \overline{X_1} \overline{X_3} \lor X_1 X_3 \overline{X_4}. \quad (12)
\]

Fig. 18 shows the first CSBF (\( d_2 = 1, \ d_3 = 0 \)) for the PDBF of supporting variables \( \{X_1, X_3, X_4\} \). Once more, each of the functions \( Z_{H_1} \) and \( Z_{H_2} \) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown in the figure (which is both a minimal sum and the complete sum), namely.

\[
Z_{H_1} = X_3 \overline{X_3} \overline{X_4} \lor X_1 X_3 \lor \overline{X_1} X_4 \lor X_3 X_4 \quad (13)
\]

\[
Z_{H_2} = \overline{X_1} \overline{X_3} \overline{X_4} \lor X_1 X_3 \overline{X_4} \lor X_1 \overline{X_3} X_4. \quad (14)
\]
Fig. 14. Both $X_3$ and $X_4$ must be included in a supporting set for $Z_{II}$, and extra conditions exclude $X_2, X_5, X_6$ and $X_8$ from such a set. Now, $X_1$ joins this set.

$$
\begin{array}{c|c|c|c}
0 & d_2 & 1 & 1 \\
\hline
X_1 & 1 & d_3 & 1 \\
\end{array}
$$

Fig. 15. The high binary dummy variable with a supporting set of three variables ($X_1, X_3, X_4$), a reduction of Fig. 14. The ratio of specified configurations/cells increased dramatically from 8/256 to 6/8.

Fig. 19 shows the first CSBF ($d_2 = d_3 = 0$) for the PDBF of supporting variables $\{X_1, X_3, X_4\}$. Also in this case, each of the functions $Z_{II}$ and $\overline{Z}_{II}$ has all-essential prime implicants (PIs), and hence it possesses the single IDF shown in the figure (which is both a minimal sum and the complete sum), namely.

$$
Z_{II} = X_1 \overline{X}_3 \overline{X}_4 \lor \overline{X}_1 X_3 \lor X_3 X_4, \quad (15)
$$

$$
\overline{Z}_{II} = \overline{X}_1 \overline{X}_3 \lor X_1 X_3 \overline{X}_4 \lor \overline{X}_3 X_4. \quad (16)
$$

Fig. 20 shows one particular CSBF among the 1024 CSBFs of a supporting set of four variables $\{X_3, X_4, X_5, X_6\}$. This CSBF is obtained by one particular specification (shown in red) of the don’t-cares in Fig. 12. Each of the functions $Z_{II}$ and $\overline{Z}_{II}$ has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). By contrast to the functions in Figs. 16-19, the present function is unate (of a specific polarity for every variable). This solution is a complementarity-preserving improvement of the one in Delreux [50], and is given by

$$
Z_{II} = X_4 \lor X_5 \lor \overline{X}_3 X_6, \quad (17)
$$

$$
\overline{Z}_{II} = X_3 \overline{X}_4 \overline{X}_5 \lor \overline{X}_4 \overline{X}_5 \overline{X}_6. \quad (18)
$$
Fig. 16. The first CSBF \((d_2 = d_3 = 1)\) for the PDBF of supporting variables \(\{X_1, X_3, X_4\}\). Each of the functions \(Z_{\mu}^+\) and \(Z_{\mu}^-\) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). A star is used to characterize the entry that renders a PI essential (being solely covered by this PI).

\[
Z_{\mu} = X_1 X_2 X_3 \lor X_1 X_2 X_4 \lor X_1 X_3 X_4
\]

Fig. 17. The second CSBF \((d_2 = 0, d_3 = 1)\) for the PDBF of supporting variables \(\{X_1, X_3, X_4\}\). Each of the functions \(Z_{\mu}^+\) and \(Z_{\mu}^-\) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). A star is used to characterize the entry that renders a PI essential (being solely covered by this PI).

\[
Z_{\mu} = X_1 X_2 X_3 \lor X_1 X_2 X_4 \lor X_1 X_3 X_4
\]
Fig. 18. The third CSBF \((d_2 = 1, d_3 = 0)\) for the PDBF of supporting variables \(\{X_1, X_3, X_4\}\). Each of the functions \(Z_H\) and \(\overline{Z}_H\) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). A star is used to characterize the entry that renders a PI essential (being solely covered by this PI).

\[
Z = X_1 X_2 X_4 \lor X_1 X_3 X_4 X_5 X_6
\]

Fig. 19. The third CSBF \((d_2 = d_3 = 0)\) for the PDBF of supporting variables \(\{X_1, X_3, X_4\}\). Each of the functions \(Z_H\) and \(\overline{Z}_H\) has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). A star is used to characterize the entry that renders a PI essential (being solely covered by this PI).

\[
Z = X_1 X_2 X_4 \lor X_1 X_3 X_4 X_5 X_6
\]
Fig. 20. One particular CSBF among the 1024 CSBFs of a supporting set of four variables $\{X_3, X_4, X_5, X_8\}$. This CSBF is obtained by one particular specification (shown in red) of the don’t-cares in Fig. 12. Each of the functions $Z_\mu$ and $\overline{Z}_\mu$ has all-essential prime implicants (PIs), and hence it possesses the single IDF shown (which is both a minimal sum and the complete sum). A star is used to characterize the entry that renders a PI essential (being solely covered by this PI). By contrast to the functions in Figs. 16-19, the present function is unate (of a specific polarity for every variable). This solution is a complementarity-preserving improvement of the one in Delreux [50].

5. CONCLUSIONS

We used a regular and modular eight-variable Karnaugh map to explore a large prominent problem of Qualitative Comparative Analysis (QCA). This problem involves an eight-variable partially-defined Boolean function (PDBF), that is dominantly unspecified. Without using the integer-programming approach, we devised a simple map procedure to discover minimal sets of supporting variables in the outset before proceeding to seek IDF representations. We found that the function considered has at least two minimal sets of supporting variables, and derived five sets (out of many possible) of complementarity-preserving solutions. Each of these sets of solutions is free from the shortcomings of the set of solutions already available, and political scientists are invited to investigate them and explore whether they make sense in the original context.

The Karnaugh map seems to be a very promising tool for QCA-related explorations that do not involve just solutions with the QCA methodology. Examples of such explorations include the issues of simplifying assumptions [100] or reducing complexity [101]. We hope that our map techniques and results would be utilized in future QCA applications.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Flament C. L’analyse booléenne de questionnaires (Boolean analysis of questionnaires). Mathématiques et Sciences humaines. 1965;12:3-10.
2. Ragin CC, Mayer SE, Drass KA. Assessing discrimination: A Boolean approach. American sociological review. 1984:221-234.
3. Wagemann C, Siewert MB. Qualitative Comparative Analysis, Wagemann et al. (Editors), Handbuch Methoden der Politikwissenschaft (Handbook of Political Science Methods), Springer. 2018:1-33.
4. Schneider CQ, Wagemann C. Standards of good practice in qualitative comparative analysis (QCA) and fuzzy-sets. Comparative Sociology. 2010;9(3):397–418.
5. Lieberhonen S. Comments on the use and utility of QCA. Qualitative Methods: Newsletter of the American Political Science Association, Organized Section on Qualitative Methods. 2004;2(2):13-14.

6. Lucas SR, Szatrowski A. Qualitative comparative analysis in critical perspective. Sociological Methodology. 2014;44(1):1-79.

7. Thiem A. Navigating the complexities of qualitative comparative analysis: Case numbers, necessity relations, and model ambiguities. Evaluation Review. 2014; 38(6):487-513.

8. Rushdi AMA, Badawi RMS. Karnaugh-map utilization in Boolean analysis: The case of war termination. Journal of Qassim University: Engineering and Computer Sciences. 2017;10(1):53-88.

9. Rushdi AMA, Badawi RMS. Karnaugh map utilization in Coincidence Analysis, Journal of King Abdulaziz University: Faculty of Computers and Information Technology Sciences. 2017;6(1-2):37-44.

10. Baumgartner M, Thiem A. Model ambiguities in configurational comparative research. Sociological Methods & Research. 2017;46(4):954-987.

11. Rushdi AMA. Utilization of Karnaugh maps in multi-value qualitative comparative analysis, International Journal of Mathematical, Engineering and Management Sciences (IJMEMS). 2018;3(1):28-46.

12. Rushdi RA, Rushdi AM. Karnaugh-map utility in medical studies: The case of Fetal Malnutrition. International Journal of Mathematical, Engineering and Management Sciences (IJMEMS). 2018;3(3):220-244.

13. Rushdi AMA, Badawi RMS. Computer engineers look at Qualitative Comparative Analysis. International Journal of Mathematical, Engineering and Management Sciences (IJMEMS). 2019;4(4):851-860.

14. Thiem A, Mkrtchyan L, Haesebrouck T, Sanchez D. Algorithmic bias in social research: A meta-analysis. PloS one. 2020;15(6):e0233625.

15. Baumgartner M, Thiem A. Often trusted but never (properly) tested: evaluating qualitative comparative analysis. Sociological Methods & Research. 2020;49(2):279-311.

16. Rushdi AMA, Zagzoog SS. Utilization of eight-variable karnaugh maps in the digital design of n-bit comparators. Chapter 5 in theory and practice of mathematics and computer science Vol., Book Publisher International, Hooghly, West Bengal, India. 2020:868-96.

17. Lee SC. Modern switching theory and digital design, prentice-hall, englewood Cliffs, New Jersey, NJ, USA; 1978.

18. Muroga S. Logic design and switching theory, John Wiley, New York, NY, USA; 1979.

19. Hill FJ, Peterson GR. Computer aided logical design with emphasis on VLSI, 4th Edition, Wiley, New York, NY, USA; 1993.

20. Roth C, Kinney L. Fundamentals of logic design,7th Edition, Cengage Learning, Stamford, CT, USA; 2014.

21. Brown FM. Boolean reasoning: The logic of Boolean equations. Kluwer Academic Publishers, Boston, USA; 1990.

22. Rushdi AM, Ba-Rukab OM. Map derivation of the closures for dependency and attribute sets and all candidate keys for a relational database. Journal of King Abdulaziz University: Engineering Sciences. 2014;25(2):3-34.

23. Rushdi AMA, Albarakati HM. Construction of general subsumptive solutions of Boolean equations via complete-sum derivation. Journal of Mathematics and Statistics. 2014;10(2):155-168.

24. Rushdi AM, Alshehri TM, Zarouan M, Rushdi MA. Utilization of the modern syllogistic method in the exploration of hidden aspects in engineering ethical dilemmas. Journal of King Abdulaziz University: Computers and Information Technology. 2014;3(1):73-127.

25. Rushdi AM, Zarouan M, Alshehri TM, Rushdi MA. The incremental version of the Modern Syllogistic Method. Journal of King Abdulaziz University: Engineering Sciences. 2015;26(1):25-51.

26. Rushdi AM, Rushdi MA. Utilization of the modern syllogistic method in the service of academic advising. In Proceedings of KAU Conference on Academic Advising in Higher Education. 2015;229-241.

27. Rushdi AM, Rushdi MA. Switching-algebraic algorithmic derivation of candidate keys in relational databases. Proceedings of the IEEE International Conference on Emerging Trends in Communication Technologies (ICETCT-2016;1-6.

28. Rushdi AM, Rushdi MA. Mathematics and examples of the modern syllogistic method
of propositional logic. In Ram, M. (Editor), Mathematics Applied in Information Systems, Bentham Science Publishers, Emirate of Sharjah, United Arab Emirates. 2018;6:123-167.

29. Rushdi AMA, Ghaleb FAM. Novel characterizations of the JK Bistables (Flip Flops). Journal of Engineering Research and Reports. 2019;4(3):1-20.

30. Rushdi AM, Goda AS. Symbolic reliability analysis via Shannon’s expansion and statistical independence. Microelectronics and Reliability. 1985;25(6):1041-1053.

31. Rushdi AM, Abdul Ghani AA. A comparison between reliability analyses based primarily on disjointness or statistical independence. Microelectronics and Reliability. 1993;33:965-978.

32. Rushdi AMA, Hassan AK. Reliability of migration between habitat patches with heterogeneous ecological corridors. Ecological Modeling. 2015;304:1-74.

33. Rushdi AMA, Hassan AK. An exposition of system reliability analysis with an ecological perspective. Ecological Indicators. 2016;63:282-295.

34. Rushdi AM, Rushdi MA. Switching-algebraic analysis of system reliability, Chapter 6 in M. Ram and P. Davim (Editors), Advances in Reliability and System Engineering, Management and Industrial Engineering Series, Springer International Publishing, Cham, Switzerland. 2017;139-161.

35. Rushdi AM, Alturki AM. Novel representations for a coherent threshold reliability system: A tale of eight signal flow graphs. Turkish Journal of Electrical Engineering & Computer Sciences. 2018;26(1):257-269.

36. Rushdi AM, Al-Qwasmi MA. Exposition and comparison of two kinds of a posteriori analysis of fault trees. Journal of King Abdulaziz University: Computing and Information Technology Sciences. 2016;5(1-2):55-74.

37. Rushdi AM, Hassan AK. On the Interplay Between Ecology and Reliability. Chapter 35 in Misra, KB (Editor), Handbook of Advanced Performability Engineering. Springer, Cham, Switzerland. 2020;785-809.

38. Rushdi AM, Al-Amoudi MA. Switching-algebraic analysis of multi-state system reliability. Journal of Engineering Research and Reports. 2018;3(3):1-22.

39. Rushdi AM. Utilization of symmetric switching functions in the symbolic reliability analysis of multistate k-out-of-n systems. International Journal of Mathematical, Engineering and Management Sciences. 2019;4(2):306-326.

40. Rushdi AM, Ghaleb FA. Boolean-based symbolic analysis for the reliability of coherent multi-state systems of heterogeneous components. Journal of King Abdulaziz University: Computing and Information Technology Sciences. 2020;9(2):1-25.

41. Crona Y, Hammer PL, Ibaraki T. Cause-effect relationships and partially defined Boolean functions. Annals of Operations Research. 1988;16(1):299-325.

42. Cutler RB, Muroga S. Useless prime implicants of incompletely specified multiple-output switching functions. International Journal of Computer & Information Sciences. 1980;9(4):337-350.

43. Coudert O. Two-level logic minimization: an overview. Integration. 1994;17(2):97-140.

44. Boros, E, Gurvich V, Hammer PL, Ibaraki T, Kogan A. Decomposability of partially defined Boolean functions. Discrete Applied Mathematics. 1995;62(1-3):51-75.

45. Boros E, Ibaraki T, Makino K. Error-free and best-fit extensions of partially defined Boolean functions. Information and Computation. 1998;140(2):254-283.

46. Makino K, Ibaraki T. Inner-core and outer-core functions of partially defined Boolean functions. Discrete Applied Mathematics. 1999;96:443-460.

47. Coudert O, Sasao T. Two-level logic minimization. In Logic Synthesis and Verification, Springer, Boston, MA. 2002;1-27.

48. Rushdi AMA, Ghaleb FAM. A tutorial exposition of semi-tensor products of matrices with a stress on their representation of Boolean functions. Journal of King Abdulaziz University: Computers and Information Technology Sciences. 2016;5(1-2):3-30.

49. Gräffhoff G, May M. Causal Regularities. In: Spohn W, Ledwig M, Esfeld M (Editors). Current Issues in Causation. Munster: Mentis; 2001;85–113.

Delreux T. The EU negotiates multilateral environmental agreements: explaining the
agent's discretion. Journal of European Public Policy. 2009;16(5):719-737.

51. Thiem A, Duşa A. Boolean minimization in social science research: A review of current software for Qualitative Comparative Analysis (QCA). Social Science Computer Review. 2013;31(4):505-521.

52. Sedoglavich V, Akorie ME, Pavlovich K. Measuring absorptive capacity in high-tech companies: Mixing qualitative and quantitative methods. Journal of Mixed Methods Research. 2015;9(3):252-272.

53. Thiem A. Parameters of fit and intermediate solutions in multi-value Qualitative Comparative Analysis. Quality & Quantity. 2015;49(2):657-74.

54. Halder AK. Karnaugh map extended to six or more variables. Electronics Letters. 1982;18(20):868-870.

55. Motil JM. Views of digital logic & probability via sets, numberings; 2017. Available: http://www.csun.edu/~jmotil/nums2.pdf

56. Dumont P, Bäck H. Why so few, and why so late? Green parties and the question of governmental participation. European Journal of Political Research. 2006;45: S35-S67.

57. Fischer J, Kaiser A, Rohlfing I. The push and pull of ministerial resignations in Germany, 1969–2005. West European Politics. 2006;29(4):709-735.

58. Gherghina S. The helping hand: the role of the EU in the democratization of post-communist Europe. Romanian Journal of Political Sciences. 2009;(02):65-79.

59. Jang DH. Significance of variations between income transfers and social care services development. Journal of Comparative Social Welfare. 2009;25(1):37-48.

60. Valtonen K, Padmore JC, Sogren M, Rock L. Lived experiences of vulnerability in the childhood of persons recovering from substance abuse. Journal of Social Work. 2009;9(1):39-60.

61. Avdagic S. When are concerted reforms feasible? Explaining the emergence of social pacts in Western Europe. Comparative Political Studies. 2010; 43(5):628-657.

62. Schensul JJ, Chandran D, Singh SK, Berg M, Singh S, Gupta K. The use of qualitative comparative analysis for critical event research in alcohol and HIV in Mumbai, India. AIDS and Behavior. 2010; 14(1):113-125.

63. Van der Maat E. Sleeping hegemons: Third-party intervention following territorial integrity transgressions. Journal of Peace Research. 2011;48(2):201-215.

64. Sager F, Anderegg C. Dealing with complex causality in realist synthesis: The promise of qualitative comparative analysis. American Journal of Evaluation. 2012;33(1):60-78.

65. Rushdi AMA. Karnaugh map, Encyclopedia of Mathematics. M. Hazewinkel (Editor), Boston, Kluwer Academic Publishers. 1997;I(Supplement): 327-328. Available: http://eom.springer.de/K/k110040.html

66. Fantauzzi G. Application of Karnaugh maps to Maitra cascades. In Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference. ACM. 1968;291-296.

67. Edwards CR, Hurst SL. A digital synthesis procedure under function symmetries and mapping methods. IEEE Transactions on Computers. 1978;27(11):985-997.

68. Heiss M. Error-detecting unit-distance code. IEEE Transactions on Instrumentation and Measurement. 1990;39(5):730-734.

69. Pomeranz I, Reddy SM. Pattern sensitivity: A property to guide test generation for combinational circuits. In Proceedings Eighth Asian Test Symposium (ATS’99). IEEE. 1999:75-80.

70. Tabandeh M. Application of Karnaugh map for easy generation of error correcting codes. Scientia Iranica. 2012;19(3):690-695.

71. Rushdi AM. Partially-redundant systems: Examples, reliability, and life expectancy. International Magazine on Advances in Computer Science and Telecommunications. 2010;1(1):1-13.

72. Rushdi AMA, Ghaleb FAM. The Walsh spectrum and the real transform of a switching function: A review with a Karnaugh-map perspective. Journal of Engineering and Computer Sciences, Qassim University. 2014;7(2):73-112.

73. Rushdi AMA, Ba-Rukab OM. Map calculation of the Shapley-Shubik voting powers: An example of the European Economic Community. International Journal of Mathematical, Engineering and
Rushdi AM, Ba-Rukab OM. Calculation of Banzhaf voting indices utilizing variable-entered Karnaugh maps. British Journal Mathematics and Computer Science. 2017;20(4):1-17.

Rushdi AM, Ba-Rukab OM. Translation of Weighted Voting Concepts to the Boolean Domain: The Case of the Banzhaf Index, Chapter 10 in Advances in Mathematics and Computer Science, Book Publisher International, Hooghly, West Bengal, India. 2021;2122-140.

Rushdi AM, Al-Khateeb DL. A review of methods for system reliability analysis: A Karnaugh-map perspective. Proceedings of the First Saudi Engineering Conference, Jeddah, Saudi Arabia. 1983;1:57-95.

Rushdi AM. Overall reliability analysis for computer-communication networks. Proceedings of the Seventh National Computer Conference, Riyadh, Saudi Arabia. 1984;23-38.

Rushdi AM. On reliability evaluation by network decomposition. IEEE Transactions on Reliability. 1984;33(5):379-384. Corrections: ibid. 1985;34(4):319.

Rushdi AM, Zagzoog SS. Design of a digital circuit for integer factorization via solving the inverse problem of logic. Journal of Advances in Mathematics and Computer Science. 2018;26(3):1-14.

Rushdi AM, Zagzoog SS, Balamesh AS. Design of a hardware circuit for integer factorization using a big Boolean algebra. Journal of Advances in Mathematics and Computer Science. 2018;27(1):1-25.

Rushdi AM, Zagzoog SS, Balamesh AS. Derivation of a scalable solution for the problem of factoring an n-bit integer. Journal of Advances in Mathematics and Computer Science. 2019;30(1):1-22.

Rushdi AM, Alsayegh AB. Reliability analysis of a commodity-supply multi-state system using the map method. Journal of Advances in Mathematics and Computer Science. 2019;31(2):1-17.

Rushdi AM, Zagzoog SS. Logical design of n-bit comparators: Pedagogical insight from eight-variable Karnaugh maps. Journal of Advances in Mathematics and Computer Science. 2019;32(3):1-20.

Lu Z, Liu G, Liao R. A pseudo Karnaugh mapping approach for datasets imbalance. In E3S Web of Conferences (ICERSD Management Sciences (IJMEMS). 2017;2(1):17-29.

Rushdi AM. Map derivation of the minimal sum of a switching function from that of its complement. Microelectronics and Reliability. 1985;25(6):1055-1065.

Rushdi AM. Improved variable-entered Karnaugh map procedures. Computers & Electrical Engineering. 1987;13(1):41-52.

Rushdi AM, Al-Yahya HA. A Boolean minimization procedure using the variable-entered Karnaugh map and the generalized consensus concept. International Journal of Electronics. 2000;87(7):769-794.

Rushdi AM, Al-Yahya HA. Further improved variable-entered Karnaugh map procedures for obtaining the irredundant forms of an incompletely-specified switching function. Journal of King Abdulaziz University: Engineering Sciences. 2001;13(1):111-152.

Rushdi AM. Map differentiation of switching functions. Microelectronics and Reliability. 1986;26(5):891-907.

Rushdi AM. Prime-implicant extraction with the aid of the variable-entered Karnaugh map. Umm Al-Qura University Journal: Science, Medicine and Engineering. 2001;13(1):53-74.

Rushdi AM, Al-Yahya HA. () Derivation of the complete sum of a switching function with the aid of the variable-entered Karnaugh map. Journal of King Saud University-Engineering Sciences. 2001;13(2):239-268.

Rushdi AM. Logic design of NAND (NOR) circuits by the entered-map-factorizing method. Microelectronics Reliability. 1987;27(4):693-701.

Rushdi AM, Ba-Rukab OM. A purely map procedure for two-level multiple-output logic minimization. International Journal of Computer Mathematics. 2007;84(1):1-10.

Rushdi AM, Ba-Rukab OM. A map procedure for two-level multiple-output logic minimization. In Proceedings of the Seventeenth National Computer Conference. 2004;521-532.

Rathore TS, Sanila KS. A comparison of two methods for realizing minimal function of several logic variables, SSRG International Journal of Electronics and Communication Engineering. 2021;8(1):6-11.
equation with applications in digital design. Current Journal of Applied Science and Technology. 2018;27(3):1-16.

97. Rushdi AM, Zagzoog SS. On ‘Big’ boolean-equation solving and its utility in combinatorial digital design. Chapter 3 in Advances in Applied Science and Technology. 2019;2:25-48.

98. Badawi RM. Utilization of logical map tools in Boolean analysis, qualitative comparative analysis and coincidence analysis. Unpublished Master Thesis, Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia; 2017.

99. Chan S. Explaining war termination: A Boolean analysis of causes. Journal of Peace Research. 2003;40(1):49-66.

100. Hesters D, Delreux T. Solving contradictory simplifying assumptions in QCA: presentation of a new best practice. COMPASSS Working Papers. 2010;(58):1-27.

101. Schneider CQ, Wagemann C. Reducing complexity in Qualitative Comparative Analysis (QCA): Remote and proximate factors and the consolidation of democracy. European Journal of Political Research. 2006;45(5):751-786.
Appendix

Appendix A: Basic Concepts of Boolean Minimization

This Appendix summarizes notions and concepts employed in the minimization of Boolean functions. More details are available in [16, 25, 34, 97].

The two literals of a Boolean variable \( x_m \), are its complemented form \( \bar{x}_m \), and its uncomplemented one \( x_m \). A product (conjunction) of literals is called a term \( T'(X) \) if a literal for each variable appears in it at most once, i.e., a term is an irredundant product (conjunction). A redundant product can be reduced to a term by eliminating repeated appearances of a literal through employment of idempotency of ‘AND.’ The constant 1 is the multiplication (ANDing) identity and is the product or term of no literals. The dual of a term is the irredundant sum (disjunction), called an alterm. The constant 0 is the addition (ORing) identity and is the sum or alterm of no literals. The dual of a Boolean variable \( x \) is the complement \( \bar{x} \), i.e., \( \bar{x} = 1 - x \). A term \( T'(X) \) subsumes each of the sixteen terms \( \bar{x}_m \) if \( T'(X) \) satisfies \( [T'(X) = 1] \) for every \( T'(X) \) satisfying \( [T'(X) = 1] \), while the converse is not necessarily true. A term/alterm \( T_i(X) \) is said to subsume another term/alterm \( T_j(X) \) if the set of literals of \( T_i(X) \) is a subset of that of \( T_j(X) \) (i.e., the literals of \( T_i(X) \) include those of \( T_j(X) \)).

A prime implicant \( P'(X) \) of a Boolean function \( f(X) \) is an implicant of \( f(X) \) such that no other term subsumes it if it is an implicant of \( f(X) \). An irredundant disjunctive form (IDF) \( IDF(f(x)) \) of a Boolean function \( f(X) \) is a disjunction of some of its prime implicants that expresses \( f(X) \), but ceases to do so upon the removal of one of these prime implicants. A minimal sum \( MS(f(X)) \) (minimal irredundant form \( MILF(f(x)) \) of a Boolean function \( f(X) \) is an irredundant disjunctive form for the function with the minimum number of prime implicants such that the total number of their literals is minimum.

An essential (core) prime implicant of \( f(X) \) is a prime implicant that appears in every irredundant disjunctive form for \( f(X) \). For every essential prime implicant, there exists an asserted minterm \( m(X) \) that subsumes it and does not subsume any other prime implicant. This means that the Karnaugh-map loop covering an essential prime implicant is the only loop that covers the cell of this asserted minterm. An absolutely eliminable prime implicant \( o(X) \) is a prime implicant that does not appear in any irredundant disjunctive form for \( f(X) \). A conditionally eliminable prime implicant \( f(X) \) is a prime implicant that appears in some irredundant disjunctive form(s) for \( f(X) \), but does not appear in other irredundant disjunctive form(s) for \( f(X) \).

Appendix B: The Complete Sum (Blake Canonical Form)

The Complete Sum \( CS(f(X)) \) of a Boolean function \( f(X) \) (also called its Blake Canonical Form \( BCF(f(X)) \)) is the disjunction (ORing) of all its prime implicants, and nothing else. The complete sum is a closure, unique and canonical formula for \( f(X) \). It is the minimal or absorptive special case of a syllogistic formula of \( f(X) \), where a syllogistic formula is defined as a sum-of-products formula, whose terms include, but are not necessarily confined to, all the prime implicants of \( f(X) \). Complete-sum construction usually requires the two operations of: (a) absorbing a term by another, and (b) generating the consensus of two ORed terms. A brief explanation of these operations follows.

B.1 Absorbing a Term by Another

If a term \( T_1(X) \) subsumes (implies) another \( T_2(X) \), then the disjunction \( (T_1(X) \lor T_2(X)) \) could simply be rewritten as \( T_2(X) \), viz.

\[
T_1(X) \lor T_2(X) = T_2(X).
\]  (B.1)

The deletion of \( T_1(X) \) in (B.1) is called absorption of the subsuming term \( T_1(X) \) in the subsumed term \( T_2(X) \). For example, the term \( XYZW \) subsumes each of the sixteen terms \( XZW, YZW, XZW, XYW, XYZ, ZW, YW, XW, YZ, Z, XY, W, Z, Y, X, and 1 \). Hence, it could be deleted if it
is ORed with any of them. The complete sum is an absorptive syllogistic formula, i.e., it is a syllogistic formula in which no term subsumes another.

B.2 Generating the Consensus of Two ORed Terms

Two terms \(T_1(X)\) and \(T_2(X)\) have a consensus if and only if they have exactly one opposition, i.e., exactly one variable that appears complemented \((\bar{X}_m)\) in one term (say \(T_1(X)\)) and appears uncomplemented \((X_m)\) in the other term. In such a case, the consensus is the ANDing of the remaining literals of the two terms, i.e.

\[
\text{consen} \ xT_1(X), T_2(X) = (T_1(X) / \bar{X}_m) \ (T_2(X) / X_m),
\]

where \((f/t)\) denotes the Boolean quotient of the function \(f\) by the term \(t\), i.e., the restriction of \(f\) when \(t\) is asserted [34, 39], viz.

\[
f/t = [f]_{t=1}.
\]

When two terms have a consensus, their disjunction can be augmented by this consensus, i.e.

\[
T_1(X) \lor T_2(X) = T_1(X) \lor T_2(X) \lor \text{consen} \ xT_1(X), T_2(X).
\]

For example, the terms \(A\bar{B}\) and \(BC\) have a single opposition and are represented on the Karnaugh map by two disjoint loops sharing a border, and hence their disjunction can be augmented by their consensus \((A\bar{B} / \bar{B}) \ (BC / B) = AC\), which is a loop extending across the common border between the original loops and covering the part \(A\bar{B}\) of \(A\bar{B}\) and the part \(ABC\) of \(BC\). By contrast, the two terms \(A\) and \(BC\) have zero opposition, and consequently non-disjoint or overlapping loops, and possess zero or no consensus. The two terms \(A\bar{B}\) and \(\bar{A}\bar{B}\) have more than one opposition, and consequently disjoint far-away loops, and again possess zero or no consensus [25].

The complete-sum formula \(CS(f)\) may be generated by a two-step iterative-consensus procedure, in which (a) a syllogistic formula \(F\) for \(f(X)\) is found by continually comparing terms and adding their consensuses (if any) to the current formula of \(f(X)\), and (b) the resulting formula is converted to an absorptive one \(\text{ABS}(F)\), again by continually comparing terms and deleting subsuming terms by absorbing them in their subsumed terms. A streamlined algorithmic version of iterative consensus is the Blake-Tison Method, which produces the complete sum by performing explicit consensus generation with respect to each bi-form variable, and following this by absorption. Alternatively, a syllogistic formula for the function might be produced (without explicit consensus generation) through multiplying out any suitable product-of-sums (pos) expression for the function to produce a sum-of-products (sop) expression.

Appendix C: Probability-Ready Expressions

A Probability-Ready Expression [34, 39] is a random expression that can be directly transformed, on a one-to-one basis, to its statistical expectation (its probability of being equal to 1) by replacing all logic variables by their statistical expectations, and also replacing logical multiplication and addition (ANDing and ORing) by their arithmetic counterparts. A logic expression is a \(PRE\) if

a) all ORed terms are disjoint (mutually exclusive), and
b) all ANDed sums (alterms) are statistically independent.
Appendix D: The Boole-Shannon Expansion

An effective way for converting a Boolean formula into a PRE form is to (repeatedly) employ the Boole-Shannon Expansion \[34, 39\], which takes the following form when implemented w.r.t. a single variable \(X_k\):

\[ f(X) = (\bar{X}_k \land f(X|0_k)) \lor (X_k \land f(X|1_k)). \]  

(\text{D.1})

This Boole-Shannon Expansion expresses the Boolean function \(f(X)\) in terms of its two subfunctions \(f(X|0_k)\) and \(f(X|1_k)\). These subfunctions are equal to the Boolean quotients \(f(X)/\bar{X}_k\) and \(f(X)/X_k\), and hence are obtained by restricting \(X_k\) in the expression \(f(X)\) to 0 and 1, respectively. If \(f(X)\) is a sop expression of \(n\) variables, the two subfunctions \(f(X|0_k)\) and \(f(X|1_k)\) are functions of at most \((n-1)\) variables. If the Boole-Shannon expansion is applied in sequence to the \(n\) variables of \(f(X)\), the expansion tree is a complete binary tree (usually called a Binary Decision Diagram) of \(2^n\) leaves. These leaves are functions of no variables, or constants, and equal the entries of a corresponding conventional Karnaugh map of \(f(X)\). Sibling nodes (nodes at the same level) of this expansion tree constitute the entries of a variable-entered (or a map-entered) Karnaugh map of \(f(X)\).

Appendix E: Unate Boolean Functions

A Boolean function \(f(X) = f(X_1, X_2, ..., X_{k-1}, X_k, X_{k+1}, ..., X_n)\) is called unate if and only if it can be represented as a normal (sum-of-products or product-of-sums) formula in which no variable appears both complemented and un-complemented, i.e., every variable is mono-form and no variable is bi-form. This Boolean function is called positive in its argument \(X_k\), if there exists a normal representation of the function in which \(X_k\) does not appear complemented. This happens if and only if every occurrence of the literal \(X_k\) is redundant and can be eliminated, i.e., if and only if there exist functions \(f_1\) and \(f_2\) (independent of \(X_k\)) such that \[80-86\]

\[ f(X) = X_k f_1(X_1, X_2, ..., X_{k-1}, X_{k+1}, ..., X_n) \lor f_2(X_1, X_2, ..., X_{k-1}, X_{k+1}, ..., X_n). \] 

(E.1)

A Boolean function \(f(X)\) is called negative in its argument \(X_k\), if there exists a normal representation of the function in which \(X_k\) does not appear un-complemented. This happens if and only if every occurrence of the literal \(\bar{X}_k\) is redundant and can be eliminated, i.e., if and only if there exist functions \(f_3\) and \(f_4\) (independent of \(X_k\)) such that

\[ f(X) = f_3(X_1, X_2, ..., X_{k-1}, X_{k+1}, ..., X_n) \lor \bar{X}_k f_4(X_1, X_2, ..., X_{k-1}, X_{k+1}, ..., X_n). \] 

(E.2)

If the function \(f(X)\) is positive in its argument \(X_k\), then its subfunctions are \(f(X|1_k) = f(X)/X_k = f_1 \lor f_2\) and \(f(X|0_k) = f(X)/\bar{X}_k = f_2\), which means that \(f(X|0_k) \leq f(X|1_k)\). Similarly, if the function \(f(X)\) is negative in its argument \(X_k\), then its subfunctions are \(f(X|1_k) = f(X)/X_k = f_3\) and \(f(X|0_k) = f(X)/\bar{X}_k = f_3 \lor f_4\), which means that \(f(X|1_k) \leq f(X|0_k)\).

All threshold (linearly-separable) functions are unate, but the converse is not true. The function \(X_1 X_2 \lor \bar{X}_3 X_4\) is an example of a unate function that is not threshold. All the prime implicants of a unate function are essential, so that it has a single irredundant disjunctive form, which serves as both its (unique) minimal sum and its complete sum.

© 2021 Rushdi and Badawi. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/68072