

Prime and Essential Prime Implicants of Boolean Functions through Cubical Representation

Saurabh Rawat

Anushree Sah

ABSTRACT

K Maps are generally and ideally, thought to be simplest form for obtaining solution of Boolean equations. Cubical Representation of Boolean equations is an alternate pick to incur a solution, otherwise to be meted out with Truth Tables, Boolean Laws and different traits of Karnaugh Maps. Largest possible k-cubes that exist for a given function are equivalent to its prime implicants. A technique of minimization of Logic functions is tried to be achieved through cubical methods. The main purpose is to make aware and utilise the advantages of cubical techniques in minimization of Logic functions. All this is done with an aim to achieve minimal cost solution.

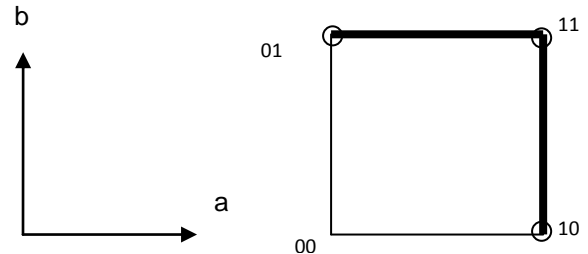
Keywords

K maps, Boolean equations, cubes.

1. INTRODUCTION

Karnaugh maps can be used to good effect for illustrating and explaining concepts, if functions have only few variables and in finding optimal implementation of logic functions. Logic functions can be represented through four different forms – truth table, algebraic expressions, venn diagrams and karnaugh maps. Algebraic rather than graphical techniques can be applied to deal with larger functions of any number of variables. Karnaugh map scheme for representing logic functions is not appropriate for use in CAD tools, so we will be focusing on an alternative representation (cubical) of logic functions which is suitable for use in CAD algorithms. Cubical representation of logic function is the approach to algebraic optimization technique; we will be putting forward and contemplating in this paper, mapping a function of n variables onto n dimensional cube.

Two dimensional cube having four corners are called vertices, which correspond to the four rows of a truth table. Each vertex is identified by two co-ordinates. Horizontal coordinate is assumed to correspond to variable a and vertical coordinate to b. Thus vertex 00 is bottom left corner, which corresponds to row 0 in the truth table. Vertex 01 is top left corner where a=0 and b=1, which corresponds to row 1 in the truth table and so on for the other two vertices.



a	b	f
0	0	0
0	1	1
1	0	1
1	1	1

1.1 Representation of $f(a,b) = \sum m(1,2,3)$

Mapping a function onto the cube by indicating with circles those vertices for which $f=1$.

$f=1$ for vertices 01, 10 and 11.

Expressing the function as a set of vertices using the notation $f = \{01, 10, 11\}$. Function f is also shown in the form of a truth table in the figure 1.1.

An edge joins two vertices for which the label differs in the value of only one variable. Therefore, if two vertices for which $f=1$ are joined by an edge, then this edge represents that portion of the function, as two individual vertices. For example, $f=1$ for vertices 10 and 11. They are joined by the edge that is labeled $1x$. x denotes the fact that corresponding variables can be either 0 or 1 so $1x$ means that $a=1$, while b can be either 0 or 1. Similarly, vertices 01 and 11 are joined by the edges labeled $x1$, indicating that a can be either 0 or 1, but $b=1$. Two vertices being represented by a single edge is embodiment of combining property. Edge $1x$ is the logical sum of vertices 10 and 11. It defines the term a , which is sum of minterms $a\bar{b} + ab$

$$1x = 10 + 11 = 1$$

$$\bar{a}\bar{b} + ab = a$$

Finding edges for which $f=1$ is equivalent to applying the combining property ,

so $f=\{01,10,11\}$ can be represented as $f=\{1x,x1\}$.

This corresponds to logic expression $f=a+b$

b

0	1
1	1

In three dimensional cube, each vertex is identified by a specific valuation of three variables function. Function $f(a,b,c)=\sum m(0,2,4,5,6)$ is mapped onto the cube. There are five vertices for which $f=1$ namely 000, 010, 100, 101 and 110. These vertices are joined by five edges, namely $x00$, $0x0$, $x10$, $1x0$, and $10x$. Because the vertices 000, 010, 100, & 110 include all valuation of a and b when c is zero, they can be specified by term $xx0$.

So $f=1$ if $c=0$ regardless of the values of a and b .

$xx0$ represent front side of the cube, f can be represented in several ways so

$$f=\{000,010,100,101,110\}$$

$$f=\{0x0,1x0,101\}$$

$$f=\{x00, x10, 101\}$$

$$f=\{x00, x10,10x\}$$

$$f=\{xx0,10x\}$$

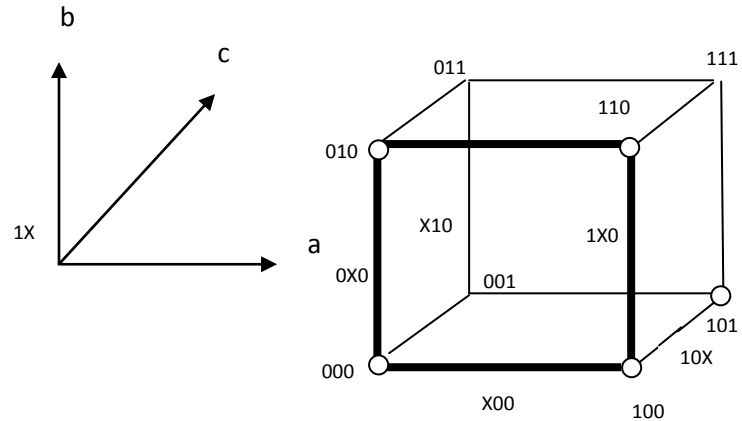
$f=\{xx0,10x\}$ is equivalent to logic expression

$$f = \bar{c} + a\bar{b}$$

$$f(a,b,c)=\sum m(0,2,4,5,6)$$

1	0	0	1
1	1	0	1

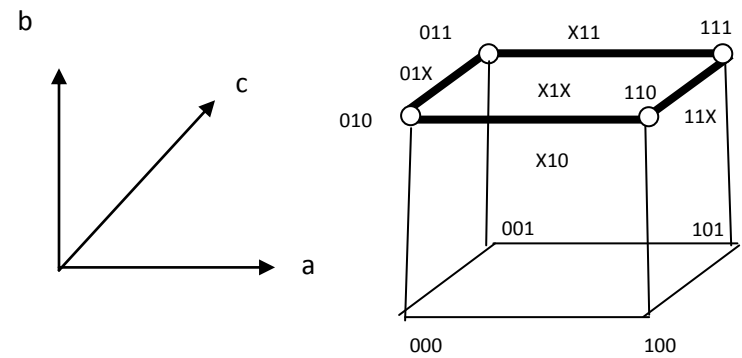
$$f = \bar{c} + a\bar{b}$$



1.2 Representation of $f(a,b,c)=\sum m(0,2,4,5,6)$

Picking up another example

$$f(a,b,c)=\sum m(2,3,7)+d(6)$$



Representation of $f(a,b,c)=\sum m(2,3,7)+d(6)$

$f=1$ for 011, 010, 111 and 110

$$f=\{010,011,110,111\}$$

$$f=\{01x,11x\}$$

$$f=\{x1x\}$$

This corresponds to logic expression

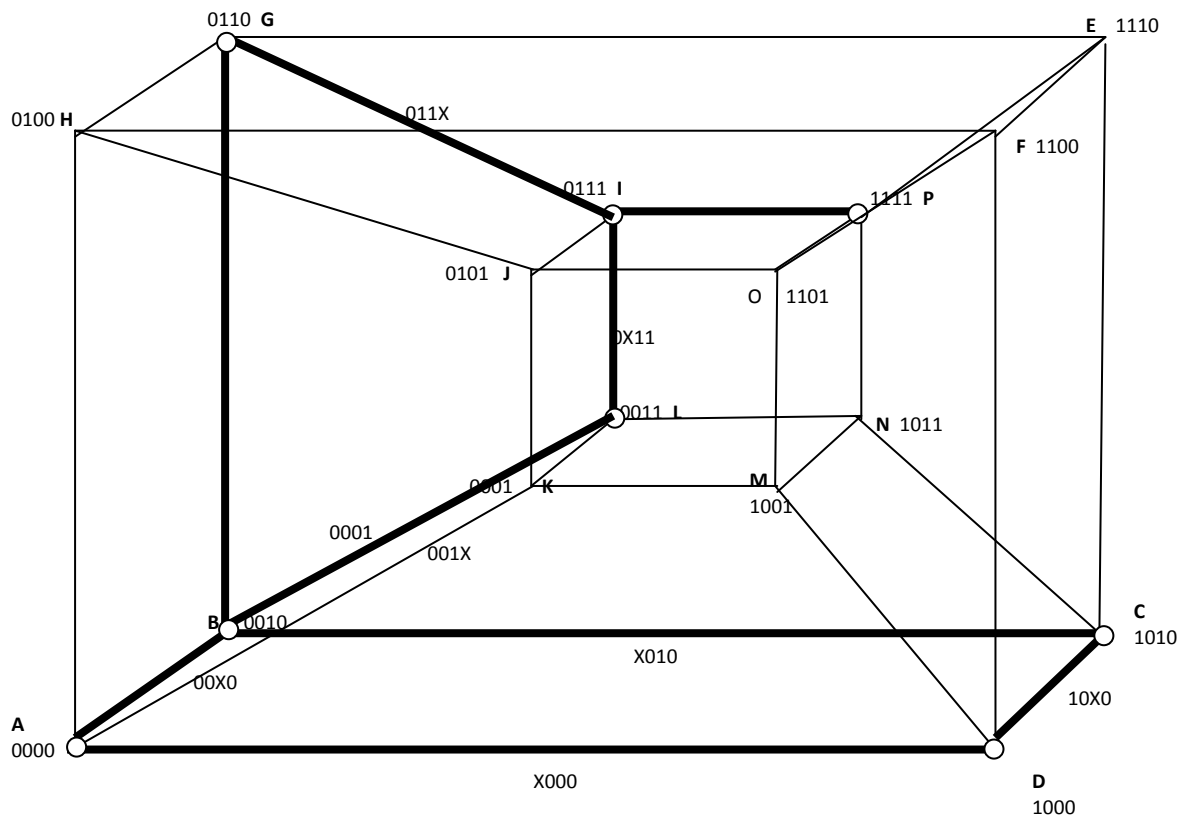
$$f = b$$

	$\bar{b}\bar{c}$	$\bar{b}c$	$b\bar{c}$	bc
a	0	0	1	1
a	0	0	1	X

Graphical images of two and three dimensional cubes are easy to draw. A four dimensional cube is more difficult. It consists of 2 three dimensional cubes with their corners connected. Simplest way to visualize a four dimensional cube is to have one cube placed inside the other cube. Assuming that a, b and c coordinates are same as in three dimensional cubes while d=0 defines outer cube and d=1 defines the inner cube.

Considering the case

$$f(a,b,c,d)=\sum m(0,2,3,6,7,8,10,15)$$



Representation of function

$$f(a,b,c,d)=\sum m(0,2,3,6,7,8,10,15)$$

$$f = \{x_0x_0, 0x_1x, x_111\}.$$

There are two groups of four adjacent vertices for which $f=1$, that can be represented as planes

ABCD and BLIG.

Group comprising 0000,0010,1000 and 1010 is represented by x0x0.

Group 0010, 0011, 0110 & 0111 is represented by 0x1x .

function f can be represented in several ways

$$f = \{0000, 0010, 0011, 0110, 0111, 1000, 1010, 1111\}.$$

$$f = \{00x0, 10x0, 0x10, 0x11, x111\}.$$

Simplest circuit is obtained if $f = \{x_0x_0, 0x_1x, x_111\}$ which is equivalent to the expression

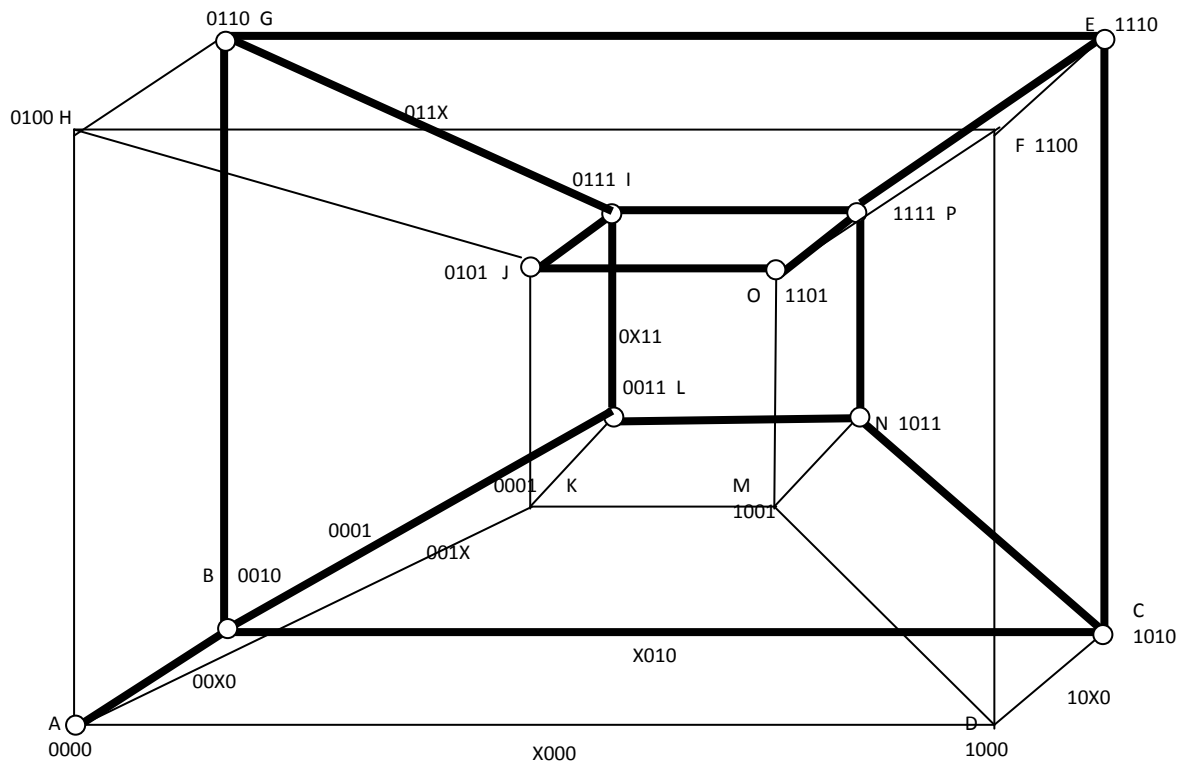
$$f = \overline{\overline{b}}\overline{\overline{d}} + \overline{\overline{a}}c + bcd$$

	$\bar{c} \bar{d}$	$\bar{c} d$	$c d$	$c \bar{d}$
$\bar{a} \bar{b}$	1	0	1	1
$\bar{a} b$	0	0	1	1
$a \bar{b}$	0	0	1	0
$a b$	1	0	0	1

$$f = \bar{b}\bar{d} + \bar{a}c + bcd$$

$$f(a,b,c,d) = \sum m\{0,2,3,5,6,7,13,14\} + d\{10,11,15\}$$

Taking up another example $f(a,b,c,d) = \sum m(0,2,3,5,6,7,13,14) + d(10,11,15)$



Two groups, one with eight adjacent vertices 0010,0011,1011,1010,0110,0111,1110,1111 and others with four adjacent vertices 0101, 0111, 1101, 1111 and further left with 0000 and 0010.
First group represented by $xx10$ and second group represented by $x1x1$.
Function f can be represented in several ways

$$f = \{0000, 0010, 0011, 0101, 0110, 0111, 1010, 1011, 1101, 1110, 1111\}$$

$$f = \{(0000, 0010), 0011, (0101, 0111, 1101, 1111), (0110, 1110), (1010, 1011)\}$$

$$f = \{00x0, 0011, x1x1, x110, 101x\}$$

$$f = \{00x0, x1x1, 0011, x110, 101x\}$$

$$f = \{00x0, x1x1, xx1x\}$$

simplest circuit is obtained if $f = \{00x0, x1x1, xx1x\}$,

which is equivalent to expression

$$f = \overline{abd} + bd + c$$

	$\overline{c} \overline{d}$	$\overline{c} d$	$c d$	$c \overline{d}$
$\overline{\overline{a}} \overline{b}$	1	0	1	1
$\overline{a} b$	0	1	1	1
$a b$	0	1	x	1
$\overline{a} b$	0	0	x	x

$$f = \overline{abd} + bd + c$$

n-dimensional cube

A function that has n variables can be mapped onto n-dimensional cube. Although it is impractical to draw graphical images of cubes that have more than more variables, it is not difficult to extent the ideas to general n variable case.

2. CONCLUSION

This paper presents an efficient manual method (cubical) for obtaining the most compact form of the subsumptive general solution of a system of Boolean equations. The method utilizes effective combination of mapping and algebraic methods and employs a minimum number of constructions .

Throughout its work, the method keeps track of one and many variable representation that leads to an immediate construction of the minimal sum-of-product expressions for the Boolean functions. The concepts and method developed herein can be utilized in various application areas of Boolean equations. The proposed outlook is based on the looping of Boolean terms. Therefore in order to take a closer look at how to loop two, four or eight 1's to get the least possible number of groups in a K-map table setting. Boolean algebra, Karnaugh maps, and CAD (Computer Aided Design) are methods of logic simplification. The goal of logic simplification is a minimal cost solution. A minimal cost solution is a valid logic reduction with the minimum number of gates with the minimum number of inputs. This paper utilized and focused on cubical representation of Boolean functions to achieve minimal cost solution.

3. REFERENCES

- [1] Fundamental of Digital Logic with Verilog Design by Stephen Brown and Zvonko Vranesic, tata mcgraw,2007,
- [2] Mano, Morris M. (1991), Digital Design, Second Edition, Prentice-Hall of India Private Limited.
- [3] Rajaraman, V., Radhakrishnan, T. (2001), An Introduction to Digital Computer Design, Fourth Edition, Prentice-Hall of India Private Limited.
- [4] Tanenbaum, A. S. (2004), Structured Computer Organization, Fourth Edition, Prentice-Hall of India Private Limited.
- [5] Leach, D. P, Malvino, A. P. and Saha, G. (2006), Digital Principles and Applications, Second Edition, Tata McGraw-Hill.
- [6] Wakerly, John F. (2005), Digital Design, Principles and Practices, Third Edition Updated, Ninth Indian Reprint, Pearson Education Inc.
- [7] Crenshaw, Jack W. (2003), A primer on Karnaugh maps, Embedded Systems Design, Retrieved from http://www.embedded.com/columns/programmerstoolbox/16100908?_requestid=264392
- [8] Kuphaldt, Tony R. (2007), Lessons in Electric Circuits, Volume IV – Digital, Fourth Edition, Available as part of the Open Book Project collection retrieved from: www.ibiblio.org/obp/electricCircuits
- [9] E. J. McCluskey, "Minimization of Boolean Functions," Bell System Technical Journal, vol. 35, no. 5, pp. 1417-1444, 1956.
- [10] Quine, W.V. (1952). "The Problem of Simplifying Truth Functions". The American Mathematical Monthly, Vol. 59, No. 8.
- [11] T. Sasao, "Worst and Best Irredundant Sum-of-Product Expressions", IEEE Transactions on Computers, 50(9) (2001),pp. 935–947
- [12] A. Mishchenko and T. Sasao, "Large-Scale SOP minimization Using Decomposition and Functional Properties", DAC 2003, June 2-6, pp. 149–154.
- [13] P.P. Tirumalai and J.T. Butler, "Minimization Algorithms for Multiple-Valued Programmable Logic Arrays", IEEE Transactions on Computers, (1991), pp. 167–177.
- [14] Y. Saad and M.H. Schultz, "Topological Properties of Hypercubes", IEEE Trans. on Comput., (1988), pp. 867–872.
- [15] E.W. Veitch, "A Chart Method for Simplifying Truth Functions", Proc. of the ACM, 1952, pp. 127–133.

- [16] M. Karnaugh, “A Map Method for Synthesis of Combinational Logic Circuits”, Trans. AIEE, Comm. and Electronics, (1953), pp. 593–599.
- [17] M.R. Dagenais, V. K. Agarwal, and N. C. Rumin, “McBoole: A New Procedure for Exact Logic Minimization”, IEEE Trans. on Computer-Aided Design, (1986), pp. 229–238.
- [18] R.L. Rudell and A. Sangiovanni-Vincentelli, “Multiple-Valued Minimization for PLA Optimization”, IEEE Trans. On Computer-Aided Design, (1987), pp. 727–750.
- [19] R.K. Brayton, R. Rudell, A. Sangiovanni-Vincentelli, and A.R. Wang, “MIS: A Multiple-Level Logic Optimization Systems”, IEEE Trans. on Computer-Aided Design, (1987), pp. 1062–1081.
- [20] K.A. Bartlett et al., “Multilevel Logic Minimization Using Implicit Don’t Cares”, IEEE Trans. on Computer-Aided Design, (1988), pp.723–740. Şirzat Kahramanli, Salih Güneş, Seral Şahan, and Fatih Başçiftçi, The Arabian Journal for Science and Engineering, Volume 32, Number 1B April 2007 114
- [21] H. Savoj, A.A. Malik, and R.K. Brayton, “Fast Two-Level logic Minimizers for Multi-Level Logic Synthesis”, IEEE Int. Conf. on Computer Aided Design, 1989, pp. 426–429.
- [22] R.K. Brayton and F. Somenzi, “Minimization of Boolean Relations”, IEEE Int. Symp. on