

MINIMALIZATION OF BOOLEAN POLYNOMIALS,  
TRUTH FUNCTIONS, AND LATTICES

MITCHELL O. LOCKS

1 *Introduction* In 1952 W. V. Quine [1] published a landmark paper on the minimalization of a Boolean polynomial or truth function. First, the polynomial is fully expanded out to the "developed form", and then one finds the "prime implicants" by employing a tabular representation to find common indicators. In 1956 E. J. McCluskey published a sequel to Quine's paper [2], in which a decimal coding scheme is employed on the binary-number equivalents of the terms of the expanded polynomial, to facilitate the simplification process. The Quine procedure, or Quine-McCluskey procedure as it is sometimes referred to (see Phister [3] or Korfhage [4]), is used in the logical design of digital computers and communications equipment, to help reduce the number of logical circuits required. The purpose of this paper is to develop a set-theoretic explanation for the process of minimalization of a Boolean polynomial, with an algorithm for minimalization which is simpler than that of Quine-McCluskey, in that a full expansion of the polynomial and a tabular representation is not necessary. Thus, the procedure is carried out algebraically, in a way which is amenable to processing entirely by a digital computer. The procedure also includes a test to determine whether a particular form is minimal.

A Boolean polynomial for a system with  $n$  binary variables represents a lattice within the universal set of  $2^n$  binary  $n$ -tuple elements. Each term (frequently called "minterm" of this polynomial is a monomial which represents a "complete subset (sublattice)" of  $n$ -tuple elements identified by  $m$  common-valued binary variables,  $m < n$ , which are used as indicators for the subset.<sup>1</sup> The objective in minimalization is to reduce the size of the polynomial, as a description for the lattice, so that it has both the smallest number of terms and the smallest average number of indicators per term. This is achieved when you find the smallest number of complete sublattices which cover the entire function. This is essentially the same thing as

*Received January 11, 1974*

*Revised version received June 9, 1974*

having the "largest" complete sublattices. To take an extreme example, the universal set, which has  $2^n$   $n$ -tuple elements, is a free Boolean algebra with no fixed-valued indicators for any of the binary components, hence there is no minimal polynomial for it.<sup>2</sup> In other words, the larger the sublattices are, the fewer the number of indicators per term, and the smaller the number of terms used to represent all the  $n$ -tuples in the lattice. The results achieved this way are identical to Quine's "prime implicants."

The essential difference between our minimalization algorithm and that of Quine is that the latter requires first an expanded tabular representation of all the  $n$ -tuples in the lattice, and then a simplification. In our procedure, the initial expansion is unnecessary, as we proceed directly to minimalize from any polynomial, by merging smaller sublattices (i.e., with more indicators) *into* larger ones (with fewer indicators).<sup>3</sup> Three types of pairwise comparison operations are performed at each iteration:

- (i) Merging partitioned subsets of the same size which differ by only the value of a single indicator: example,  $ab \vee a\bar{b} = a$  (it is understood that  $\bar{b}$  is the complement of  $b$ ).
- (ii) Eliminating redundant terms: for example, " $ab \vee a = a$ ", since " $ab$ " is a subset of " $a$ ".
- (iii) Eliminating redundant indicators: for example, " $a\bar{b} \vee b = a \vee b$ ", also " $a \vee a\bar{b} = a \vee \bar{b}$ ".

The three operations are performed successively at each iteration, in any permutation, until no further apparent simplification is possible with these operations. A "reversal test" is then used to determine whether alternative or better minimal forms can be obtained. In this test, we increase the number of indicators in the largest terms by partitioning, and also, if necessary, the number of terms. If the only possible way to minimalize the resulting expanded polynomial is the minimal form previously attained, it is unique. If a different minimal polynomial is obtained of exactly the same size, both are equally good. If a smaller polynomial results, however, the process continues until a decision is reached.

## 2 Terminology and definitions<sup>4</sup>

**2.1 Systems** A system which is represented by a Boolean polynomial is generated by a set of specifications, propositions, statements, or states which specify the interrelationships between and among  $n$  binary-valued variables or components. For each component, the value "1" denotes that the binary characteristic or signal corresponding to that particular component is present, and the value "0" that the characteristic is absent.

**2.2 Components, algebraic operations, and bounds** Since there are  $n$  components, a state of the system is a binary  $n$ -tuple; because the components are binary, there are  $2^n$  states. The collection of these  $2^n$  states is the universal set  $\mathbf{u}$ , a *free Boolean algebra* with a unique zero element  $(0, \dots, 0)$ , and a unique one element  $(1, \dots, 1)$ , and which is

subject to the usual component-by-component Boolean vector operations of addition ( $0 + 0 = 0$ ,  $1 + 0 = 1 + 1 = 1$ ), multiplication ( $0 \cdot 0 = 1 \cdot 0 = 0$ ,  $1 \cdot 1 = 1$ ) and complementation ( $\bar{0} = 1$ ,  $\bar{1} = 0$ ). Addition or multiplication of  $n$ -tuples is performed either pairwise, denoted by “+” or “ $\cdot$ ”, or in sequences, denoted by “ $\Sigma$ ” and “ $\Pi$ ”. An  $n$ -tuple which is the Boolean sum of two or more elements of  $\mathbf{U}$  is also their least upper bound, l.u.b., and the Boolean product their greatest lower bound, g.l.b. The zero and unit elements are respectively lower and upper bounds for all subsets of  $\mathbf{U}$ , but not necessarily the “greatest” lower bound or the “least” upper bound.

**2.3 Lattices and sublattices** A lattice  $\mathfrak{Q}$  is a nonempty subset of  $\mathbf{U}$ , for which every pair of elements has both a g.l.b. and an l.u.b. Since the Boolean sum of every pair or sequence of elements in  $\mathbf{U}$  is the l.u.b. and the Boolean product the g.l.b.,  $\mathbf{U}$  and every one of its subsets is also a lattice. A *closed* lattice contains both its g.l.b. and l.u.b.  $\mathfrak{Q}$  is *complete* if it contains both the g.l.b. and the l.u.b. of every pair of elements in  $\mathfrak{Q}$ . Following Rutherford ([10], p. 9), a complete lattice, that is, any complete subset of  $\mathbf{U}$ , is referred to herein as a “sublattice”. In Reference [5], it is shown that there are  $3^n$  sublattices in  $\mathbf{U}$ .

**2.4 Boolean polynomials** The Boolean polynomial for a sublattice is just a single term (monomial) consisting of  $m$  common-valued binary indicators,  $m < n$ , all of which have the same value in every one of the  $2^{n-m}$  states in the set. This term, sometimes called the “meet”, is frequently called a “minterm”. A lattice  $\mathfrak{Q}$ , such as a “truth function”, is an “inclusive or” union of sublattices. The Boolean polynomial for  $\mathfrak{Q}$  has a term for each sublattice, “joined” by the logical symbol “ $\vee$ ”. In general, the configuration of subsets is not unique, because there can be a very large combinatorial number of alternative representations. A minimal form, however, may be and frequently is unique. Even if the minimal form is not unique, the number of alternative minimal representations is usually limited.

**3 Minimalization of Boolean polynomials** Minimalizing a Boolean polynomial for  $\mathfrak{Q}$  consists of finding the smallest number of largest sublattices, none of them proper subsets of one another, which completely cover  $\mathfrak{Q}$ . The procedure is an iterative one in which smaller sublattices are merged into larger ones in stages, by a series of pairwise comparisons of the indicators. There are three possible types of steps at each iteration:

- (i) merging partitioned sublattices of the same size which differ by only the value of a single indicator,
- (ii) eliminating redundant terms,
- (iii) eliminating redundant indicators.

If operations (i), (ii), and (iii) cannot be performed any further at the completion of a given iteration, a “reversal test” is performed to determine whether any further simplification is possible.

**3.1 Merging partitioned subsets** For a system having only a single binary

variable,  $a$ , let  $a$  denote the unit value and  $\bar{a}$  the zero value. In this case, the universal set is

$$a \vee \bar{a} = \mathbf{1}.$$

For a system having two variables,  $a$  and  $b$ , the join of partitioned sublattices  $ab$  and  $\bar{a}b$  is the sublattice  $b$ ;

$$ab \vee \bar{a}b = b. \quad (1)$$

Equation (1) is a prototype of this merging operation. For example,

$$abc\bar{d}\bar{e} \vee \bar{a}b\bar{c}d\bar{e} = b\bar{c}d\bar{e}.$$

**3.2 Eliminating redundant terms** When two terms differ only in that one is longer than the other, the longer term denotes a subset of the shorter one. For example,  $abc$  is a subset of  $a$ , or  $b$ , or  $c$ , or  $ab$ , or  $ac$  or  $bc$ . Thus, we have

$$\begin{aligned} abc \vee a &= a \\ \bar{a}\bar{d}ef \vee aef &= aef \\ \bar{a}bc \vee \bar{a}b &= \bar{a}b. \end{aligned}$$

**3.3 Eliminating redundant indicators**<sup>5</sup> This operation is a combination of the other two types of simplification. It involves reducing by one the number of indicators in a longer term which represents a sublattice for which there is a complementary sublattice which is a subset of a shorter term. This type of simplification is performed when the longer term has the same indicators as the shorter term, and the same values of the indicators, except for one, which is complemented; this indicator is the one which is deleted in the longer term. Consider the example given in the introduction " $\bar{a}b \vee b = a \vee b$ ". This is proved as follows

$$\bar{a}b \vee b = \bar{a}b \vee ab \vee \bar{a}b = a \vee b.$$

Likewise,

$$a \vee \bar{a}b = ab \vee \bar{a}b \vee \bar{a}b = a \vee \bar{b}.$$

Also,

$$\begin{aligned} \bar{a}b\bar{c}d\bar{e} \vee ab &= a\bar{c}d\bar{e} \vee ab, \\ b \vee \bar{a}bc &= b \vee ac. \end{aligned}$$

**3.4 The reversal test** Reversal is a kind of "perturbation", a temporary expansion of the size of the polynomial, in order to determine whether further simplification is possible. It is performed term by term, largest terms first, whenever one of the three simplification operations cannot be used. Either of three results is obtained:

- (i) The original minimal polynomial is the only possible simplification, and is therefore unique.
- (ii) An alternative minimal form is obtained with the same number of terms and the same average number of indicators per term. In this case either the original form or the alternative is equally good.

(iii) A smaller polynomial results; therefore, simplification continues until a final decision is reached.

An example of proving uniqueness by reversal is  $\bar{a}b \vee b\bar{c}$ . Upon expanding the first term, we obtain  $\bar{a}bc \vee b\bar{c}$ . Simplifying, we have  $\bar{a}b \vee b\bar{c}$ ; therefore the polynomial is unique. An example of further simplification by reversal is adapted from Korfhage [4], p. 30:

$$x_1\bar{x}_3 \vee x_1x_2 \vee \bar{x}_2x_3 = x_1\bar{x}_2\bar{x}_3 \vee x_1x_2 \vee \bar{x}_2x_3 = x_1\bar{x}_2 \vee x_1x_2 \vee \bar{x}_2x_3 = x_1 \vee \bar{x}_2x_3,$$

which is unique. An example of obtaining an alternative solution by reversal is adapted from Example 3-5c on p. 32 of Phister [11]

$$bc \vee a\bar{c} \vee \bar{a}c \vee \bar{b}c\bar{d} = bc \vee a\bar{c} \vee \bar{a}c \vee \bar{a}b\bar{c}\bar{d} = bc \vee a\bar{c} \vee \bar{a}c \vee \bar{a}b\bar{d}.$$

**4 Examples** The following examples below of the simplification process are all adapted from Quine's 1952 paper [1] on the minimization of Boolean polynomials. Explanations are included where necessary.

Example 1 (from Quine [1], p. 525):

$$\begin{aligned} pqr s \vee \bar{p}\bar{q}rs \vee pq\bar{r}s \vee p\bar{q}\bar{r}s \vee pq\bar{r}\bar{s} \vee p\bar{q}\bar{r}\bar{s} \\ = pqs \vee \bar{p}\bar{q}rs \vee p\bar{q}\bar{r}s \vee p\bar{r}\bar{s} \\ = pqs \vee \bar{p}\bar{q}rs \vee p\bar{r}\bar{s} \vee p\bar{r}\bar{s} \\ = pqs \vee \bar{p}\bar{q}rs \vee p\bar{r}. \end{aligned} \quad (2)$$

Equation (2) is the unique minimal form. This can be ascertained by reversal, expanding the largest free term "pqs" into  $pq\bar{r}s \vee pqr s$  and trying to simplify by some way other than merging these two back together again. Since this cannot be done, the polynomial is unique.

Example 2 (identical with Example 2 from p. 523 of Quine [1]):

$$pq \vee p\bar{q}r \vee \bar{p}\bar{q}\bar{r} = pq \vee pr \vee \bar{p}\bar{q}\bar{r}, \text{ since } pq \vee p\bar{q}r = pq \vee pr.$$

Quine obtained this result by proving through a logical argument that the first appearance of  $\bar{q}$  in this expression is superfluous; we obtained it by a direct application of the third rule of simplification.

Example 3 (identical with Example 1 from p. 523 of Quine [1]). In order to simplify " $pq \vee p\bar{r} \vee \bar{q}\bar{r}$ ", a reversal is first necessary, because none of the three types of simplification can be used, therefore

$$pqr \vee pq\bar{r} \vee p\bar{r} \vee \bar{q}\bar{r} = pqr \vee p\bar{r} \vee \bar{q}\bar{r}.$$

A simplification at this point would result in the original polynomial again, therefore it is necessary to continue reversing until a decision can be made. By expanding  $p\bar{r}$  we obtain

$$pqr \vee pq\bar{r} \vee p\bar{q}\bar{r} \vee \bar{q}\bar{r} = pq \vee \bar{q}\bar{r},$$

which is the unique minimal form. Quine obtained this result by proving through a logical argument that the term " $p\bar{r}$ " is superfluous.

Example 4 (from page 526 of Quine [1]). In order to simplify  $p\bar{q} \vee \bar{p}q \vee q\bar{r} \vee \bar{q}r$ , a reversal is first necessary. By expanding  $p\bar{q}$  we obtain

$p\bar{q}\bar{r} \vee \bar{p}q \vee q\bar{r} \vee \bar{q}r$ . Since a simplification at this point would result in the original polynomial, the reversing is continued. By expanding  $\bar{p}q$  into  $\bar{p}qr$  and  $\bar{p}q\bar{r}$  we obtain  $p\bar{q}\bar{r} \vee \bar{p}qr \vee q\bar{r} \vee \bar{q}r$ , since  $\bar{p}q\bar{r} \subset q\bar{r}$ . Continue the reversal by expanding  $q\bar{r}$ ; this yields

$$p\bar{q}\bar{r} \vee \bar{p}qr \vee \bar{p}q\bar{r} \vee pqr \vee \bar{q}r = p\bar{r} \vee \bar{p}q \vee \bar{q}r. \tag{3}$$

The alternative minimal form is obtained by continuing the reversal. Expand Equation (3) into the “fully developed” form

$$pqr \vee p\bar{q}\bar{r} \vee \bar{p}qr \vee \bar{p}q\bar{r} \vee p\bar{q}r \vee \bar{p}\bar{q}r.$$

Then recombine sets pairwise front to back, wherever possible (first term with last term, first with next to last, etc). This results in the alternative minimal form

$$q\bar{r} \vee p\bar{q} \vee \bar{p}r.$$

**Example 5** (see p. 528 of Quine [1]) In order to simplify  $pqr \vee p\bar{r} \vee pq\bar{s} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{r}\bar{s}$ , the first step is to apply  $pqr \vee p\bar{r} = pq \vee p\bar{r}$ . The rest is straightforward.

$$pqr \vee p\bar{r} \vee pq\bar{s} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{r}\bar{s} = pq \vee p\bar{r} \vee pq\bar{s} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{r}\bar{s} = pq \vee p\bar{r} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{s},$$

which is the one of the four alternative minimal forms. A continuation of the reversal process would result in one of the other three minimal forms. For example

$$pq \vee p\bar{r} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{s} = pq \vee p\bar{r} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{r}\bar{s} = pqr \vee p\bar{r} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{r}\bar{s} = qr \vee p\bar{r} \vee \bar{p}r \vee \bar{p}\bar{q}\bar{s}.$$

The other two alternatives are  $p\bar{r} \vee \bar{p}r \vee pq \vee \bar{q}\bar{r}\bar{s}$  and  $p\bar{r} \vee \bar{p}r \vee qr \vee \bar{q}\bar{r}\bar{s}$ .

### NOTES

1. A complete subset or sublattice includes both its greatest lower bound (g.l.b.) and least upper bound (l.u.b.), and all elements which are both greater than the g.l.b. and less than the l.u.b.: the words “greater than” and “less than” are used in the context of partially ordered sets. In the sequel, “term”, “complete subset” and “sublattice” are used almost interchangeably where the meaning is clear from the discussion; likewise, “variable”, “component” and “indicator” are used interchangeably.
2. When the truth function represents the universal set, this is known in logic as a “tautology” if the truth value is “one”, and a “contradiction” if the truth value is “zero”. A method of proving that the system is a tautology is to show that the indicators for the truth function all disappear in simplification.
3. In this regard, the following quotation from Quine’s paper [1] on p. 531 is relevant:

Clearly, it would be desirable to find a quicker way of getting simplest normal equivalents, say by gearing the whole routine to irredundant formulas rather than to developed formulas. I have not seen how to manage this.

After an allowance is made for the differences between the terminology Quine used and that employed in this paper, the procedure described herein does obtain simplest normal equivalents without resorting to developed formulas.

4. The connection between symbolic logic and the algebraic theory of partially ordered sets is well known. However, the terminology and notation linking the two fields is not completely standardized, possibly because of the wide variety of disciplines which have made contributions to the understanding of this interface, including: philosophy, logic, mathematics and engineering. The approach taken in this paper is the same as that used in an earlier paper by the author on logical and probability analysis of systems [5]. Basically, it is Boolean notation, essentially the same as that used by G. Boole in *The Laws of Thought* [6], incorporating modern algebraic concepts, principally from lattice theory and Boolean algebra. For further information on related mathematics, the reader is referred to the books of Boolean algebra by Halmos [7], lattice theory by Birkhoff [8], and model theory by Bell and Slomson [9].
5. The merging operations described in the immediate foregoing, merging partitioned subsets and eliminating redundant terms, are included in the Quine-McCluskey procedure. Eliminating redundant indicators is not explicitly included as an operation in the descriptions of Quine-McCluskey, (see Quine [1], McCluskey [2], Phister [3], and Korfhage [4]), although it is well known in other contexts such as in analysis of Venn Diagrams. The "reversal test" which is described in the immediate sequel, is believed to be new. A PL-1 computer program which incorporates all three merging operations and the reversal test for simplifying Boolean polynomials was programmed by Myers [12].

#### REFERENCES

- [1] Quine, W. V., "The problem of simplifying truth functions," *American Mathematical Monthly*, vol. 59 (1952), pp. 521-531.
- [2] McCluskey, E. J., Jr., "Minimization of Boolean functions," *Bell System Technical Journal*, vol. 35 (1956), pp. 1417-1444.
- [3] Phister, M., Jr., *Logical Design of Digital Computers*, Wiley (1958).
- [4] Korfhage, R. R., *Logic and Algorithms*, Wiley (1966).
- [5] Locks, M. O., "Logical and probability analysis of systems," *Notre Dame Journal of Formal Logic*, vol. XIX (1978), pp. 123-136.
- [6] Boole, G., *An Investigation of the Laws of Thought*, Dover (1854).
- [7] Halmos, P. R., *Lectures on Boolean Algebra*, D. Van Nostrand (1963).
- [8] Birkhoff, Garrett, *Lattice Theory* (1967), 3rd ed., American Mathematical Society Colloquium Publications, Volume XXV.
- [9] Bell, J. L., and A. B. Slomson, *Models and Ultra-products: An Introduction*, North-Holland Co., Amsterdam (1969).
- [10] Rutherford, D. E., *Introduction to Lattice Theory*, Hafner (1965).
- [11] Phister, M., Jr., *The Logical Design of Digital Computing Machinery*, Course XL 198-24, University of California, Los Angeles, Engineering Extension, September 21, 1954.
- [12] Myers, George Edward, "Model for Boolean polynomial simplification (POLSIM)," in *MBA Research Report at Oklahoma State University* (May 1974).

*Oklahoma State University  
Stillwater, Oklahoma*