Pergamon

Mathl. Comput. Modelling Vol. 20, No. 9, pp. 97–103, 1994

0895-7177/94 $7.00 + 0.00

0895-7177(94)00166-9

Visualizing Boolean Operations on a Hypercube

M. C. CLARKE
Department of Computer Science, University of Natal
Pietermaritzburg, South Africa

(Received and accepted May 1994)

Abstract—Although it is known that every Boolean function can be represented as a subgraph of a hypercube [1], the implications of this fact have been little explored. We show a method for constructing such graphs which allows any formula of propositional logic to be visually represented. Boolean operations can be defined as manipulations of such graphs. A simple method is shown whereby the validity of propositional sequents may be checked.

Keywords—Possible models diagrams, Boolean function, Hypercube, Propositional logic, Truth table.

1. INTRODUCTION

Although it is known that every Boolean function can be represented as a subgraph of a hypercube [1], the implications of this fact have been little explored. That is, how does one construct such graphs and what does one do with them once they are constructed? Representing Boolean functions as graphs may provide an alternate scheme for automated propositional theorem proving, has been used to define concurrent processing [2], and has useful pedagogical implications [3].

This paper explains how propositional expressions can be represented by graphs which I call Possible Models Diagrams and defines procedures for combining such graphs according to the standard Boolean operators.

2. ANY BOOLEAN FUNCTION CAN BE REPRESENTED BY A GRAPH

The hypercube, or n-cube Q_n is a graph of order 2^n whose vertices are represented by n-tuples $\langle x_1, x_2, \dots x_n \rangle$ where $x_i \in \{0, 1\}$, and whose edges connect vertices which differ in exactly one term. Figure 1 shows Q_3, where each vertex is labeled with an abbreviated triple showing the values of x_1, x_2, x_3, respectively (e.g., we write the label "101" as an abbreviation for the triple $\langle 1, 0, 1 \rangle$).

For every Boolean function there is an equivalent function in the disjunctive form:

$$\begin{aligned} f(x_1, x_2, \dots x_n) = {} & \varepsilon_{m-1} x_1 x_2 \dots x_n + \varepsilon_{m-2} x_1 x_2 \dots x_{n-1} x'_n + \varepsilon_{m-3} x_1 x_2 \dots x'_{n-1} x'_n \\ & + \cdots + \varepsilon_1 x'_1 x'_2 \dots x'_{n-1} x_n + \varepsilon_0 x'_1 x'_2 \dots x'_n, \end{aligned} \tag{1}$$

where each $\varepsilon_i \in \{0, 1\}$ for $i = 1, 2, \dots, m$ and $m = 2^n$.

Typeset by $\mathcal{A}_\mathcal{M}\mathcal{S}$-TEX

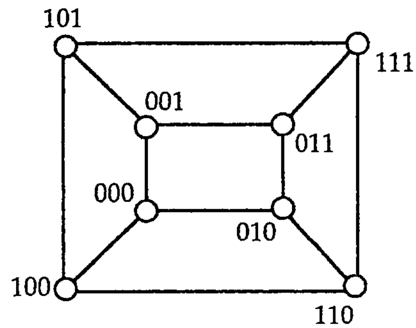


Figure 1. The 3-cube Q_3.

Harary has shown that if a Boolean function of n variables is written in this form, then the terms of the disjunction which have $\varepsilon_i = 1$ can be used to select a subset S of the vertices of Q_n. The subgraph induced by S can then be seen as a representation of the original Boolean function.

For example, the function

$$f(x_1, x_2, x_3) = x_1x_2x_3 + x_2'x_3 \tag{2}$$

could be rewritten as

$$\begin{aligned} f(x_1, x_2, x_3) = {} & 1.x_1x_2x_3 + 0.x_1x_2x_3' + 1.x_1x_2'x_3 + 0.x_1'x_2x_3 \\ & + 0.x_1x_2'x_3' + 0.x_1'x_2x_3' + 1.x_1'x_2'x_3 + 0.x_1'x_2'x_3', \end{aligned} \tag{3}$$

and hence, represented as the induced subgraph of Q_3 whose vertices are 111, 101, and 001 as in Figure 2.

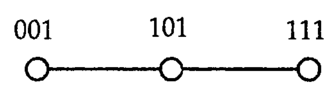


Figure 2. An induced subgraph of a Boolean function.

Perhaps we can more clearly illustrate the relationship between Q_3 and the subgraph representing (3) by leaving those vertices corresponding to terms with $\epsilon_i = 0$ open, while darkening the vertices for terms with $\epsilon_i = 1$.

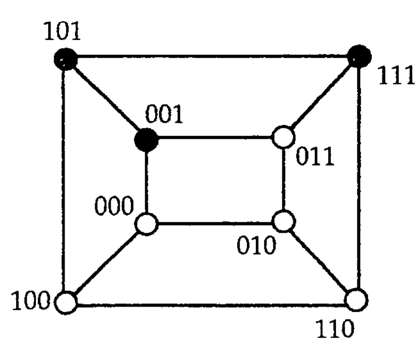


Figure 3. A more visually expressive representation.

Given such an induced subgraph, the Boolean function can easily be retrieved: simply form a disjunction whose disjuncts are given by the vertices in the subgraph (in any order). If the subgraph is identical to Q_n then the function must be tautologous; if the subgraph is empty then the function must be inconsistent; in other cases the function must be contingent.

However, given an arbitrary Boolean function, it would seem from Harary's approach [1] that one must first do substantial algebraic manipulation to find an equivalent function in the form of (1) before the subset S (and hence the subgraph induced by S) can be determined. This paper will describe another procedure for constructing this induced subgraph, and such a procedure can actually be used as a method for finding an equivalent function in the form of (1).

3. ANY PROPOSITIONAL FORMULA CAN BE EXPRESSED AS A BOOLEAN FUNCTION

It is implied by (1) that all Boolean functions can be expressed using only the operators negation, conjunction and disjunction. Nevertheless, there are other Boolean operators (such as implication and the bi-conditional) which play an important part in many systems of logic and many logical formulae are more naturally expressed using such operators. In order to broaden the applicability of hypercubes to propositional logic, we will also want to incorporate other such operators.

DEFINITION 1. [4, p. 44] *A well-formed formula* (wff) *of propositional logic is defined recursively as:*

(i) *Any propositional variable on its own is a wff.*
(ii) *If α is a wff then so is $\sim \alpha$.*
(iii) *If α and β are wffs then so are $(\alpha \& \beta)$, $(\alpha \vee \beta)$, $(\alpha \Rightarrow \beta)$ and $(\alpha \Leftrightarrow \beta)$.*

Several notes should be made about this definition:

- Whereas truth values in propositional logic are normally designated as True and False, there is a one-to-one correspondence between these values and the 1 and 0 of Boolean algebra.
- In accordance with the usual truth-table definitions of the operators $\sim$ (negation), & (conjunction), $\vee$ (disjunction), $\Rightarrow$ (implication), and $\Leftrightarrow$ (bi-conditional), each wff with n distinct propositional variables $V_1, V_2, \ldots, V_n$ defines a function from $V_1 \times V_2 \times \cdots \times V_n$ into $\{0,1\}$.
- All logical connectives (apart from negation) are formally defined as dyadic operators. Nevertheless, the propositional *conjunction* operator is associative and hence the wff $(V_1 \& (V_2 \& V_3))$ can be written as the unbracketed product $V_1 V_2 V_3$ in Boolean algebra. Likewise, nested propositional disjunctions may be written as unbracketed sums in Boolean algebra.
- Although this definition employs operators other than negation, disjunction and conjunction, such operators may still be defined as Boolean functions in the form (1). For instance, the material implication operator $\Rightarrow$ can be defined as the function:

$$\text{Implies}\ (x_1, x_2) = x_1' + x_2 = 1.x_1x_2 + 0.x1x_2' + 1.x_1'x2 + 1.x_1'x_2'. \tag{4}$$

Given Definition 1, any wff containing n distinct propositional variables may be expressed equivalently as a Boolean function of n variables, requiring only negation, disjunction and conjunction.

4. CONSTRUCTING THE GRAPH OF A PROPOSITIONAL WFF

Since all propositional wffs have an equivalent Boolean function, and every Boolean function can be expressed in the form (1), and every function in the form of (1) can be represented by an induced subgraph of a hypercube, it follows that every propositional wff may be represented as an induced subgraph of a hypercube. In this section, I describe two procedures for constructing such a graph: the first is most useful as a visual procedure, while the second is more useful if the task is to be computerized.

Hypercubes Are Better Drawn Hierarchically

Throughout the rest of this paper I will use n to denote the number of distinct propositional variables in a particular wff, thus allowing the wff to be shown as a subgraph of a hypercube of degree n. These hypercubes will be drawn hierarchically rather than in the usual form with no line-crossings: that is, instead of drawing Q_3 as in Figure 1, I will draw it as shown in Figure 4.

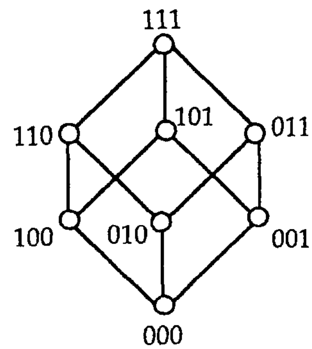


Figure 4. Preferred diagram for Q_3.

In these hierarchically-drawn hypercubes, vertices are arranged into $n+1$ rows $(R_0, R_1, \ldots, R_n)$ such that R_i contains those $\binom{n}{i}$ vertices whose n-tuple contains exactly i zeros. Thus, the top row always contains the single n-tuple $\langle 1, 1, \ldots, 1\rangle$ and the bottom row always contains just $\langle 0, 0, \ldots, 0\rangle$. Within any row, the vertices are arranged from left to right in decreasing of magnitude (when the vertices' labels are considered as binary numbers). The reason for this preference is that it allows hypercubes of any size to be constructed and labeled in a standardized manner, and allows labels to be assumed rather than explicitly written.

A hypercube drawn in this manner, with the possibility that some vertices are darkened and others left open, I call a Possible Models Diagram (PMD), for reasons which will later be clear. A PMD is a graph $G \simeq Q_n$ whose vertex set V is partitioned into two sets $T(G)$ (the darkened vertices) and $F(G)$ (the vertices left open).

Each Propositional Operation Corresponds to a Visual Manipulation of Hypercubes

Following Definition 1, we can form induced subgraphs of Q_n as follows:

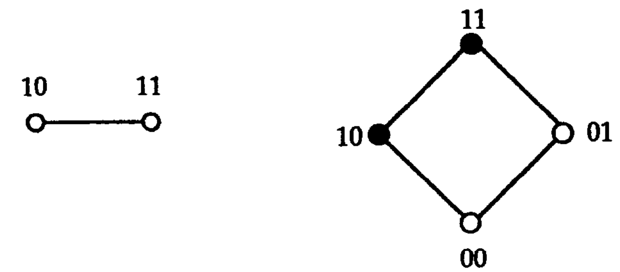
(i) Every propositional variable on its own can be represented as a subgraph containing exactly half the vertices of Q_n, namely, those vertices for which that propositional variable is True (i.e., 1). In other words, a propositional variable V_i can be represented by a PMD X in which $T(X) = \{\langle V_1, V_2, \ldots, V_n\rangle \mid V_i = 1\}$. For example, in a system with two propositional variables, the propositional variable V_1 may be represented by the induced subgraph shown in Figure 5a, or more clearly by the PMD in Figure 5b.

(ii) *Negation*: if the wff α is represented by the PMD A, then the wff $\sim \alpha$ can be formed by *reversing* the open and darkened vertices of A. That is, by constructing the PMD X with $T(X) = F(A)$.

(iii) If the wffs α and β are represented by the PMDs A and B, respectively, then the PMDs for $(\alpha \vee \beta)$, $(\alpha \& \beta)$, $(\alpha \Rightarrow \beta)$ and $(\alpha \Leftrightarrow \beta)$ can be constructed using the following visual operations:

Disjunction: form the PMD for $(\alpha \vee \beta)$ by *overlaying* A onto B. That is, form a PMD X with $T(X) = T(A) \cup T(B)$.

Conjunction: form the PMB for $(\alpha \& \beta)$ by *matching-the-dots* from A and B. That is, form a PMD X with $T(X) = T(A) \cap T(B)$.



(a) Induced subgraph for V_1 in a system of two variables.

(b) Possible Models Diagram for V_1 in a system of two variables.

Figure 5.

Material Implication: form the PMD for $(\alpha \Rightarrow \beta)$ by reversing A and overlaying onto B. That is, form a PMD X with $T(X) = F(A) \cup T(B)$.
Bi-conditional: form the PMD for $(\alpha \Leftrightarrow \beta)$ by finding vertices in A and B which have equal value. That is, form a PMD X with $T(X) = (T(A) \cap T(B)) \cup (F(A) \cap F(B))$.

ALGORITHM 1. Constructing a PMD.

Given a wff containing n distinct propositional variables, a PMD for that wff may be constructed by hand as follows:

- Draw an appropriate PMD underneath every propositional variable in the wff
- *While* (there is an operator for which no PMD has yet been constructed but whose operand(s) have PMDs already) *do*
    - Merge the PMDs of the operand(s) using the appropriate visual operation defined above

  end while
- The graph which was drawn last is the PMD for the wff

For example, consider the wff $P \Rightarrow (P \vee Q)$. Since the wff contains two distinct propositional variables, we can represent it as an induced subgraph of Q_2 and this subgraph can be constructed as a PMD as shown in Figure 6.

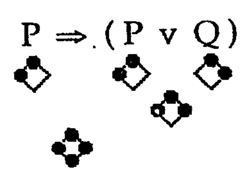


Figure 6. Constructing the PMD for a wff.

NOTE.

- When using this algorithm by hand, the vertices need not be labeled: the labels are assumed to follow the convention described earlier.
- The final PMD in Figure 6 has all vertices darkened, implying that the wff $P \Rightarrow (P \vee Q)$ is tautologous.

One can now see the purpose of the nomenclature: a Possible Models Diagram is a graph G which has one vertex for each possible model of n variables; and darkened vertices (those belonging to $T(G)$) indicate all those models in which the wff is True. Even though many wffs may have the same PMD, any two wffs with the same PMD will be logically equivalent. That is, each possible PMD represents an equivalence class of wffs.

ALGORITHM 2. Constructing a PMD.

If the construction of PMDs were to be computerized, then the following recursive algorithm may be used:

- Parse the wff and build a binary tree with operators as nodes and propositional variables as leaves
- To construct a PMD for the root node:
  - If this node's operator is not Negation then
    - Construct a PMD for the left subtree

    end if
  - Construct a PMD for the right subtree
  - Merge the two subtree PMDs in accordance with the operation defined above for this node's operator

This algorithm is simply a left-to-right depth-first traversal of the binary tree which reflects the structure of the wff.

5. A COMPARISON OF POSSIBLE MODELS DIAGRAMS AND TRUTH TABLES

There is, of course, a strong similarity between PMDs and Truth Tables. Both indicate the value of a propositional expression for every possible model: whereas a PMD has 2^n vertices, the corresponding truth table has 2^n rows. Either could be used to *define* the Boolean operators. Once a PMD has been constructed, it is a trivial matter to copy the information into a truth table, and vice-versa.

There are, however, two points in the favor of PMDs. One is that the visual nature of PMDs gives some pedagogical advantage over truth tables [3]. The second relates to the way sequents of propositional logic can be analyzed by PMDs, as described in the following section.

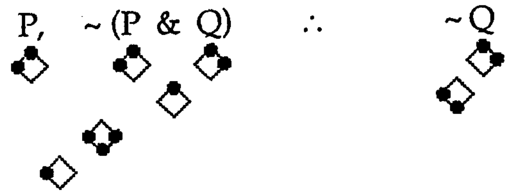
PMDs Can Be Used to Prove the Validity of Propositional Sequents

If $A_1, A_2, A_3, \ldots, A_m$ and B are propositional wffs, then $A_1, A_2, A_3, \ldots, A_m \therefore B$ is a propositional sequent claiming that the assumptions $A_1, A_2, A_3, \ldots, A_m$ entail the conclusion B. The validity of such a claim may be established by showing that the corresponding conditional $(A_1 \Rightarrow (A_2 \Rightarrow (A_3 \Rightarrow \ldots (A_m \Rightarrow B) \ldots)))$ is a tautology. As we have discussed, a wff may be shown to be tautologous by hand-constructing either a PMD or truth table, but this may be cumbersome for complex sequents.

An alternate way to check the validity of a sequent is to compare the models for which the left-hand side is true with the models for which the right-hand side is true. If the assumptions are to entail the conclusion, then every model which satisfies the assumptions must also satisfy the conclusion. In other words, the models which satisfy the conjunction[1] of the assumptions must be a subset of the models which satisfy the conclusion.

In terms of PMDs, a sequent $A_1, A_2, A_3, \ldots, A_m \therefore B$ may be analyzed as follows. First construct a PMD L for the wff $A_1 \& (A_2 \& (A_3 \& \ldots \& A_m)))$ and a separate PMD R for the conclusion B. The sequent is valid if and only if $T(L) \subseteq T(R)$. For example, suppose we want to test whether the sequent $P, \sim (P \& Q) \therefore \sim Q$ is valid. First, construct a PMD for the conjunction of the assumptions and a separate diagram for the conclusion:

[1]The commas between assumptions on the left-hand side of the sequent are taken as implicit conjunctions.



In this example, $T(L) = \{\langle 1,0\rangle\}$ and $T(R) = \{\langle 1,0\rangle, \langle 0,0\rangle\}$. Since $T(L) \subseteq T(R)$, the sequent $P, \sim (P\&Q) \mathrel{.\dot{}.} \sim Q$ must be valid.

This form of sequent validation could also be carried out using truth tables, but once again, the visual effect of the PMDs make it somewhat easier. The same thing could also be achieved using Boolean functions in this way—find a function in the form of (1) which is isomorphic to the conjunction of the assumptions; find another function in the form of (1) which is isomorphic to the conclusion; then the sequent is valid if and only if the first function is subsumed by the second.

6. CONCLUSION

Every propositional wff can be expressed as a Boolean function, and hence, as an induced subgraph of a hypercube. A Possible Models Diagram is a hypercube in which vertices may be darkened to indicate this induced subgraph. Given any wff, there is a straight-forward procedure for constructing that wff's PMD and this procedure can be executed either by hand or by computer.

PMDs can be used just as readily as truth tables to establish whether a wff is tautologous, contingent or inconsistent. Furthermore, PMDs may be used to prove the validity of any propositional sequent.

REFERENCES

1. F. Harary, The graph of a Boolean function, *Journal of Experimental and Theoretical Artificial Intelligence* (2), (1989).
2. V. Gupta, and V.R. Pratt, Gates accept concurrent behavior, In *Proc. 34th Ann. IEEE Symp. on Foundations of Comp. Sci.*, pp. 62–71, (1993).
3. M.C. Clarke, Possible models diagrams: A visual alternative to truth tables, *ACM SIGCSE Bulletin* **25** (1), (1993).
4. E.J. Lemmon, *Beginning Logic*, Nelson's University Paperbacks, (1971).
5. F. Harary, J.P. Hayes and H.-J. Wu, A survey of the theory of hypercube graphs, *Computers Math Applic.* **15** (4), (1988).