

A method for minimizing partially defined Boolean functions

By F. MÓRICZ,* A. VARGA* and P. ECSEDI-TÓTH**

A simple procedure is presented for minimizing partially defined Boolean functions. A binary tree is constructed to every such function in a natural way, then certain subtrees are used to obtain a partially defined irredundant normal form, equivalent to the starting function.

§ 1. Preliminaries

The truth-values TRUE, FALSE, and UNDEFINED will be denoted by 1, 0, and * (asterisk), respectively. The notion of partially defined Boolean function (truth function) and the notion of totally defined Boolean function (propositional formula) is used in the standard way. The collection of the partially defined Boolean functions will be denoted by \mathcal{B} , while the collection of the totally defined Boolean functions by \mathcal{F} . It is clear that $\mathcal{F} \subseteq \mathcal{B}$. We emphasize that the truth-values TRUE and FALSE are considered also as elements of \mathcal{F} , but UNDEFINED is not a partially defined Boolean function, i.e. we put

$$\{0, 1\} \subset \mathcal{F} \quad \text{and} \quad * \notin \mathcal{B}.$$

Let $\varphi, \psi_1, \dots, \psi_m$ be arbitrary formulae (in \mathcal{F}) and let A_1, \dots, A_m be arbitrary logical variables, not necessarily occurring in φ . If each occurrence of A_i in φ (if any) is substituted by ψ_i for $i=1, \dots, m$, then the resulting formula will be denoted by $\varphi[A_1|\psi_1, \dots, A_m|\psi_m]$, while the substitution process by $\langle A_1|\psi_1, \dots, A_m|\psi_m \rangle$. If $\{A_1, \dots, A_m\}$ is the (full) set of the logical variables occurring in φ and each formula ψ_i ($1 \leq i \leq m$) is identical with one of the truth-values TRUE and FALSE, then the substitution $\langle A_1|\psi_1, \dots, A_m|\psi_m \rangle$ will be called a *valuation*, and $\varphi[A_1|\psi_1, \dots, A_m|\psi_m]$ is the *value* of φ under this valuation. The value of φ is clearly logically equivalent to one of the truth-values TRUE and FALSE.

It is well-known from the propositional calculus that the value of any totally defined function φ does not depend on the order of the logical variables in the valuation. In other words, to determine the value of a formula under a valuation it is indifferent that the substitution is executed simultaneously or successively.

We note that in certain cases the same definition works for $\varphi \in \mathcal{B}$, too. In more detail, if $\langle A_1|\psi_1, \dots, A_m|\psi_m \rangle$ is a valuation for a partially defined function φ , then $\varphi[A_1|\psi_1, \dots, A_m|\psi_m]$, defined exactly as above, is either UNDEFINED or logically equivalent to one of the truth-values TRUE and FALSE.

§ 2. The tree constructing algorithm

For a partially defined Boolean function φ we have to know the set of those valuations for which φ is not defined, i.e. the set of undefined valuations. For any $\varphi \in \mathcal{B}$, let us write

$$\varphi[A_1|\psi_1, \dots, A_m|\psi_m] = *$$

if

$$\text{neither } \varphi[A_1|\psi_1, \dots, A_m|\psi_m] = 1 \text{ nor } \varphi[A_1|\psi_1, \dots, A_m|\psi_m] = 0,$$

where $\{A_1, \dots, A_m\}$ is the full set of the logical variables occurring in φ , and ψ_1, \dots, ψ_m are from the set $\{\text{TRUE}, \text{FALSE}\}$. We put

$$\varphi_* = \{ \langle A_1|\psi_1, \dots, A_m|\psi_m \rangle : \varphi[A_1|\psi_1, \dots, A_m|\psi_m] = * \}$$

and introduce the following two totally defined Boolean functions:

$$\varphi^0[A_1|\psi_1, \dots, A_m|\psi_m] = \begin{cases} \varphi[A_1|\psi_1, \dots, A_m|\psi_m] & \text{if } \langle A_1|\psi_1, \dots, A_m|\psi_m \rangle \notin \varphi_*, \\ 0 & \text{otherwise;} \end{cases}$$

$$\varphi^1[A_1|\psi_1, \dots, A_m|\psi_m] = \begin{cases} \varphi[A_1|\psi_1, \dots, A_m|\psi_m] & \text{if } \langle A_1|\psi_1, \dots, A_m|\psi_m \rangle \notin \varphi_*, \\ 1 & \text{otherwise.} \end{cases}$$

Finally, let us define a subset $[\varphi^0, \varphi^1]$ of \mathcal{F} by

$$[\varphi^0, \varphi^1] = \{ \psi \in \mathcal{F} : \text{if } \langle A_1|\psi_1, \dots, A_m|\psi_m \rangle \notin \varphi_*, \text{ then } \psi[A_1|\psi_1, \dots, A_m|\psi_m] = \varphi[A_1|\psi_1, \dots, A_m|\psi_m] \}.$$

Lemma 1. *Let the function f be defined as follows*

- (i) $f: \mathcal{B} \rightarrow \mathcal{P}(\mathcal{F})$, where $\mathcal{P}(\mathcal{F})$ is the power set of \mathcal{F} ;
- (ii) $f(\varphi) = [\varphi^0, \varphi^1]$ if $\varphi \in \mathcal{B}$.

Then f is one-one.

Proof. Trivial. \square

A somewhat stronger connection between the sets \mathcal{B} and $\mathcal{P}(\mathcal{F})$ can be easily established, too. In this paper, however, we need only the statement of the above lemma, so we do not deal with this strengthening.

Now, it is clear that a partially defined Boolean function φ may be given by the set φ_* and an arbitrary but fixed element ψ of $[\varphi^0, \varphi^1]$. By our point of interest, the function φ in question does not depend on the choice of ψ . This is what we are going to demonstrate in the subsequent paragraphs of this section. The totally defined Boolean function ψ is called a *representative* of φ and denoted by $[\varphi]$.

We turn to the tree constructing algorithm.

Let $\varphi \in \mathcal{B}$, fix an order S of the logical variables occurring in φ , and fix a representative $[\varphi]$. Then the following process will yield a binary tree.

- (i) Start with $[\varphi]$ as the initial vertex of the tree to be constructed.
- (ii) Choose the first variable if we come from (i) or the subsequent variable if we come from (vii), say A , in the fixed order S at the vertex ψ actually treated.
- (iii) Form the expressions $\psi[A|1]$ and $\psi[A|0]$.
- (iv) Apply the so-called computational rules of the propositional calculus (listed, e. g., in [2], [3], from (1) to (19)) as many times as possible in order to eliminate the truth-values from $\psi[A|1]$ and $\psi[A|0]$, unless they are truth-values.
- (v) If the elimination is terminated, then these truth-value free expressions, otherwise $\psi[A|1]$ and $\psi[A|0]$ themselves, will provide the two new vertices obtainable from ψ .
- (vi) If ψ' is obtained from ψ by substituting 1 (resp. 0) for A , then connect ψ and ψ' by an edge labelled with A (resp. \bar{A} , the negation of A).
- (vii) Stop if all the variables of S have been chosen, otherwise repeat from (ii).

As it was proved in [2], [3] this process, for every $[\varphi]$ and S , determines a unique binary tree denoted by $[\varphi]_S$. The vertices of this tree are formulae; in particular, all the lowest vertices are already truth-values, while the edges are labelled with literals, i.e. logical variables or negated logical variables. However, the tree $[\varphi]_S$ depends heavily on the representative $[\varphi]$; if another representative of φ is given, then the resulting tree will usually alter. Our next step is to remove this undesirable dependence. To this end, we recall the notion of end vertex and path (cf. [3]).

Consider the binary tree $[\varphi]_S$ constructed above. A vertex of $[\varphi]_S$ is called an *end vertex* if it is identical to a single truth-value (excluding UNDEFINED). By a *path* in $[\varphi]_S$ we mean a sequence of literals

- (i) whose first element labels an edge starting from $[\varphi]$,
 - (ii) the subsequent elements of the sequence label edges adjacent in $[\varphi]_S$,
 - (iii) the last element of the sequence labels an edge terminating at an end vertex.
- This end vertex will be called also the end vertex of the path in question.

It is clear that any path contains every literal at most once (i.e. there no loop can occur in $[\varphi]_S$).

We can easily establish a one-one connection between the set of all paths (of $[\varphi]_S$) and the set of all valuations (of the formula $[\varphi]$) in the following way:

- a) Let $p = \langle \varphi_1, \dots, \varphi_m \rangle$ be a path. Define the valuation $v_p = \langle A_1 | \psi_1, \dots, A_m | \psi_m \rangle$ by

$$\psi_j = \begin{cases} 0 & \text{if } A_j = \bar{\varphi}_j, \\ 1 & \text{otherwise.} \end{cases}$$

- b) Conversely, if $v = \langle A_1 | \psi_1, \dots, A_m | \psi_m \rangle$ is a valuation, then define the path $p_v = \langle \varphi_1, \dots, \varphi_m \rangle$ by

$$\varphi_j = \begin{cases} A_j & \text{if } \psi_j = 1, \\ \bar{A}_j & \text{otherwise.} \end{cases}$$

Lemma 2. Let $\varphi \in \mathcal{B}$ and $\psi', \psi'' \in \mathcal{F}$ be two representatives for φ . Let an order S of the logical variables occurring in φ and a valuation $v = \langle A_1 | \psi_1, \dots, A_m | \psi_m \rangle$ be fixed. If the end vertices of the two paths, associated with v in the trees ψ'_S and ψ''_S respectively, are not identical, then we necessarily have $v \in \varphi_*$.

Proof. Since $\psi' \in [\varphi^0, \varphi^1]$ and $\psi'' \in [\varphi^0, \varphi^1]$, by definition, for each valuation $\tilde{v} = \langle A_1 | \tilde{\psi}_1, \dots, A_m | \tilde{\psi}_m \rangle \notin \varphi_*$ we have

$$\begin{aligned} \psi' [A_1 | \tilde{\psi}_1, \dots, A_m | \tilde{\psi}_m] &= \varphi [A_1 | \tilde{\psi}_1, \dots, A_m | \tilde{\psi}_m] = \\ &= \psi'' [A_1 | \tilde{\psi}_1, \dots, A_m | \tilde{\psi}_m]. \end{aligned}$$

This implies immediately the statement of our Lemma. \square

By virtue of Lemma 2, for every $\varphi \in \mathcal{B}$ and fixed order S of the logical variables in φ we can uniquely define a tree φ_{S*} as follows:

- (i) Choosing an arbitrary representative for φ , let us construct the $[\varphi]_S$ -tree in the way described above;
- (ii) For each valuation $v \in \varphi_*$ let us put the sign $*$ at the end vertex of the path p_v associated with v .

Lemma 3. Let $\varphi \in \mathcal{B}$ and fix an order S of the logical variables occurring in φ . Then the tree φ_{S*} just constructed is uniquely determined (not depending on the representative $[\varphi]$).

Proof. The assertion clearly follows from the previous lemma. \square

§ 3. Minimization

Two partially defined Boolean functions, φ and ψ are said to be *equivalent* if

- (i) φ is defined under a valuation if and only if ψ is so;
- (ii) whenever φ is defined under a valuation $\langle A_1 | \psi_1, \dots, A_m | \psi_m \rangle$, then

$$\varphi [A_1 | \psi_1, \dots, A_m | \psi_m] = \psi [A_1 | \psi_1, \dots, A_m | \psi_m].$$

A $\varphi \in \mathcal{B}$ is said to be in a *disjunctive* (resp. *conjunctive*) *normal form* if $[\varphi]$ is so. If $[\varphi]$ is irredundant, then φ is also said to be *irredundant*.

We note that somewhat other definitions of these notions are possible, too. We do not go into the details of this question, since the above definitions are quite appropriate for us concerning the minimization of partially defined Boolean functions.

Let $\varphi \in \mathcal{B}$, a representative of φ , and an order S of the logical variables in φ be fixed. Construct the $[\varphi]_S$ -tree. If all the edges leading to the truth-value FALSE (resp. TRUE) are omitted in the $[\varphi]_S$ -tree, we get the so-called truncated $[\varphi]_S^0$ -tree (resp. $[\varphi]_S^1$ -tree) (cf. [2]).

The notion of maximal simplifiable subtree was introduced in [3]. Here we recall the definition.

Let a $[\varphi]_S^i$ -tree ($i=0$ or 1) and a path p in it be given. By a *maximal simplifiable subtree* (in abbreviation: MSST) of p we mean a subtree Φ of the $[\varphi]_S^i$ -tree which satisfies the following four conditions:

- (i) Φ contains p ;
- (ii) Each path in Φ starts from the initial vertex and terminates at an end vertex of the $[\varphi]_S^i$ -tree;
- (iii) Φ is of the form drawn in Fig. 1, where A_1, A_2, \dots, A_n are logical variables in φ and $\alpha_1, \dots, \alpha_{n+1}$ are sequences of literals such that each of the literals in these sequences is different from $A_1, \bar{A}_1, \dots, A_n, \bar{A}_n$; and there exists at least one path, the end vertex of which is not identical with $*$;
- (iv) The number of the paths in Φ is maximal in the sense that there exists no subtree of the $[\varphi]_S^i$ -tree that also satisfies the conditions (i), (ii), and (iii) and contains more paths than Φ contains.

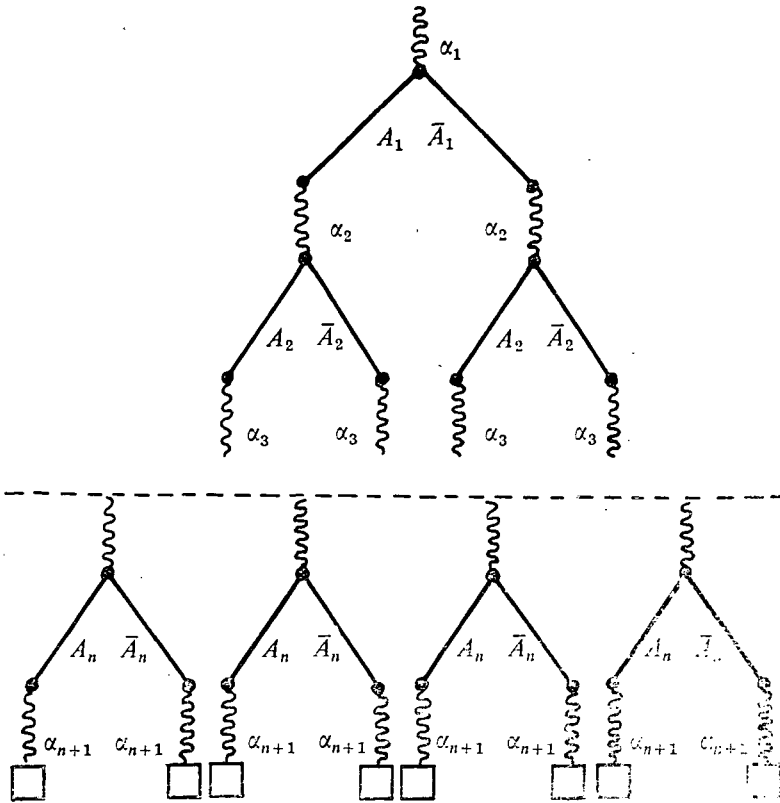


Fig. 1

These subtrees can be effectively generated by a simple way (see [2]) using an algorithm from [1].

By a *cover* of the truncated $[\varphi]_S^i$ -tree we mean a set of maximal simplifiable subtrees such that every path in $[\varphi]_S^i$ belongs to at least one MSST from this set. A cover is said to be *irredundant* if every MSST in it contains at least one path which belongs only to this MSST.

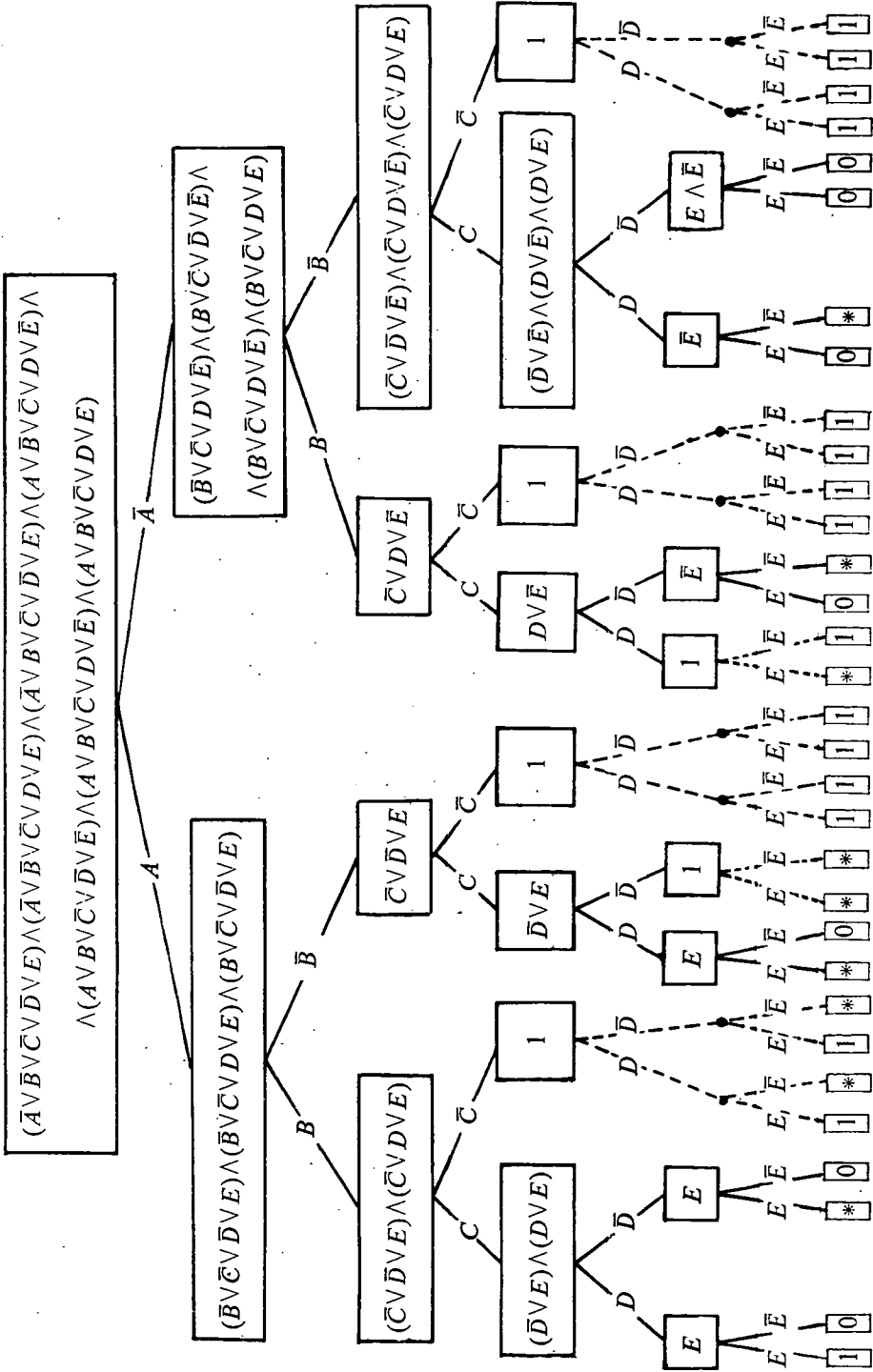


Fig. 2

In [2] we presented an algorithm for constructing an irredundant cover of an arbitrary φ_S^i -tree, when φ is a totally defined Boolean function. This algorithm can be applied without any changes to the case of partially defined Boolean functions if the end vertices $*$ in the $[\varphi]_S^i$ -tree are treated as if they were either 1 or 0 according to how it is more favourable to obtain an irredundant cover.

Thus, we find an irredundant cover of the $[\varphi]_S^i$ -tree, whence, via the main theorem of [2], we get an irredundant normal form of the representative $[\varphi]$. At the same time, this irredundant normal form of $[\varphi]$ is a representative of a partially defined irredundant normal form of φ . Taking the set φ_* into account, a partially defined irredundant normal form of φ is completely determined.

Example. Let $[\varphi]$ and φ_* be given as follows:

$$[\varphi] = (\bar{A}\bar{B}\bar{C}\bar{D}\bar{D}E) \wedge (\bar{A}\bar{B}\bar{C}\bar{D}DVE) \wedge (\bar{A}B\bar{C}\bar{D}\bar{D}VE) \wedge \\ \wedge (A\bar{B}\bar{C}\bar{D}DVE) \wedge (A\bar{B}B\bar{C}\bar{D}\bar{D}VE) \wedge (A\bar{B}B\bar{C}\bar{D}DVE) \wedge \\ \wedge (A\bar{B}B\bar{C}\bar{D}DVE),$$

$$\varphi_* = \{ \langle A, \bar{B}, C, \bar{D}, E \rangle; \langle A, B, \bar{C}, D, \bar{E} \rangle; \langle A, B, \bar{C}, \bar{D}, \bar{E} \rangle; \\ \langle A, \bar{B}, C, D, E \rangle; \langle A, \bar{B}, C, \bar{D}, E \rangle; \langle A, \bar{B}, C, \bar{D}, \bar{E} \rangle; \\ \langle \bar{A}, B, C, D, E \rangle; \langle \bar{A}, B, C, \bar{D}, \bar{E} \rangle; \langle \bar{A}, \bar{B}, C, D, \bar{E} \rangle \},$$

where A stands for $A|1$, while \bar{A} stands for $A|0$; similarly for the other letters. In the case $S = \langle A, B, C, D, E \rangle$ the $[\varphi]_S^i$ -tree is indicated in Fig. 2.

Let us consider the $[\varphi]_S^i$ -tree (Fig. 3).

An irredundant cover of the $[\varphi]_S^i$ -tree can be obtained by means of the maximal simplifiable subtrees indicated in Fig. 4.

Hence a representative of a partially defined irredundant normal form of φ will be

$$\bar{C} \vee (A \wedge E) \vee (\bar{A} \wedge B \wedge \bar{E}).$$

Finally, we should like to mention that from a practical point of view we have a considerable number of experimental evidences produced by two different PL/I

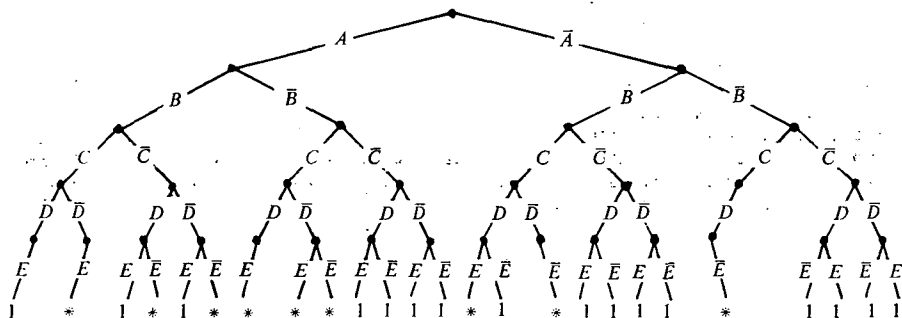


Fig. 3

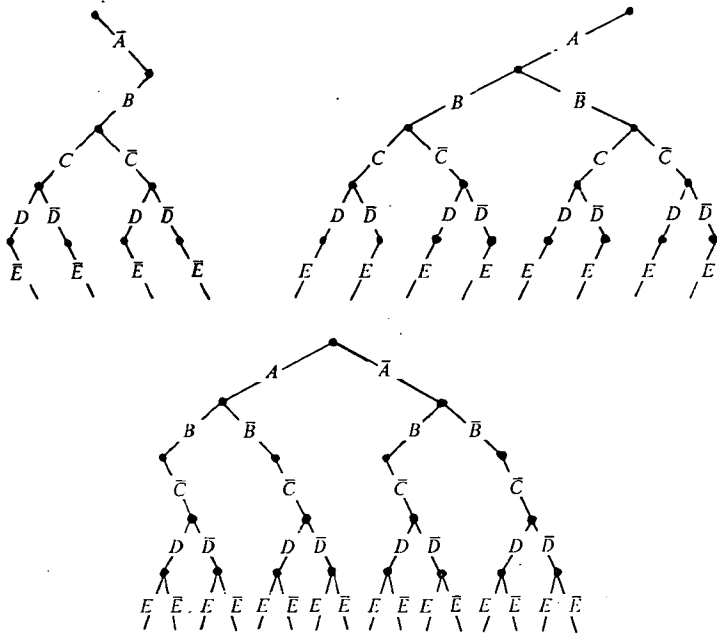


Fig. 4

realizations of our algorithm that provide minima on the average cost of other well-known algorithms such as e.g. the algorithm due to Quine—McCluskey. From a theoretical point of view, however, the proper nature of the minimization has not been understood yet.

* BOLYAI INSTITUTE OF THE
ATTILA JÓZSEF UNIVERSITY
ARADI VÉRTANÚK TERE 1.
SZEGED, HUNGARY
H-6720

** RESEARCH GROUP ON THEORY OF AUTOMATA
HUNGARIAN ACADEMY OF SCIENCES
SOMOGYI U. 7.
SZEGED, HUNGARY
H-6720

References

- [1] ECSEDI-TÓTH, P., F. MÓRICZ and A. VARGA, A note on symmetric Boolean functions, *Acta Cybernet.*, v. 3, 1978, pp. 321—326.
- [2] VARGA, A., P. ECSEDI-TÓTH and F. MÓRICZ, On a connection between algebraic and graph-theoretic minimization of truth functions, *Proceedings of the International Colloquium on Algebraic Methods in Graph Theory*, Szeged, 1978, Akadémiai Kiadó and North Holland Company.
- [3] VARGA, A., P. ECSEDI-TÓTH and F. MÓRICZ, A heuristic method for speeding up the manual optimization of Boolean functions, *Acta Tech. Acad. Sci. Hungar.*, to appear.

(Received Dec. 13, 1978)