Journal of Mathematical Research & Exposition
Vol.22, No.4, 531–537, November, 2002

On Decision Tree Complexity of Boolean Function and Yao's Question *

GAO Sui-xiang, YANG De-zhuang
(Math. Dept., Graduate School of the Chinese Academy of Sciences,
Hua Look-Keng Inst. of Appl. Math. & Info. Sci., Beijing 100039, China)

Abstract: A Boolean function $f(x_1, x_2, \cdots, x_n)$ is said to be elusive, if every decision tree algorithm computing f must examine all n variables in the worst case. In 1988, A.C.C. Yao introduced a question: Is any nontrivial monotone Boolean function that is invariant under the transitive act of group $C_m \times C_n$ elusive? The positive answer to this question supports the famous Rivest-Vuillemin conjecture on decision tree complexity. In this paper, we shall partly answer this question.

Key words: Boolean function; decision tree; complexity; Rivest-Vuillemin conjecture.

Classification: AMS(2000) 05C25,68Q05,68R05/CLC O157.1

Document code: A **Article ID:** 1000-341X(2002)04-0531-07

1. Introduction

It is well known that a tree is a connected graph without cycle. A rooted tree is a tree with a special vertex named root. Let T be a tree, when every edge of T is given a direction, we get a *directed tree*. In a directed tree, for any vertex V, the number of directed edges into V is called *indegree* of V, and the number of directed edges out of V is called *outdegree* of V.

A *rooted binary tree* T is a directed tree in which the indegree of the root vertex is 0 and the indegree of other vertices is 1, and the outdegree of any vertex is either 2 or 0. The vertices whose outdegree is 0 are called *leaves* of T. If directed edge $(x, y) \in T$, then x is called *father* of y, and y is called a *child* of x. Clearly, in a rooted binary tree, each vertex has two children but any leaf has no child.

A *Boolean function* is a function whose variable values and function value all are in $\{0, 1\}$. In general, Boolean function is represented by Boolean operations: conjunction $\wedge$, disjunction $\vee$ and negation $\neg$. For example,

$$f(x, y) = ((\neg x) \wedge y) \vee (x \wedge (\neg y)).$$

*Received date: 1999-10-19
Foundation item: Supported by the National Natural Science Foundation of China (10171095), Foundation of Chinese Academy of Science (J2001), and Foundation of GSCAS (yzjj200105)
Biography: GAO Sui-xiang (1963-), male, Ph.D., Associate Professor.

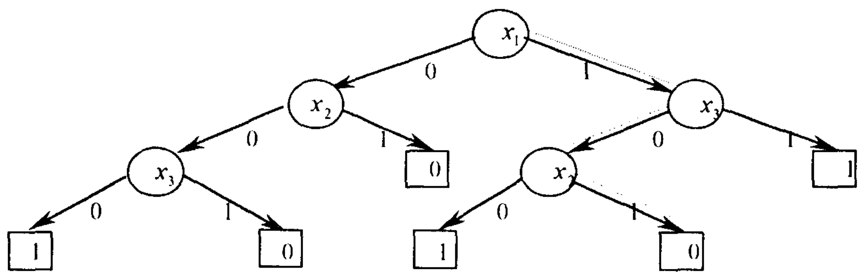
For simplicity, we usually write xy for $x \wedge y$, $x + y$ for $x \vee y$ and $\bar{x}$ for $\neg x$. So, above Boolean function can be written as

$$f(x,y) = \bar{x}y + x\bar{y}.$$

An *assignment* for a Boolean function is a mapping from its variables to $\{0,1\}$, each variable gets exactly one value from an assignment. For a Boolean function of n variables, an assignment can be seen as a binary string of length n, i.e., a string in $\{0,1\}^n$. An assignment x of Boolean function $f(x)$ is called a *truth-assignment* if $f(x) = 1$, and *false-assignment* if $f(x) = 0$. We denote by truth(x) and false(x) respectively the sets of variables taking value 1 and taking value 0 in the assignment x.

For two assignments of a Boolean function $f(x_1, x_2, \cdots, x_n)$, say, $x = (x_1, x_2, \cdots, x_n)$ and $y = (y_1, y_2, \cdots, y_n)$, if $x_i \leq y_i$ for all i, then we write $x \leq y$. A Boolean function $f(x)$ is *increasing* if $f(x) = 1$ and $x \leq y$ imply $f(y) = 1$, *decreasing* if $f(y) = 1$ and $x \leq y$ imply $f(x) = 1$, *monotone* if it is either increasing or decreasing. $f(x)$ is *nontrivial* if it is not a constant function.

A Boolean function f can also be represented by a rooted binary tree that is so-called decision tree of f. A *decision tree* of a Boolean function f is a rooted binary tree, whose non-leaf vertices are labeled by its variables, and leaves are labeled by 0 and 1. Edges of this binary tree are also labeled by 0 and 1 such that edges from a non-leaf vertex to its two children are labeled by 0 and 1 respectively, and any variable appears at most once in a path from the root to any leaf. Given an assignment to the variables of a Boolean function, we can compute the function value by its decision tree as follows: starting from the root, we look at its label. If its label is x_i, then we make a decision according to the value of x_i to decide where we go. If $x_i = 0$, then we go to the next vertex along the edge with label 0; if $x_i = 1$, then we go to the next vertex along the edge with label 1. Once a leaf is reached, the function value for the given assignment is obtained. For example, a decision tree of $f(x_1, x_2) = x_1x_3 + \bar{x}_2\bar{x}_3$ is as follows:



A path is marked in the decision tree for computing $f(1,1,0)$. Since the leaf is labeled by 0, we have $f(1,1,0) = 0$.

The decision tree representation of Boolean function is very useful in computer science. In fact, a decision tree of f gives a procedure (or say, algorithm) to compute the function value. The computation time depends on the length of path that is the number of variables on the path. The *depth* of a decision tree is the maximum length of all paths from root to leaves. Actually, the depth of a decision tree is exactly the number of queries that the algorithm must make for computing f in the worst case. In other words, the depth of a decision tree shows the computational complexity of the decision tree algorithm.

A Boolean function f may have a lot of decision trees. We denote by $D(f)$ the minimum depth of all decision trees for computing f. $D(f)$ is called the *decision tree complexity* of f. Clearly, $D(f) \le n$ when f has n variables. If $D(f) = n$, then f is said to be *elusive*.

It is known that decision tree complexity as an important measure of complexity is closely related to several other combinatorial and complexity measures, e.g., it is related to the certificate complexity (see [1]), to the block sensitivity ([2]), and to the packing of graphs ([3]). Furthermore, its logarithm is equal, up to a constant factor, to the time to compute f on a CREW PRAM ([2]).

A group G of permutations on $\{1,2,\cdots,n\}$ is called *transitive* if for any $i, j \in \{1, 2, \cdots, n\}$, there exists $\sigma \in G$ such that $\sigma(i) = j$. Let $f(x_1, x_2, \cdots, x_n)$ be a Boolean function and G be a group of permutations on $\{1, 2, \cdots, n\}$.$f(x_1, x_2, \cdots, x_n)$ is said to be *invariant* under group G if for any $\sigma \in G$,

$$f(x_1, x_2, \cdots, x_n) = f(x_{\sigma(1)}, x_{\sigma(2)}, \cdots, x_{\sigma(n)}).$$

A Boolean function $f(x_1, x_2, \cdots, x_n)$ is said to be *weakly symmetric* if there exists a transitive permutation group G on $\{1, 2, \cdots, n\}$ such that $f(x_1, x_2, \cdots, x_n)$ is invariant under G.

In last twenty years, many researchers have paid their efforts to decision tree complexity of Boolean function([3]~[13]). Especially, the following conjecture is one of focuses on which people concerned.

Rivest-Vuillemin Conjecture Every nontrivial monotone weakly symmetric Boolean function is elusive.

Authors of [10] proved that Rivest-Vuillemin Conjecture is true when n is a prime power. In general case, this conjecture is still open. In [12] A.C.C. Yao showed that any nontrivial monotone Boolean function which is invariant under the transitive act of cyclic group C_m must be elusive, and queried whether the analogous result is true for Boolean function which is invariant under the transitive act of group $C_m \times C_n$. This question is called Yao's question. Obviously, the positive answer to this question partly supports Rivest-Vuillemin Conjecture. In this paper, we make a discussion to decision tree complexity of Boolean function, mainly around Yao's question.

2. Preliminary

An *abstract complex* Δ on a finite set X is a family of subsets of X, such that if A is a member of Δ, so is every subset of A. Each member of abstract complex Δ is called a face of Δ. A *maximal face* of abstract complex Δ is a face that is not contained by another face. A *free face* is a non-maximal face that is contained by only one maximal face. An *elementary collapse* is an operation that deletes a free face together with all faces containing it. An abstract complex Δ is *collapsible* if it can be elementarily collapsed to the empty abstract complex.

The *complex* Δ_f of monotone Boolean function $f(x_1, x_2, \cdots, x_n)$ is an abstract complex that is defined as follows:

if $f(x)$ is monotone increasing, then $\Delta_f = \{\text{false}(x) | f(x) = 1\}$;

if $f(x)$ is monotone decreasing, then $\Delta_f = \{\text{truth}(x) | f(x) = 1\}$.

Since f is monotone, every subset of face A of Δ_f is still in Δ_f. Hence, abstract complex Δ_f is well defined. Each vertex of Δ_f is a variable of f.

The *Euler characteristic* of an abstract complex Δ is defined by

$$\chi(\Delta) = \sum_{A \in \Delta, A \neq \phi} (-1)^{|A|-1} = \sum_{A \in \Delta} (-1)^{|A|-1} + 1.$$

In particular, $\chi(\phi) = 1$ and $\chi(\{\phi\}) = 0$.

A permutation σ on the vertex set of abstract complex Δ is called an *automorphism* of Δ if for any face A of Δ, $\sigma(A) = \{\sigma(a) | a \in A\}$ is still a face of Δ. Let G be a group of automorphisms on Δ, an *orbit* of G is a minimal subset of vertices of Δ such that G takes no vertex out of it. It is obvious that G has only one orbit on Δ if and only if G is transitive on vertices of Δ.

Denote

$$\Delta^G = \{\{H_1, \cdots, H_k\} | H_1, \cdots, H_k \text{ are orbits of } G, \text{ and } H_1 \cup \cdots \cup H_k \in \Delta\} \cup \{\phi\}.$$

Clearly, Δ^G is also an abstract complex.

For an abelian group G (here either Z or Z_p, $p$is a prime), we may consider the homology groups with coefficients in G. Say a complex Δ is *G-acyclic* if the homology groups of Δ are

$$H_0(\Delta, G) = G, \quad H_i(\Delta, G) = 0, \quad i > 0,$$

where $H_i(\Delta, G)$ denotes the i-dimensional homology group of Δ with respect to G.

3. Main results

The following lemmas set up a bridge between algebraic topology and elusiveness.

Lemma 1[7] *Every collapsible abstract complex is Z_p-acyclic.*

Lemma 2[7] *If Δ_f is not collapsible, then f is elusive.*

Lemma 3[14][15] *Assume that G is a group of automorphisms on the finite Z_p-acyclic complex Δ. If there exists a normal subgroup G_1 of G such that $|G_1| = p^k$ (p is a prime and $k$is a positive integer) and the quotient group G/G_1 is cyclic, then $\chi(\Delta^G) = 1$.*

Lemma 4 *Let $f(x_1, x_2, \cdots, x_n)$ be a nontrivial monotone Boolean function, and G be a group of automorphisms on Δ_f. If there exists a normal subgroup G_1 of G such that (1) $|G_1| = p^k$ (p is a prime and k is a positive integer), (2) the quotient group G/G_1 is cyclic, and (3) $\chi(\Delta_f^G) \neq 1$, then f is elusive.*

Proof Suppose to the contrary that $f(x_1, x_2, \cdots, x_n)$ is not elusive. By Lemma 1 and Lemma 2, Δ_f is Z_p-acyclic. By Lemma 3, this result, combining the conditions of lemma, implies that $\chi(\Delta_f^G) = 1$, which contradicts the assumption of current lemma. □

Suppose that $f(x_1, x_2, \cdots, x_n)$ is a nontrivial monotone Boolean function, and $n = n_1 \cdot n_2$. Label all variables by elements in $Z_{n_1} \times Z_{n_2}$, say,

$$(x_1, x_2, \cdots, x_n) = (x_{11}, \cdots, x_{1n_2}, x_{21}, \cdots, x_{2n_2}, \cdots, x_{n_1 1}, \cdots, x_{n_1 n_2}).$$

Denote by G_{n_1} and G_{n_2} respectively a group of permutations on $\{1, 2, \cdots, n_1\}$ and $\{1, 2, \cdots, n_2\}$ $G_{n_1} \times G_{n_2}$ is the direct product of G_{n_1} and G_{n_2} acting on $Z_{n_1} \times Z_{n_2}$. For any $(\sigma, \tau) \in G_{n_1} \times G_{n_2}$ and any $(i, j) \in Z_{n_1} \times Z_{n_2}$,

$$(\sigma, \tau)(i, j) = (\sigma(i), \tau(j)).$$

Clearly, $G_{n_1} \times G_{n_2}$ naturally induces a group of permutations on the variables of

$$f(x_{11}, \cdots, x_{1n_2}, x_{21}, \cdots, x_{2n_2}, \cdots, x_{n_1 1}, \cdots, x_{n_1 n_2}).$$

$f(x_1, x_2, \cdots, x_n)$is said to be invariant under $G_{n_1} \times G_{n_2}$ if

$$\begin{aligned} &f(x_{11}, \cdots, x_{1n_2}, x_{21}, \cdots, x_{2n_2}, \cdots, x_{n_1 1}, \cdots, x_{n_1 n_2}) \\ &\quad = f(x_{\sigma(1)\tau(1)}, \cdots, x_{\sigma(1)\tau(n_2)}, x_{\sigma(2)\tau(1)}, \cdots, x_{\sigma(2)\tau(n_2)}, \cdots, x_{\sigma(n_1)\tau(1)}, \cdots, x_{\sigma(n_1)\tau(n_2)}) \end{aligned}$$

for all $(\sigma, \tau) \in G_{n_1} \times G_{n_2}$.

Lemma 5 *If $f(x_1, x_2, \cdots, x_n)$ is invariant under $G_{n_1} \times G_{n_2}$, then $G_{n_1} \times G_{n_2}$ induces a group of automorphisms on Δ_f.*

Proof Without loss of generality, suppose that $f(x_1, x_2, \cdots, x_n)$ is decreasing. For any $(\sigma, \tau) \in G_{n_1} \times G_{n_2}$, we are going to show that (σ, τ) is an automorphism of complex

$$\Delta_f = \{\mathrm{truth}(x) | f(x) = 1\}.$$

Take arbitrarily a face $A = \{x_{i_1 j_1}, \cdots, x_{i_k j_k}\}$ of Δ_f. Assume the assignment corresponding to this face is $(a_{11}, a_{12}, \cdots, a_{n_1 n_2})$, i.e.,

$$f(a_{11}, a_{12}, \cdots, a_{n_1 n_2}) = 1$$

and

$$\mathrm{truth}(a_{11}, a_{12}, \cdots, a_{n_1 n_2}) = A.$$

Notice that

$$f(a_{\sigma^{-1}(1)\tau^{-1}(1)}, a_{\sigma^{-1}(1)\tau^{-1}(2)}, \cdots, a_{\sigma^{-1}(n_1)\tau^{-1}(n_2)}) = 1$$

since $f(x_1, x_2, \cdots, x_n)$ is invariant under $G_{n_1} \times G_{n_2}$ and

$$(\sigma^{-1}, \tau^{-1}) \in G_{n_1} \times G_{n_2}.$$

Moreover,

$$\begin{aligned} (\sigma, \tau)(A) &= \{x_{\sigma(i_1)\tau(j_1)}, \cdots, x_{\sigma(i_k)\tau(j_k)}\} = \{x_{\sigma(i)\tau(j)} | x_{ij} \in A\} \\ &= \{x_{\sigma(i)\tau(j)} | a_{ij} = 1\} = \{x_{ij} | a_{\sigma^{-1}(i)\tau^{-1}(j)} = 1\} \\ &= \mathrm{truth}(a_{\sigma^{-1}(1)\tau^{-1}(1)}, a_{\sigma^{-1}(1)\tau^{-1}(2)}, \cdots, a_{\sigma^{-1}(n_1)\tau^{-1}(n_2)}). \end{aligned}$$

Hence $(\sigma, \tau)(A)$ is still a face of Δ_f corresponding to assignment

$$(a_{\sigma^{-1}(1)\tau^{-1}(1)}, \cdots, a_{\sigma^{-1}(n_1)\tau^{-1}(n_2)}).$$

This shows that (σ, τ) is indeed an automorphism of Δ_f. The analogous deduction can be done when $f(x_1, x_2, \cdots, x_n)$ is increasing. □

Now we can prove the following main theorem.

Theorem 1 *Let $f(x_1, x_2, \cdots, x_n)$ be a nontrivial monotone Boolean function, and $n = n_1 \cdot n_2$. If f is transitively invariant under group $G_{n_1} \times G_{n_2}$, where G_{n_1} and G_{n_2} are defined as above, $|G_{n_1}|$ is equal to a power of prime and G_{n_2} is cyclic. Then $f(x_1, x_2, \cdots, x_n)$ is elusive.*

Proof Label all variables of $f(x_1, x_2, \cdots, x_n)$ by elements in $Z_{n_1} \times Z_{n_2}$, and $G_{n_1} \times G_{n_2}$ acts on the variables as before. Denote $G = G_{n_1} \times G_{n_2}$. It is already shown in Lemma 5 that G is a group of automorphisms on Δ_f. It can be further checked that $G_{n_1} \times G_{n_2}$ has the following properties:

(1) Let $G_1 = \{(\sigma, 1) | \sigma \in G_{n_1}\}$. Then G_1 is a normal subgroup of G since for any $(\sigma, 1) \in G_1$ and $(\sigma_1, \tau_1) \in G$,

$$(\sigma_1, \tau_1)(\sigma, 1)(\sigma_1, \tau_1)^{-1} = (\sigma_1 \sigma \sigma_1^{-1}, 1) \in G_1$$

(2) The quotient group G/G_1 is cyclic since $G/G_1 \cong (G_{n_1} \times G_{n_2})/G_{n_1} \cong G_{n_2}$ and G_{n_2} is cyclic.

(3) $|G_1|$is a power of prime, and G is transitive on $Z_{n_1} \times Z_{n_2}$, by the assumptions of this theorem.

Besides, since G is transitive on $Z_{n_1} \times Z_{n_2}$, G has only one orbit on $Z_{n_1} \times Z_{n_2}$. But the monotonicity and nontrivicity of f imply that the only orbit is not in Δ_f. Thus $\Delta_f^G = \{\phi\}$. This turns out $\chi(\Delta_f^G) = 0$. By Lemma 4 , above results lead to the conclusion of current theorem. □

Corollary 1 *Let $f(x_1, x_2, \cdots, x_n)$ be a nontrivial monotone Boolean function, $n = n_1 \cdot n_2$. If $f(x_1, x_2, \cdots, x_n)$ is transitively invariant under the direct product $G_{n_1} \times G_{n_2}$ of permutation groups G_{n_1} and G_{n_2}, where $G_{n_1}$is a cyclic group of prime power order on $\{1, 2, \cdots, n_1\}$, and G_{n_2} is a cyclic group on $\{1, 2, \cdots, n_2\}$, then f is elusive.*

Proof It is an immediate result of Theorem 1. □

It is clearly that Corollary 1 partly answers the question of A. C-C. Yao.

Acknowledgments Authors are grateful to professor Ding-Zhu Du and professor Xiao-Dong Hu for insightful guidance and helpful suggestions.

References:

[1] WEGENER I. *The Complexity of Boolean Functions* [M]. Wcley-Teubner Sciences in Comput. Sci., New York, 1987.

[2] NISAN N. *CREW PRAMs and decision trees* [J]. SIAM J. Computer, 1991, 6: 999–1007.

[3] BOLLOBAS B. *Extremal Graph Theory* [M]. Academic Press, New York, 1978.

[4] BOLLOBAS B. *Complete subgraphs are elusive* [J]. J. Combinatorial Theory, Ser. B., 1976, **21**: 1–7.

[5] DU D Z. *Decision Tree Theory* [M]. Kluwer Academic Publishers, Boston, 1996.

[6] GAO Sui-Xiang. *Researches on Decision Tree Complexity* [M]. Dissertation, 1998.

[7] KAHN J, SAKS M, STRUTEVANT D. *A topological approach to evasiveness* [J]. Combinatorica, 1984, **4**: 297–306.

[8] KLEITMAN D J, KWIALKOWSKI D J. *Further results on the Aanderaa-Rosenberg conjecture* [J]. J. Combinatorial Theory, Ser. B., 1980, **28**: 85–95.

[9] MILNER E C, WELSH D J A. *On the computational complexity of graph theoretical properties* [J]. in Proc. Fifth British Combinatorial Conf., Utilitas Math. Winnipeg, 1976, 417-487.

[10] RIVEST R L, VUILLEMIN S. *A generalization and proof of the Aanderaa-Rosenberg conjecture* [J]. in Proceedings of 7^{th} ACM Symp. Theory of Computing, Albuquerque, 1975, 6–11.

[11] TRIESCH E. *Some results on elusive graph properties* [J]. SIAM J. Computing, 1994, **23**: 247–254.

[12] YAO A C-C. *Monotone bipartite graph properties are evasive* [J]. SIAM J. Computing, 1988, **17**: 517–520.

[13] YAP H P. *Some Topics in Graph Theory* [M]. London Mathematical Society Lecture Note Series 108, Cambridge University Press. 1986.

[14] OLIVER R. *Fixed-point sets of group actions on finite acyclic complex* [J]. Comment. Math. Helvetici, 1975, **50**: 155–177.

[15] SMITH P A. *Fixed-point theorems for periodic transformations* [J]. Amer. J. Math., 1941, **63**: 1–8.

布尔函数的判定树复杂性及问题

高 随 祥， 杨 德 庄

(中国科学院研究生数学部，华罗庚应用数学与信息科学研究中心，北京 100039)

摘 要: 设 $f(x_1,x_2,\cdots,x_n)$ 是一个布尔函数。如果计算 $f(x_1,x_2,\cdots,x_n)$ 的每个判定树算法在最坏情况下都要检查所有 n 个变量才能求得 f 的值，则称 f 是诡秘函数。1988年， A.C.C. Yao 提出一个问题：如果一个单调非平凡的布尔函数 $f(x_1,x_2,\cdots,x_n)$ 在循环群 $C_m\times C_n$ 的直积的可迁作用下不变，则 f 是诡秘的吗？对这个问题的肯定回答支持著名的 Rivest-Vuillemin 猜想. 本文将部分地解答这一问题.