V. K. KOROBKOV

1963

SovietRxiv

Abstract

Full Text

V. K. KOROBKOV

ESTIMATING THE NUMBER OF MONOTONE FUNCTIONS OF THE ALGEBRA OF LOGIC AND THE COMPLEXITY OF AN ALGORITHM FOR FINDING A RESOLVING SET FOR AN ARBITRARY MONOTONE FUNCTION OF THE ALGEBRA OF LOGIC

(Presented by Academician S. L. Sobolev, 22 XI 1962)

In paper ([1]) a class of algorithms was considered for finding a resolving* set for an arbitrary monotone function $f(x_1, x_2, \dots, x_n)$, depending on no more than n variables, specified by an operator A_f, which determines, for any point $\alpha \in E_n$, the value of the function $f(x_1, x_2, \dots, x_n)$ at that point. The process of finding the resolving set consisted of the following: a point $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n) \in E_n$ was chosen; with the aid of the operator A_f the value of the monotone function $f(x_1, x_2, \dots, x_n)$ at the point α was computed; depending on the value $f(\alpha_1, \alpha_2, \dots, \alpha_n)$, a new point $\beta \in E_n$ was chosen, and the process was repeated.

This process may be represented in the form of a tree**, whose vertices (with the exception of the pendant vertices) correspond to points of E_n. In what follows we shall say that a vertex a has number $(\alpha_1, \alpha_2, \dots, \alpha_n)$ if the point $(\alpha_1, \alpha_2, \dots, \alpha_n) \in E_n$ is put in correspondence with it. The arcs issuing from a vertex with number $(\alpha_1, \alpha_2, \dots, \alpha_n)$ correspond to the possible values of the operator A_f at the point $(\alpha_1, \alpha_2, \dots, \alpha_n)$, and to each pendant vertex b there corresponds a monotone function $\varphi(x_1, x_2, \dots, x_n)$ for which the conditions are satisfied:

$$G(\varphi, M) \subseteq \overline{N}; \tag{1}$$

$$\varphi(\alpha_i) = \beta_i \qquad (i = 0, 1, 2, \dots, k), \tag{2}$$

where $N = \{a_0, a_1, \dots, a_k, b\}$ is the set of vertices of the path connecting the root vertex of the tree a_0 with the given pendant vertex b, $G(\varphi, M)$ is a resolving set for the function $\varphi(x_1, x_2, \dots, x_n)$; $\overline{N} = \{a_0, a_1, \dots, a_k\}$ is the set of numbers of the vertices of the path N, and $\{\beta_0, \beta_1, \dots, \beta_k\}$ are the values of the operator A_f for the corresponding arcs $(a_0, a_1), (a_1, a_2), \dots, (a_k, b)$. Thus, for $n = 1, 2$ the trees will have the following form:

tree diagrams for $n = 1, 2$, with leaves labeled $f_1 \equiv 1,\ f_2 = x,\ f_3 \equiv 0$, and $f_1 \equiv 1,\ f_2 = x_1 \vee x_2,\ f_3 = x_1,\ f_4 = x_2,\ f_5 = x_1 \& x_2,\ f_6 \equiv 0$

Figure 1: tree diagrams for $n = 1, 2$, with leaves labeled $f_1 \equiv 1,\ f_2 = x,\ f_3 \equiv 0$, and $f_1 \equiv 1,\ f_2 = x_1 \vee x_2,\ f_3 = x_1,\ f_4 = x_2,\ f_5 = x_1 \& x_2,\ f_6 \equiv 0$

* In the present note the terminology introduced in ([1]) is used.
** For the definition of a tree and its elements, see ([2]).

It is obvious that specifying an algorithm A for finding a resolving set, even in the form of normal (3) algorithms, uniquely specifies the corresponding tree H_A, and conversely, specifying the tree uniquely determines the algorithm for finding the resolving set.

Let $F(n)$ be an algorithm for finding a resolving set for an arbitrary monotone function depending on no more than n variables, and let $H(n)$ be the tree corresponding to the algorithm $F(n)$. Consider the following characteristic of the tree $H(n)$: let $S = \{s\}$ be the set of paths connecting the root vertex with any terminal vertex of the tree, and let $\lambda(s)$ be the length of the path s; then we shall call the height of the tree $H(n)$ the number

$$\rho(H(n)) = \max_{s \in S} \lambda(s),$$

and it is natural to estimate the complexity of the algorithm $F(n)$ by the quantity $\rho(H(n))$.

The problem may be formulated as follows: it is required to construct an algorithm $F(n)$ for finding a resolving set for an arbitrary monotone function depending on no more than n variables and satisfying the following conditions:

For any monotone function $\varphi(x_1, x_2, \dots, x_k)$ $(k \leq n)$

there exists in the tree $H(n)$ one and only one terminal vertex a having properties (1), (2).

(3)

$$\rho(H(n)) \leq \rho(H'(n)), \quad \text{where } H'(n) \text{ is a tree,}$$

corresponding to an algorithm $F'(n)$ possessing property (3).

(4)

It is easy to show that, for any algorithm $F(n)$ possessing property (3), the following lemma holds.

Lemma 1.

$$\rho(H(n)) \geq C_n^{[n/2]} + C_n^{[n+2]+1}.$$

Indeed, consider the monotone function $\varphi(x_1, x_2, \dots, x_n)$, defined as follows:

$$\varphi(\alpha_1, \alpha_2, \dots, \alpha_n) = \begin{cases} 1, & \text{if } \left[\frac{n}{2}\right] + 1 \le \sum_{i=1}^{n} \alpha_i \le n, \\ 0, & \text{if } 0 \le \sum_{i=1}^{n} \alpha_i \le \left[\frac{n}{2}\right]. \end{cases}$$

It is obvious that $G(\varphi, M)$ contains exactly $C_n^{[n/2]} + C_n^{[n/2]+1}$ points; then, taking (1) and (3) into account, we easily obtain the assertion of the lemma.

In the present work an algorithm $F(n)$ is proposed, satisfying condition (3), for which

$$\rho(H(n)) \le 5C_n^{[n/2]}.$$

By virtue of (3), it is obvious that the number of monotone functions $\psi(n)$ depending on no more than n variables is equal to the number of terminal vertices of the tree $H(n)$; then from the estimate for the height of the tree $H(n)$ it follows that

$$\psi(n) \le 2^{\rho(H(n))} \le 2^{5C_n^{[n/2]}}.$$

The currently known estimate, obtained in (4), has the form

$$2^{C_n^{[n/2]}} \le \psi(n) \le n^{C_n^{[n/2]}} + 2.$$

We proceed to the description of the algorithm $F(n)$ for finding a resolving set for an arbitrary monotone function depending on no more than n variables. We shall construct the algorithm by induction on the number of variables.

For $n = 1, 2$ the corresponding trees $H(n)$ were given above. Suppose that for any $k < n$ algorithms satisfying (3) have already been constructed; we pass to the construction of the algorithm $F(n)$.

An arbitrary monotone function $f(x_1, x_2, \dots, x_n)$ can be represented as follows:

$$f(x_1, x_2, \dots, x_n) =$$

$$= \bigvee_{\{(\sigma_1, \sigma_2, \dots, \sigma_k)\}} x_1^{\sigma_1} \& x_2^{\sigma_2} \& \cdots \& x_k^{\sigma_k} \& f(\sigma_1, \sigma_2, \dots, \sigma_k, x_{k+1}, x_{k+2}, \dots, x_n),$$

where

$$f(\sigma_1, \sigma_2, \dots, \sigma_k, x_{k+1}, x_{k+2}, \dots, x_n) = f_{\sigma_1, \sigma_2, \dots, \sigma_k}(x_{k+1}, x_{k+2}, \dots, x_n)$$

are monotone functions of $n - k$ variables. Consider the points

$$\lambda_i = (\sigma_1^i, \sigma_2^i, \dots, \sigma_k^i),$$

for which

$$\sum_{j=1}^{k} \sigma_j^i = 2l + 1 \quad \left(l = 0, 1, \dots, \left[\frac{k}{2}\right]\right),$$

and renumber them in some way as λ_i $(1 \le i \le 2^{k-1})$. Consider $H(n-k)$: its vertices (with the exception of dangling vertices) correspond to the points E_{n-k}. For each point λ_j $(1 \le j \le 2^{k-1})$, define the tree $H(\lambda_j, n-k)$ as follows: in the tree $H(n-k)$, to each nondangling vertex with number $(\alpha_1, \alpha_2, \dots, \alpha_{n-k})$ assign the number $(\sigma_1^j, \sigma_2^j, \dots, \sigma_k^j, \alpha_1, \alpha_2, \dots, \alpha_{n-k})$. It is clear that the tree $H(\lambda_j, n-k)$ uniquely determines, for an arbitrary monotone function $f(x_1, \dots, x_n)$ specified by the operator A_f, the algorithm $F(n, \lambda_j)$ for finding a resolving set for the function

$$f_{\sigma_1^j, \sigma_2^j, \dots, \sigma_k^j}(x_{k+1}, \dots, x_n).$$

Successive application of the algorithms $F(n, \lambda_1)$, $F(n, \lambda_2), \dots, F(n, \lambda_{2^{k-1}})$ makes it possible to find resolving sets for all functions

$$f_{\sigma_1^i, \sigma_2^i, \dots, \sigma_k^i}(x_{k+1}, \dots, x_n),$$

for which

$$\sum_{j=1}^{k} \sigma_j^i = 2l + 1 \quad \left(l = 0, 1, \dots, \left[\frac{k}{2}\right]\right).$$

Let these sets be $G(\lambda_1)$, $G(\lambda_2), \dots, G(\lambda_{2^{k-1}})$. Then the following is true.

Lemma 2. The set

$$K_f = E_n \setminus \bigcup_{1 \le i \le 2^{k-1}} \left(G(\lambda_i) \cup \bigcup_{\alpha \in P(\lambda_i)} N(\alpha) \cup \bigcup_{\alpha \in Q(\lambda_i)} V(\alpha) \right)$$

(where $P(\lambda_i)$ is the set of upper zeros of the function

$$f_{\sigma_1^i, \sigma_2^i, \dots, \sigma_k^i}(x_{k+1}, \dots, x_n),$$

and $Q(\lambda_i)$ is the set of lower ones of the function

$$f_{\sigma_1^i,\sigma_2^i,\dots,\sigma_k^i}(x_{k+1},\dots,x_n)$$

)

contains no more than

$$C_k^{[k/2]}\cdot 2^{n-k}$$

points.

Indeed, consider two points $\alpha,\beta\in E_n$, let $\alpha=(\alpha_1,\alpha_2,\dots,\alpha_n)$, $\beta=(\beta_1,\beta_2,\dots,\beta_n)$, $\alpha_i=\beta_i$ $(k+1\le i\le n)$; then $\alpha,\beta\in K_f$ if $\tilde{\alpha}=(\alpha_1,\alpha_2,\dots,\alpha_k)$ and $\tilde{\beta}=(\beta_1,\beta_2,\dots,\beta_k)$ are incomparable Boolean vectors (5), or else $\tilde{\alpha}=\tilde{\beta}$. Indeed, if $\tilde{\alpha}$ and $\tilde{\beta}$ are comparable Boolean vectors, then, if $\tilde{\alpha}\ne\tilde{\beta}$, there is always a point $\gamma\in E_n$, following one of them and preceding the other, such that $\|\gamma\|=2l+1$, i.e. $\gamma\notin K_f$, and then, by monotonicity of $f(x_1,x_2,\dots,x_n)$, either $\alpha\notin K_f$, or $\beta\notin K_f$. Since the number of incomparable Boolean vectors in E_k does not exceed $C_k^{[k/2]}$ (see (5)), the assertion of the lemma follows from this.

Finally, we obtain that the algorithm $F(n)$ will consist of the successive application of the algorithms $F(n,\lambda_1)$, $F(n,\lambda_2),\dots,F(n,\lambda_{2^{k-1}})$, and of the algorithm $F(n,K_f)$ for computing A_f on the set K_f. It is easy to see that the algorithm $F(n)$ satisfies condition (3). From what has been set forth it follows directly that

$$\rho(H(n))\le\rho(H(n-k))\cdot 2^{k-1}+C_k^{[k/2]}\cdot 2^{n-k}. \quad (5)$$

Theorem.

$$\rho(H(n))\le 5C_n^{[n/2]}.$$

It is obvious that $\rho(H(n))\le 2^n$, and therefore the validity of the theorem for $1\le n\le 14$ follows from the relation $5C_m^{[m/2]}\ge 2^m$ $(m=1,2,\dots,14)$. Cons⋯

considerations analogous to those given above, one can easily obtain the relation*: $\rho(H(n))\leqslant 2\rho(H(n-2))+2^{n-2}$, whence it follows that $\rho(H(n))\leqslant 2^{n-1}+2^{[n/2]}$. Using this estimate, one can obtain, for $15\leqslant n\leqslant 56$,

$$\rho(H(n))\leqslant 2^{n-1}\left(1+\frac{1}{27}\right)\leqslant\sqrt{\frac{2}{\pi}}\,\frac{2^n}{\sqrt{n}}\,5\,\frac{15\sqrt{15}}{16\sqrt{14}}\,e^{-1/42}\leqslant 5C_n^{[n/2]}.$$

Here and below we use the estimates

$$\sqrt{\frac{(2m+1)^3}{8m(m+1)^2}}^{\,-1/6m}\sqrt{\frac{2}{\pi}}\,\frac{2^n}{\sqrt{n}} \leqslant C_n^{[n/2]} \leqslant \sqrt{\frac{2}{n}}\,\frac{2^n}{\sqrt{n}}\,e^{1/24m} \quad \text{for } n \geqslant 2m.$$

Assuming that the theorem is valid for all $l < n$, and putting in (5)

$$k = 2([n/6] + \operatorname{sgn}(n/6 - [n/6])),$$

we obtain, for $n \geqslant 57$,

$$\rho(H(n)) \leqslant 5C_{n-k}^{[(n-k)/2]} \cdot 2^{k-1} + C_k^{[k/2]} \cdot 2^{n-k} \leqslant$$

$$\leqslant \sqrt{\frac{2}{\pi}}\,\frac{2^n}{\sqrt{n}}\left(\frac{5\sqrt{n}\,e^{1/24[(n-k)/2]}}{2\sqrt{n-k}} + \frac{\sqrt{n}\,e^{1/24[k/2]}}{\sqrt{k}}\right) \leqslant$$

$$\leqslant \sqrt{\frac{2}{\pi}}\,\frac{2^n}{\sqrt{n}}\left(\frac{5e^{1/24\cdot 18}}{2\sqrt{37/57}} + \sqrt{3}\,e^{1/24\cdot 10}\right) \leqslant$$

$$\leqslant \sqrt{\frac{2}{\pi}}\,\frac{2^n}{\sqrt{n}}\,5\sqrt{\frac{57^3}{224\cdot 29^2}}\,e^{-1/168} \leqslant 5C_n^{[n/2]}.$$

In conclusion we give a table of lower and upper estimates for the height of the tree $H(n)$ for small n, obtained in the present work.

n	1	2	3	4	5	6	7	8
$C_n^{[n/2]} + C_n^{[n/2]+1}$	2	3	6	10	20	35	70	126
$\rho(H(n))$	2	3	6	10	20	36	72	136

Institute of Mathematics with Computing Center
of the Siberian Branch of the Academy of Sciences of the USSR

Received
12 XI 1962

CITED LITERATURE

1. V. K. Korobkov, T. L. Reznik, DAN, **147**, No. 5 (1962).

2. C. Berge, *The Theory of Graphs and Its Applications*, IL, 1962.

3. A. A. Markov, Tr. Mat. Inst. im. V. A. Steklova AN SSSR, **42** (1954).

4. E. N. Gilbert, *Cybernetics Collection*, 1, IL, 1960, p. 175.

5. V. M. Mikheev, *Problems of Cybernetics*, issue 2, Moscow, 1959, p. 69.

* Note that this relation is a refinement of (5) for $k = 2$.

Note: Figure translations are in progress. See original paper for figures.