

Algorithms for Constructing of Decision Trees

Mikhail Moshkov

Research Institute for Applied Mathematics and
Cybernetics of Nizhni Novgorod State University
10, Uljanova St., Nizhni Novgorod, 603005, Russia
e-mail: moshkov@nnucnit.unn.ac.ru

Abstract. Decision trees are widely used in different applications for problem solving and for knowledge representation. In the paper algorithms for decision tree constructing with bounds on complexity and precision are considered. In these algorithms different measures for time complexity of decision trees and different measures for uncertainty of decision tables are used. New results about precision of polynomial approximate algorithms for covering problem solving [1, 2] show that some of considered algorithms for decision tree constructing are, apparently, close to unimprovable.

1 Introduction

In 1983 the paper [4] was published (short variant was published in 1982 [3]) which contained bounds on time complexity of decision trees and algorithms for decision tree constructing (in this paper decision trees were named conditional tests). In algorithms different measures for time complexity of decision trees (depth, weighted depth and others) and different measures for uncertainty of decision tables were used. Bounds on precision for these algorithms were considered. These algorithms resemble on algorithms of J.R. Quinlan [10, 11] however they were proposed independently. Obtained results were published only in Russian.

New bounds on precision of polynomial approximate algorithms for covering problem solving [1, 2] show that some of considered in [4] algorithms are, apparently, close to unimprovable polynomial approximate algorithms for constructing decision trees with minimal depth.

This paper contains a survey of some results from [3, 4] (definitions of complexity and uncertainty measures, description of algorithms for decision tree constructing, bounds on precision of these algorithms), some results from [5] (bounds on complexity of considered algorithms and bounds on precision of algorithms for one important class of decision tables), and reasons about closeness of some algorithms to unimprovable.

Mentioned results are useful for obtaining bounds on time complexity of decision trees [7, 8]. Also these results may be useful in data mining and knowledge discovery for decision tree constructing. The considered algorithms allow variations of complexity and uncertainty measures in broad bounds: the statements

on complexity and precision of these algorithms are true for wide classes of complexity measures and uncertainty measures. Also some of these algorithms are, apparently, close to unimprovable (in the sense of precision) algorithms with polynomial time complexity. Therefore considered algorithms may be suitable in experiments for constructing of decision trees which are optimal (in some sense or other) or close to optimal.

2 Basic Definitions and Notation

The notions of decision table, decision tree, complexity measure and uncertainty measure are introduced in this section.

2.1 Decision Tables

For arbitrary nonempty set (alphabet) A the set of all finite words over A containing the empty word λ will be denoted by A^* .

Denote $\omega = \{0, 1, 2, \dots\}$ and $F = \{f_i : i \in \omega\}$. Let $k \in \omega \setminus \{0, 1\}$ and $E_k = \{0, 1, \dots, k-1\}$. We will interpret the set F as a set of attribute names and the set E_k as a set of attribute values. Let us denote $\Omega = \{(f_i, \delta) : f_i \in F, \delta \in E_k\}^*$.

An ordered triple $T = (\Delta, \nu, \mu)$ will be called a *decision table* if $\Delta \subseteq E_k^n$, $\nu : \Delta \rightarrow \omega$ and $\mu : \{1, \dots, n\} \rightarrow F$ for certain $n \in \omega \setminus \{0\}$ respectively, and $i \neq j$ implies $\mu(i) \neq \mu(j)$ for each $i, j \in \{1, \dots, n\}$.

The table T may be conveniently depicted as rectangular matrix with n columns, the rows of which are n -tuples from $\Delta(T)$. Every row δ of this matrix is labeled with a number $\nu(\delta)$ (the value of decision) while i -th column of it (for $i = 1, \dots, n$) is labeled with the element $\mu(i)$ of the set F (the name of i -th attribute).

In what follows the set Δ and the mappings ν, μ , defining the table T , will be denoted by $\Delta(T), \nu_T$ and μ_T respectively. Let us denote $\dim T = n$ and call the number n the *dimension* of the table T . Introduce also notation $P(T) = \{\mu_T(i) : i = 1, \dots, n\}$ and $\Omega(T) = \{(f_i, \delta) : f_i \in P(T), \delta \in E_k\}$.

For $u \in \Omega(T)$ define the decision table Tu as follows. If $u = \lambda$, then $Tu = T$. Let $u \neq \lambda$, $u = (f_{j_1}, \delta_1) \dots (f_{j_m}, \delta_m)$ and $i(1), \dots, i(m)$ are numbers taken from the set $\{1, \dots, n\}$ such that $\mu_T(i(1)) = f_{j_1}, \dots, \mu_T(i(m)) = f_{j_m}$. Then $\dim Tu = \dim T$, $\Delta(Tu) = \{(\sigma_1, \dots, \sigma_n) : (\sigma_1, \dots, \sigma_n) \in \Delta(T), \sigma_{i(1)} = \delta_1, \dots, \sigma_{i(m)} = \delta_m\}$, while ν_{Tu} is the restriction of the mapping ν_T to the set $\Delta(Tu)$ and μ_{Tu} is a mapping coinciding with μ_T . (In other words $T(f_{j_1}, \delta_1) \dots (f_{j_m}, \delta_m)$ is a subtable of the table T which contains only such rows which on intersection with columns labelled by f_{j_1}, \dots, f_{j_m} have numbers $\delta_1, \dots, \delta_m$ respectively.)

Denote by \mathcal{T} the set of all decision tables, while the set of all decision tables $T \in \mathcal{T}$ possessing the following property: either $\Delta(T) = \emptyset$ or there exists $m \in \omega$, such that $\nu_T(\delta) = m$ for each $\delta \in \Delta(T)$, will be denoted by \mathcal{TC} .

2.2 Decision Trees

A finite oriented tree containing exactly one node with no entering arcs will be called a *finite rooted oriented tree*. This singular node is commonly called *the root* of the finite rooted oriented tree. The nodes of the tree having no issuing arcs will be called *terminal* nodes. The non-terminal nodes of the tree will be called *work* nodes.

Let $\xi = w_1, d_1, \dots, w_m, d_m, w_{m+1}$ be a sequence of nodes and arcs of finite rooted oriented tree G such that w_1 is the root of G and w_{m+1} is a terminal node of G and for $i = 1, \dots, m$ the arc d_i issues from the node w_i and enters the node w_{i+1} . Then ξ will be called a *full path* in finite rooted oriented tree G .

A labeled finite rooted oriented tree will be called a *decision tree* if it satisfies the following conditions:

- a) every work node is labeled with an element from F ;
- b) every arc is labeled with a number from E_k , while the arcs issuing from one and the same node are labeled with distinct numbers;
- c) every terminal node is labeled with a number from ω .

Denote by \mathcal{D} the set of all possible decision trees. For $\Gamma \in \mathcal{D}$ we denote by $P(\Gamma)$ the set of elements from F used as labels at the work nodes of Γ , $\Omega(\Gamma) = \{(f_i, \delta) : f_i \in P(\Gamma), \delta \in E_k\}$ and by $\Xi(\Gamma)$ we denote the set of all full paths in the tree Γ .

Let us put into correspondence to a path $\xi, \xi \in \Xi(\Gamma)$, a word $\pi(\xi)$ from $\Omega(\Gamma)$. If the path ξ doesn't contain work nodes, then $\pi(\xi) = \lambda$. Let the path ξ contain $m > 0$ work nodes, $\xi = w_1, d_1, \dots, w_m, d_m, w_{m+1}$, and for $i = 1, \dots, m$ let the node w_i be labeled with element f_{j_i} , and the arc d_i be labeled with the number δ_i . Then $\pi(\xi) = (f_{j_1}, \delta_1) \dots (f_{j_m}, \delta_m)$.

Let $T \in \mathcal{T}$. A decision tree $\Gamma \in \mathcal{D}$ will be called a *decision tree for the table T* if it satisfies the following conditions:

- a) $P(\Gamma) \subseteq P(T)$;
- b) if $\Delta(T) \neq \emptyset$, then for each n -tuple $\bar{\delta} \in \Delta(T)$ there exists a full path $\xi \in \Xi(\Gamma)$, such that $\bar{\delta} \in \Delta(T\pi(\xi))$ and the terminal node in the path ξ is labeled with the number $\nu_T(\bar{\delta})$.

We will denote by $\mathcal{D}(T)$ the set of all decision trees for the table T .

2.3 Complexity Measures

A *complexity measure* is arbitrary computable function $\psi : F^* \rightarrow \omega$ such that for arbitrary words $\alpha, \beta \in F^*$ the following conditions satisfies:

- a) $\psi(\alpha) = 0$ if and only if $\alpha = \lambda$;
- b) $\psi(\alpha\beta) = \psi(\beta\alpha)$;
- c) $\psi(\alpha) \leq \psi(\alpha\beta)$;
- d) $\psi(\alpha\beta) \leq \psi(\alpha) + \psi(\beta)$.

Let $w : F \rightarrow \omega \setminus \{0\}$ be computable function. The following functions are complexity measures:

- 1) *the depth h* such that $h(\alpha) = |\alpha|$ for each word $\alpha \in F^*$, where $|\alpha|$ is the length of the word α ;

2) a *weighted depth* ψ_w such that $\psi_w(f_{j_1} \dots f_{j_m}) = \sum_{i=1}^m w(f_{j_i})$ for each nonempty word $f_{j_1} \dots f_{j_m} \in F^*$ and $\psi_w(\lambda) = 0$;

3) a function φ_w such that $\varphi_w(f_{j_1} \dots f_{j_m}) = \max\{w(f_{j_1}), \dots, w(f_{j_m})\}$ for each nonempty word $f_{j_1} \dots f_{j_m} \in F^*$ and $\varphi_w(\lambda) = 0$.

A *bounded complexity measure* is a complexity measure ψ such that $\psi(\alpha) \geq |\alpha|$ for each word $\alpha \in F^*$. For example, the depth and a weighted depth are bounded complexity measures.

If ψ_1 and ψ_2 are complexity measures then the functions ψ_3 and ψ_4 are complexity measures too where $\psi_3(\alpha) = \max(\psi_1(\alpha), \psi_2(\alpha))$ and $\psi_4(\alpha) = \psi_1(\alpha) + \psi_2(\alpha)$ for each word $\alpha \in F^*$. If also ψ_1 is bounded complexity measure then ψ_3 and ψ_4 are bounded complexity measures.

Extend a complexity measure ψ on the sets Ω , \mathcal{D} and \mathcal{T} . Let $\alpha \in \Omega$. For $\alpha = \lambda$ we certainly have $\psi(\alpha) = 0$. Let $\alpha \neq \lambda$ and $\alpha = (f_{j_1}, \delta_1) \dots (f_{j_m}, \delta_m)$. Then $\psi(\alpha) = \psi(f_{j_1} \dots f_{j_m})$. For $\Gamma \in \mathcal{D}$ we assume $\psi(\Gamma) = \max\{\psi(\pi(\xi)) : \xi \in \Xi(\Gamma)\}$. For $T \in \mathcal{T}$ we assume $\psi(T) = \min\{\psi(\Gamma) : \Gamma \in \mathcal{D}(T)\}$.

2.4 Uncertainty Measures

Let \mathbf{Q} be the set of all rational numbers. An *uncertainty measure* is arbitrary computable function $\gamma : \mathcal{T} \rightarrow \mathbf{Q}$ such that for each decision table $T \in \mathcal{T}$ if $\Delta(T) = \emptyset$ then $\gamma(T) = 0$ and if $\Delta(T) \neq \emptyset$ then the following conditions are satisfied:

a) $\gamma(T) \geq 0$;

b) for each $f_{i_0}, f_{i_1}, \dots, f_{i_m} \in P(T)$ and $\delta_0, \delta_1, \dots, \delta_m \in E_k$ the inequality $\gamma(T) - \gamma(T(f_0, \delta_0)) \geq \gamma(T(f_{i_1}, \delta_1) \dots (f_{i_m}, \delta_m)) - \gamma(T(f_{i_1}, \delta_1) \dots (f_{i_m}, \delta_m)(f_0, \delta_0))$ holds;

c) if $\gamma(T) < 1$ then $T \in \mathcal{TC}$, and if $T \in \mathcal{TC}$ then $\gamma(T) = 0$.

Let $T \in \mathcal{T}$. A table T' will be called a *separable sub-table of the table T* if there exists a word $\alpha \in \Omega(T)$ such that $T' = T\alpha$. The table T will be called a *boundary table* if $T \notin \mathcal{TC}$ and for each $f_i \in P(T)$ and $\delta \in E_k$ either $T(f_i, \delta) = T$ or $T(f_i, \delta) \in \mathcal{TC}$.

The following functions are uncertainty measures.

1) Function $R(x)$ where $R(T)$ is the number of non-ordered pairs of tuples $\bar{\delta}_1, \bar{\delta}_2$ from the set $\Delta(T)$ such that $\nu_T(\bar{\delta}_1) \neq \nu_T(\bar{\delta}_2)$.

2) Function $G(x)$ where $G(T)$ is the number of separable sub-tables of the table T which are boundary tables.

3) Function $H(x)$. Let $T \in \mathcal{T}$, $\dim T = n$, $\bar{\sigma}, \bar{\delta} \in \Delta(T)$, $\bar{\sigma} = (\sigma_1, \dots, \sigma_n)$, $\bar{\delta} = (\delta_1, \dots, \delta_n)$, $\{i : i \in \{1, \dots, n\}, \sigma_i = \delta_i\} = \{i_1, \dots, i_m\}$ and $\mu_T(i_1) = f_{j_1}, \dots, \mu_T(i_m) = f_{j_m}$. We will say that the pair $\bar{\sigma}, \bar{\delta}$ separates the table $T(f_{j_1}, \delta_{i_1}) \dots (f_{j_1}, \delta_{i_m})$. Then $H(T)$ is the number of non-ordered pairs of tuples $\bar{\sigma}, \bar{\delta} \in \Delta(T)$ such that $\nu_T(\bar{\sigma}) \neq \nu_T(\bar{\delta})$ and the pair $\bar{\sigma}, \bar{\delta}$ separates a boundary table.

One can prove that $G(T) \leq H(T) \leq R(T)$ for each table $T \in \mathcal{T}$.

If γ_1 and γ_2 are uncertainty measures then the function $\gamma_3 = p\gamma_1 + q\gamma_2$ is uncertainty measure where p and q are non-negative rational numbers such that $p + q \geq 1$.

It is not difficult to see that there exist algorithms with polynomial time complexity for computing functions $R(x)$ and $H(x)$ on the set \mathcal{T} . One can prove that for each table $T \in \mathcal{T}$ such that $\Delta(T) \neq \emptyset$ each separable boundary subtable of the table T is separated by some pair $\bar{\sigma}, \bar{\delta} \in \Delta(T)$ such that $\nu_T(\bar{\sigma}) \neq \nu_T(\bar{\delta})$. Using this fact it is not difficult to prove that there exists algorithm with polynomial time complexity for computing function $G(x)$ on the set \mathcal{T} .

3 Algorithms for Decision Tree Constructing

Let ψ be a bounded complexity measure and γ be an uncertainty measure. Algorithm $U_{\gamma, \psi}$ which constructs decision tree $U_{\gamma, \psi}(T)$ for given arbitrary table $T \in \mathcal{T}$ is dealt with in present section. Complexity and precision of the algorithm $U_{\gamma, \psi}$ are estimated.

3.1 Algorithm Description

Let us apply the algorithm $U_{\gamma, \psi}$ to a table $T \in \mathcal{T}$.

1-st step. Construct a tree consisting of a single node w .

Let $T \in \mathcal{TC}$. If $\Delta(T) = \emptyset$, then the node w will be assigned the number 0 as label. If $\Delta(T) \neq \emptyset$ then the node w will be labeled with number $\nu_T(\bar{\delta})$, where $\bar{\delta} \in \Delta(T)$. Proceed to the second step.

Let $T \notin \mathcal{TC}$. Label the node w with the word $\lambda \in \Omega(T)$ and proceed to the second step.

Suppose $t \geq 1$ steps have already been made. The tree obtained in the step t will be denoted by G .

(t + 1)-th step. If no one node of the tree G is labeled with word from $\Omega(T)$ then we denote the tree G by $U_{\gamma, \psi}(T)$. Operation of algorithm $U_{\gamma, \psi}$ is therewith completed.

Otherwise we choose certain node w in the tree G which is labeled with a word from $\Omega(T)$. Let the node w be labeled with word α .

If $T\alpha \in \mathcal{TC}$ then replace the word α as the label of the node w with the number $\nu_{T\alpha}(\bar{\delta})$ where $\bar{\delta} \in \Delta(T\alpha)$ and proceed to the $(t + 2)$ -th step.

Let $T\alpha \notin \mathcal{TC}$. Let σ_i be for every $f_i \in P(T)$ the minimal number from the set E_k for which $\gamma(T\alpha(f_i, \sigma_i)) = \max\{\gamma(T\alpha(f_i, \sigma)) : \sigma \in E_k\}$. Let $I\alpha = \{f_i : f_i \in P(T), \gamma(T\alpha) > \gamma(T\alpha(f_i, \sigma_i))\}$. For every $f_i \in I\alpha$ let $d(f_i) = \max\{\psi(f_i), \gamma(T\alpha) / (\gamma(T\alpha) - \gamma(T\alpha(f_i, \sigma_i)))\}$. Let p be the minimal number from ω for which $f_p \in I\alpha$ and $d(f_p) = \min\{d(f_i) : f_i \in I\alpha\}$. Assign the element f_p the node w as label instead of the word α . For every $\delta \in E_k$ such that $\Delta(T\alpha(f_p, \delta)) \neq \emptyset$ add the node $w(\delta)$ to the tree G and draw the arc from the node w to the node $w(\delta)$. This arc will be labeled with the number δ while the node $w(\delta)$ will be assigned the word $\alpha(f_p, \delta)$. Proceed to the $(t + 2)$ -th step.

Note that if the depth h have been taken as complexity measure then the value $d(f_i)$ in description of the algorithm $U_{\gamma, h}$ may be substituted by the value $\gamma(T\alpha(f_i, \sigma_i))$. The output of the algorithm remains the same.

Proposition 1. *Let ψ be a bounded complexity measure and γ be an uncertainty measure. Then for every table $T \in \mathcal{T}$ the work of algorithm $U_{\gamma,\psi}$ is completed in finite number of steps. The thereby constructed tree $U_{\gamma,\psi}(T)$ is a decision tree for the table T .*

3.2 Complexity Bounds

Consider an upper bound for number of steps made by algorithm $U_{\gamma,\psi}$ when constructing the tree $U_{\gamma,\psi}(T)$. By $|\Delta(T)|$ we will denote the cardinality of the set $\Delta(T)$.

Theorem 2. *Let ψ be a bounded complexity measure, γ be an uncertainty measure and let $T \in \mathcal{T}$. Then the process of constructing the tree $U_{\gamma,\psi}(T)$ by the algorithm $U_{\gamma,\psi}$ is carried out in at most $2|\Delta(T)| + 2$ steps.*

Detailed investigation of algorithm $U_{\gamma,\psi}$ complexity falls out of the scope of present work. Still the following sufficient condition of existence of polynomial upper bound for its time complexity can be obtained by analysis of the description of algorithm $U_{\gamma,\psi}$ on the basis of Theorem 2.

Let us suggest existence of algorithms computing the function ψ on the set F and the function γ on the set \mathcal{T} with polynomial time complexity. Then the algorithm $U_{\gamma,\psi}$ has polynomial time complexity. In particular, the algorithms $U_{R,h}, U_{G,h}, U_{H,h}$ satisfy the above condition.

3.3 Precision Bounds

For each word $\alpha \in \Omega$ we denote by $\chi(\alpha)$ the set of letters from the alphabet $\{(f_i, \delta) : f_i \in F, \delta \in E_k\}$ which are contained in α . Let ψ be a bounded complexity measure. Define the mapping $M_\psi : \mathcal{T} \rightarrow \omega$ as follows. Let $T \in \mathcal{T}$, $\dim T = n$ and $\mu_T(1) = f_{j_1}, \dots, \mu_T(n) = f_{j_n}$. For each $\bar{\delta} = (\delta_1, \dots, \delta_n) \in E_k^n$ denote $M_\psi(T, \bar{\delta}) = \min\{\psi(\alpha) : \alpha \in \Omega(T), \chi(\alpha) \subseteq \{(f_{j_1}, \delta_1), \dots, (f_{j_n}, \delta_n)\}, T\alpha \in \mathcal{TC}\}$. Then $M_\psi(T) = \max\{M_\psi(T, \bar{\delta}) : \bar{\delta} \in E_k^n\}$. The value of this mapping will be used in precision bounds for algorithm $U_{\gamma,\psi}$.

Proposition 3. *Let ψ be bounded complexity measure and $T \in \mathcal{T}$. Then $\psi(T) \geq M_\psi(T)$.*

The upper bound on the complexity of the decision tree constructed by the algorithm $U_{\gamma,\psi}$ is considered in the below statement. We denote $\psi_\gamma(T) = \psi(U_{\gamma,\psi}(T))$.

Theorem 4. *Let ψ be a bounded complexity measure, γ be an uncertainty measure and $T \in \mathcal{T}$. Then*

$$\psi_\gamma(T) \leq \begin{cases} M_\psi(T), & \text{if } M_\psi(T) \leq 1; \\ (M_\psi(T))^2 \ln \gamma(T) + M_\psi(T), & \text{if } M_\psi(T) \geq 2. \end{cases}$$

If the depth h is taken as complexity measure then the following statement holds.

Theorem 5. *Let γ be an uncertainty measure and $T \in \mathcal{T}$. Then*

$$h_\gamma(T) \leq \begin{cases} M_h(T), & \text{if } M_h(T) \leq 1; \\ M_h(T)(\ln \gamma(T) - \ln M_h(T) + 1), & \text{if } M_h(T) \geq 2. \end{cases}$$

If the depth h is taken as complexity measure and a function $\gamma \in \{R, G, H\}$ is taken as uncertainty measure then the following statement holds.

Theorem 6. *Let γ be an uncertainty measure from the set $\{R, G, H\}$ and $T \in \mathcal{T}$. Then*

$$h_\gamma(T) \leq \begin{cases} h(T), & \text{if } h(T) \leq 1; \\ h(T)(\ln \gamma(T) - \ln h(T) + 1), & \text{if } h(T) \geq 2. \end{cases}$$

The following statement allows us to estimate the quality of bounds from Theorem 5 and 6 for uncertainty measures from the set $\{R, G, H\}$.

Theorem 7. *Let γ be an uncertainty measure from the set $\{R, G, H\}$, $A_\gamma = \{(M_h(T), \gamma(T)) : T \in \mathcal{T}, \Delta(T) \neq \emptyset\}$ and $B_\gamma = \{(h(T), \gamma(T)) : T \in \mathcal{T}, \Delta(T) \neq \emptyset\}$. Then $A_\gamma = B_\gamma = \{(0, 0)\} \cup \{(m, r) : m, r \in \omega \setminus \{0\}, m \leq r\}$ and for each pair $(m, r) \in A_\gamma$ there exists a table $T(m, r) \in \mathcal{T}$ such that $M_h(T(m, r)) = h(T(m, r)) = m, \gamma(T(m, r)) = r$ and*

$$h_\gamma(T(m, r)) \geq \begin{cases} m, & \text{if } m < 2 \text{ or } r < 3m; \\ \lfloor (m - 1)(\ln r - \ln 3m) \rfloor + m, & \text{if } m \geq 2 \text{ and } r \geq 3m. \end{cases}$$

Theorem 7 implies that bounds of Theorem 5 and Theorem 6 for uncertainty measures from the set $\{R, G, H\}$ don't allow essential improvement. Non-existence of function $f : \omega \rightarrow \omega$, such that for each table $T \in \mathcal{T}$ the inequality $h_\gamma(T) \leq f(h(T))$ holds, follows additionally from Theorem 7 for each uncertainty measure $\gamma \in \{R, G, H\}$.

The situation with so called diagnostic decision tables is different. We will say that a table $T \in \mathcal{T}$ is *diagnostic decision table* if $\Delta(T) \neq \emptyset$ and for each $\bar{\sigma}, \bar{\delta} \in \Delta(T)$ such that $\bar{\sigma} \neq \bar{\delta}$ the relation $\nu_T(\bar{\sigma}) \neq \nu_T(\bar{\delta})$ holds. Diagnostic decision tables are meeting in pattern recognition [6] and fault diagnosis [9].

Proposition 8. *Let T be a diagnostic table from \mathcal{T} . Then*

$$h_R(T) \leq 2(h(T))^2 \ln k + h(T).$$

3.4 On Unimprovability of Some Algorithms

Let S be a set of N points and $\mathcal{F} = \{S_1, \dots, S_m\}$ a collection of subsets of S . *Set Covering problem* is the problem of selecting as few as possible subsets from \mathcal{F} such that every point in S is contained in at least one of the selected subsets. This problem is NP-hard.

In [2] it was proved that for each $0 < c < 1/4$, the Set Covering problem cannot be approximated within factor of $c \log_2 N$ in polynomial time unless $NP \subset DTIME(n^{\text{poly} \log_2 n})$. In [1] it was proved that for each $\varepsilon > 0$, the Set Covering problem cannot be approximated within factor of $(1 - \varepsilon) \ln N$ in polynomial time unless $NP \subset DTIME(n^{O(\log_2 \log_2 n)})$.

Let $T \in \mathcal{T}$. *Decision Tree Constructing problem* is the problem of searching decision tree with minimal depth for the table T . Using mentioned result from [1] it is not difficult to prove that for each $\varepsilon > 0$, the Decision Tree Constructing problem cannot be approximated within factor of $(1 - \varepsilon) \ln R(T)$ in polynomial time unless $NP \subset DTIME(n^{O(\log_2 \log_2 n)})$.

Taking into account that algorithms $U_{R,h}, U_{G,h}, U_{H,h}$ have polynomial time complexity, using inequalities $G(T) \leq H(T) \leq R(T)$ (which are true for each $T \in \mathcal{T}$) and using Theorem 6 we obtain that unless $NP \subset DTIME(n^{O(\log_2 \log_2 n)})$ then algorithms $U_{R,h}, U_{G,h}, U_{H,h}$ are close to unimprovable approximate polynomial algorithms for Decision Tree Constructing problem solving.

References

1. Feige, U.: A threshold of $\ln n$ for approximating set cover (Preliminary version). Proceedings of 28th Annual ACM Symposium on the Theory of Computing (1996) 314–318
2. Lund, C., Yannakakis, M.: On the hardness of approximating minimization problems. J. ACM **45**(5) (1994) 960–981
3. Moshkov, M.Ju.: On conditional tests. Dokl. Akad. Nauk SSSR **265** (1982) 550–552 (in Russian)
4. Moshkov, M.Ju.: Conditional tests. Edited by S.V. Yablonskii. Problemy Cybernetics **40**. Nauka Publishers, Moscow (1983) 131–170 (in Russian)
5. Moshkov, M.Ju.: Decision Trees. Theory and Applications. Nizhni Novgorod University Publishers, Nizhni Novgorod (1994) (in Russian)
6. Moshkov, M.Ju.: Complexity of decision trees for regular language word recognition. Preproceedings of the Second International Conference Developments in Language Theory. Magdeburg, Germany (1995)
7. Moshkov, M.Ju.: On the depth of decision trees over infinite information systems. Proceedings of the Congress "Information Processing and Management of Uncertainty in Knowledge-Based Systems". Granada, Spain (1996) 885–886
8. Moshkov, M.Ju.: On global Shannon functions of two-valued information systems. Proceedings of the Fourth International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery. Tokyo, Japan (1996) 142–143
9. Moshkov, M.Ju.: Diagnosis of constant faults of circuits. Proceedings of the Fourth International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery. Tokyo, Japan (1996) 325–327
10. Quinlan, J.R.: Discovering rules by induction from large collections of examples. Experts Systems in the Microelectronic Age. Edited by D. Michie. Edinburg University Press (1979)
11. Quinlan, J.R.: Induction of decision trees. Machine Learning **1** (1986) 81–106