Aplikace matematiky

Jaroslav Morávek
A geometrical method in combinatorial complexity

Aplikace matematiky, Vol. 26 (1981), No. 2, 82–96

Persistent URL: http://dml.cz/dmlcz/103900

SVAZEK 26 (1981) APLIKACE MATEMATIKY ČÍSLO 2

A GEOMETRICAL METHOD IN COMBINATORIAL COMPLEXITY

JAROSLAV MORÁVEK

(Received December 29, 1977)

INTRODUCTION

This paper continuates the author's research of [1] – [4]. In Section I we introduce a problem of classifying points of an n-dimensional linear space with respect to a finite nonempty family of polyhedral sets which covers the space. By using examples, the possibility of reduction of a wide class of practically relevant computational combinatorial problems to this classification problem is demonstrated. This class of problems contains e.g. many well-known problems of sorting, searching and discrete optimization.

In Section II a set of formal algorithms for solving the above classification problem is introduced. The aim of this definition is to formalize the intuitive concept of algorithm operating over real-valued data and composed from additions, subtractions, multiplications by real constants and comparisons, as the unique elementary operations (elementary steps.) This concept of formal algorithm is essentially the same as that from the previous author's papers ([1] – [3]: linear separating algorithm, [4]: localization algorithm). Let us mention the major modifications:

1) The previous definition of an algorithm was based essentially on the language of the graph theory. In this paper, the definition of the algorithm is based on an algebraic language of strings over a 3-element alphabet.

2) In [1] – [3] the polyhedral sets corresponding to the classification problem are defined by using only linear homogeneous functions, whereas in this paper more general linear affine functions are used. Thus we discuss in [1] – [3] only polyhedral cones instead of more general polyhedral sets discussed in this paper. In accordance with the last fact, the algorithms discussed in this paper compare the values of arbitrary linear affine functions, whereas in [1] – [3] only comparisons of linear homogeneous functions are allowed as elementary steps.

3) The classification problem discussed here is a generalization of the classification problem discussed in [4] and called there the localization problem. In [4] the space

is divided only into two polyhedral sets: An arbitrary solid convex polyhedral set and its complement.

Identically with [1]–[4], the measure of complexity of an algorithm is introduced as the maximum number of required comparisons; the maximum is taken over all input data.

Section II is concluded by a theorem concerning existence of an algorithm for solving the general classification problem of Section I. This result is a generalization of the existence theorem from [2].

The main result of the paper is contained in Section III: A general lower bound for the number of comparisons required by an algorithm. This lower bound depends, roughly speaking, on the minimum number of convex parts into which polyhedral sets of the classification problem can be divided. It is shown by using an example that the derived lower bound is exact.

In the concluding section IV, the use of the general lower bound from Section III is illustrated by the knapsack problem. This yields a lower bound for the number of comparisons required by this problem. This result was obtained originally by the present author in [1] (1967) and [2] (1969), see also the monograph [5], p. 428. The same lower bound for the knapsack problem was rediscovered recently by Dobkin and Lipton [6] and [18].

I. THE CLASSIFICATION PROBLEM

1.1. Let E^n denote an n-dimensional linear space over the field of real numbers R. A function $f : E^n \to R$ is said to be *linear affine* if

$$\forall \mathbf{x}, \mathbf{y} \in E^n \; \forall \lambda \in R (f(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) = \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y})) .$$

A subset $H \subset E^n$ is called a *hyperplane* in E^n if there exists a non-constant linear affine function f such that

$$H = \{\mathbf{x} \in E^n \mid f(\mathbf{x}) = 0\} .$$

A subset $G \subset E^n$ is called a *halfspace* in E^n if there exists a non-constant linear affine function f such that either:

$$G = \{\mathbf{x} \in E^n \mid f(\mathbf{x}) > 0\}$$

or:

$$G = \{\mathbf{x} \in E^n \mid f(\mathbf{x}) \geqq 0\} .$$

In the first case G is called an *open halfspace*, in the other a *closed halfspace*.

A subset $C \subset E^n$ is called a *simple polyhedral set* (abbreviation SPS) if C can be expressed as an intersection of a finite (including void) family of halfspaces. It follows from this definition that, in particular, $\emptyset$ and E^n are SPS-s.

A subset $S \subset E^n$ is called a *polyhedral set* (abbreviation PS) if S can be expressed as a union of a finite (including void) family of SPS-s. It follows from this definition that, in particular, $\emptyset$, E^n and each SPS are PS-s. Let us notice that the set of all PS-s is an algebra of subsets of E^n, generated by the set of all halfspaces of E^n*).

A subset $M \subset E^n$ is called *convex* if

$$\forall \mathbf{x}, \mathbf{y} \in M \ \forall \lambda \in R(0 \leqq \lambda \leqq 1 \Rightarrow \lambda \mathbf{x} + (1 - \lambda)\, \mathbf{y} \in M) .$$

In particular, $\emptyset$ and E^n are convex, and each SPS is convex. (It is not true, however, that each convex PS is SPS.)

In E^n we assume the *usual* topology, i.e. the coarsest topology with respect to which all linear affine functions are continuous. Equivalently speaking, the usual topology in E^n is generated by the set of all open halfspaces as a subbase. The closure of a set $M \subset E^n$ with respect to the usual topology will be denoted by $Cl(M)$ and the interior of M by $Int(M)$. Let us notice that the terms open and closed halfspaces introduced above are in accordance with this topological terminology.

A subset $M \subset E^n$ is called *connected* (with respect to the usual topology) if there exists no pair of nonempty sets X, Y such that $M = X \cup Y$ and $Cl(X) \cap Y = X \cap \cap Cl(Y) = \emptyset$. Given a set $P \subset E^n$, a nonempty subset $P_0 \subset P$ is called a *connected component* of P if P_0 is a maximal (with respect to the inclusion) connected subset of P, cf. [7]. Observe that each convex set in E^n is connected.

In this paper several examples of the general theory are discussed. In most of them we set $E^n := R^n$, where R^n denotes the usual n-dimensional arithmetical space, elements of which are n-tuples of real numbers.

1.2. Let $\mathfrak{S} = \{S_\iota\}_{\iota \in I}$ be a finite non-void indexed family of PS-s, satisfying the condition

$$\bigcup_{\iota \in I} S_\iota = E^n . \tag{1}$$

Our aim is to discuss the computational complexity of the following computational problem, introduced essentially by the present author in [1], cf. [2]:

Given an arbitrary element $\mathbf{x} \in E^n$, one is asked to determine an $\iota \in I$ such that $\mathbf{x} \in S_\iota$. (This subscript ι is not determined uniquely, in general, since we do not assume that $\mathfrak{S}$ is a disjoint decomposition.)

The stated problem will be called the *problem of classification of points* $\mathbf{x} \in E^n$ *with respect to* $\mathfrak{S}$, briefly the *classification problem* (*for* $\mathfrak{S}$), or the $\mathfrak{S}$-*problem*.

In terms of the language of the data processing the S-problem can be briefly stated as follows (cf. [8]):

DATA: $\mathbf{x} \in E^n$

PROBLEM: Determine an $\iota \in I$ such that $\mathbf{x} \in S_\iota$.

---

*) Our definition of the polyhedral set is more general than the usual one, according to which a polyhedral set is connected and closed.

This way of simplified formulations of computational problems will be frequently used throughout the rest of this paper.

1.3. A wide class of practically relevant computational problems can be reduced to the $\mathfrak{S}$-problem, as e.g. various problems of sorting, searching and combinatorial optimization. This idea as well as typical methods of such reductions are illustrated by the following examples:

Example 1. *Finding the k-th minimal element* (cf. [9]–[11]).

DATA: $(a_1, a_2, \ldots, a_n) \in R^n$

PROBLEM: Determine $r \in \{1, 2, \ldots, n\}$ such that there exists $J \subset (\{1, 2, \ldots, n\} \setminus \{r\})$ satisfying the following conditions:

1) $\operatorname{card}(J) = k - 1$
2) $a_j \leqq a_r$ for all $j \in J$
3) $a_j \geqq a_r$ for all $j \in \{1, \ldots, n\} \setminus J$

In the formulation of this problem it is assumed that n and k are given positive integers, $n \geqq k$. The element a_r is called the *k-th minimal element among* $a_1, a_2, \ldots, a_n$. To demonstrate the reduction of this problem to an appropriate $\mathfrak{S}$-problem we set

$$I := \{1, 2, \ldots, n\} ; \qquad E^n := R^n ;$$

$$\boldsymbol{x} := (a_1, a_2, \ldots, a_n) ; \qquad \iota := r ;$$

$$S_\iota := S_r := \{(a_1, a_2, \ldots, a_n) \in R^n \mid a_r \text{ is the } k\text{-th minimal element among } a_1, a_2, \ldots, a_n\} .$$

It is easy to see that S_r are PS-s, and the indexed family $\{S_r\}_{r=1}^n$ satisfies condition (1). Moreover, $(a_1, a_2, \ldots, a_n) \in S_r$ if and only if a_r is the k-th minimal element, which completes the proof of the reducibility.

Remark. The special case for $|k - \frac{1}{2}(n + 1)| < 1$ of this problem is called the *median problem*.

Example 2. *Travelling-salesman problem* (see e.g. [12] or [13]).

DATA: $p \times p$ real-valued matrix

$$\mathbb{A} = \begin{pmatrix} 0, & a_{12}, & \ldots, & a_{1p} \\ a_{21}, & 0, & \ldots, & a_{2p} \\ \ldots & \ldots & \ldots & \ldots \\ a_{p1}, & a_{p2}, & \ldots, & 0 \end{pmatrix}$$

having all diagonal entries zero; p is a given positive integer.

PROBLEM: Determine a permutation $(i_1, i_2, \dots, i_{p-1})$ of $\{1, 2, \dots, p-1\}$ such that

$$a_{i_1 i_2} + a_{i_2 i_3} + \dots + a_{i_{p-1} p} + a_{p i_1} \leqq a_{j_1 j_2} + a_{j_2 j_3} + \dots + a_{j_{p-1} p} + a_{p j_1} \text{ for all permutations } (j_1, j_2, \dots, j_{p-1}) \text{ of } \{1, 2, \dots, p-1\}.$$

In order to demonstrate the reducibility of the travelling salesman problem to an $\mathfrak{S}$-problem we set:

$n := p(p-1)$;

$E^n :=$ the natural linear space of all matrices $\mathbb{A}$ (isomorphic to $R^{p(p-1)}$);

$\iota :=$ permutation $(i_1, i_2, \dots, i_{p-1})$ of $\{1, 2, \dots, p-1\}$

$I :=$ the set of all permutations of $\{1, 2, \dots, p-1\}$;

$S_\iota := \{\mathbb{A} \mid a_{i_1 i_2} + \dots + a_{p i_1} \leqq a_{j_1 j_2} + \dots + a_{p j_1}$ for all permutations $(j_1, j_2, \dots, j_{p-1})$ of $\{1, 2, \dots, p-1\}\}$.

Now S_ι is a PS*) for each $\iota \in I$ and the indexed family $\mathfrak{S} = \{S_\iota\}_{\iota \in I}$ satisfies condition (1) since for each $\mathbb{A}$ there is a permutation $(j_1, j_2, \dots, j_{p-1})$ of $\{1, 2, \dots, p-1\}$ which is the solution of the corresponding travelling salesman problem. Finally, $\mathbb{A} \in S_\iota$ is equivalent to the assertion: '$\iota = (i_1, i_2, \dots, i_{p-1})$ is the solution of the corresponding travelling salesman problem.'

This proves the reducibility.

Example 3. The following problem is closely related to the so called *knapsack problem* (see e.g. [14]).

DATA: $(a_1, a_2, \dots, a_m, a) \in R^{m+1}$

PROBLEM: Is there $(x_1, x_2, \dots, x_m) \in \{0, 1\}^m \stackrel{\text{def}}{:=} \underbrace{\{0, 1\} \times \dots \times \{0, 1\}}_{m}$ such that

$$\sum_{j=1}^{m} a_j x_j = a\,?$$

In this problem m is a given positive integer.

This problem itself is also frequently called the *knapsack problem* (see e.g. [8]). For the sake of brevity we use this simplified terminology in this paper.

In order to demonstrate the reducibility of the knapsack problem to a corresponding $\mathfrak{S}$-problem we set

$$n := m + 1\,; \quad E^n := R^{m+1}\,; \quad I := \{0, 1\}\,;$$

$$\mathbf{x} := (a_1, a_2, \dots, a_m, a) \quad \text{and} \quad \mathfrak{S} := \{S_0, S_1\}\,,$$

*) Actually S_ι is a SPS in this special case.

where

$$S_0 := \{(a_1, a_2, \dots, a_m, a) \in R^{m+1} \mid \text{There exists an } m\text{-tuple}$$
$$(x_1, x_2, \dots, x_m) \in \{0, 1\}^m \text{ such that } \sum_{j=1}^{m} a_j x_j = a\} ;$$
$$S_1 := R^{m+1} \setminus S_0 .$$

Similarly, the general integer linear programming problem with bounded variables can be reduced to an $\mathfrak{S}$-problem. This reduction essentially follows from [2].

II. LINEAR COMPARISON ALGORITHMS

2.1. Let $W \overset{\text{def}}{=} \{1, 0, -1\}$ and let W^* denote the set of all *strings* over W, where a string over W is a finite (including void) sequence $w_1 w_2, \dots, w_l$ of some elements of W written without commas and parentheses. In particular, W^* contains the void string, denoted in this paper by Θ, which is obtained from $w_1 w_2, \dots, w_l$ by setting $l = 0$. Number l is called the length of string $w_1 w_2 \dots w_l$.

2.2. A finite non-void subset $\mathsf{T} \subset W^*$ is called a *trichotomical tree* (for the sake of brevity we shall use the term *tree*) if T has the following properties:

If $w_1 w_2 \dots w_l \in \mathsf{T}$ and $l > 0$ then:

a) $w_1 w_2 \dots w_\lambda \in \mathsf{T}$ for all $\lambda = 0, 1, \dots, l - 1$ and

b) $w_1 w_2 \dots w_{l-1} w \in \mathsf{T}$ for all $w \in W$.

In particular, it follows from this definition that $\Theta \in \mathsf{T}$ for each tree T.

The number $\delta(\mathsf{T}) \overset{\text{def}}{=} \max \{l \mid w_1, \dots, w_l \in \mathsf{T}\}$ is called the *depth* of T, i.e. $\delta(\mathsf{T})$ is the maximum length of a string in T.

A string $w_1 w_2 \dots w_l \in \varphi$ is said to be *nonfinal* if there is a $w \in W$ such that $w_1 w_2 \dots$ $\dots w_l w \in \mathsf{T}$; on the contrary $w_1 w_2, \dots, w_l$ is called final. Let us denote by Tnf and Tf the set of all nonfinal and final strings of T, respectively.

A final string $w_1 w_2 \dots w_l \in \mathsf{Tf}$ is said to be *regular* if $w_\lambda \neq 0$ for all $\lambda = 1, 2, \dots, l$; let us denote by Tr the set of all regular strings of T.

2.3. **Lemma.** *For each tree* T *the following inequality holds*:

$$\operatorname{card}(\mathsf{Tr}) \leqq 2^{\delta(\mathsf{T})} . \qquad \square$$

The assertion of this lemma can be easily proved by induction with respect to the depth of T.

2.4. Let us denote by F the set of all non-constant linear affine functions $f : E^n \to R$. In order to define a formal algorithm for the solution of the introduced classification problem we shall first assign elements of F to nonfinal strings of the tree.

Definition. An ordered pair (T, φ), where T is a tree and where $\varphi : \mathsf{Tnf} \to F$, is called a *linear comparison algorithm* over E^n (abbreviation *LCA*).

Our final step will consist in connecting the concept of LCA with an $\mathfrak{S}$-problem. First, we introduce an auxiliary notation: For $w_1 w_2, \ldots, w_l \in \mathsf{T}$ set

$$E(w_1 w_2 \ldots w_l) := \{\mathbf{x} \in E^n \mid \bigwedge_{\lambda=1}^{l} \operatorname{sign}(\varphi(w_1 \ldots w_{\lambda-1})(\mathbf{x})) = w_\lambda\}$$

$$= \bigcap_{\lambda=1}^{l} \{\mathbf{x} \in E^n \mid \operatorname{sign}(\varphi(w_1 \ldots w_{\lambda-1})(\mathbf{x})) = w_\lambda\},$$

where function sign : $R \to R$ is defined as follows:

$$\operatorname{sign}(y) = 1 \quad \text{if} \quad y > 0, \quad \operatorname{sign}(y) = 0 \quad \text{if} \quad y = 0$$

and

$$\operatorname{sign}(y) = -1 \quad \text{if} \quad y < 0.$$

The sets $E(w_1 w_2 \ldots w_l)$ are obviously SPS-s. Set $E(w_1 w_2 \ldots w_l)$ will be called an *output set* of (T, φ) if $w_1 w_2 \ldots w_l \in \mathsf{Tf}$.

2.5. **Lemma.** *The indexed family of all output sets of* T, *is a partition of* E^n, *i.e.*

$$E^n = \bigcup \{E(w_1 w_2 \ldots w_l) \mid w_1 w_2, \ldots, w_l \in \mathsf{Tf}\}$$

and

$$E(w_1 w_2 \ldots w_l) \cap E(\tilde{w}_1 \tilde{w}_2 \ldots \tilde{w}_{\tilde{l}}) = \emptyset \quad \text{if} \quad w_1, \ldots, w_l \neq \tilde{w}_1, \ldots, \tilde{w}_{\tilde{l}}.$$

Moreover, $E(w_1 w_2 \ldots w_l)$ *is open if* $w_1 w_2 \ldots w_l \in \mathsf{Tr}$, *and* $E(w_1 w_2 \ldots w_l)$ *is nowhere dense if* $w_1 w_2 \ldots w_l \in \mathsf{Tf} \setminus \mathsf{Tr}$. □

(The proof is obvious.)

2.6. **Definition.** An ordered triplet $\mathscr{A} = (\mathsf{T}, \varphi, \psi)$, where (T, φ) is an LCA over E^n and where $\psi : \mathsf{Tf} \to I$, will be called a *linear comparison algorithm for the* $\mathfrak{S}$*-problem* (or briefly: *LCA for* $\mathfrak{S}$) if $\mathscr{A}$ satisfies the following condition:

(2) $E(w_1 w_2 \ldots w_l) \subset S_\iota$ if $w_1, \ldots, w_l \in \mathsf{Tf}$ and $\iota = \psi(w_1, \ldots, w_l)$.

The set of all LCA-s for $\mathfrak{S}$ will be denoted by $\mathfrak{A}\langle\mathfrak{S}\rangle$.

2.7. An LCA for $\mathfrak{S}$ can be informally interpreted as a computing procedure for the solution of the $\mathfrak{S}$-problem, controlled by the following set of rules:

START:	from Θ;
CHECKING:	Check the condition $w_1 w_2, \ldots, w_l \in \mathsf{Tf}$; If $w_1 w_2 \ldots w_l \in \mathsf{Tf}$ go to STOP;
COMPARING:	Compute $w_{l+1} := \operatorname{sign}(\varphi(w_1 w_2 \ldots w_l)(\mathbf{x}))$, replace the string $w_1, \ldots, w_l$ by $w_1, \ldots, w_l w_{l+1}$ and to to CHECKING;
STOP:	Compute $\iota := \psi(w_1 w_2 \ldots w_l)$ and halt; ι yields the solution of the $\mathfrak{S}$-problem.

2.8. **Definition.** Let $\mathscr{A} = (\mathsf{T}, \varphi, \psi) \in \mathfrak{A}\langle\mathfrak{S}\rangle$. The number

$$comp(\mathscr{A}) \overset{\text{def}}{=} comp(\mathsf{T}, \varphi, \psi) \overset{\text{def}}{=} \delta(\mathsf{T})$$

will be called the *measure of complexity* of $\mathscr{A}$.

From the point of view of the informal interpretation 2.7, $comp(\mathscr{A})$ corresponds to the maximum number of all comparisons (i.e. evaluations of the function sign (.)), required by $\mathscr{A}$ in the process of computation, where the maximum is taken over all $\boldsymbol{x} \in E^n$.

2.9. Example of LCA. By the following simple example we show how a natural computing procedure can be converted into the formal language of Definitions 2.6. and 2.8. Let us consider the following special case of Example 1 from 1.3:

DATA: $(a_1, a_2, a_3) \in R^3$;

PROBLEM: Find the minimal element among a_1, a_2, a_3.

For the solution of this problem we shall use the following algorithm written in ALGOL 60:

```
if a1 ≧ a2 then begin if a2 > a3 then goto A3
                           else goto A2
                end
if a1 ≧ a3 then goto A3 else goto A1;
.........................................
Ai : ... comment ai is the minimum element among a1, a2, a3;
```

In order to convert this procedure into the formal language of Definition 2.6 we introduce first linear affine functions $f_1, f_2, f_3 : R^3 \to R$ as follows:

$$f_1(a_1, a_2, a_3) \overset{\text{def}}{=} a_1 - a_2\,; \quad f_2(a_1, a_2, a_3) \overset{\text{def}}{=} a_2 - a_3\,;$$
$$f_3(a_1, a_2, a_3) \overset{\text{def}}{=} a_1 - a_3$$

for $(a_1, a_2, a_3) \in R^3$.

Now, let us set:

$$^0\mathsf{T} := \text{the set of all strings over } W \text{ of lengths at most } 2;$$
$$^0\varphi(\Theta) := f_1\,; \quad {}^0\varphi(1) := {}^0\varphi(0) := f_2\,; \quad {}^0\varphi(-1) := f_3\,;$$
$$^0\psi(11) := {}^0\psi(01) := {}^0\psi(-11) := {}^0\psi(-10) := 3\,;$$
$$^0\psi(10) := {}^0\psi(1-1) := {}^0\psi(00) := {}^0\psi(0-1) := 2\,;$$
$$^0\psi(-1-1) := 1\,.$$

The verification of the fact that $({}^0\mathsf{T}, {}^0\varphi, {}^0\psi)$ is an LCA for the classification problem of finding the minimum element among a_1, a_2, a_3 requires the checking of validity of condition (2) of Definition 2.6 for all outputs sets

$$ {}^0\mathsf{E}(w_1 w_2) \quad \text{of} \quad ({}^0\mathsf{T}, {}^0\varphi, {}^0\psi) \, . $$

(Example:

$$ {}^0\mathsf{E}(1-1) = \{(a_1, a_2, a_3) \in R^3 \mid {}^0\varphi(\Theta)(a_1, a_2, a_3) > 0 \, , \quad {}^0\varphi(1)(a_1, a_2, a_3) < 0\} = $$
$$ = \{(a_1, a_2, a_3) \in R^3 \mid a_1 - a_2 > 0 \, , \ a_2 - a_3 < 0\} = $$
$$ = \{(a_1, a_2, a_3) \in R^3 \mid a_2 \leqq \min(a_1, a_3)\} \, , $$

which is in accordance with ${}^0\psi(1-1) = 2$.

2.10. In the conclusion of this section we shall discuss the question of existence of LCA for $\mathfrak{S}$. We shall prove a result which generalizes an existence theorem of [2], where the sets S_ι in the $\mathfrak{S}$-problem are polyhedral cones.

Theorem. *For each $\mathfrak{S}$-problem we hawe $\mathfrak{A}\langle\mathfrak{S}\rangle \neq \emptyset$, i.e., for each $\mathfrak{S}$-problem there exists an* LCA *for $\mathfrak{S}$.*

Proof. Each set S_ι of $\mathfrak{S}$ is a union of a finite family of SPS-s, and each of these SPS-s is an intersection of a finite family of halfspaces. Let $\{H_1, H_2, \ldots, H_t\}$ be the set of all boundary hyperplanes of these halfspaces. For each $\tau = 1, 2, \ldots, t$ there exists $f_\tau \in F$ such that

$$ H_\tau = \{\mathbf{x} \in E^n \mid f_\tau(\mathbf{x}) = 0\} \, . $$

Now, for each string $w_1 w_2, \ldots, w_t$ over W having the length t we set

$$ G(w_1 w_2 \ldots w_t) := \bigcap_{\tau=1}^{t} \{\varphi \in E^n \mid \operatorname{sign}(f_\tau(\mathbf{x})) = w_\tau\} $$

and consider the finite indexed family

$$ \mathfrak{G} := \{G(w_1 w_2 \ldots w_t) \mid w_1 w_2, \ldots, w_t \text{ is an arbitrary string of the length } t\} \, . $$

It follows immediately from the definition of $\mathfrak{G}$ that $\mathfrak{G}$ is a partition of E^n and

$$ (3) \qquad \forall G(w_1, \ldots, w_t) \in \mathfrak{G} \quad \exists \iota \in I \quad (G(w_1, \ldots, w_t) \subset S_\iota) \, . $$

Now, let us set ${}^e\mathscr{A} := ({}^e\mathsf{T}, {}^e\varphi, {}^e\psi)$, where

1) ${}^e\mathsf{T} :=$ the set of all string of W^* of the length $\leqq t$;
2) ${}^e\varphi(w_1 w_2 \ldots w_{\tau-1}) := f_\tau$ for $\tau = 1, 2, \ldots, t$;
3) ${}^e\psi(w_1 w_2 \ldots w_t) := \iota$, where $\iota \in I$ is chosen arbitrarily but to satisfy the condition $G(w_1 w_2 \ldots w_t) \subset S_\iota$; the satisfiability of this condition follows from (3).

It is easy to see that ${}^e\mathscr{A} \in \mathfrak{A}\langle\mathfrak{S}\rangle$. Indeed, let us notice that $\mathfrak{G}$ is equal to the indexed family of all output sets of $\langle {}^e\mathsf{T}, {}^e\varphi \rangle$, hence condition (3) guarantees the validity of condition (2) of Definition 2.6. □

3.1. Now we can state the following problem. One is asked to determine $\mathscr{A}_* \in \mathfrak{A}\langle\mathfrak{S}\rangle$ such that

$$comp(\mathscr{A}_*) = \min\{comp(\mathscr{A}) \mid \mathscr{A} \in \mathfrak{A}\langle\mathfrak{S}\rangle\}$$

The algorithm $\mathscr{A}_*$ is called the *optimum LCA for solving the $\mathfrak{S}$-problem* (briefly: *optimum LCA for $\mathfrak{S}$*).

The problem of finding an optimum LCA for a general $\mathfrak{S}$ seems to be extremely difficult. Thus we must be satisfied with some particular results, e.g. solving the problem for particular but interesting $\mathfrak{S}$, or obtaining bounds for $comp(\mathscr{A}_*)$. The results of both of these types were obtained by the author in [1]–[4].

It is the main purpose of this paper to derive a new lower bound for $comp(\mathscr{A}_*)$. This lower bound depends, roughly speaking, on the minimum number of convex parts into which one can decompose PS-s S_ι of $\mathfrak{S}$.

3.2. **Definition.** Let $M \subset E^n$ and let $\mathfrak{X} = \{X_\alpha\}_{\alpha\in A}$ be an indexed family of convex sets $X_\alpha \subset E^n$. $\mathfrak{X}$ will be called a *convex generating family* of M if

$$\bigcup_{\alpha\in A} X_\alpha \subset M \subset \bigcup_{\alpha\in A} Cl(X_\alpha)\,.$$

The minimum cardinality of a convex generating family of M will be called the *index of convexity* of M and denoted by $ic(M)$.

3.3. **Lemma.** *For each $M \subset E^n$:*

(i) $ic(M) = 0$ *if and only if* $M = \emptyset$,
(ii) $ic(M) = 1$ *if M is convex and* $M \neq \emptyset$,
(iii) $ic(M)$ *is finite if M is a PS,*
(iv) $ic(M) \geqq k$, *where k is the cardinality of the set of all connected components of M,*
(v) $ic(M)$ *equals the cardinality of the set of all connected components of M if each connected component is convex.*

Proof. Parts (i)–(iii) are obvious. To prove (iv) we assume by contradiction that $ic(M) < k$. Let $\mathfrak{X} = \{X_\alpha\}_{\alpha\in A}$ be a convex generating family of M such that $\operatorname{card}(A) = ic(M)$ and let $\{C_\beta\}_{\beta\in B}$ be the set of all connected components of M, hence $\operatorname{card}(B) = k$. Since each X_α for $\alpha \in A$ is a convex and therefore connected subset of M we have: For each $\alpha \in A$ there exists at most one $\beta \in B$ such that $Cl(X_\alpha) \cap C_\beta \neq \emptyset$ (actually, $X_\alpha \subset C_\beta$ for this β). However, $\operatorname{card}(A) < \operatorname{card}(B)$. Hence there exists a $\beta_0 \in B$ such that $Cl(X_\alpha) \cap C_{\beta_0} = \emptyset$ for all $\alpha \in C$. Thus

$$C_{\beta_0} \subset M \setminus \bigcup_{\alpha\in A} Cl(X_\alpha) = \emptyset\,,$$

which contradicts $C_{\beta_0} \neq \emptyset$, thus completing the proof of (iv). Assertion (v) follows immediately from (iv). □

3.4. **Definition.** Let $\mathfrak{S} = \{S_\iota\}_{\iota \in I}$ be an indexed family of PS-s satisfying condition (1). $\mathfrak{S}$ is said to be a *quasipartition* (of E^n) if

$$Int(S_\iota) \cap Int(S_\varkappa) = \emptyset \quad \text{for} \quad \iota \neq \varkappa .$$

In particular, it follows from this definition that each $\mathfrak{S}$ which is a partition of E^n is also a quasipartition. Moreover, it is easy to see that the families $\mathfrak{S}$ from Examples 1–3 of 1.3 are quasipartitions.

3.5. **Lemma.** *Let $\mathfrak{S}$ be a quasipartition of E^n, and let $\mathscr{A} = (\mathsf{T}, \varphi, \psi) \in \mathfrak{A}\langle\mathfrak{S}\rangle$. Then for each regular string $w_1 w_2 \dots w_l \in \mathrm{Tr}$ and for each $\iota \in I$ the following implication holds:*

$$E(w_1, \dots, w_l) \cap Int(S_\iota) \neq \emptyset \Rightarrow E(w_1, \dots, w_l) \subset Int(S_\iota) .$$

Proof. Letting $\varkappa := \psi(w_1 w_2 \dots w_l)$, we have from condition (2) of Definition 2.6

$$E(w_1 w_2, \dots, w_l) \subset S_\varkappa .$$

However, $E(w_1, \dots, w_l)$ is open since $w_1, \dots, w_l \in \mathrm{Tr}$ (Lemma 2.5), hence $E(w_1, \dots, w_l) \subset Int(S_\varkappa)$. Finally we have

$Int(S_\varkappa) \cap Int(S_\iota) \subset E(w_1 w_2 \dots w_l) \cap Int(S_\iota) \neq \emptyset$, which yields $\varkappa = \iota$, thus completing the proof. □

3.6. **Theorem.** *Let $\mathfrak{S} = \{S_\iota\}_{\iota \in I}$ be a quasipartition of E^n. Then for each $\mathscr{A} \in \mathfrak{A}\langle\mathfrak{S}\rangle$*

$$comp(\mathscr{A}) \geqq \left]\log_2 \left(\sum_{\iota \in I} ic(Int(S_\iota))\right)\right[,$$

where $].[: R \to R$ is defined as follows:

For each $y \in R$, $]y[:=$ minimum integer z such that $z \geqq y$.

Proof. Let $\mathscr{A} = (\mathsf{T}, \varphi, \psi) \in \mathfrak{A}\langle\mathfrak{S}\rangle$ and let Tr be the set of all regular strings of φ. In view of Lemma 2.3,

$$\log_2 (\mathrm{card}\, (\mathrm{Tr})) \leqq \delta(\mathsf{T}) = comp(\mathscr{A}) .$$

Since, moreover $comp(\mathscr{A}) = \delta(\mathsf{T})$ is an integer, it is sufficient to prove the inequality

$$\mathrm{card}\, (\mathrm{Tr}) \geqq \sum_{\iota \in I} ic(Int(S_\iota)) .$$

Let us assume on the contrary that

$$\mathrm{card}\, (\mathrm{Tr}) < \sum_{\iota \in I} ic(Int(S_\iota)) ,$$

and for each $\iota \in I$ let

$$\mathrm{Tr}^{(\iota)} := \{w_1 w_2 \dots w_l \in \mathrm{Tr} \mid E(w_1, \dots, w_l) \cap Int(S_\iota) \neq \emptyset\} .$$

In view of Lemma 3.5 we have

$$\mathrm{Tr}^{(\iota)} = \{w_1, \ldots, w_l \in \mathrm{Tr} \mid E(w_1, \ldots, w_l) \subset Int(S_\iota)\}\,.$$

Since $\mathfrak{S}$ is a quasipartition the sets $\mathrm{Tr}^{(\iota)}$ are pairwise disjoint, and hence

$$\sum_{\iota \in I} \mathrm{card}\,(\mathrm{Tr}^{(\iota)}) \leqq \mathrm{card}\,(\mathrm{Tr}) < \sum_{\iota \in I} ic(Int(S_\iota))\,.$$

Thus, there exists a $\mu \in I$ such that

$$\mathrm{card}\,(\mathrm{Tr}^{(\mu)}) < ic(Int(S_\mu))\,. \tag{4}$$

Furthermore, it follows from the definition of $\mathrm{Tr}^{(\iota)}$ that

$$S_\mu \supset \bigcup \{E(w_1 w_2 \ldots w_l) \mid w_1 w_2 \ldots w_l \in \mathrm{Tr}^{(\mu)}\}\,.$$

But $E(w_1, \ldots, w_l)$ is open for $w_1 w_2 \ldots w_l \in \mathrm{Tr}^{(\mu)}$ (Lemma 2.5), hence

$$Int(S_\mu) \supset \bigcup \{E(w_1, \ldots, w_l) \mid w_1, \ldots, w_l \in \mathrm{Tr}^{(\mu)}\}\,. \tag{5}$$

On the other hand, it follows from the definition of $\mathrm{Tr}^{(\mu)}$ that

$$Int(S_\mu) \setminus \bigcup_{\mathrm{Tr}^{(\mu)}} E(w_1 w_2 \ldots w_l) \subset \bigcup_{\mathrm{Tf} \setminus \mathrm{Tr}} E(w_1 w_2 \ldots w_l)\,.$$

The set on the right-hand side of the above inclusion is nowhere dense since it is a finite union of nonwhere dense sets (see Lemma 2.5). Thus

$$Int(S_\mu) \setminus \bigcup_{\mathrm{Tr}^{(\mu)}} E(w_1 w_2 \ldots w_l)$$

is also nowhere dense, and since $Int(S_\mu)$ is open we obtain finally

$$Cl(\bigcup_{\mathrm{Tr}^{(\mu)}} E(w_1 w_2 \ldots w_l)) \supset Int(S_\mu)$$

or equivalently

$$\bigcup_{\mathrm{Tr}^{(\mu)}} Cl(E(w_1 w_2 \ldots w_l)) \supset Int(S_\mu)$$

since $\mathrm{Tr}^{(\mu)}$ is finite.

By combining the above fact and (5) we observe that

$$\{E(w_1 w_2 \ldots w_l) \mid w_1 w_2 \ldots w_l \in \mathrm{Tr}^{(\mu)}\}$$

is a convex generating family of $Int(S_\mu)$, which contradicts (4). The proof is complete. $\square$

3.7. The following example shows that the lower bound in Theorem 3.6 is exact. Let us choose a non-constant linear affine function $f : E^n \to R$, an integer $m \geqq 2$ and real numbers $c_1, c_2, \ldots, c_{m-1}$ such that

$$c_0 < c_1 < c_2 < \ldots < c_{m-1} < c_m\,,$$

where

$$c_0 := -\infty \quad \text{and} \quad c_m := +\infty\,.$$

Now, let

$$S_0 := \bigcup_{i=1}^{m-1} \{\mathbf{x} \in E^n \mid f(\mathbf{x}) = c_i\} = \{\mathbf{x} \in E^n \mid \bigvee_{i=1}^{m-1} (f(\mathbf{x}) = c_i)\} ;$$

$$S_1 := E^n \setminus S_0 = \bigcup_{i=1}^{m} \{\mathbf{x} \in E^n \mid c_{i-1} < f(\mathbf{x}) < c_i\} ;$$

$$I := \{0, 1\} ; \quad \mathfrak{S} := \{S_0, S_1\} .$$

Now we observe that S_0, S_1 are PS-s, $ic(Int(S_0)) = 0$, $ic(Int(S_1)) = m$ (Lemma 3.3) and $\mathfrak{S}$ is a quasipartition of E^n. Hence we may apply Theorem 3.6 which yields:

The following inequality is satisfied for each LCA $\mathscr{A}$ for the above $\mathfrak{S}$-problem:

$$comp(\mathscr{A}) \geqq \,]\log_2 (ic(Int(S_0)) + ic(Int(S_1)))[\,=\,]\log_2 m[\,.$$

On the other hand, it is easy to construct an LCA for the above $\mathfrak{S}$-problem, having the measure of complexity just $]\log_2 m[$. Informally speaking, this algorithm is based on the optimum policy of successive halving the integer interval $\{0, 1, \ldots, m\}$.

IV. AN APPLICATION AND CONCLUDING REMARKS

4.1. Theorem 3.6. will be now applied to the knapsack problem (Example 3 of 1.3) to obtain a lower bound for the number of comparisons, required by this problem and proved originally by the present author in 1967 [1], cf. also [2].

4.2. To state this result we use the concept of *threshold function* (see e.g. [5]): A function $p : \{0, 1\}^m \to \{0, 1\}$ is called a *threshold function* of m variables if there exists an $(m + 1)$ -tuple $(a_1, a_2, \ldots, a_m, a) \in R^{m+1}$ such that

$$p(x_1, \ldots, x_m) = 1 \quad \text{if} \quad \sum_{j=1}^{m} a_j x_j > a ,$$
$$= 0 \quad \text{if} \quad \sum_{j=1}^{m} a_j x_j < a .$$

Let $\prod_m$ denote the set of all threshold functions of m variables and let $\pi_m :=$ $:= \text{card}\,(\prod_m)$. The following bounds for π_m are known, cf. [15], [16] and [17]:

$$\limsup_{m \to \infty} m^{-2} \log_2 \pi_m \leqq 1 ,$$

$$\liminf_{m \to \infty} m^{-2} \log_2 \pi_m \geqq \tfrac{1}{2} .$$

4.3. **Theorem.** *Let* $I := \{0, 1\}$, $\mathfrak{S} := \{S_0, S_1\}$, *where*

$$S_0 := \{(a_1, \ldots, a_m, a) \in R^{m+1} \mid \textit{There exists } (x_1, \ldots, x_m) \in \{0, 1\}^m \textit{ such that } \sum_{j=1}^{m} a_j x_j = a\} ,$$

$$S_1 := R^{m+1} \setminus S_0 .$$

Then for each $\mathscr{A} \in \mathfrak{A}\langle\mathfrak{S}\rangle$,

$$comp(\mathscr{A}) \geqq \,]\log_2 \pi_m[\,.$$

Proof. In view of $ic(Int(S_0)) = ic(\emptyset) = 0$ it is sufficient to verify $ic(Int(S_1)) = \pi_m$, and apply Theorem 3.6. Now S_1 is open, each connected component of $Int(S_1) = S_1$ is convex (it is, actually, an SPS) and hence in view of Lemma 3.3 $ic(Int(S_1)) = ic(S_1)$ equals the number of all connected components of S_1.

Thus it is sufficient to find some bijection between the set of all connected componnents of S_1 and the set of all threshold functions of m variables. But for each connected component M of S_1 there exists just one subset $B \subset \{0, 1\}^m$ such that

$$M = \{(a_1, a_2, \ldots, a_m, a) \in R^{m+1} \mid \sum_{j=1}^{m} a_j x_j > a \text{ if } (x_1, x_2, \ldots, x_m) \in B, \text{ and}$$

$$\sum_{j=1}^{m} a_j x_j < a \quad \text{if} \quad (x_1, \ldots, x_m) \in \{0, 1\}^m \setminus B\}\,.$$

Let $\chi_B : \{0, 1\}^m \to \{0, 1\}$ be the characteristic function of B, i.e.

$$\chi_B(x_1, \ldots, x_m) = 1 \quad \text{if and only if} \quad (x_1, \ldots, x_m) \in B\,.$$

It is easy to see that the mapping '$M \mapsto \chi_B$' is a bijection of the set of all connected components of S_1 onto $\prod_m$, which completes the proof. $\square$

4.4. Concluding remarks. 1) The same lower bound can be obtained for the general linear programming problem with $\{0, 1\}$-variables, see [2]. In [2] lower bounds are also obtained for the number of comparisons required by the integer linear programming problem with uniformly bounded variables and by a certain problem of integer polynomial programming.

2) The proof technique used in Theorem 3.6 is in effect the usual and very general entropy (cardinality) method, based on the count of all essential cases, occuring in an algorithm. In order to derive more exact lower bounds for concrete $\mathfrak{S}$-problems, such as the knapsack problem or the travelling salesman problem, some new proof techniques are needed. better reflecting the intrinsic combinatorial structure of the problems.

3) Added in proofs: The author's main result from [4] has been rediscovered in a paper by *A. C. Yao* and *R. L. Rivest*: On the Polyhedral Decision Problem. SIAM J. Comput. 9 (1980) 2, pp. 343—347.

References

[1] *J. Morávek:* On the Complexity of Discrete Programming Problems. The talk on the 6th International Symposium on Mathematical Programming, Princeton University 1967.

[2] *J. Morávek:* On the Complexity of Discrete Programming Problems. Aplikace Matematiky, 14 (1969), pp. 442—474.

[3] *J. Morávek:* A Note upon Minimal Path Problem. Journal of Math. Analysis and Appl., 30 (1970) 3, pp. 702—717.

[4] *J. Morávek:* A Localization Problem in Geometry and Complexity of Discrete Programming, Kybernetika, 8 (1972) 6, pp. 498—516.
[5] *S. Muroga:* Threshold Logic and Its Applications. John Wiley & Sons, New York, London, Sydney, 1971.
[6] *D. Dobkin* and *R. J. Lipton:* A Lower Bound of $\frac{1}{2}n^2$ on Linear Search Programs for the Knapsack Problem. Mathematical Foundations of Computer Science 1976, Gdansk, published in Lecture Notes in Computer Science 45, 1976.
[7] *J. L. Kelley:* General Topology. Van Nostrand Reinhold Company, 1955.
[8] *R. M. Karp:* Reducibility among Combinatorial Problems. Complexity of Computer Computations, Proceedings, Plenum Press 1972. Editors: R. E. Miller, J. W. Thatcher.
[9] *S. S. Kislicyn:* On the Selection of the k^{th} Element of an Ordered Set by Pairwise Comparisons, (Russian), Sibirsk. Mat. Zh., Vol. 5, pp. 557—564.
[10] *M. Blum, R. W. Floyd, V. Pratt, R. Rivest* and *R. Tarjan:* Time Bounds for Selection, JCSS 7 (1973), pp. 448—461.
[11] *J. Pohl:* A Sorting Problem and Its Complexity. Communications of ACM, 15 (1972) 6, pp. 462—466.
[12] *R. E. Bellman:* Dynamic Programming Treatment of the Travelling Salesman Problem. J. Assoc. for Comp. Mach. 9 (1962), pp. 61—63.
[13] *M. Held* and *R. M. Karp:* A Dynamic Programming Approach to Sequencing Problems, J. Soc. Ind. and Appl. Math. 10 (1962) 1.
[14] *А. А. Корбут, Ю. Ю. Финкельштейн:* Дискретное программирование. Москва, Наука 1969.
[15] *R. O. Winder*: Bounds of Threshold Gate Realizability, TRNS IEEE EC 12 (1963).
[16] *S. Yajima* and *T. Ibaraki:* A Lower Bound of the Number of Threshold Functions, TRNS IEEE EC 14 (1965) 6.
[17] *M. Bloch* and *J. Morávek:* Bounds of the Number of Threshold Functions. Informations Processing Machines, 13 (1967), pp. 67—73.
[18] *D. Dobkin* and *R. J. Lipton*: A Lower Bound of $\frac{1}{2}n^2$ on Linear Search Programs for the Knapsack Problem. JCSS, 16 (1978) 3, pp. 413—417.

Souhrn

GEOMETRICKÁ METODA V KOMBINATORICKÉ SLOŽITOSTI

JAROSLAV MORÁVEK

Je získán dolní odhad pro počet srovnání, nutných k řešení výpočetního problému klasifikace libovolně zvoleného bodu Euklidovského prostoru, vzhledem k danému, konečnému systému polyedrických (obecně nekonvexních) množin, pokrývajících prostor. Získaný dolní odhad závisí, zhruba řečeno, na minimálním počtu konvexních částí, na něž lze rozložit zmíněné polyedrické množiny. Dolní odhad je aplikován na úlohu o ranci.

Author's address: RNDr. *Jaroslav Morávek*, CSc., Matematický ústav ČSAV, Žitná 25, 115 67 Praha 1.