Tomus 1. Fasciculus 3.

ACTA CYBERNETICA

FORUM CENTRALE PUBLICATIONUM CYBERNETICARUM HUNGARICUM

Szeged, 1972

Curat: Universitas Szegediensis de Attila József nominata

1. kötet 3. füzet

ACTA CYBERNETICA

A HAZAI KIBERNETIKAI KUTATÁSOK KÖZPONTI PUBLIKÁCIÓS FÓRUMA

Szeged, 1972. június
A szegedi József Attila Tudományegyetem gondozásában

On some enumeration questions concerning trees and tree-type networks

By A. ÁDÁM and J. BAGYINSZKI

To the memory of Dr Catherine Rényi and Professor Alfréd Rényi

Zusammenfassung. Im § 3 werden gewisse Fragen der Abzählung von Wurzel-Bäumen betrachtet. Sei T ein Wurzel-Baum mit der Wurzel R, bezeichnen wir durch k die Anzahl der Kanten von T. Teilen wir die Kanten in Klassen durch die folgende Relation ein: zwei Kanten sind äquivalent, wenn sie auseinander ohne Berühren von R erreichbar sind. Existieren genau $\varkappa_i$ Äquivalenzklassen, die aus je i Kanten bestehen (wobei i die Zahlen $1, 2, 3, \ldots, k$ durchläuft), so sagen wir, daß die Partition von T der Vektor $K=\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle$ ist. Wir erhalten drei Formeln für die Anzahl $S_K(k)$ der numerierten Bäume von der Partition K unter die Annahme, daß die Nummer der Wurzel als 1 fixiert wird und die übrigen Punkte die Nummern $2, 3, \ldots, k+1$ (auf beliebige Weise) bekommen. Eine dieser Formeln stimmt im Wesentlichen mit einem (in verschiedener Weise bewiesenen) Resultat von J. Dénes überein. Aus unseren Ergebnissen ist auch die wohlbekannte Formel von Cayley ableitbar (Corollary 1).

In den Paragraphen 4—5 wird ein zeitliches Verhalten dem Wurzel-Baum T laut des Modells der früheren Arbeit [1] zugeordnet, so daß die Kanten in die Richtung der Wurzel gerichtet sind und jeder Punkt P_i einen im Intervall (0, 1) liegenden beliebigen Anfangswert $\beta(P_i)$ hat. Wir definieren fünf Typen von mit den Werten $\beta(P_i)$ versehenen Bäumen, die fünf charakteristischen Arten des Verhaltens entsprechen (Proposition 6). Im § 4 studieren wir die Wahrscheinlichkeit des Ereignisses, daß der Baum zu einem oder anderem Typ gehört, wenn sowohl der Baum (als ein Graph) wie die Werte $\beta(P_i)$ zufällig gewählt sind.

§ 1. Introduction

§ 3 is devoted to some enumeration questions of rooted trees. In Theorems 1, 2 and Corollary 2 several formulae for the number of labelled rooted trees having a fixed partition of the number k of edges are obtained, supposing that the root is labelled by 1 and the other vertices by $2, 3, \ldots, k+1$. From our results the well-known Cayley enumeration formula can be deduced, too (Corollary 1).

In §§ 4—5, a temporal behaviour is assigned to the rooted tree T in sense of the model exposed in the former paper [1], such that each edge is directed towards

the root and any vertex P_i has an arbitrary initial value $\beta(P_i)$ lying in the interval $(0, 1)$. We define five types of trees, being supplemented with the values $\beta(P_i)$; these types correspond to five characteristic features of behaviour (Proposition 6). We study in § 4 the probability of the event that the tree belongs to one or another type provided that the tree (as a graph) and the values $\beta(P_i)$ are chosen randomly.

A large collection of results and methods concerning the enumeration questions of labelled trees is contained in the lecture note [4] of Moon. The articles of Dénes [3] and A. Rényi [6] deal with subjects closely connected with the present paper; especially, our Corollary 2 follows easily from Theorem 5 of [3] (by adding a new vertex as a root to the graph and by connecting the root to one vertex in each component). The publication [7] of C. and A. Rényi is devoted to the generalization of the questions of counting for the case of k-trees.

§ 2. Preliminaries

We suppose that the reader is familiar with the basic notions of graph theory. If the edge e and the vertex P are incident, then we say, equivalently, that P is a *terminal* of e.

Let H be a finite set and $H_1, H_2, \ldots, H_j$ be some pairwise disjoint non-empty subsets of H. If the union of $H_1, H_2, \ldots, H_j$ equals to H, then we say that $H_1, H_2, \ldots, H_j$ form a *set-partition* of H. (The ordering of the H_i's is indifferent.)

Let k be a natural number. If the members of the vector

$$K=\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle$$

consisting of k non-negative integers satisfy the equality

$$1 \cdot \varkappa_1+2 \cdot \varkappa_2+3 \cdot \varkappa_3+\cdots+k \cdot \varkappa_k = k, \tag{2. 1}$$

then we say that K is a *numerical partition* of the number k.

We speak about a partition simply if the context makes doubtless whether a numerical one or a set-partition is dealt with.

Let a set-partition $H_1, H_2, \ldots, H_j$ of the set H consisting of k elements be given. If, among the subsets $H_1, H_2, \ldots, H_j$,

there are $\varkappa_1$ subsets each consisting of 1 element,

there are $\varkappa_2$ subsets each consisting of 2 elements,

...

and there are $\varkappa_k$ subsets each consisting of k elements,

then ((2. 1) is obviously fulfilled and) we say that the numerical partition, assigned to the partition of H in question, is $\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle$.

Denote by Ω_k the set of all numerical partitions of the number k. If we write $\sum_{\Omega_k}$, then the summation must be taken for all elements K of Ω_k.

Let a, b be real numbers such that $a \leqq b$. By the *closed interval* $[a, b]$ we mean the set of the real numbers x satisfying $a \leqq x \leqq b$. By the *open interval* (a, b) we understand the set of the real numbers fulfilling $a < x < b$. In analogy, we define $[a, b)$ and $(a, b]$ by the conditions $a \leqq x < b$ and $a < x \leqq b$, respectively.

We shall often write $\exp x$ instead of e^x where e is the base of natural logarithms (this is useful if a long expression occurs in the role of x). The Bürmann—Langrange-

formula concerning the series expansion of inverse functions is supposed to be known (see [2]). We shall use the subsequent Proposition I, II (of analytic character):

Proposition I. *There holds the identity*

$$(f(x)=)\prod_{j=1}^{\infty}\left(\sum_{\varkappa_j=0}^{\infty}(a_{j-1}x^j)^{\varkappa_j}\frac{1}{\varkappa_j!}\right)=1+\sum_{k=1}^{\infty}A_k x^k \tag{2.2}$$

in the real interval (u, v) *where*

$$A_k=\sum_{\Omega_k}\prod_{j=1}^{k}a_{j-1}^{\varkappa_j}\frac{1}{\varkappa_j!}$$

if the power series on the right-hand side of (2. 2) *is uniformly convergent in* (u, v).

Proof. Let the expression on the left-hand side of (2. 2) be ordered according to the increasing powers of x. Then the coefficient of x^k gets an additive contribution from all the possible partitions of the number k; the contribution of any single partition is $\prod_{j=1}^{k}a_{j-1}^{\varkappa_j}\frac{1}{\varkappa_j!}$.

Before stating Proposition II, we introduce three notations z_m, $T_r(x)$, $Z(x)$ as follows:

$$T_r(x)=\cos(r\arccos x)$$

(i.e. $T_r(x)$ is the Chebyshev polynomial of degree r),

$$z_m=6\cdot 2^{1/2}\pi^{-3/4}m^{-1/4}e^{-\frac{1}{4}\sqrt{\pi m}}\left(1+O(m^{-1/2})\right)\quad\text{if}\quad m\to\infty,$$

$$Z(x)=\frac{1}{12x}\left(z_0+2\sum_{m=1}^{\infty}(-1)^m z_m T_{2m}\left(\frac{1}{x}\right)\right)\quad\text{where}\quad x\geqq 1. \tag{2.3}$$

Proposition II. *There holds the identity*

$$n\cdot\Gamma(n)=n!=(2\pi)^{1/2}\cdot n^{n+1/2}\cdot\exp\left(-n+Z(n)\right),$$

consequently, the right-hand side of the definition (2. 3) *is convergent.*

The proof of Proposition II may be found in [5]. We note that the analogon of the convergence conclusion of this proposition does not hold for Stirling series.

§ 3. The enumeration of rooted trees

A *rooted tree* is a finite connected undirected graph without circuits in which a vertex is distinguished. The distinguished vertex is called the *root* of the tree. If R is the root and P is an arbitrary vertex in a rooted tree, then the distance of R and P is called also the *height* of P.[1]

[1] In §§ 4—5 we shall consider the rooted trees as *directed* graphs in such a manner that each edge is oriented toward the vertex of smaller height.

Let T be a rooted tree and R the root in it; suppose that the degree of R is 1. We say that the *partition* of the tree T is

$$K_0 = \langle \overset{1}{\overbrace{0}}, \overset{2}{\overbrace{0}}, \overset{3}{\overbrace{0}}, \ldots, \overset{k-1}{\overbrace{0}}, \overset{k}{\overbrace{1}} \rangle.$$

Denote by P the single vertex adjacent to R. If we delete R and the edge between P, R, then we get a tree T'; we agree that P should be the root of T'. The rooted tree T', defined in this manner, is called the *truncated tree* of T. (It is defined only if the degree of the root is one.) If the number of edges of T is k, then T' has $k-1$ edges (consequently, k vertices).

Now let T be a rooted tree (with the root R) such that the degree d of R is at least two. Denote the edges incident to R by $e_1, e_2, \ldots, e_d$, and their terminals, different from R, by $P_1, P_2, \ldots, P_d$, respectively. We define d new rooted trees $T_1, T_2, \ldots, T_d$ in the following four steps:

(i) we delete R, $e_1, e_2, \ldots, e_d$,

(ii) we introduce d new vertices $R_1, R_2, \ldots, R_d$ and d new edges $e'_1, e'_2, \ldots, e'_d$,

(iii) for each number i $(1 \leqq i \leqq d)$, let e'_i be incident to R_i and P_i,

(iv) for each i $(1 \leqq i \leqq d)$ let T_i be that connected component of the graph, built up in the previous steps, which contains R_i; let R_i be the root of T_i.

The process, described in (i), (ii), (iii), (iv), is called the *dismembering* of the tree T (having a root of degree >1) and every T_i is called a *branch* of T.

If, for each number j $(1 \leqq j \leqq k)$, there are exactly $\varkappa_j$ branches T_i such that the number of edges of any T_i equals to j, then we say that the *partition* of T is

$$\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle.$$

Evidently, this expression is a partition of the number of edges of T.

Let T be a rooted tree with k edges. T has $k+1$ vertices. Let us assign $k+1$ different natural numbers to the vertices of T. The tree T together with such an assignment is called a *labelled rooted tree*. If we require, in addition, that the assigned numbers should be $1, 2, 3, \ldots, k, k+1$ and, especially, the root should have the number 1, then we speak on a *standardly labelled rooted tree*.

We denote by $N(k)$ the number of the (non-isomorphic) labelled rooted trees with k edges when the set of numbers, corresponding to the vertices, is fixed. Furthermore, we denote by $S(k)$ the number of the standardly labelled rooted trees with k edges. If K is a partition of k and only the trees having partition K are counted, then the analogous numbers are denoted by $N_K(k)$ and $S_K(k)$, respectively. Obviously,

$$N(k) = \sum_{\Omega_k} N_K(k) \quad \text{and} \quad S(k) = \sum_{\Omega_k} S_K(k).$$

In case $k=1$ we have evidently

Proposition 1. *For the single partition* $K_0 = \langle 1 \rangle$ *of* 1

$$N(1) = N_{K_0}(1) = 2 \quad \textit{and} \quad S(1) = S_{K_0}(1) = 1$$

hold.

Proposition 2. *If K is an arbitrary partition of k, then*

$$N_K(k) = (k+1) \cdot S_K(k). \tag{3.1}$$

Remark. We get from (3.1) $N(k) = (k+1)S(k)$ by summarizing for all partitions K.

Proof. We can suppose (without an essential restriction of the generality) that the vertices are labelled with the numbers $1, 2, \ldots, k+1$ in the non-standard case too. Let the set $\mathfrak{S}$ of all the trees (with k edges) labelled with these numbers be considered. For any element T of $\mathfrak{S}$, let us consider the vertex P to which 1 corresponds. If we interchange the labelling of R and P, then we get a standardly labelled tree. In the mapping, defined by this interchanging, every standardly labelled tree is obtained exactly $k+1$ times.

Proposition 3. *For the partition* $K_0=\langle 0, 0, \ldots, 0, 1\rangle$ *of* k, *the equality*

$$S_{K_0}(k) = N(k-1)$$

is satisfied.

Proof. Let us consider the set of the standardly labelled trees T (with k edges) the partition of which is K_0. If we form the truncated trees of the T's, we get a one-to-one correspondence with the set of the trees with $k-1$ edges, being labelled with the numbers $2, 3, \ldots, k+1$.

Theorem 1. *Let* $K=\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k\rangle$ *be an arbitrary partition of the number* k. *Then*

$$S_K(k) = k! \prod_{i=1}^{k} \left(\left(\frac{N(i-1)}{i!} \right)^{\varkappa_i} \frac{1}{\varkappa_i!} \right). \tag{3.2}$$

Remarks. $N(0)$ is regarded to be 1. If $\varkappa_i=0$, then the i-th factor of the product in (3.2) equals to 1.

Proof. Let us consider the set $\mathfrak{S}$ of all the standardly labelled rooted trees, with k edges, having the numerical partition K; moreover, all the set-partitions Λ of the set $\{2, 3, \ldots, k+1\}$ to which the numerical partition K corresponds. To each element T of $\mathfrak{S}$, we assign a set-partition Λ as follows: two numbers i, j belong to a common class precisely if the vertices, labelled with i and j, are in the same branch of T. Let the set-partition Γ of $\mathfrak{S}$ be defined so that $T(\in \mathfrak{S})$ and $T'(\in \mathfrak{S})$ are in a common class when the same set-partition Λ is assigned to them.

It is easy to see that the number of the set-partitions Λ is

$$k! \Big/ \prod_{i=1}^{k} ((i!)^{\varkappa_i} \varkappa_i!).$$

Furthermore, for any fixed Λ, there exist

$$\prod_{i=1}^{k} (N(i-1))^{\varkappa_i}$$

trees T lying in a common class modulo Γ (this can be pointed out if one considers the forest consisting of the truncated trees of the branches of T). The product of the obtained quantities yields the formula exposed in the theorem.

In the remaining parts of this §, we shall show that the well-known formula of Cayley may be deduced as a consequence of Theorem 1, moreover, two explicit formulae for the quantity $S_K(k)$ will be given.

Corollary 1 (the enumeration formula of Cayley).

$$S(k) = (k+1)^{k-1}.$$

Proof. Let us summarize both sides of (3. 2) for all the partitions of k. We get, by use of (3. 1), the recursion

$$S(k) = \sum_{\Omega_k} S_k(k) = k! \sum_{\Omega_k} \left(\prod_{j=1}^{k} \left(\frac{S(j-1)}{(j-1)!} \right)^{\varkappa_j} \frac{1}{\varkappa_j!} \right). \tag{3. 3}$$

This recursion can be solved by the method of generating functions. Let the exponential generating function of $S(k)$ be defined as

$$G(x) = \sum_{k=0}^{\infty} \frac{S(k)}{k!} x^{k+1}$$

(the empty product $\prod_{j=1}^{0}$ is regarded to be 1). By utilizing (3. 3) and Proposition I, we get the functional equation

$$G(x) = \sum_{k=0}^{\infty} x \left[\sum_{\Omega_k} \left(\prod_{j=1}^{k} \left(\frac{S(j-1)x^j}{(j-1)!} \right)^{\varkappa_j} \frac{1}{\varkappa_j!} \right) \right] = x \prod_{j=1}^{\infty} \left[\sum_{\varkappa_j=0}^{\infty} \left(\frac{S(j-1)x^j}{(j-1)!} \right)^{\varkappa_j} \frac{1}{\varkappa_j!} \right] =$$

$$= x \prod_{j=1}^{\infty} \exp\left\{ \frac{S(j-1)x^j}{(j-1)!} \right\} = x \exp\left\{ \sum_{j=1}^{\infty} \frac{S(j-1)x^j}{(j-1)!} \right\} = x \cdot e^{G(x)}$$

for $G(x)$. Since the Bürmann-Lagrange series expansion formula (see [2], p. 22) implies that the single solution of the functional equation

$$x = G(x) e^{-G(x)} \tag{3. 4}$$

is

$$G(x) = \sum_{k=0}^{\infty} \frac{(k+1)^{k-1}}{k!} x^{k+1}, \tag{3. 5}$$

the assertion is proved.

The next statement is essentially the same as a result of Dénes ([3], Theorem 5), proved by him with other methods.

Corollary 2.

$$S_K(k) = k! \prod_{i=1}^{k} \left(\left(\frac{i^{i-2}}{(i-1)!} \right)^{\varkappa_i} \frac{1}{\varkappa_i!} \right). \tag{3. 6}$$

Proof. We get from (3. 2) the formula (3. 6) by substituting i^{i-2} for $S(i-1)$ (in sense of Corollary 1).

Corollaries 1, 2 imply at once

Corollary 3. *Denote the quotient* $S_K(k)/S(k)$ *by* $F_k(K)$. *Then*

$$F_k(K) = \frac{k!}{(k+1)^{k-1}} \prod_{i=1}^{k} \left[\left(\frac{i^{i-1}}{i!} \right)^{\varkappa_i} \frac{1}{\varkappa_i!} \right]. \tag{3. 7}$$

Theorem 2. *For any partition* $K=\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k\rangle$ *of* k, *we have*

$$S_K(k) = (2\pi)^{-\frac{\varkappa-1}{2}} k^{k+\frac{1}{2}} \left(\prod_{i=1}^{k} (i^{\frac{3\varkappa_i}{2}} \varkappa_i!)\right)^{-1} \exp\left(Z(k) - \sum_{i=1}^{k} \varkappa_i Z(i)\right)$$

where $\varkappa = \varkappa_1+\varkappa_2+\cdots+\varkappa_k$ ($Z(n)$ *was defined in* § 2).

Proof. Let Corollary 2 be taken into account. Since

$$\frac{i^{i-2}}{(i-1)!} = \frac{i \cdot i^{i-2}}{i!},$$

we obtain the formula stated in the theorem in such a manner that Proposition II. is applied for $i!$ and $k!$, furthermore, the obvious possibilities for simplifying are performed. The proof is completed.

It remains an open problem to get a simpler formula being asymptotically equal to the quantity

$$\left(\prod_{i=1}^{k} (i^{\frac{3\varkappa_i}{2}} \varkappa_i!)\right)^{-1} \exp\left(Z(k) - \sum_{i=1}^{k} \varkappa_i Z(i)\right)$$

occurring in Theorem 2. We did not succeed in doing this.

§ 4. Some enumeration questions of networks with a rooted tree structure

By a *network*, we understand a finite directed graph G together with a function β defined on the vertex set of G, the range of β is the (real) open interval (0, 1).[2] The number $\beta(P)$ is called the *state* of the vertex P.[3]

In what follows, we shall consider networks formed from rooted trees, any edge being oriented toward its terminal having smaller height. We suppose that the states are assigned randomly to the vertices. Hence, we can assume that $\beta(P)\neq \neq\beta(Q)$ if $P\neq Q$ because the complementary event is (possible but) of probability 0.

The state of the root of a network is called the *state of the network*, too.

Assume that the root of the network G is of (in-)degree 1. Let the truncated tree G' be formed and, to the vertices of G', let the same states be attributed as their states in G. In this case the network G' is called the *truncated network* of G. — The term "*branch of a network*" is used in an analogous sense.

Let e be an edge going from P to Q. For the sake of the brevity, we say that e is a *red edge* or *green edge* according as $\beta(P)<\beta(Q)$ or $\beta(P)>\beta(Q)$, respectively.

We are going to introduce a partition of the set of networks into the types A, B, C, D, E. These types will be defined inductively by the twelve rules (i)—(xii)

[2] This definition (and the subsequent ones still more) has a certain formal character. The reasonable meaning of the notions introduced now will be explained in § 5 where we shall attribute a temporal behavior to the networks, starting with the states $\beta(P)$ assigned to the vertices.

[3] Now we have required that any state $\beta(P)$ must differ from 0 and 1. This was done for the simplicity's sake, because, on the one hand, the possibility when some values $\beta(P)$ are equal to 0 or 1 will be an event of probability 0, on the other hand, our treatment would be more lengthy and intricate if the states 0, 1 were allowed.

We emphasize that the numbers 0, 1 as states will *not* be excluded in § 5.

to be exposed. The root is denoted by R. In the rules (ii), (iii), (iv), (v), (vi), the degree of R is supposed to be 1; in the rules (vii), (viii), (ix), (x), (xi), (xii) the degree of R is supposed to be at least 2. If R is of degree 1, then let e_R be the single edge incident to R.

(i) If G has only one vertex (and no edge), then G belongs to the type A.

(ii) If the truncated network G' of G is either of type C or of type E, then G belongs to the type A.

(iii) If the edge e_R is red and G' is of type B or D, then G belongs to the type B.

(iv) If the edge e_R is green and G' is of type B or D, then G belongs to the type C.

(v) If the edge e_R is red and G' is of type A, then G belongs to the type D.

(vi) If the edge e_R is green and G' is of type A, then G belongs to the type E.

(vii) If G has a branch being of type E, then G belongs to the type E.

(viii) If G has two branches being of type C and D (respectively), then G belongs to the type E.

(ix) If G has no branch of type D or E but it has a branch being of type C, then G belongs to the type C.

(x) If G has no branch of type C or E but it has a branch being of type D, then G belongs to the type D.

(xi) If G has no branch of type C, D or E but it has a branch being of type B, then G belongs to the type B.

(xii) If every branch of G is of type A, then G belongs to the type A.

e_R \ G'	A	B	C	D	E
green	E	C	A	C	A
red	D	B	A	B	A

Table 1.

The rules (ii), (iii), (iv), (v), (vi) are illustrated in Table 1. The rules (vii), (viii), (ix), (x), (xi), (xii) can be summarized by saying that the strength of the five types is the partial ordering seen in Table 2.

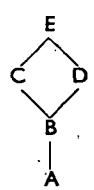

Table 2.

Let N be a network. We agree in some notations. The number of edges of N is k. The state of (the root of) N is β $(0<\beta<1)$. The partition of (the graph of) N is

$$K=\langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle.$$

The partition

$$\langle \overset{1}{\smile}\overset{2}{0}, \overset{3}{0}, \ldots, \overset{k-1}{\smile}\overset{k}{0}, 1\rangle$$

is denoted by K_0. K_1 denotes an arbitrary partition of k different from K_0.

In what follows, we use a small letter p or a Capital one P according as the probability, to be denoted, does or does not depend on β (resp.). (In the latter case, β can vary in the interval (0, 1).) After p, the variable β will or will not be written out.

For a partition K of k, we denote by p_x^K the probability of the event that a randomly chosen network of partition K (with k edges), being of state β, belongs to the type X where X can be any of A, B, C, D, E (and, accordingly, the subscript of p is a small letter a, b, c, d or e). We write $p_x^{(k)}$ for the analogous probability when k is fixed but not K. We denote by $P_x^{(k)}$ the probability of the fact that a network, chosen randomly out of all networks having k edges, belongs to the type X.

We adopt three hypotheses (H1), (H2), (H3):

(H1) All the graph-theoretical structures of forming a rooted tree from k edges (distinghished from each other by the isomorphy of standardly labelled trees) are equiprobable.

(H2) The state of a vertex P is chosen from the real interval (0, 1) in sense of the uniform distribution.

(H3) The states of two different vertices P, Q are chosen independently of each other.

If these hypotheses are accepted, then the rules (i)—(xii) imply the following recursive system for the probabilities introduced above:

$$p_x^{(k)} = \sum_{\Omega_k} p_x^K F_k(K) \tag{4.1}$$

(the quantities $F_k(K)$ were determined in Corollary 3)

$$P_x^{(k)} = \int_0^1 p_x^{(k)} d\beta \tag{4.2}$$

(where x can be any of a, b, c, d, e)

$$p_a^{K_0} = P_c^{(k-1)} + P_e^{(k-1)} \tag{4.3}$$

$$p_b^{K_0} = \int_0^\beta \left(p_b^{(k-1)}(\beta') + p_d^{(k-1)}(\beta')\right) d\beta' \tag{4.4}$$

$$p_c^{K_0} = \int_\beta^1 \left(p_b^{(k-1)}(\beta') + p_d^{(k-1)}(\beta')\right) d\beta' \tag{4.5}$$

$$p_d^{K_0} = \int_0^\beta p_a^{(k-1)}(\beta')\, d\beta' \tag{4.6}$$

$$p_e^{K_0} = \int_\beta^1 p_a^{(k-1)}(\beta')\, d\beta' \tag{4.7}$$

$$p_a^{K_1} = \prod_{j=1}^{k} (p_a^{(j)})^{\varkappa_j} \tag{4.8}$$

$$p_b^{K_1} = \prod_{j=1}^{k} (p_a^{(j)} + p_b^{(j)})^{\varkappa_j} - p_a^{K_1} \tag{4.9}$$

$$p_c^{K_1} = \prod_{j=1}^{k} (p_a^{(j)} + p_b^{(j)} + p_c^{(j)})^{\varkappa_j} - (p_a^{K_1} + p_b^{K_1}) \tag{4.10}$$

$$p_d^{K_1} = \prod_{j=1}^{k} (p_a^{(j)} + p_b^{(j)} + p_d^{(j)})^{\varkappa_j} - (p_a^{K_1} + p_b^{K_1}) \tag{4.11}$$

$$p_e^{K_1} = 1 - (p_a^{K_1} + p_b^{K_1} + p_c^{K_1} + p_d^{K_1}) \tag{4.12}$$

(in (4. 8)—(4. 12), $K_1 = \langle \varkappa_1, \varkappa_2, \ldots, \varkappa_k \rangle$). Indeed, equality (4. 1) follows from (H1); (4. 2) is implied by (H2), (H3). The equalities (4. 3)—(4. 7) are consequences of the rules (ii)—(vi), respectively. (4. 8)—(4. 12) follow by analyzing (vii)—(xii) if one takes into account that these rules do not contradict to each other and the premissa of them form a full system of events (if events having probability 0 are disregarded).

We are going to point out that the solution of the equation system (4. 3)—(4. 12) can be reduced to solving a recursive equation system such that the recursive system depends on the expressions

$$x_k = p_x^{(k)} \frac{k^{k-1}}{k!}$$

(where x may be any of a, b, c, d, e) and, of course, on the number k (but is independent of the partition K of k).

Proposition 4. *Let us introduce the simpler notation*

$$\Sigma\Pi_{y,z,\ldots,u}$$

for the expression

$$\sum_{\Omega_k} \prod_{j=1}^{k} \left\{ (y_j + z_j + \cdots + u_j)^{\varkappa_j} \frac{1}{\varkappa_j!} \right\}.$$

The system (4. 3)—(4. 12) *implies the subsequent system of equations* (4. 13)—(4. 17):

$$\left[1 + \left(\frac{k+1}{k}\right)^{k-1}\right] a_k = \Sigma\Pi_a + \left(\frac{k}{k-1}\right)^{k-2} \int_0^1 (c_{k-1} + e_{k-1}^{*}) d\beta' \tag{4.13}$$

$$\left[1 + \left(\frac{k+1}{k}\right)^{k-1}\right] (a_k + b_k) = \tag{4.14}$$

$$= \Sigma\Pi_{a,b} + \left(\frac{k}{k-1}\right)^{k-2} \left\{ \int_0^{\beta} (b_{k-1} + d_{k-1}) d\beta' + \int_0^1 (c_{k-1} + e_{k-1}) d\beta' \right\}$$

$$\left[1+\left(\frac{k+1}{k}\right)^{k-1}\right](a_k+b_k+c_k) =$$

(4. 15)

$$= \Sigma\Pi_{a,b,c}+\left(\frac{k}{k-1}\right)^{k-2}\int_0^1 (b_{k-1}+c_{k-1}+d_{k-1}+e_{k-1})d\beta'$$

$$\left[1+\left(\frac{k+1}{k}\right)^{k-1}\right](a_k+b_k+d_k) =$$

(4. 16)

$$= \Sigma\Pi_{a,b,d}+\left(\frac{k}{k-1}\right)^{k-2}\left\{\int_0^\beta (a_{k-1}+b_{k-1}+d_{k-1})d\beta'+\int_0^1 (c_{k-1}+e_{k-1})d\beta'\right\}$$

(4. 17)
$$\frac{k^{k-1}}{k!} = a_k+b_k+c_k+d_k+e_k.$$

Proof. We shall use the following terminology: if two equations of form $X=Y$, $Z=W$ are given, then the equation $XZ=YW$ is called the product of them.

Let us form the product of any of (4. 3), (4. 4), (4. 5), (4. 6) with (3. 7), applied for K_0; similarly, let the product of any of (4. 8)—(4. 12) with (3. 7), applied for K_1, be formed. Furthermore, let the sums corresponding to (4. 1) be formed for each of the subscripts a, b, c, d, e (for x), concerning all the partitions of k. This equation system can be deduced by use of (3. 7) to the system (4. 13)—(4. 17).

We did not succeed in solving the system (4. 13)—(4. 17) completely. However, we can prove some partial results:

Theorem 3. *The following assertions hold:*

(A) *Any of* a_k, b_k+c_k, d_k+e_k *is a rational expression of* k, *independent of* β.

(B) b_k *and* d_k *are polynomials of* β *with degree exactly* k, *with non-negative (rational) coefficients, without a term of degree zero.*

(C)

$$a_k+b_k+c_k+d_k+e_k = \frac{k^{k-1}}{k!}.$$

(D) *Each of* a_k, b_k+c_k, d_k+e_k, b_k, d_k *is contained in the interval* $\left[0, \frac{k^{k-1}}{k!}\right]$.

(E) *Each of* a_k, b_k, c_k, d_k, e_k *is a polynomial of* β *with coefficients being rational in* k.

Proof. First we verify the independence statements of the assertion (A). The last term of (4. 13) does not depend on β, because the limits of the integration concerning β are constant. Subtract a_k from both sides; we get $\left[\frac{k+1}{k}\right]^{k-1} a_k$ on the left-hand side, and a sum on the right-hand one each term of which is a product of expressions a_j (with $j<k$) (the summation is taken over all partitions of the number k except the one-member partition $j=k$ that was subtracted). Hence the independence of a_k can be obtained by induction with respect to k.

(4. 13) implies by an analogous deduction that $a_k+b_k+c_k$ is independent on β. Since a_k proved to be independent, the same holds for b_k+c_k, too.

The independence of d_k+e_k follows from (4. 17) and the previous parts of the proof.

The rationality statements of (A) can be obtained as consequences of (E) (to be proved later). Now we are going to prove (B).

Let (4. 13) be subtracted from (4. 14). We get by the independence of a_k:

$$\left[1+\left(\frac{k+1}{k}\right)^{k-1}\right]b_k = \Sigma\Pi_{a,b}-\Sigma\Pi_a+$$

$$+\left(\frac{k}{k-1}\right)^{k-2}\int_0^\beta (b_{k-1}+d_{k-1})d\beta' = \tag{4.18}$$

$$=\sum_{\Omega_k}\left(\prod_{j=1}^{k}\frac{1}{\varkappa_j!}\right)\left(\prod_{j=1}^{k}(a_j+b_j)^{\varkappa_j}-\prod_{j=1}^{k}a_j^{\varkappa_j}\right)+\left(\frac{k}{k-1}\right)^{k-2}\int_0^\beta (b_{k-1}+d_{k-1})d\beta'.$$

The binomial theorem enables the subsequent transformation:

$$\prod_{j=1}^{k}(a_j+b_j)^{\varkappa_j}-\prod_{j=1}^{k}a_j^{\varkappa_j}=\prod_{j=1}^{k}\left\{a_j^{\varkappa_j}+\sum_{l=0}^{\varkappa_j-1}\binom{\varkappa_j}{l}a_j^l b_j^{\varkappa_j-l}\right\}-\prod_{j=1}^{k}a_j^{\varkappa_j}$$

(where the empty sum of type $\sum_{l=0}^{-1}$ or $\sum_{l=0}^{0}$ is regarded to be 0). If we multiply out in the first product, then an expression is yielded every term of which contains a power of b_j (with a positive exponent) as a factor, because precisely that term $\prod_{j=1}^{k}a_j^{\varkappa_j}$ is subtracted which does not contain such a power. It is clear that subtraction cannot occur in the remaining terms, furthermore, if b_k has been subtracted from both sides of (4. 13), every subscript j of a b_j on the right-hand side of the resulting equality satisfies $j<k$. This implies the statement, to be proved, by induction, with regard to the following remarks. The right-hand side is a sum each term of which is a polynomial of degree $\Sigma\varkappa_j\cdot j = k$ with non-negative coefficients without a term of degree zero (by the induction hypothesis). The latter term containing the integral is the integral of a polynomial with non-negative coefficients on the interval $[0, \beta]$, the degree of this polynomial is exactly $k-1$; hence the integration yields a polynomial exactly of degree k with non-negative coefficients, without a term of degree zero.

Thus the assertion of (B) concerning b_k is proved. By the analogy, we give the proof for d_k only in outlines. We subtract (4. 14) from (4. 16); afterwards, we calculate with a_j+b_j, d_j, $a_{k-1}\beta$ instead of a_j, b_j, $\int_0^\beta (b_{k-1}+d_{k-1})d\beta'$ (resp.) occurring in the above proof concerning b_k.

(C) coincides with (4. 17).

In order to prove (D), first we note that the definition of x_k implies that each of a_k, b_k, c_k, d_k, e_k is contained in the interval $\left[0, \frac{k^{k-1}}{k!}\right]$. Since the values $p_b^{(k)}+p_c^{(k)}$ and $p_d^{(k)}+p_e^{(k)}$ are probabilities, also b_k+c_k, d_k+e_k belong to $\left[0, \frac{k^{k-1}}{k!}\right]$.

Finally also the assertions of (E) will be proved by induction (with respect to k). Suppose that (E) is true with $k-1$ (instead of k). (4. 13) implies that the assertion (with k) holds for a_k; (4. 14) implies that it is valid for a_k+b_k; hence it is true for b_k, too. Similarly, the assertion follows from (4. 15) and (4. 14) for c_k, from (4. 16) and (4. 14) for d_k, thus (by (4. 17)) for e_k as well.

The inductive proof is completed by Tables 3, 4. Table 4 contains the values of $p_x^{(k)}$ and x_k if k is 0, 1, 2, 3, 4; similarly, Table 3 gives the values of p_x^K when $0 \leqq k \leqq 4$.

k	0	1	2		3		
K	$\langle 0\rangle$	$\langle 1\rangle$	$\langle 0,1\rangle$	$\langle 2,0\rangle$	$\langle 0,0,1\rangle$	$\langle 1,1,0\rangle$	$\langle 3,0,0\rangle$
$F_k(K)$	1	1	2/3	1/3	9/16	3/8	1/16
p_a^K	1	0	1/2	0	4/9	0	0
p_b^K	0	0	$\beta^2/2$	0	$2\beta^3/9$	0	0
p_c^K	0	0	$(1-\beta^2)/2$	0	$(2-2\beta^3)/9$	0	0
p_d^K	0	β	0	β^2	$\beta/3$	$(\beta+2\beta^2)/3$	β^3
p_e^K	0	$1-\beta$	0	$1-\beta^2$	$(1-\beta)/3$	$(3-\beta-2\beta^2)/3$	$1-\beta^3$

k	4			
K	$\langle 0,0,0,1\rangle$	$\langle 1,0,1,0\rangle$	$\langle 0,2,0,0\rangle$	$\langle 2,1,0,0\rangle$
$F_k(K)$	64/125	36/125	12/125	12/125
p_a^K	89/192	0	1/9	0
p_b^K	$(30\beta^2+16\beta^3+9\beta^4)/192$	0	$(2\beta^2+\beta^4)/9$	0
p_c^K	$(55-30\beta^2-16\beta^3-9\beta^4)/192$	0	$(3-2\beta^2-\beta^4)/9$	0
p_d^K	$\beta/4$	$(4\beta+5\beta^2+4\beta^3+3\beta^4)/16$	$(2\beta^2+3\beta^4)/9$	$(\beta^2+2\beta^4)/3$
p_e^K	$(1-\beta)/4$	$(16-4\beta-5\beta^2-4\beta^3-3\beta^4)/16$	$(5-2\beta^2-3\beta^4)/9$	$(3-\beta^2-2\beta^4)/3$

Table 3.

Proposition 5. *Let us introduce the notations* m_k, u_k, v_k, w_k, z_k *by*

$$m_k = (2\pi)^{-\frac{1}{2}} k^{-\frac{3}{2}} e^k,$$

$$u_k = a_k+b_k, \quad v_k = a_k+b_k+c_k,$$

$$w_k = a_k+b_k+d_k, \quad z_k = d_k+e_k.$$

k	0	1	2	3	4
$p_a^{(k)}$	1	0	1/3	1/4	31/125
a_k	0	0	1/3	3/8	248/375
$P_a^{(k)}$	1	0	1/3	1/4	31/125
$p_b^{(k)}$	0	0	$\beta^2/3$	$\beta^3/8$	$(38\beta^2+16\beta^3+13\beta^4)/375$
b_k	0	0	$\beta^2/3$	$3\beta^3/16$	$(304\beta^2+128\beta^3+104\beta^4)/1125$
$P_b^{(k)}$	0	0	1/9	1/32	289/5625
$p_c^{(k)}$	0	0	$(1-\beta^2)/3$	$(1-\beta^3)/8$	$(67-38\beta^2-16\beta^3-13\beta^4)/375$
c_k	0	0	$(1-\beta^2)/3$	$(3-3\beta^3)/16$	$(536-304\beta^2-128\beta^3-104\beta^4)/1125$
$P_c^{(k)}$	0	0	2/9	3/32	716/5625
$p_d^{(k)}$	0	β	$\beta^2/3$	$(5\beta+4\beta^2+\beta^3)/16$	$(300\beta+215\beta^2+108\beta^3+237\beta^4)/1500$
d_k	0	β	$\beta^2/3$	$(15\beta+12\beta^2+3\beta^3)/32$	$(600\beta-430\beta^2-216\beta^3-474\beta^4)/1125$
$P_d^{(k)}$	0	1/2	1/9	49/192	4441/22500
$p_e^{(k)}$	0	$1-\beta$	$(1-\beta^2)/3$	$(10-5\beta-4\beta^2-\beta^3)/16$	$(860-300\beta-215\beta^2-108\beta^3-237\beta^4)/1500$
e_k	0	$1-\beta$	$(1-\beta^2)/3$	$(30-15\beta-12\beta^2-3\beta^3)/32$	$(1720-600\beta-430\beta^2-216\beta^3-474\beta^4)/1125$
$P_e^{(k)}$	0	1/2	2/9	71/192	8459/22500

Table 4.

If $k\to\infty$, then the following equation system (4. 19)—(4. 23) *is asymptotically vaild for the polynomials u_k, w_k and the constants a_k, v_k, z_k:*

$$(1+e)a_k = \Sigma\Pi_a + m_k - e\int_0^1 w_{k-1}d\beta' \tag{4.19}$$

$$(1+e)u_k = \Sigma\Pi_u + m_k - e\left\{a_{k-1}\beta + \int_\beta^1 w_{k-1}d\beta'\right\} \tag{4.20}$$

$$(1+e)v_k = \Sigma\Pi_v + m_k - e\cdot a_{k-1} \tag{4.21}$$

$$(1+e)w_k = \Sigma\Pi_w + m_k - e\int_\beta^1 w_{k-1}d\beta' \tag{4.22}$$

$$z_k + v_k = m_k \tag{4.23}$$

Proof. Let us apply the substitution

$$c_{k-1}+e_{k-1} = \frac{(k-1)^{k-2}}{(k-1)!} - w_{k-1}$$

and term-wise integration in (4. 13), using (4. 17). Let analogous transformations be performed for (4. 14), (4. 15), (4. 16) (e.g., in case of (4. 14), the substitution

$$b_{k-1}+c_{k-1}+d_{k-1}+e_{k-1} = \frac{(k-1)^{k-2}}{(k-1)!} - a_{k-1}$$

is to be applied). Taking the asymptotic equalities

$$\left(\frac{k+1}{k}\right)^{k-1} \sim \left(\frac{k}{k-1}\right)^{k-2} \sim e,$$

$$\frac{k^{k-1}}{k!} \sim (2\pi)^{-\frac{1}{2}} k^{-\frac{3}{2}} e^{k}$$

into account, we get the system (4. 19)—(4. 23).

Remark. For the particular choice $\beta=0$,

$$w_k(0)=u_k(0)=a_k$$

holds (this follows from Theorem 3 as well from (4. 19)—(4. 23)).

§ 5. The connection between the type and the behavior of a tree-structure network

The types A, B, C, D, E were distinguished in § 4 in a formal way so that the reader should feel the lack of a convincing motivation. Now (as this was promised in Footnote 2) we are going to point out that the fact that a network N is contained in one or other of the types A, B, C, D, E implies entirely unlike consequences if the behaviour of the network is studied, as it was introduced in Section 3 of the former article [1], starting with the values $\beta(P)$.

We suppose that the reader is familiar with Sections 1—3 of [1]. Let N be a tree-type network, let us denote the vertices of N by $P_1, P_2, \ldots, P_{k+1}$ (where k is the number of edges of N) such that the subscripts constitute a standard labelling. To any P_i, let us assign a function $\alpha_i(t)$ by the method explained in Sect. 3 of [1] such that the initial values are determined by $\alpha_i(0)=\beta(P_i)$ (where $1\leqq i\leqq k+1$). Especially, to the root P_1 the function $\alpha_1(t)$ is attributed. We have

Proposition 6. *If the assumptions, exposed previously, are accepted, then the following six statements are valid for the network N:*

(I) *If N belongs to one of the types* A, B, C, D, E, *then the functions* $\alpha_i(t)$ *are defined at least in the interval* $[0, \tau]$ *(where* $1\leqq i\leqq k+1$*).*[4]

(II) *If N belongs to the type* A, *then* $\alpha_1(\tau)=1$.

(III) *If N belongs to the type* B, *then* $0<\alpha_1(\tau)<1$ *and there exists a t such that* $0<t<\tau$ *and* $\alpha_1(t)=1$.

(IV) *If N belongs to the type* C, *then* $0<\alpha_1(\tau)<1$ *and* $\alpha_1(t)<1$ *for every t lying in the interval* $[0, \tau]$.

(V) *If N belongs to the type* D, *then* $\alpha_1(\tau)=0$ *and there exists a t such that* $0<t<\tau$ *and* $\alpha_1(t)=1$.

(VI) *If N belongs to the type* E, *then* $\alpha_1(\tau)=0$ *and* $\alpha_1(t)<1$ *for every t lying in the interval* $[0, \tau]$.

[4] The words "at least" mean that the α_i's may also be defined for some (possibly all) values t fulfilling $t>\tau$.

Remark. Since the conclusions of (II)—(VI) exclude each other, each of (II)—(VI) holds with the formulation "if and only if" provided that N is contained in some of the five types.

Proof. (I) does not require a separate treatment (it follows from the other five assertions). To prove (II)—(VI), we use induction with respect to the number of vertices of N. The type of a network was defined in § 4 by the rules (i)—(xii) recursively; now twelve cases can be distinguished corresponding to these rules.

If N has a single vertex, then, on the one hand, it is of type A by (i); on the other hand, evidently $\alpha_1(t)=1$ if $t \geqq \tau(1-\beta(P_1))$, especially, $\alpha_1(\tau)=1$.

Assume that the number of vertices of N is $k+1$ and the assertions (II)—(VI) hold for the networks having at most k vertices. We distinguish eleven cases corresponding to (ii)—(xii).

Suppose that N is of type A by virtue of (ii). Denote (by P_1 the root of N and) by P_2 the root of the truncated network N'. There exists the edge $\overrightarrow{P_2P_1}$ and no other edge is incident with P_1 (in N). By the induction hypothesis, the conclusion of (IV) or (VI) holds for N, thus $\alpha_2(t)<1$ is valid in the whole interval $[0, \tau]$. Hence $\alpha_1(t)=1$ in the interval $[\tau(1-\beta(P_1)), \tau]$.

Assume that N belongs to the type B in consequence of (iii). Either the conclusion of (III) or that of (V) holds for N'; in both cases, $\alpha_2(t)=1$ is satisfiable with some t in $(0, \tau)$. Let t_0 be the minimal t such that $t \leqq t' \leqq \tau$ implies $\alpha_2(t')<1$ (it exists since $\alpha_2(\tau)<1$ and the functions α are continuous from right); it is clear that the value of α_1 grows from 0 to $(\tau-t_0)/\tau$ in the interval $[t_0, t]$. Because $\overrightarrow{P_2P_1}$ is a red edge, $\beta(P_2)<\beta(P_1)$, hence $\alpha_1(\tau(1-\beta(P_1))) = 1$.

If N is of type C in sense of (iv), then $\beta(P_2)>\beta(P_1)$, thus α_1 grows in the interval $[0, \tau(1-\beta(P_2)))$ from $\beta(P_1)$ towards $1-\beta(P_2)+\beta(P_1)\,(<1)$ (without reaching it), furthermore $\alpha_2(\tau(1-\beta(P_2))) = 1$ and $\alpha_1(\tau(1-\beta(P_2))) = 0$. $\alpha_1(t) \leqq \beta(P_2)<1$ whenever $\tau(1-\beta(P_2)) \leqq t \leqq \tau$.

Still we have to prove $0<\alpha_1(\tau)$. If N' is of type B, then this is obviously valid. If N' is of type D and there exists a t' such that $0<t'<\tau$ and the implication

$$t' \leqq t \leqq \tau \Rightarrow \alpha_2(t)<1$$

is true, then evidently $\alpha_1(\tau) \geqq (\tau-t')/\tau > 0$. If N' is of type D and no t' (with the mentioned property) exists, then it is clear that some α_i grows in the interval $[0, \tau]$ from 0 to 1; however, $\alpha_i(0)(=\beta(P_i))=0$ was excluded (cf. the hypothesis (H2)).

If the type of N is determined by (v) or (vi), then the proof can be carried out by similar ideas.

If one of (vii)—(xii) decides the type of N, then the conclusion of the corresponding statement of Proposition 6 can be proved by use of the subsequent principle (following from the behaviour defined in [1]): if the out-degree of P_1 is at least two, then the value $\alpha_1(t)$ (at any instant t) equals to the minimum of the values that result if the values assigned to P_1 (at t) are calculated for the several branches of N.

MATHEMATICAL INSTITUTE
OF THE HUNGARIAN ACADEMY OF SCIENCES
BUDAPEST, HUNGARY

CENTRAL RESEARCH INSTITUTE FOR PHYSICS
OF THE HUNGARIAN ACADEMY OF SCIENCES
BUDAPEST, HUNGARY

References

[1] ÁDÁM, A., Simulation of rhythmic nervous activities, II. (Mathematical models for the function of networks with cyclic inhibition), *Kybernetik*, v. 5, 1968, pp. 103—109.
[2] DE BRUIJN, N. G., *Asymptotic Methods in Analysis*, Amsterdam, 1958.
[3] DÉNES, J., The representations of a permutation as the product of a minimal number of transpositions, and its connection with the theory of graphs, *MTA Mat. Kut. Int. Közl.*, v. 4, 1959, pp. 63—71.
[4] MOON, J. W., *Counting labelled trees: a survey of methods and results*, Biennial Seminar of the Canadian Math. Congress, Vancouver, 1968.
[5] NÉMETH, G., A Stirling-sor Csebisev sorfejtése (Chebyshev expansion of the Stirling series), *Mat. Lapok*, v. 18, 1967, pp. 329—333 (in Hungarian).
[6] RÉNYI, A., Some remarks on the theory of trees, *MTA Mat. Kut. Int. Közl.* v. 4, 1959, pp. 73—85.
[7] RÉNYI, C. & A. RÉNYI, The Prüfer code for k-trees, *Colloquia Mathematica Societatis János Bolyai*, 4., Combinatorial Theory and its Applications, Balatonfüred (Hungary), 1969, pp. 945—971.
[8] RIORDAN, J., *An Introduction to Combinatorial Analysis*, New York, 1958 (Russian translation: Moscow, 1963).

(Received Sept. 11, 1970)

Dual pushdown automata and context sensitive grammars

By GY. RÉVÉSZ

1. Introduction

In recent years, a number of generalizations of pushdown automata have been studied. The basic model of pushdown automata bears an equivalence relationship to context-free grammars as shown by Chomsky [1] and Schützenberger [2]. Gray, Harrison and Ibarra extended this model as they studied two-way pushdown automata [3] while Ginsburg, Greibach and Harrison introduced stack automata [4 and 5]. A stack automaton is essentially a pushdown automaton which is allowed to scan the inside of its pushdown store without having to erase, i.e., in a read only mode. Stack automata are closely related to context sensitive grammars [6], but they are not equivalent to them. (See e.g. in [7].)

In the present paper we offer a new model called dual pushdown automaton (DUPA), since it has two pushdown stores which are complementary to each other. This model can be motivated by a normal form of context sensitive grammars which we shall see later. It can be seen that dual pushdown automata are equivalent to context sensitive grammars and, which is the same, to linear bounded automata [8 and 9].

To every context sensitive grammar in normal form we can construct a DUPA that always performs the leftmost replacement(s) while parsing sentences of the given context sensitive language. This feature may be useful for parsing from left to right, which is of great importance in connection with the direct interpretation of algorithmic languages by machine (without translation) as suggested by Kalmár [10]. Namely, according to the concept of Kalmár's formula directed computer the execution of an algorithm written in a mathematical formula language proceeds as follows. The description of the algorithm, i.e., the program of the calculation is analysed from left to right and, whenever a syntactic unit is recognized, it is semantically interpreted. Naturally, for this purpose we need a suitable language where no back tracking is necessary for the syntactic analysis. It seems useful to treat this problem with the aid of context sensitive grammars even if we are concerned with context-free languages only.

In the present paper we discuss only the basic relation of dual pushdown automata to context sensitive grammars. The problem of left-to-right parsing with respect to a specific subclass of context sensitive (namely, unilateral context sensitive) grammars has been studied in [11] whose results can very likely be generalized for context sensitive grammars in normal form. However, the problem of transforming

unfeasible grammars into suitable forms has not been solved yet in general. This problem is also related to the problem of simplifying given arbitrary dual pushdown automata.

2. Preliminaries

The set of words (including the empty word ε) over a finite set of symbols V will be denoted by V^*. Individual symbols will be denoted by small latin letters while words and sets of symbols by capitals.

Definition 1. A context sensitive grammar is a quadruple $G=(T, V, s, P)$, where T and V are finite sets of symbols, $T \subset V$, $s \in V-T$ and P is a finite set of ordered pairs — called rules — of the form $XqY \to XQY$, where $q \in V-T$ while X, Y and Q are in V^* and $Q \neq \varepsilon$ (i.e., Q non-empty).

Definition 2. A context sensitive grammar G is said to be in normal form if every rule in P is of the form $a \to b$ or $a \to bc$ or $ac \to bc$ or $ab \to ac$, where a, b and c are in V.

Definition 3. For a given context sensitive grammar G and two words A and $B \in V^*$, B is an immediate consequence of A (in symbols $A \Rightarrow B$), if there exists a rule $XqY \to XQY$ in P such that $A=UXqYZ$ and $B=UXQYZ$ for some $U, Z \in V^*$.

Definition 4. For a given context sensitive grammar and two words A and $B \in V^*$, B is derivable from A (in symbols $A \overset{*}{\Rightarrow} B$), if there exists a finite sequence of words $X_0, X_1, \ldots, X_n$ each in V^* such that $A=X_0$, $B=X_n$ and $X_i \Rightarrow X_{i+1}$ for $0 \leqq i < n$. The sequence $X_0, X_1, \ldots, X_n$ is then called a derivation of B from A with respect to G.

Definition 5. For a given context sensitive grammar G the set of words

$$L_G = \{W | s \overset{*}{\Rightarrow} W\} \cap T^*$$

is the language generated by G.

Two grammars are called weak-equivalent if they generate the same language.

A DUPA may be informally illustrated as in Fig. 1. Each move of the device is determined by the actual state of the finite state control and the topmost symbols in the two pushdown stores.

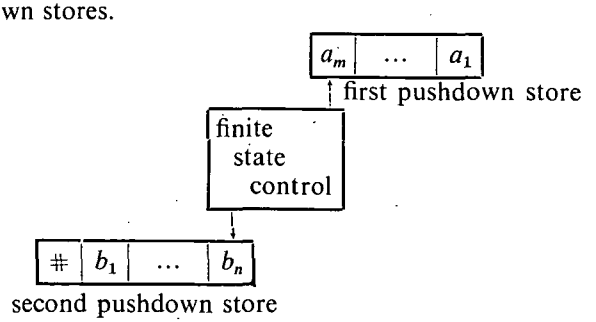

Fig. 1. Dual pushdown automaton

Each move consists of moving the two read-write heads at most one square right or left and writing a new symbol on one of the two read-write positions. The movements of the two read-write heads are coordinated such that only four types exist: $\{R, S, L, D\}$, i.e., right, stay, left, delete.

The DUPA is stopped, if it attempts to read a symbol from an empty pushdown store, i.e., if its first read-write head is to be positioned below the bottom of the corresponding pushdown store.

We now give the formal definition of a DUPA.

Definition 6. A dual pushdown automaton is an 8-tuple $A=(K, \Sigma, T, s, \#, \delta, p_1, F)$ where

(i) K is a finite nonempty set (of states),
(ii) Σ and T are finite nonempty sets (of symbols), $T \subset \Sigma$, $s \in \Sigma - T$,
(iii) $\#$ is the left endmarker: $\# \notin \Sigma$,
(iv) δ is a mapping from $K \times (\Sigma \cup \{\#\}) \times \Sigma$ into the finite subsets of $K \times \Sigma \times \times \{R, S, L, D\}$ such that $(p', x', L) \notin \delta(p, \#, x)$ and $(p', x', D) \notin \delta(p, \#, x)$ for any p, p', x, x'.
(v) $p_1 \in K$ (initial state) and $F \subseteq K$ (final states).

If the mapping δ is unique then A is deterministic otherwise it is nondeterministic.

Definition 7. A configuration of a DUPA is any element of the set $K \times \# \Sigma^* ! \Sigma^*$, where $! \notin \Sigma$.

Definition 8. Let $\vdash$ be the binary relation defined on the set of configurations as follows.

For arbitrary $a \in (\Sigma \cup \{\#\})$, and $b, c \in \Sigma$ and $X \in (\Sigma \cup \{\#\})^*$, $Y \in \Sigma^*$

$$
\begin{aligned}
&(p, Xa!bY) \vdash (p', Xac!Y) \quad \text{if} \quad (p', c, R) \in \delta(p, a, b),\\
&(p, Xa!bY) \vdash (p', Xa!cY) \quad \text{if} \quad (p', c, S) \in \delta(p, a, b),\\
&(p, Xa!bY) \vdash (p', X!cbY) \quad \text{if} \quad (p', c, L) \in \delta(p, a, b),\\
&(p, Xa!bY) \vdash (p', X!cY) \quad \text{if} \quad (p', c, D) \in \delta(p, a, b).
\end{aligned}
$$

Definition 9. Let $\models$ be the transitive closure of $\vdash$, i.e., for configurations z and z', $z \models z'$ if there exists a sequence of configurations $z_0, z_1, \ldots, z_n$ such that $z_0 = z$, $z_n = z'$ and $z_i \vdash z_{i+1}$ for $0 \leqq i < n$.

Definition 10. A word $W \in \Sigma^*$ is accepted by a DUPA if $(p_1, \# !W) \models (p_f, \# s!)$ for some $p_f \in F$.

Definition 11. The set of all terminal words ($W \in T^*$) accepted by a DUPA is called the language accepted by it.

3. The relationship of DUPA to context sensitive grammars

Theorem 1. The language accepted by a DUPA can be generated by a context sensitive grammar in normal form.

Proof. To each DUPA we construct a context sensitive grammar as follows. Let $a_i \in V$ for every $a_i \in \Sigma$. Moreover to every pair (p_j, a_i) in $K \times \Sigma$ a new element

$a_i^{(j)} \in V$ will be defined. The set of rules will be defined such that

$$\begin{aligned} a_k^{(i)} &\to a_r^{(j)} \in P \quad \text{if} \quad (p_i, a_k, R) \in \delta(p_j, \#, a_r) \\ &\quad \text{or} \quad (p_i, a_k, S) \in \delta(p_j, \#, a_r), \\ a_l a_k^{(i)} &\to a_l a_r^{(j)} \in P \quad \text{if} \quad (p_i, a_k, R) \in \delta(p_j, a_l, a_r) \\ &\quad \text{or} \quad (p_i, a_k, S) \in \delta(p_j, a_l, a_r), \\ a_k^{(i)} a_r &\to a_l^{(j)} a_r \in P \quad \text{if} \quad (p_i, a_k, L) \in \delta(p_j, a_l, a_r), \\ a_k^{(i)} &\to a_l^{(j)} a_r \in P \quad \text{if} \quad (p_i, a_k, D) \in \delta(p_j, a_l, a_r). \end{aligned}$$

In addition to that

$$a_k^{(1)} \to a_k \in P \text{ for every } a_k \in T,$$

$$\left.\begin{aligned} a_l^{(i)} a_r &\to a_l a_r^{(i)} \in P \\ a_l a_r^{(i)} &\to a_l^{(i)} a_r \in P \end{aligned}\right\} \text{ for every } i, l, r$$

and

$$s \to a_r^{(j)} \in P \quad \text{if} \quad (p_f, s, R) \in \delta(p_j, \#, a_r) \text{ for}$$

some $p_f \in F$.

It can be easily verified that each word accepted by the DUPA can be generated by the grammar, if we follow the way of accepting the given word in reversed order.

On the other hand, to each word generated by the grammar a sequence of moves of the DUPA can be specified that corresponds to the reversed derivation of the given word.

Some of the rules of the grammar constructed above are of the form $ab \to cd$, which is not allowed in the normal form (see Definition 2.), but each of these can be replaced by three rules of the form $ab \to ab'$, $ab' \to cb'$ and $cb' \to cd$.

Theorem 2. The language generated by a context sensitive grammar is accepted by a DUPA having one internal state only.

Proof. It is known that each context sensitive grammar is weak-equivalent to one in normal form [9]. Thus, we have to consider context sensitive grammars in normal form only. The corresponding DUPA will be defined as follows:

Let $\Sigma = V$ and the mapping δ defined such that if $a_k \to a_r \in P$ then $(p_1, a_k, S) \in \delta(p_1, a_l, a_r)$ for every $a_l \in V$, if $a_k \to a_l a_r \in P$ then $(p_1, a_k, D) \in \delta(p_1, a_l, a_r)$, if $a_l a_k \to a_l a_r \in P$ then $(p_1, a_k, S) \in \delta(p_1, a_l, a_r)$, if $a_k a_r \to a_l a_r \in P$ then $(p_1, a_k, L) \in \delta(p_1, a_l, a_r)$. Moreover

$$\begin{aligned} &(p_1, a_r, R) \in \delta(p_1, a_l, a_r) \\ &(p_1, a_r, R) \in \delta(p_1, \#, a_r) \\ &(p_1, a_l, L) \in \delta(p_1, a_l, a_r) \end{aligned}$$

for every a_l, a_r in V.

It can be seen again that each word generated by the grammar is accepted by the DUPA and vica versa.

Corollary. Each DUPA is equivalent to a DUPA having one internal state only.

Thus, we can say that the finite state control of the DUPA is superfluous since it can be replaced by a single state control.

Naturally the number of internal states will be decreased at the cost of increasing the number of auxiliary symbols. The construction of a minimal (in some sense) DUPA to a given context sensitive grammar is an open question.

Deterministic DUPA can be easily implemented and used for practical purposes, but it is to be ensured that the language to be recognized is of suitable structure. Usually the grammar generating the language must be transformed into an appropriate form (if possible) and the transformed grammar is more complex than the original one. These questions are not discussed here, since they are not sufficiently elaborated yet.

References

[1] CHOMSKY, N., Context-free grammars and pushdown storage, *Quart. Progress Report*, v. 65, Research Laboratory of Electronics, MIT, 1962.
[2] SCHÜTZENBERGER, M. P., On context-freee languages and push-down automata, *Information and Control*, v. 6, 1963, pp. 246—264.
[3] GRAY, J. N., M. A. HARRISON, O. H. IBARRA, Two-way pushdown automata, *Information and Control*, v. 11, 1967, pp. 30—70.
[4] GINSBURG, S., S. A. GREIBACH, M. A. HARRISON, One-way stack automata, *J. Assoc. Comput. Mach.*, v. 14, 1967, pp. 389—418.
[5] GINSBURG, S., S. A. GREIBACH, M. A. HARRISON, Stack automata and compiling, *J. Assoc. Comput. Mach.*, v. 14, 1967, pp. 172—201.
[6] HOPCROFT, J. E. & J. D. ULLMAN, Sets accepted by one-way stack automata are context sensitive, *Information and Control*, v. 13, 1968, pp. 114—133.
[7] AHO, A. V. & J. D. ULLMAN, The theory of languages, *Math. Systems Theory*, v. 2, 1968, pp. 97—125.
[8] LANDWEBER, P. S., Three theorems on phrase structure grammars of type 1. *Information and Control*, v. 6, 1963, pp. 131—136.
[9] KURODA, S. Y., Classes of languages and linear bounded automata, *Information and Control*, v. 7, 1964, pp. 207—223.
[10] KALMÁR, L., On a digital computer which can be programmed in a mathematical formula language. *II. Hungarian Mathematical Congress*, v. 2, Section V. Budapest 1960, pp. 3—16.
[11] RÉVÉSZ, GY., Syntactic analysis and unilateral context sensitive grammars, *Studia Sci. Math. Hungar.*, v. 4, 1969, pp. 267—278.

(Received Oct. 30, 1970)

Замечание к теореме о полноте системы конечных автоматов

F. FERENCZY

В настоящей статье мы указываем на ошибку в доказательстве теоремы о полноте системы конечных автоматов и исправим ее. Эта теорема первый раз была опубликована в статье А. А. Летичевского [1] и мы будем предполагать знакомство читателя с этой работой.

Упомянутая теорема А. А. Летичевского гласит:

Для того, чтобы система автоматов обладала свойством полноты, необходимо и достаточно, чтобы она содержала автомат с разделяющим состоянием.

Доказательство достаточности этой теоремы в [1] опирается, в частности, на тот факт что, используя автомат с разделяющим состоянием, можно реализовать автомат, имеющий соединимую систему множеств. Рассматриваются два типа автоматов с разделяющим состоянием a_0, которые мы будем называть α-автоматами и β-автоматами, соответсвенно. Именно, в случае $a_1 \neq a_0$ и $a_1' \neq a_0$ мы говорим об α-автомате, а в случае $a_1 = a_0 - 0$ β-автомате.

В доказательстве для случая α-автомата, используются пути вида

$$s = (a_1, a_2, \ldots, a_{m-1}, a_0)$$

и

$$s' = (a_1', a_2', \ldots, a_{n-1}', a_0)$$

в этом автомате. Хотя этот факт не подчеркивается, ясно, что требуется от каждого из этих путей не повторять состояния, т. е. $a_i \neq a_j$ $(i \neq j)$ в s, и $a_0 \neq a_i'$, $a_i' \neq a_j'(i \neq j)$ в s'. Между тем выясняется, что применяя к α-автоматам способ, указанный в [1], нелзя всегда получить автомат, обладающий соединимой системой множеств. Рассмотрим для примера автомат $\mathscr{A} = (A, X)$ с множеством состояний $A = \{a_0, a_1, a_1', b\}$ и множеством входов $X = \{x_0, y_0\}$, со следующей таблицей переходов:

	a_0	a_1	a_1'	b
x_0	a_1	a_0	a_1	b
y_0	a_1'	b	b	b

Поскольку $a_0 x_0 = a_1$, $a_1 x_0 = a_0$, и $a_0 y_0 = a_1'$, $a_1' x_0 = a_1$, $a_1 x_0 = a_0$, то a_0 есть разделяющее состояние автомата $\mathscr{A} = (A, X)$, и в качестве используемых путей имеем:

$$s = (a_1, a_0), \qquad s' = (a_1', a_1, a_0).$$

Согласно доказательству проведенному в [1], мы построим автомат $\mathscr{A}^5$ — прямое произведение, в которое $\mathscr{A}$ входит 5 раз. Пусть состояния, принадлежащие V, будут $u=(a_1, a_1, a_0, a_0, a_1')$ и $v=(a_1, a_1, a_1', a_0, a_0)$. Если мы теперь переведем u в состояние, принадлежащее V_1, а v в состояние, принадлежащее V_2, способом указанным в [1], получим одно и то же состояние $(a_0, a_0, a_1, a_1', a_1)$, и оказывается неправильным утверждение о том, что множества V_1 и V_2 — непересекающиеся. Этот неприятный факт случается очевидно из-за того, что состояние a_1 входит и в путь s', т. е. $a_2'=a_1$.

Отсюда видно, что доказательство возможности реализации автомата со соединимой системой множеств с помощью α-автомата в [1] остается в силе только тогда, когда в α-автомате найдутся пути s и s', непересекающиеся ни в одном из состояний a_1 и a_1'. α-автоматы с этим свойством назовем α_1-автоматами. Рассмотренный пример автомата $\mathscr{A}=(A, X)$ одновременно показывает, что существуют α-автоматы не являющиеся α_1-автоматами. Такие α-автоматы назовем α_2-автоматами.

Мы теперь покажем, что используя α_2-автомат также возможно реализовать автомат, имеющий соединимую систему множеств. Легко видеть, что у каждого α_2-автомата найдутся пути s и s', не повторяющие состояния и пересекающиеся только в одном из состояний a_1 и a_1'. Предположим, что у α_2-автомата $\mathscr{A}$, s и s' пересекаются в a_1. Это значит, что для подходящего $k(1\leqq k\leqq n-2)$ имеем $a_{k+1}'=a_1$. Можем убедиться также, что состояние a_k', предшествующее состоянии a_1 в пути s', не входит и в путь s. Когда бы это сучилось существовали бы и такие пути s и s', которые не пересекаются ни в одном из состояний a_1 и a_1', т. е. тогда автомат $\mathscr{A}$ не может быть α_2-автоматом.

Строим автомат $\mathscr{A}^{2(m+n)}$ — прямое произведение, в которое $\mathscr{A}$ входит $2(m+n)$ раз. В этом автомате рассмотрим множество V состояний, обладающих свойством: каждое $a_i(i\neq 0)$ и a_i' входит в качестве компоненты 2 раза; если для некоторых состояний выполняются равенства вида $a_i=a_j'$, то такие состояния должны входить 4 раза: два раза в качестве a_i и два раза в качестве a_j'; состояние a_0 входит в качестве компоненты 4 раза. Так, в каждом состоянии из V, a_1 имеет точно 4 вхождений, а a_1' и a_k' — точно 2 вхождений в качестве компоненты. Разобьем множество V на два непересекающиеся множества V_1 и V_2 так, чтобы V_1 содержало те и только те состояния из V, у которых ни одна компонента, равная a_1, не расположена между двумя компонентами, равными a_1'. Так как у каждого состояния v, принадлежащем V, есть два компонента, равных a_0, между которыми не расположен ни один компонент, равняющийся a_k', и одновременно существуют два компонента, равных a_0, между которыми расположен по крайней мере один компонент, равняющийся a_0, то имеются входы z_1 и z_2 такие, что vz_1 и vz_2 принадлежат V_1 и V_2 соответственно, т. е. V_1 и V_2 образуют соединимую систему множеств.

A remark on the theorem of the completenes of the systems of finite automata

The author showed that the proof of the following theorem of A. A. Letičevskij, *A system of finite automata is complete if and only if contains an automaton whith a dividing state*, published in his paper *Uslovyja polnoty dlja konečnyh avtomatov* (Zurnal vyčislitel'noj matematiki i matematičeskoj fiziki, v. 4, 1961, pp. 702—710), conatins an error and corrected it.

Namely, in the sufficiency proof two classes of automata whith dividing state were considered. However the proof for the first class cannot be applied to all the automata which should be covered by the class. Therefore, in the present paper the author has divided the first mentioned class into two subclasses and has completed the proof for the mentioned subclass for which the origina Letičevskij-s proof was not applicable.

Литература

[1] Летичевский, А. А., Условия полноты для конечных автоматов, *Журнал вычислительной математики и математической физики*, т. 4, 1961, стр. 702—710.

(Поступило 11-ого февраля 1971 г.)

Computer simulation of the information preprocessing in the input of the cerebellar cortex

BY A. PELLIONISZ

1. Introduction

Since the classical studies of Ramon y Cajal (1911) extensive work has been carried out in order to reveal the neuronal organization of the cerebellar cortex. The morphology has been elucidated in many particularly also ultrastructural details by neuroanatomists (Fox *et al.* 1954, 1962, 1967; Szentágothai and Rajkovits, 1959; Gray, 1961; Hámori, 1964; Hámori and Szentágothai, 1964, 1965, 1966) and tentative circuit diagrams have been suggested (Szentágothai, 1963, 1965). The physiological properties of different types of neurons have been established electrophysiologically (especially by Eccles and his collaborators, 1964, 1966). As a result of these studies considerable progress was attained also in the interpretation of the function of the cerebellar neuronal circuits which has led to the possibility of some structuro-functional synthesis of the cerebellar network. (Eccles *et al.* 1967a.)

All these efforts have paved the way for preliminary attempts at computer simulations of the cerebellar neuron network (Pellionisz, 1970). By simulation of cerebellar neuronal fields of restricted but nevertheless substantial size (in the order of 10^4 neurons) one could get some insight into the holistic activity of whole fields of the cerebellar cortex. In our first step at modeling the cerebellar circuits neurons were considered as McCulloch-Pitts type elements, and the transfer of an arbitrary random excitation pattern arriving simultaneously through the mossy fibers was simulated.

In this paper the simulation of the transfer of excitation patterns is applied with the objective of a further analysis of the mossy fibre input. First, in order to explain the structural basis of this approach, a short review of the neuronal arrangement of the cerebellar granular layer will be given. As this layer receives all the mossy fibre input, any volleys of information (before entering the higher layers of the cerebellar cortex) undergo a certain kind of preprocessing in this remarkably simple and regular neuronal structure. Looking at this structure the first obvious question that comes to one's mind is: What may be the functional significance of this preprocessing? In the first part of this paper it will be shown how this question might be answered by analyzing the transfer of excitation patterns in the model neuron circuit. The second part is to demonstrate that even complex physiological events can be readily explained by this approach: The electrophysiologically observed "pattern sensitive" inhibition of the Golgi cells (Precht and Llinás, 1969) will be interpreted by computer simulation of excitation patterns.

2. Neuronal organization of the cerebellar granular layer

It would be far beyond the scope of this paper to discuss in full details the structure of this neuron arrangement. The reader is, therefore, referred to the anatomical literature and the recent comprehensive treatment by Eccles *et al.* 1967 a. Characteristic features of the architecture of the granular layer are shown in the Fig. 1. The inputs to the layer are the mossy fibres (MF) that upon entering the layer branch several times and develop presynaptic expansions, each giving rise to a complex synaptic apparatus, to a so-called "cerebellar glomerulus" (GL). The space is densely packed with very small granule cells (GR) having 3—5 dendrites each. Their axons constitute the output lines of the layer. They ascend to the molecular layer, where they bifurcate in *T*-shape manner to give rise to the parallel fibres (PF). The terminals

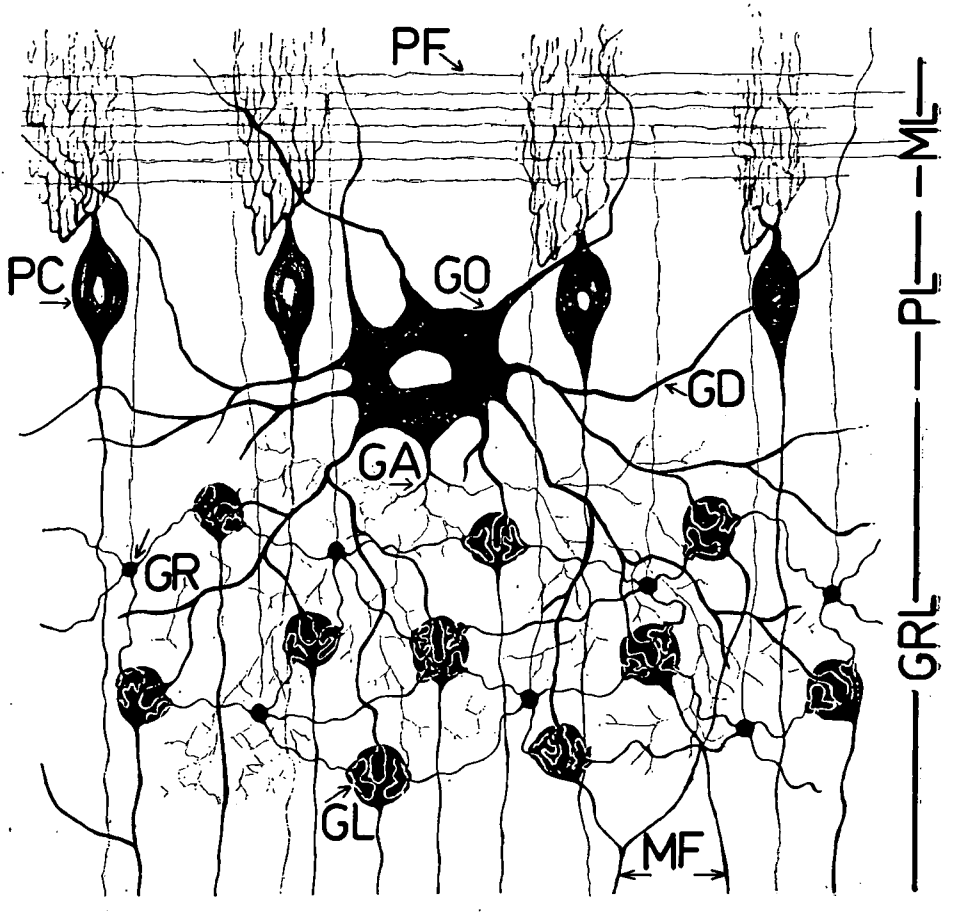

Fig. 1. Simplified, schematic view of the cerebellar architecture. Lamination of the cortex is indicated at right: GRL: granular layer; PL: Purkinje cell layer; ML: molecular layer; MF: mossy fibres; GL: cerebellar glomeruli; GR: granule cells; GO: Golgi cell; GD: Golgi cell dendrites; GA: Golgi axon; PC: Purkinje cell; PF: parallel fibres.

of the mossy fibres in the glomeruli establish excitatory synapses with the granule cell dendrites. A glomerulus receives always only one mossy fibre terminal. The mossy fibres synapse in the glomeruli also with the descending dendrites (GD) of the Golgi cell (GO). The axonal terminals of the Golgi cell (GA) descend also into the glomeruli, and exercise a postsynaptic inhibitory influence on the dendrites of granule cells. (Golgi cells have excitatory synapses also with the parallel fibres, this indirect input, however, will be neglected in the model for the time being.)

3. Pattern-transfer in the mossy fibre-granule cell neuronal net

In order to model the function of the structure, a connectivity chart has to be deduced first. Fig. 2 shows a simplified model of the connectivities among mossy fibre terminals (glomeruli) and granule cells, by placing all these neurons into a two dimensional field. The mossy fibres entering the layer end in a glomerulus each. Granule cells are assumed to have four dendrites, which enter into glomeruli situated around the granule cell. The functioning of this system can be visualized (Pellionisz, 1970) by considering a pattern of the excited glomeruli at a particular instant, and computing the transfer of this excitation pattern to the granule cells, if they are considered McCulloch-Pitts elements.

In Fig. 2 the glomeruli, considered to be excited at a particular instant, for example, are shown in black. Let us assume that the threshold of the granule cells be 3, i.e. simultaneous excitation of three of the four glomerular synapses would fire the granule cell. The granule cells excited under these circumstances are also shown in black. In this way, the pattern of excited glomeruli is easily transformed into a granule cell excitation pattern. But as nobody knows the real threshold of the granule cells, all the four possibilities have to be considered (each granule cell having four synaptic sites, of unitary function each, it has obviously four possible thresholds, i.e. if no other influence were exercised upon the granule cell). The transfer for all the four possible thresholds are shown in Fig. 3. A randomly generated pattern of active glomeruli are shown here and the transfer into excitation patterns of granule cells if their threshold is supposed to be 1, 2, 3 or 4, respectively. From these patterns one gets the visual impression that — independently of the threshold — as a result of the pattern-transformation a

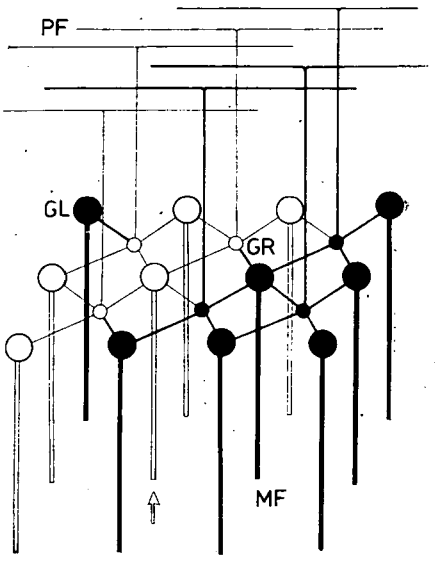

Fig. 2. Model of the mossy fibre-granule cell neuronal connexions. MF: mossy fibres; GL: glomeruli; GR: granule cells; PF: parallel fibres. Mossy fibres (and glomeruli) supposed to be excited in a particular instant are shown in black. If the granule cells are considered threshold elements, granule cells, shown in black are excited (granule cell threshold is assumed to be 3). Note, that the state of the mossy fibre marked with arrow is irrelevant (c. f. p. 160).

concentration of the excitatory spots emerges. At the granule cell threshold of 1 or 4, however, the result of the transfer is an almost entirely black (or white) pattern. The granule cell threshold, therefore, seems very unlikely to be 1 or 4; it is most probably 2 or 3. This preprocessing, however, can be interpreted not only as a visual impression but can be considered from a theoretical aspect as well:

Note, that in Fig. 2 the mossy fibre, marked with arrow, carries *no information* under the existing conditions: i.e. no matter, whether excited or not, the granule cell pattern would remain the same. That means, that there is a *redundancy* in the functioning of the mossy fibre-granule cell cerebellar input channel, which provides an increased reliability in this input. The error-suppressing effect of this redundant transformation can be numerically estimated:

Determine the probability that an erroneous activity of a single mossy fibre terminal (i.e. an excited state instead of non-excitation, or vice versa) *does not effect*

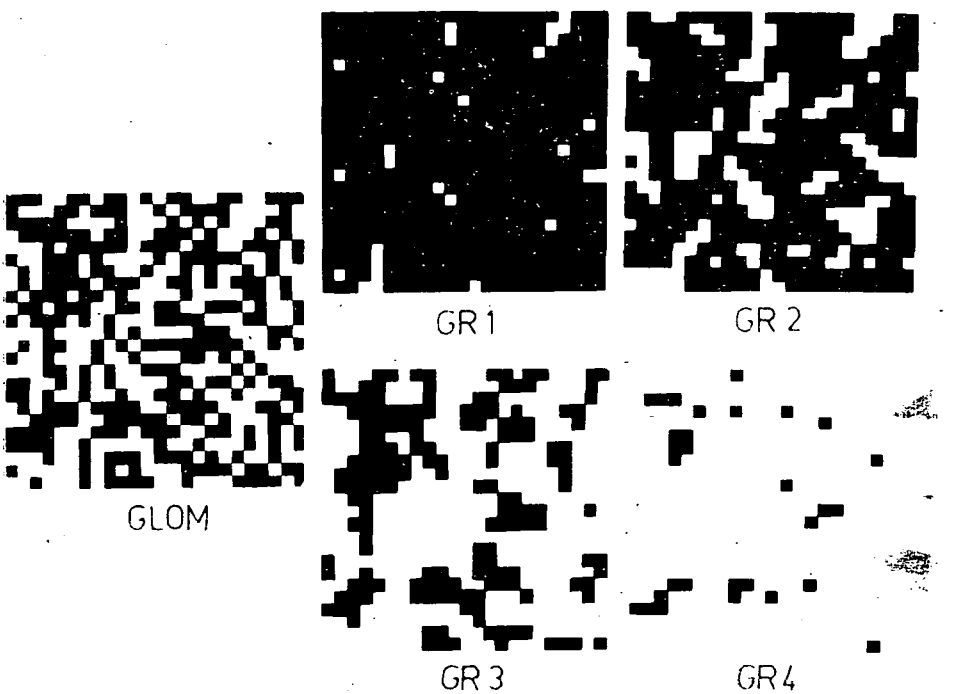

Fig. 3. Transfer of a randomly generated excitation pattern of 24×24 glomeruli (GLOM) to patterns of excited granule cells if their threshold is considered, 1, 2, 3, 4 in GR 1, GR 2, GR 3, GR 4 respectively. Black squares represent excited neurons.

any change in the granule cell excitation pattern. Consider that every mossy fibre terminal (glomerulus) is connected to four granule cells in the model, and these cells in turn are connected to eight other mossy fibre terminals (see Fig. 2). Thus each erroneously activated mossy terminal has $2^8=256$ different possible pattern-environments. The granule cell patterns have been computed for all the 256 possible cases at 1, 2, 3 or 4 values of the granule cell threshold. At 1 or 4 values of the

granule cell threshold, no change in granule cell excitation pattern would occur in 161 cases if the state of the central mossy fibre terminal were changed to 1 from 0 or vice versa. At 2 or 3 values of the threshold the output pattern is indifferent to a single change in the input pattern in 47 cases. Therefore, if the probability of all the possible patterns is considered equal, the probability, that a single error of a mossy fiber terminal will not be carried on to the granule cell pattern is 0,63 (at 1 or 4 values of the threshold) and 0,18 (at 2 or 3 values of the granule cell threshold). The 0,63 probability of the error suppressing largely limits the capacity of the mossy fibre channel, and in addition, very asymmetrically: the threshold of 1 favours low activity-level in the mossy fibre patterns, the threshold of 4 favours the highly excited patterns. As there is no reason to postulate such asymmetry in the functioning of the pattern transfer, the 1 or 4 threshold seems again unprobable, as long as Golgi inhibition is not introduced. This case is discussed in a following study (Szentágothai and Pellionisz, 1971).

It is worth mentioning, that besides the redundancy of the transform itself, there is another kind of redundancy in the flow of information: the *redundancy in the neuronal structure.* In the model the number of glomeruli and granule cells are considered equal, consequently the numbers of the possible input- and output patterns are identical, both being 2^n (if n is the number of the elements in the pattern).

In the real cerebellar granular layer, however, there are about 27 times as many granule cells as there are glomeruli (Palkovits *et al.* 1972) and, therefore, there can be approximately 2^{27n} output patterns, while the number of the possible inputs is 2^n. Considering, that every glomerulus pattern determines one and only one granule cell pattern (if the granule cell threshold is fixed) the structural redundancy is enormous.

Both considerations lead to the notion that the granular layer might play an error-suppressing role in the mossy fibre input of the cerebellar cortex. It is worth while to draw attention to the fair agreement between a real neuronal structure and theoretical studies (Neumann, 1956) dealing with formal neuronal networks, in which information restoring organs in such networks had been postulated.

4. Model of the pattern-sensitive Golgi cell inhibition in the granular layer

In this Chapter an experimental observation of complex interaction events in the mossy fibre input will be demonstrated and explained by computer simulation of the excitation pattern transfer.

In experiments performed by Precht and Llinás (1969) mass activity of granule cells had been recorded by microelectrodes introduced into the floccular area of cat's cerebellum. The field potenctial of great many granule cells could be evoked by electrical stimulation both of the ipsilateral and of the contralateral VIIIth nerve, since there is an overlapping mossy fibre input to this area of the granular layer from both the ipsi- and contralateral VIIIth nerves.

If the test stimulus was preceeded by an identical conditioning volley, the second granule cell field response was drastically reduced. This phenomenon is attributed to a Golgi cell inhibition exercised upon the granule cells (see Fig. 1) and it is in good accordance with the morphological observation by Hámori and Szentágothai

(1966) that Golgi cells have fairly large direct inputs from the mossy fibres. (Subsequently these connexions have been also confirmed electrophysiologically by Eccles *et al.* 1967a.) Figs. 4, 5, 6 show experimentally measured field potential recordings (EXP) (by Precht and Llinás, 1969), and the simulated results (SIM) (see Figs. 8, 9, 10). In Fig. 4 A shows the field potential evoked by single ipsilateral stimulation of the VIIIth nerve. In AA the response had been conditioned by an identical

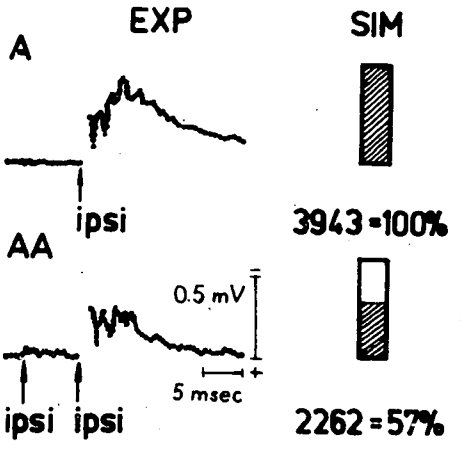

Fig. 4

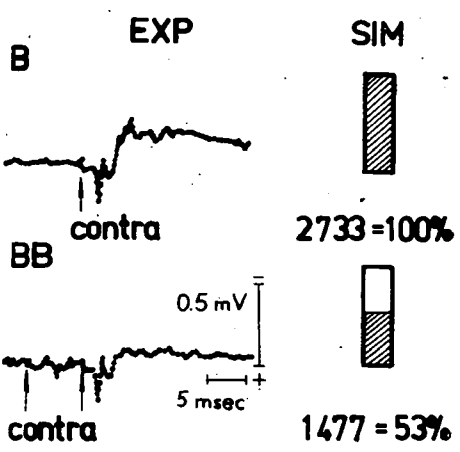

Fig. 5

Composite diagram of experimental recordings (EXP, after Precht and Llinás, 1969) and computer simulated results (SIM) of homonymous ipsilateral VIIIth nerve stimulation. A shows granule cell field potential evoked by single stimulation. In AA the stimulus was conditioned ipsilaterally; the response is considerably reduced. Arrows show the location of the stimuli. Simulated results indicate the number of active granule cells in the simulated pattern (c. f. Fig. 8).

Responses at homonymous contralateral stimulation (c. f. Fig. 4). In B the granule cell field was evoked by single contralateral impulse; in BB a double contralateral impulse was applied: shown as Fig. 4.

preceeding stimulus. Records A and AA are averaged from 16—16 responses. AA shows the response to the second stimulus exclusively as the first response has been subtracted from this record.

Similarly, Fig. 5 shows the responses to homonymous contralateral stimulation. In B the field potential was evoked by single contralateral stimulus, in BB the response had been conditioned also contralaterally. The amplitudes of the second responses in Figs. 4 and 5 are considerably reduced.

At heteronymous stimulation, however, when the ipsilateral stimulation had been conditioned by preceeding contralateral stimulus (or vice versa) the second response showed only a slight decrease: In Fig. 6 BA shows the *only slightly reduced* granule cell field potential response to ipsilateral VIIIth nerve stimulation (conditioned by contralateral stimulus), AB shows the response evoked by contralateral stimulus if the conditioning stimulus was applied ipsilaterally.

This unexpected phenomenon, labelled as "pattern sensitive Golgi cell inhibition" will be modelled and an attempt at its explanation will be made by simulat-

ing the granule cell excitation patterns emerging upon different combinations of stimulation. The model might reveal in very highly schematized form *patterns* that, if existing, would be hidden from microelectrode recording, in which only the average activity-level of a pattern can be measured.

The model considers a two-dimensional field, consisting of 100×100 glomeruli and 100×100 granule cells, in a configuration as shown in Fig. 2. First a pattern-pair of the excited glomeruli is generated by a computer according to the ipsi- and contralateral stimulation. Then the granule cell patterns are computed from these input patterns, without and with considering the effect of Golgi inhibition.

It is supposed that the 10^4 glomeruli are innervated exclusively by two (an ipsilateral and a contralateral) bundles of mossy fibres. In order to try to imitate the realistic innervation of a field of glomeruli by a single mossy fibre bundle, let us assume a quasi-random distribution of the glomeruli excited for example by the ipsilateral mossy fibre bundle as follows: (Fig. 7)

1. The field of 100×100 glomeruli is divided into 100 subordinate quadrangular areas, containing 10×10 glomeruli each.
2. In each subordinate area either 30 or 70% of the glomeruli can be fired by stimulating one of the two mossy fibre bundles (B and A in Fig. 7).
3. About 50% of the subordinate areas are of 70% activity (dominant areas, marked by A), but the distribution of the dominant areas is random.

As every glomerulus can be thrown into action in the model either by ipsilateral or by contralateral stimulation, the patterns of glomeruli in Fig. 8 A GLOM and in Fig. 9 B GLOM have to be complementary to each other.

Fig. 8 shows the patterns set up in the model by ipsilateral stimulation. A GLOM shows the pattern of exited glomeruli at ipsilateral stimulation, and A GRAN shows the pattern of granule cells transformed from A GLOM assuming a granule cell threshold of 3. The number of active granule cells corresponding to the amplitude of the field potential in the experiment is shown in Fig. 8 (compare with Fig. 4). Similarly in Fig. 9 B GLOM shows the glomerulus activity pattern evoked by contralateral mossy fibre

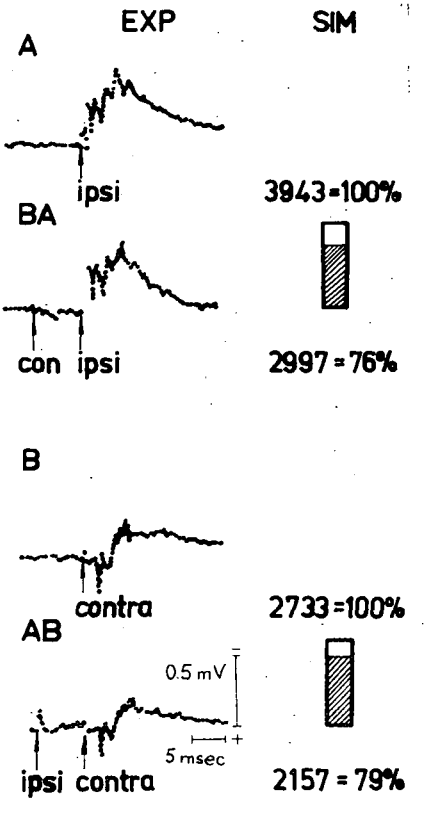

Fig. 6. Recorded (EXP) and simulated (SIM) granule cell responses at heteronymous stimulation of the VIIIth nerve. If the ipsilateral stimulation was conditioned contralaterally (BA) the second response is only slightly reduced compared to A. Similarly the granule cell field potential evoked by contralateral stimulus (B) decreases only slightly if an ipsilateral conditioning impulse is applied (AB).

activation, and B GRAN represents the transformed granule cell pattern (compare with Fig. 5).

The inhibition of mossy fibre-granule cell relay by the Golgi cells upon repeated stimulation, either ipsilaterally or contralaterally, was taken into account in this model as follows: The inhibitory Golgi cells (Fig. 1) show a territorial arrangement in the granular layer, which territories do not overlap significantly (Eccles *et al.* 1967 a). Each Golgi cell controls the mossy fibre-granule cell relay approximately in its own territory i.e. where its dendrits receive their excitatory inputs from the mossy fibres. Accordingly, the modelled neuronal field is divided into 5×5 rectangular territories (Fig. 7) corresponding to a Golgi cell each and comprising 4 neighbouring subordinate areas.

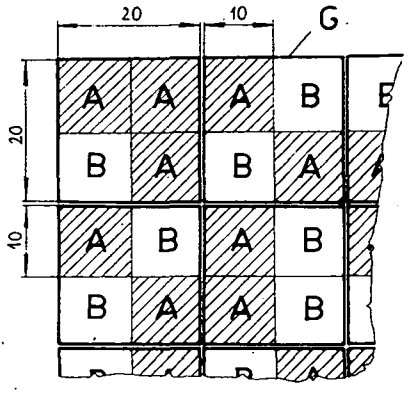

Fig. 7. Schematic diagram of a part of the 100×100 glomerulus pattern, generated by the computer so as to imitate the realistic excitation patterns emerging at for example ipsilateral stimulation. The field is divided into subordinate areas of 10×10 glomeruli. In these areas the average glomerulusactivity is randomly 70% (dominant areas, A) or 30% (B). Every Golgi cell owns four subordinate areas (G).

The Golgi cells are supposed to be thrown into action by the mossy fibres if in the majority of the 4 subordinate areas the glomerulus activity is dominant. The Golgi cells, after having been activated by a mossy fibre volley, will in turn inhibit the glomeruli in their territories for a short time. When the next glomerulus pattern appears during this period of inhibition the activity of glomeruli in these territory will be reduced (to an assumed 10%).

In Fig. 8 AA GLOM and in Fig. 9 BB GLOM shows the glomerulus-patterns evoked by the *second* stimulus at homonymous (ipsi- or contralateral) stimulation. In these patterns the most active areas of the previous A or B pattern are largely blotted out. Therefore, in the granule cell responses transformed from these patterns (in Fig. 8 AA GRAN and in Fig. 9 BB GRAN) the full number of the active granule cells is remarkably smaller (also indicated in the corresponding Figs. 4 and 5).

Upon heteronymous stimulation, however, the second response is inhibited by Golgi cells activated by the *inverse* excitation pattern: accordingly not the most active, but inversely the *least* excited areas will be suppressed by the Golgi cell inhibition. See in Fig. 10, where AB GLOM shows the response to the second B stimulus, conditioned by a previous A stimulus, and in BA GLOM vice versa. The number of the activated granule cells, therefore, is only slightly decreased in AB GRAN (Fig. 10) as compared to BB GRAN (Fig. 9) or in BA GRAN (Fig. 10) as compared to AA GRAN (Fig. 8). (See also Fig. 6.)

The results of the model can be summarized in saying that by a computer simulation based on the micromorphology of the cerebellar granular layer it can be explained why the Golgi inhibition is more effective upon homonymous stimulation then upon heteronymous pairing of the stimuli.

It has to be emphasized that in spite of the quantitative data, the model must

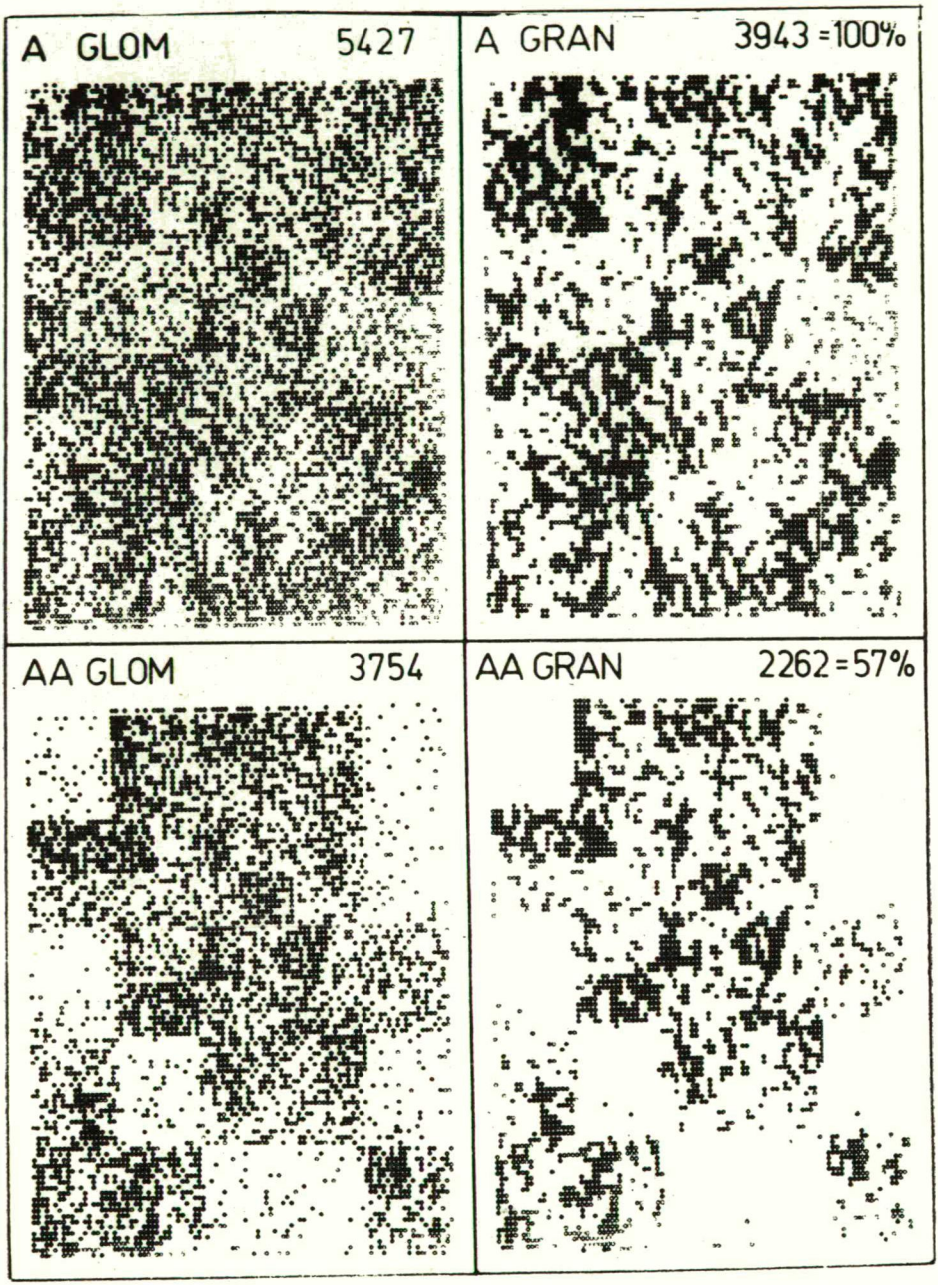

Fig. 8. Computer simulation of glomerulus- (GLOM) and granule cell (GRAN) excitation patterns at homonymous ipsilateral stimulation. (Black asterisks represent excited glomeruli or granule cells.) Responses to single ipsilateral stimulation are shown above (A), the second responses at double ipsilateral stimulation the previously most active spots in AA GLOM (and therefore in AA GRAN) are drastically inhibited by the Golgi cells. The number of the excited neurons in the patterns are also indicated (c. f. Fig. 4).

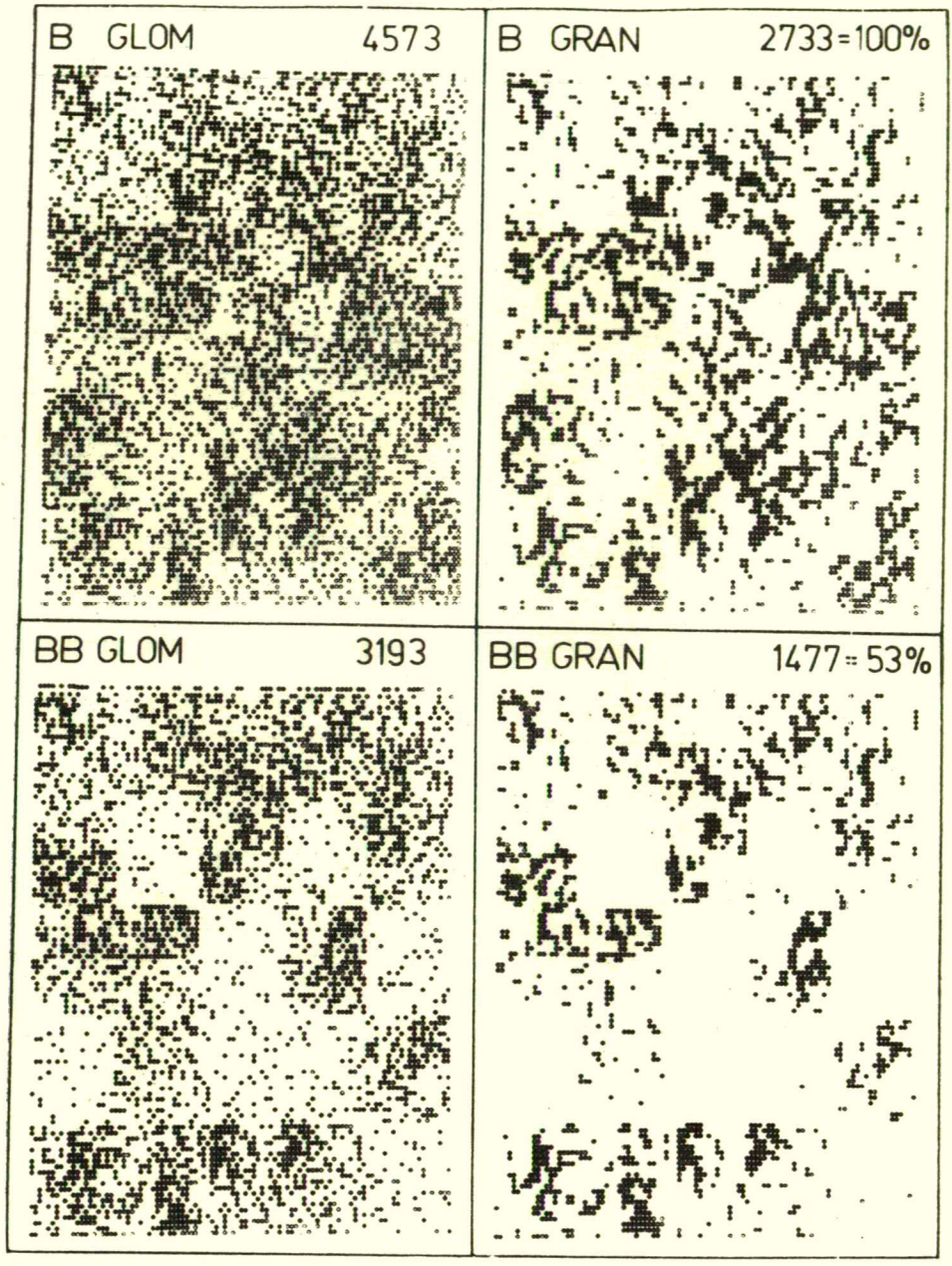

Fig. 9. Modelled excitation patterns evoked by homonymous contralateral stimulation (c. f. Fig. 9). B GLOM shows the response to single contralateral stimulation (it is the inverse pattern of A GLOM). B GRAN is computed from B GLOM by the threshold 3. At contralaterally conditioned contralateral stimulation the response is BB GLOM and BB GRAN. Note the largely reduced second responses (c. f. Fig. 5).

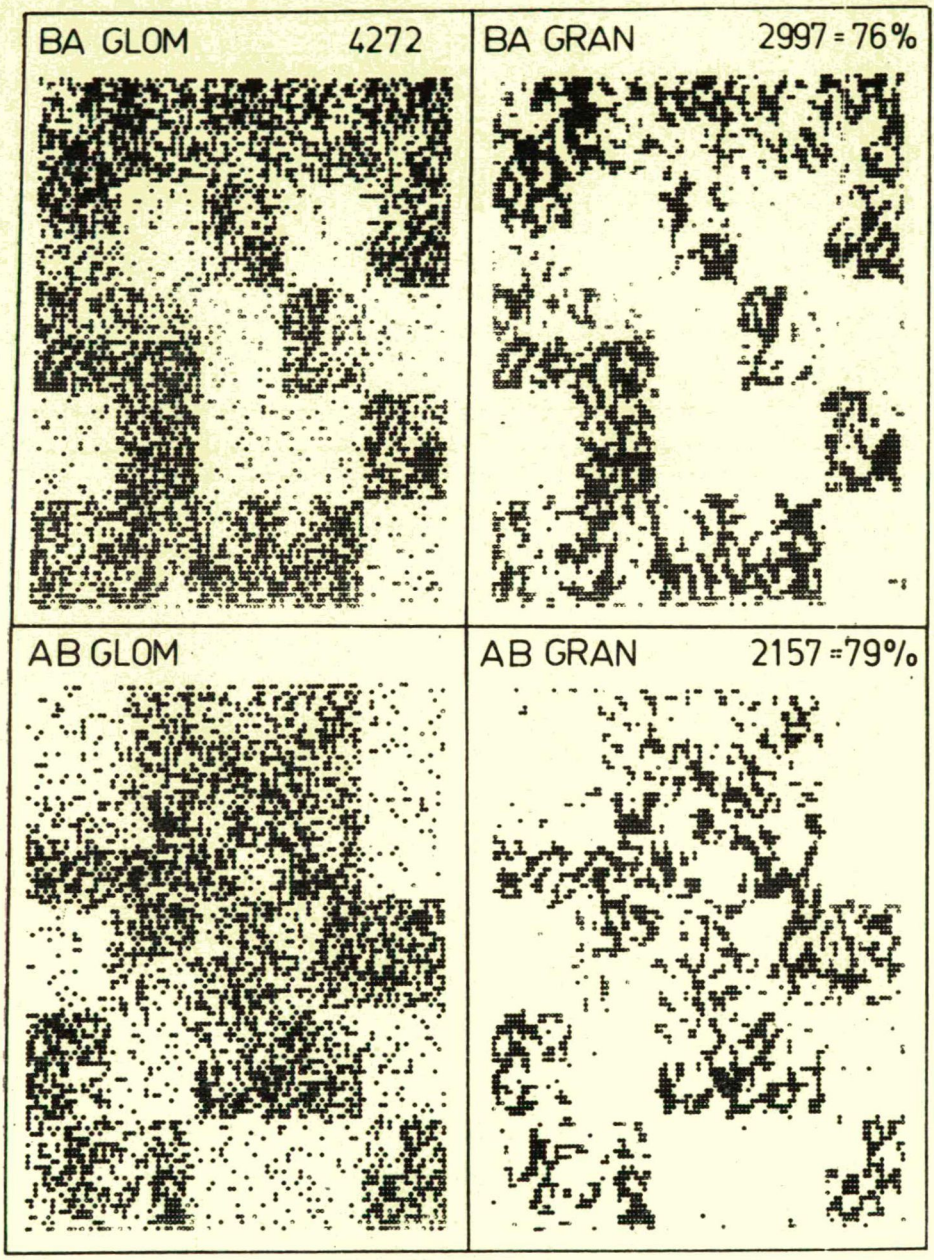

Fig. 10. Simulated patterns evoked by heteronymous VIIIth nerve stimulation. BA GLOM shows the response to ipsilateral stimulus conditioned contralaterally. The Golgi inhibition is activated by the conditioning impulse, therefore the pattern is inhibited in the least active areas. Accordingly in the transformed BA GRAN pattern the number of active granule cells is only slightly reduced as compared to A GRAN and AA GRAN. Similarly AB GLOM is the pattern of excitation at contralateral stimulation, previously conditioned ipsilaterally (c. f. Fig. 6).

not be considered as a quantitatively precise description of the physiological events but rather as a qualitative explanation of an unexpected experimental result. This remark is necessary since this model has several free parameters (as models generally do) and any variation in their values (for example in the threshold-values) can effect numerical deviations in the results. The qualitative result, however, is fairly indifferent to threshold-variations. The whole simulation has also been carried out for example with a 2 value of the granule cell threshold. In this case the results are slightly modified, however, the much smaller effectiveness of the inhibition at heteronymous stimulation has remained qualitatively identically demonstrated:

Granule cell threshold=3

response	ipsilateral	contralateral
stimulation		
homonymous	57%	54%
heteronymous	74%	78%

Granule cell threshold=2

response	ipsilateral	contralateral
stimulation		
homonymous	66%	65%
heteronymous	76%	78%

5. Acknowledgement

The simulation has been performed on a computer of ODRA 1013 type. The time required has been granted by the Chair of Numerical and Computer Mathematics fo the Roland Eötvös University, Budapest. It is gratefully acknowledged to the Head of the Chair: Dr. J. Mogyoródi.

1st DEPARTMENT OF ANATOMY
SEMMELWEIS UNIVERSITY MEDICAL SCHOOL
BUDAPEST, HUNGARY

References

ECCLES, J. C., R. LLINÁS, K. SASAKI, Golgi cell inhibition in the cerebellar cortex, *Nature*, v. 204, 1964, pp. 1265—1266.

ECCLES, J. C., R. LLINÁS, K. SASAKI, The inhibitory interneurones within the cerebellar cortex, *Exp. Brain Res.*, v. 1, 1966, pp. 1—16.

ECCLES, J. C., R. LLINÁS, K. SASAKI, Parallel fibre stimulation and responses included thereby in the Purkinje cells of the cerebellum, *Exp. Brain Res.*, v. 1, 1966, pp. 17—39.

ECCLES, J. C., R. LLINÁS, K. SASAKI, The mossy fibre-granule cell relay of the cerebellum and its inhibitory control by Golgi cells, *Exp. Brain Res.*, v. 1, 1966, pp. 82—101.

ECCLES, J. C., M. ITO, J. SZENTÁGOTHAI, *The cerebellum as a neuronal machine*, Springer-Verlag, New York, Inc. 1967.

ECCLES, J. C., K. SASAKI, P. STRATA, Interpretation of the potential fields generated in the cerebellar cortex by a mossy fibre volley, *Exp. Brain Res.*, v. 3, 1967, pp. 58—80.

FOX, C. A. & E. G. BERTRAN, Connections of the Golgi cells and the intermediate cells of Lugaro in the cerebellar cortex of the monkey, *Anat. Rec.*, v. 118, 1954, p. 423.

Fox, C. A., The structure of the cerebellar cortex, *Correlative anatomy of the nervous system*, (ed. E. C. Crosby, T. H. Humphrey, E. W. Caner, New York, MacMillan) 1962, pp. 193—198.

Fox, C. A., D. E. Hillman, K. A. Siegesmund, C. R. Dutta, The primate cerebellar cortex: A Golgi and electron microscopical study, *Progr. in Brain Res.*, ed. C. A. Fox and R. Snider, Amsterdam, Elsevier, v. 25, 1967, pp. 174—225.

Gray, E. G., Granule cells, mossy synapses and Purkinje spine synapses of the cerebellum. Light and electron microscopic observations, *J. Anat.*, v. 95, 1961, pp. 345—356.

Hámori, J., Identification in the cerebellar isles of Golgi II. axon endings by aid of experimental degeneration, *Electron Microscopy*, Proc. of Third European Regional Conf., (ed M. Titlbach, Prague, Publishing House of Chechoslov. Acad. Sci.) v. B, 1964, pp. 291—292.

Hámori, J. & J. Szentágothai, "Crossing over" synapse. An electron microscope study of the molecular layer of the cerebellar cortex, *Acta Biol. Hung.*, v. 15, 1964, pp. 95—117.

Hámori, J. & J. Szentágothai, The Purkinje cell baskets: Ultrastructure of an inhibitory synapse, *Acta Biol. Hung.*, v. 15, 1965, pp. 465—479.

Hámori, J. & J. Szentágothai, Participation of Golgi neurone processes in the cerebellar glomeruli: An electron microscope study, *Exp. Brain Res.*, v. 2, 1966, pp. 35—48.

Neumann, J., Probabilistic logics and the synthesis of reliable organisms from unreliable components, *Automata Studies*, Princeton, Univ. Press., 1956.

Palkovits, M., P. Magyar, J. Szentágothai, Quantitative histological analysis of the cerebellar cortex in the cat. IV. Mossy fibre-Purkinje cell numerical transfer, *Brain Res.* (in press).

Pellionisz, A., Computer simulation of the pattern transfer of large cerebellar neuronal fields, *Acta Biochim. et Biophys. Acad. Sci. Hung.*, v. 5, 1970, pp. 71—79.

Precht, W. & R. Llinás, Functional Organization of the vestibular afferents to the cerebellar cortex of frog and cat, *Exp. Brain Res.*, v. 9, 1969, pp. 40—52.

Ramón y & S. Cajal, *Histologie du sistème nerveaux de l'homme et des vertèbrés*, Paris, Maloine, 1911.

Szentágothai, J. & K. Rajkovits, Über den Ursprung der Kletterfasern des Kleinhirns, *Z. Anat. EntwGesch.*, v. 121, 1959, pp. 130—141.

Szentágothai, J., Újabb adatok a synapsis funkcionális anatómiájához (New data on the functional anatomy of synapses), *MTA, Biol. Orv. Tud. Oszt. Közl.*, v. 6, 1963, pp. 217—227.

Szentágothai, J., The use of degeneration methods in the investigations of short neuronal connexions. *Progr. in Brain Res.*, (ed. M. Singer and J. P. Schadé, Amsterdam, Elsevier) v. 14, 1965, pp. 1—32.

Szentágothai, J. & A. Pellionisz, The neuron network of the cerebellar cortex and attempt at its modelling by computer simulation, *Biophysics of cells and organs*, Verlag der Wiener Medizinischen Akademie, 1971, pp. 291—296.

(Received August 15, 1970)

Cutting plane methods for solving nonconvex programming problems

By F. FORGÓ

1. Introduction

It is well known that the solution of the programming problem

$$f(\mathbf{x}) \to \max \tag{1. 1}$$

subject to

$$\mathbf{x} \in L,$$

— where L is a subset of the Euclidean n-space E^n and f is a scalar-valued function — can be very difficult unless L is convex and $f(\mathbf{x})$ is quasiconcave (see: [1], [2]). For special cases of (1. 1) efficient methods have been developed among which the so called "cutting plane" methods are of considerable importance (see: [3], [4], [5], [6]).

In this paper we want to apply the cutting plane idea — developed originally in [6] for quadratic objective function, in [5] and later but independently in [7] for convex objective function — to more general programming problems including such as

maximizing a quasiconvex function over a convex polyhedron
maximizing a quasiconvex function over the lattice points of a convex polyhedron
mixed zero-one integer programming with convex objective function to be maximized
fixed charge problems with convex objective function
separable nonlinear programming with linear constraints
general continuous nonlinear programming
general pure integer programming.

2. A method for accelerating the full description method

Let the problem be the following[1]:

$$f(\mathbf{x}) \to \max \tag{2. 1}$$

subject to

$$\mathbf{A}\mathbf{x} \leqq \mathbf{b},$$

[1] Throughout the paper $\mathbf{A}, \mathbf{B} \ldots$ denote matrices, $\mathbf{a}, \mathbf{b} \ldots$ denote vectors, $*$ stands for transposition and $\mathbf{e}_j$ is the j^{th} identity vector.

where $\mathbf{x} \in E^n$, $\mathbf{b} \in E^m$, $\mathbf{A}$ is an m by n matrix, $L = \{\mathbf{x} | \mathbf{A}\mathbf{x} \leqq \mathbf{b}\}$ is nonempty and bounded, $f(\mathbf{x})$ is continuous and quasiconvex over the whole E^n. This latter means that for all $\mathbf{x}_1$, $\mathbf{x}_2$ and λ $(0 \leqq \lambda \leqq 1)$

$$f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) \leqq \max \{f(\mathbf{x}_1), f(\mathbf{x}_2)\}. \tag{2. 2}$$

It is known ([2]) that among the global maximumpoints of (2. 1) there is at least one extreme point of L. This gives a basis to a method of solving (2. 1) called "full description method" ([8], [9]) which generates in some way all extreme points of L and then we can choose that (those) extreme point(s) which give(s) the maximal objective function's value. Unfortunately in cases of practical problems this method fails because of the large number of extreme points.

In this section we give a method which is based on an arbitrary variant of the full description method but it does not require usually the determination of all extreme points.

We shall call the method capable of leading us through all extreme points of L the "wandering method". (A realization of a "wandering method" is e.g. [8] and [9]). We call a point $\hat{\mathbf{x}}$ of the convex polyhedron L a nondegenerate basic solution if $\mathbf{A}$ and $\mathbf{b}$ can be partitioned in the following manner:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}, \quad \begin{matrix} \mathbf{A}_1\hat{\mathbf{x}} = \mathbf{b}_1, \\ \mathbf{A}_2\hat{\mathbf{x}} < \mathbf{b}_2, \end{matrix} \tag{2. 3}$$

where $\mathbf{A}_1$ is nonsingular. All other basic solutions are called degenerate.

To begin with let us determine an extreme point of L, say $\mathbf{x}_0$. If $\mathbf{x}_0$ is degenerate, then applying the "wandering method" find a nondegenerate basic solution $\bar{\mathbf{x}}_0$. If $\mathbf{x}_0$ is nondegenerate, then $\bar{\mathbf{x}}_0 = \mathbf{x}_0$. Let the maximal objective function's value through the path leading from $\mathbf{x}_0$ to $\bar{\mathbf{x}}_0$ be $C_0' = C_0$. If all basic solutions of L are degenerate, then we have to determine all extreme points. In this case our method reduces to the full description method and $C_0 = \max\limits_{\mathbf{x} \in L} f(\mathbf{x})$.

Since $\bar{\mathbf{x}}_0$ is nondegenerate we can transform (2. 2) into the equivalent problem:

$$f(\bar{\mathbf{x}}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \to \max$$

subject to

$$\mathbf{y} \geqq \mathbf{O}, \tag{2. 4}$$

$$\mathbf{A}_3\mathbf{y} \leqq \mathbf{b}_3,$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}, \quad \begin{matrix} \mathbf{A}_1\bar{\mathbf{x}}_0 = \mathbf{b}_1, \\ \mathbf{A}_2\bar{\mathbf{x}}_0 < \mathbf{b}_2, \end{matrix} \quad \mathbf{y} = \mathbf{b}_1 - \mathbf{A}_1\mathbf{x}, \tag{2. 5}$$

$$\mathbf{A}_3 = -\mathbf{A}_2\mathbf{A}_1^{-1}, \quad \mathbf{b}_3 = \mathbf{b}_2 - \mathbf{A}_2\bar{\mathbf{x}}_0.$$

The objective function of (2. 4) is also quasiconvex since $f(\mathbf{x})$ is assumed to be quasiconvex over the entire E^n. Since $\bar{\mathbf{x}}_0$ is nondegenerate $\mathbf{y} = \mathbf{O}$ in problem (2. 4) has exactly n adjacent extreme points:

$$\alpha_1\mathbf{e}_1, \alpha_2\mathbf{e}_2, \ldots, \alpha_n\mathbf{e}_n,$$

where $\alpha_j > 0$ $(j = 1, \ldots, n)$.

Let

$$\bar{C}_0 = \max\{C_0', \max_{1 \leqq j \leqq n} f(\bar{\mathbf{x}}_0 - \alpha_j \mathbf{A}_1^{-1}\mathbf{e}_j)\} \tag{2.6}$$

and t_j the maximal number ($t_j = \infty$ is admitted) for which the inequality

$$f(\bar{\mathbf{x}}_0 - t\mathbf{A}_1^{-1}\mathbf{e}_j) \leqq \bar{C}_0 \qquad (j = 1, \ldots n) \tag{2.7}$$

holds ($t_j > 0$ since $f(\bar{\mathbf{x}}_0 - \alpha_j \mathbf{A}_1^{-1}\mathbf{e}_j) \leqq \bar{C}_0$).

Denote

$$\mathbf{t}^* = (1/t_1, \ldots, 1/t_n).$$

(If $t_j = \infty$, then $1/t_j = 0$ by definition.)

We shall distinguish two cases:

(i) $|\mathbf{t}^*\mathbf{A}_1| \leqq T$,
(ii) $|\mathbf{t}^*\mathbf{A}_1| > T$,

where T is a fixed positive number.

In case (i) we consider the problem:

$$f(\bar{\mathbf{x}}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \to \max$$

subject to

$$\mathbf{y} \geqq \mathbf{O}, \tag{2.8}$$

$$\mathbf{t}^*\mathbf{y} \leqq 1.$$

By (2.7) it is clear that the global maximum of (2.8) does not exceed $\bar{C}_0$. Therefore the cutting inequality

$$\mathbf{t}^*\mathbf{y} \geqq 1 \tag{2.9}$$

and its transformation by (2.5)

$$\mathbf{h}^*\mathbf{x} \leqq h_0, \tag{2.10}$$

where $\mathbf{h}^* = \mathbf{t}^*\mathbf{A}_1$ and $h_0 = \mathbf{t}^*\mathbf{b}_1 - 1$ excludes a region of L, where $f(\mathbf{x}) \leqq \bar{C}_0$.

In case (ii) let

$$\mathbf{d}^* = (1/\alpha_1, \ldots, 1/\alpha_n)$$

and consider the inequality

$$\mathbf{d}^*\mathbf{A}_1\mathbf{x} \leqq \mathbf{d}^*\mathbf{b}_1 - 1. \tag{2.11}$$

It can easily be seen that (2.11) cuts off the simplex with vertices

$$\bar{\mathbf{x}}_0, \bar{\mathbf{x}}_0 - \alpha_1\mathbf{A}_1^{-1}\mathbf{e}_1, \ldots, \bar{\mathbf{x}}_0 - \alpha_n\mathbf{A}_1^{-1}\mathbf{e}_n.$$

Adjoining inequality (2.10) or (2.11) to the original constraints of (2.1) we reduce the feasible set L. Let the new feasible set be L_1 ($L = L_0$). Then the whole procedure can be repeated with the obvious modification that in Step $k+1$

$$C_k = \max\{\bar{C}_{k-1}, C_k'\}.$$

When computing $\bar{C}_k$ by (2.6) we replace the index 0 by k and C_k' denotes the maximal objective function's value along the path leading to a nondegenerate basic solution in Step $k+1$.

It is clear that

$$L_0 \supset L_1 \supset \ldots \supset L_k \supset \ldots,$$

$$C_0 \leqq C_1 \leqq \ldots \leqq C_k \leqq C_{k+1} \ldots.$$

The procedure terminates if for some index $p \geqq 1$, $L_p = \emptyset$. Then $\mathbf{x}'$ is a solution of (2. 1) if $f(\mathbf{x}') = \overline{C}_{p-1}$.

We shall prove that the procedure terminates in finite number of steps. For the proof we need a simple lemma.

Lemma 1. If $\overline{\mathbf{x}}_k$ is the nondegenerate basic solution obtained in Step $k+1$ and cut (2. 10) is applied, then

$$|\overline{\mathbf{x}}_k - \mathbf{u}_k| \geqq T^{-1},$$

where $\mathbf{u}_k$ is the orthogonal projection of $\overline{\mathbf{x}}_k$ onto the hyperplane $\mathbf{h}^*\mathbf{x} = h_0$ $(\mathbf{h} \neq \mathbf{O})$.

Proof. By the definition of $\mathbf{h}$ and h_0 it follows that $\overline{\mathbf{x}}_k$ is on the hyperplane $\mathbf{h}^*\mathbf{x} = h_0 + 1$. Write Schwarz's inequality for $\mathbf{h}$ and $\overline{\mathbf{x}}_k - \mathbf{u}_k$

$$|\mathbf{h}^*(\overline{\mathbf{x}}_k - \mathbf{u}_k)| \leqq |\mathbf{h}| |\overline{\mathbf{x}}_k - \mathbf{u}_k|.$$

Since the left hand side equals 1 we get the desired inequality

$$|\overline{\mathbf{x}}_k - \mathbf{u}_k| \geqq |\mathbf{h}|^{-1} = |\mathbf{t}^*\mathbf{A}_1|^{-1} \geqq T^{-1}.$$

Theorem 1. There is an index $p \geqq 1$ for which $L_p = \emptyset$.

Proof. It is sufficient to prove that cut (2. 10) cannot be applied infinite times since (2. 11) cuts off a simplex and every polyhedron consists of finitely many simpleces.

Suppose on the contrary that (2. 10) occurs infinite times. Then the sequence of nondegenerate basic solutions determined in the steps when (2. 10) is used has at least one cluster point $\hat{\mathbf{x}}$ because L is bounded. Thus there is a neighbourhood $K(\hat{\mathbf{x}}, \varepsilon)$ of $\hat{\mathbf{x}}$ and an index r such that for a $k \geqq r$, $\overline{\mathbf{x}}_k \in K(\hat{\mathbf{x}}, \varepsilon)$. In the step when $\overline{\mathbf{x}}_k$ is cut off by inequality (2. 10) Lemma 1. assures that the distance of $\overline{\mathbf{x}}_k$ from the cutting plane is at least T^{-1}. Thus ε can be chosen so small that the entire $K(\hat{\mathbf{x}}, \varepsilon)$ lies on the infeasible side of the cutting plane. But this is a contradiction.

Remarks

1. If $f(\mathbf{x})$ is strictly convex that is for any $\mathbf{x}_1 \neq \mathbf{x}_2$ and $0 < \lambda < 1$ the inequality $f(\lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) < \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2)$ holds, then the method described above gives all global maximumpoints. We have never cut such points where the objective function's value equals the maximum obtained so far and since $f(\mathbf{x})$ is strictly convex every global (and local) maximumpoint is an extreme point of L.

2. It is clear that the procedure works well with an arbitrary extreme point as a starting solution in each step but it seems us more advantageous to start with a local vertex maximumpoint. (This is a point that has no adjacent extreme point of higher objective function's value.)

3. It is obvious that the efficiency of the method is greatly reduced if degeneration occurs frequently. Therefore it is of disadvantage if cut (2. 11) has to be applied many times since this cut increases the number of degenerate basic solutions. In the next section we give a variant of this method which is insensitive to degeneration.

4. By the construction of the method the number of constraints increases. But simultaniously some of the old constraints may become redundant. (We call a constraint redundant if there is no feasible point satisfying it as an equality.) For elimination of the redundant constraints the method proposed in [6] can be used.

5. The procedure can be simplified to a great extent if we are content with an "ε optimal" solution of problem (2. 1). We call $\mathbf{z}\in L$ an "ε optimal" solution if

$$\max_{\mathbf{x}\in L} f(\mathbf{x}) \leqq f(\mathbf{z})+\varepsilon, \quad \varepsilon>0.$$

In this case it is sufficient to find a local vertex maximumpoint in every step (not necessarily nondegenerate) and tranformation (2. 5) can be carried out with any basis associated with the extreme point in question. Inequality (2. 7) is changed by replacing the right hand side to $\bar{C}_0+\varepsilon$. Since $f(\mathbf{x})$ is continuous every t_j will be positive and cut (2.10) excludes a proper subset of the feasible region. Thus cut (2. 11) is not necessary and we do not need the "wandering method" too.

It is an open question whether this modified procedure terminates in finite number of steps.

3. Maximizing a quasiconvex function over the lattice points of a convex polyhedron

Let the problem be

$$f(\mathbf{x}) \to \max$$

subject to

$$\mathbf{Ax}\leqq\mathbf{b}, \tag{3. 1}$$

$$\mathbf{x}=\text{integer},$$

where

(i) $L=\{\mathbf{x}|\mathbf{Ax}\leqq\mathbf{b}\}$ is nonempty and bounded,
(ii) The entries of $\mathbf{A}$ and $\mathbf{b}$ are integers,
(iii) $f(\mathbf{x})$ is continuous and quasiconvex on E^n.

The method proposed for solving (3. 1) consists of iterational steps. In each step we reduce the feasible region. Denote the feasible set in Step k by L_k.

Step 0. Find a feasible point to (3. 1) with any method of integer programming. If there is no such point, then (3. 1) has no solution. Otherwise go to Step 1.

Step k.

a) Find a local (vertex) maximumpoint $\mathbf{x}_k$ of L_k ($L_1=L$).

b) Do transformation (2. 5) and determine t_j as the maximal positive number satisfying the inequality

$$f(\mathbf{x}_k-t\mathbf{A}_{11}^{-1}\mathbf{e}_j) \leqq C_k+\varepsilon, \quad \varepsilon>0 \qquad (j=1, \ldots, n), \tag{3. 2}$$

where $\mathbf{A}_{11}$ is a nonsingular submatrix of $\mathbf{A}_1$ (we have not assumed nondegeneracy!) and C_k is the maximal objective function's value obtained so far on lattice points of L. Then we construct the vector $\mathbf{t}_k$ as in Section 2. and test the inequality $|\mathbf{t}_k^*\mathbf{A}_{11}|\leqq T$. If it is satisfied by $\mathbf{t}_k$ or $\mathbf{x}_k=$integer, then we reduce L_k by cut (2. 10). If $\mathbf{x}_k$ has at least one noninteger component and $|\mathbf{t}_k^*\mathbf{A}_{11}|>T$, or there is no positive t satisfying (3. 2) then reduce L_k by a Gomory cut (see [1] p. 272). Let L_{k+1} be the new feasible set and go to Step $k+1$.

The procedure terminates if for some $p\geqq 1$, $L_p=\emptyset$.

Theorem 2. After finite number of steps $L_p=\emptyset$ for some $p\geqq 1$.

Proof. Cut (2. 10) cannot be applied infinite many times by the reasoning in the proof of Theorem 1. and because the number of lattice points of L is finite. Furthermore the application of the Gomory cuts provides an integer point after finite number of steps ([1] p. 276).

Remarks

1. The procedure described above gives "only" an "ε optimal" solution which is always satisfactory in practical situations. But if $f(\mathbf{x})$ takes on integral values for any integer $\mathbf{x}$ (e.g. $f(\mathbf{x})$ is a polynomial with integer coefficients), then we can replace ε by 1 and determine at least one "true" optimal solution of (3. 1).

2. It is clear that this procedure can be used instead of the method proposed in Section 2. in almost all practical cases since the integrity stipulation is very week if we choose proper scale. In addition if in (2. 1) every extreme point of L is integer (e.g. (2. 1) is a transportation problem with integer parameters [11]), then the procedure of this section can be applied without changing the scale.

4. Mixed zero-one integer programming with convex objective function to be maximized

Let us consider the problem

$$F(\mathbf{x}) \to \max$$

subject to

$$\begin{gathered} 0 \leqq x_j \leqq 1, \quad x_j = \text{integer} \qquad (j=1, \ldots, p), \quad p \geqq 1 \\ 0 \leqq x_j \leqq k_j \qquad (j = p+1, \ldots, n), \\ \sum_{j=1}^{n} a_{ij} x_j \leqq b_i \qquad (i=1, \ldots, m), \end{gathered} \tag{4. 1}$$

where $\mathbf{x}=(x_1, \ldots, x_n)$ and $F(\mathbf{x})$ is convex on E^n.

For the solution of (4. 1) we can apply the full description method. The following theorem gives the basis for doing so.

Theorem 3. Among the optimal points of (4. 1) there is at least one extreme, point of L. (L denotes the set of points satisfying the conditions of (4. 1) ignoring the integrity stipulations.)

Proof. Let $\mathbf{z}$ be an optimal solution of (4. 1). Fix the first p components of $\mathbf{z}$ and consider the problem:

$$F(z_1, \ldots, z_p, x_{p+1}, \ldots, x_n) \to \max$$

subject to

$$\begin{gathered} 0 \leqq x_j \leqq k_j \qquad (j = p+1, \ldots, n), \\ \sum_{j=p+1}^{n} a_{ij} x_j \leqq b_i - \sum_{j=1}^{p} a_{ij} z_j \qquad (i=1, \ldots, m). \end{gathered} \tag{4. 2}$$

Let $\mathbf{y}$ be an optimal extreme point of (4. 2). (There is at least one such point since $(z_{p+1}, \ldots, z_n)$ is a feasible point and $F(\mathbf{x})$ is convex.) Let

$$\mathbf{x}_0^* = (z_1, \ldots, z_p, \mathbf{y}^*)$$

$\mathbf{x}_0$ is an optimal solution of (4. 1) since $F(\mathbf{x}_0) \geqq F(\mathbf{z})$. Suppose that $\mathbf{x}_0$ is not an extreme point. Then there are points $\mathbf{x}_1 \in L$, $\mathbf{x}_2 \in L$, $\mathbf{x}_1 \neq \mathbf{x}_2$ such that $\mathbf{x}_0 = \frac{1}{2}(\mathbf{x}_1 + \mathbf{x}_2)$. Since the first p entries of $\mathbf{x}_0$ are 0 or 1 the first p components of $\mathbf{x}_1$ and $\mathbf{x}_2$ are equal. But the last $n-p$ components of $\mathbf{x}_1$ and $\mathbf{x}_2$ must coincide because $\mathbf{y}$ is an extreme point of (4. 2). This contradicts the assumption $\mathbf{x}_1 \neq \mathbf{x}_2$. Thus $\mathbf{x}_0$ is an optimal extreme point.

Our purpose is to apply the methods of Section 2. for (4. 1) to accelerate the full description method. The following theorem provides a continuous equivalent to problem (4. 1).

Theorem 4. Consider the programming problem:

$$F(\mathbf{x}) - \lambda \sum_{j=1}^{p} x_j(1-x_j) \to \max \qquad (\lambda > 0) \tag{4.3}$$

subject to

$$\mathbf{x} \in L.$$

There exists a real number $\lambda_0 > 0$ so that for all $\lambda \geqq \lambda_0$ the set of optimal extreme points of (4. 1) and (4. 3) coincide.

Proof. Let

$$\delta_0 = \min_{x \in L'} \sum_{j=1}^{p} x_j(1-x_j),$$

where L' denotes the set of those extreme points of L which have at least one non-integer component among their first p components.

Let $\mathbf{x}^0(\lambda) = (x_1^0(\lambda), \ldots, x_n^0(\lambda))^*$ be an arbitrary optimal extreme point of (4. 3). Then

$$F(\mathbf{z}) \leqq F(\mathbf{x}^0(\lambda)) - \lambda \sum_{j=1}^{p} x_j^0(\lambda)[1 - x_j^0(\lambda)], \tag{4.4}$$

where $\mathbf{z}$ is an optimal extreme point of (4. 1). If one of the first p components of $\mathbf{x}^0(\lambda)$ is not integer, then from (4. 4) it follows

$$\frac{1}{\lambda}[F(x^0(\lambda)) - F(z)] \geqq \sum_{j=1}^{p} x_j^0(\lambda)[1 - x_j^0(\lambda)] \geqq \delta_0 > 0. \tag{4.5}$$

Thus we see that if λ is sufficiently large, then $\mathbf{x}^0(\lambda)$ cannot have noninteger components among its first p entries. Consequently there is a λ_0 such that for $\lambda \geqq \lambda_0$ every optimal extreme point of (4. 3) is an optimal solution to (4. 1).

Conversely if $\mathbf{z}$ is an optimal extreme point of (4. 1), then $\mathbf{z}$ has to be optimal for (4. 3) because of (4. 5).

For practical computation we need an estimation for λ_0. Suppose that we are content with an "almost feasible" "ε optimal" solution of (4. 1). We call $\mathbf{y}$ "δ feasible" "ε optimal" ($\delta > 0$, $\varepsilon > 0$) solution of (4. 1) if $\mathbf{y}$ can violate the conditions

$$\sum_{j=1}^{n} a_{ij} x_j \leqq b_i \qquad (i = 1, \ldots m),$$

by no more than δ and $F(\mathbf{y}) \geqq F(\mathbf{z})-\varepsilon$ where $\mathbf{z}$ is an optimal solution of (4. 1). The following theorem provides an estimation for λ_0.

Theorem 5. Assume that

(i) for every $\mathbf{x}_1 \in T$, $\mathbf{x}_2 \in T$ where

$$T=\{\mathbf{x} | 0 \leqq x_j \leqq 1 \quad (j=1, \ldots, p), \quad 0 \leqq x_j \leqq k_j \quad (j=p+1, \ldots, n)\}$$

the inequality

$$|F(\mathbf{x}_1)-F(\mathbf{x}_2)| \leqq M|\mathbf{x}_1-\mathbf{x}_2|^a$$

holds where M and a are positive constants,

(ii) $K_a \leqq F(\mathbf{z}) \leqq K_f$ where $\mathbf{z}$ is any feasible solution of (4. 1),

(iii) $\mathbf{A}=[a_{ij}]$ has no zero rows and columns.

If λ satisfies the inequality

$$\lambda \geqq \frac{K_f-K_a}{\alpha_0(1-\alpha_0)} \tag{4.6}$$

then every vector obtained from an optimal extreme point of (4. 3) by rounding the first p components to the nearest integer is a "δ feasible" "ε optimal" solution of (4. 1) where

$$\alpha_0=\min(\hat{\alpha}, \bar{\alpha})$$

$$\hat{\alpha} \leqq \min\left\{\frac{1}{2}, \frac{1}{\sqrt{p}} \sqrt[a]{\frac{\varepsilon}{M}}\right\}; \quad \hat{\alpha}>0 \tag{4.7}$$

$$\bar{\alpha}<\min\left\{\min_{1 \leqq i \leqq m} \frac{\delta}{\sum_{j=1}^{p}|a_{ij}|}, \frac{1}{2}\right\}; \quad \bar{\alpha}>0. \tag{4.8}$$

Proof. Let $\mathbf{x}^0(\lambda)$ be an optimal extreme point of (4. 3) with λ satisfying (4. 6) and $\mathbf{z}$ an optimal extreme point of (4. 1). Then by (4. 5), (4. 6) and assumption (ii) we obtain

$$x_j^0(\lambda)[1-x_j^0(\lambda)] \leqq \sum_{j=1}^{p} x_j^0(\lambda)[1-x_j^0(\lambda)] \leqq \frac{1}{\lambda}\left[F(\mathbf{x}^0(\lambda))-F(\mathbf{z})\right] \leqq$$

$$\leqq \frac{1}{\lambda}(K_f-K_a) \leqq \alpha_0(1-\alpha_0) \leqq \bar{\alpha}(1-\bar{\alpha}) \qquad (j=1, \ldots, p). \tag{4.9}$$

This implies that one of the following inequalities holds

$$0 \leqq x_j^0(\lambda) \leqq \bar{\alpha}$$

$$1-\bar{\alpha} \leqq x_j^0(\lambda) \leqq 1 \qquad (j=1, \ldots, p). \tag{4.10}$$

Denote by $\bar{\mathbf{x}}(\lambda)$ the vector obtained from $\mathbf{x}^0(\lambda)$ by rounding the first p components to the nearest integer. By (4. 8) we get

$$\sum_{j=1}^{n} a_{ij}\bar{x}_j(\lambda) \leqq b_i+\bar{\alpha}\sum_{j=1}^{p}|a_{ij}| \leqq b_i+\delta.$$

This means that $\bar{\mathbf{x}}(\lambda)=(\bar{x}_1(\lambda), \ldots, \bar{x}_n(\lambda))$ is "δ feasible".

To prove the "ε optimality" we obtain by assumption (i), (4. 7) and (4. 10) the inequalities:

$$|F(\mathbf{z})-F(\bar{\mathbf{x}}(\lambda))| \leqq M|\mathbf{x}^0(\lambda)-\bar{\mathbf{x}}(\lambda)|^a \leqq M(\sqrt{p}\delta)^a \leqq M\left(\sqrt{p}\,\frac{1}{\sqrt{p}}\sqrt[a]{\frac{\varepsilon}{M}}\right)^a = \varepsilon$$

which means that $\bar{\mathbf{x}}(\lambda)$ is "ε optimal".

Corollaries

1. If every a_{ij} and b_i is integer and $p=n$, then by choosing $\delta<1$, $\bar{\mathbf{x}}(\lambda)$ is a feasible solution of (4. 1). Furthermore if $F(\mathbf{x})$ takes on integral values for every integer $\mathbf{x}$, then by choosing $\varepsilon<1$ we obtain an optimal solution of (4. 1).

2. If $\mathbf{x}^0(\lambda)$ is a "Δ optimal" solution of (4. 3), then by changing (4. 6) to

$$\lambda \geqq \frac{K_f-K_a+\Delta}{\alpha_0(1-\alpha_0)}$$

we get a "δ feasible", "$\Delta+\varepsilon$ optimal" solution $\bar{\mathbf{x}}(\lambda)$ by rounding $\mathbf{x}^0(\lambda)$.

In the pure integer case δ and $\Delta+\varepsilon$ have to be chosen smaller then 1 in order to get an optimal solution of (4. 1).

For the solution of (4. 3) we can apply the methods proposed in Section 2. and 3. If $F(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j$, then (4. 1) is the mixed zero-one integer linear programming problem. In this case we can increase the efficiency of our cutting plane method by adjoining to the constraint set the inequality

$$\sum_{j=1}^{n} c_j x_j \geqq F_k+\Delta,$$

where F_k is the largest objective function's value obtained up to Step k. In the pure case Δ can be chosen 1 provided all the c_j-s are integer.

5. Fixed charge problems with convex objective function

The following problem occurs very frequently in economic applications:

A production vector has to be found which satisfies a number of linear constraints and minimizes a cost function composed of individual cost functions having a fixed cost at $x_j=0$. For $x_j>0$ the cost function is concave.

In mathematical terms the problem to be solved is the following:

$$f(\mathbf{x}) = -\sum_{j=1}^{n} f_j(x_j) \to \max$$

subject to

$$0 \leqq x_j \leqq k_j \qquad (j=1, \ldots, n), \tag{5. 1}$$

$$\mathbf{x} \in L,$$

where L is a convex polyhedron and

$$f_j(x_j) = \begin{cases} 0 & \text{if} \quad x_j = 0, \\ g_j(x_j) & \text{if} \quad x_j > 0, \end{cases} \quad \lim_{x_j \to +0} g_j(x_j) = A_j \geqq 0 \qquad (j = 1, \ldots, n),$$

$g_j(x_j)$ is a concave monotone increasing function.

We can formulate (5. 1) as a mixed zero-one integer programming problem in the following manner: (For convenience we suppose that $A_j > 0$ $(j = 1, \ldots, p)$ and $A_j = 0$ $(j = p+1, \ldots, n)$.)

$$F(\mathbf{x}, \xi) = -\sum_{j=1}^{p} [A_j \xi_j + g_j(x_j)] + \sum_{j=p+1}^{n} g_j(x_j) \to \max$$

subject to

$$\begin{gathered} 0 \leqq x_j \leqq k_j \qquad (j = 1, \ldots, n), \\ \mathbf{x} \in L \\ x_j - k_j \xi_j \leqq 0 \\ 0 \leqq \xi_j \leqq 1, \quad \xi_j = \text{integer} \qquad (j = 1, \ldots, p). \end{gathered} \tag{5. 2}$$

Since (5. 2) is of type (4. 1) the method proposed in Section 4. can be used for solving (5. 2). From computational point of view it is not indifferent that (5. 2) has p new variables. In this section we give a method for solving (5. 1) without increasing the number of variables.

Without any loss of generality we may assume that $p = n$. The following theorem asserts the existence of a continuous equivalent to (5. 1).

Theorem 6. Let us consider the problem:

$$\bar{f}(\mathbf{x}, \mathbf{r}) \to \max$$

subject to

$$\begin{gathered} 0 \leqq x_j \leqq k_j \qquad (j = 1, \ldots, n), \\ \mathbf{x} \in L, \end{gathered} \tag{5. 3}$$

where $\mathbf{r}^* = (r_1, \ldots, r_n)$ $(\mathbf{r} \geqq \mathbf{O})$ is a parameter vector and

$$\bar{f}(\mathbf{x}, \mathbf{r}) = -\sum_{j=1}^{n} \bar{f}_j(x_j, r_j),$$

$$\bar{f}_j(x_j, r_j) = \begin{cases} m_j x_j & \text{if} \quad x_j \leqq r_j, \\ g_j(x_j) & \text{if} \quad x_j > r_j, \end{cases} \qquad (j = 1, \ldots, n),$$

$$m_j = \frac{g_j(r_j)}{r_j}.$$

There exists a positive vector $\mathbf{r}_0$ such that for all $\mathbf{r}$ $(0 < \mathbf{r} \leqq \mathbf{r}_0)$ the sets of optimal extreme points of (5. 1) and (5. 3) coincide.

Proof. Let $\mathbf{x}_0$ and $\mathbf{x}_1(\mathbf{r})$ be optimal solutions to (5. 1) and (5. 3) resp. Since both $f(\mathbf{x})$ and $\bar{f}(\mathbf{x}, \mathbf{r})$ are concave functions we may assume that $\mathbf{x}_0$ and $\mathbf{x}_1(\mathbf{r})$ are extreme points. Let $\mathbf{x}^*=(x_1, \ldots, x_n)$ and

$$s = \min_{\mathbf{x}\in L'} \left(\min_{1\leqq j\leqq n} x_j \right) \quad (x_j > 0),$$

where L' denotes the extreme points of the common feasible region of (5. 1) and (5. 3). (We can disregard of the trivial case if $\mathbf{O}\in L'$ since in this case $\mathbf{x}_1(\mathbf{r})=\mathbf{x}_0=\mathbf{O}$ for any positive $\mathbf{r}$ by the monotonicity of the functions $g_j(x_j)$.) Let $\mathbf{r}_0$ be a positive vector satisfying $\mathbf{r}_0^*\mathbf{e}_j \leqq s$ $(j=1, \ldots, n)$. Then

$$\bar{f}(\mathbf{x}_1(\mathbf{r}_0), \mathbf{r}_0) = f(\mathbf{x}_1(\mathbf{r}_0)).$$

Since $\bar{f}_j(\mathbf{x}_1^*(\mathbf{r}_0)\mathbf{e}_j, \mathbf{r}_0^*\mathbf{e}_j)=0$ if $\mathbf{x}_1^*(\mathbf{r}_0)\mathbf{e}_j=0$ and $\bar{f}_j(\mathbf{x}_1^*(\mathbf{r}_0)\mathbf{e}_j, \mathbf{r}_0^*\mathbf{e}_j)=g_j(\mathbf{x}_1^*(\mathbf{r}_0)\mathbf{e}_j)$ if $\mathbf{x}_1^*(\mathbf{r}_0)\mathbf{e}_j \geqq s \geqq \mathbf{r}_0^*\mathbf{e}_j$ $(j=1, \ldots, n)$. But by the optimality of $\mathbf{x}_1(\mathbf{r}_0)$ it follows

$$f(\mathbf{x}_1(\mathbf{r}_0)) = \bar{f}(\mathbf{x}_1(\mathbf{r}_0), \mathbf{r}_0) \geqq \bar{f}(\mathbf{x}_0, \mathbf{r}_0) = f(\mathbf{x}_0)$$

which means that $\mathbf{x}_1(\mathbf{r}_0)$ is optimal to (5. 1). Conversely by the optimality of $\mathbf{x}_0$

$$\bar{f}(\mathbf{x}_0, \mathbf{r}_0) = f(\mathbf{x}_0) \geqq f(\mathbf{x}_1(\mathbf{r}_0)) = \bar{f}(\mathbf{x}_1(\mathbf{r}_0), \mathbf{r}_0)$$

which means that $\mathbf{x}_0$ is optimal to (5. 3) if $\mathbf{r}\leqq\mathbf{r}_0$.

Corollary. The objective function of (5. 3) is convex on E^n and therefore the method of Section 2. can be used to solve it.

The only difficulty is that we cannot give an a priori estimation on $\mathbf{r}_0$. Fortunately by a slight modification of the algorithm described in Section 2. we do not need the exact value of $\mathbf{r}_0$. There are only two places where changes have to be done:

1. In (2. 4) $f(\bar{\mathbf{x}}_0-\mathbf{A}_1^{-1}\mathbf{y})$ is defined only for those values of $\mathbf{y}$ where

$$\bar{\mathbf{x}}_0-\mathbf{A}_1^{-1}\mathbf{y} \geqq \mathbf{0}. \tag{5.4}$$

2. In the definition of t_j ((2. 7)) (5. 4) has also to be taken into consideration. Thus t_j is the maximal number for which the inequalities

$$\left.\begin{aligned} f(\bar{\mathbf{x}}_0-t\mathbf{A}_1^{-1}\mathbf{e}_j) &\leqq \bar{C}_0 \\ \bar{\mathbf{x}}_0-t\mathbf{A}_1^{-1}\mathbf{e}_j &\geqq \mathbf{0} \end{aligned}\right\} \quad (j=1, \ldots, n) \tag{5.5}$$

hold.

All other statements of Section 2. including Theorem 1. remain valid. Naturally our method can be combined with other methods e.g. approximative methods like [14] since any good approximative solution can serve as a starting point for the cutting plane method. Of course the difficulties caused by degeneration can be overcome by searching for an "ε optimal" solution.

6. Separable nonlinear programming with linear constraints

Nonlinear programming with general objective function is a rather undiscovered field of mathematical programming. There are methods based on the idea of approximation with polygons, [1], [15], algorithms applying "branch and bound" [16], [17]

and full description methods [18]. We shall apply the cutting plane method of Section 2. for accelerating the full description method. We begin with the simple case of separable objective function and thereafter we discuss the general programming problem.

$$f(\mathbf{x}) = \sum_{j=1}^{n} f_j(x_j) \to \max$$

subject to

$$\mathbf{O} \leqq \mathbf{x} \leqq \mathbf{k} \qquad (\mathbf{k} \geqq \mathbf{O}) \tag{6.1}$$

$$\mathbf{A}\mathbf{x} \leqq \mathbf{b}.$$

Suppose that

(i) $L = \{\mathbf{x} | \mathbf{O} \leqq \mathbf{x} \leqq \mathbf{k},\ \mathbf{A}\mathbf{x} \leqq \mathbf{b}\} \neq \emptyset$,

(ii) for every x_j^1, x_j^2 satisfying $0 \leqq x_j^r \leqq \mathbf{k}^* \mathbf{e}_j$ $(r = 1, 2)$ holds the inequality:

$$|f_j(x_j^1) - f_j(x_j^2)| \leqq M_j |x_j^1 - x_j^2| \qquad (j = 1, \ldots, n), \tag{6.2}$$

where M_j is constant,

(iii) $f_j(x_j) \equiv -\infty$ for $x_j < 0$ and $x_j > \mathbf{k}^* \mathbf{e}_j$ $(j = 1, \ldots, n)$. Our purpose is to determine an "ε optimal" feasible solution $\bar{\mathbf{x}} \in L$.

The method for solving (6.1) consists of iterational steps. To start with let us determine an extreme point of $L = L_0$, say $\mathbf{x}_0 = (x_1^0, \ldots, x_n^0)^*$. Let us assume that we have a "good" approximative solution $\mathbf{y}_0 \in L$. ($\mathbf{y}_0$ can be e.g. a local maximum-point of (6.1) which can be obtained by several local methods such as gradient methods, linear approximation e.t.c.)

Put $K_0 = f(\mathbf{y}_0)$ and define $\bar{f}(\mathbf{x})$ in the following manner:

$$\bar{f}(\mathbf{x}) = \sum_{j=1}^{n} \bar{f}_j(x_j),$$

where $\bar{f}_j(x_j)$ is convex, $\bar{f}_j(x_j) \geqq f_j(x_j)$ for all x_j, $\bar{f}_j(x_j^0) = f_j(x_j^0)$ $(j = 1, \ldots, n)$.

Because of Property (ii) $\bar{f}(\mathbf{x})$ is defined over the entire E^n. Since $\mathbf{x}_0$ is a vertex of L transformation (2.5) can be carried out. Now consider the problem (see (2.4))

$$\bar{f}(\mathbf{x}_0 - A_1^{-1}\mathbf{y}) \to \max$$

subject to

$$\mathbf{y} \geqq \mathbf{O} \tag{6.3}$$

$$\mathbf{A}_3 \mathbf{y} \leqq \mathbf{b}_3.$$

By the definition of $\bar{f}(\mathbf{x})$

$$\bar{f}(\mathbf{x}_0) = f(\mathbf{x}_0).$$

Let t_j be the maximal number (but at most M, a large fixed positive number) satisfying

$$\bar{f}(\mathbf{x}_0 - t\mathbf{A}_1^{-1}\mathbf{e}_j) \leqq K_0 + \varepsilon \qquad (j = 1, \ldots, n). \tag{6.4}$$

Each t_j is positive since $\bar{f}(\mathbf{x}_0) \leqq K_0$ and $\bar{f}(\mathbf{x})$ is continuous. (Since it is convex over E^n.) Let

$$\mathbf{t}^* = (1/t_1, \ldots, 1/t_n).$$

Take a fixed positive number T and apply the cut

$$\mathbf{t}^*\mathbf{y} \geqq 1 \tag{6.5}$$

if $|\mathbf{t}^*A_1| \leqq T$. With cut (6.5) we have excluded a region where

$$\bar{f}(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \leqq K_0 + \varepsilon$$

and since $\bar{f}(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \geqq f(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y})$ the relation

$$f(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \leqq K_0 + \varepsilon$$

holds for every $\mathbf{y}$ in the excluded region.

If $|\mathbf{t}^*\mathbf{A}_1| > T$, then we apply the cut

$$\bar{\mathbf{t}}^*\mathbf{y} \geqq 1, \tag{6.6}$$

where

$$\bar{\mathbf{t}}^* = (1/\alpha t_1, \ldots, 1/\alpha t_n)$$

and α is chosen so that $|\bar{\mathbf{t}}^*\mathbf{A}_1| = T$ is satisfied.

Of course in this case we can only guarantee that for all $\mathbf{y}$ in the excluded region

$$f(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \leqq \bar{f}(\mathbf{x}_0 - \mathbf{A}_1^{-1}\mathbf{y}) \leqq \max_{1 \leqq j \leqq n} \bar{f}(\mathbf{x}_0 - \alpha t_j \mathbf{A}_1^{-1}\mathbf{e}_j) = P_0.$$

The whole procedure is repeated for the reduced polyhedron L_1. We have seen in Section 2. (Theorem 1.) that after finite number of steps $L_p = \emptyset$ for some $p \geqq 1$.

Naturally if in the course of computations we arrive at a vector which gives higher objective function's value than K_0, then starting from this point we can find a better local maximumpoint with objective function's value $K_1 > K_0$ and replace K_0 by K_1.

After having arrived at the situation where $L_p = \emptyset$ the best solution $\mathbf{y}_r$ obtained so far satisfies the inequality

$$f(\mathbf{y}_r) \leqq \max_{0 \leqq k \leqq p-1} R_k,$$

where $R_k = K_k + \varepsilon$ if in Step k cut (6.5) was used and $R_k = P_k$ if cut (6.6) was applied. Thus if

$$\max_{0 \leqq k \leqq p-1} R_k = K_{k'} + \varepsilon \quad \text{for some} \quad 0 \leqq k' \leqq p-1 \tag{6.7}$$

then $\mathbf{y}_r$ is an "ε optimal" solution of (6.1) if

$$f(\mathbf{y}_r) = K_k.$$

Let $Q = \{q_1, \ldots, q_r\}$ be the set of indices for which

$$P_{q_s} > \max_{0 \leqq k \leqq p-1} K_k + \varepsilon \qquad (s = 1, \ldots, r).$$

For each q_s there can be associated a problem:

$$f(\mathbf{x}_{q_s} - B_{q_s}^{-1}\mathbf{y}) \to \max$$

subject to

$$\mathbf{y} \geqq \mathbf{O}$$

$$\mathbf{A}_{q_s}\mathbf{y} \leqq \mathbf{b}_{q_s} \qquad (s=1, \ldots, r), \tag{6.8}$$

$$\bar{\mathbf{t}}^*_{q_s}\mathbf{y} \leqq 1,$$

where $\mathbf{A}_{q_s}\mathbf{y} \leqq \mathbf{b}_{q_s}$, $\mathbf{y} \geqq \mathbf{O}$ defines L_{q_s-1} after having done transformation (2.5), $\mathbf{B}_{q_s}$ is the matrix of transformation, $\bar{\mathbf{t}}_{q_s}$ is defined in (6.6), $\mathbf{x}_{q_s}$ is the actual extreme point of L_{q_s-1}.

We shall decompose (6.8) into d subproblems having the form:

$$f(\mathbf{x}_{q_s} - \mathbf{B}^{-1}_{q_s}\mathbf{y}) \to \max$$

subject to

$$\mathbf{y} \geqq \mathbf{O}$$

$$\mathbf{A}_{q_s}\mathbf{y} \leqq \mathbf{b}_{q_s} \qquad (l=1, \ldots, d). \tag{6.9}$$

$$\bar{\mathbf{t}}^*_{q_s}\mathbf{y} = \frac{l}{d}$$

Let $\mathbf{s} \neq \mathbf{O}$ be an arbitrary feasible point of (6.8) and $\mathbf{v}$ the intersection of the ray determined by $\mathbf{O}$ and $\mathbf{s}$ with the hyperplane $\bar{\mathbf{t}}^*_{q_s}\mathbf{y}=1$. Let further l be the index for which the inequality

$$\frac{l}{d} \leqq \bar{\mathbf{t}}^*_{q_s}\mathbf{s} \leqq \frac{l+1}{d}$$

holds. Denote by $\mathbf{r}$ the intersection of the ray $(\mathbf{O}, \mathbf{s})$ with hyperplane $\bar{\mathbf{t}}_{q_s}\mathbf{y} = \frac{l}{d}$. Since $\mathbf{r}$ and $\mathbf{s}$ are on the ray $(\mathbf{O}, \mathbf{v})$ they can be written in the following way:

$$\mathbf{r} = \lambda_r \mathbf{v},$$

$$\mathbf{s} = \lambda_s \mathbf{v},$$

where $\lambda_r = \frac{l}{d}$. Then the following relations hold:

$$|\mathbf{r}-\mathbf{s}| = |\lambda_r - \lambda_s|\,|\mathbf{v}| = \left|\frac{l}{d} - \lambda_s\right| |\mathbf{v}| = \left|\frac{l}{d} - \bar{\mathbf{t}}^*_{q_s}\mathbf{s}\right| |\mathbf{v}| \leqq \left|\frac{l}{d} - \frac{l+1}{d}\right| |\mathbf{v}| = \frac{1}{d}|\mathbf{v}|.$$

Since

$$|\mathbf{v}| \leqq \max_{1 \leqq j \leqq n} \bar{\mathbf{t}}^*_{q_s}\mathbf{e}_j \leqq M,$$

d can be chosen so large that $|\mathbf{r}-\mathbf{s}| \leqq \delta$ for given $\delta > 0$. But because of property (ii) if δ is small enough, then

$$|f(\mathbf{r}) - f(\mathbf{s})| \leqq \frac{\varepsilon}{2}.$$

This means that if we can solve problem (6.9) for each l, then the objective function's value of an "$\varepsilon/2$ optimal" solution of problem (6.9) cannot differ from the optimum of (6.8) by more than ε. But the feasible set of (6.9) is of lower dimension than that of (6.8).

For solving subproblems (6. 9) we can apply the same procedure as for (6. 1). It is clear that after finite number of steps either situation (6. 7) occurs or the dimension of the subproblems reduces to zero. In both cases we obtain at least one "ε optimal" solution of (6. 1).

Of course the right hand side of inequality (6. 4) may increase by discovering new better solutions and those subproblems of type (6. 8) where P_{q_s} does not exceed the best objective function's value obtained so far can be dropped.

To illustrate the method let us take a numerical example:

$$f(x_1, x_2) = -(x_1-1)^3+x_2-1 \to \max.$$

Subject to

$$\begin{aligned} 0 \leqq x_1 \leqq 2 \\ 0 \leqq x_2 \leqq 2 \\ -15x_1+10x_2 \leqq 2 \\ -3x_1+4x_2 \leqq 2. \end{aligned} \tag{6. 10}$$

First of all determine a local maximumpoint. For this purpose we can use e.g. the method of Zangwill [19]. If we start from $x_1=2$, $x_2=2$, then this method leads us to the local maximumpoint $x_1 = \frac{3}{2}$, $x_2 = \frac{13}{8}$ where the objective function's value $f\left(\frac{3}{2}, \frac{13}{8}\right) = K_1 = \frac{1}{2}$.

According to the method proposed in this section we have to start with an arbitrary extreme point. Let this be $x_1=2$, $x_2=2$. The construction of the functions $\bar{f}_1^{(1)}(x_1)$ and $\bar{f}_2^{(1)}(x_2)$ is an elementary task. (The upper index denotes the number of iterations.)

$$\bar{f}_1^{(1)}(x_1) = -3x_1+5,$$

$$\bar{f}_2^{(1)}(x_2) = x_2-1.$$

The matrix of the transformation and its inverse is the following:

$$\mathbf{A}_1 = \begin{bmatrix} -3 & 4 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{A}_1^{-1} = \begin{bmatrix} 0 & 1 \\ 1/4 & 3/4 \end{bmatrix}.$$

We have to find the maximal positive solutions to the inequalities: (The admissable error $\varepsilon=0,1$)

$$\bar{f}^{(1)}\left(\begin{bmatrix} 2 \\ 2 \end{bmatrix} - t\begin{bmatrix} 0 \\ 1/4 \end{bmatrix}\right) = \bar{f}_1^{(1)}(2)+\bar{f}_2^{(1)}\left(2-\frac{1}{4}t\right) \leqq \frac{1}{2}+\frac{1}{10} = \frac{3}{5},$$

$$\bar{f}^{(1)}\left(\begin{bmatrix} 2 \\ 2 \end{bmatrix} - t\begin{bmatrix} 1 \\ 3/4 \end{bmatrix}\right) = \bar{f}_1^{(1)}(2-t)+\bar{f}_2^{(1)}\left(2-\frac{3}{4}t\right) \leqq \frac{3}{5}.$$

The solutions are $t_1^{(1)}=\infty$, $t_2^{(1)}=4/15$. Thus the cutting inequality obtained in the first step is

$$[0,\ 15/4]\begin{bmatrix} -3 & 4 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leqq [0,\ 15/4]\begin{bmatrix} 2 \\ 2 \end{bmatrix} - 1$$

or briefly

$$x_1 \leqq \frac{26}{15}.$$

In the second iterational step let our starting extreme point be $x_1 = \frac{26}{15}$, $x_2 = \frac{9}{5}$. Since $f\left(\frac{26}{15}, \frac{9}{5}\right) < \frac{1}{2}$, $K_2 = K_1 = \frac{1}{2}$. The matrix of transformation remains unchanged but $\bar{f}^{(2)}(\mathbf{x})$ will be different from $\bar{f}^{(1)}(\mathbf{x})$. By simple computation we get

$$\bar{f}_1^{(2)}(x_1) = -\frac{121}{75}x_1 + \frac{8107}{3375},$$

$$\bar{f}_2^{(2)}(x_2) = x_2 - 1.$$

Consider the inequalities

$$\bar{f}^{(2)}\left(\begin{bmatrix} \frac{26}{15} \\ \frac{9}{5} \end{bmatrix} - t \begin{bmatrix} 0 \\ \frac{1}{4} \end{bmatrix}\right) = \bar{f}_1^{(2)}\left(\frac{26}{15}\right) + \bar{f}_2^{(2)}\left(\frac{9}{5} - \frac{1}{4}t\right) \leqq \frac{3}{5},$$

$$\bar{f}^{(2)}\left(\begin{bmatrix} \frac{26}{15} \\ \frac{9}{5} \end{bmatrix} - t \begin{bmatrix} 1 \\ \frac{3}{4} \end{bmatrix}\right) = \bar{f}_1^{(2)}\left(\frac{26}{15} - t\right) + \bar{f}_2^{(2)}\left(\frac{9}{5} - \frac{3}{4}t\right) \leqq \frac{3}{5}.$$

The maximal positive solutions $t_1^{(2)} = \infty$, $t_2^{(2)} = \frac{196\,800}{344\,729}$. The cutting inequality

$$x_1 \leqq \frac{482\,658}{344\,729} < \frac{3}{2}.$$

To make the calculations simple we take the less sharp cut

$$x_1 \leqq \frac{3}{2}.$$

Our starting solution in the third iterational step is: $x_1 = \frac{3}{2}$, $x_2 = \frac{13}{8}$. The matrix of transformation is also unchanged and $K_3 = K_2 = K_1$.

$$\bar{f}_1^{(3)}(x_1) = -\frac{3}{3}x_1 + 1,$$

$$\bar{f}_2^{(3)}(x_2) = x_2 - 1.$$

Consider the inequalities:

$$\bar{f}^{(3)}\left(\begin{bmatrix}\frac{3}{2}\\ \frac{13}{8}\end{bmatrix}-t\begin{bmatrix}0\\ \frac{1}{4}\end{bmatrix}\right)=\bar{f}_1^{(3)}\left(\frac{3}{2}\right)+\bar{f}_2^{(3)}\left(\frac{13}{8}-\frac{1}{4}t\right)\leqq\frac{3}{5},$$

$$\bar{f}^{(3)}\left(\begin{bmatrix}\frac{3}{2}\\ \frac{13}{8}\end{bmatrix}-t\begin{bmatrix}1\\ \frac{3}{4}\end{bmatrix}\right)=\bar{f}_1^{(3)}\left(\frac{3}{2}-t\right)+\bar{f}_2^{(3)}\left(\frac{13}{8}-\frac{3}{4}t\right)\leqq\frac{3}{5}.$$

The maximal positive solutions are: $t_1^{(3)}=\infty$, $t_2^{(3)}=\infty$. Thus $L_3=\emptyset$ which means that $x_1=\frac{3}{2}$, $x_2=\frac{13}{8}$ is an "0,1 optimal" solution of (6. 10).
(Throughout the calculations we have assumed M and T very large.)

7. The solution of general continuous nonlinear programming problems

As a first step of generalization let us drop the separability stipulation for $f(\mathbf{x})$. That is we consider the problem

$$f(\mathbf{x})\to\max$$

subject to

$$\mathbf{Ax}\leqq\mathbf{b}, \tag{7. 1}$$

where

(i) $L=\{\mathbf{x}|\mathbf{Ax}\leqq\mathbf{b}\}$ is nonvoid and bounded,

(ii) for every closed, bounded, convex set $C\subset E^n$ there exists a constant M_c such that for all $\mathbf{x}_1, \mathbf{x}_2\in C$

$$|f(\mathbf{x}_1)-f(\mathbf{x}_2)|\leqq M_c|\mathbf{x}_1-\mathbf{x}_2|. \tag{7. 2}$$

The method proposed to solve (7. 1) is very similar to the method of Section 6. Since we have used the separability of the objective function in the construction of $\bar{f}(\mathbf{x})$ we define $\bar{f}(\mathbf{x})$ for (7. 1) in an other way. Let

$$\bar{f}_c(\mathbf{x})=M_c|\mathbf{x}-\mathbf{x}_0|+f(\mathbf{x}_0), \tag{7. 3}$$

where $\mathbf{x}_0$ is the starting extreme point, M_c is the constant belonging to a closed, bounded, convex set C (see (7. 2)). It is easy to prove that $\bar{f}_c(\mathbf{x})$ is convex and if $f(\mathbf{x})$ is continuously differentiable on E^n, then

$$M_c\geqq\max_{\mathbf{x}\in C}|f'(\mathbf{x})|, \tag{7. 4}$$

where $f'(\mathbf{x})$ is the gradient vector of $f(\mathbf{x})$.

Since in the definition of $\bar{f}_c(\mathbf{x})$ the set C is involved we have to modify the procedure of determining t_j. In this case t_j is the maximal number for which the

relations

$$\begin{gathered} \bar{f}_c(\mathbf{x}_0 - t\mathbf{A}_1^{-1}\mathbf{e}_j) \leqq K_0 + \varepsilon \\ \mathbf{x}_0 - t\mathbf{A}_1^{-1}\mathbf{e}_j \in C \end{gathered} \qquad (j=1, \ldots, n) \tag{7.5}$$

hold.

All other steps of the method of Section 6. remain unchanged.

It is clear that the efficiency of a particular cut depends greatly on the choice of C. Theoretically we ought to choose C to minimize $|\mathbf{t}^*\mathbf{A}_1|$. But this is a very difficult problem solving. Instead of solving this problem we propose choosing a region depending on one parameter (e.g. a ball with radius λ, a cube with edge-length λ etc.) and to solve the one variable minimization problem.

Now we are able to treat the general continuous nonlinear programming problem:

$$f(\mathbf{x}) \to \max$$

subject to

$$\begin{gathered} \mathbf{x} \in L, \\ g_k(\mathbf{x}) \leqq 0 \qquad (k=1, \ldots, p), \end{gathered} \tag{7.6}$$

where L is a bounded convex polyhedral set and the functions $f(\mathbf{x}), g_1(\mathbf{x}), \ldots, g_p(\mathbf{x})$ are continuously differentiable over E^n.

By using the idea of Fiacco and McCormick [20] we reduce (7. 6) to (7. 1) and then we apply the method of cutting planes.

Problem (7. 6) can always be transformed into the following problem:

$$-\exp z \to \max$$

subject to

$$\begin{gathered} \mathbf{y} \in S, \\ h_k(\mathbf{y}) = 0 \qquad (k=1, \ldots, p), \\ \varphi(\mathbf{y}) - z = 0, \end{gathered} \tag{7.7}$$

where S is a convex polyhedron.

We shall search for a "(δ, ϱ) solution" ($\delta > 0$, $\varrho > 0$) of (7. 7). A point $(\mathbf{y}_0, z_0)$ is called "(δ, ϱ) solution" of (7. 7) if

$$\begin{gathered} \mathbf{y}_0 \in S, \\ |h_k(\mathbf{y}_0)| \leqq \delta \qquad (k=1, \ldots, p), \\ z_0 \geqq \bar{z} - \varrho, \end{gathered} \tag{7.8}$$

where $\bar{z}$ is optimal to (7. 7).

Consider the following problem:

$$F(\mathbf{y}, z, a_t) = -\exp z - a_t \left[\sum_{k=1}^{p} h_k^2(\mathbf{y}) + (\varphi(\mathbf{y}) - z)^2 \right] \to \max \tag{7.9}$$

subject to

$$\mathbf{y} \in S,$$

where a_t is a positive parameter.

(7. 9) can always be solved since S is bounded. Let $(\mathbf{y}_t, z_t)$ be an "ε_t optimal" solution of (7. 9) ($\varepsilon_t > 0$).

Theorem 7. If $\lim_{t\to\infty} a_t = \infty$ and $\lim_{t\to\infty} \varepsilon_t = 0$, then every cluster point of the sequence $\{(\mathbf{y}_t, z_t)\}$ is an optimal solution of (7. 7).

Proof. Let $(\bar{\mathbf{y}}, \bar{z})$ be an optimal solution of (7. 7). By the definition of $(\mathbf{y}_t, z_t)$ the following inequalities hold

$$-\exp z_t - a_t\left[\sum_{k=1}^{p} h_k^2(\mathbf{y}_t) + (\varphi(\mathbf{y}_t) - z_t)^2\right] \geqq$$

$$\geqq -\exp \bar{z} - a_t\left[\sum_{k=1}^{p} h_k^2(\bar{\mathbf{y}}) + (\varphi(\bar{\mathbf{y}}) - \bar{z})^2\right] - \varepsilon_t = -\exp \bar{z} - \varepsilon_t.$$

Hence

$$0 \leqq h_k^2(y_t) \leqq \frac{1}{a_t}[-\exp z_t + \exp \bar{z} + \varepsilon_t] \leqq \frac{1}{a_t}[\exp \bar{z} + \varepsilon_t] \qquad (k = 1, \ldots, p). \quad (7.10)$$

From (7. 10) we get for any cluster point $(\hat{\mathbf{y}}, \hat{z})$

$$h_k(\hat{\mathbf{y}}) = 0 \qquad (k = 1, \ldots, p),$$

which means that $\hat{\mathbf{y}}$ is feasible. By the same reasoning we obtain

$$h_k(\hat{\mathbf{y}}) - \hat{z} = 0.$$

Also from (7. 10)

$$\exp z_t \leqq \exp \bar{z} + \varepsilon_t$$

which means that if $\varepsilon_t \to 0$, then $z_t \to \bar{z}$.

Corollary. Let us assume that we know lower and upper bounds for $\bar{z}$.

$$N \leqq \bar{z} \leqq M. \quad (7.11)$$

Then from (7. 10)

$$h_k^2(\mathbf{y}_t) \leqq \frac{1}{a_t}(\exp \bar{z} + \varepsilon_t) \leqq \frac{1}{a_t}(\exp M + \varepsilon_0) \quad (7.12)$$

$$(\varepsilon_0 \geqq \varepsilon_t;\ t = 1, 2, \ldots) \quad (k = 1, \ldots, p),$$

$$|h_k(\mathbf{y}_t)| \leqq \sqrt{\frac{\exp M + \varepsilon_0}{a_t}}.$$

If we want $|h_k(\mathbf{y}_t)| \leqq \delta$ to hold, then a_t has to be chosen to satisfy

$$a_t \geqq \frac{\exp M + \varepsilon_0}{\delta^2}. \quad (7.13)$$

Furthermore from (7. 10)

$$\exp z_t - \exp \bar{z} \leqq \varepsilon_0$$

$$\exp\{(z_t - \bar{z}) + \bar{z}\} - \exp \bar{z} \leqq \varepsilon_0$$

$$\exp \bar{z}[\exp(z_t - \bar{z}) - 1] \leqq \varepsilon_0$$

and using inequality (7. 11) we obtain the estimation

$$z_t - \bar{z} \leqq \log\left(\frac{\varepsilon_0}{\exp \bar{z}} + 1\right) \leqq \log\left(\frac{\varepsilon_0}{\exp N} + 1\right).$$

If we want $z_t - \bar{z} \leqq \varrho$ to hold, then we have to choose ε_0 to satisfy

$$\varepsilon_0 \leqq (\exp \varrho - 1) \exp N. \tag{7. 14}$$

Summing up. If we find an "ε_0 optimal" solution to (7. 9) where ε_0 satisfies (7. 14) and a_t satisfies (7. 13), then this solution is a "(δ, ϱ) solution" of (7. 7).

To solve (7. 7) we can apply the cutting plane method described in this section.

8. General pure integer programming

Let us consider the problem

$$f(\mathbf{x}) \to \max$$

subject to

$$\mathbf{x} \in L, \tag{8. 1}$$

$$\mathbf{x} = \text{integer},$$

where L is a polyhedron and $f(\mathbf{x})$ satisfies Property (ii) in Section 7.

The method of Section 6. and 7. can be modified to be able to solve (8. 1) too. The main steps of the procedure are as follows:

Step 0. Find a feasible point (if there is any) of (8. 1) with an integer programming algorithm.

Step k. Take an extreme point $\mathbf{x}_k$ of L_k ($L = L_1$). Denote by K_k the maximal objective function's value obtained so far on integer points of L. Let $\mathbf{A}_1$ be the matrix of transformation and T a fixed positive number.

Case 1. $\mathbf{x}_k$ is noninteger, $f(\mathbf{x}_k) \leqq K_k$.

If $|\mathbf{t}^* \mathbf{A}_1| \leqq T$, then apply cut (6. 5).

If $|\mathbf{t}^* \mathbf{A}_1| > T$, then make a Gomory cut or construct subproblems (6. 9).

Case 2. $\mathbf{x}_k$ is noninteger, $f(\mathbf{x}_k) > K_k$.

Make a Gomory cut.

Case 3. $\mathbf{x}_k$ is integer.

Apply cut (6. 5).

It can easily be proved along the lines of the proof of Theorem 1, Theorem 2 and Section 6. that these procedures converge in finite number of steps to an "ε optimal" solution of (8. 1).

9. Computational considerations

For the various algorithms contained in the previous sections concrete computational experiences are available only for application of the cutting plane method to the pure zero-one integer linear programming. Detailed description of test problems and results will be reported elsewhere. However we can mention in advance that finding the optimal solution needs much less computational effort than verifying the optimality. We think that an optimal solution of zero-one integer linear programming problems up to 120 variables can be obtained by the cutting plane method within acceptable time interval with the best computers available in Hungary. It may happen however that we cannot make sure that this is the optimal solution.

There are special problems where existence theorems assure that there is at least one integer feasible solution and every feasible point is a solution of the problem (e.g. finding an equilibrium point of a bimatrix game [21]). In these cases the cutting plane method seems to be able to solve the problem completely.

DEPT. OF MATEMATICS
KARL MARX UNIVERSITY OF ECONOMICS
BUDAPEST, HUNGARY

References

[1] HADLEY, G., *Nonlinear and dynamic programming,* Addison Wesley, London, 1964.
[2] MARTOS, B., Quasi-convexity and quasi-monotonicity in nonlinear programming, *Studia Sci. Math. Hungar.*, v. 2, 1967, pp. 265—273.
[3] KELLEY, J. E. JR., The cutting plane method for solving convex programs, *SIAM J. Appl. Math.*, v. 8, 1960, pp. 703—712.
[4] GOMORY, R. E., Outline of an algorithm for integer solutions to linear programs, *Bull. Amer. Math. Soc.*, v. 64, 1958, pp. 275—278.
[5] HOANG TUI., Concave programming under linear constraints, *Soviet Math. Dokl.*, v. 5, 1964, pp. 1437—1440.
[6] RITTER, K., A method for solving maximum problems with nonconcave quadratic function, *Z. Wahrscheinlichkeitstheorie und Verw. Gebiete,* v. 4, 1966, pp. 340—351.
[7] FORGÓ, F., A method for maximizing a convex function subject to linear constraints (in Hungarian), *Döntési modellek II.* Közgazdasági és Jogi Könyvkiadó, Budapest, 1969, pp. 211—236.
[8] CHARNES, A., W. COOPER, A. HENDERSON, *An introduction to linear programming,* John Wiley and Sons, New York, 1953.
[9] BALINSKI, M. L., An algorithm for finding all vertices of convex polyhedral sets, *SIAM J. Appl. Math.*, v. 9, 1961, pp. 72—88.
[10] BEALE, E. M. L., Survey of integer programming, *Operational Research Quarterly,* v. 16, 1965, pp. 219—228.
[11] HADLEY, G., *Linear programming,* Addison Wesley, Reading, Mass. 1961.
[12] FORGÓ, F., Relationship between mixed zero-one integer linear programming and certain quadratic programming problems, *Studia Sci. Math. Hungar.*, v. 4, 1969, pp. 37—43.
[13] RAGHAVACHARI, M., On connections between zero-one integer programming and concave programming under linear constraints, *Operations Res.*, v. 17, 1969, pp. 680—684.
[14] BALINSKI, M. L., Fixed cost transportation problems, *Naval Res. Logist. Quart.*, v. 8, 1961, pp. 41—54.
[15] BEALE, E. M. L., *Numerical methods, Nonlinear Programming,* North Holland Publ. Comp. Amsterdam, 1967, pp. 135—205.
[16] LAWLER, E. L. & D. E. WOOD, Branch and bound methods, a survey, *Operations Res.*, v. 14, 1966, pp. 699—719.

[17] FALK, J. E. & R. M. SOLAND, *An algorithm for separable nonconvex programming*, Internal Report, Research Analysis Corporation, McLean, Virginia, 1968.
[18] MANAS, M., An algorithm for a nonconvex programming problem, *Ekonom. Mat. Obzor*, v. 2, 1966, pp. 202—211.
[19] ZANGWILL, W., The convex simplex method, *Management Sci.*, v. 14, 1968, pp. 221—238.
[20] FIACCO, A. V. & G. P. MCCORMICK, The slacked unconstrained minimization technique for convex programming, *SIAM J. Appl. Math.*, v. 15, 1967, pp. 505—515.
[21] MILLS, H., Equilibrium points in finite games, *SIAM J. Appl. Math.*, v. 8, 1960, pp. 397—402.
[22] FORGÓ, F., *Cutting plane methods for solving nonconvex programming problems*, Technical Rreport, Department of Mathematics, Karl Marx University of Economics, Budapest, March 1970.

(Received June 25, 1970)

A method for chronological ordering of archeological sites

By ANNAMÁRIA G. VARGA

1. Introduction

The chronological ordering of archeological material is an important question of the archeological investigation. For the solution of this problem, besides the classical archeological methods, various methods using tools of natural sciences and mathematics are known.

In this paper we are going to describe a mathematical method based on the theory of regression. This theory gives a natural approach to the problem of chronological ordering. By the aid of this theory we are able to decide in which cases the chronological order obtained by the method of Brainerd-Robinson [1] and by similar methods can be accepted. The idea of the application of the theory of regression was given by an analysis of the methods of Brainerd-Robinson and Dempsey-Baumhoff [2].

2. Prerequisites

The purpose of this section is to summarize concepts and to state results which are familiar to mathematicians but not to archeologists and which will be used in what follows. Whenever the word 'set' is used it will be interpreted to mean a subset of a given set which will be denoted by S. If x is an element of S, and E is a subset of S, the notation $x \in E$ means, that x belongs to E; the negation of this assertion, i.e. the statement that x does not belong to E, will be denoted by $x \notin E$. If E and F are subsets of S, the notation $E \subset F$ means that E is a subset of F i.e. that every point of E belongs to F. Two sets E and F are called equal if and only if they contain exactly the same elements or, equivalently, if and only if $E \subset F$ and $F \subset E$.

If $P(x)$ is a proposition concerning x then the symbol $\{x\colon P(x)\}$ denotes the set of those elements x for which the proposition $P(x)$ is true. In general the brace notation $\{...\}$ will be reserved for the formation of sets. Thus for instance if x and y are elements then $\{x, y\}$ denotes the set whose only elements are x and y.

If $\mathbf{E}$ is any set of subsets of S, the set of all points of S which belong to *at least one* set of $\mathbf{E}$ is called the *union* of the sets of $\mathbf{E}$; it will be denoted by $\cup\mathbf{E}$ or $\cup\{E\colon E \in \mathbf{E}\}$. For the union of a special set of sets various special notations are used. If for instance $\mathbf{E} = \{E_1, E_2, \ldots, E_n\}$, then $\cup\mathbf{E}$ is denoted also by $E_1 \cup E_2 \cup \ldots \cup E_n$ or $\bigcup_{i=1}^{n} E_i$.

If $\mathbf{E}$ any set of subsets of S, the set of all elements of S which belong to *every set* of $\mathbf{E}$ is called the *intersection* of the sets of $\mathbf{E}$; it will be denoted by $\cap\mathbf{E}$ or $\cap\{E\colon E\in\mathbf{E}\}$.

Two sets E and F are called *disjoint* if they have no elements in common. A disjoint set is a set $\mathbf{E}$ of sets such that every two distinct sets of $\mathbf{E}$ are disjoint.

If E and F are subsets of S, the *difference* between E and F, denoted by $E-F$, is the set of all elements of E which do not belong to F. The symmetric difference of two sets E and F, denoted by, $E\triangle F$ is defined by $E\triangle F=(E-F)\cup(F-E)$. It is the set of all elements which belong to one and only one of E and F.

Let R be any set whose elements are called, for suggestivity, points. If to each pair x, y of elements of R a non-negative real number, denoted by $\varrho(x, y)$ and called the distance of x and y, is attached such that

(1) if $x=y$ then $\varrho(x, y)=0$,
(2) if $\varrho(x, y)=0$ then $x=y$,
(3) $\varrho(x, y)=\varrho(y, x)$,
(4) for each three elements x, y, z of R

$$\varrho(x, y)\leqq\varrho(x, z)+\varrho(z, y),$$

the resulting "space" M is called a *metric space* over the groundset R with *metric* ϱ.

A function ϱ which satisfies (1), (3), (4) only, is called a *pseudo-metric* and the resulting space is called a *pseudo-metric space* M over the groundset R with pseudo-metric ϱ.

Let M be a pseudo-metric space and let D be the family of all sets $G_x=\{y\in M\colon \varrho(x, y)=0\}$. If $u\in G_x$ and $v\in G_y$ then

$$\varrho(u, v)\leqq\varrho(u, x)+\varrho(x, y)+\varrho(y, v)=\varrho(x, y).$$

Consequently, since in this case it is also true that $x\in G_u$ and $y\in G_v$, $\varrho(u, v)=\varrho(x, y)$. Let A and B be two members of D and let $\tau(A, B)$ be equal to $\varrho(x, y)$ for every x in A and for every y in B. Thus D with the function $\tau(A, B)$ is a metric space. In the sequel we shall call the set D with $\tau(A, B)$ the metric space induced by the pseudo-metric space M. A set N is called a subset of a metric space M provided N is a subset of the groundset R of M and the distance of any two points x, y of N is the same as their distance in M. If N and L are subsets of two metric space M and Q, respectively, we say N is congruent to L provided there exists a one-to-one distance-preserving correspondence between the points of N and the points of L; that is for every pair x, y of points of N $\varrho(x, y)=\varrho'(x', y')$, where x', y' are the points of L that correspond, respectively, to points x, y of N and ϱ, ϱ' denote the distance in N and L, respectively.

A subset N of a metric space M is congruently imbeddable in a metric space Q provided there is a subset L of Q such that N is congruent to L.

We shall apply in the sequel the theory of regression. We need the linear regression. For our purposes it is necessary to know only the following. We consider n points (x_1, y_1), (x_2, y_2), ..., (x_n, y_n) in the plane. It is convenient to write the equation of the straight line which we fit to these n points in the from

$$y'=a+b(x-\bar{x}), \tag{1}$$

where $\bar{x}$ is the arithmetic mean of $x_1, x_2, \ldots, x_n$; b is the slope of this line and a is the y intercept on the line $x=\bar{x}$. The y intercept on the y axis is $a-b\bar{x}$. The problem is to determine the parameters a and b so that the sum of the squares

$$\sum_{i=1}^{n}(y_i-y_i')^2$$

will be a minimum. When y' is replaced by its value as given by (1), it becomes clear that this sum is a function of a and b only. If this function is denoted by $F(a, b)$ then

$$F(a,b)=\sum_{i=1}^{n}[y_i-a-b(x_i-\bar{x})]^2.$$

If this function is to have a minimum value, it is necessary that its partial derivates vanish there; hence, a and b must satisfy the equations

$$\frac{\partial F}{\partial a}=\sum_{i=1}^{n}2[y_i-a-b(x_i-\bar{x})][-1]=0,$$

$$\frac{\partial F}{\partial b}=\sum_{i=1}^{n}2[y_i-a-b(x_i-\bar{x})][-x_i-\bar{x}]=0.$$

When the summations are performed term by term and the sums that involve y_i are transposed, these equations assume the form

$$an+b\sum_{i=1}^{n}(x_i-\bar{x})=\sum_{i=1}^{n}y_i$$

$$a\sum_{i=1}^{n}(x_i-\bar{x})+b\sum_{i=1}^{n}(x_i-\bar{x})^2=\sum_{i=1}^{n}(x_i-\bar{x})y_i.$$

Since $\sum_{i=1}^{n}(x_i-\bar{x})=0$, the solution of these equations is given by

$$a=\bar{y} \quad \text{and} \quad b=\frac{\sum_{i=1}^{n}(x_i-\bar{x})y_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}.$$

These values when inserted in (1) yield the line $y'-\bar{y}=b(x_i-\bar{x})$ which is usually called the regression line.

If we write

$$S_{xy}=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})y_i$$

and

$$S_x^2=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2,$$

then we may write

$$b=\frac{S_{xy}}{S_x^2}=\frac{S_{xy}}{S_xS_y}\cdot\frac{S_y}{S_x}=r\frac{S_y}{S_x}.$$

Here $r = \frac{S_{xy}}{S_x S_y}$ is called the correlation coefficient. The value of r must satisfy the inequality $-1 \leqq r \leqq 1$. The value of r will be equal to ± 1 if and only if, the points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ lie on the regression line.

3. The archeological bases of the methods of Brainerd-Robinson and Dempsey-Baumhoff

Let us assume that we compare n sites. We denote by S_i $(i=1, 2, \ldots, n)$ the set of the objects of i-th site and by T_i $(i=1, 2, \ldots, n)$ the set of the types of the i-th site. Put $S = \bigcup_{i=1}^{n} S_i$ and $T = \bigcup_{i=1}^{n} T_i$. The number A_{KS_i} means the precentage of the objects of type K belonging to the i-th site. The correlation between site i and site j according to Brainerd and Robinson is defined by

$$X_{ij} = 200 - \sum_{K \in T} |A_{KS_i} - A_{KS_j}|. \tag{2}$$

This may be written in the following from

$$X_{ij} = 200 - \sum_{K \in T_i - T_j} A_{KS_i} - \sum_{K \in T_j - T_i} A_{KS_j} - \sum_{K \in T_i \cap T_j} |A_{KS_i} - A_{KS_j}|.$$

From this one can easily seen that the method of Brainard-Robinson is based on the following principle. If two sites have types in essentially different percentages or if there are types which belong to one of the two sites but absent other site then the two sites originate from different times.

If $T_i = T_j$ i.e. the i-th and the j-th sites have the same types then in the above formula the first and second sums are equal to zero. Thus the agremeent between the i-th and j-th sites is determined by third sum. If the desagreement is small between i-th and j-th sites then the members of the foregoing sum $\left(\sum_{K \in T_i \cap T_j} |A_{KS_i} - A_{KS_j}|\right)$ are also small. This is the only case, according to Brainerd-Robinson's method, the two sites are of an age.

This means that the percentage of each type is approximately the same in the both sites.

Even if the site S_i contains essentially more objects than the site S_j, consequently, the site S_i has a greater number of the objects of the type K than the site S_j. The point of view of archeology this is such a requirement regarding to two sites which only rarely holds.

The element X_{ij} of the matrix used in Dempsey-Baumhoff's method is given by formula

$$X_{ij} = N - \sum_{K \in T_i \triangle T_j} 1, \tag{3}$$

where N means the number of all types belonging to sites S_i $(i=1, 2, \ldots, n)$. According to this formula the method of Dempsey-Baumhoff is based on the following principle. If two sites have the same types than both sites are of an age. However there exist such types which belong to one of the two sites only then they originate from different times.

These principles show that the two methods are essentially different. Later we shall return this question and we shall formulate the difference between these methods in the language of mathematics.

4. The mathematical analysis of the methods of Brainard-Robinson and Dempsey-Baumhoff

The first method assigns to each pair (S_i, S_j) $(i, j = 1, 2, \ldots, n)$ of sites the number given by formula (2), the second one assigns the number given by formula (3).

Let us correspond to each pair (S_i, S_j) either the number

$$\sum_{K \in T_i \triangle T_j} |A_{KS_i} - A_{KS_j}| + \sum_{K \in T_i \cap T} |A_{KS_i} - A_{KS_j}| \tag{4}$$

or the number

$$\sum_{K \in T_i \triangle T_j} 1. \tag{5}$$

For the sake of brevity, let us denote the number (4) by $r(S_i, S_j)$ and the number (5) by $\varrho(S_i, S_j)$, respectively. The function corresponding to the first method is $200 - r(S_i, S_j)$ and the function corresponding to the second one is $N - \varrho(S_i, S_j)$. It is clear that the determination of chronological order we may use the function $r(S_i, S_j)$ instead of $200 - r(S_i, S_j)$ in the case of the first method and the function $\varrho(S_i, S_j)$ instead of $N - \varrho(S_i, S_j)$ in the case of the second one.

We shall prove that the functions r and ϱ satisfy the

$$\varrho(S_i, S_j) \leqq \varrho(S_i, S_k) + \varrho(S_k, S_j) \tag{6}$$

and

$$r(S_i, S_j) \leqq r(S_i, S_k) + r(S_k, S_j) \tag{7}$$

inequalities, respectively.

First we prove the inequality (6). Let us correspond to each subset L of the set T the number of the element of L (that is the number of types contained in L) which we denote by $\mu(L)$. The domain of the function $\mu(L)$ is the set $P(T)$ of all subsets of T and its values are non-negativ numbers. If L and M are disjoint subsets of T then

$$\mu(L \cup M) = \mu(L) + \mu(M),$$

i.e. the function $\mu(L)$ is additive.

The function $\varrho(S_i, S_j)$ can be given with the aid of function $\mu(L)$ as follows

$$\varrho(S_i, S_j) = \mu(T_i \triangle T_j).$$

Thus the inequality (6) obviously follows from the additivity of μ.

After this we are going to prove the inequality (7). This may be rewritten in the following form

$$\sum_{K \in T_i \cup T_j} |A_{KS_i} - A_{KS_j}| \leqq \sum_{K \in T_i \cup T_k} |A_{KS_i} - A_{KS_k}| + \sum_{K \in T_k \cup T_j} |A_{KS_k} - A_{KS_j}|. \tag{8}$$

Now the left-hand side of (8) in detail is

$$(9)\qquad \sum_{K\in T_i-(T_j\cup T_k)} A_{KS_i}+\sum_{K\in T_j-(T_i\cup T_k)} A_{KS_j}+\sum_{K\in(T_i\cap T_j)-T_k}|A_{KS_i}-A_{KS_j}|+ \\ +\sum_{K\in(T_j\cap T_k)-T_i} A_{KS_j}+\sum_{K\in(T_i\cap T_k)-T_j} A_{KS_i}+\sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_i}-A_{KS_j}|$$

and the right-hand side of (8)

$$(10)\qquad \sum_{K\in T_i-(T_j\cup T_k)} A_{KS_i}+\sum_{K\in T_k-(T_i\cup T_j)} A_{KS_k}+\sum_{K\in(T_i\cap T_j)-T_k} A_{KS_i}+ \\ +\sum_{K\in(T_i\cap T_k)-T_j}|A_{KS_i}-A_{KS_j}|+\sum_{K\in(T_j\cap T_k)-T_i} A_{KS_k}+\sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_i}-A_{KS_k}|+ \\ +\sum_{K\in T_k-(T_i\cap T_j)} A_{KS_k}+\sum_{K\in T_j-(T_i\cup T_k)} A_{KS_j}+\sum_{K\in(T_i\cap T_j)-T_k} A_{KS_j}+ \\ +\sum_{K\in(T_i\cap T_k)-T_j} A_{KS_k}+\sum_{K\in(T_j\cap T_k)-T_i}|A_{KS_k}-A_{KS_j}|+\sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_k}-A_{KS_j}|.$$

We omit from (9) and (10) the members occuring in the both (9) and (10).

By the application of the triangle inequality we get

$$(11)\qquad \sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_i}-A_{KS_j}|\leqq\sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_i}-A_{KS_k}|+\sum_{K\in T_i\cap T_j\cap T_k}|A_{KS_k}-A_{KS_j}|;$$

$$(12)\qquad \sum_{K\in(T_i\cap T_j)-T_k}|A_{KS_i}-A_{KS_j}|\leqq\sum_{K\in(T_i\cap T_j)-T_k} A_{KS_i}+\sum_{K\in(T_i\cap T_j)-T_k} A_{KS_j};$$

$$(13)\qquad \sum_{K\in(T_i\cap T_k)-T_j} A_{KS_i}\leqq\sum_{K\in(T_i\cap T_k)-T_j}|A_{KS_i}-A_{KS_k}|+\sum_{K\in(T_i\cap T_k)-T_j} A_{KS_k};$$

$$(14)\qquad \sum_{K\in(T_j\cap T_k)-T_i} A_{KS_j}\leqq\sum_{K\in(T_j\cap T_k)-T_i}|A_{KS_k}-A_{KS_j}|+\sum_{K\in(T_j\cap T_k)-T_i} A_{KS_k}.$$

The left-hand sides of (11), (12), (13), (14) add up the left-hand side of the (8) and similarly the right-hand sides of (11), (12), (13), (14) add up the right-hand side of the (8), disregarding the omitted members and the sum

$$2\sum_{K\in T_k-(T_i\cup T_j)} A_{KS_k}.$$

From this we can infer that the inequality (8) and automatically the inequality (7) holds..

5. The application of the regression theory to the chronological seriation

From the foregoing it can be easily seen that the function $r(S_i, S_j)$ in the method of Brainerd—Robinson and the function $\varrho(S_i, S_j)$ in the method of Dempsey-Baumhoff determine each a pseudometric space. In the prerequisites it was shown that a pseudo-metric induces a metric on the set of all sets $G_i=\{S_j\colon \varrho(S_i, S_j)\}=0$. Thus we may assume, with no loss of generality, that the function $r(S_i, S_j)$ and $\varrho(S_i, S_j)$ are metrics. Arises the question what kind of a metric are induced by the function r and ϱ in the set of the sites. Are they similar to the metric of the straight line or euclidean plane. Precisely, they are whether or not congruently imbeddable in the euclidean plane. It may happen that the imbedding is not possible.

Namely, let us consider, four sites A, B, C, D. Assume that each of the sites have the same types: I, J, K, L, M, N, P, Q. Assume moreover, that in the site A the type I occurs in percentage 25, the type M in percentage 45, and the other types occur in percentages $5-5$; in the site B the type J occurs in percentage 25, the type N in percentage 45 and the other types in percentages $5-5$; in the site C the type K occurs in percentage 25, the P in percentage 45 and the other types occur in percentages $5-5$; and finally in the site D the type L occurs in the percentage 25, the type Q in percentage 45 and the other types occur in percentages $5-5$.

By the method of Brainerd-Robinson

$$r(A, B)=r(A, C)=r(A, D)=r(B, C)=r(B, D)=r(C, D)=120,$$

i.e. the distance of each pair of the four sites is the same. Since we cannot find in the plane four distinct points such that any pair of them has the same non-zero distance, the metric space determined by the set $\{A, B, C, D\}$ and the metric r is not congruently imbeddable in the plane. We may make a similar example in the case of the method of Dempsey-Baumhoff. It is easy to see that in such cases neither the Brainerd-Robinson's method nor Dempsey-Baumhoff's method cannot give a chronological order.

In the reality, however, such cases occur only when we commit an error in the preparation of the archeological material or in our calculations. After a new examination we may find the trouble.

We have seen the difference between principles on which the methods of Brainerd-Robinson and Dempsey-Baumhoff are based. This may the right time to straighten out the different in another way. Arises the question that the metric space induced by the sites and the metric $\varrho(S_i, S_j)$ can be congruently imbeddable in the metric space induced by the sites and the metric $r(S_i, S_j)$. In general this is not possible. Consequently, the chronological orders obtained by the two methods are not the same, because both methods determine the chronological order comparing the sizes of the distances of the sites.

In order to establish the chronological seriation we need at least demand that the metric space induced by the set of sites and the function r or ϱ be congruently imbeddable in the plane. But in this case it is reasonable to apply the theory of regression. First we must decide that the metric space induced by the function $r(S_i, S_j)$ on the set of the sites are imbeddable whether or not in the plane. If this is not possible then we must examine preliminary analyses particularly the isolation of the types.

It is known various methods to decide the possibility of the imbedding. We may use the following general theorem [3].

An arbitrary metric space S with metric r is congruently imbeddable in euclidean n-dimensional space if and only if (i) S contains an $t+1$-tuple $p_0, p_1, \ldots, p_t$ $(t \leqq n)$ such that the determinant

$$D(p_0, p_1, \ldots, p_k) = \begin{vmatrix} 0 & 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & 0 & r^2(p_0p_1) & \cdot & \cdot & \cdot & r^2(p_0p_k) \\ 1 & r^2(p_1p_0) & 0 & \cdot & \cdot & \cdot & r^2(p_1p_k) \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & r^2(p_kp_0) & \cdot & \cdot & \cdot & \cdot & 0 \end{vmatrix}$$

where $k=1, 2, \ldots, t$, has the sign of $(-1)^{k+1}$, (ii) for every pair (x, y) of points of S the determinants $D(p_0, p_1, \ldots, p_t, x)$, $D(p_0, p_1, \ldots, p_t, y)$, $D(p_0, p_1, \ldots, p_t, x, y)$ vanish. We use this theorem in the case of $n=2$. Since each set of three points of a metric space is congruently contained in the euclidean plane, we must verify that for any four points p_0, p_1, p_2, p_3 of the metric space of the sites the determinant

$$D(p_0, p_1, p_2, p_3) = \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & r^2(p_0p_1) & r^2(p_0p_2) & r^2(p_0p_3) \\ 1 & r^2(p_0p_1) & 0 & r^2(p_1p_2) & r^2(p_1p_3) \\ 1 & r^2(p_0p_2) & r^2(p_1p_2) & 0 & r^2(p_2p_3) \\ 1 & r^2(p_0p_3) & r^2(p_1p_3) & r^2(p_2p_3) & 0 \end{vmatrix}$$

vanish. If the metric space determined by the set of the sites and the metric r is imbeddable in the plane then we do imbedding (for example graphically). After this we compute the coordinates of the points of the plane corresponding to the sites and with the aid of the theory of regression the regression line to points corresponding in the plane to the sites. The correlation coefficient shows the position of the points which represent the sites in the plane, relative to the regression line. If correlation coefficient is equal to $+1$ or -1 then every point lies on the regression line. If the correlation coefficient differs from ± 1 then there exist points do not lie on the regression line. If the correlation coefficient is close to ± 1 then the distances of the points from the regression line which are outside of the regression line are small.

Inasmuch as each point is on the regression line, we consider the position of the points on this line as the chronological order of the sites. Otherwise we project the points onto the regression line perpendicularly, and we consider the position of the images as the chronological order. Thus the reliability of the chronological seriation depend upon the value of the correlation coefficient. If the correlation coefficient is close $+1$ or -1 then the chronological seriation is satisfactory. The advantage of this method is that we can control simultaneously the preciseness of the preliminary analyses.

References

[1] BRAINERD, G. W., The place of chronological ordering in archeological analysis, *Amer. Antiquity*, v. 16, 1951, pp. 301—313.
ROBINSON, W. S., A method for chronological deposits, *Amer. Antiquity*, v. 16, 1951, pp. 293—301.
[2] DEMPSEY, P. & M. BAUMHOFF, The statistical use of artifact distributions to establish chronological sequence, *Amer. Antiquity*, v. 28, 1963, pp. 496—509.
[3] BLUMENTHAL, L. M., *Theory and applications of distance geometry*, Oxford, 1953, p. 104.

(Received March 23, 1970)

Применение алгоритмов обучения в метеорологии для предсказывания конвективной активности

G. Szalay*, L. Molnár**, O. Gulyás**

Введение

Одной из целей нашей работы являлось исследование возможностей применения алгоритмов обучения или распознавания, посредством осуществления конкретных практических применений. Другой целью — в связи с предыдущем — являлось сравнение алгоритмов между собой с точки зрения надежности, скорости сходимости и других факторов, на основании достигнутых при решении практических задач результатов.

В настоящей работе показывается применение нескольких алгоритмов обучения, действующих по методу обучения с учителем, для задачи, связанной с предсказыванием погоды.

Метеорологическая задача относится к предсказыванию конвективной активности, являющейся мерой атмосферной грозовой деятельности.

В начале данной работы ради облегчения понимания дается краткий обзор использованных алгоритмов обучения и реализующих их программ для ЭВМ.

Эта работа является частью деятельности имеющей место в НИИ Связи (ТКИ), по областям диагностики посредством ЭВМ и распознавания образов.

Решение излагаемой задачи произведено в рамках сотрудничества между Центральным Метеорологическим Институтом и НИИ Связи.

Создание метеорологической модели и подготовка данных были произведены в Центральном Метеорологическом Институте, а теоретические исследования, связанные с алгоритмами, изготовление машинных программ и опыты посредством ЭВМ произвелись в НИИ Связи. Оценка результатов произведена сотрудниками обоих Институтов сообща.

Полученные до сих пор результаты показали, что метеорологическая модель и излагаемые алгоритмы пригодны для решения поставленной задачи.

Однако нашей конечной целью является внедрение процедур для ЭВМ, использующих алгоритмы обучения в оперативную метеослужбу, при дальнейшем их развитии и увеличении их эффективности.

1. О выборе алгоритмов

В распознавании образов известно множество различных вариантов алгоритмов обучения. Подробный обзор их находится в [1]—[4].

Для выбранной нами задачи наиболее полходящими являются алгоритмы действующие по методу обучения с учителем.

Первую группу алгоритмов, использованных при решении задачи, составляют алгоритмы, действующие методом потенциальных функций. Метод потенциальных функций является одним из наиболее основательно разработанных видов алгоритмов обучения.

Эвристическое введение его содержится в [5] и [14], а теоретические основы и наиболее известные разновидности изложены в [6] и [7].

В НИИ Связи, кроме исследования и реализации при помощи ЭВМ, занимались и обобщением этих алгоритмов и другими теоретическими исследованиями [8]—[11].

Выбор метода алгоритмов потенциальных функций сверх вышеуказанных мотивировался и тем, что эти алгоритмы уже были успешно применены в других областях метеорологии [19].

Вторым использованным алгоритмом являлся метод полиномиальной дискриминации [ПДМ]. Теоретические основы ПДМ содержатся в [12], а модифицированные и усовершенствованные варианты в [14].

Его связь с методом потенциальных функций показывается в [14] и [15]. Основным достоинством ПДМ является то, что он очень надежно классифицирует и в случае наличия относительно малой обучающей последовательности. В работе [13] показывается успешное применение ПДМ в кардиологии. Третий использованный алгоритм, реализующий простейший из видов многослойных сетей пороговых элементов, так называемый алгоритм "committee machine" (СМ). СМ способен образовать кусочно-линейную поверхность разделения.

Теоретические основы линейного и кусочно-линейного разделения, а также разные варианты этого алгоритма содержатся в [3] и [16]. Об успешном применении для предсказывания осадков алгоритма кусочно-линейного разделения сообщается в [17].

В последующем рассмотрим общие черты алгоритмов обучения, действующих по методу обучения с учителем, далее коротко излагаются теоретические основы вышеуказанных трех алгоритмов.

2. Алгоритмы обучения действующие по методу обучения с учителем

Излагаемые нами алгоритмы принадлежат к группе алгоритмов разделения с помощью разделяющих функций.

Наши алгоритмы обучения служат для разделения подмножеств n-мерного Эвклидова пространства R^n. Без ограничения общности можем предполагать число разделяемых подмножеств равным двум. Соответственно пусть будут заданы множества A и B в *основном пространстве* X, для которых

$$\begin{aligned} A \cup B &\subset X, \\ A \cap B &= \emptyset. \end{aligned} \tag{2.1}$$

Разделяющей функцией (функцией разделения) называются такие функции $f(x)$, что

$$\operatorname{sign} f(x) = \begin{cases} +1 \text{ при } x \in A, \\ -1 \text{ при } x \in B. \end{cases} \tag{2.2}$$

Материалом обучения или *обучающей последовательностью* называется множество точек $\{x^i \in X\}_1^\infty$, если (1) выбор каждой точки x^i производится независимо (2) по общему распределению вероятностей $p(x)$.

У алгоритмов обучения, действующих по методу обучения с учителем, к каждой точке обучающей последовательности принадлежит указание о том, к какому множеству следует отнести данную точку, и это называется *учением*. Обучение же представляет собой последовательность вероятностных переменных $\{\hat{x}^i\}_1^\infty$, где

$$\begin{aligned} \hat{x}^i &= f(x^i), \text{ или} \\ \hat{x}^i &= \operatorname{sign} f(x^i). \end{aligned} \tag{2.3}$$

Пусть $\hat{\pi}_n$ и π_n являются совокупностями n первых элементов обучения и обучающей последовательности соответственно.

Задачей алгоритмов обучения является выработка такой функции $f_n(x)$, которая на основании обучающей последовательности и обучения «в некотором смысле» образует $f(x)$, т. е.

$$f_n(x) = f_n(x; \pi_n, \hat{\pi}_n) \cong f(x), \tag{2.4}$$

где $\pi_n = \{x^1, x^2, \ldots, x^n\}$,

$\hat{\pi}_n = \{\hat{x}^1, \hat{x}^2, \ldots, \hat{x}^n\}$.

2.а. *Метод потенциальных функций*

При алгоритмах потенциальных функций предполагаем, что разделяющая функция допускает запись в виде:

$$f(x) = \sum_{i=1}^{N} c_i \varphi_i(x) \tag{2.а.1}$$

где $\{\varphi_i(x)\}_{i=1}^N$ является линейной системой независимых функций.

n-ое приближение разделяющей функции $f_n(x)$ образуется следующим рекурсивным путем:

$$f_n(x) = f_{n-1}(x) + r_n(x^n) \cdot K(x, x^n) \tag{2.а.2}$$

где определяемая в виде

$$K(x, y) = \sum_{i=1}^{N} \varphi_i(x) \cdot \varphi_i(y) \tag{2.а.3}$$

функция называется потенциальной функцией, а корректирующая переменная $r_k(x^k)$ имеет различный, описанный в [6] и [7] вид при каждом конкретном алгоритме.

В дальнейшем при

$$r_k^{\Theta}(x^k) = \Theta[f(x^k) - f_{k-1}(x^k)]$$
$$0 \leqq \Theta \leqq 2/\sup_x K(x, x) \qquad (2.\text{a}.4)$$

алгоритм называем тета-алгоритмом, а при

$$r_k^{\gamma}(x^k) = \gamma_k \cdot \operatorname{sign}[f(x^k) - f_{k-1}(x^k)]$$
$$\gamma_k > 0, \quad \sum_1^{\infty} \gamma_k = \infty, \quad \sum_1^{\infty} \gamma_k^2 < \infty \qquad (2.\text{a}.5)$$

— гамма-алгоритмом.

Теоремы сходимости, относящиеся к алгоритмам содержатся в [7] и [9].

В сделанной нами программе, реализующей алгоритмы обучения по методу потенциальных функций для ЭВМ, выбрали разделяющую функцию в виде полинома.

Количество переменных и число степеней полинома ограничивается лишь емкостью памяти ЭВМ. Коэффициенты полинома вычисляются рекурсивным путем, поэтому нет необходимости сохранять в памяти точки обучения. Вследствие этого потребность в памяти в течении обучения остается постоянной и не зависит от длины обучающей последовательности.

Программа — подобно всем описываемым в настоящем докладе программам — написана на языке Алгол — 60 в машинной репрезентации GIER—ALGOL 4.

2.б. *Метод полиномиальной дискриминации (ПДМ)*

Сводка теоретических основ алгоритма содержится в [12]. Алгоритм ПДМ способен дать не только приближение разделяющей функции, но кроме того, дает и приблизительную оценку распределения вероятностей точек обучения, принадлежащих к каждому отдельному классу образов.

Оценка вышеуказанных функций плотностей распределений вероятностей производится таким образом, что для каждой точки обучающей последовательности назначается так называемая функция интерполяции и вычисляется их среднее значение.

В случае одномерного основного пространства отношения иллюстрируются на рисунке I.

С целью облегчения вычислительной работы в качестве интерполяционных функций были выбраны экспоненциальные функции.

Таким образом принадлежащая к классу A функция плотности принимает вид:

$$f_A(\mathbf{x}) = \frac{1}{(\sigma\sqrt{2\pi})^p} \frac{1}{m} \sum_{i=1}^{m} \exp\left[-\sum_{k=1}^{p} (x_{aik} - x_k)^2 / 2\sigma^2\right], \qquad (2.\text{б}.1)$$

где:

m — число точек обучающей последовательности, принадлежащих к классу A,

p — число мер основного пространства,

$\mathbf{x}_{ai} = (x_{ai1}, x_{ai2}, \ldots, x_{aip})$ — i-тая точка обучающей последовательности.

При формировании правила решения, применялась Бейесова стратегия. При проблеме классификации в две категории точка $\mathbf{x}$ причисляется к категории

A, если

$$h_A \cdot l_A \cdot f_A(\mathbf{x}) > h_B \cdot l_B \cdot f_B(\mathbf{x}), \qquad (2.\text{б}.2)$$

где

h_A и h_B — априорная вероятность нахождения принадлежащей к классу A или B точки,

l_A и l_B — величины, характеризующие потери, которые возникают при неправильной классификации относящихся к классам A и B точек.

(Вышеуказанное правило решения легко обобщается и для проблемы классификации на $M > 2$ категорий.)

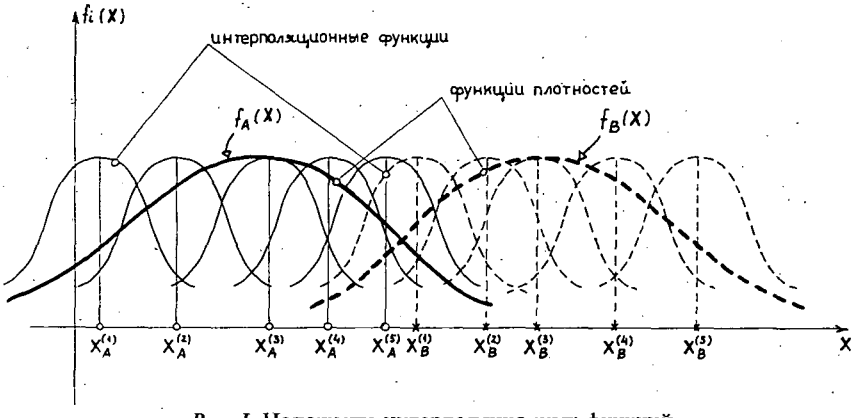

Рис. I. Наложение интерполяционных функций

Однако, алгоритм, применяющий функции плотностей распределений вида (2.б.1) применим лишь при сравнительно небольшом количестве точек — представителей, т. к. в процессе обучения необходимо хранение всех точек обучающей последовательности, при этом классификация неизвестной точки требует пропорциональной длины обучающей последовательности вычислительной работы.

Этот недостаток устраняется разложением в ряд Тейлора выражения (2.б.1).

После разложения в ряд функция $f_A(\mathbf{x})$ разлагается — подробно описанным в [12] и [14] образом — на произведение независимой от точек обучающей последовательности положительной функции и на полином, поэтому для дальнейших классификаций достаточно применять этот полином. Коэффициенты полинома вычисляются из координат точек обучающей последовательности рекурсивным путем. В этом случае достаточно хранить вместо точек обучающей последовательности только коэффициенты полинома. Следовательно, потребность в памяти в процессе обучения постоянна.

Классификация неизвестных точек состоит из вычисления значения полинома подставляя координаты точки. Однако за вышеуказанные преимущества придется платить возникающей вследствие усечения ряда ошибкой. Опыт показывает, что при решении практических задач точность классификации не снижается значительно, по сравнению с методом классификации, применяющим функции плотностей вида (2.б.1).

Следует отметить, что применив ПДМ — в противоположность применения алгоритма потенциальных функций — полученная разделяющая функция не зависит от порядка следования точек обучающей последовательности. Эффективность алгоритма в значительной мере зависит от правильного выбора свободного параметра σ в (2.б.1) и участвующих в Бейесовом правиле решения коэффициентов потерь.

Программа, реализующая алгоритм для ЭВМ, состоит из рекурсивного вычисления коэффициентов полинома.

2.в. *Алгоритм кусочно-линейного разделения*

Известно, что линейным разделением решается лишь сравнительно узкий круг практических задач. Одним из возможных обобщений линейного разделения является метод кусочно — линейного разделения. Кусочно — линейную разделяющую поверхность можно создать соединением в сеть линейных решающих элементов (TLU: threshold logic unit).

Каждый решающий элемент (называется и пороговым элементом TLU), имеющий весовой вектор (вектор решения) **w**, классифицирует точку обучающей последовательности **x** по значению функции sign (**w**, **x**).

Сеть, осуществляющая кусочно-линейное разделение (т. н. "committee machine") является простейшим типом многослойных сетей пороговых элементов.

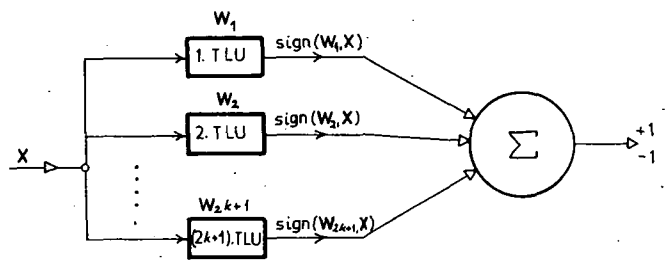

Рис. 2. "Committee machine"

Единственный находящийся во втором слою TLU считает «голоса» и решает на основании большинства (из этого следует, что в первом слою необходимо иметь нечетное число TLU).

В процессе обучения при неправильном решении устройства по заданному правилу изменяем векторы решений некоторых TLU. Например по описанному в [4] и использованному нами методу, к векторам решений нескольких определенных, неправильно классифицировавших элементов добавили произведение точки обучающей последовательности с заданным постоянным коэффициентом: $\mathbf{w}_j'' = \mathbf{w}_j' + C \cdot \mathbf{x}$ где

$\mathbf{w}_j'$ — вектор решения TLU до учения,
$\mathbf{w}_j''$ — вектор решения TLU после учения,
C — постоянный коэффициент,
$\mathbf{x}$ — неправильно классифицированная точка обучающей последовательности.

Эта процедура повторяется до того, пока не добъемся правильной классификации точки **x**.

Для алгоритма кусочно-линейного разделения неизвестны теоремы о его сходимости. О сходимости обучения можно судить лишь на основании опытов.

На скорость обучения начальная настройка элементов имеет существенное влияние.

3. Метеорологическая модель задачи

Одним из типичных областей применения алгоритмов обучения является предсказывание погоды.

Применение в метеорологии методов распознавания образов и алгоритмов обучения излагается в [18], [20] и [21] в общем виде, а об успешном их использовании для конкретных метеорологических задач сообщается в [17] и [19]. В [17] сообщено об опытах, связанных с предсказыванием осадков, с помощью алгоритма кусочно-линейного разделения. В [18] описывается причисление к определенным погодным зонам отдельных территориальных частей СССР, с применением алгоритма самообучения, действующего по методу потенциальных функций. Судя по статьям, опубликованным в литературе, алгоритмы обучения дали согласные с ранее использованными методами, но в отдельных случаях превышающие их точность результаты. Однако, вопреки вышеуказанным результатам, в метеорологической практике машинные методы, применяющие алгоритмы обучения еще не получили распространение. Применение алгоритмов обучения испытано лишь в нескольких из многочисленных областей метеорологии.

В настоящей работе показывается решение задачи предсказывания конвективной активности, являющейся мерой грозовой деятельности атмосферы, с применением алгоритмов обучения.

В Венгрии в Центральном Метеорологическом Институте в последние годы была произведена интенсивная исследовательская работа в области метеорологических явлений, связанных с предсказыванием конвективной активности. Разработаны модель и метод для ЭВМ для объективного анализа условий окружения, благоприятствующих образованию конвективных процессов. Подробный отчет об этом находится в [22].

Индикаторы, разработанные в [22], составляли координаты точек обучающей последовательности, а последующая фактическая конвективная активность составляла учение.

Опыты, произведенные в процессе решения задачи, подробно описываются в [23].

Достигнутые результаты и получаемые из них выводы содержатся в [24]. Изложение результатов с точки зрения алгоритма ПДМ дается в [25].

В последующем дается описание в общих чертах существенных частей метеорологической модели, далее даются результаты, полученные применением разных алгоритмов, наконец, сравнение алгоритмов между собой.

В процессах атмосферных движений важную роль играют вертикальные потоки, которые появляются вследствие возмущений неустойчивых распределений воздушных масс.

Эти упорядоченные воздушные потоки называются конвекцией. Наиболее развитой их формой является конвекция дождевых кучевых облаков, что вертикально охватывает всю тропосферу, ее горизонтальные размеры порядка 10-ти км, а в линейной формации может достигать и несколько сотен км. Конвекция дождевых кучевых облаков всегда образуется при благоприятном содействии многочисленных факторов.

Принято считать самыми важными из этих факторов следующие:

а) достаточное содержание водяного пара в нижних слоях воздуха;

б) потенциальная неустойчивость гидростатического равновесия воздушного столба;

в) существование механизма динамики, вызывающего выделения энергии неустойчивости.

В работе [22] описана процедура, применимая для ЭВМ, которая выявляет обстоятельства, определяющие формирование явлений конвекции дождевых кучевых облаков.

Для охарактеризования условий, при которых происходит образование конвективной активности, нами применены 12 параметров, выбранных таким образом, чтобы они с необходимой полнотой и весовым соотношением представляли вышеуказанные три фактора.

Для этого предикторы определены таким образом, чтобы каждый индикатор состоял из параметров одинаковой физической природы.

Определены всего 4 индикатора конвективной активности.

Индикатор I_1 состоит из гидростатических параметров. Среди его членов имеются: индекс устойчивости, индекс влажности, величина, характеризующая содержания влаги нижних уровней, и наконец параметр, характеризующий взаимность структур полей температуры и влажности.

Индикатор I_2 представляет собой развитие неустойчивости воздушного столба и состоит из разности геострофических адвекций температур на уровнях 500 и 850 миллибар.

Индикатор I_3 содержит такие параметры, которые указывают на существование механизмов, активизирующих скрытую неустойчивость.

Среди этих параметров находятся генез вихря скорости, вычисленная для уровня 850 мбар, а также функция генеза фронта накопления температуры и точки росы, далее геострофическая адвекция относительного вихря скорости.

Посредством индикатора I_4 попытались характеризовать одновременное присутствие развития неустойчивости и механизма активизации. Этот индикатор отмечает те области, внутри которых господствующими являются горизонтальная сходимость потока воздуха на нижних уровнях, а на средних уровнях горизонтальная расходимость потока воздуха; далее внизу наблюдаются адвекция теплого и влажного, а наверху адвекция сухого и холодного воздуха.

Иными словами I_4 отмечает те области, внутри которых наблюдается оптимальная комбинация развития неустойчивости возушного столба, переустройство порядка вертикального распределения влаги, и триггер, возбуждающий конвективную циркуляцию.

Подводим итоги: целью настоящей работы является предсказывание величины конвективной активности. Из вышесказанного следует, что состояние атмосферы описывают 4 индикатора конвективной активности.

Площадь Западной и Средней Европы покрыли прямоугольной сетью, состоящей из 13×8 узловых точек сети. Таким образом получили $12 \times 7 = 84$ квадрата, состояние воздушного пространства внутри которых описывается индикаторами в 4-х точках сети (рис. 3).

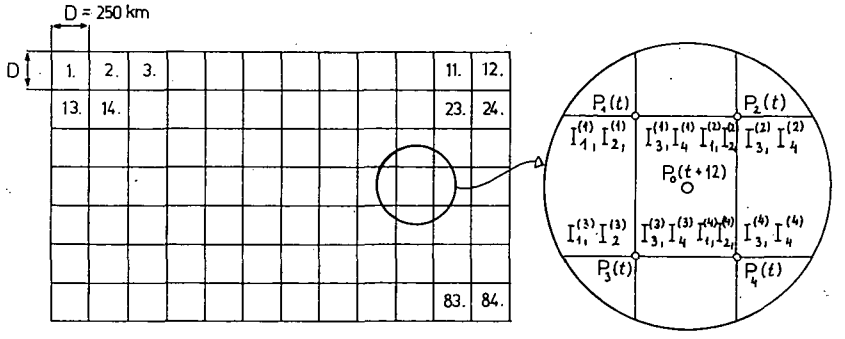

Рис. 3. Ортогональная сеть, покрывающая исследованную область

Каждый квадрат, длиной стороны 250 км, характеризуется 16 параметрами, т. е. при выработке существенных признаков задача отображается в 16-мерное Эвклидово пространство.

На основании этих 16 параметров следует определить появление или отсутствие конвективной активности в квадрате на по следующие 12 часов. Для опыта нами использованы индикаторы, вычисленные из данных, полученных высотными измерениями состояния воздуха 11-го июля 1968-го года в 12^{00} по Гринвичу. Параметр, описывающий фактическую конвективную активность в промежутке времени 12^{00}—24^{00} по Гринвичу представляет собой учение.

Кроме 250-километровой сети, была применена и сеть с длиной стороны квадрата 125 км. Данные каждой сети были определены наложением прямоугольной сети на обработанные для сферической системы координат изолинии, и для каждой точки — по мере необходимости — были проведены интерполяции значений.

Величины параметра конвективной активности определялись следующим образом:

а) Если не было конвективной активности, то $u = -1$,
б) При редких ливнях $u=1$,
в) При многочисленных ливнях $u=2$,
г) При грозе, отмеченной какой-то станцией $u=3$,
д) При сильной грозовой деятельности $u=4$.

4. Результаты, полученные применением программ и следствия, вытекающие из них

Каждый алгоритм из трех оказался пригодным для решения метеорологической задачи. Для проверки в случае алгоритма потенциальных функций, содержащего потенциальную функцию в виде полинома, и в случае алгоритма кусочно-линейного разделения была использована обучающая последовательность, что допустимо потому, что эти процедуры являются итеративными процедурами и автоматически не разделяют правильно на обучающую последовательность.

Однако для проверки алгоритма ПДМ была использована последовательность, отличающаяся от обучающей. Проверка производилась двумя различными методами. В случае первого метода данные находящиеся в нашем распоряжении были разделены на две части: первая часть была использована для обучения, а вторая для проверки (ПДМ/I). При втором методе обучение было проведено столько раз, сколько имелось точек обучающей последовательности. Из имеющихся точек была выделена одна, остальными было проведено обучение, а потом с помощью полученной разделяющей функции была произведена классификация оставшейся точки (ПДМ/II).

Ниже показываются несколько характерных результатов.

В таблице H_1 и H_2 обозначают ошибки первого и второго родов. Ошибкой первого рода называется случай, когда алгоритм предсказывает ясную погоду, а в действительности появляется гроза.

Под точностью понимаем относительную частоту правильного предсказания.

В случае линейной разделяющей функции алгоритмы потенциальных функций дали следующие результаты: (таблица I).

Таблица I. Результаты алгоритмов потенциальных функций

Алгоритм	Перфолента данных	Точки проверки (тест)	H_1	H_2	$H=H_1+H_2$	Точность (%)
Тета	«250 km»	$84 = 33_{\text{гр.}}+51$	9	6	15	82.2
Гамма	«250 km»	$84 = 33_{\text{гр.}}+51$	14	10	24	71.5
Тета	«125 km»	$336 = 92_{\text{гр.}}+244$	20	30	50	85.2
Гамма	«125 km»	$336 = 92_{\text{гр.}}+244$	36	21	57	83.1

Из таблицы видно, что тета-алгоритм, у которого величина коррекции не зависит от положения точки внутри обучающей последовательности, дал лучшие результаты чем гамма-алгоритм, где величина коррекции убывает в процессе обучения. При применении разделяющих функций более высокого порядка чем линейные точность не увеличивалась, что наводит на мысль о том, что положение не таково будто бы два множества в 16-и мерном пространстве, соответствующие состояниям «гроза» и «не гроза», являются непересекающимися, только линия раздела очень сложна-, а наоборот, эти два множества пересекаются (см. рис. 4).

В случае итеративного алгоритма, выработанная разделяющая функция подвергается влиянию порядка следования точек обучающей последовательности. Поэтому, у алгоритмов потенциальных функций нами было проведено и испытание с целью определения того, как влияет порядок следования точек обучающей последовательности на точность разделения.

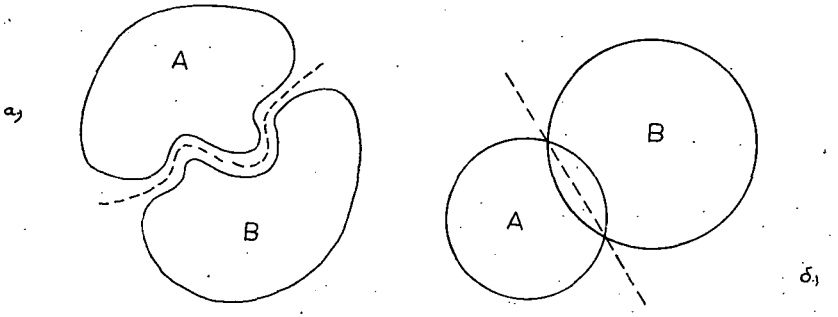

Рис. 4. Характер разделяемых множеств

Для этой цели были изготовлены несколько «смешанных» последовательностей данных. «Смешанная» последовательность получена следующим образом: последовательность точек, полученных для 250-километровой сети смешали относительно их «естественного» порядка следования (см. рис. 6). Результаты, полученные такими «смешанными» данными, показаны в таблице II и полученны с применением линейного тета-алгоритма.

Таблица II. Результаты, полученные применив несколько «смешанных» последовательностей данных

Перфолента данных	Число однородных отрезков	H_1	H_2	H	Точность (%)
«естественная»	15	9	6	15	82.2
«случайная»	35	8	4	12	85.7
«равномерная/1»	67	6	6	12	85.7
«равномерная/2»	67	5	5	10	88.1

В первой строке таблицы содержатся результаты последовательности данных с «естественной» очередностью. Так как грозовые зоны образуются обычно на связных областях, поэтому здесь получено относительно небольшое число однородных отрезков (внутри однородного отрезка находятся либо одни «грозовые» точки, либо одни точки «ясной погоды»). Во второй строке показан результат такой последовательность данных, где порядок следования точек установлен с помощью генератора случайных чисел. Здесь число однородных отрезков больше.

В третьей и четвертой строках показаны результаты таких последовательностей данных, где точки обучающей полседовательности поставлены в поря-

док следования, при котором старались придать распределению «грозовых» точек как можно большую равномерность.

Здесь число однородных отрезков максимальное. Из опытов видно, что имеется тенденция к повышению точности предсказывания с увеличением числа однородных отрезков.

Характерные результаты алгоритма ПДМ содержатся в таблице III.

Таблица III. Результаты алгоритма ПДМ

Метод испытания	Перфолента данных	Серия тестов	H_1	H_2	H	Точность (%)
ПДМ/I	«250 km»	$42 = 22_{гр.} + 20$	4	1	5	88.1
ПДМ/II	«250 km»	$84 = 33_{гр.} + 51$	9	8	17	80.0
ПДМ/I	«125 km»	$168 = 35_{гр.} + 133$	4	16	20	88.1

Алгоритм ПДМ надежно работает. Разделяющая функция, полученная при помощи обучающей последовательности, очень хорошо классифицирует также и серии тестов, отличающихся от обучающей последовательности.

Результаты, полученные с применением алгоритма кусочно-линейного разделения, приведены в таблице IV. Опыты, проведенные нами, показали, что в общем случае алгоритм кусочно-линейного разделения существенно сходится со скоростью, существенно меньшей, чем алгоритм потенциальных функций. Результаты, полученные применив исходную обучающую последовательность, оказались довольно слабыми (1). Поэтому нами была увеличена длина обучающей последовательности таким образом, что произвели обучения с простым 5-, 10-кратным повторением исходной обучающей последовательности. Удлинив таким образом обучающую последовательность, мы получили существенно улучшенные результаты (2 и 3).

Таблица IV. Результаты алгоритма кусочно-линейного разделения

Длина обучающей последовательности	H_1	H_2	H	Точность (%)
1-кратная	24	2	26	69.1
5-кратная	15	0	15	82.2
10-кратная	2	4	6	92.8

Результаты получены с применением 3 TLU и при значении корректирующего множителя $C=1$. Исходная обучающая последовательность состояла из 84 точек 250-километровой сети. Увеличение числа TLU не дало улучшение точности классификации.

Наконец, нами исследовалось и то, есть ли возможность уменьшать число мер. Результаты, приведенные в таблице V, показали, что задание с данными 16-мерного пространства является в значительной мере излишним.

В первой строке обозначение R3/1, 3, 4 означает, что из 4-х индикаторов для каждой точки один остался неиспользованным, и так применялись I_1, I_3

Таблица V. Результаты данных с уменьшенным числом мер

№	Вид сокращения	Число мер	H_1	H_2	H	Точность (%)
1	R3/1, 3, 4	12	7	6	13	84.5
2	R2/1, 4	8	7	7	14	83.1
3	R1/4	4	4	15	19	77.4
4	R4/среднее	4	11	7	18	78.5
5	R2/1, 4/среднее	2	7	8	15	82.2
6	R2/1, 4/среднее	2	4	2	6	85.7

и I_4. Таким образом проблема стала сокращенной в 12-мерное пространство. Во второй строке R2/1, 4 означает, что применены лишь индикаторы I_1 и I_4, тогда проблема станет 8-мерной. Из результатов видно, что в 12-мерном и даже в 8-мерном пространстве разделение осуществимо без уменьшения точности. Далее (3) видно, что при использовании лишь индикатора I_4, тем самым сократив проблему в 4-мерное пространство, точность не уменьшается значительно. Проблему можно сократить в 4-мерную и другим способом: образованием среднего из значений индикаторов, принятых ими в четырех углах квадрата. В этом случае нами достигнута приблизительно такая же точность разделения, что и при другой четырехмерной проблеме.

Особенный интерес представляют собой результаты, находящиеся в 5-ой и 6-ой строках таблицы V. В этих случаях проблема уже двухмерная, что достигнуто таким образом, что в последовательности вышеописанным способом вычислены средние значения. Применив эту двухмерную обучающую последовательность, алгоритм обучения классифицировал с такой же точностью, что и в случае исходного 16-мерного пространства данных! А программа ПДМ/I с такой же перфолентой данных, когда для обучения применялись первые 42 точки, а для проверки остальные, дала очень хорошую точность классификации (приблизительно 86%), что также очень мало отличалась от 88%-ного результата, полученного применив 16-мерную обучающую последовательность при подобных обстоятельствах. Ясно видно на основании опыта, полученного в связи с сокращением данных, что применение алгоритмов имеет значение также с точки зрения усовершенствования метеорологической модели, так как с их помощью можно судить об относительной важности каждого индикатора, с точки зрения точности предсказывания. Можно установить первостепенную важность индикатора I_4, ибо достигаемая на основании одного I_4 точность предсказывания мало отстает от точности получаемой при одновременном применении всех четырех индикатовор.

Вторым по важности можно отметить I_1, так как применив I_1 и I_4 точность достигает значения, которое имелось при применении всех четырех. Менее важны индикаторы I_2 и I_3, т. к. их отсутствие не имеет заметное влияние на точность классификации.

Результаты, находящиеся в табл. V, получены линейным тета-алгоритмом потенциальных функций, применив данные 250-километровой сети, за исключением 6-ой строки, полученной алгоритмом ПДМ/I.

Наконец, была исследована и причина появления ошибочных решений алгоритмов. Для этого сравнивали результаты девяти различных алгоритмов — разных вариантов показанных ранее алгоритмов — и приняли решение на основании большинства. Таким путем получены приведенные в таблице VI результаты.

Результаты, полученные применив вышеуказанные 9 алгоритмов, изображены на рис. 5. На рис. 5 изображена квадратическая сеть натянутая на карту Европы с длиной строны квадрата 250 км. Цифра, находящаяся в левом верхнем углу квадрата, является его порядковым номером. Отметим для ориентировки, что Лондон находится в 40-ом, а Будапешт в 58-ом квадратах.

Цифры, находящиеся в правом, нижнем углу квадратов, показывают, сколько из названных 9-ти алгоритмов дали неправильное решение. Буквы внутри квадрата отмечают, какие именно алгоритмы давали неправильное решение согласно следующему коду:

— от a) до g) буквы отмечают результаты линейного тета-алгоритма первоначально принятого вида при применении обучающей последовательности с первоначальным порядком следования точек (g)) и обучающими последовательностями различных видов «смешанных» порядков следования точек.

— h) относится к результатам, приведенным в строке 2 таблицы IV, полученным с применением алгоритма кусочно-линейного разделения.

— i) отмечает результаты приведенные в строке 2 таблицы III алгоритма ПДМ II.

Штрихованные квадраты отмечают те территории, на которых в действительности была гроза, а обрамленные жирной линией квадраты соответствуют тем территориям, где *большинство алгоритмов* дало *неверное решение.*

Общее количество таких квадратов 11, следовательно решение, принятое на основании решения большинства алгоритмов, является правильным в 87%-ах.

Пропорции решений алгоритмов:

9:0	в 45 случаях	Правильное решение: в 74 случаях (87%)
8:1	в 14 случаях	
7:2	в 7 случаях	
6:3	в 4 случаях	
5:4	в 3 случаях	
4:5	в 2 случаях	Неправильное решение: в 11 случаях (13%)
3:6	в 2 случаях	
2:7	в 1 случаях	
1:8	в 4 случаях	
0:9	в 2 случаях	

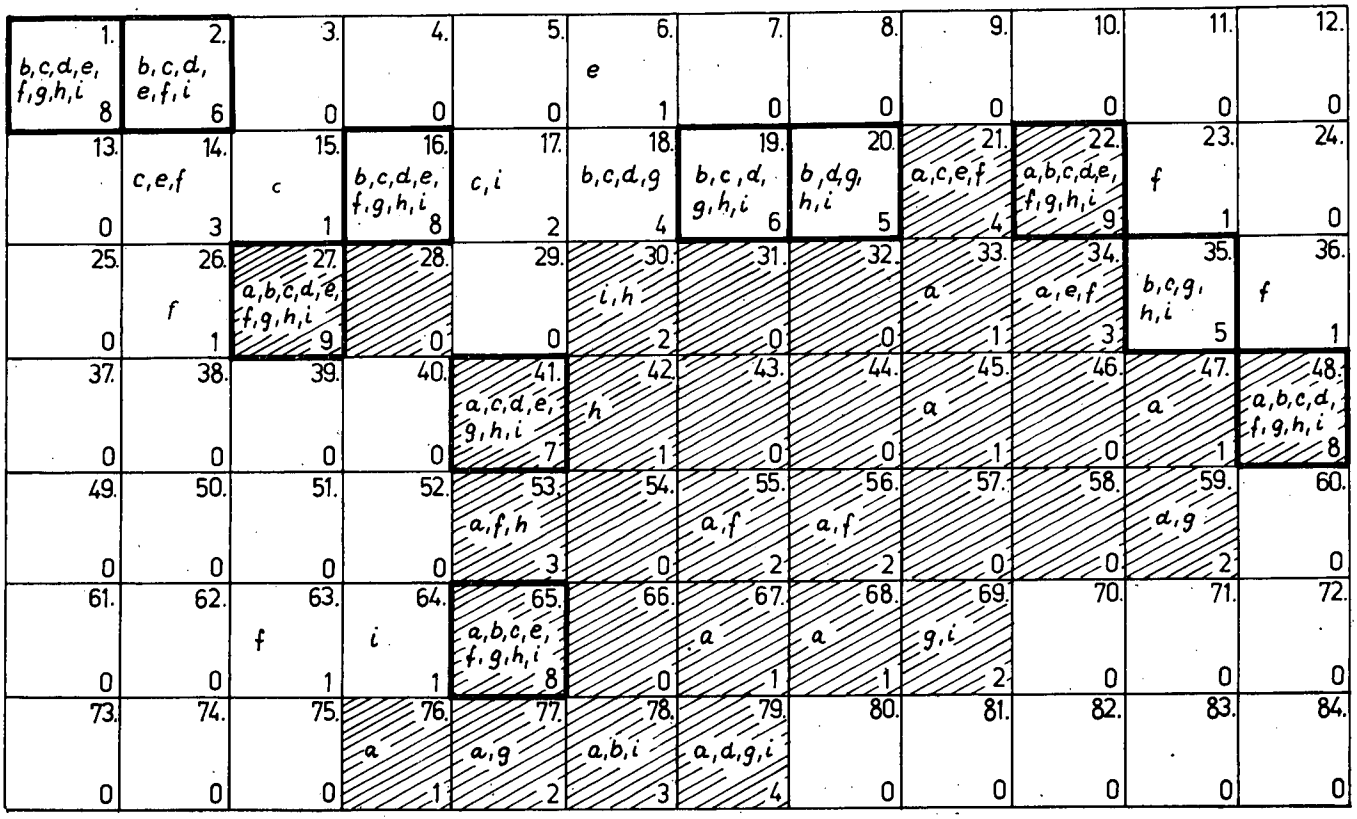

Рис. 5. Территориальное размещение неправильных решений

Наблюдения показали, что ошибки, сделанные таким образом, были совершены в тех областях, которые разместились на границе грозовой зоны или происходили в тех областях, где данные измерения не являлись достоверными. Следовательно, появившиеся ошибки главным образом происходят из несовершенства метеорологической модели, а также из неточности измеренных данных. На это указывает и то обстоятельство, что тогда, когда исключили из обучающей последовательности те 11 точек, для которых решения большинства алгоритмов дали неправильный результат, то применив обучающую последовательность из 73-х остальных точек, линейный тета-алгоритм потенциальных функций классифицировал с очень высокой точностью (см. таблицу VII) при однократном пропускании обучающей последовательности с точностью 94,6%, а при трехкратном пропускании — с точностью 97,4 %.

Таблица VI. Результат решений большинства девяти алгоритмов

H_1	H_2	H	Точность (%)
5	6	11	87

Таблица VII. Рост точности при отбрасывании не надежных точек обучающей последовательности

Кратность пропусканий	H_1	H_2	H	Точность (%)
1	3	1	4	94.6
3	2	0	2	97.4

Подытожив все вышеизложенное, можно сказать, что в случае изложенной выше метеорологической модели конвективная активность хорошо предсказывается с помощью показанных алгоритмов обучения. Надежность предсказывания превышает надежности синоптической практики; а кроме этого: главным достоинством процедур-применяющих алгоритм обучения по сравнению с синоптической практикой является то, что они дают объективный метод для предсказывания.

Можем рассчитывать на дальнейшее увеличение точности предсказывания путем увеличения длины обучающей последовательности.

В ближайшем будущем мы расширим круг исследований на предсказывания распределения во времени, — до сих пор нами исследовалось лишь пространственное распределение — а потом также и на предсказывание совместного пространственно-временного распределения. Из этих исследований рассчитываем получить такой опыт, который позволит усовершенствовать метеорологическую модель.

Application of learning algorithms in meteorology for the prediction of the convective activity

In this paper the application of some supervised learning algorithms in connection with meteorological prediction is described. The meteorological task, solved in cooperation of the Central Institute of Forecasting and the Research Institute for Telecommunication, related to the prediction of convective activity, or more popularly, to the prediction of rainstorms.

First a brief survey of the general theory of supervised learning is given, then the algorithms used in our experiments are discussed, and the computer programs realizing them are presented. For solving the above mentioned task three algorithms were used: the method of potential functions, introduced by Aizerman, Braverman, Rozonoer and others [6], the polynomial discriminant method of Specht [12] and the piecewise-linear separator of Nilsson [3] called committee machine. In the fol-

lowing the most important details of the meteorological model, thoroughly discussed in [22], is given. Finally the results and conclusions are discussed.

On the basis of the results it can be stated that the learning algorithms used in our experiments are able to predict the convective activity with high reliability, which exceeds the usual standards of more traditional techniques in meteorology.

* CENTRAL INSTITUTE OF FORECASTING, BUDAPEST, HUNGARY
II. KITAIBEL PÁL U. 1.

** RESEARCH INSTITUTE FOR TELECOMMUNICATION, BUDAPEST, HUNGARY
II. GÁBOR ÁRON U. 65.

Литература

[1] Васильев, В. И., Распознающие системы, *Наукова Думка*, Киев, 1969, стр. 292.
[2] Цыпкин, Я. З., Адаптация, обучение и самообучение в автоматических системах, *Автоматика и Телемеханика*, т. 27, 1966, стр. 23—61.
[3] Nilsson, N. J., *Learning machines*, New York, McGraw Hill, 1965.
[4] Csibi, S. & O. Gulyás, Задачи по распознаванию образов (на венгерском языке), TKI, *Szeminárium Közlemények*, 1969.
[5] Башкиров, О. А., Э. М. Браверман, И. Б. Мучник, Алгоритмы обучения распознаванию зрительных образов, основанные на использовании потенциальных функций, *Автоматика и Телемеханика*, т. 25, 1964, стр. 692—695.
[6] Айзерман, М. А., Э. М. Браверман, Л. И. Розоноер, Теоретические основы потенциальных функций в задаче об обучении автоматов разделению входных ситуаций на классы, *Автоматика и Телемеханика*, т. 25, 1964, стр. 917—936.
[7] Браверман, Э. М., О методе потенциальных функций, *Автоматика и Телемеханика*, т. 27, 1966, стр. 2205—2213.
[8] Gulyás, O., Метод потенциальных функций (на венгерском языке), TKI, *Szeminárium Közlemények*, 1969.
[9] Gulyás, O., On extended potential function type learning algorithms and their convergence rate, *Problems of Control and Information Theory*, v. 1, 1972, pp. 51—64.
[10] Hoffer, A., Распознавание образов с помощью алгоритма обучения, О выборе потенциальной функции, Дипломная работа (на венгерском языке), ELTE, TTK, Кафедра теории вероятностей, 1970.
[11] Molnár, L. & O. Gulyás, Программа обучения по методу потенциальных функций (на венгерском языке), TKI, *Szeminárium Közlemények*, 1969.
[12] Specht, D. F., Generation of polynomial discriminant functions for pattern recognition, *IEEE Trans.*, v. EC—16, 1967, pp. 308—319.
[13] Specht, D. F., Vectorcardiographic diagnosis using the polynomial discriminant method of pattern recognition, *IEEE Trans.*, v. BME—14, 1967, pp. 90—95.
[14] Meisel, W. S., Potential functions in mathematical pattern recognition, *IEEE Trans.*, v. C—18, 1969, pp. 911—918.
[15] Molnár, L., Метод распознавания образов полиномиальной дискриминацией (на венгерском языке), TKI, *Szeminárium Közlemények*, 1970.
[16] Esze, T. & J. Németh, Линейное и кусочно-линейное разделения (на венгерском языке), TKI, *Szeminárium Közlemények*, 1969.
[17] Widrow, B. & F. W. Smith, *Pattern recognizing control systems*, Washington, 1964, pp. 288—317.
[18] Сонечкин, Д. М., Математическая теория классификации и ее применение в метеорологии, *Метеорология и гидрология*, 1969, стр. 24—34.
[19] Сонечкин, Д. М., Об объективной классификации метеорологических явлений и ситуаций с помощью ЭВМ, *Метеорология и гидрология*, 1968, стр. 12—21.
[20] Баргов, Н. А., О классификации синоптических процессов, *Метеорология и гидрология*, 1969, стр. 3—12.
[21] Груза, Г. Б., Прогноз погоды и задача «распознавание образов» в кибернетике, *Метеорология и гидрология*, 1968, стр. 13—21.
[22] Szalay, G. & G. Götz, Метод для объективного анализа условий окружения конвективных процессов (на венгерском языке), *Időjárás*, v. 75, 1971, pp. 90—102.

[23] Szalay, G., L. Molnár, O. Gulyás, Применение алгоритмов обучения для метеорологического прогнозирования (на венгерском языке), *Междуинститутский доклад*, KEI—TKI, 1970.
[24] Szalay, G., L. Molnár, O. Gulyás, Использование алгоритмов обучения для метеорологического прогнозирования (на венгерском языке), Доклад на *VI-ой Венгерской конференции по автоматизации*, Будапешт, 1970.
[25] Molnár, L., The polynomial discriminant method of pattern recognition and its use in meteorology (paper presented at *The VIth Yugoslav International Symposium on Information Processing*, Bled, 1970).

(Поступило 8-ого марта 1971 г.)

INDEX—TARTALOM

Felelős szerkesztő és kiadó: Kalmár László
A kézirat a nyomdába érkezett: 1971. január hó
Megjelenés: 1972. június hó
Példányszám: 1000. Terjedelem: 7,68 (A/5) ív
Készült monószedéssel, íves magasnyomással,
az Msz 5601-90 és az Msz 5602-55 szabvány szerint
71-2921—Szegedi Nyomda