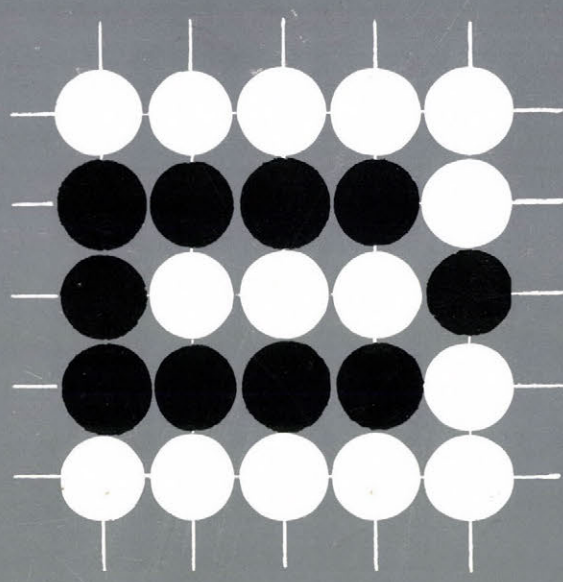
közlemények

29/1983

MTA Számítástechnikai és Automatizálási Kutató Intézet Budapest



MAGYAR TUDOMÁNYOS AKADÉMIA
SZÁMITÁSTECHNIKAI ÉS AUTOMATIZÁLÁSI KUTATÓ INTÉZETE

K Ö Z L E M É N Y E K

29/1983

Felelős kiadó:

VÁMOS TIBOR

ISBN 963 311 168 4
ISSN 0133-7459

Coopinvest - Főfotó 83-123o

CONTENT

TARTALOMJEGYZÉK

С О Д Е Р Ж А Н И Е

Közlemények 29/1983 7-20

INFORMATION SYSTEM FOR PLANNING A RESEARCH PROGRAM

Perfecto Dipotet
Institute of Mathematic, Cybernetic
and Computing Sciences
Habana, Cuba

INTRODUCTION

The developing countries face great problems in the efficient exploitation of their resources. In order to solve the top-priority problems related to social and economic development, it is necessary to concentrate to the maximum the efforts of all the organizations, mainly those of research intitutions. The solving of each one of these problems requires the implementation of complex, long-range research programs, with well-defined aims and the participation of several research and production organizations.

In this paper are presented:

Some considerations about information systems and their applications for planning purposes.

A methodology to develop the Programs Plan.

Algorithms and methods to determine the structure formed by the programs themes, and also to analize these structures in order to obtain formal criteria helping to the supervision and control of Program activities

INFORMATION SYSTEMS CONCEPTS AND DEFINITIONS

An information system is simply a means to an end, that is, it is established so as to provide a service or form of control for an object system (LANGEFOR, 74). "Object system" within the context of this paper refers to a complex research program, for which the type of information which is relevant, as well as the extent and detail of the investigations is determined by the programs authorities. The methodology described here is orientated to wards such authorities as a guide to the acquisition of the necessary information base helping them to plan and to manage the research program.

Plans are developed in order to solve certain problems. Within the process of problem solving three main phases (DUTTON,78) can be identified: problem finding, solution finding and solution implementation. Problem finding refers to the phase in which the problem is identified and specified. In our case it refers to the determination of Program tasks and operational objectives. Solution finding refers to the phase in which we look for several solutions (if possible) to the previously specified problem and then we select the optimal alternative. Solution implementation refers to the phase in which the selected solution is implemented (Figure 1).

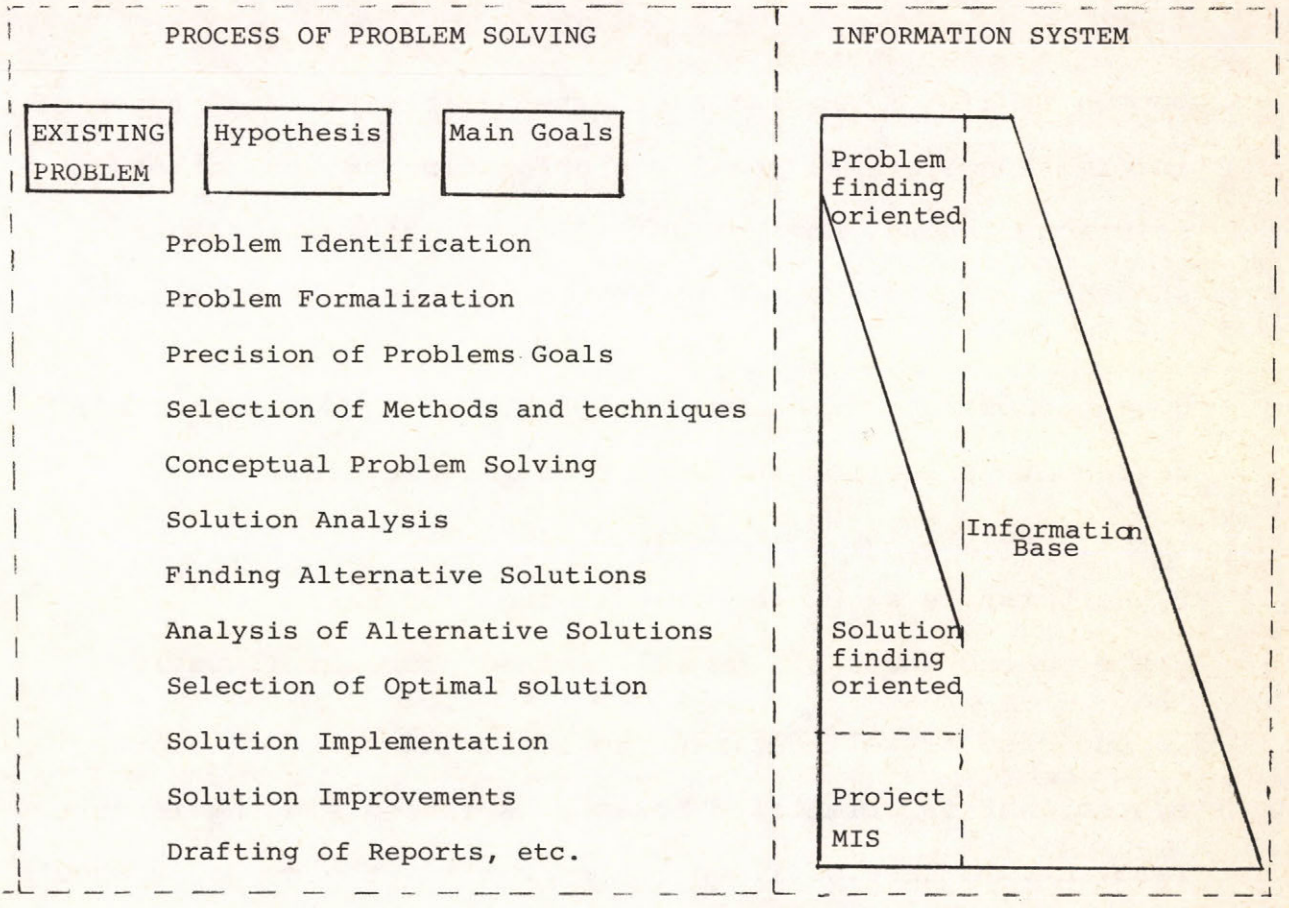


Figure 1

steps within the process of problem solving and its associated information system (DAENZER,78; BOSMAN,73; MAN.78)

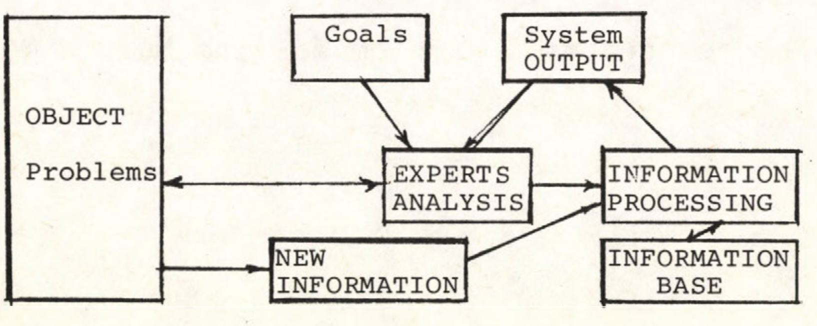


Figure 2

Problem Finding Information Sysytem for Ill-structured Problems (DIPOTET, 79).

Dutton (DUTTON, 78) distinguish two different ideal types of information systems: one for problem finding, the other for solution finding. However an efficient information system should be flexible enough to enable its use in both stages.

Bosman (BOSMAN, 73) classifies problems according to their degree of definition in three different levels:

- the level of the well-defined problems;
- the first level of the ill defined problems;
- the second level of the ill defined problems (fuzzy).

In the case of well-defined problems, the part of information system that is orientated towards solution finding is more relevant. In case of ill defined problems the emphasis should be on problem finding information and, generally, it is necessary to develop heuristic procedures for satisfying both stages. Figure 2 shows the initial stage of a problem finging information systems which, after gradual transformation (DIPOTET-79) can also be used as a problem solving one.

PLANNING THE RESEARCH PROGRAM

Production and service enterprises deal with concrete and well-defined tasks and subtasks. Scientific institutions deal with research themes. Thus, themes must be formulated for each institution from the activities and jobs belonging to the tasks of the program assigned to them.

Next we will present the procedures that must be carried out for collecting and processing the data that will enable us to derive the research plan.

Let a set J = (1,..., n) of research institutions belonging to one Organization which must carry out a research program P in a given time T.

The Scientific Council of the Organization divides the Program P into several sets of important tasks $P_1, P_2, \ldots, P_m$

Then, $P = (P_1, \ldots, P_m)$

We use the form given in Fig. 3 to obtain the listing of the institutions of the organization vs the tasks that they are going to undertake, respectively. For each $P_i \in P$; $i \in I = (1, \ldots, m)$ the Scientific Council establishes the deadline time $t_i \leq T$.

This deadline time t_i depends on several factors, but mainly on the will of the user and the domestic requirements of the Organization.

The performance of each task is divided into r subtasks. In our case these subtasks are the proposed steps for the information systems development in the process of problem solving (FIGURE 1). It means the following subtasks:

1) task identification
2) formalization of the task
3) precision of tasks goals
4) selection of methods and techniques
5) conceptual task solving
6) solution analysis
7) finding alternative solutions
8) analysis of alternative solutions
9) selection of optimal solution
10) solution implementation
11) solution improvements
12) drafting of reports, hand books and users manuals.

The form shown in Fig. 4 offers the listing of all institutions vs the subtasks where they will participate, respectively.

The Research Council build up, using these forms (see fig. 4), the matrix $A^j = \| a^j_{il} \|_{mxr}$ where $a^j_{il} \in (0,1)$; $i \in I = (1,\dots,m)$

$j \in J = (1,\dots,n)$

$l \in L = (1,\dots,r)$

$a^j_{il} = 1$ means that institute j participates in the carrying out of task i in the subtask l. The institution j lists the jobs and activities to be performed within the time interval t_i, for each case $a^j_{il} = 1$. The research themes are elaborated with the former list and the unification and generalization of other activities. The resources needed are established as well as the onset and completion dates.

With the information received from the themes of the Program for each task P_i, $P_i \in P$; $i \in I$; we establish its working stages

$(P_i(1), \ldots, P_i(t_i))$, where $P_i(t)$, $i \in I$; $t \leq t_i$; is the working stage in time t.

For each $P_i(i)$ we determine its resource vector

$r_i(t) = (r_{i1}(t), \ldots, r_{ik}(t), \ldots, r_{iq}(t))$

where $k \in K = (1, \ldots, q)$ is the resource number.

Then $\sum_{i \in I} r_{ik}(t) = R_k(t)$, resource requirements k in time t.

In those cases when $\sum a^j_{il} = 0$; $j \in J$; $i \in I$; $l \in L$; in other words, when none of the n institutions participate in the solution of one subtask a_{il}, it is necessary to find other institutions that would open new themes concerning subtask a_{il}

The form shown in Fig. 5 is used to list the research themes of the Program vs the subtask where they will take part, repectively. These three forms are the additional blanks that must be filled out in the organization and planning (ACC-80) of the Program. In Fig 5, $L_i, L_j, \ldots, L_z$CL.

Program P is then formed by a set W = (1,...,s) of research themes. Let $a^w_{il} \in (o,1)$ denote each element (subtask a_{il} related to theme w) in Fig. 5, $w \in W$; $i \in I$; $l \in L$ each theme w, $w \in W$ is then related to a set A^w of subtask a^w_{il}.

Then, $A^k \cap A^0 = A^{k0}$; k, $0 \in W$; is the set of subtasks where both themes k, 0 participate in simultaneously, cardinal N_{k0} of set A^{k0} is considered to indicate some relationship between themes k and 0; k, $0 \in W$.

The graph shown in fig. 6 is the matrix $N = \| N_{k0} \|_{sxs}$ formed by the cardinals of the intersections sets (See Fig. 6) will, of course, be symmetric in respect to the main diagonal.

In our case, we separate from the graph a subgraph, the maximum linked tree. Each node of the tree will be a theme. The value of the links will be given by their correspondents elements in matrix N, indicating some degree of relationship among the themes.

INSTITUTIONS / TASKS	1	2	....	n
P_1	+			+
P_2		+		
.				
.				
.				
.				
.				
P_m		+		+

Institutions vs Tasks

Fig. 3.

INSTITUTION j

SUBTASKS / TASKS	1	2	....	r	NAME OF PARTICIPANTS
P_1					
P_2					
.					
.					
.					
.					
P_m					

Institutions vs subtasks

THEMES \ TASKS	P_1	P_2	. . . .	P_m
10180230(1)	L_i	L_j		L_k
10280222(2)	L_0	L_p		L_u
. . . .				
(w)	L_x	L_y		L_z

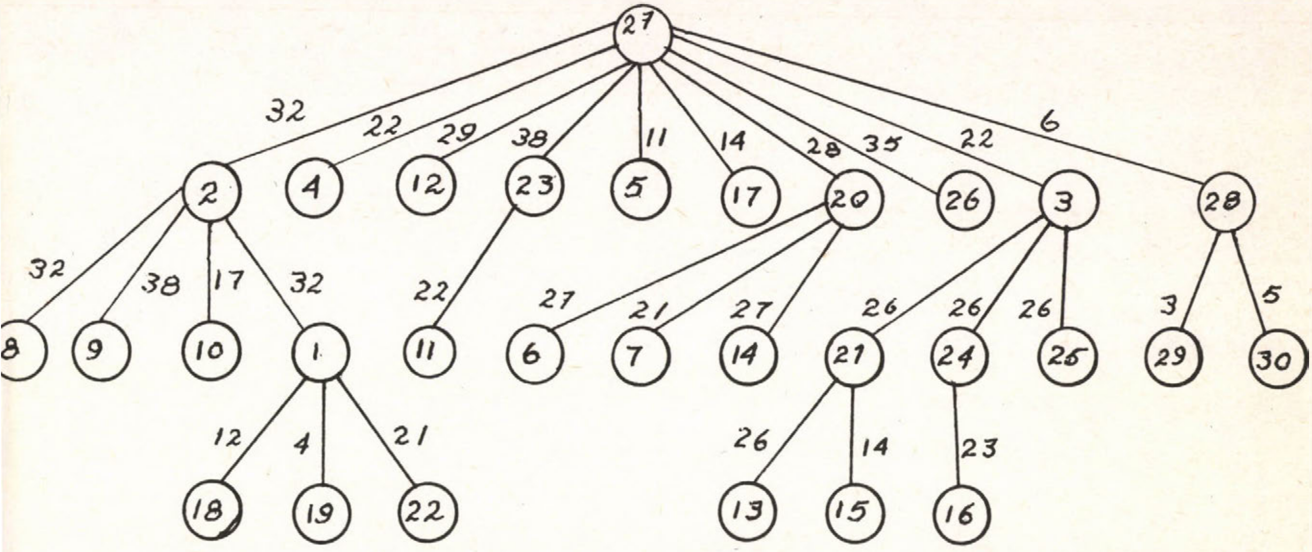
Themes vs subtasks

Fig. 5

THEMES \ THEMES	1	2	. . . .	s
1	N_{11}	N_{12}		N_{1s}
2	N_{21}	N_{22}		N_{2s}
. . . .				
s	N_{s1}	N_{s2}		N_{ss}

Matrix formed by the cardinals of the intersections sets

Fig. 6.



MAXIMUM LINKED TREE

Fig. 7.

In Fig. 7 the maximum linked tree is shown (DIPOTET, 80). In our case, this subgraph aids people in decision making concerning the management of the research program. For example:

- it is obvious that theme 27 in the subgraph is really a "bottleneck" and it is absolutely necessary to assure its resource allocation;
- the subtrees derived from nodes 2 and 3 respectively may be considered as subprograms to improve program management;
- themes 4, 12, 5, 17, 26 are practically isolated and it may be possible that works within them may begin in advance or be delayed (within time interval t_i) according to resource allocation problems.

What we have presented above are only examples. There are other applications of the tree and it is also possible (DIPOTET, 80) to derive other useful subgraphs from the graph shown in fig. 6.

REFERENCES

Dipotet P. 1980: Elaboracion de un sistem de Information para la Planificacion de las Actividades Investigative-economicas de la ACC. Informe tecnico.
IMACC. ACC. Habana.

Dipotet P. Jachaturov V. 1979: O. Niekotorix Vaprosax Razrabotki Informatsionnoi Sistemi pri Realizatsii Kompleksnij Program Isliedovanii.
VIEN KHOA HOC VIET NAM. HANOI.

Dipotet P. 1979: Elaboracion de un Sistema Hombre-Máquina para la Proyeccion del Desarrollo Socio-Economico de un Nuevo Territorio. Informe Tecnico. IMACC. Habana.

Jachaturov V. Figueroa M. 1979: Microlocalizacion de Objetos Socio-Economicos en Condiciones de Information Difusa.
Informe Tecnico. IMACC. Habana.

Figueroa M. 1978: Programas en FORTRAN para la Microlocalizacion de ESBEC en la Isla de la Juventud.
MACC. ACC. Habana.

CC. 1980: Procedimientos para la Planificacion de Temas de Investigacion.
PACI. ACC. Habana.

Dipotet P. 1981: Planning a Research Program to Develop an Agricultural Region. IFIP Conference on "Computer Applications in Food Production and Agricultural Engineering".
Habana, October 1981.

Langefors B. 1974: Information Systems. Proceedings of the IFIP Congress Information Processing 74, Stockholm, North Holland Publishing Company.

Dutton, W H, Kraemer, K L, 1978: Management utilization of Computers in American Local Governments. Communications ACM, Vol 21. No 3.

Bosman, A. 1973: IEE-structured Problems and Information Systems. Seminar on Strategic Problem Solving. University of Groningen, the Netherlands. 1973.

Daenzer, W F. 1976: Systems Engineering, Leitfaden zur Methodischen Durch führung Umfam greicher Planungsvorhaben Peter Hanstein Verlag GmbH, Köln/Verlag Industrielle Organisation, Zürich

Összefoglalás

Kutatási program tervező információs rendszer

A cikk információs rendszerekkel és tervezési célu alkalmazásaikkal foglalkozik. A szerző a program tervezés módszertani kérdéseitől kezdve tárgyalja a problémát. Algoritmusokat és módszereket ad a program témák strukturájának meghatározására, valamint elemzi ezeket a strukturákat a-ból a célból, hogy segitse a program irányitását és ellenőrzését.

Информационная система для проектирования научного исследования

Перфекто Дипотет

В стате описываются некоторые информационные системы в связи с их применяемостью для проектирования научного исследования. Дается общая методика проектирования. Алгоритмы и методы прелагаются для определения структуры программ исследований, а также дается анализ этих структур для управления программами.

Közlemények 29/1983 21-30

A QUERY SUBSYSTEM FOR A RELATIONAL DATABASE SYSTEM

LUCIANO GARCIA

MIGUEL KATRIB

EDUARDO QUESADA

Universidad de la Habana
Centro Nacional de Investigaciones Cientificas
CUBA

INTRODUCTION

The present article describes INTERROG, the query subsystem of LORKA, a relational database system (dbs) designed and implemented by the authors [3,4,5]. As it is - well known, relational models [1] accomplish in the most optimal way one of the more important constructive requirements in the design and implementation of dbs: independence of the logical structure of requests from the storage organization of data. It is considered that this independence should be achieved in the sense freeing the nonprogrammer user of all the machine-dependent thecnicalities for accessing data unnecessary for the logical structuralization of requests. Besides that, relational models provide as no other dos model an adequate framework for developing natural language interface and enhancing the dbs with a deductive capacity to deal with implicit information.

In the relational model the information of a database is arranged into *relations*, each one containing a collection of *tuples*. A relation can be thought of as a table *Fig. 1*, with columns named after the different attributes which compose the relation and rows of attribute values representing the tuples.

DOCUMENTS

TITLE	AUTHOR	DESCRIPTORS	REF-NO
T1	A1	(D1 D3 D4 D6)	RN1
T2	A2	(D3 D5 D6)	RN2
T3	A3	(D1 D5)	RN3
T4	A4	(D3 D4 D6)	RN4

BORROWERS

NAME	ADDR	CARD-NO
N1	ADDR1	0215
N2	ADDR2	0236
N3	ADDR3	0725
N4	ADDR4	0630

LOANS

REF-NO	CARD-NO	DATA
RN1	0236	1 2 83
RN3	0725	4 1 83
RN2	0215	11 2 83
RN4	0630	20 4 82

Fig. 1.

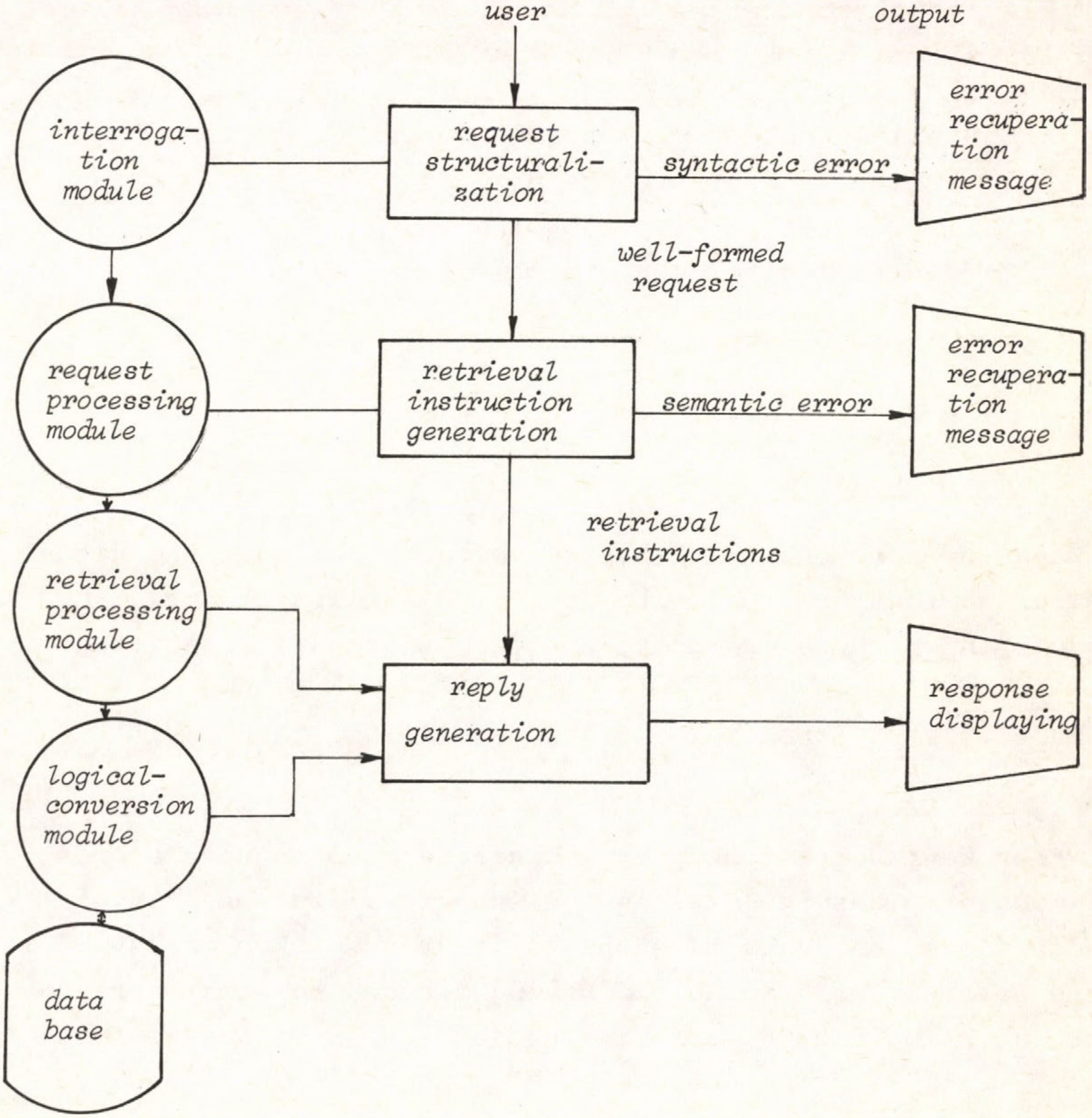

user
output
interrogation module
request structuralization
syntactic error
error recuperation message
well-formed request
request processing module
retrieval instruction generation
semantic error
error recuperation message
retrieval instructions
retrieval processing module
reply generation
response displaying
logical-conversion module
data base

Fig. 2

The general design of INTERROG is depicted in *Fig. 2.* It is composed of four fundamental modules:

The request expressing module.
The request processing module.
The retrieval processing module.
The logical-conversion module.

which are going now to be explained in some length.

The request expressing module

When the user enters the system after a wellcome the visual terminal depicts the following (only the sublist of query commands in which we are going to concentrate are - depicted)

P(ROJECTION) S(ELECTION) B(OTH PROJECTION SELECTION) (1)

After keying the appropriated character the request expressing module guides the user in the structuralization of his request and at the same time checks it step by step for syntactic errors. For example; if having decided to enter a request which implies to perform a projection after a selection over two relations the user keys "B". The following lines are depicted one after another (user reponses to each one are include for purposes of illustration and are associated with the examples given in *Fig. 1.*):

Enter the atribute list

*(ADDR NAME? (2)

Enter the selection formula

*(= 1-2-83 DATE? (3)

Enter the relation list

*(BORROWERS LOANS? (4)

As can be appreciated the user constructs are LISP-like lists. The projection-list (2) is a list composed of an arbitrary numbers of atoms which denote atributes of the relations involved. The order in which these names appear in the projection list is arbitrarily determined by the users and reflects the order in which he wants the associated columns of values of the corresponding tuples to be displayed.

The formula-list (3) represents a formula of an applied first-order predicate calculus built with the propositional operators NOT, OR and AND, quantifier-free, with variables (attribute names) and constants (attribute values) as terms.

The elemental formulae are of the form (R T1 T2) where R is either an arithmetical relational operator (<,>,=,<>, <=,>=), either a set relational operator (MEMBER, SUBSET, DISJOINT) and T1 and T2 are terms. The formula is built in prefix notation and the operators AND and OR may have an arbitrary numbers of operands.

A relation-list (4) is composed of one or more atoms which denote the relations involved in the request. When there are more than one relation names in the list, INTERROG performs automatically the natural join operation whenever possible.

The character "?" at the end of any list is for terminating the list, and it is left in this way to the request expressing module to supply all the end parentheses needed.

If a syntactical error is committed through (2) - (4) the user receives an appropriate message and he has the option of recuperating it or of making a fresh start from (1).

The request processing module

When the request is well-formed it is delivered to the request processing module which first checks it for semantic errors. A semantic error is committed, for example, when a projection-list or a selection-formula contains an atom which is not an attribute name or an attribute value. The request processing module detects all kind of semantic - errors and sends appropriate messages to the user who is obliged to make a fresh start.

After passing successfully the semantical test, the request is finally accepted as input data by the request - processing module which proceeds to generate from it the retrieval instructions. This means that there are not *ad hoc built-in* retrieval instructions. As we have already explained the system has to cope with any projection-list or formula and the latter (see more examples further) can have a very complex structure. The request processing - module is provided with procedures which transform projection lists and formulae in retrieval instruction *lists* which in turn are to be applied as functions to the relations. The same thing occurs if the request implies the performance of a natural join of two relations: this can have an arbitrary number of attributes in common and the module generates the appropriate retrieval selection formula and projection list for the execution of the join.

The retrieval and conversion modules

The retrieval instruction lists are passed on the retrieval module where they are going to be applied as LISP functions to the appropriate relations. The retrieval process takes place in the computer working memory in a LISP-like manner. To accomplish that, the relations are brought to the working memory via the conversion module whose task is to transform the physical data structure of the relations in the suitable for processing LISP--like list structure in the working memory. In this way both retrieval instructions and relations are LISP-like listsand the process of aplying the firsts to the seconds is carried out through the LISP_EVAL function whose output is the response to the request.

It is in this way that the system succeeds in extablishing a complete independence of the logical reception and processing of requests from the physical organization of the dbs.

After being retrieved, the results are visually displayed with printing as an option. The displaying takes place inmediately to the evaluation of each tuples of the relations involved. In our example if a tuple results from the natural join of the relations and the retrieval instruction formula evaluates it to true, it is immediately displayed and in case there is a projection list, only the row of values of the corresponding attributes is displayed. The answer is displayed with additional counting and average.

Additional features of INTERROG

The query language of INTERROG has the expressive power of the relationsl calculus [2]. Nevertheless the structuralization of a request by the user following the "menu" does not required the mathematics of the relational calculus. - The user has only to think in terms of attributes associated to logical formulae and relations. Thus, the request structuralization is non-procedural and it is completely left to INTERROG to generate an *optimal* retrieval procedure: it is not the user but INTERROG the one which optimizes a Join taking advantage of an appropriate selection formula or of ordered-key relations.

The relation attributes have ranges which can take the following classes of types: scalar types like *nominal* (string), *literal* (finetely enumerative values), *integer*, *longinteger*, *real* and structured types like *date* (record of the form day--month-year) and *list* (dynamics sets of scalar and date types). It is possible to perform operation over date and list values. From a date we can single out the day, the month or the year:

```
(AND(=AUTHOR A3)(OR(=(YEAR DATE)83) (=(YEAR DATE)82?
```

with list values we can perform the already mentioned relational set-theoretical operations:

```
(AND(=AUTHOR A1)(SUBSET(D3 D4) DESCRIPTORS?

(OR(AND(=(YEAR DATE)82)(DISJOINT(D2 D1)DESCRIPTORS))

(AND(=(YEAR DATE)83)(SUBSET(D1 D5)DESCRIPTORS?
```

Others functions are included:

```
F(ILE
```

to create a PASCAL file from the request,

```
T(EMPORARY
```

to create a temporary relation from the request, and

```
M(AX        M(IN
```

to obtain the maximum minimum value in an attribute field of the tuples satisfaying the request.

CONCLUSIONS

INTERROG is the query subsystem of the first prototype of LORKA which was implemented in UCSD-PASCAL for cuban mini and microcomputers. This system are being used in Biomedicine in the Cuban National Center of Scientific Researchs. Further versions of LORKA already under development will enhance this prototype with natural language interface and deductive capacity.

REFERENCES

[1] Codd, E.G. "A relational model of data for large shared data banks". CACM V 13, N 6 1970 p. 377-387

[2] Codd, E.F. "Relational completeness of data base sublanguages". Data Base Systems Rustin R. editor. Englewood Cliffs, N.J. 1972 p. 65-98.

[3] Garcia, L. "Un sistema de interrogación de una base de datos fundado en la estructura de lista". 3er Evento Cientifico de la Universidad de la Habana, 1981.

[4] Garcia, L., Katrib, M., Quesada, E. "Realización del Sistema de interrogación del sistema de base de datos relacional LORKA". ler Congreso Nacional de Matematica, Academia de Ciencias, Ciudad Habana, Cuba.

[5] Katrib, M. "Realización del lenguaje LISP en la configuración CID-300-10". 3er Evento Cientifico de la Universidad de la Habana, 1981.

A relációs adatbázis-rendszernek egy lekérdezési alrendszere

L. Garcia, M. Katrib, E. Quesada

Összefoglalás

A cikk ismerteti a szerzők által kifejlesztett LORKA elnevezésü relációs adatbázis-rendszernek lekérdezési alrendszerét.

ПОДСИСТЕМА ЗАПРОСА РЕЛЯЦИОННОЙ СИСТЕМЫ БАЗЫ ДАННЫХ

Л. Гарция, М. Катриб, Е. Квесада

Резюме

В статье описывается подсистема запроса реляционной системы базы данных LORKA разработанной авторами.

Közlemények 29/1983 31-39

CONGRUENCES ON CLOSED SETS OF SELF-DUAL FUNCTIONS IN MANY-VALUED LOGICS AND ON CLOSED SETS OF LINEAR FUNCTIONS IN PRIME-VALIED LOGICS

V.V. Gorlov and D. Lau

A congruence on a closed set M in the k-valued logic P_k, $k\geq 2$, is an equivalence relation on M which is compatible with the operations (permutation and identification of variables, addition of fictitious variables and substitution) of P_k.

In this paper we prove that the congurences on closed sets of self-dual functions of P_k are determined by congruences on closed sets of non-self-dual functions.

Moreover, we determine all congruences on the closed sets of linear functions (see [1] and [8]) in prime-valued logic.

1. Basic concepts

Let E_k denote the set $\{0,1,\ldots,k-1\}$, where $k\geq 2$. Let P_k^n denote the set of all functions $f^n:E_k^n\to E_k$ $(n\geq 1)$ and put $P_k = \bigcup_{n\geq 1} P_k^n$. If there is no danger of confusion, the superscript n of the function f^n is omitted.

The set of all function of P_k^1 having exactly 1 values we denote by $P_k^{[1]}$.

The functions $e_i^n\in P_k$ $(1\leq i\leq n)$ defined by $e_i^n(x_1,\ldots,x_n)=x_i$ are called projections. The n-ary constant function with value a is denoted by c_a^n.

The operations on P_k are ζ, τ, Δ, ∇, $*$, which are defined by

$$(\zeta f)(x_1,\ldots,x_n)= f(x_2,x_3,\ldots,x_n,x_1)$$
$$(\tau f)(x_1,\ldots,x_n)= f(x_2,x_1,x_3,\ldots,x_n)$$
$$(\Delta f)(x_1,\ldots,x_{n-1})= f(x_1,x_1,x_2,\ldots,x_{n-1})$$
$$(\nabla f)(x_1,\ldots,x_{n+1})= f(x_2,x_3,\ldots,x_{n+1})$$

$$(f*g)(x_1,x_2,\ldots,x_{n+m-1}) = f(g(x_1,\ldots,x_m),x_{m+1},\ldots,x_{n+m-1}),$$

where f^n, $g^m \in P_k$ (see [5] or [6]).

Superpositions over the set A_P_k are functions obtained from A by using the operations $\zeta,\tau,\Delta,\nabla,*$ finitely many times. The closure $[A]$ of a set $A \subseteq P_k$ is the set of all superpositions over A. A set A is said to be closed if $[A]=A$.

An equivalence relation κ on a closed set A is called a congruence on A iff $f \sim g\ (\kappa)$, $s \sim t\ (\kappa)$ imply $f*s \sim g*t\ (\kappa)$ and $\alpha f \sim \alpha g\ (\kappa)$ for all $\alpha \in \{\zeta,\tau,\Delta,\nabla\}$ and for all $f,g,s,t \in A$. A.I.Mal'cev showed in |5| that every closed set A_P_k has three trivial congruences κ_0, κ_a and κ_1:

$$f \sim g\ (\kappa_0) :\Longleftrightarrow f=g \wedge \{f,g\} \subseteq A$$

$$f^n \sim g^m\ (\kappa_a) :\Longleftrightarrow n=m \wedge \{f,g\} \subseteq A$$

$$f \sim g\ (\kappa_1) :\Longleftrightarrow \{f,g\} \subseteq A.$$

Let κ and κ' be two congruences on A. We write $\kappa \subseteq \kappa'$ iff $f \sim g\ (\kappa)$ implies $f \sim g\ (\kappa')$ for all f, $g \in A$.

For the other undefined notations we refer the reader to [1]-[8], particularly to [6].

For the proofs of our theorems we need the following lemma which is well known.

1.1 LEMMA ([2]). Let A be a closed set in P_k containing the projections. If κ is a congruence on A with $\kappa \not\subseteq \kappa_a$, then $\kappa = \kappa_1$.

2. Congruences on closed sets of self-dual functions of P_k

Let $s(x)=x+1$ mod k and let S be the set of all functions of P_k preserving the relation $\{(a,s(a)) \mid a \in E_k\}$. The functions of S are called self-dual functions. If k is a prime number then S is a maximal closed set of P_k ([7]). In [3] for a maximal closed set of self-dual functions it was proved that a function $f^n \in P_k$ is a function of S iff there exists a function $F^{n-1} \in P_k$ with the property

$$f(\tilde{x})= \sum_{i=0}^{k-1} j_i(x_1)\cdot s^i(F(s^{k-i}(x_2),\dots,x^{k-i}(x_n)))\ mod\ k, \quad (1)$$

where $s^i(x)$: $x+i\ mod\ k$; $j_i(x)= \begin{cases} 1 & if\ x\ i \\ 0 & otherwise \end{cases}$ and

$F(x_1,\dots,x_{n-1})=f(0,x_1,\dots,x_{n-1})$. The proof in [3] does not use the property that k is prime. Therefore (1) is right for every k. Because of this property we can define a bijective mapping α of $S(\_P_k)$ onto $P_k^!: \underset{n\geq 0}{} \{f^n:E_k^n \to E_k\}$ as follows:

$\alpha : f \to F.$

2.1 LEMMA. The mapping α has the following properties:

(i) For the operations $\hat{\zeta},\hat{\tau},\hat{\Delta},\hat{\nabla},\hat{*}$ defined by

$(\hat{\zeta}f)(x_1,\dots,x_n)=f(x_1,x_3,x_4,\dots,x_n,x_2)$
$(\hat{\tau}f)(x_1,\dots,x_n)=f(x_1,x_3,x_2,x_4,\dots,x_n)$
$(\hat{\Delta}f)(x_1,\dots,x_{n-1})=f(x_1,x_2,x_2,x_3,\dots,x_{n-1})$
$(\hat{\nabla}f)(x_1,\dots,x_{n+1})=f(x_1,x_3,x_4,\dots,x_n)$
$(f\hat{*}g)(x_1,\dots,x_{n+m-2})=f(x_1,g(x_1,x_2,\dots,x_m),x_{m+1},\dots,x_{n+m-2})$

is $\alpha(\hat{\zeta}f)=\zeta F$, $\alpha(\hat{\tau}f)=\tau F$, $\alpha(\hat{\Delta}f)=\Delta F$, $\alpha(\hat{\nabla}f)=\nabla F$ and $\alpha(f\hat{*}g)=F*G$, i.e. the algebra $<S;\hat{\zeta},\hat{\tau},\hat{\Delta},\hat{\nabla},\hat{*}>$ is isomorphic to the algebra $<P_k^!;\zeta,\iota,\Delta,\nabla,*>$.

(ii) For every closed subset $A(\neq\emptyset)$ of S is $\alpha(A)$ a closed set, $\alpha(A)\not\subseteq S$ and $A\subseteq\alpha(A)$.

PROOF. (i) is easy to check.
Let A be a closed subset of S. Then by (i) we get that $\alpha(A)$ is likewise a closed set. Assume $\alpha(A)\subseteq S$. Then

$$F(x_2,\dots,x_n)=s^i(F(s^{k-i}(x_2),\dots,s^{k-i}(x_n)))$$

for $i=0,1,\ldots,k-1$ and for every $f^n \in A$ (see [3]). Thus by (1) we get that the variable x_1 in every function $f \in A$ is fictitious. however, this is not possible. Therefore is $\alpha(A) \not\subseteq S$.
Let $f \in A$. Then is $\nabla f \in A$ and therefore $\alpha(\nabla f) = f \in \alpha(A)$, i.e. $A \subseteq \alpha(A)$.

2.2 THEOREM. Let A be a closed subset of S, κ a congruence on A and let $\alpha(\kappa)$ defined by

$$F \sim G(\alpha(\kappa)) :\Longleftrightarrow \alpha^{-1} F \sim \alpha^{-1} G \ (\kappa)$$

an equivalence relation on $\alpha(A)$. Then

(i) $\alpha(\kappa)$ is a congruence on $\alpha(A)$ and

(ii) $\alpha(\kappa)_{/A} =$, i.e. the congruences on A we can get by restriction of the congruences on $\alpha(A)$ to A.

PROOF. (i). Since κ is a congruence on A κ is also compatible with the operations $\hat{\zeta}, \hat{\tau}, \hat{\Delta}, \hat{\nabla}, \hat{*}$. Then by 2.1 (i) follows that $\alpha(\kappa)$ is a congruence on $\alpha(A)$.
(ii) By 2.1 (ii) is $A \subset \alpha(A)$ and therefore $\alpha(\kappa)_{/A}$ is a congruence on A. Let f and g be functions of A. If $f \sim g\ (\kappa)$ then $\nabla f \sim \nabla g\ (\kappa)$ and by definition of $\alpha(\kappa)$ is $\alpha(\nabla f) = f \sim g = \alpha(\nabla g)(\alpha(\kappa)_{/A})$, i.e. $\subseteq \alpha(\kappa)_{/A}$. If $f \sim g\ (\alpha(\kappa)_{/A})$ then by definition of $\alpha(\kappa)$ we get that $\alpha^{-1} f \sim \alpha^{-1} g\ (\kappa)$. Since f and g are functions of S is $\alpha^{-1} f = \nabla f$ and $\alpha^{-1} g = \nabla g$. Therefore is $\nabla f \sim \nabla g\ (\kappa)$ and $\Delta(\nabla f) = f \sim g = \Delta(\nabla g)\ (\kappa)$, i.e. $\alpha(\kappa)_{/A} \subseteq \kappa$. Thus $\alpha(\kappa)_{/A} =$.

3. Congruences on some closed subsets of $[P_k^1]$

In this section we prove a theorem which we need for the determination of the congruences on the closed subsets of linear functions.

Let $C \subseteq P_k^{[1]}$, G a subgroup os $\langle P_k^{[k]};*\rangle$, where the functions of G preserve the set C, U a normal subroup of the group G and let μ be an equivalence relation on C which is preserved by the functions of G.

It is easy to check that the equivalence relation $^{U,\mu}$ on $[G\cup C]$ defined by

$$f^n \sim g^m (\kappa^{U,\mu}) :\Longleftrightarrow n = m \wedge (\exists i\ \exists f', g' \in G\cup C:\ f(\tilde{x}) = f'(x_i) \wedge g(\tilde{x}) = g'(x_i) \wedge (f' * U = g' * U \vee f' \sim g'(\mu)))$$

is a congruence on $[G\cup C]$.

3.1 THEOREM. Let G be a subgroup of $\langle P_k^{[k]}; * \rangle$, $C \subseteq P_k^{[1]}$ and $G \subseteq Pol\ C$. Then exactly on $[G\cup C]$ there exist the congruences κ_0, κ_a, κ_1 and congruences of the type $\kappa^{U,\mu}$, where U is a normal subgroup of G and μ is an equivalence relation on G which is preserved by the functions of G.

PROOF. Let κ be a congruence on $[G\cup C]$ and $\kappa \neq \kappa_1$. Then by 1.1 is $\kappa \subseteq \kappa_a$. We have to ditinguish the following cases:

Case 1: There exist κ-congruent functions f^n and g^n with $\Delta^{n-1} f \in G$ and $\Delta^{n-1} g \in C$.

Then is $\Delta^{n-1} f \sim \Delta^{n-1} g(\kappa)$. Thus $e_1^1 \sim \Delta^{n-1} g =: c_a(\kappa)$, $a \in E_k$.

By this we have for every function $h^m \in [G\cup C]$:

$e_1^1 * h = h \sim c_a^m = c_a * h(\kappa)$, i.e. $\kappa = \kappa_a$.

Case 2: There exist κ-congruent functions f^n and g^n with $f(x_1,\dots,x_n) = f'(x_i), g(x_1,\dots,x_n) = g'(x_j)$, $\{f', g'\} \subseteq G$ and $i \neq j$.

Without loss of generality we can assume that $i=1$ and $j=2$. The inverse functions of f' and g' we denote by f'' and g'', respectively. Then we have

$f(f''(x_1), g''(x_2), x_2, \dots, x_2) = e_1^2(x_1, x_2)$

$\sim g(f''(x_1), g''(x_2), x_2, \dots, x_2) = e_2^2(x_1, x_2)\ (\kappa)$. Therefore is

$e_1^2(s(\tilde{x}), t(\tilde{x})) = s(\tilde{x}) \sim t(\tilde{x}) = e_2^2(s(\tilde{x}), t(\tilde{x}))\ (\kappa)$ for every s

and t of $[G\cup C]^m$, $m \geq 1$, i.e. $\kappa = \kappa_a$.

Case 3: Two n-ary functions f and g are κ-congruent if and only if either $\{f,g\}\subseteq[C]$ or there exist an i and $f',g'\in G$ with $f(x_1,\ldots,x_n)=f'(x_i)$, $g(x_1,\ldots,x_n)=g'(x_i)$.

In this case the congruence κ is exactly determined by $\kappa_{/G}$ and $\kappa_{/C}$.

As we know, the congruence on a group G are determined by a normal group U of G and $f\sim g(\ )$ iff $f*U=g*U$ for all f, $g\in G$.

Obviously, $_{/C}$ is an equivalence relation on C which is preserved by every function of G.

Thus $=\ ^{U},\ /C$.

4. Congruences on closed sets of linear functions in prime-valued logics

Let p be a fixed prime number. L denote the set of all linear functions over $<E_p;\ +,\ \cdot\ mod\ p>$ in P_p, i.e.

$$L:=\bigcup_{n\geq 1}\{f^n\in P_p\,|\,\exists a_i: f(\tilde{x})=a_0+\sum_{i=1}^{n}a_i\cdot x_i\ mod\ p\}.$$

In |1| it was proved that L has exactly the following closed subsets:

$$L\cap S=\bigcup_{n\geq 1}\{f^n\in L\,|\,a_1+a_2+\ldots+a_n=1\ mod\ p\},$$

$$L\cap Pol(a)=\bigcup_{n\geq 1}\{f^n\in P_p\,|\,\exists a_i: f(\tilde{x})=a+\sum_{i=1}^{n}a_i\cdot(x_1-a)\},$$

$a\in E_p$,

$L\cap S\cap Pol(0)$ and closed subsets A with $A\subseteq[L^1]$.

If $A\subseteq[L^{[1]}]$, then it is easy to see that the closed set A has only congruences of the type κ^{μ} and of the type κ_a^{μ} defined by

$$f^n \sim g^m(\kappa^{\mu}) :\Longleftrightarrow \Delta^{n-1}f \sim \Delta^{m-1}g\ (\mu) \quad \text{and}$$

$$f^n \sim g^m(\kappa_a^{\mu}) :\Longleftrightarrow n=m \wedge \Delta^{n-1}f \sim \Delta^{n-1}g(\mu),$$

where μ is an any equivalence relation on A^1.
If $A \nsubseteq [L^{[1]}]$ and $A \subseteq [L^1]$ then the congruences on A follow by theorem 3.1.

We denote by κ_c an equivalence relation defined by

$$f^n \sim g^m(\kappa_c) :\Longleftrightarrow n=m \wedge (\exists a : f(\tilde{x})=a+g(\tilde{x}) \bmod p).$$

Obviously, κ_c is a congruence on L. We will show that κ_c is the only nontrivial congruence on $A \subseteq L$ for $A \nsubseteq [L^1]$.

4.1 THEOREM ([4]). κ_0, κ_c, κ_a and κ_1 are the only congruences on L.

4.2 THEOREM. κ_0, κ_a and κ_1 are the only congruences on $L \cap Pol(a)$ for every $a \in E_p$.

PROOF. Clearly, the closed sets $L \cap Pol(a)$, $a \in E_p$, are mutually isomorphic. Therefore we can assume that $a=0$. Let κ be a congruence on $L \cap Pol(0)$. The following two cases are possible:

Case 1: $\kappa \nsubseteq \kappa_a$.
By 1.1 is $\kappa = \kappa_1$.
Case 2: $\kappa_0 \subset \kappa \subseteq \kappa_a$.

Then there exist κ-congruent functions f^n, g^n and an n-tuple $\tilde{a}=(a_1,\dots,a_n)$ with $f(\tilde{a}) \neq g(\tilde{a})$. Therefore is
$f(a_1 \cdot x, a_2 \cdot x, \dots, a_n \cdot x) =: a \cdot x \sim b \cdot x := g(a_1 \cdot x, a_2 \cdot x, \dots, a_n \cdot x)\ (\kappa)$,
where $a \neq b$.

The functions $h(x,y)=x-y \bmod p$ and $t(x)=(a-b)^{-1} \cdot x$ belong to $L \cap Pol(0)$. Thus we get $h(ax,ax)=c_0 \sim h(ax,bx) := h'(x)\ (\ \ )$ and $c_0^1 = t * c_0^1 \sim t * h' = e_1^1\ (\kappa)$. This implies that $c_0^1 * r^m = c_0^m \sim r = e_1^1 * r\ (\kappa)$ for every $r^m \in L \cap Pol(0)$, $m \geq 1$. Therefore $\kappa = \kappa_a$.

4.3 THEOREM. κ_0, κ_c, κ_a and κ_1 are the only congruences on $L\cap S$.

PROOF. Obviously, $\alpha(L\cap S)=L$. Therefore, using 2.2 and 4.1 we have the the theorem.

4.4 THEOREM. κ_0, κ_a and κ_1 are the only congruences on $L\cap S\cap Pol(0)$.

PROOF. The theorem follows from $\alpha(L\cap S\cap Pol(0))=L\cap Pol(0)$, 2.2 and 4.2.

We also remark that the structure of the congruences becomes more complicated, if k is not a prime number. If k is square-free, then follows by [8] and by [4] (theorem 3.6) that every closed subset of L has only a finite number of congruences. But, if k is not square-free then there exist closed subsets of L with an infinite number of congruences. Finally we give an example for a closed subset with a such a property.

Let $Z:=\bigcup_{n\geq 1}\{f^n\in P_4 \mid \exists a_i\in\{0,2\}: f(\tilde{x})=\sum_{i=1}^{n} a_i\cdot x_i \bmod p\}$, let $r(f)$ be the number of the non-fictitious variables of the function f.

Further let χ_i be an equivalence relation defined by

$$f^n\sim g^m(\chi_i):\Longleftrightarrow f=g\vee(n=m\wedge r(f)\leq i\wedge r(g)\leq i),$$

$i=1,2,\ldots$. If is easy to prove that χ_i for all $i\geq 1$ is a congruence on Z of P_4.

REFERENCES

[1] J. Bagyinszki; J. Demetrovics, The lattice of linear classes in prime-valued logics, Banach Center Publications, PWN, 8 (1979).

[2] V.V. Gorlov, On congruences on closed Post classes (in Russian), Mat. Zametki, 13 (1973), 725-734.

[3] D. Lau, Bestimmung der Ordnung maximaler Klassen von Funktionen der k-wetigen Logik. Zeitschr. f. math. Logik ung Grundl. der Math., Bd. 24 (1978), 79-96.

[4] D. Lau, Congruences on closed sets of k-valued logic. Colloquia Mathematica Soc. J. Bolyai, Vol. 28. 417-440.

[5] A.I. Mal'cev, Iterative algebras and Post's varieties (in Russian), Algebra i Logika, 5 (1966), 5-24.

[6] R. Pöschel; L.A. Kaluznin, Funktionen- und Relationalgebren, DVW, Berlin 1979.

[7] I.G. Rosenberg, Über die funktionale Vollständigkeit in den mehrwetigen Logiken. Rozpr. CSAV Rada Mat. Prir. Ved. Praha, 80, 4 (1970), 3-93.

[8] A Szendrei, On closed sets of linear operations over a finite set of square-free cardinality, EIK 14 (1978) 11, 547-559.

Közlemények 29/1983 41-58

ОБ ЭКВИВАЛЕНТНОСТИ И УСТОЙЧИВОСТИ ДЛЯ ЭВРИСТИЧЕСКИХ АЛГОРИТМОВ РАСПОЗНАВАНИЯ И ИХ ПРИМЕНЕНИЯ

ХОАНГ КИЕМ и ХО ТУ БАО

Институт Информатики и Кибернетики

Ханой - СРВ

В данной работе исследованы вопросы эквивалентности и устойчивости алгоритмов распознавания, приводятся результаты решения нескольких задач в геологии.

В последнее время теория распознавания образов находит широкое применение во многих областях народного хозяйства. Часто при решении конкретных задач перед математиками-прикладниками возникают интересные вопросы, от ответов на которые зависит эффективность применяемого метода. При решении нескольких задач в геологическом исследовании в СРВ нам пришлось решать вопросы эквивалентности и устойчивости распознающих алгоритмов. В этой работе даётся попытка решения этих вопросов.

Сначала приводим краткое описание некоторых классов распознающих алгоритмов.

§ 1. ОПИСАНИЕ АЛГОРИТМОВ РАСПОЗНАВАНИЯ

Пусть задано множество допустимых объектов $\{S\}$. Множество $\{S\}$ покрыто конечным числом подмножеств $K_1,\dots, K_\ell$, называемых классами, $\{S\}=\bigcup_{j=1}^{\ell} K_j$. Каждый объект S представляется набором из n значений признаков $I(S)=(a_1(S),\dots, a_n(S))$ Далее задаётся некоторая начальная информация I_o о классах. Тогда основная задача распознавания Z состоит в следующем: для каждого объекта выделенного подмножества $\tilde{S}^q \subseteq \{S\}$ установить, используя информацию I_o, к каким из классов $K_1, \ldots, K_\ell$ принадлежит этот объект.

Введем набор предикатов $P_j(S)=$"$S \in K_j$", $j=1,\ldots,\ell$. Набор $\alpha(S) = (P_1(S),\ldots,P_\ell(S))$ называется информационным вектором объекта S. Таким образом, результат решения задачи Z может быть представлен в виде информационной матрицы $\|P_j(S^i)\|_{q\times\ell}$, $K_1,\ldots, K_\ell$ -классы, $\tilde{S}^q = \{S^1,\ldots,S^q\}$. Если кодировать, например символом Δ, отказ от вычисления $P_j(S^i)$, то можно считать, что алгоритм A решения задачи Z вычислит матрицу $\|\beta_{ij}\|_{q\times\ell}$. Если $\beta_{ij}\in\{0,1\}$, то β_{ij} -значение $P_j(S^i)$, вычисленное алгоритмом A. При $\beta_{ij}=\Delta$ алгоритм отказался от вычисления $P_j(S^i)$.

Известно, что каждый алгоритм распознавания A может быть представлен через последовательное выполнение алгоритмов B и C, $A = B\cdot C$, причём если $A(z) = \|\beta_{ij}\|_{q\times\ell}$, то $B(z)=\|b_{ij}\|_{q\times\ell}$, b_{ij} -действительные числа, $C(\|b_{ij}\|_{q\times\ell})=\|\beta_{ij}\|_{q\times\ell}$. Подалгоритм C называется в дальнейшем решающим правилом и подалгоритм B - распознающим оператором, который переводит начальную информацию I_o и описания распознаваемых объектов $I(S^1),\ldots,I(S^q)$ в числовую матрицу $\|b_{ij}\|_{q\times\ell}$, где b_{ij} есть значение функции принадлежности S^i классу K_j, $i=1,\ldots,q$; $j=1,\ldots,\ell$ [1]

Чтобы вычислить матрицу $\|b_{ij}\|_{q\times\ell}$, почти все алгоритмы распознавания требуют вычисления гипер-матрицы

$$HB(Z) = \|hb_{ij}\|_{q\times\ell}, \quad hb_{ij}= \{b_{ij}^k\}_{k=1}^{N_j}, \quad N_j=\mathrm{card}(\tilde{K}_j), \quad \tilde{K}_j=K_j \cap \tilde{S}^m$$

b_{ij}^{k} - вещественные или комплексные числа.

После определения трансформации $F: HB \to B$, имеем

$$F\|hb_{ij}\|_{q\times \ell} = \|F[hb_{ij}]\|_{q\times \ell} = \|b_{ij}\|_{q\times \ell}$$

Обычно рассматриваются трансформации, имеющие следующие виды:

$F[hb_{ij}] = M[hb_{ij}]$ - среднее или математическое ожидание множества $hb_{ij} = \{\rho(S^i, S_j^k)\}_{k=1}^{N_j}$

Алгоритмы, определяющие через $F[hb_{ij}]=M[hb_{ij}]$, называются М-алгоритмами распознавания /М-алг/

$F[hb_{ij}]=extr[hb_{ij}]$ - верхнее значение множества hb_{ij}

Алгоритмы, определяющие через $F[hb_{ij}] = extr[hb_{ij}]$, называются extr -алгоритмами распознавания /extr - алг/.

В дальнейшем, покажем, что почти все алгоритмы распознавания являются или М-алгоритмами или extr-алгоритмами.

(a) Алгоритмы вычисления оценок [1]

Напоминаем основные этапы определения этих алгоритмов:

(1) Указание системы опорных множеств Ω_A

(2) Определение функции близости $B_{\tilde{\omega}}(S,S^i)$

(3) Вычисление оценки $\Gamma_{\tilde{\omega}}(S,S^i)$ объекта S по объекту S^i и опорному множеству Ω

$$\Gamma_{\tilde{\omega}}(S,S^i) = f(B_{\omega}(S,S^i), \gamma(S_i), \tilde{p}(\tilde{\omega}))$$

(4) Вычисление оценки $\Gamma_{\tilde{\omega}}^{j}(S)$ объекта S по опорному множеству Ω по классу K_j

$$\Gamma^j_{\tilde\omega}(S) = \varphi(\Gamma_{\tilde\omega}(S,S^j_1), \dots, \Gamma_{\tilde\omega}(S,S^j_{N_j}))$$

(5) Вычисление оценки $\Gamma_j(S)$ по классу K_j

$$\Gamma_j(S) = \frac{1}{\mathrm{card}(\Omega_A)} \sum_{\tilde\omega \leftrightarrow \Omega \in \Omega_A} \frac{1}{\mathrm{card}(\tilde K_j)} \sum_{S^i \in \tilde K_j} \Gamma_{\tilde\omega}(S,S^i).$$

(6) Решающие правила в алгоритмах вычисления оценок

Отметим, что эти алгоритмы являются М-алгоритмами распознавания.

(б) Алгоритмы типа потенциальных функций [2]. Потенциальные функции обычно имеют виды:

$$\varphi(S,S') = e^{-\alpha\rho(S,S')}$$

или

$$\varphi(S,S') = \frac{c_1}{c_2+c_3\rho(S,S')}$$

где α, c_1, c_2, c_3 - константы, ρ - расстояние между S и S'. Это монотонные функции по ρ. Потенциал между объектом S и классом K_j определяется

$$\varphi(S,K_j) = \frac{1}{\mathrm{card}(\tilde K_j)} \sum_{S' \in \tilde K_j} \varphi(S,S')$$

В этом случае, имеем $hb_{ij} = \{\varphi(S^i, S^k_j)\}_{k=1}^{\mathrm{card}(\tilde K_j)}$, $b_{ij} = \varphi(S,K_j)$

Определение алгоритмов в этой модели состоит из двух этапов:

а/ Вычислить потенциальную матрицу $\|b_{ij}\|_{q\times\ell}$, b_{ij} -потенциал между объектом $S^i \in S^q$ и классом K_j.

б/ Решающее правило: Объект S относится к классу K_ω если

$$\varphi(S,K_\omega) = \max_j \varphi(S,K_j)$$

Очевидно, что алгоритмы типа потенциальных функций являются $M_{f(\rho)}$ - алгоритмами.

(в) Алгоритмы, основанные на принципе ближайшего соседа /БС-алг/ широко используются для решения задач распознавания образов. Элемент b_{ij} матрицы $\|b_{ij}\|_{q\times\ell}$ определяется расстоянием между $S^i \in \tilde{S}^q$ и его ближайшим соседом в $\tilde{K}_j$. Это значит

$$b_{ij} = \min_{S_\omega \in \tilde{K}_j} \rho(S^i, S_\omega)$$

и также имеем

$$hb_{ij} = \{\rho(S^i, S^k_j)\}_{k=1}^{\mathrm{card}(\tilde{K}_j)}$$

Итак, алгоритм ближайшего соседа определяет принадлежность S^i к классу K_ω, если $b_{iu} = \min_j b_{ij}$

Ясно, что это extr-алгоритм.

(г) Алгоритмы распознавания с параметрами [1].

На практике используются некоторые модели алгоритмов распознавания образов с параметрами. Параметры $\alpha_1,\alpha_2,\dots,\alpha_n$ - весами признаков. Параметры $\beta_1,\dots,\beta_\ell$ - весами классов. Параметры $\gamma_1^1,\gamma_1^2,\dots,\gamma_1^{N_1},\dots,\gamma_\ell^1,\dots,\gamma_\ell^{N_\ell}$ - весами объектов. В дальнейшем, обозначим, например, М-алг с параметрами признаков $\alpha_1,\dots,\alpha_n$ через M^α-алг, М-алг с параметрами α,β,γ через $M^{\alpha,\beta,\gamma}$-алг, и т.д.

§ 2. ЭКВИВАЛЕНТНОСТЬ ЭВРИСТИЧЕСКИХ АЛГОРИТМОВ РАСПОЗНАВАНИЯ

Алгоритмы распознавания A и B называются эквивалентными если

$$\forall_{i,j} : C^A_{ij} = C^B_{ij}$$

где $\|C^A_{ij}\|$ и $\|C^B_{ij}\|$ являются информационными матрицами для A и B. Эквивалентные алгоритмы A и B обозначим $A \sim B$.

Алгоритм распознавания A и B называются эквивалентными по вероятности, если

$$\forall_{i,j} : \lim_{card(\tilde{S}^m)\to\infty} P(C^A_{ij} = C^B_{ij}) = 1$$

Эквивалентные алгоритмы по вероятности A и B обозначим $A \overset{prob}{\sim} B$.

Будем рассматривать эквивалентность /Эквивалентность по вероятности/ для некоторых эвристических алгоритмов распознавания:

M_ρ - алгоритмы распознавания: $b_{ij} = M[hb_{ij}]$

$M_{f(\rho)}$ - алгоритмы распознавания: $b_{ij} = M[f(hb_{ij})]$

extr - алгоритмы распознавания: $b_{ij} = \text{extr}\ hb_{ij}$

extr $f(\rho)$- алгоритмы распознавания: $b_{ij} = \text{extr}\ f(hb_{ij})$

где

$$f(hb_{ij}) = \{f(b^1_{ij}),\dots,f(b^{N_j}_{ij})\}$$

<u>ТЕОРЕМА 1</u>. Если функция монотонна по ρ, то

$$\text{extr}\ \rho \text{ - алг} \sim \text{extr}\ f(\rho) \text{ -алг}$$

<u>Доказательство</u>. Заметим, что $\text{extr}\ f(\rho) = f(\text{extr}\ \rho)$, если f является монотонной по ρ. Тогда сразу имеем

$$\text{extr}\ \rho \text{ - алг} \sim \text{extr}\ f(\rho) \text{ - алг}$$

<u>ТЕОРЕМА 2</u>. Пусть распределение $\rho(\dot{S},S_k), S_k \in K_j$ является нормальным (m_j,σ) или экспоненциальным (λ), тогда

$$M_\rho \text{ - алгоритм} \overset{prob}{\sim} \text{extr } \rho \text{ - алгоритм}$$

<u>Доказательство</u>. Это доказательство проводится на основе следующего утверждения [6].

Обозначим через $E_n(x_k)$, $D_n(x_k)$ - математическое ожидание и дисперсию k-ого верхнего значения эталона из n элементов с нормальным распределением (m,σ). Тогда

$$E_n(x_k) = m - \sigma\left(\sqrt{2\ln n} - \frac{\ln\ln n + \ln 4\pi + 2[S_1(k) - C]}{2\sqrt{2\ln n}} + 0\left(\frac{1}{\ln n}\right)\right)$$

$$D_n^2(x_k) = \frac{\sigma^2}{2\ln n}\left[\frac{\pi^2}{6} - S_2(k)\right] + 0\left(\frac{1}{\ln^2 n}\right)$$

где $S_1(k) = \sum_{i=1}^{n-k+1} \frac{1}{i}$, $S_2(k) = \sum_{i=1}^{n-k+1} \frac{1}{i^2}$, C- постоянная Эйлера. Для экспоненциального распределения (λ), имеем

$$E_n(x_k) = m \sum_{i=n-k+1}^{n} \frac{1}{i}, \quad D_n^2(x_k) = m^2 \sum_{i=n-k+1}^{n} \frac{1}{i^2}, \quad m = \frac{1}{\lambda}$$

Так как решающее правило основывается на $\underset{j}{\text{extr}}\{b_{ij}\}$, то для доказательства теоремы достаточно доказать, что

$$E_n[\bar{x}(u)] = \text{extr } E_n[\bar{x}(j)] \overset{n\to\infty}{\Longleftrightarrow} E_n[x_k(u)] = \text{extr } E_n[x_k(j)],$$

$$j=1,\dots,\ell$$

где $E_n[\bar{x}(j)]$ и $E_n[x_k(j)]$ обозначают математическое ожидание k-ого верхнего значения j-ого эталона из n элементов. Для простоты и не теряя общности, можем предполагать, что extr = min и докажем в этом случае

$$E_n[\bar{x}(u)] = \min_j E_n[\bar{x}(j)] \overset{n\to\infty}{\Longleftrightarrow} E_n[x_k(u)] = \min_j E_n[x_k(j)]$$

Это эквивалентно

$$\forall j,j' = 1,\dots,\ell: m_j \le m_{j'} \overset{n\to\infty}{\Longleftrightarrow} E_n[x_k(j)] \le E_n[x_k(j')]$$

Тогда теорема доказана в силу вышеуказанного утверждения.

<u>ТЕОРЕМА 3</u>. Если функция f монотонна по ρ и распределение ρ является нормальным (m_j,σ) или экспоненциальным (λ), то справедлива следующая схема

$$\begin{array}{ccc} M_\rho \text{ - алг} & \overset{prob}{\sim}\text{———} & M_{f(\rho)} \text{ алг} \\ \text{prob}\wr\,\Big| & & \Big|\,\wr\text{prob} \\ \text{extr } \rho \text{ - алг} & \text{———} & \text{extr } f(\rho) \text{ - алг} \end{array}$$

<u>Доказательство</u>. Из доказанных теорем, имеем

$$M_\rho \text{ - алг} \overset{prob}{\sim} \text{extr } \rho \text{ - алг}$$

$$\text{extr } \rho \text{ - алг} \sim \text{extr } f(\rho) \text{ - алг}$$

Сразу вытекает

$$M_{f(\rho)} \text{ - алг} \overset{prob}{\sim} \text{extr } f(\rho) \text{ - алг}$$

Наконец, учитывая, что

$$M_\rho \text{ - алг} \overset{prob}{\sim} \text{extr } \rho \text{ - алг} \sim \text{extr}(\rho) \text{ - алг} \overset{prob}{\sim} M_{f(\rho)} \text{ - алг}$$

Тогда

$$M_\rho \text{ - алг} \overset{prob}{\sim} M_{f(\rho)} \text{ - алг}$$

Доказана теорема.

Рассмотрим некоторый результат об эквивалентности M_ρ^γ -алгоритма с M_ρ -алгоритмом.

Расстояние $\rho_k = \rho(S,S^k), S^k \in K_j$ - случайная величина ρ

Вес объектов класса $K_j : \gamma^1, \ldots, \gamma^{N_1}$ - случайная величина γ

$\bar{\rho}$- арифметическое среднее расстояний между S и $\tilde{K}_j$

$$\bar{\rho} = \frac{1}{N_j} \sum_{k=1}^{N_j} \rho_k$$

$\bar{\rho}_\gamma$ - арифметическое среднее расстояний с весами объектов между S и $\tilde{K}_j$

$$\bar{\rho}_\gamma = \frac{\sum_{k=1}^{N_j} \rho_k \gamma_k}{\sum_{k=1}^{N_j} \gamma_k}$$

$M(\bar{\rho}_\gamma)$- действительное среднее расстояний с весами объектов между S и K_j

$$M(\bar{\rho}_\gamma) = \lim_{N_j \to \infty} \overline{\rho_\gamma}$$

$\hat{\sigma}^2$, $(\hat{\sigma}^2_{\rho\gamma})$ - экспериментальная дисперсия случайной величины ρ /с арифметическим средним с весами/

$$\hat{\sigma}^2_\rho = \frac{\sum_{k=1}^{N_j} (\rho_k - \bar{\rho})^2}{N_j - 1}, \quad \hat{\sigma}^2_{\rho\gamma} = \frac{\sum_{k=1}^{N_j} (\rho_k - \bar{\rho}_\gamma)^2}{N_j - 1}$$

$\hat{V} = \frac{\hat{\sigma}_\rho}{\rho}, \hat{V}_\gamma = \frac{\hat{\sigma}_\gamma}{\bar{\gamma}}$ - трансформированные коэффициенты случайных величин ρ и γ

$\hat{r}_{\rho\gamma} = \frac{\sum_{k=1}^{N_j} (\rho_k - \bar{\rho})(\gamma_k - \bar{\gamma})}{N_j \bar{\rho} \bar{\gamma}}$ - эксперимантальный коэффициент корреляции между ρ и γ

Тогда имеем:

<u>ТЕОРЕМА 4</u>. Если ρ и γ статистически независимы, то

$$M_{\rho}^{\gamma} - \text{алг} \overset{prob}{\sim} M_{\rho} - \text{алг}$$

<u>Доказательство</u>. Будем доказывать, что

$$\bar{\rho}_{\gamma} = \bar{\rho}\,(1 + \hat{r}_{\gamma}\cdot\hat{V}_{\rho}\,\hat{V}_{\gamma})$$

Имеем

$$\bar{\rho}(1 + \hat{r}_{\rho\gamma}\,\hat{V}_{\rho}\,\hat{V}_{\gamma}) = \bar{\rho}\,\left(1 + \frac{\sum_{k=1}^{N_j}(\rho_k - \bar{\rho})(\gamma_k - \bar{\rho})}{N_j\,\hat{\sigma}_{\rho}\cdot\hat{\sigma}_{\gamma}}\cdot\frac{\hat{\sigma}}{\bar{\rho}}\cdot\frac{\hat{\sigma}_{\gamma}}{\bar{\gamma}}\right)$$

$$= \bar{\rho}\,\left(1 + \frac{\sum_{k=1}^{N_j}(\rho_k - \bar{\rho})(\gamma_k - \bar{\gamma})}{N_j\cdot\bar{\rho}\cdot\bar{\gamma}}\right) = \bar{\rho}\left(1 + \frac{\sum_{k=1}^{N_j}\rho_k\gamma_k - N_j\,\bar{\rho}\,\bar{\gamma}}{N_j\cdot\bar{\rho}\cdot\bar{\gamma}}\right)$$

$$= \bar{\rho}\,\left(1 + \frac{\sum_{k=1}^{N_j}\rho_k\,\gamma_k}{N_j\,\bar{\rho}\,\bar{\gamma}} - 1\right) = \frac{\sum_{k=1}^{N_j}\rho_k\gamma_k}{N_j\,\bar{\gamma}} = \frac{\sum_{k=1}^{N_j}\rho_k\gamma_k}{\sum_{k=1}^{N_j}\gamma_k} = \bar{\rho}_{\gamma}$$

Так как $\lim\limits_{N_j\to\infty}\bar{\rho}_{\gamma} = M(\bar{\rho}_{\gamma})$, $\lim\limits_{N_j\to\infty}\bar{\rho} = M(\bar{\rho})$ – математическое ожидание случайной величины ρ ,

$$\lim_{N_j\to\infty}\hat{r}_{\rho\gamma} = r_{\rho\gamma}\ ; \quad \lim_{N_j\to\infty}\hat{V}$$

То когда N_j стремится к бесконечности, справедливо

$$M(\bar{\rho}_{\gamma}) = M(\bar{\rho})[1 + r_{\rho\gamma}\,V_{\rho}\,V_{\gamma}]$$

Ясно, что если $r_{\rho\gamma}=0$, то

$$M(\bar{\rho}_{\gamma}) = M(\bar{\rho})$$

Теорема доказана.

§ 3. УСТОЙЧИВОСТЬ И НОРМИРОВАНИЕ ЭВРИСТИЧЕСКИХ АЛГОРИТМОВ РАСПОЗНАВАНИЯ

Дисперсия алгоритма распознавания А определяется следующим образом:

$$D(A) = D(\|b_{ij}\|_{m\times \ell}) = \sum_{i,j} D(b_{ij}), \quad D(b_{ij}) = D\,\{\rho(S',(K_j \setminus S') \cup S^i)\}$$

ТЕОРЕМА 5. Сохранив все условия теоремы 3, имеется

$$D(\text{БС-алг}) = \min D(\text{extr } \rho\text{-алг}$$

Доказательство. Известно, что БС-алг является min ρ -алг. Тогда достаточно доказать

$$\forall i,j \quad D(b_{ij}^{\min \rho}) = \min D(b_{ij}^{\text{extr}\,\rho})$$

где
$$b_{ij}^{\min \rho} = \min_k \{b_{ij}^k\}, \quad b_{ij}^{\text{extr}\,\rho} = \text{extr}_r \{b_{ij}^k\}$$

Из указанного утверждения теоремы 3, следует

$$D_n^2 (b_{ij}^{\text{extr}\,\rho}) = \frac{\sigma^2}{2 \ln n} \left[\frac{\pi^2}{6} - S_2(k) \right] + 0\left(\frac{1}{\ln^2 n} \right)$$

для случая нормального распределения и

$$D_n^2 (b_{ij}^{\text{extr}\,\rho}) = m^2 \sum_{i=n-k+1}^{n} \frac{1}{i^2}$$

для случая экспотенциального распределения. В обоих случаях $D_n(b_{ij}^{\text{extr}\rho})$ достигаются максимум при $k=1$, т.е

$$D_n(b_{ij}^{\min\rho}) = \min D_n(b_{ij}^{\text{extr}\rho})$$

СЛЕДСТВИЕ. Если распределение $\{\rho(S,S^k);S^k \in K_j\}$ является нормальным, то выполняется

$$D\ (\text{БС-алг}) \le D\ (\text{М-алг})$$

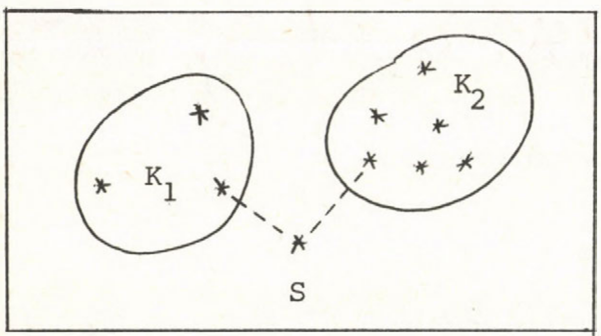
Это неравенство вытекает из

M_ρ -алгоритм = медиана ρ-алгоритм $\in$ {extr ρ-алгоритм}

Будем определять устойчивость распознающих алгоритмов через его дисперсию. Чем меньше его дисперсия, тем больше его устойчивость. В условиях теоремы 3 и из следствия теоремы 5 следует, что БС-алгоритм имеет наибольшую устойчивость. Эти результаты используются при проектировании комбинированной системы распознающих алгоритмов.

Возникает вопрос можно ли модифицировать алгоритмы распознавания для повышения их эффективности.

Сначала рассмотрим наглядный пример, в котором БС-алг - используется для задачи двух классов.



$\rho(S,K_1) = \rho(S,K_2)$

$S \in$?

Заметим, что

$$\rho(S,K_1) \approx \rho(S',K_1 \setminus S'),\ \forall S' \in K_1$$

$$\rho(S,K_2) \gg \rho(S'',K_2 \setminus S''),\ \forall S'' \in K_2$$

Несмотря на то, что $\rho(S,K_1) = \rho(S,K_2)$, S может относиться к классу K_1.

Следующий метод позволяет модифицировать распознающие алгоритмы для повышения их эффективности зависио от того, что структуры классов.

- Сначала, вычисляем

1/ $\rho(S, K_j)$

2/ $\{\rho(S',K_j)\}_{\forall S' \in \tilde{K}_j}$ или $\{\rho(S',K_j\ S')\}_{\forall S' \in \tilde{K}_j}$

- Обозначим

$M(\rho(S,K_j)) = M\{\rho(S',K_j)\}$ или $M\{\rho(S',K_j \setminus S')\}$

$D(\rho(S,K_j)) = D\{\rho(S',K_j)\}$ или $D\{\rho(S',K_j \setminus S')\}$

Имеем слудующую нормированную формулу

$$\tilde{\rho}(S,K_j) = \left| \frac{\rho(S,K_j) - M(\rho(S,K_j))}{D(\rho(S,K_j))} \right|$$

Алгоритм распознавания, модифицированный по указанному методу называется нормированным алгоритмом распознавания и обозначается через

Ясно, что если все распределения классов K_j одинаковы, то

$$\tilde{\rho}(S, K_j) \equiv \rho(S, K_j) \quad , \text{ т.е } \quad \tilde{A} \equiv A$$

<u>ТЕОРЕМА 6</u>. $\tilde{M}_\rho$ -алгоритм распознавания эквивалент $\tilde{M}_m$-алгоритму распознавания

Где $\tilde{M}_\rho : M_\rho$ - нормированный алгоритм распознавания

$\tilde{M}_m : M_m$ - нормированный алгоритм распознавания

$M_m = B.C, \quad B = \|b_{ij}\|_{m \times \ell}, \quad b_{ij} = (S^i, M(K_j))$

ρ_E - эвклидовое расстояние, $\rho = \rho_E^2$

<u>Доказательство</u>. Известно, что [5] по теореме Гиугельса

$$\rho(S,K_j) = \rho(S,M(K_j)) + D(K_j)$$

Непосредственно используя теорему Гиугельса и указанное определение выражения $M(\rho(S,K_j))$, $D(\rho(S,K_j))$ можно записать

$$M[\rho(S,K_j)] = \underset{S'\in K_j}{M}[\rho(S',M(K_j))+D(K_j)] = \underset{S'\in K_j}{M}[\rho(S',M(K_j))] + D(K_j)$$

$$D[\rho(S,K_j)] = \underset{S'\in K_j}{D}[\rho(S',M(K_j))+D(K_j)] = \underset{S'\in K_j}{D}[\rho(S',M(K_j)]$$

Тогда, нормированная формула для $\rho(S,K_j)$ задает таким образом

$$\tilde{\rho}(S,K_j) = \frac{\rho(S,M(K_j))+D(K_j)-\underset{S'\in K_j}{M}[\rho(S',M(K_j))] - D(K_j)}{\underset{S'\in K_j}{D}[\rho(S',M(K_j))]}$$

$$= \frac{\rho(S,M(K_j)) - \underset{S'\in K_j}{M}[\rho(S',M(K_j))]}{\underset{S'\in K_j}{D}[\rho(S',M(K_j))]}$$

Теорема доказана.

§ 4. НЕКОТОРЫЕ РЕЗУЛЬТАТЫ ПРИМЕНЕНИЯ КОМБИНИРОВАННОЙ СИСТЕМЫ РАСПОЗНАЮЩИХ АЛГОРИТМОВ В ГЕОЛОГИЧЕСКОМ ИССЛЕДОВАНИИ

Как известно, в геологическом исследовании часто встречаются задачи распознавания и классификации. Перед нами были поставлены конкретные задачи геологами. Для решения этих задач была составлена комбинированная система распознающих алгоритмов, состоящая из алгоритмов модели вычисления оценок, алгоритмов ближайшего соседа, алгоритмов К-внутригрупповых средних, алгоритмов типа потенциальных функций. Эта система реализована на ЭВМ и даёт хороший результат. Опишем вкратце эти задачи:

1. Распознавание гранитоидов с количественной информацией.

   Гранитоид является видом популярных объектов на Севере Вьетнама. Были собраны и проанализированы тысячи блоков гранитоидов.

   Задача состоит в том, что из таких блоков распознавать блоки, содержащие руды, например, свинцовая руда, антимон .... Каждый блок гранитоида объект определяется 10 числовыми признаками, являющими содержание компонентов SiO_2, TiO_2, Al_2O_3 ,... [8]

2. Классификация формации руд Pb - Zn [8]

   В этой задаче каждый распознаваемый объект рудная точка характеризуется 36 признаками, из которых 27 признаков о минералах, 3 признака об элементах, 2 признака об их отношениях и 4 признака о других породах. Знаем только о качественных информациях признаков, например, "значительное значение", "незначительное значение", ...

   По определенному разбиению эталона из 70 объектов на 4 формации, нужно распознавать сотни других рудных точек.

3. Классификация данных нефтяной скважины

   Нефтяной промысел является молодой областью в СРВ. Она еще не имеет необходимого количества обработанных данных, служащих эталонами. Поэтому на первых шагах обработки данных нефтяной скважины возникает вопрос автоматической классификации. Нефтяники предложили нам несколько задач, одна из которых - автоматическая классификация по глубине нефтяной скважины для выявления пластов песка, углей, глины, и т.д. Каждый распознаваемый объект характеризуется 7 признаками: диаметром скважины, удельным сопротивлением, естественным электрическим полем, нейтрон-кароттажем, γ -кароттажем ... . Мы обработали несколько скважин.

Эквивалентность и устойчивость распознаваемых алгоритмов применяются для сравнения результатов и определения весов алгоритмов при построении комбинированной системы следующих алгоритмов:

- алгоритм, основанный на принципе ближайшего соседа;
- алгоритмы вычисления оценок;
- модифицированный алгоритм K-внутригрупповых средних;
- алгоритмы типа потенциальных функций.

Применение этой системы для решения вышеуказанных задач указывает на то, что

- вероятность для того, чтобы результаты 4 алгоритмов были одинаковы является 0,72;
- верояность для того, чтобы результаты 3 алгоритмов были одинаковы является 0,78;
- вероятность для того, чтобы резульаты 2 алгоритмов были одинаковы является 0,99.

Авторы считают своим приятным долгом поблагодарить доктора Бак Хынг Кханг за внимание к работе и обсуждения результатов.

ЛИТЕРАТУРА

1 ЖУРАВЛЁВ Ю.И.: Об алгебраическом подходе к решению задач распознавания и классификации. Сб."Проблемы кибернетики", вып. 33, М., 1978.

2 АЙЗЕРМАН М.А., БРАВЕРМАН Э.М., РОЗОНОЗР Л.И.: Метод потенциальных функций в теорию обучения машин. Москва, Наука, 1970.

3 БАК ХЫНГ КХАНГ: Об одном параметрическом семействе статистических алгоритмов распознавания, Ж. Выч. Матем. и Мат. Физики, № 5, 1977.

4 Логико-информационные решения геологических задач. Из-во Наука, 1975.

5 DIDAY E.: Recent progress in distances and similarity measures in pattern recognition, Rapport de recherche, № 76, IRIA, 1974.

6 CRAMER H.: Mathematical Methods of Statistics, 1966.

7 HOANG KIEM: Some models and the recognition algorithms for complex survey /Thesis candidate for doctor in Institute of Computer Science and Cybernetice/, Hanoi, 1981.

8 BACH HUNG KHANG, HO TU BAO, NGUYEN HOANG PHUNG, GIANG VU THANG: The first application results of recognition methods in geological survey, Conference 1981 of Institute of Computer Science and Cybernetics, Hanoi.

ABOUT THE EQUIVALENCE AND STABILITY OF HEURISTIC ALGORITHM OF PATTERN-RECOGNITION WITH APPLICATION

Hoang Kiem, Ho Tu Bao
Hanoi, Vietnam

In the paper the question of the stability and equivalence of pattern-recognition algorithms are investigated. The results are applied to some problems of geology. The solutions of these problems are also discussed in the paper.

A HEURISZTIKUS ALAKFELISMERÉSI ALGORITMUSOK EKVIVALENCIÁJA ÉS STABILITÁSA ÉS ALKALMAZÁSAI

Hoang Kiem, Ho Tu Bao
Hanoi, Vietnám

A cikkben a heurisztikus alakfelismerési algoritmusokat vizsgáljuk, az ekvivalenciájukat és a stabilitásukat. Az eredmények geologiai feladatokra való alkalmazásáról is beszámolunk.

Közlemenyek 29/1983 59-73

РАСПОЗНАЮЩИЕ АЛГОРИТМЫ В ГИПЕРКОМПЛЕКСНОМ ПРОСТРАНСТВЕ

Бак Хынг Кханг, Хоанг Киэм

Институт Информатики и Кибернетики, СРВ

Распознавание образов в настоящее время находит широкое применение для решения практических задач во многих областях народного хозяйства, например, в геологии, медицине, гидрометеологии, технической диагностике, прогнозировании экономических, социальных процессов ... . Успехи в прикладном направлении являются бесспорными и общепризнанными. Однако до сих пор еще не выработана единая теория в области распознавания образов хотя попыток было больше чем достаточно. К настоящему времени в рамках математической теории распознавания образов оформилось несколько различных научных направлений. По существу можно говорить о нескольких теориях распознавания: статистической, алгебраической, структурной. Каждая из них имеет свою направленность и резко отличающийся от других набор обсуждаемых проблем.

Данная работа примыкает к алгебраической теории распознавания, развиваемой в работах [4 - 5] . Она отличается от упоминаемых работ тем, что модель распознающих алгоритмов строится в так называемом гиперкомплексном пространстве. Это вызывается тем, что при распознавании снимков снятых с самолетов, искусственных спутников и космических кораблей, объект выражается через n-мерный вектор, каждый компонент которого есть гиперкомплесное число, например, кватерион.

В данной работе доказывается, что n-мерное евклидово-пространство на k-ом гиперкомплексном поле будет гомоморфно k.n-мерному евклидову-пространству. Тем самым доказывается, что распознающие алгоритмы в n-мерном евклидовом-пространстве на k.n-ом гиперкомплексном поле являются распознающим алгоритмом в k.n-мерном евклидовом-пространстве. В частности алгоритм, распознающий цветные снимки [2] являются распознающим алгоритмом в 4n-мерном пространстве.

Другие результаты являются уставлениями условий корректности линейных решающих правил.

Сначала мы приводим некоторые необходимые понятия и определения.

§ 1. НЕКОТОРЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

1.1. Гиперкомплексное число

Число x называется $k+1$ -ым гиперкомплексным числом если:

$$x = x_o + x_1 e_1 + \ldots + x_k e_k$$

где $x_i \; (i = \overline{o,k})$ - действительные числа;

$e_i \; (i = \overline{1,k})$ - базисные элементы с некоторым законом умножения:

$$e_i e_j = p_{ij,o} + p_{ij,1} \cdot e_1 + \ldots + p_{ij,k} \cdot e_k$$

где

$p_{ij,l}$ - действительные числа, $l=\overline{o,k}$

Примеры некоторых особых гиперкомплексных чисел:

- Комплексные числа: $k = (a + bi)$
- Кватернион: $Q = (a + bi + cj + dk)$
- Октава: $O = \{a + bi + cj + dk + AE + BI + CJ + DK\}$

Эти гиперкомплексные числа играют особую роль в теориях нормированной алгебры, разделимой ассоциативной алгебры [1] и в настоящее время, они имеют некотрые приложения в построении распознавающих моделей, например, на основе кватерниона [2].

1.2. Гиперкомплексное пространство

Известно, что псевдоэвклидово пространство индекса является эвклидовым, в котором система n-базисных векторов $e_1, e_2, \ldots, e_n$ есть k-векторов e_i, где $e_i^2 < 0$.

В теории дифференциального уравнения, мы встречали псевдоэвклидово пространство индекса 1, когда рассматриваем волновое уравнение n-переменных. Особенно, в теории относительности установились отношения 4-мерного время-пространства с псевдоэвклидовым пространством индекса 1 [3].

Гиперкомплексное пространство есть псевдоэвклидово пространство индекса К на гиперкомплексном поле.

Если К=0 и гиперкомплексное поле является комплексным, то мы имеем комплексное эвклидово пространство или унитарное пространство с известным скалярным произведением.

Если К=0 и гиперкомлексное поле является кватернионным полем, то имеем эвклидово пространство на кватернионном поле [2].

1.3. Скалярное поле в гиперкомплексном пространстве

Скалярное произведение в эвклидовом пространстве удовлетворяет аксиомам:

1. $(x, x) \geqslant 0$, $(x, x) = 0 \iff x = 0$
2. $(x, y) = (y, x)$
3. $(x, ky) = k(x, y)$
4. $(x + y, z) = (x, z) + (y, z)$

Если
$$x = x_1 e_1 + x_2 e_2 + \ldots + x_n e_n$$
$$y = y_1 e_1 + y_2 e_2 + \ldots + y_n e_n$$

при

$$(e_i, e_j) = \delta_{ij} \quad \text{то} \quad (x, y) = \sum_{i=1}^{n} x_i y_i$$

В 4-ом метапсевдоэвклидовом пространстве индекса 1, минковское скалярное произведение определяется:

$$(x, y) = -x_1 y_1 + \sum_{i=2}^{4} x_i y_i \quad (e_1^2 = -1,\ e_2^2 = e_3^2 = e_4^2 = 1)$$

В эвклидовом пространстве на комплексном поле имеем следующие способы определения /зависит от наличия аксиомы 1 или нет/

а - $(x, y) = \sum_{i=1}^{n} x_i y_i$ /не удовлетворяет аксиоме 1/

б - $(x, y) = \sum_{i=1}^{n} x_i \overline{y_i}$ $\overline{y_i}$: комплексная сопряжённая

в - $(x, y) = \overline{(y, x)}$ /комплексная сопряжённая (y, x)/

Скалярное произведение, определяющее по б -, является особым случаем скалярного умножения, определяющего по в - /скалярное умножение в унитарном пространстве/.

Чтобы определить скалярное произведение в гиперкомплексном пространстве, рассмотрим следующую лемму:

<u>Лемма 1:</u> Задано 2 вектора x, y в гиперкомплексном пространстве на ассоциативном поле:

$$x = x_1 e_1 + x_2 e_2 + \dots + x_n e_n$$

$$y = y_1 e_1 + y_2 e_2 + \dots + y_n e_n$$

Если в этом пространстве определяется скалярное произведение, удовлетворяющее аксиомам (2), (3) , (4), то

$$(x, y) = \sum_{i,j} x_i y_i (e_i, e_j)$$

Доказательство:

Из (2), (3), (4) следует:

3') $(kx, y) = k(x, y)$

4') $(x + y, z) = (x, z) + (y, z)$

Из аксиом 3) , 3') имеем: 3'') $(kx, ly) = kl(x, y)$
Из 4-, 4'-) и 3''-) следует:

$$(x, y) = (x_1 y_1 + x_2 y_2 + \ldots + x_n y_n, y_1 e_1 + y_2 e_2 + \ldots + y_n e_n) = \sum_{i,j} x_i y_i (e_i, e_j)$$

Следствие 1: Если $\{e_i\}$, $i = \overline{1, n}$ есть система ортонормированных базисных векторов, то:

$$(x, y) = \sum_{i=1}^{n} x_i y_i$$

Следствие 2: Если $\{e_i\}$, $i = \overline{1, n}$ есть система базисных векторов в псевдоэвклидовом пространстве индекса К, т.е.:

$$e_1^2 = e_2^2 = \ldots = e_k^2 = -1, e_{k+1}^2 = \ldots = e_n^2 = 1$$

и $(e_i, e_j) = 0$, $(i, j = \overline{1, n}, i \neq j)$, то

$$(x, y) = -\sum_{i=1}^{k} x_i y_i + \sum_{i=k+1}^{n} x_i y_i$$

то скалярное произведение в 4-мерном время-пространстве минковское скалярное произведение в 4-мерном время-пространстве является особым видом скалярного произведения.

Лемма 2: Пусть дано 2 вектора x, y в гиперкомплексном пространстве на неассоциативном поле

$$x = x_1e_1 + x_2e_2 + \ldots + x_ne_n$$

$$y = y_1e_1 + y_2e_2 + \ldots + y_ne_n$$

Если в этом пространстве определяется скалярное произведение, удовлетворяющее аксиомам 2, 3, 4, то:

$$(x, y) = \sum_{i,j} \frac{1}{2} (x_iy_j + y_jx_i) (e_i, e_j)$$

Доказательство:

Из леммы 1 имеем: $(x, y) = \sum_{i,j} x_iy_j (e_i, e_j)$

$$(y, x) = \sum_{i,j} y_ix_j (e_i, e_j) = \sum_{i,j} y_jx_i (e_j, e_i) = \sum_{i,j} y_jx_i (e_i, e_j)$$

$$(x, y) + (y, x) = \sum_{i,j} (x_iy_j + y_jx_i) (e_i, e_j)$$

$$(x, y) = (y, x) = \sum_{i,j} \frac{1}{2} (x_iy_j + y_jx_i) (e_i, e_j)$$

Следствие 1: Если $\{e_i\}$, $i = \overline{1, n}$ есть система базисных векторов в псевдоэвклидовом пространстве индекса K:

$$e_1^2 = e_2^2 = \ldots = e_k^2 = -1, \; e_{k+1}^2 = \ldots = e_n^2 = 1 \quad \text{и}$$

$$(e_i, e_j) = 0, \; (i, j = \overline{1, n}, \; i \neq j)$$

то

$$(x, y) = -\frac{1}{2} \sum_{i=1}^{k} (x_iy_i + y_ix_i) + \frac{1}{2} \sum_{i=k+1}^{n} (x_iy_i + y_ix_i)$$

Следствие 2: Если $\{e_i\}$, $i = \overline{1, n}$ есть система ортонормированных базисных векторов гиперкомплексного пространства на неассоциативном поле, то:

$$(x, y) = \frac{1}{2} \sum_{i=1}^{n} (x_iy_i + y_ix_i)$$

1.4. Правило умножения гиперкомплексных чисел

Известно, что умножение гиперкомплексных чисел по обычному основывается на правиле умножения $e_i \cdot e_j$ как указано выше, в пункте 1.1.

Кроме того, мы определим еще некоторые следующие правила умножения:

1 - $x \cdot y = x \cdot \bar{y}$

2 - $x \cdot y = \bar{x} \cdot y$

3 - $x \cdot y = \bar{x} \cdot \bar{y}$

где $\bar{y}$ и $\bar{x}$ - сопряженные гиперкомплексные числа y и x.

Выбор конкретного правила умножения зависит от надобности и содержания проблемы, которую надо решать.

§ 2. РАСПОЗНОВАНИЕ ОБРАЗОВ В ГИПЕРКОМПЛЕКСНОМ ПРОСТРАНСТВЕ

2.1. Образ в гиперкомплексном пространстве

В настоящее время существуют некоторые модели алгоритмов распознования в гиперкомплексном пространстве. Например, модель распознающих алгоритмов на цветных снимках [2].

По этой модели каждый объект образа выражается через n-мерный вектор, каждая компонента которого есть кватернион. Эта модель лучше отражает процессы рождения и преобразования образов, и распознающий алгоритм на кватернионном векторном пространстве, как мы рассмотрим ниже, действительно является распознающим алгоритмом по метрике в 4n - мерном пространстве.

Для обобщения, считаем образ в гиперкомплексном пространстве как множество векторов и каждая компонента которых есть гиперкомплексное число. Эти векторы можно определить в эвклидовом или псевдоэвклидовом пространствах.

2.2. Расстояние в гиперкомплексном пространстве

Известно, что расстояние между двумя векторами x и y в эвклидовом пространстве определяется:

$$\rho(x, y) = \sqrt{(x - y, x - y)}$$

Это определение негодится для гиперкомплексного пространства, потому что, как указано выше, скалярное произведение двух векторов в гиперкомплексном пространстве будет гиперкомплексным числом.

Для удовлетворения определению о расстоянии мы должны определить скалярное произведение в гиперкомплексном пространстве, чтобы $(x, x) > 0, \forall x \neq 0$

Теорема 1: n-мерное пространство на k-ом гиперкомплексном поле со скалярным произведением, определяемым:

1/ $(x, y) = \frac{1}{2}[(x, \bar{y}) + (y, \bar{x})]$

2/ $(\lambda x, y) = \lambda (x, y)$

3/ $(x_1 + x_2, y) = (x_1, y) + (x_2, y)$

4/ $(x, x) \geqslant 0$, $(x, x) > 0$ если $x \neq 0$

будет гомоморфно $k \cdot n$ -мерному эвкливовому пространству, где $\bar{y}$ и $\bar{x}$ - векторы, компоненты которых являются гиперкомплексными сопряженными компонентами y и x .

Доказательство: Из леммы 2, следствия /1/ имеем:

$$(x, y) = \frac{1}{2}\sum_{i=1}^{n}(x_i y_i + y_i x_i)$$

Определить скалярное произведение по лемме 2 имеем:

$$(x, y) = \frac{1}{2} \sum_{i=1}^{n} (x_i\bar{y}_i + y_i\bar{x}_i)$$

x_i, y_i - есть К-ые гиперкомплексные числа.

$$x_i = x_{i1}e_1 + x_{i2}e_2 + \ldots + x_{ik}e_k$$

и

$$\bar{x}_i = x_{i1}e_1 - x_{i2}e_2 - \ldots - x_{ik}e_k$$

$$y_i = y_{i1}e_1 + \ldots + y_{ik}e_k \quad \text{и} \quad \bar{y}_i = y_{i1}e_1 - \ldots - y_{ik}e_k$$

/обозначим $e_1 \equiv 1$ /.

Так получим

$$(x, y) = \frac{1}{2} \sum_{i=1}^{n} [(\sum_{j=1}^{k} x_{ij}e_j)(y_{i1}e_i - \sum_{j=2}^{k} y_{ij}e_j) + (\sum_{j=1}^{k} y_{ij}e_j)(x_{i1}e_1 - \sum_{j=2}^{k} x_{ij}e_j)]$$

После упрощения мы имеем:

$$(x, y) = \sum_{i=1}^{n} \sum_{j=1}^{k} x_{ij} y_{ij} .$$

Таким образом каждому вектору x в гиперкомплексном пространстве соответствует n векторов с k компонентами

$$x' : (x_{11}, x_{12}, \ldots, x_{1k}, \ldots, x_{n1}, x_{n2}, \ldots, x_{nk})$$

Ясно, что это соответствие есть гомоморфизм, потому что оно сохранит скалярность $(x, y) = (x', y')$.

<u>Следствие</u>: n -мерное эвклидово пространство на кватернионном поле со скалярным умножением, определяющим в теореме 1, будет гомоморфно $4n$ -мерному эвклидовому пространству.

Гомоморфизм гиперкомплексного пространства с эвклидовым пространством позволяет использовать распознающие алгоритмы, построенные в эвклидовом пространстве для решения задачи распознавания в гиперкомплексном пространстве.

2.3. Решающее правило с порогом в гиперкомплексном пространстве

2.3.1. Линейное правило с порогом в действительном эвклидовом пространстве

Пусть дана система линейных функций:

$$\Gamma_j(x) = a_{j1} x_1 + a_{j2} x_2 + \ldots + a_{jl} x_l + a_{jl+1}\ ; \quad j = \overline{1, l} \tag{1}$$

и решающее правило с порогом

$$a_j^A(S') = \begin{cases} 1, & \text{если } \Gamma_j(S') > \delta_{j1} \\ 0, & \text{если } \Gamma_j(S') < \delta_{j1} \\ \Delta, & \text{если } \delta_{j2} \leqslant \Gamma_j(S') \leqslant \delta_{j1} \end{cases} \tag{2}$$

где δ_{j1}, δ_{j2} – константы.

Обозначим:

$$\| A \| = \begin{bmatrix} a_{11} & a_{12} \cdots & a_{1l} \\ a_{21} & a_{22} \cdots & a_{2l} \\ \vdots & & \\ a_{l1} & a_{l2} \cdots & a_{ll} \end{bmatrix}$$

Имеем следующую систему.

ТЕОРЕМА: Линейное решающее правило с порогом $a_j^A(S')$ определяемое по (2), является корректным, если $\det \| A \| \neq 0$

Понятия о корректном алгоритме и решающем правиле определяются в [4].

<u>Доказательство</u>: Допустим, что $\tilde{S}^q = S'_1,\ldots,S'_q$ есть множество q допустимых объектов с истинной информационной матрицей $\|d_{ij}\|_{ql}$. Условия для правильной классификации объектов $S'_1, S'_2,\ldots,S'_q$ соответствующих информационной матрице по решающему правилу $a_j^A(S')$ являются

$$\left.\begin{array}{l} a_{11}b_{k1} + a_{12}b_{k2} + \ldots + a_{1l}b_{kl} + a'_{1,l+1} \geq 0 \\ a_{21}b_{k1} + a_{22}b_{k2} + \ldots + a_{2l}b_{kl} + a'_{2,l+1} \geq 0 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{21}b_{k1} + a_{l2}b_{k2} + \ldots + a_{ll}b_{kl} + a'_{l,l+2} \geq 0 \end{array}\right\} \quad k = \overline{1,q} \qquad (3)$$

Здесь $a'_{j,l+1} = a_{j,l+1} - \delta_{j,u}, \quad u = 1,2$

Ясно, что система неравенств (3) будет совместна если следующая система совместна

$$a_{j1}x_1 + \ldots + a_{jl}x_l + a'_{j,l+1} \geq 0; \quad j = \overline{1,l} \qquad (4)$$

Как известно, что необходимое и достаточное условие совместимости системы (4) является:

$$\det \begin{Vmatrix} a_{11} & a_{12} & \cdots & a_{1l} \\ a_{21} & a_{22} & \cdots & a_{2l} \\ \cdots & \cdots & \cdots & \cdots \\ a_{l1} & a_{l2} & \cdots & a_{ll} \end{Vmatrix} \neq 0$$

2.3.2. Линейное решающее правило с порогом в гиперкомплексном пространстве

Обозначим

$$a_j = (a_{j,l+1}, a_{j1}, a_{j2}, \ldots, a_{jl})$$
$$x = (1, x_1, x_2, \ldots, x_l)$$

Перепишем систему линейных функций (1) в следующем виде

$$\Gamma_j(x) = (a_j, x), \quad j = \overline{1, l}$$

В n-мерном эвклидовом пространстве К-мерном гиперкомплексном поле со скалярным произведением, определяемым в теореме 1, мы можем расширять концепции решающего правила с линейным порогом таким образом:

$$a_j^*(s') = \begin{cases} 1, & \text{если: } (a_j^*, x^*) > \delta_{j1} \\ 0, & \text{если: } (a^*, x^*) < \delta_{j2} \\ \Delta, & \text{если: } \delta_{j2} \leqslant (a_j^*, x^*) \leqslant \delta_{j1} \end{cases}$$

Здесь a_j^* и x^* n-мерные векторы, каждая компонента которых является К-мерным гиперкомплексным числом.

Понятие об корректных алгоритмах и решающих правилах в основном определяется по [4, 5], здесь мы только расширяем концепции о матрицах, соответствующих распознающим операторам [4, 5] с элементами в гиперкомплексном пространстве.

Каждая i-ая компонента a_j^* имеет вид:

$$a_{ji}^* = a_{ji}^1 + a_{ji}^2 + \ldots + a_{ji}^k e_{k-1}$$

Обозначим:

$$A^{*}=\begin{bmatrix} a_{11}^{1} & a_{11}^{2} & \dots & a_{11}^{k} & a_{12}^{1} & a_{12}^{2} & \dots & a_{12}^{k} & \dots & a_{1l}^{1} & a_{1l}^{2} & \dots & a_{1l}^{k} \\ a_{21}^{1} & a_{21}^{2} & \dots & a_{21}^{k} & a_{22}^{1} & a_{22}^{2} & \dots & a_{22}^{k} & \dots & a_{2l}^{1} & a_{2l}^{2} & \dots & a_{2l}^{k} \\ \dots & & & & & & & & & & & & \dots \\ a_{l1}^{1} & a_{l1}^{2} & \dots & a_{l1}^{k} & a_{l2}^{1} & a_{l2}^{2} & \dots & a_{l2}^{k} & \dots & a_{ll}^{1} & a_{ll}^{2} & \dots & a_{ll}^{k} \end{bmatrix}$$

Для сокращения обозначим:

$$\left\| A^{*} \right\| = \left\| a_{ij}^{P} \right\|_{l,\, l(k)} ; \quad \begin{matrix} P = \overline{1, k} \\ i, j = \overline{1, l} \end{matrix}$$

<u>ТЕОРЕМА 3</u>: Линейное решающее правило с порогом определяемое по [5] является корректным если:

$$\text{ранг } \left\| A^{*} \right\| = 1 \ .$$

<u>Доказательство</u>: Пусть $\tilde{S}^{q} = \{S'_1, S'_2, \dots, S'_q\}$ является множеством q выборочных объектов в гиперкомплексном пространстве с информационной матрицей $\| a_{ij} \|_{ql}$. Условие для правильной классификации объектов $S'_1, S'_2, \dots, S'_q$ соответствующих информационной матрице $\| a_{ij} \|_{ql}$ по решающему правилу $a_j^{A^*}(S')$ является:

$$\left. \begin{matrix} (a_j^{*}, S'_1) - \delta_{j,\omega} \gtrless 0 \\ (a_j^{*}, S'_2) - \delta_{j,\omega} \gtrless 0 \\ \vdots \\ (a_j^{*}, S'_q) - \delta_{j,\omega} \gtrless 0 \end{matrix} \right\} \quad \begin{matrix} \omega = 1, 2 \\ j = 1, l \end{matrix} \tag{6}$$

Ясно, что система неравенств (6) будет совместима, если совместна следующая система

$$(a_j^{*}, x) - \delta_{j,\omega} \gtrless 0 ; \quad j = \overline{1, l} \tag{7}$$

Разлагая ее имеем:

$$\sum_{i=1}^{l} \sum_{j=1}^{k} a_{ji}^{p} x_{i}^{p} - \delta_{j,\omega} \geqslant 0 \qquad j=\overline{1,l}$$

Следовательно, что необходимым и достаточным условием совместности системы (7) является:

$$\text{ранг} \left\| a_{ij}^{p} \right\|_{l,k\cdot l} = l \qquad \begin{matrix} p=\overline{1,k} \\ i,j=\overline{1,l} \end{matrix}$$

Теорема доказана.

ЛИТЕРАТУРА

1. Кантор И.Л., Солодовник А.С.: Гиперкомплексные числа. Изд. Наука, М., 1973.

2. Журник И.Г.: О некоторых вопросах опознавания образов на цветных снимках. Геодезия и Аэрофотосъемка, № 6, 1978.

3. Кильчевский А.: Элементы тензорного анализа и его приложение в механике. Изд-во Наука, М., 1963.

4. Журавлев Ю.И.: Об алгебраическом подходе к решению задач распознования и классификации. Сб. Проблемы кибернетики, вып. 33, М., Наука, 1978.

5. Бак Хынг Кханг: Полнота линейных замыканий некоторых классов алгоритмов распознавания. ДАН СССР, № 4, 1978.

PATTERN-RECOGNITION ALGORITHMS IN HYPERCOMPLEX SPACE

Bak Hing Khang, Hoang Kiem
Hanoi, Vietnam

The paper deals with the questions belonging to algebraic theory of pathern-recognition. The algorithms of pattern--recognition are investigated in so called hypercomplex, space (say n-dimensional space of quaternions). The main field of applications is the study of pictures ganied by air-planes or satelites.

ALAKFELISMERÉSI ALGORITMUSOK A HIPERKOMPLEX TÉRBEN

Bak Hing Khang, Hoang Kiem
Hanoi, Vietnam

A cikk az alakfelismerés algebrai elméletéhez kapcsolódik. Az alakfelismerési algoritmusokat a szerzők u.n. hiperkomplex térben vizsgálják mondjuk a kvaternionok n-dimenziós térben . A fő alkalmazási terület a repülőgépek ill. ürhajók által nyert képek kiértékelése.

Közlemenyek 29/1983 75-91

THE ALGEBRAIC STRUCTURE OF PRIMITIVE RECURSIVE FUNCTIONS

István Szalkai

student of Eötvös Loránd University,

Budapest

ABSTRACT

In this article I looked at the set of primitive recursive functions as an algebraic structure with two operations: composition o and iteration $\square$, see below. I prove that there is no endomorphism on this structure besides ID and $\underline{0}$ (see Theorem I). After this I prove that certain sets of functions from N to N (for example the set of primitive recursive functions) cannot be generated with one function with the help of operations o and $\square$ (see Theorem II and III). In my paper I also write some problems concerning this topic.

INTRODUCTION

Denote by N the set of non-negative integers. We say that the function f is generated from the functions g and h by primitive recursion if there is a k such that

$$f : N^{k+1} \to N, \quad g : N^{k+2} \to N, \quad h : N^{k} \to N$$

and for every $\underline{m} \in N^k$: $f(0,\underline{m})=h(\underline{m})$ and $f(n+1,\underline{m})=g(n,\underline{m},f(n,\underline{m}))$. Let us define three special functions as follows:

$\underset{\sim}{O} : \mathrm{N} \to \{0\}$ *i.e.* $\forall n \in \mathrm{N}$ $\underset{\sim}{O}(n)=0$

$S : \mathrm{N} \to \mathrm{N}$: $\forall n \in \mathrm{N}$ $S(n)=n+1$

$\underline{\cdot} : \mathrm{N} \to \mathrm{N}$: $\forall n \in \mathrm{N}$ $\underline{\cdot}(n)= \begin{cases} n-1 & \text{if } n \geq 1 \\ 0 & \text{otherwise} \end{cases}$

These are called basic functions.

In general denote by $\underset{\sim}{c}$ the constant function from N to N having the value c, $c \in \mathrm{N}$.

A function f from N^k to N, $k \geq 1$, is called primitive recursive if it can be generated from the basic functions with the help of primitive recursion, composition and projections from N^k to N in finite steps. Since there is a calculable (i.e. primitive recursive) bijection between N^k and N we can restrict ourselves to functions from N to N. Denote by *PR* the set of the primitive recursive functions from N to N. For the usual definition of primitive recursive functions see [2], [4] or [5].

For an arbitrary function f from N to N denote by $f^{\square}$ the "iterand-function" of f from 0, i.e. let $f^{\square}(0)=0$ and for every n, $f^{\square}(n+1)=f(f\ (n))$. Furthermore we shall denote by quadres (n) the quadratic residuum of n, i.e. the distance between n and the greatest square number no greater than n. For example quadres $(16)=0$, quadres $(53)=4$, etc. Denote by o the operation of composition and by + the operation of addition of two functions. R.M. Robinson [1] proved that every primitive recursive function from N to N can be generated from the basic function S and the quadres with the help of the operations o, + and The starting point of my investigations was this fact. Denote by $\langle PR, o, \square \rangle$ the set *PR* as an algebraic structure with operations o and $\square$. The main purpose of this paper is to investigate the algebraic properties of $\langle PR, o, \square \rangle$.

It is easy to see that $\langle PR, o \rangle$ is a semigroup with a unit element id (since $id=S^{\square} \in PR$), where the only left-hand-singular elements are the constant ones and there is no right-hand-singular element. Obviously $\square$ is a unary operation from *PR* to *PR* and $Ker\,\square \supset Im$. It easy to see that $Ker\,\square = \{f : f(0)=0\}$ and *quadres* $\in Ker\,\square \setminus Im\,\square$.

For every f $f^{\square\square}=\underset{\sim}{0}$, so the only fixpoint of $\square$ is $\underset{\sim}{0}$. (To prove this we have to use algebraic considerations only: we have to observe that if $f^{\square}=f$ then $f^{\square\square}=f$ and $\underset{\sim}{0}=f^{\square\square}=f^{\square}=f$.)

If $\;\; = \langle A, g_1, \ldots, g_m \rangle$ is an arbitrary algebraic structure, then every homomorphism from A to A is called endomorphism. Denote by *End( )* the set of these endomorphisms. The identity operation ID on A (i.e. $ID(a)=a$ for every $a \in A$) is trivially an element of *End( )*. If contains a null element 0 for every operations of then the null-operation $\underline{0}$ (i.e. $\underline{0}(a)=0$ for every element of A) is also an endomorphism on . It is obvious that $\underset{\sim}{0}$ is the null element of PR, i.e. $\underset{\sim}{0} \circ \underset{\sim}{0} = \underset{\sim}{0}$ and $\underset{\sim}{0}^{\square} = \underset{\sim}{0}$. So we got that ID and $\underline{0}$ are elements of $End\langle PR, o, \;\rangle$. In the next section I prove that $End\langle PR, o, \square \rangle = \{\underline{0}, ID\}$ (see Theorem I). After this I examine a more general question: what are the similar structures such that for them this theorem holds (see Theorem I.A and I.B). As I know well, these questions have not been investigated yet. Only the automorphisms of degrees were studied in [5] and [6]. In this paper I investigate the structure of null degree itself.

Above we have seen two equivalent definitions of primitive recursive functions (the usual one and the definition of R.M. Robinson.) There are still more equivalent definitions of them (see [3], [4]) and there are a lot of open problems concerning how to generate PR in a more simple way. I prove a theorem concerning this (see Theorem II) which seems to be an interesting one.

Namely the theorem says the following:

In what follows denote by $\langle a \rangle$ the set of functions f from N to N which can be built from the function a with the help of operations o and $\square$ in finite steps. Let a be an arbitrary function from N to N. Then the Theorem says that

either there is no bijection in $\langle a \rangle$

or $\forall f \in \langle a \rangle$ f is either injective or $R(f)$ is finite.

($R(f)$ denotes the range of f, i.e. $R(f)=\{f(i): i\in N\}$). Denote by PR^+ the monotone increasing elements of PR. The corollary of the above Theorem says that there is no $a\in PR$ such that $\langle a\rangle = PR$ or $\langle a\rangle = PR^+$ (see Theorem III). It is not known whether the operation + can be eliminated from the definition of primitive recursive functions or not. To be more exact the problem is the following:

PROBLEM 1.: Are there $u,v\in PR$ such that PR can be generated from the functions u and v with the help of the operations o and ?

To solve this problem I suggest to choose another σ_2^1 (see in [2], §.7.16.)

If such u and v exist then Theorem III implies that the result of R.M.Robinson is sharp. A similar result was proved in [3], Theorem 6. Let f^{-1} (inversion) denote the following operation: for every surjective f let f^{-1} be a function such that $f^{-1}(x)=min\{y: f(y)=x\}$. J.Robinson [3] proved that R cannot be generated from only one function with the help of operations o and f^{-1} (R denotes the set of general recursive functions). In her proof she eliminated the operation + with the help of a *, a "mirror-operation".

My Theorem I shows that her method is not applicable in our case because * is an endomorphism on R.

The results of this paper seem to be the first ones concerning the properties of $\langle PR, o, \Box\rangle$. I think it is interesting and useful to investigate similar problems, for example

to study other properties of the operators o and $\Box$

to investigate other operations on PR

(for example $(\Sigma f)(n)=f(0)+f(1)+\ldots+f(n)$ or f^{-1})

to raise other usual and unusual algebraic questions in the algebraic structure $\langle PR, o, \Box\rangle$, etc.

I thank Emil W. Kiss and B. Uhrin for their useful remarks.

ENDOMORPHISMS

The main purpose of present section is to prove Theorem I. To do this we need some abbreviations and two lemmas, but first I explain some remarks on the endomorphisms on PR.

Let $c \in N$, $c \neq 0$ and let L be the following endomoprhism: for $\forall f \in PR$ let $L(f)=\underset{\sim}{c}$ i.e. $L:PR \to \{\underset{\sim}{c}\}$. Since $\underset{\sim}{c}$ is a left-hand-singular element of PR, so it is easy to see that $L \in End\langle PR, o\rangle$. Since $\underset{\sim}{c}^{\square}(0)=0$ so $\underset{\sim}{c}^{\square} \neq \underset{\sim}{c}$ so $L \notin End\langle PR, \square\rangle$ i.e. we have got that $L \in End\langle PR, o\rangle \setminus End\langle PR, \square\rangle$. Conversely let $\hat{s}g=(S\circ\underset{\sim}{0})^{\square}$ be the usual signum function: $sg(0)=0$ and $sg(n)=1$ for $n \neq 0$.

Let $L(f)=sg\circ f\circ sg$ for every $f \in PR$. Then it is easy to see that $L \in End\langle PR, \square\rangle$. (We have to examine where f is equal to 0 and where it is not.) Furthermore by Lemma 1 we can say that $L \notin End\langle PR, o\rangle$ so $L \in End\langle PR, \square\rangle \setminus End\langle PR, o\rangle$. If we want an easier example for $L \in End\langle PR, \square\rangle \setminus End\langle PR, o\rangle$ then let $L(f)=f^{\square}$ for every $f \in PR$.

The following lemma is useful both to the above elementary investigations and to prove the main theorem.

LEMMA 1. Let $u, v \in PR$ be arbitrary functions such that $v\ u \neq id$ and u is not a constant function. Let $L(f)=u\ f\ v$ for every $f \in PR$. Then $L \notin End\langle PR, o\rangle$.

PROOF: Let $x_1, x_2, Z \in \mathrm{N}$ be such that $(v\circ u)(Z)=y \neq Z$ and $u(x_1) \neq u(x_2)$ Furthermore let $f, g \in PR$ be szch that

$$g(v(0))=Z \quad and \quad f(Z)=x_1, \quad f(y)=x_2.$$

Such f and g obviously exist. For example let $g(n)=0$ for $n \neq v(0)$ and $f(n)=0$ for $n \neq Z$ and $n \neq y$. From the usual definition of primitive recursive functions it is easy to see that if $f(n) \neq 0$ only for finite n then f is primitive recursive (see for example [2].)

Then $L(f\circ g)(0)=(u\circ f\circ g\circ v)(0)=u(x_1)\neq u(x_2)=(u\circ f\circ v\circ u\circ g\circ v)(0)$ $=(L(f)\circ L(g))(0)$

so $L(f\circ g)\neq L(f)\circ L(g)$ i.e. $L\notin End\langle PR,o\rangle$. ■

Concerning this lemma the following problem arises:

PROBLEM 2: Are there functions u and v such that for every $f\in PR$:

$$f^{\square}=u\circ f\circ v \quad ?$$

The corollary of the next lemma will be useful for the proof of Theorem I.

LEMMA 2. Let $f\in PR$ and $f(0)=0$ Assume that f^{-1} exists (from this point f^{-1} will denote the usual invers function of f for the operations o, i.e. $f^{-1}\circ f=f\circ f^{-1}=id$.)

Furthermore let $f^{-1}\in PR$.

Then $\exists!$ $g\in PR$ such that $f=g^{\square}$.

PROOF: Let $g=f\circ S\circ f^{-1}$ then $g\in PR$ and $g^{\square}(n)=(f\circ S\circ f^{-1})\circ\underbrace{(f\circ S\circ f^{-1})\circ}_{n\ times}$ $\ldots(f\circ S\circ f^{-1})(0)$ i.e. a suitable g exists.

If $f=g^{\square}$ then $f(n+1)=g^{\square}(n+1)=g(g^{\square}(n))=g(f(n))$ i.e. $f\circ S=g\circ f$ i.e. $f\circ S\circ f^{-1}=g$ i.e. there is only one correct g. ■

COROLLARY: $id=f^{\square}$ $\iff$ $f=S$.

THEOREM I. If $L\in End\langle PR,o,\square\rangle$ then $L=\underline{0}$ or $L=ID$.

PROOF: There are two cases:

a.) $L(id)=id$ and b.) $L(id)\neq id$

CASE a.) $L(id)=id$

Then $id=L(id)=L(S^{\Box})=(L(S))^{\Box}$ and from this we get that $L(S)=S$ using the corollary of Lemma 2. For every $c\in$N and function f from N to N let $f^{c}=\underbrace{f\circ f\circ\ldots\circ f}_{c\ times}$ for $c\neq 0$ and $f^{0}=id$.

For an arbitrary constant function $\underset{\sim}{c}$ $\underset{\sim}{c}=S^{c}\circ\underset{\sim}{0}$ and $L(\underset{\sim}{c})=L(S^{c}\circ\underset{\sim}{0})=$ $L(S^{c}\circ id^{\Box})=L(S)^{c}\circ L(id)^{\Box}=S^{c}\circ id^{\Box}=S^{c}\circ\underset{\sim}{0}=\underset{\sim}{c}$.

Furthermore, for every $f\in PR$ and $c\in$N $\underset{\smile}{f(c)}=L(\underset{\smile}{f(c)})=L(f\circ\underset{\sim}{c})=L(f)\circ\underset{\sim}{c}=$ $=\underset{\smile}{L(f)(c)}$

i.e. $f(c)=L(f)(c)$ which implies $f=L(f)$

i.e. $L=ID$.

CASE b.) $L(id)\neq id$

To spare place, for every $f\in PR$ denote $f'=L(f)$ and let $N'=\bigcup\{R(f')\ :\ f\in PR\}$ i.e. all elements of N contained in $R(f')$ for any $f\in PR$. At first I examine whether N' equals to N or not. For every $f\in PR$ $id\circ f=f$ so $id'\circ f'=f'$ i.e. $\forall c\in$N $id'(f'(c))=f'(c)$. In other words $id'(d)=d$ if $d\in N'$ since $f'(c)\in N'$ or $\forall d\in N'$ $id'(d)=d$. Denote this fact by $id'|_{N'}=id|_{N'}$. Obviously by the definition of N', $R(id')\subseteq N'$. Conversely $\forall d\in N'$ $id'(d)=d$ so $R(id')\supseteq N'$ i.e. $R(id')=N'$.

From this $N'\neq$N follows because $id'|_{N'}=id|_{N'}$ and $id'\neq id$.

Obviously $\underset{\sim}{0}'=(id^{\Box\Box})'=id'^{\Box\Box}=\underset{\sim}{0}$.

At this point we prove the following proposition:

if $f'|_{N'}=g'|_{N'}$ (i.e. $\forall d\in N'$ $f'(d)=g'(d)$) then $f'=g'$.

Since for every $y\in$N $(f')(y)=(f'\ id')(y)=f'(id'(y))=g'(id'(y))=$ $g'\ id')(y)=(g')(y)$ so $f'=g'$.

For every $a\in$N we know that

$$\underset{\sim}{a}'=(S^{a}\circ\underset{\sim}{0})'=S'^{a}\circ\underset{\sim}{0}=\underset{\smile}{S'^{\Box}(a)}=S'^{\Box}\circ\underset{\sim}{a}=S^{\Box'}\circ\underset{\sim}{a}=id'\circ\underset{\sim}{a}$$

i.e. (*) $\forall a\in N$ $id'\circ\underset{\sim}{a}'=id'\circ\underset{\sim}{a}$.

Especially if $a\in N'$ then $\underset{\sim}{a}'=\underset{\sim}{a}$.

Then $id''\tilde{a} = id''\circ\tilde{a}' = (id'\circ\tilde{a})' = (\tilde{a})' = \tilde{a}' = \tilde{a}$

so $id''|_{N'} = id|_{N'} = id'|_{N'}$.

and by the previous proposition $id'' = id'$.

Furthermore for every $f\in PR$ and $y\in N$ $(id'\circ f\circ id')(y)\in N'$ and

$f' = id'\circ f'\circ id'$

so $(id'\circ f\circ id')(y) = [(id'\circ f\circ id')(y)]' = [id'\circ f\circ id'\circ\tilde{y}]' =$

$= id''\circ f'\circ id'\circ\tilde{y}' = id'\circ f'\circ id'\circ\tilde{y}'$.

using (*) we got $= id'\circ f'\circ id'\circ\tilde{y} = f'\circ\tilde{y} = f'(y)$

i.e. $id'\circ f\circ id' = f'$.

We know that $R(id') = N'\neq N$, so $id'\circ id'\neq id$. Moreover $id'(0)=S'^{\square}(0)=0$ in other words $R(id')\ni 0$. If id' is a constant function (in other words if there is one element in $R(id')$ only) then id' must be the function $\tilde{0}$. Then for every $f\in PR$ $L(f)=\tilde{0}$ because $f'=id'$ $f'=\tilde{0}\circ f'=\tilde{0}$ i.e. $L=\underline{0}$.

Now suppose that $id'\neq\tilde{0}$, in other words id' is not constant. So we can apply the Lemma 1 choosing $u=v=id'$ because we have seen, that $f'=id'\circ f\circ id'$ for every $f\in PR$ and $id'\circ id'\neq id$. On the Lemma 1 we can say that $L\notin End\langle PR,\circ\rangle$ so $L\notin End\langle PR,\circ,\square\rangle$. This contradict to our assumtion and this contradiction proves our theorem. ■

The following corollary shows the importance of this theorem:

COROLLARY: Let $g_1,g_2,\ldots,g_k\in PR$ and $0_1,\ldots,0_r$ be operations on PR. Suppose that PR can be generated from the functions $g_1,\ldots,g_k$ with the help of operations $0_1,\ldots 0_r$. Further suppose that there is a finite procedure how to calculate the function $f^{\square}$ from the functions $f\in PR$, $g_1,\ldots g_k$ with the help of the above operations. (See the usual proofs of equivalence of the different forms of primitive recursive functions.)

Let (*) denotes the following condition:

(*) $L\in End\langle PR,0_1,\ldots,0_r\rangle$ and $L(g_i)=g_i$

for $i=1,2,\ldots,k$

If *(*)* holds then $L \in End\langle PR, o, \square \rangle$ so $L = ID$.
The corollary says that our theorem is true in many usual strucutres of primitive recursive functions.

The proof of Theorem I shows that we used only a few properties of our structure $\langle PR, o, \square \rangle$. This implies the following generalization of this Theorem:

THEOREM I.A. Let $\langle P, o, \square \rangle$ be an <u>arbitrary</u> algebraic structure such that the following axioms hold:

a.) $\langle P, o \rangle$ is a semigroup with unit element id

b.) $\exists ! s \in P : s^{\square} = id$

Denote by PS the set of the left-hand-singular elements of $\langle P, o \rangle$ then

c.) $\forall f, g \in P : (\forall c \in PS : f \circ c = g \circ c) \Rightarrow f = g$

d.) $\exists c_o \in PS \ \forall f \in P : f^{\square\square} = c_o$

e.) $\forall c \in PS \ \exists k_c \in \mathrm{N} : c = \underbrace{s \circ s \circ \ldots \circ s \circ}_{k_c \ times} c_o$

Then: if $L \in End\langle P, o, \square \rangle$ and $L(id) = id$ then $L = ID$

PROOF: (only sketch) Analogous to the proof of case a.) of Theorem I:

b.) $\Rightarrow L(s) = s$, d.) $\Rightarrow L(c_o) = c_o$

e.) $\Rightarrow \forall c \in PS : L(c) = c$ and finally from c.) we get $L(f) = f$ for every $f \in P$ i.e. $L = ID$ ∎

This theorem is a generalization of Case a.) only. Theorem I.B below says how to generalize the whole Theorem I in similar way.

THEOREM I.B. Let $\langle P, o, \quad \rangle$ be an <u>arbitrary</u> algebraic structure and suppose that all the axioms a.) - e.) hold.
Suppose that the following axiom hold too.

f.) if $v \circ u \neq id$, $u \notin PS$ and $L(f) = u \circ f \circ v$ for every $f \in P$ then $L \notin End\langle P, o \rangle$.

Then for every $L \in End<P,o,\square>$ $L=ID$ or $L=\underline{0}$

PROOF: (sketch, analogous, to the proof of Theorem I).
There are two cases:
CASE a.) $L(id)=id$, see Theorem I.A.
CASE b.) $L(id)\neq id$.
In this case let $f'=L(f)$ as in the proof of Theorem I.
Let $R(g) := \{g \circ c : c \in PS\}$ for every $g \in P$ (this is the analogue of the range of a function).
Denote by N' the set $\bigcup\{R(g) : g \in P\}$. By the axiom a.) $R(id') = N' \subsetneq PS$ and $id'|_{N'}=id|_{N'}$ and $N'\neq PS$ because $id'\neq id$. By d.) $c'_o=c_o$ and $R(c_o)=\{c_o\}$ because $c_o \in PS$.

<u>Proposition:</u> if $f'|_{N'}=g'|_{N'}$ then $f'=g'$.
Proof: by a.) $f'|_{PS}=g'|_{PS}$ and by c.) $f'=g'$.
Especially for every $c \in PS$ $c'=c$. From this $id''=id'$ follows because $id''|_{N'}=id'|_{N'}$ by a.) and the above proposition. From the above results we get that for every $f \in P$ $f'=id' \circ f \circ id'$, using e.). If $id' \in PS$ then $id'=c_o$, or in other words $L=\underline{0}$.
If $L\neq\underline{0}$ then $id'\notin PS$ and by f.) we get that $L\notin End<P,o>$ i.e. $L\notin <P,o,\square>$. This contradict to our assumption. This contradiction.proves the theorem. ■

At this point some problems arise. For example:
PROBLEM 3. Are the axioms a.) -e.) independent or not?

One can ask similar question about the axioms a.) -f.).
PROBLEM 4. To give more general algebraic form of these theorems above.
Similar problems arise in the next section concerning Theorem II and Theorem III.

GENERATIONS

The main result of this section is Theorem III which is a consequence of Theorem II. Theorem III says that PR^+ and even PR cannot be generated from one function with the help of our operations o and $\Box$. In this section I am dealing with arbitrary functions from N to N, not only functions from PR, except in Theorem III. The proof of Theorem II is made throught some lemmas.

LEMMA 3. Let f be an arbitrary function from N to N. If $f^{\Box}$ is not surjective then $R(f^{\Box})$ is a finite set.

PROOF: It comes from the definition of f that if $f^{\Box}(n)=f^{\Box}(m)$ for any $m>n$ then $R(f^{\Box})=\{f\ (0),\ f^{\Box}(1),\dots,f^{\Box}(m)\}$. ■

REMARK: In the case above $f^{\Box}$ is a periodical function and its period is $m-n$. We can ask whether for every periodic function f there is a function g such that $f=g^{\Box}$. The answer is the following: Let $a_i=f(i)$, $i\in$N and the sequence (a_i) is periodic from the place n and its period is $m-n$ (i.e. $\forall j\in$N $a_{n+j}=a_{m+j}$.) Then there is a function g such that $f=g^{\Box}$ if and only if the numbers $a_0,a_1,\dots,a_{m-1}$ all are distinct and $a_0=0$.
Obviously, $f^{\Box}$ is bijective if and only if $f^{\Box}$ is surjective. Moreover it is easy to see that if g is an injective function, $R(g)\not\ni 0$ then $g^{\Box}$ is an injective one, too. The following lemmas investigate this problem in more details.

LEMMA 4. If f is not injective then $f^{\Box}$ is not surjective.
PROOF: Let $k_1\neq k_2$ and $i=f(k_1)=f(k_2)$.
Suppose that $f^{\Box}$ is surjective.
Then there are h_1, $h_2\in$N such that $k_1=f^{\Box}(h_1)$ and $k_2=f^{\Box}(h_2)$ and $h_1\neq h_2$.

Then $i=f(k_1)=f(f^{\square}(h_1))=f^{\square}(h_1+1)$ and on similar way we get that $i=f^{\square}(h_2+1)$. We know that $h_1+1\neq h_2+1$ and by Lemma 3 R(*f* is a finite set i.e. *f* is not a surjective function. This contradiction proves the lemma. ■

From this point for an arbitrary function *a* from N to N denote by <*a*> the generation of *a* with the help of the operations o and □ i.e. the set of those functions from N to N which we can built from *a* with the help of the above operations in finite steps.

LEMMA 5. If *a* is an arbitrary injective function then for every element *f* of <*a*>

f is injective or *R(f)* is a finite set.

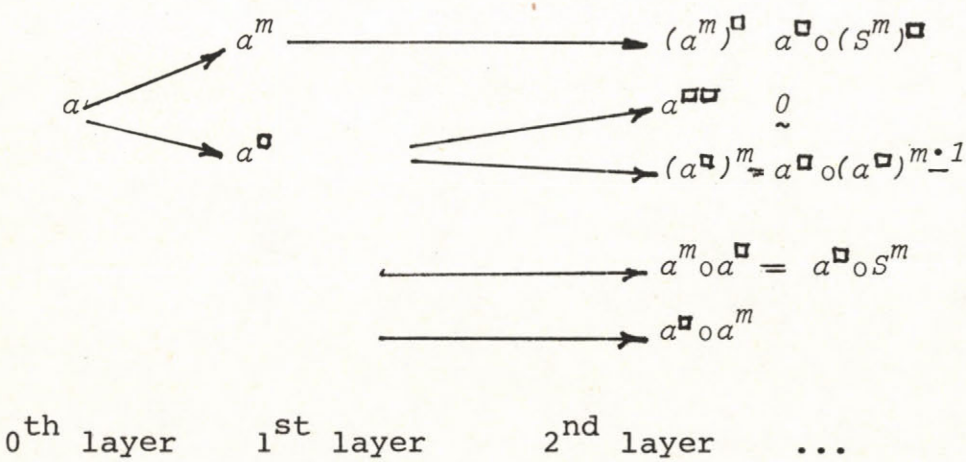
PROOF: Let us investigate how we build the elements of <*a*> in more detail. Then prove the lemma by induction. For every natural number *m* and function *f* from N to N
$f^m \circ f^{\square} = f^{\square} \circ S^m$ and $(f^m)^{\square} = f^{\square} \circ (S^m)$.

Taking these identities into account we get the following scheme, when we construct <*a*> on the basis of the system below:

the 0^{th} layer is $\{a\}$
we get the $r+1^{\text{th}}$ layer on the basis of the followings:
first from the elements of the r^{th} layer
using □ or m^{th} power of o
second from the different elements of the r^{th} layer using o
third from the different elements of the r^{th} layer and layers number less then r using o.

We can see easily with induction on the number of layers that for every element *f* of <*a*> *f* is injective or R(*f*) is a finite set.

(Because if for h and g: h is injective or $R(h)$ is finite and respectively for g then $h \circ g$ and $h^{\square}$ have the same property, too.) ■



Scheme

LEMMA 6. For every element f of $\langle a \rangle$ either there is a suitable k such that $f=a^k$ or $R(f) \subseteq R(a^{\square})$.

PROOF: Similar to the proof of the previous lemma. An induction as above shows how to build the elements of $\langle a \rangle$. We can see easily with induction on the number of layers that for every function f laying in the part of the above figure fenced with dotted line that $R(f) \subseteq R(a^{\square})$. Because for every function g and h $R(h \circ g) \subseteq R(h)$ and $R(g\ \ )$ _ $R(g)$ $\{0\}$ furthermore if $R(h) \subseteq R(a)$ then $R(a^m \circ h)$ _ $R(a^m \circ a^{\square}) \subseteq R(a^{\square})$. The reader can prove this lemma in detail himself.

With the help of above lemmas there is no difficulty in proving our main theorems.

THEOREM II. Let a be an arbitrary function from N to N.

Then

either there is no bijection in $\langle a \rangle$

or $\forall f \in \langle a \rangle$ f is injective or $R(f)$ is finite.

PROOF: If a is injective then see Lemma 5.
If a is not injective then a^m is not injective as well. In this case I prove that there is no bijection in $\langle a \rangle$. Proving by indirect way, suppose that there is a bijective element f of $\langle a \rangle$. Since f is injective $f \neq a^m$ for all $m \in$ N. But f is surjective and by Lemma 6 $a^{\square}$ must be a surjective function.
By Lemma 4 this is a contradiction, which proves the theorem.

From this theorem it is easy to show the main theorem:

THEOREM III. There is no primitive recursive function such that it can generate all the monoton increasing primitive recursive functions even not all the primitive recursive functions. In other words:

$$\neg \exists a \in PR : \quad \langle a \rangle = PR^{+} \quad or \quad \langle a \rangle = PR$$

PROOF: By Theorem II id and $\underline{\cdot}$ (its definition see in the Introduction) cannot be at the same time in $\langle a \rangle$ for every function a from N to N.
This proves the theorem. ■

One can put the following problem similar to Problem 4:

PROBLEM 5. To give more algebraic form of the above theorems in similar way as it was shown in Theorem I.A and I.B. To solve this problem (other exercise) I suggest to take each element f of P as a function mapping from PS into PS: let $f(c) = f \circ c$ for every element c of PS.

PROBLEM 6. We have seen that for example $\langle S\rangle \_ PR^{+}$. So far I have not found any element of $PR^{+}\setminus\langle S\rangle$ yet. So the problem is to show any element of $PR^{+}\setminus\langle S\rangle$.

REMARK: One can investigate other subspaces of PR and prove that they cannot be generated from one element only. For example there is no difficulty in showing that the following subspaces have this property:

$PR_{fin} = \{ f$: $R(f)$ is a finite set }

$PR_{n} = \{ f$: in $R(f)$ there are at most n elements }

$PR_{inj} = \{ f$: f is injective or $R(f)$ is finite }

$PR_{con} = \{ c, c^{\square} : c \in N \}$

$PR_{per} = \{ f$: f is periodic }

$PR_{pern} = \{ f$: f is periodic and its period contains no more than n elements }

$PR_{o} = \{ f$: $f(0)=0\} = Ker\,\square$

$PR_{bij} = \{ f$: $f(0)$ and f is bijective }

PRsurj= { f : f is no surjective }

REFERENCES

[1] ROBINSON, R.M. Primitive recursive functions. Bull. Amer. Math. Soc. 1 (1947), 925-942.

[2] PETER,R. Recursive Functions. Akadémia, Budapest, 1967.

[3] ROBINSON,J. General recursive functions. Proc. Amer. Math. Soc. 1 (1950), 703-718.

[4] MONK, D.J. Mathematical Logic. Springer-Verlag, New-York, 1976.

[5] SHOARE, R.I. Recursive enumerable sets and degrees. Bull. Amer. Math. Soc. (1978. nov.) 1149-1162.

[6] HERRINGTON,L. et SHOARE, R.A. Definable degrees and automorphisms of ν. Bull. Amer. Math. Soc. (1981. jan.) 97-99.

A primitiv rekurziv függvények algebrai strukturájáról

Szalkai István

Összefoglalás

E cikkben a primitiv rek. fv.-ek halmazát algebrai strukturaként tekintettem: a kompozició (o) és a 0-tól való iteráció (□) müveletekkel.

Belátom, hogy e strukturában az ID és 0 endomorfizmusokon kivül nincsen más endomorfizmus (ld. I. Tétel).

Ezután részletesen megvizsgálom, hogy N^N mely részhalmazai generálhatók egy függvénnyel a fenti két operáció segitségével. (ld. II. Tétel).
Ebből speciálisan adódik, hogy a prim. rek. függvények halmaza nem generálható egy függvénnyel, (ld. III.Tétel). Néhány idevágó problémát is emlitek, a tételek algebrai általánositásai után.

АЛГЕБРАИЧЕСКАЯ СТРУКТУРА ПРИМИТИВНЫХ РЕКУРСИВНЫХ ФУНКЦИЙ

И. Салкаи

Резюме

Статья занимается множеством примитивных рекурсивных функций как алгебраической структурой. В этой структуре существуют две операции: композиция (o) и итерация с места 0 / □ , см. во Введении/. Доказывается, что над этой структурой существуют только два эндоморфизма: ID и 0. /см. Теорема I./ После этого статья занимается вопросом: какие подмножества N^N /функции из N в N / можно получить из одной функции. /см. Теорема II/. Из этой теоремы получается важная Теорема III: подмножество примитивных рекурсивных функций нельзя составлять из одной функции. Кроме этих доказываются теоремы в общей алгебраической форме и дано несколько проблем.

Közlemények 29/1983 93-114

PLANNING PERENNIAL CROP PRODUCTION IN AGRICULTURAL INVESTMENT PROJECTS

Elmira Morin Tejado
Perfecto Dipotet Barcada

Instituto de Matematica Cibernetica y Computacion
Academia de Ciencias de Cuba

INTRODUCTION

The main purpose of this paper is to describe a model which, we hope, will be a working tool for the managers of the citrus enterprises in our country. Managers may obtain with the model different long range alternatives for the development of their enterprises. These alternatives allow them to plan in advance: the investments policy; the principal material resources needed; the labour force required, mainly the qualified one, and the finantial policy for the planning horizon.

In the first part of the paper /Problem Description/ we present the main tasks to be solved by the model, the constraints derived from the socio-economical environment and the particular requirements of the end users.

In the second part /Model construction/ we explain the algorithm we have followed in order to obtain the groups of parcels with similar description /taxons/ within the Enterprise. Next we present the model describing the process for the production of citrus, the main normative and technological parameters and also the material and economical contraints. Last, we give a short description of the algorithm used for the microlocalization of economical objects within the enterprise.

In the third part /Result Analysis/ we briefly describe the principal output of the model and the problems we plan to solve in the near future.

This paper is part of a research program /DIPOTET-81/ related to the long range development of a new agricultural region in a developing country.

PROBLEM DESCRIPTION

In Fig. 1. we present the main blocks appearing in the process for the production of citric fruits and citric byproducts. It is possible to see the rather big number of operations to be done in order to get the fruit to the end users. It is demanded from us to develop a model giving different alternatives for the long range development plan of the enterprise. It is known, that for the planning period /T/ the demand of citrics is higher than the production possibilities óf the region.

The main optimization criterium is"the maximum profit for the planning period" T.

A feasibility study was carried out in order to identify the particular conditions and actual possibilities of the enterprise and its environment. Results of the former study are:

- There are different kinds of citric fruits planted, with not the same technology, in different types of soils, indifferent years, etc. The parcels with the same former characteristics must be grouped together, looking for model simplification and better understanding of the enterprise development process.

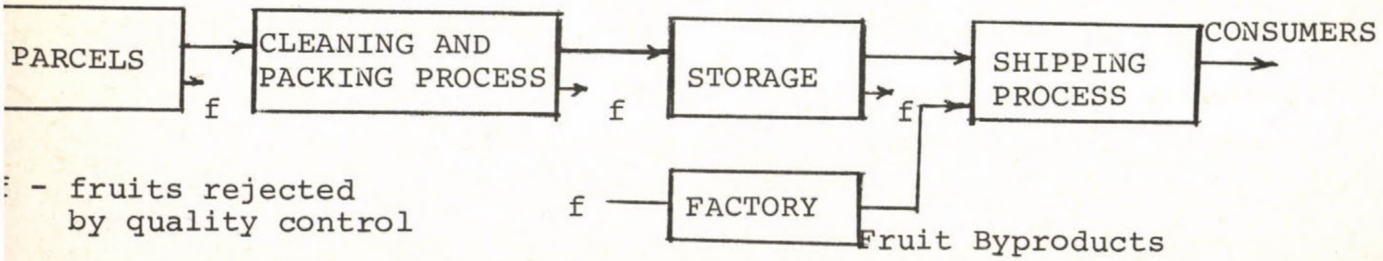
- Five years after plantation it is possible to get fruits from citrics trees and during the next seven years the productivity increases in a more or less linear way. After the 12th year this productivity remains approximately constant until the 40th year. Cultural and agrochemical attention to parcels change according to their age and fruit production

- The planning horizon T is less than 20 years. It is not possible to plant more than S_a ha/year, and it is not possible to invest more than I_g pesos on each five year plan. S_T is the total surface available for planting. The enterprise normative and technological parameters are known. Actual prices for fruits and by products and their trends are also known.

- The enterprise development budget can not be less than 10 % of the net profit. The enterprise must also support the social infrastructure needed for the workers and their respecitve families.

- It is strongly recomended not to introduce new forms for data gathering because people reject these changes, and it is then needed to wait a rather long time in order to overcome this difficulties.

- Model outputs must be: the fruit and by product production per year; the enterprise investments in the planning period T; the labour force required for the proposed plan; the places where the factories, workshops, buildings, etc.; will be built; and, in general, information concerning the enterprise development policy in T.



Citrics Production Process

MODEL CONSTRUCTION

In order to fulfill user requirements, we must perform the following tasks: classification of citrus parcels; model for planning the long range production of fruits and fruit by products and a model for the microlocalization of enterprise socio-economical objects /buildings, factories, workshops, etc./.

Classification of Citrus Parcels

In developing countries research and development in food production find generally a lot of problems bacause some times we have not enough qualified people and/or finantial resources and also because we have not easy access to modern technology. For these reasons, enterprise managers employed all available technology and method trying to overcome actual difficulties in their production plan. The net result of that situation is that there is not uniformity int he technological and normative parameters and then it is necessary to classify the parcels and group together those with the same description.Next we present the classification process:

- Let $c=(c_1,\dots,c_n)$ be a set of parcels planted with citrus trees.

- Let $A=(a_1,\dots,a_m)$ be a set of attributes related to citrus parcels. For example: age; distance between planted trees; type of soil; parcel area; planting technology, etc.

- Let $V=(v_1,\dots,v_l)$ be the set of values the attributes can take.

- We define the descriptor d_{ij} as a pair /attributes value/ (a_i, v_j) $i \in I = 1,\ldots,m$; $j \in J = 1,\ldots,\ell$.

- A description $Y(c_k)$ of the parcel $c_k, k \in K=1,\ldots,n$; is the string $d_{1b}\ d_{2c} \ldots d_{m_f}$; $b,c,\ldots,f \in V$.
  In this paper, two parcels c_j, c_k; are equals if
  $Y(c_j) = Y(c_k)$; $j,k \in K$
  one taxon is a set of parcels $c_j, c_k,\ldots,c_p$; $j,k,p \in K$;
  having equal descriptions. It means:
  $Y(c_j) = Y(c_k) = \ldots = Y(c_p)$

- If parcels c_j and c_k have the same description $Y(c_j) = Y(c_k)$, their respective productivities /Tons/area ought to be approximately the same. That's why we use the attribute parcels productivity for validating the classification process.

- A computer program have been developed to list all parcels belonging to each one of the taxons and also to check descriptors for noise and redundancy /DIPOTET-80/. If parcel productivity /Ton/ha/ is not approximately constant, within each taxon then we must look for errors in data collection and, eventually, to improve the classification process if needed. Enterprise managers ought to take some decisions with parcels whose production is completely outside of expected values.

- Sometime it is convenient to group taxons together according to some given criteria to simplity the enterprise structure and management information system. This approach is used in the following model.

Model for Planning the Production of Citrics.

First we describe the process relating /DIPOTET-81/, parcel production usage.

Let:

$I = (70, 71, \ldots, 85)$ be the set of years when trees are planted;

$J = (1, \ldots, s)$ be the set of parcels available for the production of citrics in the given region;

$Z = (1, \ldots, T)$ be the set of time periods where T is the planning horizont;

$x_j(t)$= be the number of hectareas used for j type citrics at time period t, $j \in J$ and $t \in \mathcal{T}$

$k_j^+(t)$= be the number of hectareas used for new plantings of j type citrics at year t, $j \in J$; $t \in \mathcal{T}$;

k_j^- t = be the number of hectareas of perennial of type j removed at year t, $t \in \mathcal{T}$;

b_{jk} = proportion of lands with trees planted at year k progressing to year j, $J > K$; $j, k \in J$.

The state equations are then defined as

$$x_j(t+1) = \sum_{k=1}^{s} b_{j_k} \, x_k(t) + k_j^+(t) - k_j^-(t)$$

or in matrix form

$$x(t+1) = Bx(t) + k^+(t) - k^-(t)$$

where $x(t) = \big(x_1(t), \ldots, x_s(t)\big)$ is the state vector and

$k^+(t) = \left(k_1^+(t), \ldots, k_s^+(t)\right)$; $k^-(t) = \left(k_1^-(t), \ldots, k_s^-(t)\right)$

are the control vectors.

Next we present a table relating state equations, age and production/year.

	Age of trees	Production/ha
$x_1(t)$	0-1	
$x_2(t)$	1-2	
$x_3(t)$	2-3	
$x_4(t)$	3-4	
$x_5(t)$	4-5	a_o
$x_6(t)$	5-6	a_1
$\vdots$	$\vdots$	$\vdots$
$x_\omega(t)$	$5 < \omega < 13$	$(1+\alpha)^{\omega-6}\, a_1$
$x_\omega(t)$	$12 < \omega < 40$	$(1+\alpha)^6\, a_1$

The state equations for new plantings are:

$$x_1(t+1) = k_1^+(t)$$

The trees in the second year

$$x_2(t+1) = b_{21}(t)$$

$$\vdots \qquad \vdots$$

$$x_{12}(t+1) = b_{12-12}\, x_{12}(t) + b_{12-11}\, x_{11}(t)$$

Model Parameters:

a_{ijt} = production in year $t, t \in \mathcal{T}$, for one ha planted at year $i, i \in I$, for j type parcel, $j \in J$.

c_{ijt} = processing cost at year t, for one ha planted at year $i, i \in I$, for j type parcel, $j \in J$;

c_{ij} = $\sum_{t=83}^{83+T} c_{ijt}$ integrated cost related to processing one ha planted at year $i, i \in I$, for j type parcel, $j \in J$;

p_{kt} = price for one Ton. of purchased production of product $k, k \in K = (1, \ldots, g$, at year $t, t \in \mathcal{T}$;

s_{ij} = area for j type parcel planted at year i, $i \in I$;

s_i = area of new parcels ready to be planted at year $i, i \in I$;

Q^t = sinking funds at year $t, t \in \mathcal{T}$, for the factory processing the fruits;

q^t = sinking funds /for unity of capacity/ for the enlargement of the factory at year $t, t \in \mathcal{T}$

M = total capacity for the factory at the beginning of the planning period;

w_l = actual processing capacity for product $l, l \in K$, for the factory at the beginning of the planning period;

d_{lk} = needed increment in l type capacity when k type production is incremented in one unity, $l, k \in K$

i_{lk} = intensity of k type production, $k \in K$, demanding l type capacity increment for the next period;

f_{ij} = labour force related to one ha of j type parcel planted at year $i, i \in I$; $j \in J$;

f_k = labour force related to the production of one Ton of k type product;

e_{jm} = quality of m type product, $m \in M = \{1, \ldots, g\}$, used in one ha of j type parcel;

f_t = labour force available at year $t, t \in \mathcal{T}$;

e_{mt} = volume of m type product, $m \in M$, available at year t, $t \in \mathcal{T}$.

r_t = contribution, in peso/worker, at year $t, t \in \mathcal{T}$, for social infrastructure;

α_t = development fund at year $t, t \in \mathcal{T}$;

G_T = Total profit in the planning period, $T \in \mathcal{T}$;

Decision Variables:

x_{ij} = area for j type parcels planted at year i, $i \in I$; $j \in J$;

x_i = area ready to be planted at year i, $i \in I$;

y_{kt} = volume of product k, $k \in K$, produced at year t, $t \in \mathcal{T}$;

f_t = labour force needed at year t, $t \in \mathcal{T}$;

e_{mt} = volume of m type product, $m \in M$, needed at year t, $t \in \mathcal{T}$;

y_t = volume of fruits produced at year t, $t \in \mathcal{T}$;

z_k^T = enlargement of the capacity of the processing factory at T years;

V_t = profit obtained by fruit production and processing at year t, $t \in \mathcal{T}$;

N_t = investments at year t, $t \in \mathcal{T}$; for the enlargement of social infrastructure;

Constraints:

$$\sum_{i=83}^{90} x_{ij} \leq s_j \; ; \quad i \in I, \quad j \in J;$$

Conditions constraining j type parcels planted at i;

$$x_i \leq s_i, \quad i \in I;$$

Conditions constraining new parcels planted at i;

$$f_t \leq f_t' \quad , \quad t \in \mathcal{T}$$

Conditions constraining needed labour force at year t,

$$e_{mt} \leq e_{mt}' \quad , \quad t \in \mathcal{T}$$

Conditions constraining m type products needed at year t;

$$\sum_{k=1}^{q} p_{kt} y_{kt} - \sum_{i=83}^{t} \sum_{j \in J} (c_{ijt} + c_{it}) x_i - \sum_{t'=83}^{t} q^{t'} z^{t'} - V^t = Q^t$$

$$t', t \in \mathcal{T}$$

balance conditions to form profits resulting from production, processing and purchase of the fruits at year $t, t \in \mathcal{T}$;

$$\sum_{i=83}^{t} \sum_{j \in J} (a_{ijt} x_{ij} + a_{jt} x_j) - y^t = 0 \qquad t \in \mathcal{T};$$

conditions to produce the volume of fruits at year t;

$$y_t - \sum_{t'=83}^{t} z^{t'} \leq M, \qquad t \in \mathcal{T};$$

conditions for the factory to process the volume of fruits produced;

$N_t - f_t \cdot r_t = 0$

balance condition for social infrastructure;

it is demanded to maximize the general profit for the planning period. It means:

$$\sum_{t=83}^{83+T} p_t y_t - \sum_{i=83}^{90} \left(\sum c_{ij} x_{ij} + c_i x_i \right) - \sum_{t'=83}^{90} q_{t'} z_{t'} - \sum_{t=83}^{t+T} N_t - Q \rightarrow max$$

MICROLOCALIZATION OF SOCIO-ECONOMICAL OBJECTS

The elements of the general problem are:

- a set $J = (1, \ldots, n)$ of raw materials sources;
- a set $I = (1, \ldots, m)$ of points where it is possible to place the enterprises for processing the raw materials;
- a set $\tau = (1,2,\ldots,T)$ of time periods, where T is the planning period;
- a known function $b: J \longrightarrow R$, whose values b_j^t represent the volumes of raw materials coming from the sources $j, j \in J$, at time $t \in \tau$;
- an unknown /a priori/ function $X: I \longrightarrow R$ whose value X_i^t is the capacity of point $i, i \in I$, for processing raw material al time $t \in \tau$;
- a function $x: I \times J \longrightarrow$ such that x_{ij}^t is the quantity of raw material from source j, $j \in J$, to be elaborated in point i, $i \in I$, at time $t \in \tau$;
- a function $c: I \times J \longrightarrow R$ transportation cost of raw material from j, $j \in J$, to i, $i \in I$, at time $t \in \tau$; if $d: I \times J \longrightarrow R$ is the distance matrix for points i and j, then, generally, the transportation point is proportional $(p: R \longrightarrow R)$ to the distance and $c_{ij} = p \cdot d_{ij}$;
- a function $x: I \times J \longrightarrow R$ such that x_{ij}^t is the quantity of raw material from source j, $j \in J$, to be elaborated in point i, $i \in I$, at time $t \in \tau$;
- a known function $T: I \longrightarrow R$ whose value T_i is the building cost for object i, $i \in I$;

- known function K: $I \longrightarrow R$ whose value K_i^t is the processing cost of raw material unity in the object i, $i \in I$, at time $t \in \mathcal{T}$;
- $g_i(X_i^t)$ are the building and processing cost for the enterprise i, $i \in I$, at time t, $t \in \mathcal{T}$, depending on capacity X_i;

Let $X_i^t = \sum_{j \in J} x_{ij}^t$ then the task of optimal object microlocalization for a given region may be formulated as:

to determine, $\min\limits_{x_{ij}^t} \sum_{i \in I} \sum_{j \in J} \sum_{t \in \mathcal{T}} c_{ij}^t x_{ij}^t + \sum_{i \in I} \sum_{t \in \mathcal{T}} g_i(X_i^t)$;

the constraints are:

$$\sum_{i \in I} x_{ij}^t = b_j^t$$

$$X_i^t \equiv \sum_{j \in J} x_{ij}^t \leq a_i^t$$

$$x_{ij}^t \geq 0$$

If functions $g_i(X_i^t)$ are linear, then we face a transport dynamic task and to solve it we apply the linear programming methods. Nevertheless, generally, these functions $g_i(X_i^t)$ are not linear, but discrete discontinuoa ones, difficulting the solution for this task in the general case.

If $g_i(X_i^t) = (K_i X_i^t + C_i)$ Sign X_i^t /DIPOTET-81/ then our task will be reduced to:

$$\min_{x_{ij}^t} \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} (c_{ij}^t + K_i^t)\, x_{ij}^t + \sum_{i \in I} T_i \text{ Sign } X_i^t$$

$$\sum_{i \in I} x_{ij}^t = b_j^t$$

$$\sum_{j \in J} x_{ij}^t \geq a_i^t;\ x_{ij}^t \geq 0$$

This task is a multi-extremes one in non-linear programming. It may be solved using combinatorical methods, using the "branch and bound" method. However, for using these methods in medium size problmes, powerful computers are needed.

For long range planning, in our case, it is possible to use a simplified model not considering capacity (X_i^t) constraints.

Then, our task will be the following:

to determine,

$$\min_{x_{ij}^t} \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} d_{ij}^t\, x_{ij}^t + \sum_{i \in I} T_i \text{ Sign } X_i^t$$

$$\sum_{i \in I} x_{ij}^t = b_j^t,\ x_{ij}^t \geq 0$$

were $d_{ij}^t = c_{ij}^t + K_i^t$

Next we present the combinatorical version for the former problem statement.

Let $w \subset I$ be a subset where we suppose must be built the objects to microlocalize.

Then, on the set of all subsets $w \subset I$ it is possible to define function $P(w)$ in the following way:

$$P(w) = \min_{x_{ij}^t} \sum_{i \in w} \sum_{j \in J} \sum_{t \in \tau} d_{ij}^t \, x_{ij}^t + \sum_{i \in w} T_i ;$$

$$\sum_{i \in w} x_{ij}^t = b_j^t$$

$$x_{ij}^t \geq 0, \; i \in w$$

There are not constraints linking the variables by t, then we may write:

$$P(w) = \sum_{t \in \tau} \min_{x_{ij}^t} \sum_{i \in w} \sum_{j \in J} d_{ij} \, x_{ij}^t + \sum_{i \in w} T_i$$

There are not constraints of the type $x_i^t \leq a_i^t$ then it is possible to write:

$$P(w) = \sum_{t \in \tau} \sum_{j \in J} b_j^t \min d_{ij}^t + \sum_{i \in w} T_i$$

$$P(w) = \sum_{t \in \tau} \sum_{j \in J} \min_{i \in w} b_j^t \, d_{ij}^t + \sum_{i \in w} T_i$$

To solve this task it is needed: to build T matrix

$\left\| b_j^t \; d_{ij}^t \right\|_{mxn}$; then, to find $\sum_j \min b_j^t \, d_{ij}^t$ for every t, $t \in \mathcal{T}$; then to sum up /index t/ and to add $\sum_{i \in w} T_i$.

In the most simple case, when $d^t_{ij} = d_{ij}$ /it means $c^t_{ij} = c_{ij}$; $K^t_i = K_i$/ it is possible to write:

$$P(w) = \min_{x^t_{ij}} \sum_{i \in w} \sum_{j \in J} d_{ij} \sum_{t \in \mathcal{T}} x^t_{ij} + \sum_{i \in w} T_i$$

We have not constraints $X^t_i \leq a^t_i$, then x^t_{ij} for every t must take value 0 or /exlusive/ b^t_j, because of constraints.

$$\sum_{i \in w} x^t_{ij} = b^t_j$$

Then, to determine the P(w) value it is possible to use the following

$$P(w) = \sum_{j \in J} \min d_{ij} \left(\sum_{t \in \mathcal{T}} b^t_j \right) + \sum_{i \in w} T_i$$

Now, the computations to find P(w) are very simplified because the task is now a not dynamic one and it is possible to calculate

$$P(w) = \sum_{j \in J} \min_{i \in w} d_{ij} \bar{b}_j + \sum_{i \in w} T_i ; \qquad (I)$$

here $\bar{b}_j = \sum_{t \in \mathcal{T}} b^t_j$ is only once calculated before computing the P(w) values for every $w \subset I$.

We remark that in this simple case P(w) is related to distance and building cost and /I/ fulfill /for the moment/ the requirements of our problem /objects micro-localization in a developing region/. Program for the solution of this particular case may be found in /Peña-Viesielovski, 81/.

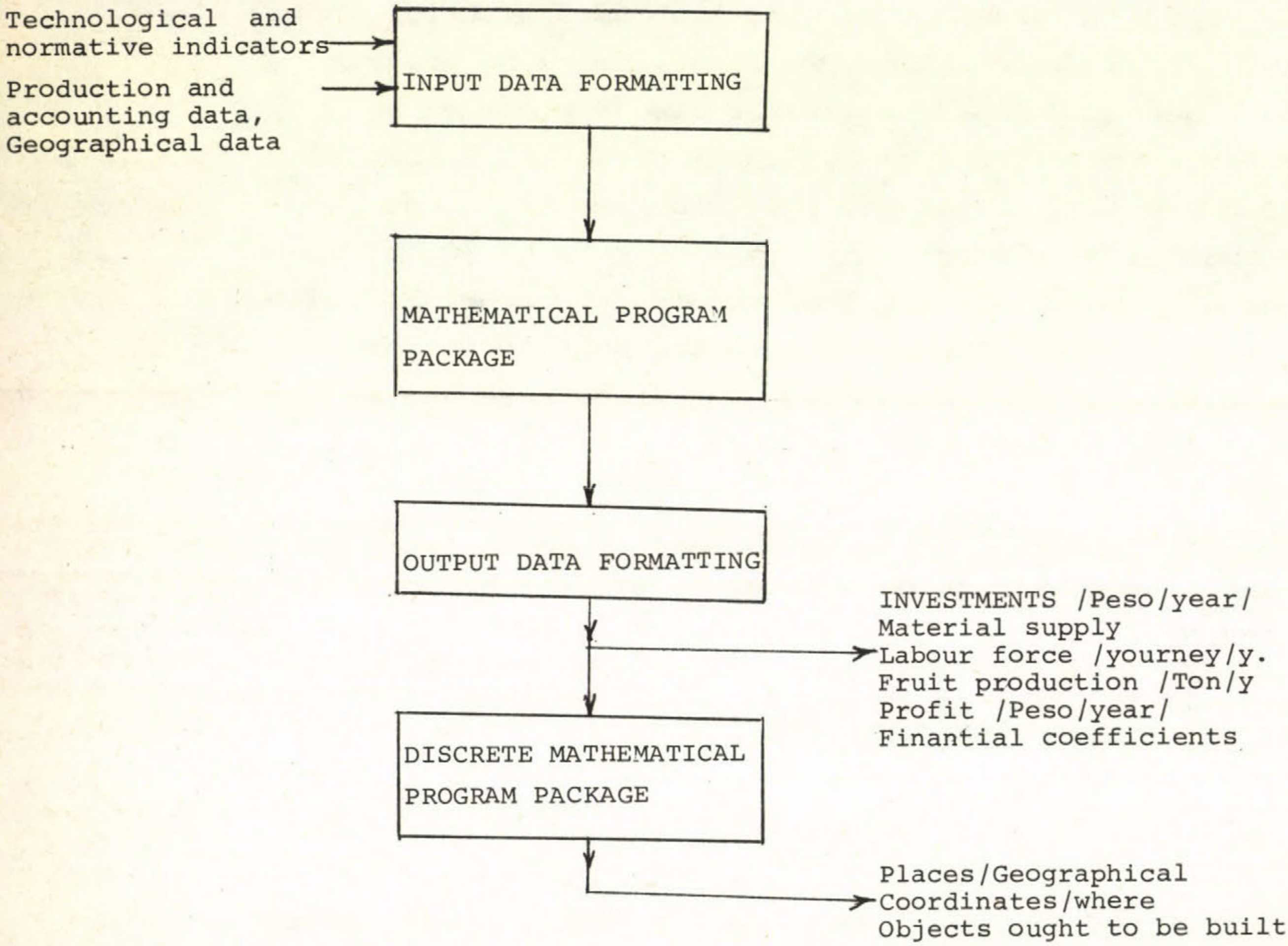
RESULT ANALYSIS

In Fig. 2 we present the complete model with its input and output information. Two data formatting programs have been developed to get the linear programming package to accept data and to produce tables according to users preferences and past experiences. Input information is collected from the same forms /without changes/ officially implemented in the enterprise. The output information consists os the main indicators required for the long range Enterprise plan and is presented in the demanded format.

The main results of the model are:

- It is possible to obtain the investment policy for the next T years.
- It is possible to detect bottlenecks and critical parameters in the enterprise.
- It is possible to calculate the finantial indicators. /profit/year, production cost, cost/peso, salaries, etc./ and to monitor them in the planning period.
- It is possible to change cofficients and parameters looking for better solutions /technological and finential/ in the planning period.
- The model is a working tool for managers and must be improved with the further development of the enterprise information basis.

This model is part of a management information system for citrus enterprises /DIPOTET-83/ being developed in the Cuban Academy of Sciences.



Main Blocks of the Model

Fig. 2.

REFERENCES

DIPOTET P., JACHATUROV V.R.-81: PLANNING A RESEAECH PROGRAM FOR THE DEVELOPMENT OF AN AGRICULTURAL REGION
IFIP Working Conference on Computers Applications in Agricultue and Food Production.
Habana, October 1981.

DIPOTET P.-81: ELABORACION DE UN SISTEMA HOMBRE-MAQUINA PARA LA PROYECCION DEL DESARROLLO DE NUEVOS TERRITORIOS
T-319-04. Estudio de Orientacion
IMACC. Academia de Ciencias de Cuba.

MORIN E., DIPOTET P.-82: MODELO PARA LA PLANIFICACION A LARGO PLAZO DE UNA EMPRESA PORCINA.
IMACC. Academia de Ciencias de Cuba.

VIESIEWOVSKI V. ET AL.-82: ALGORITHMS AND PROGRAMS FOR OBJECT MICROLOCALIZATION.
Technical Report. IMACC. ACC Habana. Cuba.

Összefoglalás

Évelő növények hozamának tervezése

A dolgozat egy matematikai modellt ir le a citrusfélék hozamának tervezésére. Gazdasági megfontolásokból következik az elérendő célok és kényszeritő körülmények matematikailag leirható rendszere. A leirt algoritmus az egyes növényeket hasonló tulajdonságaik alapján cluster-ekbe sorolja. Kapcsolódó probléma a termelést alkotó objektumok optimális elhelyezése. A szerző ezt matematikai programozási feladatra vezeti vissza. A matematikai modell számitógépes megoldásával foglalkozik a dolgozat befejező része.

Планирование урожайности многолетних растений в сельскохозяйственных проектах

В статье описывается математическая модель планирования урожайности цитрусовых. Из экономических соображений следует система достижимых целей и принуждающих условий. Эта система описывается математическими средствами, образуются кластэры типов растений на базе близких друг к другу свойств. Тесно связана с проблемой урожайности проблема обработки. Отптимальность местоположения объектов процесса обработки решается путем математического программирования. Реализация алгоритмов решения возможна с помощью вычислительной машины.

Közlemények 29/1983 115-121

MINKOWSKI'S CONVEX BODY THEOREM AND THE MEASURE OF COVERING R^n BY A SET

Béla UHRIN

Computer and Automation Institute
Hungarian Academy of Sciences
1502 Budapest, Pf.63, Hungary

Let Λ R^n be a (geometric) lattice and let $A:=(a_1,a_2,\ldots,a_n)$, $a_i \in R^n$, $i=1,\ldots,n$, be a basis of Λ, i.e. $\Lambda=\{u\in R^n:\ u=\Sigma_{i=1}^n\ u_i\ a_i,$ u_i *integer*, $i=1,\ldots,n\}$. The set $P:=\{x\in R^n : x = \Sigma_{i=1}^n\ \lambda_i\ a_i,$ $0\leq\lambda_i<1,\ i=1,\ldots,n\}$ is the unit cell of A. Denote by $\mu(S)$ the Lebesgue measure of S R^n and by $|S|$ the cardinality of the finite set S. The zero vector of R^n is denoted by θ. By $S+H$ we mean the algebraic (or Minkowski) sum of S,H R^n, i.e. $S+H :=$ $\{x\in R^n : x=s+h,\ s\in S,\ h\in H\}$. The number $d\Lambda:=|det(a_1,\ldots,a_n)|$ is the determinant of Λ, we assume that $d\Lambda>0$.

A generalized form of Minkowski's convex body theorem asserts that if $K\subset R^n$ is convex and θ-symmetric (i.e. $K=-K$) and $(\mu(\frac{1}{2}K)/d\Lambda)>1$, then $|K\cap\Lambda|>1$.

The aim of this note is to show that it is not the ratio $(\mu(\frac{1}{2}K)/d\Lambda)$ that decides upon the number of lattice points being in K, but rather the ratio $(\mu(\frac{1}{2}K)/c(\frac{1}{2}K,\Lambda))$, where $c(\frac{1}{2}K,\Lambda)$ $(\leq d\Lambda)$ is a number that might be called "the measure of covering R^n by $\frac{1}{2}K$".

Before the exact definition of this number, let us recall that the pair $\{S,\Lambda\}$, $S\subset R^n$, is called an "(S,Λ)-covering of R^n" if

$$(1)\qquad \bigcup_{u\in\Lambda} (S+u) = R^n.$$

AMS Classification Number 1980: 10E05, 10E30.

The ratio $(\mu(S)/d\Lambda)$ is called the density of (S,Λ)-covering (see [1], p. 182.).

Definition 1. We say that $\{S,\Lambda\}$ is an "almost (S,Λ)-covering of R^n" if

$$(2) \qquad \mu(R^n \setminus \bigcup_{u\in\Lambda} (S+u))=0. \qquad \square$$

The condition (2) is equivalent to

$$(3) \qquad \mu(P \setminus \bigcup_{u\in\Lambda} ((S+u)\cap P))=0$$

(an excercise for the reader).

Denote

$$(4) \qquad S_P := \bigcup_{u\in\Lambda} ((S+u)\cap P).$$

Assertion 1. Let $A:=(a_1,\ldots,a_n)$ and $A' := (a_1',\ldots,a_n')$ be two bases of Λ and P and P' be unit cells of A and A', respectively. Then

$$(5) \qquad \mu(S_P) = \mu(S_{P'}),$$

consenquently

$$(6) \qquad \mu(P\setminus S_P) = \mu(P' \quad S_P') . \qquad \square$$

Proof:

$$\mu(S_P) = \mu(\bigcup_{v\in\Lambda}((\bigcup_{u\in\Lambda}((S+u)\cap P))\cap(P'+v))) =$$

$$= \mu(\bigcup_{w\in\Lambda} \quad \bigcup_{r\in\Lambda}((S+r)\cap P'\cap(P+w))) = \mu(S_{P'})$$

and $\qquad \mu(P) = \mu(P') = d\Lambda. \qquad \blacksquare$

Definition 2. The measure $\mu(S_P)$ is called "the measure of covering R^n by S" and is denoted by $c(S,\Lambda)$ ((5) shows that it depends only on S and Λ). □

Assertion 2. The pair $\{S,\Lambda\}$ is an "almost (S,Λ)-covering of R^n" if and only if $c(S,\Lambda)=d\Lambda$. □

Proof: Trivial consequence of definitions. ■

This assertion justifies the name given to $c(S,\Lambda)$ in Def.2. Denote by φ_Λ the canonical map of R^n onto the torus group $T:=R^n/\Lambda$. There is an interesting connection between $c(S,\Lambda)$ and φ_Λ, namely we have.

Assertion 3. For any L-measurable set $S\subset R^n$

$$(5') \qquad c(S,\Lambda) = \bar{\mu}(\varphi_\Lambda(S)),$$

where $\bar{\mu}$ is the measure on T generated by μ. □

Proof: Denote by ψ_P the isomorphism of T onto P. The measure $\bar{\mu}$ on T is then defined as $\bar{\mu}(H) := \mu(\psi_P(H))$, H T (this does not depend on P). Now, we can see easily that $S_P=\psi_P(\varphi_\Lambda(S))$. ■

Now, we have

Theorem 1. Let $K\subset R^n$ be convex and θ-symmetric. If $(\mu(\frac{1}{2}K)/c(\frac{1}{2}K,\Lambda))>m$ (where m is a positive integer), then K contains at least m pairs of non-zero lattice points u_i, $-u_i\in\Lambda$, $u_i\neq\theta$, $i=1,\ldots,m$, such that all members of the set $H:=\{u_1,\ldots,u_m,-u_1,\ldots,-u_m\}$ are mutually different (i.e. $|H|=2m$). □

Proof: We shall prove that

$$(7) \qquad \tfrac{1}{2}(1+|K\cap\Lambda|) \geq (\mu(\tfrac{1}{2}K)/c(\tfrac{1}{2}K,\Lambda)).$$

The proof of (7) depends on the following two trivial facts:

- For any sets $S,H\subset R^n$ of finite cardinality

(F1) $$|S+H| \geq |S| + |H| - 1.$$

- For any Lebesgue-measurable set $S\subset R^n$

(F2) $$\mu(S) = \int_P |(S-x)\cap\Lambda| \; dx .$$

As to the proof of (7), first we can easily see that

(8) $$(\tfrac{1}{2}K)_P := \bigcup_{u\in\Lambda} ((\tfrac{1}{2}K+u))\cap P)=\{x\in P: (\tfrac{1}{2}K-x)\cap\Lambda\neq\emptyset\}.$$

Secondly, for any $x\in(\tfrac{1}{2}K)_P$ we have

(9) $$K\cap\Lambda = ((\tfrac{1}{2}K-x)-(\tfrac{1}{2}K-x))\cap\Lambda \supseteq ((\tfrac{1}{2}K-x)\cap\Lambda)-((\tfrac{1}{2}K-x)\cap\Lambda).$$

[Here we used the convexity and θ-symmetry of K, i.e. $\tfrac{1}{2}K-\tfrac{1}{2}K=K$. The assumption $\mu(K)>0$ implies, using (F2), that $\mu((\tfrac{1}{2}K)_P)>0 (\Rightarrow (\tfrac{1}{2}K)_P\neq\emptyset)$.]
Using (F1), the relation (9) implies

(10) $$|K\cap\Lambda| \geq 2|(\tfrac{1}{2}K-x)\cap\Lambda|-1.$$

Integrating both sides of (10) over $(\tfrac{1}{2}K)_P$, using (8), (F2) and taking into account that $\mu((\tfrac{1}{2}K)_P)=c(\tfrac{1}{2}K,\Lambda)$, we get (7). To finish the proof, (7) imply

(11) $$q := |K\cap\Lambda| \geq 2m.$$

Write $K\cap\Lambda$ in the lexicographically increasing order, say $u_1<u_2<\ldots<u_m<u_{m+1}<\ldots<u_{q-m}<u_{q-m+1}<\ldots<u_{2m}<\ldots<u_q$. Clearly $\theta\in K\cap\Lambda$, i.e. $u_r=\theta$ for some $1\leq r\leq q$.

This means that there are exactly $r-1$ elements $u_k<\theta$ and exactly $q-r$ elements $u_i>\theta$. Assume that $r\leq m$. Inequality (11) implies $q-r\geq q-m\geq m$. But $K\cap\Lambda$ is θ-symmetric, hence $q-r\geq m$ implies that at least m elements $u_k<\theta$ belong to $K\cap\Lambda$ (all $-u_i$) and this is a contradiction (because there are exactly $r-1\leq m-1$ such elements only). We come similarly to a contradiction if we assume that $r \geq q-m+1$. Hence

$$(12) \qquad m+1 \leq r \leq q-m,$$

showing that we have at least m mutually different elements $u_i < \theta$, so $-u_i > \theta$ are also mutually different and also different from u_i. ■

Remark. The above idea can be used to prove that the cardinality of any θ-symmetric set $S \quad R^n$ containing θ is odd. Indeed, assume $|S|=2k$. Writing S in the lexicographic order, the index of θ cannot be $\leq k$, hence it is $\geq k+1$ that is again a contradiction. □

Using the same method of proof as in Theorem 1 we can prove:

Theorem 2. Let $S \subset R^n$ be any L-measurable set. Then

$$(13) \qquad \tfrac{1}{2}(1+|(S-S) \cap \Lambda|) \geq (\mu(S)/c(S,\Lambda)),$$

consenquently, if $(\mu(S)/c(S,\Lambda))>m$, then $S-S$ contains m pairs of mutually different non-zero lattice points $u_i, -u_i \in \Lambda$, $i=1,\dots,m$, such that the cardinality of $H:=\{u_i, -u_i,\ i=1,\dots,m\}$ is $2m$. □

This theorem contains Thm 1 as a special case. The first generalization of Minkowski's theorem of this type is due to Blichfeldt (see |1|). It is clear that $c(S,\Lambda) \leq d\Lambda$, hence (7) and (13) yield substantial sharpening of Minkowski's and Blichfeldt's result, respectively. Many examples can be found such that $c(\tfrac{1}{2}K,\Lambda) << d\Lambda$ (i.e. $\{\tfrac{1}{2}K,\Lambda\}$ is far from being an $(\tfrac{1}{2}K,\Lambda)$-covering of R^n) and $\mu(K) << 2^n d\Lambda$, but their ratio is great enough to ensure many lattice points in K.

A deatiled discussion and further development of the proof of Minkowski's theorem via (9), (10) and (7) can be found in [2], [3] and [4].

REFERENCES

[1] C.G.LEKKERKERKER, "Geometry of Numbers", Wolters-Noordhoff Groningen, North-Holland, Amsterdam, 1969.

[2] B.UHRIN, A note to Minkowski's convex body theorem, Matematikai Lapok, 28 /4/, /1980/, 323-326, /in Hungarian/

[3] B.UHRIN, Some useful estimations in geometry of numbers, Period. Math. Hungar., 11 /2/, /1980/, 95-103.

[4] B.UHRIN, A generalization of Minkowski's convex body theorem, J. of Number Theory, 13 /2/, /1981/, 192-209.

A Minkowski-féle konvex-test tétel és az R^n-nek egy adott halmazzal vett lefedési mértéke

Uhrin Béla

Összefoglaló

Legyen Λ R^n egy geometriai rács, P az alapcellája és $d\Lambda = \mu(P)$ a determinánsa. Egy tetszőleges S R^n Lebesgne mérhető halmaz esetében az $c(S,\Lambda)=\mu(\bigcup_{u\in\Lambda}((S+u)\ P))$ mértéket az "R^n-nek az S halmas által vett lefedési mértékének" nevezzük. Bebizonyitjuk, hogy az $S-S$ (algebrai) differencia-halmazban levő rácspontok számánál nem $\mu(S)/d\Lambda$ hányados szàmit (ahogyan eddig ismert volt), hanem a $\mu(S)/c(s,\Lambda)$ hányados. Ez élesiti Minkowski ill. Blichfeldt klasszikus eredményét.

ТЕОРЕМА МИНКОВСКОГО О ВЫПУКЛЫХ ТЕЛАХ И МЕРА ПОКРЫТИЯ R^n МНОЖЕСТВОМ

Б. Ухрин

Резюме

В статье доказывается, что число точек решетки $\Lambda \subset R^n$ содержащихся в /алгебраической/ разнице S-S множества $S \subset R^n$ измеряется числом $\mu(S)/c(S,\Lambda)$, а не числом $\mu(S)/d\Lambda$. Здесь $c(S,\Lambda) \leq d\Lambda$ есть мера покрытия R^n множеством S, определенная как $c(S,\Lambda)=\mu(\bigcup_{u\in\Lambda}((S+u)\cap P))$. Этот результат улучшает классические результаты Минковского и Блихфелдта.