4[de]
[th]

SUID-AFRIKAANSE REKENAARSIMPOSIUM
SOUTH AFRICAN COMPUTER SYMPOSIUM

HOLIDAY INN PRETORIA
JULIE 1 – 3 JULY 1987

Proceedings

of the

4th South African Computer Symposium

Holiday Inn, Pretoria

1 – 3 July 1987

edited by

Pieter Kritzinger

Computer Science Department
University of Cape Town

PREFACE

Computer science is an emerging discipline which is having difficulty in being recognised as a worthy member of the sciences. I will paraphrase John Hopcroft, co-winner of the 1986 Turing Award, when, during a recent interview, he said that the primary reason for the lack of recognition, is the age of our researchers. Probably not one of the researchers who presented their work at this symposium is older than 45. I know of no computer scientist in South Africa who is in a position where (s)he can affect funding priorities. As far as I know we have no representation on any of the committees of the Foundation for Research Development and for our Afrikaans speaking fraternity, none who is a member of the *Akademie vir Wetenskap en Kuns*. It will take time and conscious effort to establish our presence. The same is true of course for our universities. Again, with one exception, I know of no dean of a science faculty, vice-principal or principal who is a computer scientist. We consequently spend an enormous amount of time trying to explain the needs of computer science and its difficulties. I believe this symposium is a further step towards accreditation by our peers and superiors from the other sciences.

The total number of papers submitted to the Programme Committee for consideration was 34. Each paper was reviewed by three persons knowledgeable in the field it represents. Of those submitted, 23 were finally selected for inclusion in the symposium. As a result the overall quality of the papers is high and as a computer science community in Africa we can be justly proud of the final programme.

This is the fourth in the series of South African computer symposia. This year the symposium is sponsored by the Computer Society of South Africa (CSSA), the South African Institute for Computer Scientists and the local IFIP Committee. The executive director of the CSSA and his staff deserve warm thanks for handling the organisation as well as they have, while the Organising Committee provided Derrick and I with very valuable advice.

Finally I would like to express my sincere appreciation to the authors, to the members of the Programme Committee and particularly the reviewers. Without the kind cooperation of everyone, this symposium would not have taken place.

Pieter Kritzinger
July 1987.

SYMPOSIUM CHAIRMAN: PS Kritzinger, University of Cape Town.

SYMPOSIUM CO-CHAIRMAN: D Kourie, University of Pretoria.

MEMBERS OF THE PROGRAMME COMMITTEE

Judy Bishop, Witwatersrand University

Chris Bornman, UNISA

Hannes de Beer, Potchefstroom University

Gideon de Kock, Port Elizabeth University

Jaap Kies, Western Cape University

Derrick Kourie, Pretoria University

Pieter Kritzinger, Cape Town University

Tony Krzesinski, Stellenbosch University

Michael Laidlaw, Durban Westville University

Peter Lay, Cape Town University

Ken MacGregor, Cape Town University

Theo McDonald, Orange Free State University

Jan Oosthuizen, University of the North

Dennis Riordan, Rhodes University

Alan Sartori-Angus, Natal University

John Shochot, Witwatersrand University

Theuns Smith, Rand Afrikaans University

Trevor Turton, ISM (Pty) Ltd

Gerrit Wiechers, Infoplan.

LIST OF REVIEWERS

BERMAN Sonia
BISHOP Judy
BORNMAN Chris
CAREY Chris
CHERENACK Paul
DE BEER Hannes
DE VILLIERS Pieter
GORRINGE Pen
KIES Jaap
KOURIE Derrick
KRITZINGER Pieter
KRZESINSKI Tony
LAIDLAW Michael
LAY Peter
MacGREGOR Ken
MATTISON Keith
McDONALD Theo
RENNHACKKAMP Martin
RIORDAN Denis
SATORI-ANGUS Alan
SCHOCHOT John
SMITH Theuns
TURTON Trevor
VAN DEN HEEVER Roelf
VAN ROOYEN Hester
VON SOLMS Basie
VOS Koos

TABLE OF CONTENTS

Computer Languages III

Tutorial

DAY 1

07h30	Registration and Coffee.
08h45	Welcoming address, President of the South African Institute of Computer Scientists, Dr. G. Wiechers.
09h00	Invited Lecture. Professor D. Teichrow, University of Michigan. *Software Engineering, ... What Can We Expect in the Future.*
10h00	COFFEE
	Computer Languages I. Chairman: G. Wiechers.
10h15	S. Berman, University of Cape Town. *SPS-Algol: Semantic Constructs for a Persistent Programming Language.*
10h50	A. Watson, University of the Witwatersrand. *Petri Net Topologies for a Specification Language.*
11h25	A. Mori, University of Cape Town. *Towards a Programming Environment Standard in USP.*
11h50	J. Bishop, University of the Witwatersrand. *ADA for Multiprocessors: Some Problems and Solutions.*
12h30	LUNCH

	Computer Graphics. Chairman: D. Kourie	**Operating Systems.** Chairman: K. MacGregor.
14h00	C. F. Scheepers, CSIR. *Polygon Shading on Vector Type Devices.*	M. Morris, UNISA. *The Development of a Fault Tolerant System for a Real-time Environment.*
14h35	P. Gorringe, CSIR. *Hidden Surface Elimination in Raster Graphics Using Visigrams.*	B. H. Venter, CSIR. *A New General-purpose Operating System.*
15h15	COFFEE	COFFEE
	Database Systems I. Chairman: B. von Solms.	**Computer Languages II.** Chairman: J. Bishop.
15h30	M.E. Orlowska, UNISA. *On Syntax and Semantics Related to Incomplete Information Databases.*	E.M. Ehlers and B. von Solms, Randse Afrikaanse Universiteit. *The Representation of Chemical Structures by Random Context Structure Grammars.*
16h05	M.H. Rennhackkamp, Stellenbosch University. *Modelling Distributed Database Concurrency Control Overheads*	W. van Biljon, CSIR. *A Generalised Expression Structure.*

18h00	Cocktail Party in Cullinan Room A.

DAY 2

08h30	Keynote Address by Profesor Niklaus Wirth, Swiss Federal Institute for Technology, Zurich. *An Extensible System and a Programming Tool for Workstation Computers.*
	Computer Networks and Protocols I. Chairman: P.S. Kritzinger.
09h30	A.E. Krzesinski, University of Stellenbosch. *An Approximate Solution Method for Multiclass Queueing Networks with State Dependent Routing and Window Flow Control.*
10h05	J. Punt, University of Cape Town. *A Protocol Validation System.*
10h30	COFFEE
	Computer Networks and Protocols II. Chairman: R. van der Heever.
11h00	P.S. Kritzinger, University of Cape Town. *Protocol Performance using Image Protocols.*
11h35	Invited Lecture by Professor G. Lasker, University of Windsor, Ontario. *The Relationship of Natural and Artificial Intelligence.*
12h30	LUNCH

	Artificial Intelligence. Chairman: G. Lasker.	**Information Systems.** Chairman: D. Teichrow.
14h00	A. Cooper, CSIR *A Data Structure for Exchanging Geographic Information.*	L. du Plessis and C. Bornman, UNISA. *The ELSIM Language: an FSM-based Language for the ELSIM SEE.*
14h35	A. I. Newcombe, University of Cape Town and R. Rada, National Library of Medicine, Maryland. *Strategies for Automatic Indexing and Thesaurus Building.*	J. Mende, University of the Witwatersrand. *Three Packaging Rules for Information System Design.*
15h15	COFFEE	COFFEE
	Database Systems II. Chairman: C. Bornman.	**Computer Languages III.** Chairman: N. Wirth.
15h30	M.J. Philips, CSIR. *An Approach to Direct End-user Usage of Mutiple Databases.*	W. van Biljon, CSIR. *Experience with a Pattern-matching Code Generator.*
16h05	S. Berman, University of Cape Town. *A Semantic Data Model Approach to Logical Data Independence.*	C. Mueller, University of the Witwatersrand. *Set-oriented Functional Style of Programming.*

16h45	Open Forum with professors G. Lasker, D. Teichrow and N. Wirth. Moderator: Dr. D. Jacobson.
19h30	Symposium Banquet in Cullinan Room. Guest speaker, Dr. D. Jacobson,. Group Executive: Technology, Allied Technologies Limited.

DAY 3

08h00 Registration (Tutorial only).

08h30 Tutorial.
The Tutorial will be given by professor Niklaus Wirth, Division of Computer Science, Swiss Federal Institute of Technology, Zurich.

The use of Modula-2 in Software Engineering.
Topics to be covererd include:

What is Software Engineering?
Data types and structures.
Modularization and information hiding.
Definition and implementation parts.
Separate compilation with type checking.
Facilities to express concurrency.
Pompous programming style.
What could be excluded?

12h15 **Close of Symposium.**

12h30 LUNCH

ON SYNTAX AND SEMANTICS RELATED TO INCOMPLETE INFORMATION DATABASES

M.E. ORLOWSKA
Department of Computer Science
and Information Systems,
University of South Africa

ABSTRACT

We propose a mathematical model of a database with incomplete information, which we call the N-system. Basically speaking, such a system stores information concerning properties of some objects. The information may be incomplete in that it may be known that the object has a property, from the interval $\langle\alpha,\beta\rangle \subset \langle 0,1\rangle$. In the particular case where $\langle\alpha,\beta\rangle=\langle 0,1\rangle$, means that it may be not known whether an object has a property, the N-system is equivalent to Lipski's system [3]. In another particular case, when $\langle\alpha,\beta\rangle=\langle 1,1\rangle$, the N-system presented corresponds to Codd's relational model of databases.

The first level, the level of terms of a query language to communicate with the N-system, is described and its semantics and syntax are defined. A system of axioms, which serves as a basis for equivalent transformations of queries is presented (only for terms). Syntactical and semantic equivalences for terms are defined and completeness properties are shown.

Introduction

The problem of information incompleteness seems to be very complex, and, at the same time, inherent in most human activities and particularly in the domain Data Bases.

There are many sides to the topic, of which only a few will be treated in this paper. First of all, let us notice that there are many kinds of information incompleteness or, at least, situations which superficially look like information incompleteness.

To mention some of them:

1) "We don't know but we shall know in a specified time."
2) "We don't know and we shall never know."
3) "We don't know and it is not known whether or not we shall ever know."
4) "We do not even know whether the question makes any sense."
5) "It is very probable (almost sure) that ....."
6) "With probability p......"
7) "With probability $p \in \langle\alpha,\beta\rangle \subseteq \langle 0,1\rangle$"
8) "To some extent (to some degree) ......"

et cetera.

Roughly speaking, our study concerns cases (1),(2),(3),(6) and (7). We do not draw any distinction between cases (1) (2) and (3) because, for a fact F, we shall consider only three cases:

(i) We know F is true,
(ii) We know F is false
(iii) We don't know whether or not F is true.

Our considerations have a certain common ground with the Dempster-Shafer theory of evidence. See [17]. The Dempster-Shafer theory of evidence has attracted considerable attention as a promising method of dealing with some of the basic problems arising in the combination of evidence, data fusion, and uncertainty in expert systems.

To develop a better understanding of these problems and methods of dealing with the management of uncertainty (partial information), a good background in probability theory, Boolean algebraic theory and logic is required. [6] *W. Marek and Z. Pawlak* contains a description of a mathematical model (introduced previously by *Pawlak* [13], and independently by *E Wong and T.C. Chiang* [16]) of an information storage and retrieval system, abbreviated to ISR system (see also *W. Lipski* and *W. Marek* [5]). An ISR system is a mathematical model of a relational database with complete information.

The theory of ISR systems is based on the formalized language which is intermediate between propositional and predicate calculi. However, the theory corresponds to Codd's theory of relational databases [1].

In [3] and [4] *W. Lipski* has extended the model of an ISR system proposed by *Marek and Pawlak* [6] to cover the case of some incomplete information called the L-system for short. In the theory of L-systems only cases (1),(2) and (3) of incompleteness are considered. Another important study of the problems related to incomplete information has been undertaken by Jaegermann [2]. In [15] *T. Traczyk and W. Marek* and in [14] *T. Traczyk* have shown that Pawlak's model is a particular instance of a much more general case - a stochastic information system (case (6)).

In [8] the present author has presented a common approach to all the above-mentioned systems, and has shown that it is possible to express the ISR system, the L-system and the stochastic informational system as a quadruple $S=\langle X,A,R_I,V\rangle$ where X is a set of objects, A is a nonempty set of descriptors, R_I is an equivalence relation on A and V is a map defined on the product AxX with the values in the set of all subcontinua of the interval $\langle 0,1\rangle$.

A very natural consequence of these considerations was a generalization of the above-mentioned informational systems. The generalized system is called an N-system.

An N-system is a relational model of a database with incomplete information.

The following cases of noncompleteness of information are admissible:

- It is not known whether or not an object $x\in X$ has a property $a\in A$,
- We know that the probability that the object $x\in X$ has a property $a\in A$ is a number from the interval $\langle\alpha,\beta\rangle\subseteq\langle 0,1\rangle$.

The process of changing (increasing) the information plays a vital role in our theory. The "dynamic" approach allows our study to be considered not only from an information retrieval point of view, but also in the broader context of modelling the process of collecting and representing knowledge.

Some important algebraic properties of N-systems representing "dynamic" changes of information have been studied in [8] and also in [9]. In this paper the first level - the level of terms of a query language to N-systems - is presented.

Basic notions

A mathematical model of an N-system is given below. Let $C(\langle 0,1\rangle)$ be the set of all subcontinua (subsets which are closed and connected) of the interval $\langle 0,1\rangle$.

Let $V: Z \longrightarrow C(\langle 0,1\rangle)$ be a function, such that for each $z \in Z$, $V(z)$ is an interval $\langle \alpha,\beta\rangle \subset \langle 0,1\rangle$. We will call the numbers α and β the lower and upper values of the function V, denoted by the symbols $\alpha = V_*(z)$, $\beta = V^*(z)$.

Definition 1 (N-system)

By N-system we mean a quadruple

$S = \langle X, A, R_I, V\rangle$

where

- X – finite, nonempty set of objects
- A – finite set of descriptors
- R_I – an equivalence relation on A; the equivalence classes of R_I are intended to be attributes and are denoted by $\{A_i\}_{i \in I}$

$V: D_V \longrightarrow C(\langle 0,1\rangle)$, where D_V is a subset of the product AxX and the following conditions are satisfied:

1) $\sum_{a \in A_i} V_*(a,x) \leq 1$ for $i \in I$

2) if for some $x \in X$ and $i \in I$;
$\{a \in A_i;\ (a,x) \in D_V\} = A_i$ then
$\sum_{a \in A_i} V^*(a,x) \geq 1$.

<u>Fact:</u> So the notion of the N-system which has been defined is common generalization of the notions of the ISR system, the L-system and the stochastic informational system. See [8], [11].

In the case where the set of attributes contains only one set, i.e. the relation R_I has only one equivalent class, instead of $S=\langle X,A,R_I,V\rangle$ we will write $S=\langle X,A,V\rangle$ as an abbreviated form.

<u>The first level of language to the N-system</u>

The main task of an information system is to answer queries submitted by the user(s). To communicate with the system - more specifically to formulate queries - the user has a query language at his disposal. In the special case when the information is complete, semantics is intuitively evident, and is "the only natural one". This is no longer true when the information is incomplete. The need for a precise formal semantics is essential here. It is also clear that a query can be interpreted in many different ways, of which none is distinguished as "the only natural one". This fact raises the problem of making sure that the user's intention, as expressed in a query, and the system's understanding of that query coincide.

We will describe the semantics of a query language to N-systems-$\mathcal{L}_N$ below. We will concentrate on the first level of the language $\mathcal{L}_N$ - the level of terms. The second level - the level of formulas - will be presented in a separate paper.

Let $S=\langle X,A,V\rangle$ be a single attribute N-system where $A=\{a_1,a_2,\ldots,a_n\}$.
Roughly speaking, the foundamental queries to an N-system are of the form:

> "List all objects $x \in X$ such that the probability that x is described by the property $a \in A$ belongs to the interval $\langle\alpha,\beta\rangle \subset \langle 0,1\rangle$".

It is easy to see that in the case when the N-system is a maximal extension of some N-system [9] then an arbitrary object $x \in X$ is either described by $a \in A$ with probability, from the interval $\langle\alpha,\beta\rangle \subset \langle 0,1\rangle$ or is not described.

Using only the information contained in the N-system we may state that:

(1) The object x has the property a, with the probability from the interval $\langle\alpha,\beta\rangle$:

$$[\ \langle V_*(a,x),V^*(a,x)\rangle \subset \langle\alpha,\beta\rangle\ ]$$

(2) The object x may have the property a with the probability from the interval $\langle\alpha,\beta\rangle$

$$[\ \langle V_*(a,x),V^*(a,x)\rangle \cap \langle\alpha,\beta\rangle \neq \emptyset\ ]$$

(3) The object x has no property a with probability from the interval $\langle\alpha,\beta\rangle$

$$[\ \langle V_*(a,x),V^*(a,x)\rangle \cap \langle\alpha,\beta\rangle = \emptyset\ ]$$

We postulate the following form of a term of the language $\mathcal{L}_N$: A term is the pair (a,P), such that $a \in A$ and $P \subset \nabla$ where ∇ is the characterization of the set $C(\langle 0,1\rangle)$. (For details of the geometric construction and some algebraic properties of the set ∇, see [10])

Definition 2 (Alphabet of the language $\mathcal{L}_N$)

The alphabet of $\mathcal{L}_N$ consists of:

(i) All pairs $(b_1,\dots,b_m, P)$, where $b_i \in \tilde{A}$, $P \subset \nabla^m$, $i = 1,2,\dots,m$, taken as constants of the language $\mathcal{L}_N$, where

$\tilde{A} = \{b = a_{i_1} \vee \dots \vee a_{i_k};\ a_{i_j} \in A,\ 1 \le i_1 \le \dots \le i_k \le n\}$

is the set of algebraic expressions on A. ($\tilde{A}$ is an atomic Boolean algebra, with atoms $a_1,\dots,a_n$ and operations

$(\wedge, \vee, -, \mathbb{0}, \mathbb{1})$

(ii) Constants $\mathbb{0}, \mathbb{1}$,

(iii) Symbols of Boolean operations $+, \cdot, \sim, \to$,

Note that our language does not contain variables.

Definition 3 (Terms of the language $\mathcal{L}_N$)

The set $\mathcal{T}'$ of terms $\mathcal{L}_N$ is the least set $\mathcal{T}$ satisfying the following conditions:

(i) $\mathbb{0}, \mathbb{1} \in \mathcal{T}$

(ii) $(b_1,\dots,b_m,P) \in \mathcal{T}$ where $P \subset \nabla^m$, $b_i \in \tilde{A}$ $i = 1,\dots,m$.

(iii) if $t,s \in \mathcal{T}$ then $t+s \in \mathcal{T}$, $t \cdot s \in \mathcal{T}$, $\sim t \in \mathcal{T}$, $t \to s \in \mathcal{T}$

It is obvious that each term of the form (a,P) is a particular case of the general form of the term $(b_1,\dots,b_m,P)$ in Definition 3.

Parentheses are used, if necessary, in the obvious way.

As will become apparent later on, the order of a sum or product is immaterial, so we shall abbreviate finite sums and products as $\sum_{i\in J} t_i$ and $\prod_{i\in J} t_i$, respectively.

Intuitively, terms are the names of certain features of objects more "complex" than those expressed by descriptors and intervals.

Semantics - interpretation of terms

We shall now define the semantics of our query language $\mathcal{L}_N$. For any query Q we shall define its value denoted by $\|Q\|$, which intentionally is the response of the N-system to the query Q. It is intuitively clear what the value of query is if we think of $\mathbb{0},\mathbb{1},-,+,\cdot,\rightarrow$ as corresponding to $\emptyset$,X, and the set theoretical operations of complementation, union, intersection and the operation "$(X\setminus A)\cup B$" respectively.

More formally we have the following definition.

<u>Definition 4 (Value of term)</u>

The value of the term $t\in\mathcal{T}$ in an N-system S=<X,A,V> denoted by $\|t\|_S$ is defined inductively as the following subset of X;

1) $\|\mathbb{0}\|_S=\emptyset$, $\|\mathbb{1}\|_S=X$,

2) $\|(b_1,\ldots,b_m,P)\|_S=\{x\in X;\ (\tilde{V}(b_1,x),\ldots,\ \tilde{V}(b_m,x))\in P\}$ where $\tilde{V}$ is an extension of the function V on the set of algebraic expressions $\tilde{A}$; $\tilde{V}:\tilde{A}\times X\longrightarrow C(\langle 0,1\rangle)$ defined as follows:

If $b=a_{i_1}\vee\ldots\vee a_{i_k}$ then $V(b,x)=\langle V_*(b,x),\ V^*(b,x)\rangle$

where

$$\tilde{V}_*(b,x)=\max\Big(\sum_{j=i}^{k} V_*(a_{i_j},x),\ 1-\sum_{i\notin\{i_1,\ldots,i_k\}} V^*(a_i,x)\Big)$$

$$\tilde{V}^*(b,x)=\min\Big(\sum_{j=i}^{k} V^*(a_{i_j},x),1-\sum_{i\notin\{i_1,\ldots,i_k\}} V_*(a_i,x)\Big)$$

3) $\|t+s\|_S = \|t\|_S \cup \|s\|_S$

4) $\|t\cdot s\|_S = \|t\|_s \cap \|s\|_S$

5) $\|t\to s\|_S = (X\setminus\|t\|_S) \cup \|s\|_S$

6) $\|{\sim}t\|_S = X \setminus \|t\|_S$

In practice it may also be useful to consider atomic formulas of the type $t\leq s$. However, since $t\leq s$ can always be expressed by $t\cdot{\sim}s=0$, we will not explicitly introduce $\leq$ into the language.

We shall prove the following theorem:

Theorem 5

For each N-system $S=\langle X,A,V\rangle$ we have:

(1) $\|(b,\emptyset\|_S = \emptyset$

(2) $\|(b,\nabla)\|_S = X,$

(3) $\|(a_1\vee a_2\vee\ldots\vee a_m,\{1\})\|_S = X$

(4) $\|(b,P_1)+(b,P_2)\|_S = \|(b,P_1 \cup P_2)\|_S$

(5) $\|(b,P_1)\cdot(b,P_2)\|_S = \|(b,P_1 \cap P_2\|_S$

(6) $\|(b_1,P_1)\cdot(b_2,P_2)\|_S = \|(b_1,b_2,P_1\times P_2)\|_S$

(7) $\|({\sim}b,P)\|_S = \|(b, \nabla \setminus P\|_S$

where $b=(a_{i_1} \vee \ldots \vee a_{i_m}) \in \tilde{A}$, $P \subset \nabla^m$

Proof:

(1) and (2) are obvious from the definition of the value of the term,

(3); – it follows from the simple fact that for each $x \in X$ we have

$$\tilde{V}(a_1 \vee a_2 \vee \ldots \vee a_m, x)=\{1\}$$

(4); $\|(b,P_1)+(b,P_2)\|_S=\|(b,P_1)\|_S \cup \|(b,P_2)\|_S=$

$$= \{x \in X;\ \tilde{V}(b,x) \in P_1\} \cup \{x \in X;\ \tilde{V}(b,x) \in P_2\}$$
$$= \{x \in X;\ \tilde{V}(b,x) \in P_1 \vee \tilde{V}(b,x) \in P_2\} =$$
$$= \{x \in X;\ \tilde{V}(b,x) \in P_1 \cup P_2\}=\|(b,P_1 \cup P_2\|_S$$

The proof of (5), (6) and (7) is similar to the proof of (4). ■

Definition 6.

Two terms t and s are semantically equivalent, denoted $t \approx s$, if for every N-system S

$$\|t\|_S = \|s\|_S.$$

A query entering an N-system can be replaced by any other semantically equivalent query. In practice, it is reasonable to transform the query into some semantically equivalent query which minimizes the cost of retrieval. We want the transformation to be carried out by purely syntactical means without accessing the file which stores the information about the objects. To this end we can use the set of axioms (rules of transformation) during the transformation process.

Prior to a presentation of axioms for terms in $\mathcal{L}_N$, we introduce and consider some special auxiliary sets.

Let $p=(\alpha,\beta)\in\nabla$ and let its corresponding interval in the characterization of the set $C(\langle 0,1\rangle)$ (see [10]) be

$\bar{p}=\langle\alpha,\beta\rangle \subset \langle 0,1\rangle$.

Let

$$f_{i_1,\ldots,i_k} : R^n \longrightarrow R$$

be a function defined as follows:

$$f_{i_1,\ldots,i_k}(z)=\sum_{j=1}^{k} z_{i_j}.$$

where $z=(z_1,z_2,\ldots,z_n)$.

With $p\in\nabla$ we associate a set

$$P_{i_1,\ldots,i_k}=\{(\alpha,\beta)\in H' \;;\; f_{i_1,\ldots,i_k}(\langle\alpha,\beta\rangle \cap H)=\bar{p}\;\}$$

where $H=\{(z_1,z_2,\ldots,z_n)\in I^n \;;\sum_{i=1}^{n} z_i=1\}$,

$$H'=\{(\alpha,\beta)\in H'' \;;\; \forall\; i=1,\ldots,n \quad \alpha_i+\sum_{\substack{j=1\\ j\neq i}}^{n}\beta_j \geq 1 \text{ and } \beta_i+\sum_{\substack{j=1\\ j\neq i}}^{n}\alpha_j \leq 1\}$$

$$H''=\{(\alpha,\beta)\in R^{2n};\; \alpha,\beta\in I^n,\; \alpha\leq\beta \quad \sum_{i=1}^{n}\alpha_i\leq 1\leq\sum_{i=1}^{n}\beta_i\},$$

$I=\langle 0,1\rangle$.

For a more intuitive interpretation of the sets H,H',H'', see [12].

The set $P_{i_1,...,i_k}$ can be understood as a set of all possible proper representations (see [12]) of the object $x \in X$ such that

$\tilde{V}(b,x)=\bar{p}$ where $b=a_{i_1} \vee ... \vee a_{i_k}$.

Let $P \subset \nabla^m$, where m is a natural number, then every point of the set P is an ordered sequence $(p^1,p^2,...,p^m)$ where $p^i \in \nabla$, $i=1,2,...,m$.

Now we shall define the following three auxiliary sets:

(6) $P^1_{i_1,...,i_k} = \{(h', p^2,...,p^m) \in H' \times \nabla^{m-1};$
$h' \in (p^1)_{i_1,...,i_k}$, $(p^1,p^2,...,p^m) \in P\}$

(7) $P^{1,1} = \{(p^1,p^2,...,p^{m-1}) \in \nabla^{m-1};\ (p^1,p^1,p^2,...,p^{m-1}) \in P\}$

(8) $P^{j,k} = \{p^1,p^2,...,p^m) \in \nabla^m;$
$(p^1,...,p^{k-1}, p^j, p^{k+1},..., p^{j-1}, p^k, p^{j+1},...,p^m) \in P\}$

In the figure below we demonstrate a simple example of two intervals $\langle\alpha,\beta\rangle$ and $\langle\alpha',\beta'\rangle$ such that $(\alpha,\beta),(\alpha',\beta') \in P_{1,3}$.

The intersections of these two intervals with H are shown. The sums of the first and the third co-ordinates of points belonging to these intersections constitute the whole interval $\bar{p}$.

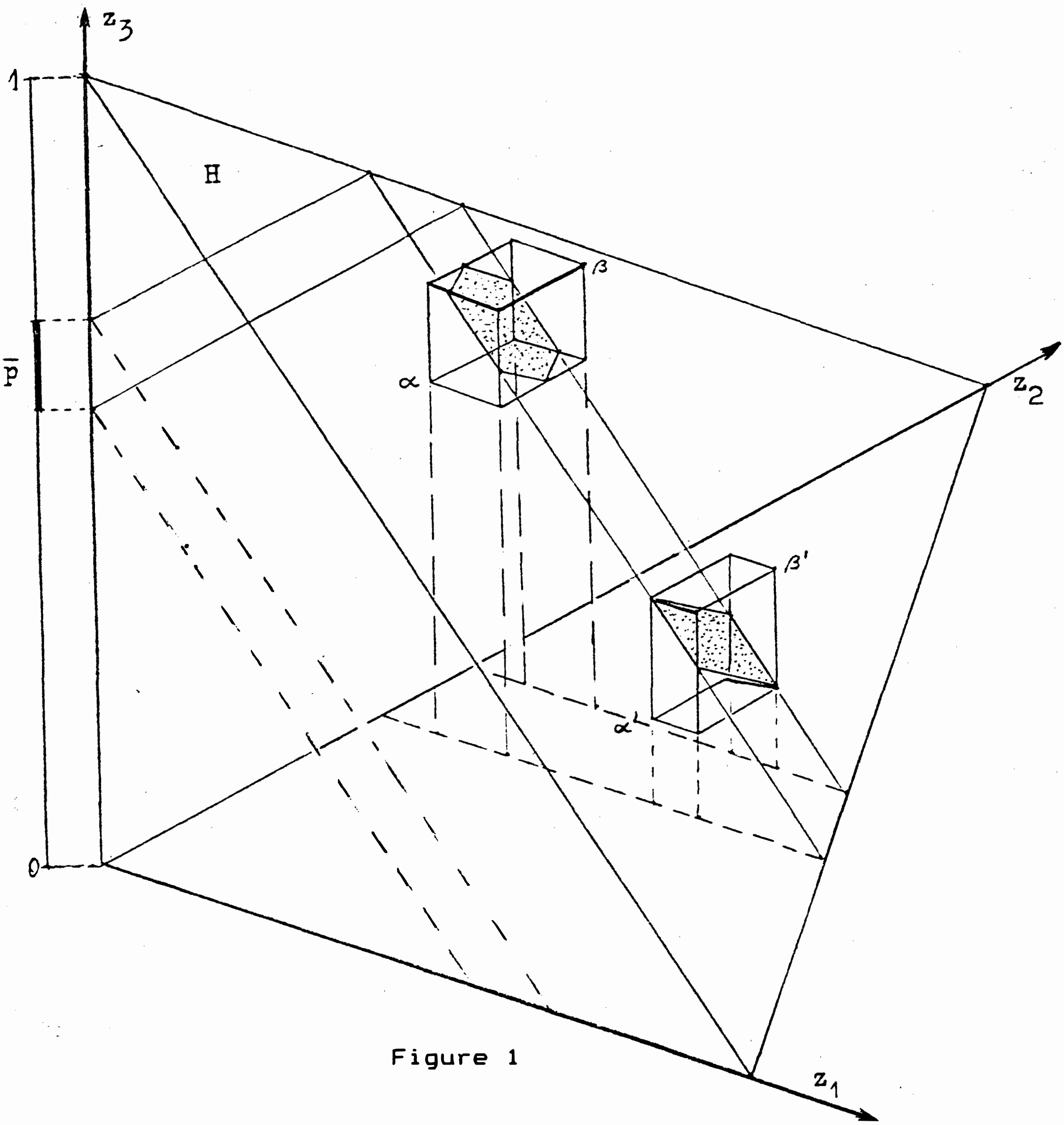

Figure 1

<u>Axioms for queries to N-system</u>

<u>Definition 7</u> (Axioms for terms)

The set 𝔹 of axioms consists of:

B1: Substitutions of terms into the axioms of Boolean algebra.

B2: $(b_1,\ldots,b_m, P_1) + (b_1,\ldots,b_m,P_2)=$
$=(b_1,\ldots,b_m,P_1 \cup P_2)$

B3: $(b_1,\ldots, b_m,P_1) \circ (b_1,\ldots,b_m,P_2)= (b_1,\ldots,b_m,P_1 \cap P_2)$

B4: $\sim(b_1,\ldots,b_m,P)=(b_1,\ldots,b_m,\nabla^{m}\setminus P)$

B5: $(b_1,\ldots,b_m, \emptyset)=\mathbb{0}$, $(a_1 \vee a_2 \vee\ldots\vee a_n,\{1\})=\mathbb{1}$

B6: $(b_1,b_2,\ldots,b_m,P)=(a_1,\ldots,a_n,b_2,\ldots,b_m, P^1_{i_1,\ldots,i_k})$

where $b_1=a_{i_1} \vee a_{i_2} \vee\ldots\vee a_{i_k}$ and $P^1_{i_1,\ldots,i_k}$ is defined by (6)

B7: $(b_1,b_1,b_2,\ldots,b_{m-1},P)=(b_1,b_2,\ldots,b_{m-1},P^{1,1})$

where $P^{1,1}$ is defined by (7)

B8: $(b_1,\ldots,b_m,P)=$

$=(b_1,\ldots,b_{j-1},b_k,b_{j+1},\ldots,b_{k-1},b_j,b_{k+1},\ldots,b_m,P^{k,j})$

where $P^{k,j}$ is defined by (8), $1\leq j\leq k\leq m$.

Definition 8 (Syntactical equivalence of terms)

Two terms t and s are syntactically equivalent, denoted t=s, if and only if there exists a term $u\in\mathcal{T}$ such that the terms t and s can be transformed into the term u using the set of axioms $\mathbb{B}$ and the inference rule *Modus Ponens*.

Theorem 9

Syntactical equivalence of two terms $t,s\in\mathcal{T}$ implies their semantic equivalence;

$$(t=s) \Longrightarrow \|t\|_S = \|s\|_S \text{ for each N-system S.}$$

Proof: We should show that the values of terms, from two sides of equalities in the set of axioms $\mathbb{B}$, are equal in any fixed N-system S.

B1 - B5; they are a direct consequence of Theorem 5.

B6; let $x\in X$

$$x\in\|(b_1,\ldots,b_m,P)\|_S \Longleftrightarrow \forall(p^1,\ldots,p^m\in p)$$

$$(\tilde{V}(b_1,x),\ldots,\tilde{V}(b_m,x))=(p^1,\ldots,p^m) \underset{*}{\Longleftrightarrow}$$

$$(V(a_1,x),\ldots,V(a_n,x))\epsilon p^1_{i_1,\ldots,i_k},\ \tilde{V}(b_2,x)=p^2,\ldots,$$

$$\tilde{V}(b_m,x)=p^m \Longleftrightarrow V(a_1,x),\ldots,V(a_n,x),$$

$$\tilde{V}(b_2,x),\ldots,\tilde{V}(b_m,x)\epsilon P^1_{i_1,\ldots,i_k} \Longleftrightarrow$$

$$\Longleftrightarrow x\epsilon\|(a_1,\ldots,a_n,b_2,\ldots,b_m,P^1_{i_1,\ldots,i_k})\|_S.$$

Let us explain equivalence $\underset{*}{\Longleftrightarrow}$. Let $r_S(x)=\langle\alpha,\beta\rangle$ where

$$r_S(x)=\{(z_1,z_2,\ldots,z_n)\epsilon R^n\ ;\ V_*(a_i,x)\le z_i\le V^*(a_i,x)$$
$$i=1,2,\ldots,n\}.$$

Then

$$\tilde{V}(b_1,x) = p^1 \Longleftrightarrow f_{i_1,\ldots,i_k}(\langle\alpha,\beta\rangle \cap H)= \bar{p}^1 \Longleftrightarrow$$

$$\Longleftrightarrow (\alpha,\beta)\epsilon p^1_{i_1,\ldots,i_k} \Longleftrightarrow (V(a_1,x),\ldots,V(a_n,x))=$$

$$(\alpha,\beta)\ \epsilon\ p^1_{i_1,\ldots,i_k}.$$

B7; $X\epsilon\|(b_1,b_1,b_2,\ldots,b_m,P)\|_S \Longleftrightarrow$

$$\Longleftrightarrow (\tilde{V}(b_1,x),\tilde{V}(b_1,x),\tilde{V}(b_2,x),\ldots,\tilde{V}(b_m,x))\epsilon P \Longleftrightarrow$$

$$\Longleftrightarrow \tilde{V}(b_1,x),\tilde{V}(b_2,x),\ldots,\tilde{V}(b_m,x)\epsilon P^{1,1} \Longleftrightarrow$$

$$\Longleftrightarrow x\epsilon\|(b_1,b_2,\ldots,b_m,P^{1,1})\|_S.$$

B8; this is a direct consequence of the definition of the set $P^{k,j}$.

■

Normal form theorems play an important role in the theory of N-systems. On the one hand they are the main tool for proving completeness results and, on the other hand, they have far-reaching implementational consequences.

In this paper we shall present only one normal form, but a far more intensive discussion of the normalization of N-systems will be presented in a separate paper.

Definition 10 (Canonical form)

A term $t \in \mathcal{T}$ is in canonical form if $t=(a_1,a_2,\dots,a_n,P)$ where $P \subset H'$.

Theorem 11

For each term t there is a term s in canonical form such that $\vdash t=s$.

Proof: It is rather obvious that the terms $\mathbb{0}$ and $\mathbb{1}$ can be represented in the form:

$$\mathbb{1} = (a_1,\dots,a_n,H')$$
$$\mathbb{0} = (a_1,\dots,a_n,\emptyset)$$

Let $t=(b_1,\dots,b_m,P)$ be an arbitrary term. Using the axiom B6 in definition 9 we have

$$t=(b_1,\dots,b_m,P)=(a_1,\dots,a_n,b_2,\dots,b_m,P^1_{i_1,\dots,i_k}).$$

Now applying n-times the axiom B8 we may write

$$t=(b_2,a_1,\dots,a_n,b_3,\dots,b_m, ((((P^1_{i_1,\dots,i_k})^{n+1,n})^{n,n-1})\dots)^{2,1})$$

Repeated application of B6 and B8 gives us the following form

$$t=(a_1,\dots,a_n,a_1,\dots,a_n,\dots,a_1,\dots,a_n,Q')$$

where $Q' \subset (H')^m$

Now, using B7 and B8 repeatedly, we have:

$t=(a_1,\ldots,a_n,Q)$, $Q\subset H'$

but this is the cononical form of the term $(b_1,\ldots,b_m,P)$.

Therefore our proof is complete.

■

To complete our consideration of the completeness property, we have to prove that semantic equivalence for terms implies their syntactical equivalence.

Theorem 12

Let t and s be two terms. If for every N-system S

$\|t\|_S = \|s\|_S$ then $t = s$.

Proof: Firstly we shall show that for each term $t\in\mathcal{T}$, $t\neq 0$, there exists an N-system S such that

$\|t\|_S \neq \emptyset$

Using Theorem 11 we may transform t into canonical form

$(a_1,a_2,\ldots,a_n,P)$

where $P\subset H'$ and $P\neq\emptyset$. There is then some $\langle\alpha,\beta\rangle$ such that $\langle\alpha,\beta\rangle\in P$.

Let x be an object in an N-system $S=\langle\{x\},A,V\rangle$ such that

$V(a_i,x)=\langle\alpha_i,\beta_i\rangle \quad i=1,\ldots,n,$

so far in this N-system S

$x\in\|t\|_S$ then $\|t\|_S\neq\emptyset$.

Let us suppose that $t\neq s$, $t,s\in\mathcal{T}$.

We have

$(t\cdot{\sim}s)+({\sim}t\cdot s) \neq 0$

From the first part of this proof it follows that for some N-system S the value of the term

$$\|(t\cdot{\sim}s)+({\sim}t\cdot s)\|_S \neq \emptyset$$

so

$$\|t\|_S \cap (X \setminus \|s\|_S) \cup (X \setminus \|t\|_S) \cap \|s\|_S \neq \emptyset$$

but this means that

$$\|t\|_S \neq \|s\|_S$$

and this is the end of the proof.

The conjunction of Theorems 11 and 12 is known as the Completeness Theorem.

The completeness property for terms for a multi-attribute N-system $S=\langle X,A,R_I,V\rangle$ can easily be obtained by applying an algebraic operation called the direct product (see [9]), to single-attribute N-systems S_i, i=1,2,...,I. This procedure is rather technical and is not presented in this paper.

References:

[1] CODD E.F.: *A relational model of data for large shared data banks.* CACM 13:6, June 1970, 377-387.

[2] JAEGERMANN M.: *Information storage and retrieval systems - mathematical foundations.* Systems with incomplete information. CC PAS Report 214, 1975.

[3] LIPSKI W.: *On databases with incomplete information.* JACM 28:1, January 1981.

[4] LIPSKI W.: *On semantic issues connected with incomplete information databases.* ACM TODS 4:3, September 1979.

[5] LIPSKI W. and MAREK W.: *Information systems: On queries involving cardinalities.* Information Systems 4:3, Elmsford, NY: Pergamon Press, Inc.

[6] MAREK W. and PAWLAK Z.: *Information storage and retrieval systems: Mathematical foundations.* Theoretical Computer Science. 1(4), April 1979.

[7] MAYER D.: *The theory of relational databases.* Computer Science Press, 1983.

[8] ORLOWSKA M.E.: *Algebraic and topological properties of databases with incomplete information.* D.Sc. thesis. Technical University of Warsaw, Poland 1980.

[9] ORLOWSKA M.E.: *Dynamic properties of N-incomplete databases.* Demonstratio Mathematica (to appear). (Preliminary version available as Unisa Research Report RR/87-03.

[10] ORLOWSKA M.E.: *Geometric characterization of the set of subintervals of the interval <0,1>.* Unisa Research Report RR/87-01.

[11] ORLOWSKA M.E.: *Common approach to some informational systems.* Unisa Research Report RR/87-02.

[12] ORLOWSKA M.E.: *Geometric representation of objects in relational databases.* Paper presented at SACLA conference, 1986.

[13] PAWLAK Z.: *Mathematical foundations of information retrieval*, CC.PAS Reports No. 101, 1973.

[14] TRACZYK T.: *Common extension of Boolean information systems.* Fundamenta Informaticae. II, 1978.

[15] TRACZYK. T. and MAREK W.: *Stochastic information systems,* CC PAS No 247, 1976.

[16] WONG E. and CHIANG T.C.: *Canonical structure in attribute based file organization,* C ACM 14, 1971.

[17] ZADEH L.A.: *A simple view of the Dempster-Shafer theory of evidence and its implementation for the rule of combination.* AI Magazine, Summer 1986.