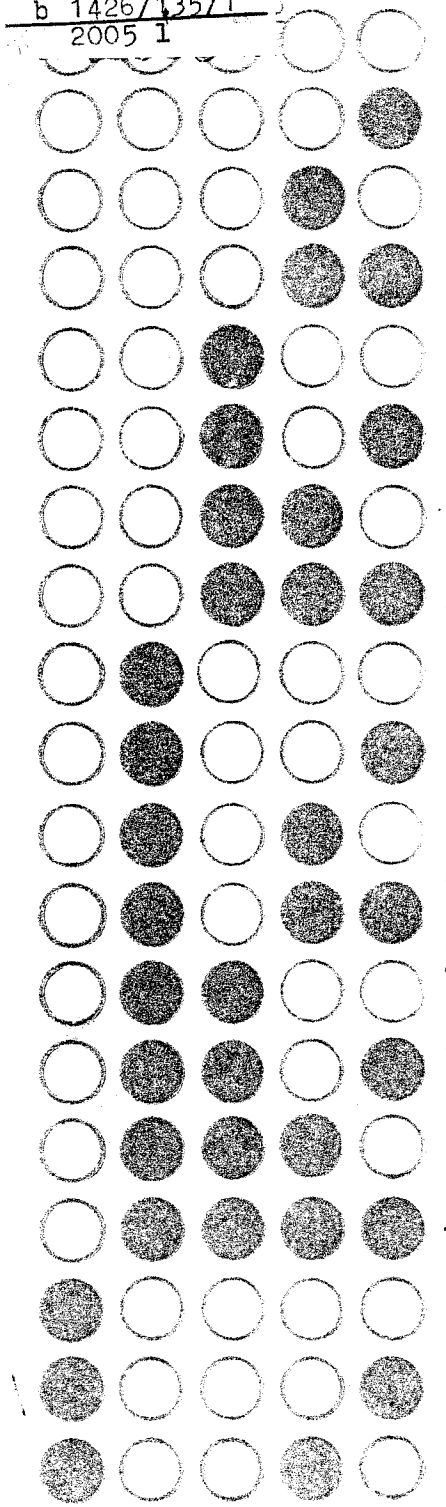
PRACE CO PAN • CC PAS REPORTS



Wiktor Marek, Zdzisław Pawlak

Mathematical foundations of information storage and retrieval

Part 1

135
1973
WARSZAWA

CENTRUM OBLICZENIOWE POLSKIEJ AKADEMII NAUK
COMPUTATION CENTRE POLISH ACADEMY OF SCIENCES
WARSAW, PKiN, P. O. Box 22, POLAND

Wiktor Marek, Zdzisław Pawlak

MATHEMATICAL FOUNDATIONS OF INFORMATION STORAGE AND RETRIEVAL

Part 1

135

Warszawa 1973

Mailing address: Dr. Wiktor Marek
Institute of Mathematics PAS
ul. Śniadeckich 8
00-950 Warszawa
P.O. Box 187

Prof. Dr. Zdzisław Pawlak
Computation Centre PAS
00-901 Warszawa PKiN
P.O. Box 22

Nakład 450 egz. Ark. wyd. 0,40; ark. druk. 0,675
Papier offset. kl. III, 70 g, 70 x 100. Oddano do
druku w październiku 1973 r. W. D. N. Zam. nr 755
R-30

Abstract • Содержание • Streszczenie

This note extends the results of [1] thus giving more adequate description of information storage and retrieval systems.

Математическое описание процесса поиска
и хранение информации. Первая часть

Расширяя результаты [1] приводим более подлежащее математическое описание процесса поиска информации.

Matematyczne podstawy gromadzenia
i wyszukiwania informacji. Część 1

Rozszerzając wyniki [1] dajemy bardziej adekwatny opis matematyczny procesu wyszukiwania informacji.

§ 0. INTRODUCTION

The purpose of this paper is an extension of the results of [1]. Probably the approach we present here is a better approximation of the reality than that one in the above paper. It should be mentioned that even though the present paper extends [1] properly it is possible to interprete our system within the frame work of [1].

We assume knowledge of standard settheoretical and logical notation like $\mathcal{P}(X)$ (powerset of X), $\mathfrak{f}:X \to Y$ etc.

§ 1. SYNTAX

Definition 1.1. Let A,I be given two nonempty, disjoint sets, let $\{A_i\}_{i\in I}$ be some fixed partition of A (i.e. (i)(i') $(i \neq i' \Rightarrow A_i \cap A_{i'} = \emptyset)$, $\bigcup_{i\in I} A_i = A$).

For given set A we define the language L_A as follows: The alphabet of L_A contains:

1° Constants c_a (for $a\in A$)

2° Symbols T, F, V, $\wedge$

3° Auxiliary symbols $\neg$, v , &, $\rightarrow$, $\sim$, + , · , $\to$

4° Symbol =.

Definition 1.2. The set $\mathfrak{T}$ of terms is the least set satisfying 1°&2°;

1° $T\in\mathfrak{T}$, $F\in\mathfrak{T}$, $c_a\in\mathfrak{T}$ $(a\in A)$

2° If t_1, $t_2\in\mathfrak{T}$ then $\sim t_1$, t_1+t_2, $t_1\cdot t_2$, $t_1 \to t_2 \in\mathfrak{T}$

In the sequel t,s (possibly with indices) range over terms.

Definition 1.3. The set F of formulas is the least set satisfying 1°&2°

1° If t_1, t_2 are terms then $\ulcorner t_1 = t_2 \urcorner \in F$

2° If $\varphi_1, \varphi_1 \in F$ then $\neg\varphi_1$ $\varphi_1 \vee \varphi_2$, $\varphi_1 \& \varphi_2$, $\varphi_1 \Rightarrow \varphi_2 \in F$.

We assume as axioms:

1° Substitutions of proposition calcules axioms (cf[2])

- for formulas

2° Substitutions of the axioms of Boolean Algebra

- for terms

3° If $i \in I$, $a \in A_i$, then

$$c_a = \neg\Big(\sum_{\substack{b \in A_i \\ b \neq a}} c_b\Big)$$

Where $\sum$ is an abbrenation for sum of bigger amount of terms. (In case when each A_i, $i \in I$ is finite. If we admit A_i infinite, we need some modifications in syntax). A is called basic dictionary and I - family of features.

§ 2. SEMANTICS, CONNECTIONS WITH THE SYNTAX

Definition 2.1. A system of information storage over basic dictionary A and family of features I is a quadruple $S = \langle X, A, I, U \rangle$ where $U: A \longrightarrow P(X)$ satisfies conditions

1° If $i \in I$, $a, b \in A_i$, $a \neq b$ then $U(a) \cap U(b) = \emptyset$

2° If $i \in I$ then

$$\bigcup_{a \in A_i} U(a) = X$$

Definition 2.2. Valuation of terms.

Given system $S = \langle X, A, I, U \rangle$ we define inductively $\|t\|_S$, $\|\varphi\|_S$ as follows:

(a) $\|c_a\|_S = U(a)$

(b) $\|\sim t\|_S = X - \|t\|_S$

(c) $\|t_1 \cdot t_2\|_S = \|t_1\|_S \cap \|t_2\|_S$

(d) $\|t_1 + t_2\|_S = \|t_1\|_S \cup \|t_2\|_S$

(e) $\|F\|_S = \emptyset$

(f) $\|T\|_S = X$

(g) $\|t_1 \longrightarrow t_2\|_S = (X - \|t_1\|_S) \cup (\|t_2\|_S)$

Now assume $\|t\|_s$ is defined for all $t \in \mathcal{T}$

$$\|t_1 = t_2\|_S = \begin{cases} V & \text{if } \|t_1\|_S = \|t_2\|_S \\ \Lambda & \text{if } \|t_1\|_S \neq \|t_2\|_S \end{cases}$$

$$\|\neg\varphi\|_S = \begin{cases} \Lambda & \text{if } \|\varphi\|_S = V \\ V & \text{if } \|\varphi\|_S = \Lambda \end{cases}$$

For other connectives we extend our definition in natural way.

Theorem 2.1. If φ is an axiom then $\|\varphi\|_s = V$

Proof: Inductively through definition of axioms.

Definition 2.3. Let $S = \langle X, A, I, U \rangle$ be a system $x \in X$.

(a) An information on x in S is a function

$f: I \longrightarrow A$ such that $f(i) \in A_i$ and

$(i)_I$ $(x \in U(f(i)))$

(b) A description of x in S is a term

$$\bigsqcap_{i \in I} c_{f(i)}$$

Clearly an information on x determines a description of x (up to possible order of I).

Definition 2.4. A system S is selective iff for all $x \in X$, if t is a description of x in S then $\|t\|_S = \{x\}$

Thus selective system is the one in which any two elements are distinguishable.

§ 3. COMPLETENESS PROPERTY OF INFORMATIONAL SYSTEMS

Definition 3.1. (a) We define $c_a^0 = c_a$, $c_a^1 = \sim c_a$

(b) A term t is called primitive if $t = c_{a_0}^{\varepsilon_0} \cdot \ldots \cdot c_{a_k}^{\varepsilon_k}$ where each ε_j is 0 or 1.

(c) A term t is in normal additive form if $t = \sum t_j$ where each t_j is primitive term.

(d) A term t is in positive form if $\sim$, $\rightarrow$ does not occur in t.

The axioms accepted (§ 1) allow us to prove formulas (in the theory of information systems). We use $\vdash$ to denote existence of proof of the formula.

Theorem 3.1. (a) If t is a term then there is a term t_1 in normal additive form such that $\vdash t = t_1$

(b) If t is a term then there is a term t_2 in positive normal additive form such that $\vdash t = t_2$.

Proof: (a) A proof of this sort may be found in [2]. (b) By (a) we may assume that t is in normal form. Using axiom 1.3. (3°) and law $x = y \rightarrow \sim x = \sim y$ we get $\sim c_a = \sum_{\substack{b\in A_i \\ b\neq a}} c_b$. Now in any place where there is a negation we substitute appropriate sum and use distribution laws.

Definition 3.2. (a) A primitive term is called complete iff for every $i\in I$ there is exactly one $a\in A_i$ such that c_a occurs in t.

(b) A term t is in complete positive normal additive form iff $t = \sum_k t_k$ and each t_k is complete positive primitive term.

Theorem 3.2. If I is finite then for each term t there is exactly one term t_3 (being in complete positive normal additive form) such that $\vdash t = t_3$

Proof: It is clear that it is enough to prove that each primitive positive term is equivalent to a term in c.p.n.a. form.

Proceeding inductively we assume that for given $i\in I$ no c_a (with a A_c) occur in t. Since $\sim c_a = \sum_{\substack{b\in A_c \\ b\neq a}}$ thus $\sum_{b\in A_i} c_b = T$ and so, since $t \wedge T = t$ we get $t \wedge \sum_{b\in A_i} c_b = t$. Using distributive

laws we diminish the number of i's which are not represented in t. Uniquenes (up to the order of I) is obvious.

Using usual reasoning we can get analogous results on dual - multiplicative form. Since they are similar we will not purse the matter here. As a consequence of the theorem 2 we will get some completeness results.

Definition 3.3. We introduce relations $\leqslant$, $\approx$ on $\mathcal{T}$ as follows

1° $t_1 \leqslant t_2 \iff$ There is t such that $\vdash t + t_1 = t_2$

2° $t_1 \approx t_2 \iff \vdash t_1 = t_2$

This is nothing else but Lindenbaum algebra on $\mathcal{T}$

Lemma 3.3. (a) $\leqslant$ is a partical ordering in $\mathcal{T}$

(b) $\approx$ is an equivalence relation in $\mathcal{T}$

(c) $\leqslant$ generates $\approx$ i:e $t_1 \leqslant t_2 \,\&\, t_2 \leqslant t_1 \Rightarrow t_1 \approx t_2$

Proof: (a) and (b) are obvious.

(c) There are terms t and s such that $\vdash t + t_1 = t_2$ and $\vdash s + t_2 = t_1$ thus $\vdash t + s + t_1 = t_1$ thus $\vdash t + t + s + t_1 = t + t_1$ but $\vdash t + t = t$ and so $\vdash t + s + t_1 = t + t_1$

i.e $\vdash t + s + t_1 = t_2$ i.e $\vdash t_1 = t_2$

Clearly if $t_1 \leqslant t_2$ then for all systems S $\|t_1\|_S \subseteq \|t_2\|_S$. However converse property also is true.

Definition 3.4. (a) term t is semantically less then term s if for all information systems S

$$\|t\|_S \subseteq \|s\|_S$$

(b) Term t is semantically equal to the term s iff for all information systems S : $\|t\|_S = \|s\|_S$.

Theorem 3.4. (Completeness property for terms)

(a) The term t is semantically less then the term s iff $t \leqslant s$

(b) The term t is semantically equal to the term s iff $t \approx s$.

Proof: (b) $\Rightarrow$ follows from (a) and 3.3(c).

(b) $\Leftarrow$ follows from adequacy of our axioms.

(a) $\Leftarrow$ was already remarked after the proof of 3.3.

(a) $\Rightarrow$ We may assume that both t and s are in complete normal positive additive form. If all primitive components of t occur as primitive components of s then clearly $t \leqslant s$. Assume now that there is component of t which does not occur in s. (This is nothing else than assuming not $t \leqslant s$) Then, since values of all primitive positive complete terms are identical or disjoint we easily construct a system S in which $\|t\|_S \not\subseteq \|s\|_S$.

Note that the construction here reminds the components as presented in [3].

Using theorem 3.4. we are able to prove:

Theorem 3.5. (Completeness property for formulas) $\vdash \varphi$ iff for all information systems S, $\|\varphi\|_S = V$.

The proof imitates usual Henkin technique for completeness proof and we do not give it here. Finally let us see how our system is connected with that of [1].

Let X be a set of elementary descriptors (as in [1]) consider an additional set $\bar{X}$, disjoint with X and $\varphi : X \underset{\text{onto}}{\rightarrow} \bar{X}$. We assume, for $x \in X \sim x = \varphi(x)$. In this way we get I = X, each $A_i = \{x, \varphi(x)\}$. Elementary descriptors are primitive terms in our language. The essential result of [1], theorem 5 is thus obtained as theorem 3.1 within our framework.

In the further work we will present operations on information systems and other results concerning selective systems.

Mathematical Institute of Polish Academy of Sciences
Computing Center of Polish Academy of Sciences

Received September 20, 1973

REFERENCES

[1] Z. Pawlak; Math. Foundations of information retrieval CC PAS Reports 101

[2] R. Lyndon: Notes on Logic, Princeton, 1966

[3] K. Kuratowski, A. Mostowski: Set theory, Warszawa, 1967