EMIL L. POST AND THE PROBLEM OF MECHANICAL PROVABILITY

A Survey of Post's Contributions in the Centenary of His Birth

THE CHAIR OF LOGIC, INFORMATICS AND PHILOSOPHY OF SCIENCE
UNIVERSITY OF BIAŁYSTOK
Białystok 1998

EMIL L. POST AND THE PROBLEM OF MECHANICAL PROVABILITY

Series: STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2(15)

EMIL L. POST AND THE PROBLEM OF MECHANICAL PROVABILITY

A Survey of Post's Contributions in the Centenary of His Birth

THE CHAIR OF LOGIC, INFORMATICS AND PHILOSOPHY OF SCIENCE
UNIVERSITY OF BIAŁYSTOK
Białystok 1998

Series: STUDIES IN LOGIC, GRAMMAR AND RHETORIC
edited by:
The Chair of Logic, Informatics
and Philosophy of Science
University of Białystok
ul. Liniarskiego 4
15-420 Białystok
e-mail: logika@hum.uw.bialystok.pl
tel. (085) 7457510

Refereed: prof. Jerzy Pogonowski

Editor for the volume: Halina Święczkowska
Cover design: Krzysztof Michałowski
Type-setting: Stanisław Żukowski

The volume was partly supported by KBN Grant No. 8 T11C 018 12

ISBN 83-87884-07-3
ISSN 0860-150X

WYDAWNICTWO UNIWERSYTETU W BIAŁYMSTOKU
15-097 Białystok, ul. M. Skłodowskiej-Curie 14, tel. (085) 7457059, 7457058

CONTENTS

Witold Marciszewski

CAN TURING-POST MACHINE PRODUCE PREREQUISITES FOR AUTOMATED REASONING?*

INTRODUCTION TO THIS VOLUME

Alan Turing appealed to humans that they should give a chance to digital machines, that is, not demand too much from them at so early stage of their evolution. The human brain, he argued, owing to a long process of evolutionary development, could succeed in winning qualities which in the moment cannot be enjoyed by our younger brothers.

That claim seems right. However, on the other hand, let us give humans a chance that happens to be denied by the adherents of strong AI. I mean those of them who care for "political correctness" in treating what they regard as a supressed minority, to wit machines. According to that approach, it is fair to speak of astonishing performances of computing machines, but not fair to remind how much they owe to the preparatory work of human experts.

Anyway, it is worth while to remind these prerequisites. Especially those involved in automated (ie, mechanized) reasoning, for it is a specially important activity expected from digital machines. When having this in view, we become capable of seeing the following problem: which of these preparatory tasks can be taken over by machines in a future? Are there, may be, any tasks which require a greater computational power than that of a digital machine? May such a power be possessed by analog machines alone (or still another brand of machines)? If so, is not so that the prerequisites for mechanical reasoning require that greater computational power?

* * *

* The work presented here was supported by the Polish Ministry of Science ('Komitet Badań Naukowych'), Committee for Informatics [etc.], under the grant to the Chair of Logic, Informatics and Philosophy of Science at the University of Białystok, No. 8 T11C 018 12, for research concerning *AI vs. Natural Intelligence from the Point of View of Mechanization of Reasoning.*

Before addressing the issues listed, this introductory comment should report on a session commemorating Emil L. Post's centenary of birth. The reason is that this session was to start a series of annual workshops to tackle the issues mentioned above. These can be summarized as dealing with the question of whether there should be systems surpassing the computational power of Turing-Post machine.

As for the term "Turing-Post Machine", this terminological usage follows that of Martin Davis who uses so the hyphen to hint at the convergence of Turing's and Post's seminal results (eg, in his paper on what is computing, in Arthur Sten's (Ed.) *Mathematics Today. Twelve Informal Essays*, Springer--Verlag 1979). There are, obviously, differences between Turing's and Post's definitions of an abstract machine (or, an algorithm or a program) but in the present context the differences are negligible. A fairly detailed comparison of these two machines is found in Z. Pawlak's *Logika dla Inżynierów* [Logic for Engineers], Warsaw, PWN 1970.

Why the centenary of Post's birth was celebrated in Poland in a special way? Let me answer starting from an anecdote.

1. Post's achievements and logical research in Poland

There is a story about Emil L. Post regarding a talk he had with Alfred Tarski. The latter playfully expressed his astonishment that Post, an American logician, contributed so much to propositional calculi, being that time a field explored mainly by Polish authors. Post – as reported by Tarski – replied that he would not see himself as an exception as he was also born in Poland.

He was actually born into a Jewish family on February 11, 1897 in Augustów, in the North-Eastern region of Poland, nowadays centred around Białystok as its regional capital; with his parents he emigrated to New York in 1904. Martin Davis, Post's pupil and the editor of his works, himself renowned for contributions to mechanical reasoning, when met collegaues busy at the same field in the University of Białystok, proved to be acquainted with the name of this city as mentioned by his master.

No wonder that the people interested in logic and computer science both in the University and the Politechnic in Białystok did not fail to recall the centenary of his birth. In December 1997, they arranged session in Białystok to survey Post's contributions to logic and foundations of informatics. Among the invited speakers there were eminent Polish scholars active in those fields to which Post contributed his famous results, to wit, multi-valued logics, algebras, logical matrices, algorithmic procedures. The present volume contains most of the papers read at that session, and some additional texts, including this introductory one.

Let it be added, after the year which passed from the Post Session in 1997, that in September 1998 there was held the Workshop (in the 50th Anniversary of Turing's Report *Intelligent Machinery*), entitled *Turing Machine and Mechanization of Reasoning*. The next Workshop, planned for 1999, is to discuss computational power of minds in terms of Turing-Post machine.

It is procedures of theorem-proving in what Białystok logicians are specially engaged. The leading role is played by the Section of Informatics in the Institute of Mathematics at the University of Białystok. The Section is run by Andrzej Trybulec, the author of Mizar – the system for representing mathematical knowledge, this representation being encoded into a specially devised language, and equipped with a software for proof checking and for knowledge management.

There is a close partnership between the Mizar team and the same University's Chair of Logic, Informatics and Philosophy of Science. (In Poland, a chair is an academic unit of the rank of an institute but of smaller size.) This Chair is engaged both in the automatic theorem-proving research and in philosophical foundations of such research, those going back to Leibniz (as manifested in the preceding volume of this Chair's series "Studies in Logic, Grammar and Rhetoric"), and those regarding the founders of modern logic. At the same time, the teaching of logic carried out by the Chair requires a theoretical reflection on relations between algorithmic methods of reasoning and those due to natural language and some inborn intelligence (as supposed to be enjoyed by our audiences).

When Tarski, as mentioned above, compared Post's contributions with those of Polish logicians of his time, he meant just a research in propositional calculi. However, some later research of Polish logicians is close to other Post's inquiries, namely those concerned with decidability and provability. As for decidability, it was Tarski himself to be merited for enormous contributions, continued by Andrzej Mostowski, Andrzej Grzegorczyk, and others.

The issue of provability entered a new stage with machine-assisted theorem proving (whose practising in Poland was exemplified above with the case of Mizar). This research too is close to Post's ideas, to wit his notion of a machine to calculate computable functions; among them a distinguished role is played by consequence operations in proving theorems.

2. From ideas recorded in an internal language to provability in formal systems

There is a complex evolutionary process from some primordial ideas to the provability of a sentence (of a formal system) which involves symbols which can get interpreted as standing for the ideas in question. For instance, before some propositions of Peano arithmetic became provable, there must have been a long evolution of the concept of natural number. Provability is a syntactic property of sentences that appears in a sophisticated process of abstraction, to wit the abstraction from semantic properties (meaning, etc) as possessed by the sentence in question at previous stages. Obviously, also at the start we can obtain meaningless strings of symbols constructed according to strict formation rules, we can select some strings as primitive, and then process such strings according to certain transformation rules; then provability is a property of sentences resulting from such transformations. In such a game there would be no need to disregard semantic properties since these would be lacking from the very start.

However, this is not the way people proceed when carry out projects of automated (or mechanized) theorem-proving (forming a substantial part of AI research). People wish that machines assist their looking for truth. Hence, they feed machines with what they regard as true axioms and reliable rules of inference, and wish that computers prove true theorems. Thus, the main problem consists in the suitable devising of axioms and rules so that they could be manipulated by a machine.

The problem attacked in this discussion is whether such evolution could be simulated by a Turing-Post machine. The problem might not arise, as common sense seems at once to suggest the answer "no", were not the case that some authors, including Turing, claimed that all cognitive processes can be carried out by Turing-Post machines. If so, then the development of concepts (that from primordial ideas to the stage of provability) would be no exception (let it be noted that Post himself was very far from such claims, characteristic of strong AI).

To attack this problem, one should not forget that the machine in question deals with discrete symbols alone, hence at any stage of evolution the things involved have to be represented by such symbols. However, the evolution starts from unverbalized ideas due to some perceptions (eg, perception of fire which after a process of generalization ends with creating a general name like "fire"). Hence – one must conjecture – such ideas are expressed in symbols of what has been called an *internal language or private language.*

This is a key notion in *The Language of Thought* by Jerry A. Fodor, The Harvester Press 1976, esp. the Section "Why There Has To Be a Private Language". For the present discussion the following Fodor's statement is of utmost consequence (the bracketed phrases are found in a relevant context of the quoted passage). "Computational models of such processes [considered action, concept learning, perceptual integration] are the only ones we've got. Computational models presuppose representational systems. But the representational systems of preverbal and infrahuman organisms surely cannot be natural languages. So either we abandon such preverbal and infrahuman psychology, or we admit some thinking, at least, isn't done in English." (p. 56) Let it be added that a machine language, an internal language of a computer, belongs to preverbal "psychology", and thus can, to same extent, be compared to a human internal language.

It is concept learning, including concept formation ("the learning from himself", as called by Plato and Augustine), that the present discussion is to pertain. Once having had concepts, we can form axiomatic propositions since they render relations involved in the contents of primitive concepts. Such axioms, in turn, enable other propositions to be provable. Provided there are discrete internal symbols to represent concepts, one can hope that the process of concept formation and its interplay with linguistic processes is likely to be rendered in terms of Turing-Post machine. Thus the way to provability leads through the following stages:

1. a concept appears in an internal language;
2. [this concept is put into a spoken language];
3. this concept is put into a written language;
4. this concept is put into a (written) formalized language;
5. the symbols of formalized language which express this concept are encoded into an arithmetical language feasible for a machine.

The proposition bracketed hints at a stage which may be omitted, but often it turns out to be useful, hence it is listed tentatively. Note also that the pronoun "this" is used here in a special way; the object referred to changes with passing from one stage to the next, as usual in any development; eg, the concept verbalized is not identical "with itself" as occurring at the preverbal level since it gets better defined (however, apart from such a pronoun there is no other linguistic device to render the process in question).

The keen interest in the problem of provability of propositions is due to its relation to the decision problem, belonging to the central issues of the theory of science. Let me recall the concepts involved.

The *decision procedure* for a formalized system S is a method of deciding in each case whether a given sentence stated in the language of S is provable by means of devices available in S. The *decision problem* for S is the problem of whether there is a decision procedure for S. A system is called *decidable* or undecidable according to the answer to the decision problem is in the affirmative or in the negative.

The notion of provability was being explained above with various contexts. Let this be summed up in the following definition. A sentence is said to be *provable* in a system S if a proof of that sentence exists; that is, a sequence of sentences exists whose last term is the given sentence and each of whose terms is either an axiom of S or is deducible from preceding terms (or class of preceding terms) of the sequence by the inference rules. Obviously, the notion of provability, so defined, is a syntactical notion, the term "syntax", obviously, being referred here to an external language.

If there is an internal language (as argued above), and any language and any process of problem solving have to consist of discrete symbols (as claimed by Turing), then the internal language also must consist of discrete signals; moreover, these must be ruled by a syntactic system (otherwise a set of signals would not deserve to be called a language).

The above statement is of consequence for the strategy of AI debate. Now, the adherents of strong AI will win if they give us a precise description of an internal language being the source of a formal system, say, Peano arithmetic; let that supposed internal language be called IS(Ar) ("IS" for "internal source"). They should then define the vocabulary and syntax of IS(Ar); or, at least, should prove that such a definition is possible (if one must wait still for relevant empirical research in biology). Moreover, they should produce (or discover) an algorithm of encoding symbol strings of IS(Ar) into symbol strings of the formalized Peano arithmetic.

Anyway, their task will not end with that. There is a very important phenomenon in the development of human thought (both in individual histories and in history of civilization) that consists in gaining ever greater clarity of the concepts involved. According to the strategy suggested above, this process should be also accounted for in terms of transition from an internal langauge to a formal system (ie, one with well defined provability). Such a transition involves, let me recall, an algorithm of encoding the internal language in question into the respective language of a formal system (which thus becomes a mirror of its internal source). This issue, sophisticated enough, requires special attention to be paid in the next Section.

3. Solvability requires clear notions. How these are to be attained?

A terminological remark is in order at the start. In what follows there appear the terms "decidability" and "solvability". In a sense, they refer to the same situation, while their use depends on the context in the following way. Solvability is a property of problems, while decidability is a property of theories; a problem is said to be solvable if there is a decidable theory in which this question can be answered. Thus, whenever we discuss one of them, the other is within sight too.

This Section continues the issue of the capabilities of Turing-Post (ie, digital) machine, compared with those of the mind, in the following point. Solvability of a problem, or decidability of a respective theory, which amounts to provability of relevant theorems, requires clear terms in which the problem could be stated. However, such a clarity results from an evolutionary process starting somewhere at the level of an internal language. Could such a process be carried out by a digital machine?

In looking for an answer, let me start from recalling a method of discussion which is characteristic of the AI debate but which seems to lead to nowhere. Then another approach, based on the idea of internal language will be suggested.

There is a curious epistemological puzzle about the problem of decidability. The same formal results produced by Gödel, Turing, Church, and Post, induce people to dramatically differring conclusions as for capabilities of the human mind. One thing is sure, as a mathematical result: there are problems unsolvable for a machine as defined by Turing and Post, hence unsolvable for a digital computer. Then some authors, let us call them Mentalists, reason as follows.

> People answer such unsolvable questions when find that some formally unprovable sentence is true, for instance that being known as the Gödelian sentence. Hence some mental abilities exceed possibilities of digital computer.

On the other hand, there are Antimentalists who argue as follows.

> This is a subjective feeling of a human that he/she solves a problem which a machine cannot solve. However, such a feeling lacks objective validation, since objectively validated is solely what is computed by Turing machine. Moreover – the argument runs further – what is unsolvable for a particular machine can be solved by another machine fed with stronger axioms and/or rules. Exactly, likewise a wiser man can solve problems which exceed abilities of one being less wise. Thus, there is no essential difference between human and mechanical intelligence.

A Mentalist may reply that

> his learned collegaue prudently used the passive voice 'fed' when speaking about axioms, rules, and programs as employed by a machine. This is to mean, there must have been a human who equipped the machine with such devices, and thus the human mind surpasses its mechanical counterpart.

His opponent would riposte that

> again, the difference is spurious since people are also fed with principles, rules and programs either by other people, as in education, or by Nature, called also Evolution, which provides humans and other living beings with genetic software. Moreover, when a machine is employed as a teacher to instruct a human, it modifies a human software likewise a human programmer controlling a machine.

Thus, arguments of neither side are conclusive. Therefore, instead of comparing and assessing performances of humans and machines, we should rather reflect on the laws of mathematical creativity, as encouraged by Post.

Let us note (as discussed below) that the very essence of the creative power of humans lies in their ability of having thoughts which lack clearness. Is this compatible with being a discrete state machine, such as a digital computer? If not, then there does exist an essential difference between humans and computers. Such compatibility seems dubious since the discrete state machines, as known so far, owe their discreteness to the use of discrete external (available to an observer) symbols, while thoughts lacking clearness may exist without external symbols. However, it may be that such a thought reduces to a sequence of neuronal signals which are as discrete as symbols appearing at the tape of Turing machine; to prove their discreteness is a challenge to Antimentalists.

Unclearness is crucial for the development of human understanding. It is a driving force in the process of forming a language, the process supported by some inferences, as can be exemplified by the history of the concept of infinity.

It is the creative insight of Galileo which caused anxiety for paradoxical consequences of the concept of infinite set of integers. Galileo noticed that the multitude of all squares, cubes, etc equals to that of all integers. Let me dwell upon that story as a suggestive exemplification. Here is a text by Galileo to give us an impression of his dilemma (note, when Galileo speaks of numbers, he means integers).

> "If I assert that all numbers, including both squares and non-squares, are more than the squares alone, I shall speak truth, shall I not?"

After this is confirmed by an interlocutor of the dialogue, Galileo puts forward an argument for the opposite.

> "If I should ask further how many squares there are one might reply truly that there are as many as the corresponding number of roots, since every square has its own root and every root its own square, while no square has more than one root and no root more than one square.
>
> But if I inquire how many roots there are, it cannot be denied that there are as many as there are numbers because every number is a root of some square. This being granted we must say that there are as many squares as there are numbers because they are just as numerous as their roots, and all numbers are roots. Yet at the outset we said there are many more numbers as squares, since the larger portion of them are not squares."

The above passage is quoted after Galileo Galilei (in the English translation) *Dialogues Concerning Two New Sciences*, translated by Henry Crew and Alfonso de Salvio, Dover Publications, 1914, p. 32. Let it be noted that even more instructive case of dealing with paradoxes of infinity is found in G. W. Leibniz's "Accessio ad arithmeticam infinitorum" (1672) included in the 1st vol. of the mathematical series of Leibniz's Works, published by Akademie--Verlag, Berlin 1976 (that case deserves a special treatment in another essay).

It would require too much space to follow Galileo's reasoning to solve this paradox. The quoted passage may suffice to trace a part of a way from less clear to more clear a concept.

A member of a primitive tribe in which the concept of number is very far from being clear, has no troubles of the kind discussed by Galileo. The problem could not be raised until one defined some functions of integers (squares, etc) and, moreover, attained at the idea of an infinite set of integers.

At such a stage, there appears a person, like Galileo, who notices the problem of consistency of the conceptual system in question. The problem results from two chains of inferences which produce mutually contradictory conclusions. Attempts at making the system free from contradiction involve new conceptual constructions and new reasonings, as those which led Dedekind, Cantor and others to the concepts of countable sets, continuum, etc. At the new stage, though much more sophisticated, there may again appear contradictions, as the set-theoretical antinomies. Then one needs new conceptual constructions (as those in the theory of types, or those in ZF), and so on.

Now the question arises whether such processes of elucidating concepts can be carried out by Turing-Post machine. One thing is certain: new symbols have to be invented to render new perceptions. For instance, the operation of finding the square of a given integer should be properly symbolized in order to be manageable by a machine. Once symbolized, it becomes manageable in a mechanical way; but what about new perceptions and rendering them with suitably created symbols?

Is it possible for a machine to carry out such processes? If so, there must be an algorithm to encode concepts as records in an internal language into external (observable) symbols to express these concepts, and such symbols should be produced by the machine. I do not deny that such an algorithm may exist. However, I insist on the conditional conjecture: if the encoding algorithm does exist, what is being encoded into external symbols amounts to neuronal records. These records would be physical counterparts of what is given in human consciousness as concepts capable of increasing clarity.

The proposed conjecture is crucial for any AI project dealing with provability. Artificial intelligence should not only prove theorems in a symbolic language fed into a machine. It should also be able to create such a language. The creation would require having unverbalized concepts (resulting from perceptions of the external world), then verbalizing them in more and more precise a language, up to a symbolic formalized system; only then when such symbols are produced, they can be manipulated in the procedure of automatic theorem-proving.

Tu sum up, the problem of this discussion can be stated as a problem of symmetry. There are two sets of processes ordered in time, and separated by the stage at which some sentences become provable. The processes following that stage are those of successive mechanizing the theory in question, as planned in projects of mechanizing mathematical activity. It is that part which can be carried out by Turing-Post machine. Is there the case that the earlier processes, those being a prerequisite for the stage of provability, can be also carried out by Turing-Post machine? This is the question. Perhaps the question of *to be or not to be* for strong AI.

Witold Marciszewski
Chair of Logic, Informatics and Philosophy of Science
at the University of Białystok
e-mail: witmar@calculemus.org
home page: www.calculemus.org/wit.html

STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2 (15)

Roman Murawski

E. L. POST AND THE DEVELOPMENT OF MATHEMATICAL LOGIC AND RECURSION THEORY

Abstract

In the paper the contribution of Emil L. Post (1897–1954) to mathematical logic and recursion theory will be considered. In particular we shall study: (1) the meaning of results of his doctoral dissertation to the development of the metamathematical studies of propositional calculus, (2) the significance of his studies on canonical systems for the theory of formal languages as well as for the foundations of computation theory, (3) his anticipation of Gödel's and Church's results on incompleteness and undecidability, (4) his results on the undecidabiity of various algebraic formal systems and finally (5) his contribution to establishing and to the development of recursion theory as an independent domain of the foundations of mathematics. At the end some remarks on the philosophical and methodological background of his results will be made.

1. Post's doctoral dissertation

Most of scientific papers by Emil Leon Post were devoted to mathematical logic and to the foundations of mathematics[1].

His first paper in logic was the doctoral dissertation from 1920 published in 1921 in *American Journal of Mathematics* under the title "Introduction to a general theory of elementary propositions" (cf. [14]). It was written under the influence of *Principia Mathematica* by A.N. Whitehead and B. Russell (cf. [24]) on the one hand (Post participated in a seminar

[1] Post published 14 papers and 19 abstracts (one of the papers was published after his death): 4 papers and 8 abstracts were devoted to algebra and analysis and 10 papers and 11 abstracts to mathematical logic and foundations of mathematics.

led by Keyser at Columbia University devoted to *Principia*) and *A Survey of Symbolic Logic* by C.I. Lewis (cf. [9]). Post's doctoral dissertation contained the first metamathematical results concerning a system of logic. In particular Post isolated the part of *Principia* called today the propositional calculus, introduced the truth table method and showed that the system of axioms for the propositional calculus given by Whitehead and Russell is complete, consistent and decidable (in fact Post spoke about the finiteness problem instead of decidability). He also proved that this system is complete in the sense called today after him, i.e., that if one adds to this system an unprovable formula as a new axiom then the extended system will be inconsistent. It is worth mentioning here that the completeness, the consistency and the independence of the axioms of the propositional calculus of *Prinicipia* was proved by Paul Bernays in his *Habilitationsschrift* in 1918 but this result has not been published untill 1926 (cf. [1]) and Post did not know it.

In Post's dissertation one finds also an idea of many-valued logics obtained by generalizing the 2-valued truth table to m-valued truth tables ($m \geq 2$). Post proposed also a general method of studying systems of logic (treated as systems for inferences) by finitary symbol manipulations (later he called such systems canonical systems of the type A) – hence he considered formal logical systems as combinatorial systems.

In doctoral dissertation Post mentioned also his studies of systems of 2-valued truth functions closed under superposition. In particular he showed that every truth function is definable in terms of negation and disjunction.

Those latter considerations were developed in the monograph *The Two-Valued Iterative Systems of Mathematical Logic* from 1941 (cf. [16]). One finds there among others the result called today Post's Functional Completeness Theorem. It gives a sufficient and necessary condition for a set of 2-valued truth functions to be complete, i.e., to have he property that any 2-valued truth function is definable in terms of it. Post distinguished five properties of truth functions. His theorem says that a given set X of truth functions is complete if and only if for every of those five properties there exists in X a truth function which does not have it. Did Post prove this theorem? In [16] one finds no proof satisfying the standards accepted today. The reason for that was Post's baroque notation (it was in fact an unprecise adaptation of the imprecise notation of Jevons from his *Pure Logic*, cf. [7]), other reason was the fact that Post seemed to be simultaneously pursuing several different topics. The proof of Post's theorem satisfying today's standards can be found in the paper by Pelletier and Martin [13].

2. Canonical systems

During a year stay at Princeton University 1920–21 (he was awarded the post-doctoral Procter Fellowship) Post studied mainly canonical systems. Those systems founded the theory of formal languages. Post's results in this field were the anticipation of famous results by Gödel and Church on the incompleteness and undecidability of first order logic.

The investigations mentioned above were connected with the following ideas: the whole *Principia Mathematica* can be considered as a canonical system of type A, i.e., as a system of signs in an alphabet which can be manipulated according to a given set of rules. Post wanted to solve the decision problem (in his terminology: the finiteness problem) for canonical systems of type A. In his way he wanted to find a method which would decide for any given formula of the system of *Principia* whether it is formally provable in it or not. Since *Principia* are a formalization of the whole of mathematics, this method would give a decision procedure for the whole of mathematics.

Besides canonical systems of type A Post introduced systems of type B and C. Just the latter are known today as Post's production systems. Let us define them using the contemporary terminology and notation.

Let Σ be a finite alphabet. Its elements are called terminals. One uses also non-terminal symbols $P_{\cdot}$. A canonical production has the form:

$$
\begin{array}{cccccccc}
g_{11} & P_{i'_1} & g_{12} & P_{i'_2} & \cdots & g_{1m_1} & P_{i'_{m_1}} & g_{1(m_1+1)} \\
g_{21} & P_{i''_1} & g_{22} & P_{i''_2} & \cdots & g_{2m_2} & P_{i''_{m_2}} & g_{2(m_2+1)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
g_{k1} & P_{i^{(k)}_1} & g_{k2} & P_{i^{(k)}_2} & \cdots & g_{km_k} & P_{i^{(k)}_{m_k}} & g_{k(m_k+1)} \\
 & & & & \Downarrow & & & \\
g_1 & P_{i_1} & g_2 & P_{i_2} & \cdots & g_m & P_{i_m} & g_{m+1}
\end{array}
$$

where g_k are strings on the alphabet Σ, P_k are variable strings (non-terminals) and each of the P's in the line following the $\Downarrow$ also occurs as one of the P's above $\Downarrow$. A system in canonical form of type C consists of a finite set of strings (Post called them initial assertions) together with a finite set of canonical productions. It generates of course a subset of Σ^* (= the set of all finite strings over Σ) which can be obtained from the initial assertions by the iterations of canonical productions. Today it is known that such sets are exactly recursively enumerable languages.

Post proved the equivalence of canonical systems of type A, B and C. He also showed that this part of *Principia* which corresponds to the first-order predicate calculus can be formalized as a canonical system of type B and consequently also of type C. Moreover the set of all provable formulas of the system of *Principia* can be regarded as a set of strings of symbols which can be generated by certain canonical system of type C.

Post proved also the important Normal Form Theorem for canonical systems of type C. Let us say that a canonical system of type C is normal if and only if it has exactly one initial assertion and each of its productions has the form

$$gP \Longrightarrow P\overline{g}.$$

A set of strings $U \subseteq \Sigma^*$ is said to be normal if and only if there exists a normal system on an alphabet Δ containing Σ and generating a set $\mathcal{N}$ such that $U = \mathcal{N} \cap \Sigma^*$. Post's Normal Form Theorem says now that if $U \subseteq \Sigma^*$ is a set of strings generated by some canonical system of type C then U is normal (a proof can be found in [17] and [22]).

One of the consequences of Normal Form Theorem is that a decision procedure for canonical systems of type C would induce a decision procedure for the whole system of *Principia*.

Post started his reasearches in this direction by considering normal systems of a special form, so-called tag systems. Tag systems are normal systems in which all g's in production rules are of the same length (but not necessarily the $\overline{g}$'s) and $\overline{g}$'s depend only on the first symbol of the appropriate string g. In particular Post started by studying the following simple system:

$$agP \Longrightarrow Paa,$$

$$bgP \Longrightarrow Pbbab,$$

where $g \in \{a,b\}^*$ and $|g| = 2$. This case turned out to be difficult (the question whether such systems are decidable is open till today).

Post supposed that tag systems are recursively undecidabe. This was proved by M. Minsky in 1961 (cf. [11]).

Note that the procedure of generating expressions by a canonical system of type C is similar to the process of generating computable functions from certain given initial functions by iterations of certain given operations on functions[2]. Hence those systems can be regarded as a formalism making precise the intuitive notion of effective computability. Consequently one can

[2] It was shown later that the definition of effective computability by Post's canonical systems is equivalent to the definitions in the language of λ-definability, of recursive functions or by Turing machines.

formulate a thesis analogous to Church's thesis and called Post's thesis which says that any finitely given language is generated by rules of some canonical normal system.

An application of Cantor's diagonal method led Post to the conclusion that the decision problem for normal systems has a negative solution. In 1921 Post sketched a formal proof of this. He wrote in [22] (pp. 421–422):

> We (...) conclude that the finiteness problem for the class of all normal systems is unsolvable, that is, there is no finite method which would uniformly enable us to tell of an arbitrary normal system and arbitrary sequence on the letters thereof whether that sequence is or is not generated by the operations of the system from the primitive sequence of the system.

Those considerations led him also to the conclusion about the incompleteness, i.e., to the conclusion that:

> *not only was every (finitary) symbolic logic incomplete relative to a certain fixed class of propositions (...) but that every such logic was extendable relative to that class of propositions*

(cf. [17]). In [17] he also wrote:

> *No normal-deductive-system is equivalent to the complete logical system (if such there be); better, given any normal-deductive-system there exists another which second proves more theorems (to put it roughly) than the first*

and he added

> *A complete symbolic logic is impossible.*

Those results anticipated results by Gödel and Church on the incompleteness and undecidability of systems of first-order logic (cf. [6], [2] and [3]). Post knew of course that his results are, as he wrote, "fragmentary". He never published them (though in 1941, hence already after the publication of Gödel's and Church's results) he tried to publish results of his investigations from 1920–1921).

What was his reaction to the appearance of the Gödel's paper [6]? On the one hand he was disappointed that not his name will be connected with results he anticipated and on the other he admired the genius and contribution of Gödel. In a postcard to Gödel sent on 19th October 1938 Post wrote:

> I am afraid that I took advantage of you on this, I hope but our first meeting. But for fifteen years I had carried around the thought of astounding the mathematical world with my unorthodox ideas, and meeting the man chiefly responsible for the vanishing of that dream rather carried me away.
>
> Since you seemed interested in my way of arriving at these new developments perhaps Church can show you a long letter I wrote to him about them. As

for any claims I might make perhaps the best I can say is that I would have *proved* Gödel's Theorem in 1921 – had I been Gödel.

And in the letter to Gödel from 30th October 1938 he wrote:

... after all it is not ideas but the execution of ideas that constitute a mark of greatness.

Considering decidability problem one should mention Post's contribution to making precise the notion of effective computability. Post was of the opinion that Herbrand-Gödel's and Church-Kleene's definitions were both lacking in that neither constituted a "fundamental" analysis of the notion of algorithmic process. In 1936 (cf. [15]) he proposed a definition based on the operations of marking an empty box and erasing the mark in a marked box. Note the similarity with the definition given at the same time by A. Turing (cf. [23]). The difference between both approaches consists is that Turing formulated his definition in terms of an idealized computer while Post in terms of a program (a list of instructions written in a given language).

At the beginning of the 40's Post wrote a paper in which he tried to describe his studies on the incompleteness and undecidability from 1920–1921 anticipating Gödel's and Church's results. It is the paper "Absolutely unsolvable problems and relatively undecidable propositons – account of an anticipation". It was submitted in 1941 to *American Journal of Mathematics*. In a letter to H. Weyl accompanying the manuscript Post explained why he did not publish his results twenty years earlier and wants to do it now, i.e., after the publications by Gödel and Church. Among reasons he mentions problems he had with publishing his earlier papers (in particular of [14] and [16]) which did not find a recognition and appreciation by mathematicians as well as the problems with the health which delayed the preparation of full detailed proofs. Though the editors appreciated the significance of Post's investigations and results, the paper has been rejected. Communicating this decision H. Weyl wrote in a letter to Post from 2nd March 1942:

... I have little doubt that twenty years ago your work, partly because of its then revolutionary character, did not find its due recognition. However, we cannot turn the clock back; in the meantime Gödel, Church and others have done what they have done, and the American Journal is no place for historical accounts; ... (Personally, you may be comforted by the certainty that most of the leading logicians, at least in this country, know in a general way of your anticipation.)

Only a small part of Post's paper has been published, i.e., the part containg his Normal Form Theorem (cf. [17]). The full version of the paper

"Absolutely unsolvable problems and relatively undecidable propositions – account of an anticipation" was published posthumously in 1965 in Davis' book *The Undecidable.*

3. Recursion theory

Main Post's results which found the recognition and appreciation belong to the recursion theory. They were of course connected with his investigations described above.

Considering Post's results in the recursion theory one must tell first of all about his paper "Recursively enumerable sets of positive integers and their decision problems" (cf. [18]). It turned out to be the most influential of his publications and foundamental for the whole recursion theory. Recursion theory was for the first time presented in it as an autonomous branch of mathematics.

One finds there:

- a theorem called today Post's Theorem and stating that a set X is recursive if and only if the set X and its complement $\neg X$ are recursively enumerable[3],
- a theorem stating that (a) any infinite recursively enumerable set has an infinite recursive subset and (b) there exists a recursively enumerable set which is not recursive.

Main subject of the considered paper is the mutual reducibility of recursively enumerable sets. Recall that a set X is said to be many-one reducible to a set Y if and only if there exists a recursive function f such that

$$x \in X \equiv f(x) \in Y.$$

If the function f is one-one then we say about one-one reducibility.

Post proved the existence of a recursively enumerable set K which is complete with respect to many-one (one-one) reducibility, i.e., such that any recursively enumerable set X is many-one (one-one) reducible to K. Hence K has a maximal degree of unsolvability with respect to many-one (one-one) reducibility. Post constructed also a recursively enumerable set which is simple, i.e., has the property that there exists no infinite recursively enumerable subset of its complement. Such a set cannot be of a maximal degree with respect to one-one reducibility. This led Post to the formulation of the fol-

[3] Recall that a set X is recursively enumerable if and only if there exists a recursive function f such that X is the image of f if and only if there exists a recursive relation R such that $x \in X \equiv \exists y R(x, y)$.

lowing problem called today Post's problem: does there exist a recursively enumerable but not recursive set of a degree lower than the degree of the complete set K with respect to a given type of reducibility?

Post did not succeed in solving this problem. Though he proved in 1948 (cf. [21]) the existence of sets of a degree lower than the degree of the set K but those sets were not recursively enumerable.

Post's problem was solved independently by A. A. Muchnik (cf. [12]) and R. Friedberg (cf. [5]) by a new method introduced by them – and called the priority method. This method proved later to be really fruitful – many results in the recursion theory, in particular in the theory of degrees, have been obtained by it. They showed that there exist two recursively enumerable sets A and B such that A is not recursive in B and B is not recursive in A.

As a consequence one obtains that degrees of unsolvability (even the degrees of recursively enumerable sets) are not lineary ordered and that there are recursively enumerable degrees other than the degree of recursive sets and the degree of the complete set (sets A and B constructed by Friednerg and Muchnik are examples of such sets).

Discussing here Post's paper "Recursively enumerable sets of positive integers and their decision problems" one should mention the way in which Post presented his results in it. This way became a norm and a standard in the recursion theory. It consisted in giving rather informal proofs with a description of intuitions. Post saw the need of providing formal proofs but on the other hand he wrote:

> ... the real mathematics involved must lie in the informal development. For in every instance the informal "proof" was first obtained; and once gotten, tranforming it into the formal proof turned out to be a routine chore.

The considered paper had important long term effects. It was the beginning of extensive studies of recursively enumerable sets, in particular of various types of reducibility. Here one can also see the source of such important notion as polynomial time reducibility or of studies connected with NP-completeness.

Post continued his studies of degrees of unsolvability in the paper "Degrees of recursive unsolvability (preliminary report)" (cf. [21]) where he generalized the notion of a degree to the case of any sets (not necessarily recursively enumerable) and proved the existence of a pair of incomparable degrees (both were lower than the degree of the complete recursively enumerable set K). He announced also a theorem (called today Post's theorem) stating that a set X is recursive in a set $A \in \Sigma^0_n(\Pi^0_n)$ if and only if X is a Δ^0_{n+1} set.

One should also mention here the joint paper by Post with S. C. Kleene "The upper semi-lattice of degrees of recursive unsolvability" (cf. [8]). One finds there among others a theorem stating that the ordered set of degrees of unsolvability contains a densely ordered subset.

4. Undecidability of algebraic combinatorial systems

Discussing Post's contribution to the recursion theory one should say about his results on undecidability of algebraic combinatorial systems. They provided "mathematical" examples of undecidable problems.

Let us start with the correspondence problem. It can be formulated as follows. A correspondence system is a finite set of ordered pairs $(g_1, h_1), (g_2, h_2), \ldots, (g_n, h_n)$ such that $g_i, h_i \in \Sigma^*$, Σ a given finite alphabet. Such a system is said to be solvable if there exists a sequence $i_1, i_2, \ldots, i_k$ such that $1 \le i_1, i_2, \ldots, i_k \le n$ and

$$g_{i_1} g_{i_2} \ldots g_{i_k} = h_{i_1} h_{i_2} \ldots h_{i_k}.$$

The corrspondence problem is to provide an algorithm for determining of a given correspondence system whether it is solvable or not. Post proved in the paper "A variant of a recursively unsolvable problem" (cf. [19]) that such an algorithm does not exist, hence the correspondence problem is not recursively solvable. This result plays an important role in the theory of formal languages.

Church suggested to Post to study also the decidability problem for Thue's systems known also as the word problem for monoids or semi-groups. It was formulated by the Norwegian mathematician Axel Thue in 1914 and can be formulated as follows: Let Σ be a finite alphabet. Define an equivalence relation $\approx$ in Σ^* by giving a finite set of pairs of words for which this equivalence holds, i.e., by putting

$$u_1 \approx v_1, u_2 \approx v_2, \ldots, u_n \approx v_n,$$

and closing this under the substitution of v_i for u_i $(i = 1, \ldots, n)$. The problem consists now in providing an algorithm for determining of an arbitrary pair $(u, v) \in \Sigma^* \times \Sigma^*$ of strings whether or not $u \approx v$. In the paper "Recursive unsolvability of a problem of Thue" (cf. [20]) Post gave an example of a set of initial pairs defining an equivalence relation $\approx$ for which the word problem is unsolvable. In the proof Post used Turing machines (in fact he showed that the theory of Turing machines can be interpreted in terms of the word problem in such a way that an algorithm for the latter could be tranformed into an algorithm for a problem concerning Turing machines known to be unsolvable). Post also gave a very careful technical critique of Turing's paper [23].

It is worth adding that the recursive unsolvability of the word problem was established independently by A. A. Markov in 1947 (cf. [10]) who based his proof on Post's normal systems.

5. Post's philosophical and methodological ideas

Discussing the works and results of Post in the field of mathemtaical logic and recursion theory one should consider their philosophical and methodological background.

Post – similarly as Gödel – emphasized the significance of the absolutness and the fundamental character of the notion of recursive solvability. He attempted also to explain the notion of provability – more exactly, he wanted to find a precise notion which would explain the intuitive notion of provability in arithmetic in such a way as the notion of recursiveness explains the notion of effective computability and solvability. He hoped that this will enable us to find absolutely undecidable arithmetical propositions. Later (before the death) he added to this also the problem of providing an absolute explication and explanation of the general mathematical notion of definability (he was convinced that this should be done even before giving the absolute explication of the notion of provability).

Post emphasized (cf. [22], p. 64):

> I study Mathematics as a product of the human mind and not as absolute.

He was convinced that mathematical thinking is in fact creative. He wrote in [22], p. 4:

> ... mathematical thinking is, and must be, essentially creative.

He thought also that the human capacity to know cannot be closed and reduced to a formal system. And added (cf. [22], p. 4):

> ... creativeness of human mathematics has a counterpart inescapable limitations thereof – witness the absolutely unsolvable (combinatory) problems.

On the other hand he was convinced that the results on the undecidability and incompleteness indicate that human capacity to know with respect to mathematics are in fact bounded in spite of the creativeness of the mathematical thinking. He wrote in [22], p. 56:

> The unsolvability of the finiteness problem for all normal systems, and the essential incompleteness of all symbolic logics, are evidences of limitations in man's mathematical powers, creative though these be.

Post claimed that there exist absolutely undecidable (i.e., unsolvable by no methods and means) propositions and that there is no complete system of logic[4]. A consequence of this was in his opinion the fact that (cf. [22], p. 55):

> logic must not only in some parts of its description (as in the operations), but in its very operation be informal. Better still, we may write *The Logical Process is Essentially Creative.*

Consequently the human mind can never be replaced by a machine. He wrote in [22], p. 55:

> We see that a *machine* would never give a complete logic; for once the machine is made *we* could prove a theorem it does not prove.

6. The significance of Post's results for the development of the mathematical logic and the foundations of mathematics

The above considerations lead us to the conclusion that Post's works and results contributed very much to the development of mathematical logic and the foundations of mathematics. His works (together with works of J. Łukasiewicz) initiated the investigations on many-valued logics and on the Post algebras connected with them. His studies of the propositional calculus (the results of which were included in his doctoral dissertation) were the first metamathematical studies of a system of logic. Most significant were probably his works and results in the recursion theory. They contributed very much to establishing this field as an autonomous branch of the foundations of mathematics. They began intensive studies on degrees of unsolvability, in particular of recursively enumerable degrees, investigations on the (un)decidability of various systems, in particular combinatorial systems in algebra and on the various types of recursive reducibility. They influenced also the researches in the computer science (though Post showed no interest in computers). They were also very important for the theory of formal languages.

Post's investigations and results were in a sense ahead of his time, were precursory (compare his anticipation of Gödel's and Church's results descri-

[4] In [22], p. 54, he wrote: "*A complete symbolic logic is impossible*".

bed above). This had of course negative consequences as the fear of being not understood properly and the delay of publication of the results. Problems with health, in particular the illness under which he suffered almost the whole life, also hindered him from publishing the results at proper time. Many of Post's results were left in an incomplete form. He tried the whole time to improve his results and to find the most general form which also caused some delay. Nevertheless his contributio to the mathemtaical logic and to the foundations of mathematics was really significant.

7. References

[1] Bernays P., Axiomatische Untersuchungen des Aussagenkalküls der *Principia Mathematica*, *Mathematische Zeitschrift* 25 (1926), 305–-320.

[2] Church A., An unsolvable problem of elementary number theory, *American Journal of Mathematics* 58 (1936), 345–363.

[3] Church A., A note on the Entscheidungsproblem, *Journal of Symbolic Logic* 1 (1936), 40–41. Correction, *ibid,*, 101–102.

[4] Davis M. (Ed.), *Solvability, Provability, Definability: The Collected Works of Emil L. Post*, Birkhäuser, Boston-Basel-Berlin 1994.

[5] Friedberg R., Two recursively enumerable sets of incomparable degress of unsolvability (solution of Post's problem), *Proceedings of the National Academy of Sciences (USA)* 43 (1957), 236–238.

[6] Gödel K., Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme. I, *Monatshefte für Mathematik und Physik* 38 (1931), 173–198.

[7] Jevons W.S., *Pure Logic, or the Logic of Quality apart from Quantity with remarks on Boole's System and the Relation of Logic and Mathematics*, E. Stanford, London-New York 1864.

[8] Kleene S.C. and Post E.L., The upper semi-lattice of degrees of recursive unsolvability, *Annals of Mathematics* 59 (1954), 379–407; reprinted in [4], pp. 514–542.

[9] Lewis C.I., *A Survey of Symbolic Logic*, University of California Press, Berkeley 1918.

[10] Markov A.A., Nievazmoznost' niekatorych algoritmov v teorii asotiativnych sistem (On the impossibility of certain algorithms in the theory of associative systems), *Doklady Akademii Nauk SSSR* 55 (1947), 587–590; English translation in: *Comptes rendus de l'académie des Sciences de l'URSS* 55 (1947), 583–586.

[11] Minsky M., Recursive unsolvability of Post's problem of tag and other topics in the theory of Turing machines, *Annals of Mathematics* 74 (1961), 437–455.

[12] Muchnik A..A., Nerazreshimost' problemy svodimosti algoritmov (Negative answer to the problem of reducibility of the theory of algorithms), *Doklady Akademii Nauk SSSR* 108 (1956), 194–197.

[13] Pelletier F.J., Martin N.M, Post's functional completeness theorem, *Notre Dame Journal of Formal Logic* 31 (1990), 462–475.

[14] Post E.L., Introduction to a general theory of elementary propositions, *American Mathematical Journal* 43 (1921), 163–185; reprinted in J. van Heijenoort, *From Frege to Gödel. A Source Book in Mathematical Logic 1879–1931*, Harvard University Press, Cambridge, Mass., 1967, pp. 264–283, ; also in [4], pp. 21–43.

[15] Post E.L., Finite combinatory processes – Formulation I, *Journal of Symbolic Logic* 1 (1936), 103–105; reprinted in [4], pp. 103–105.

[16] Post E.L., *The Two-Valued Iterative Systems of Mathematical Logic*, Princeton University Press, Princeton 1941; reprinted in [4], pp. 249–374.

[17] Post E.L., Formal reductions of the general combinatorial decision problem, *American Journal of Mathematics* 65 (1943), 197–215; reprinted in [4], pp. 442–460.

[18] Post E.L., Recursively enumerable sets of positive integers and their decision problems, *Bulletin of the American Mathematical Society* 50 (1944), 284–316; reprinted in [4], pp. 461–494.

[19] Post E.L., A variant of a recursively unsolvable problem, *Bulletin of the American Mathematical Society* 52 (1946), 264–268; reprinted in [4], pp. 495–500.

[20] Post E.L., Recursive unsolvability of a problem of Thue, *Journal of Symbolic Logic* 12 (1947), 1–11; reprinted in [4], pp. 503–513.

[21] Post E.L., Degrees of recursive unsolvability (preliminary report), *Bulletin of the American Mathematical Society* 54 (1948), 641–642; reprinted in [4], pp. 549–550.

[22] Post E.L., Absolutely unsolvable problems and relatively undecidable propositions – account of an anticipation, first published in Davis M. (Ed.), *The Undecidable*, Raven Press, New York 1965, pp. 340–433; reprinted in [4], pp. 375–441.

[23] Turing A., On computable numbers with an application to the Entscheidungsproblem, *Proceedings of London Mathematical Society*, ser. 2, 42 (1936–37), 230–265. Correction, *ibid.*, 43 (1937), 544–546.

[24] Whitehead A.N. and Russell B., *Principia Mathematica*, Cambridge University Press, Cambridge, vol. I 1910, vol. II 1912, vol. III 1913.

Roman Murawski
Faculty of Mathematics and Computer Science
Adam Mickiewicz University
e-mail: rmur@math.amu.edu.pl

STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2 (15)

Ewa Orłowska

POST ALGEBRAS AND POST LOGICS

Emil Post doctoral dissertation Post (1921) contained a description of an n-valued, functionally complete algebra, for a finite $n \geq 2$. The notion of Post algebra was introduced in Rosenbloom (1942). In Rousseau (1969, 1970) an equivalent formulation of the class of Post algebras was given which became a starting point for extensive research. Since then various generalisations of Post algebras inspired by applications in computer science have been developed. This note in a brief survey of major classes of Post algebras.

1. Plain semi-Post algebras

These algebras were introduced and investigated in Cat Ho (1973), Cat Ho and Rasiowa (1987, 1989, 1992). Let $(T, \leq)$ be a poset and let ET be the set of ideals of T together with the empty set $\emptyset$. Clearly, $T \in ET$. It is known that any $s \in ET$ is of the form $s = \bigcup\{s(t) : s(t) \subseteq s\}$, where $s(t) = \{w \in T : w \leq t\}$. The system $(ET, \subseteq)$ is a complete lattice, where join and meet are set-theoretical union and intersection, respectively.

An abstract algebra

(P) $\quad \mathbf{P} = (P, \cup, \cap, \rightarrow, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$

where $\cup$, $\cap$, $\rightarrow$ are 2-argument operations, $\neg$, d_t for $t \in T$ are unary operations and e_s for $s \in ET$ are 0-argument operations (constants) is a plain semi-Post algebra (psP-algebra) of type T provided that the following conditions are satisfied:

(p0) $\quad (P, \cup, \cap, \rightarrow, \neg)$

is a Heyting algebra with the zero element $\mathbf{0} = e_\emptyset$ and the unit element $\mathbf{1} = e_T$, For any $a, b \in P$

(p1) $d_t(a \cup b) = d_t a \cup d_t b$,

(p2) $d_t(a \cap b) = d_t a \cap d_t b$,

(p3) $d_w d_t a = d_t a$,

(p4) $d_t e_s = \mathbf{1}$ if $t \in s$, otherwise $d_t e_s = \mathbf{0}$

(p5) $d_t a \cup \neg d_t a = \mathbf{1}$

(p6) $a = \bigcup\{e_{s(t)} \cap d_t a : t \in T\}$ where $\bigcup$ is the least upper bound in P

Let $\mathbf{P} = (P, \cup, \cap, \rightarrow, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$ be a psP-algebra of type $(T, \leq)$.

By B_P we denote the set of elements of P of the form $d_t a$, $t \in T$. Then

Proposition 1.1

(a) B_P is closed under the operations $\cup, \cap, \rightarrow, \neg$ of P

(b) The algebra $\mathbf{B}_P = (B_P, \cup, \cap, \rightarrow, \neg, \mathbf{1}, \mathbf{0})$ is a Boolean algebra.

Let C_P be the set of all complemented elements in the distributive lattice $(P, \cup, \cap)$. Then

Proposition 1.2

(a) C_P is closed under the operations $\cup, \cap, \rightarrow, \neg$ of P

(b) The algebra $\mathbf{C}_P = (C_P, \cup, \cap, \rightarrow, \neg, \mathbf{1}, \mathbf{0})$ is a Boolean algebra

(c) For every $a \in C_P$, $d_t \neg a = \neg d_t a$, $t \in T$.

However, B_P and C_P do not always equal. Consider a poset $(T, \leq)$ such that $T = \{a, b, c\}$ and $\leq = \{(b, a)\}$. Then $ET = \{\emptyset, \{b\}, \{c\}, \{b, c\}, \{a, b\}, T\}$, $B_P = \{\emptyset, T\}$, and $C_P = \{\emptyset, T, \{c\}, \{a, b\}\}$.

Proposition 1.3 (Epstein lemma)

For any set $\{a_j : j \in J\}$ of elements in P it holds

(a) $a = (\mathbf{P})\bigcup\{a_j : j \in J\}$ iff for every $t \in T$ $d_t a = (\mathbf{B}_P)\bigcup\{d_t a_j : j \in J\}$

(b) $a = (\mathbf{P})\bigcap\{a_j : j \in J\}$ iff for every $t \in T$ $d_t a = (\mathbf{B}_P)\bigcap\{d_t a_j : j \in J\}$

where $(\mathbf{P})\bigcup$, $(\mathbf{P})\bigcap$, $(\mathbf{B}_P)\bigcup$, $(\mathbf{B}_P)\bigcap$ denote joins and meets in the algebras $\mathbf{P}$ and $\mathbf{B}_P$, respectively.

Proposition 1.4

(a) $d_t(a \rightarrow c) = \bigcap\{d_w a \rightarrow d_w c : w \leq t\}$

(b) $d_t \neg a = \bigcap\{\neg d_w a : w \leq t\}$

(c) $d_w a \leq d_t a$ whenever $w \leq t$, for any $w, t \in T$

(d) $a \leq b$ iff $d_t a \leq d_t b$ for all $t \in T$

(e) $e_w \leq e_t$ iff $w \subseteq t$, for any $w, t \in ET$

It follows that every psP-algebra of type $(T, \leq)$ uniquely determines the set

$M(P) = \{\bigcap\{d_w a \rightarrow d_w c : w \leq t\} : t \in T\}$ of infinite meets of P.

Observe that for any sets $s', s'' \in ET$ there exists the relative pseudo--complement $s' \rightarrow s''$ defined by

$$s' \rightarrow s'' = \bigcup\{s \in ET \rightarrow s' \cap s \subseteq s''\}$$

and the pseudo-complement $\neg s'$ defined by

$$\neg s' = s' \rightarrow \emptyset == \bigcup\{s \in ET : s' \cap s = \emptyset\}$$

Clearly, $s' \rightarrow s''$, $\neg s' \in ET$.

Proposition 1.5

(a) For any poset $(T, \leq)$, the system $(ET, \cup, \cap, \rightarrow, \neg, T, \emptyset)$, where $\cup$, $\cap$ are set-theoretical operations of union and intersection, respectively, and $\rightarrow$, $\neg$ are defined as above, is a Heyting algebra with the unit element T and zero element $\emptyset$.

(b) Given a psP-algebra $\mathbf{P} = (P, \cup, \cap, \rightarrow, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$, let $EP = \{e_s : s \in ET\}$.
Then $(EP, \leq)$ is a poset isomorphic to $(ET, \subseteq)$.

Condition (b) follows from Proposition 1.4(e).

Example 1.1

An important example of a psP-algebra is the following algebra, referred to as a basic psP-algebra:

$$(ET, \cup, \cap, \rightarrow, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$$

where $(ET, \cup, \cap, \rightarrow, \neg)$ is the Heyting algebra defined above and the operations d_t, $t \in T$, and e_s, $s \in ET$, are defined by:

$$e_s = s, \text{ in particular } e_\emptyset = \emptyset \text{ and } e_T = T,$$

$$d_t s = T \text{ if } t \in s, \text{ otherwise } d_t s = \emptyset.$$

Proposition 1.6

The basic psP-algebra is functionally complete, that is any n-argument operation $f: ET^n \to ET$, $n = 0, 1, \ldots$, is definable with the operations of this algebra.

Given a Boolean algebra $\mathbf{B} = (B\cup, \cap, \to, -, 1_B, 0_B)$ and a poset $(T, \leq)$, by a descending T-sequence of elements of B we mean an indexed family $(b_t)_{t\in T}$ of elements of B such that $w \leq t$ in T implies $b_t \leq b_w$ in B (for the sake of simplicity we denote the Boolean ordering of B with the same symbol). We say that B and T satisfy condition (erpc) of existence of relative pseudo-complement if

(erpc) For any two descending T-sequences $b = (b_t)_{t\in T}$, $c = (c_t)_{t\in T}$ of elements of B there exists $(\mathbf{B})\bigcap\{b_w \to c_w : w \leq t\}$ for all $t \in T$.

Example 1.2

We present a psP-algebra $\mathbf{P}_T(\mathbf{B})$ of type T determined by a Boolean algebra $\mathbf{B} = (B, \cup, \cap, \to, -, 1_B, 0_B)$ such that $\mathbf{B}$ and T satisfy condition (erpc). The universe $P(B)$ of $\mathbf{P}_T(\mathbf{B})$ is the set of all descending T-sequences of elements of B. We define a partial ordering $\leq$ on $P(B)$ as follows. Let $b = (b_t)_{t\in T}$ and $c = (c_t)_{t\in T}$ be any elements of $P(B)$. Then

$$b \leq c \text{ in } P(B) \text{ iff } b_t \leq c_t \text{ in } B \text{ for all } t \in T.$$

The system $(P(B), \leq)$ is a lattice with join and meet defined by

$$b \cup c = (b_t \cup c_t)_{t\in T},\ b \cap c = (b_t \cap c_t)_{t\in T}.$$

Since B and T satisfy (erpc), for any b, c in $P(B)$ there exists the relative pseudo-complement $b \to c$ and

$$b \to c = (x_t)_{t\in T}, \text{ where } x_t = \bigcap\{b_w \to c_w : w \leq t\}.$$

For every $s \in ET$ we define

$$e_s = (x_t)_{t\in T}, \text{ where } x_t = 1_B \text{ if } t \in s, \text{ otherwise } x_t = 0_B.$$

Moreover, we put

$$d_w b = (x_t)_{t\in T}, \text{ where } x_t = b_w \text{ for every } t \in T,$$

$$\neg b = (x_t)_{t\in T}, \text{ where } x_t = \bigcap\{-b_w : w \leq t\}.$$

It is easy to verify that the algebra

$$\mathbf{P}_T(\mathbf{B}) = (P(B), \cup, \cap, \to, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$$

defined above is a psP-algebra of type $(T, \leq)$.

Proposition 1.7

Let a Boolean algebra $\mathbf{B}$ and a poset $(T, \leq)$ satisfying (erpc) be given. Let $\mathbf{P}$ be the algebra $\mathbf{P}_T(\mathbf{B})$ defined as in Example 1.2. Then the algebra $\mathbf{B}_P$ (see Proposition 1.4) is isomorphic to $\mathbf{B}$.

Example 1.3

A particular instance of the algebra defined in Example 1.2 is a set algebra obtained by taking the field of all subsets of a set as the respective Boolean algebra. Let U be a nonempty set and let $B(U)$ be the field of all subsets of U. We have $1_{B(U)} = U$ and $0_{B(U)} = \emptyset$. For any poset $(T, \leq)$, $B(U)$ and T satisfy condition (erpc). Let $P(B(U))$ be the set of all descending T-sequences of sets from $B(U)$. The ordering on $P(B(U))$ is the inclusion. The algebra $\mathbf{P}_T(\mathbf{B}(U)) = (P(B(U)), \cup, \cap, \to, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$ defined as in Example 1.2 is a psP-algebra of type $(T, \leq)$. The infinite joins in the axiom (p6) are set unions.

Proposition 1.8 (Representation theorem)

Let $\mathbf{P} = (P, \cup, \cap, \to, \neg, \{d_t : t \in T\}, \{e_s : s \in ET\})$ be a psP-algebra of type $(T, \leq)$. If T is denumerable and either well-founded or the set $M(P)$ is denumerable (in particular if P is denumerable), then for any denumerable set Q of infinite joins and meets in P there exists the field $B(U)$ of all subsets of a nonempty set U and a monomorphism h from $\mathbf{P}$ into $\mathbf{P}_T(\mathbf{B}(U))$ preserving all the operations in Q.

2. Post algebras of order m

The first axiom system for the algebras characterising Post's m-valued logics, for a finite m greater than 2, was presented in Rosenbloom (1942). He called them Post algebras. The axiomatisation was then simplified in Epstein (1960) and Traczyk (1964). Traczyk proved the equational definability of the class of Post algebras. Over the years the theory of Post algebras and several generalisations of these algebras have been developed. Here we define Post algebras of order m as a particular case of psP-algebras.

Let $(T_m, \leq)$ be a poset such that $T_m = \{1, \ldots, m-1\}$, where m is a natural number greater than 2, and $\leq$ is a natural ordering in T_m. Then $ET_m = \{\emptyset, s(1), \ldots, s(m-1)\}$, where $s(t) = \{w \in T_m : w \leq t\}$. Clearly, $(ET_m, \subseteq)$ is isomorphic to $\{0, 1, \ldots, m-1\}$ with the natural ordering. Hence, we can identify these two posets and assume that constants e_s are indexed with elements from $\{0, 1, \ldots, m-1\}$.

By a Post algebra of order m we mean a psP-algebra of type $(T_m, \leq)$.

It can be easily shown that this definition is equivalent to the standard definition of Rousseau (1969, 1970).

Example 2.1

A classical example of a Post algebra of order m is an m-element Post algebra such that $P = \{e_0, \ldots, e_{m-1}\}$, and for $i, j \in \{0, 1, \ldots, m-1\}$ the operations in P are defined as follows:

(ex1) $e_i \cup e_j = e_{\max(i,j)}$,

(ex2) $e_i \cap e_j = e_{\min(i,j)}$,

(ex3) $e_i \to e_j = \mathbf{1}$ if $i \leq j$, otherwise $e_i \to e_j = e_t j$,

(ex4) $\neg e_i = e_i \to \mathbf{0}$,

(ex5) $d_i e_j = \mathbf{1}$ if $i \leq j$, otherwise $d_i e_j = \mathbf{0}$.

Proposition 2.1

(a) $e_0 \to a = e_{m-1}$

(b) $e_s \to a = \bigcup\{d_t a \cap e_t : t < s\} \cup d_s a$

(c) $e_{m-1} \to a = a$

(d) $a \to e_s = e_s \cup \neg d_{s+1} a$, for $s = 0, \ldots, m-2$

(e) $a \to e_{m-1} = e_{m-1}$.

We define disjoint operations c_s for $s \in \{0, 1, \ldots, m-1\}$ as follows:

(c1) $c_0 a = \neg d_1 a = \neg a$

(c2) $c_s a = d_s a \cap \neg d_{s+1} a$ for $s \in T \setminus \{m-1\}$

(c3) $c_{m-1} a = d_{m-1} a$.

We clearly have

$$c_s a \cap c_t a = e_0 \text{ for } s \neq t.$$

Any element a of P has the following disjoint representation:

(c4) $a = \bigcup\{c_t a \cap e_t : t \in T\}$.

Theorems analogous to propositions 1.1,...,1.8 hold and constructions from examples 1.2 and 1.3 carry over to the case of type $(T_m, \leq)$. Algebras presented in Example 1.1 can be identified with those defined in Example 2.1. Moreover, the algebras $\mathbf{B_P}$ and $\mathbf{C_P}$ corresponding to a Post algebra of order m coincide.

The m-valued Post logic is a propositional logic with binary connectives $\vee, \wedge, \to$, unary connectives $\neg$, D_t for $t \in T$, and propositional constants E_s for $s \in \{0, 1, \ldots, m-1\}$. The algebraic semantics for the logic is determined in the standard way by the class of Post algebras of order m. A Hilbert-style axiomatisation of m-valued Post logic and its completeness with respect to the algebraic semantics is presented in Rasiowa (1969). The main results on m-valued Post logic include: Model existence theorem (Rasiowa 1970), Craig interpolation theorem (Rasiowa 1970), Herbrand theorem (Perkowska 1972).

Applications of the m-valued Post logic are concerned with the theory of programming. An algorithmic logic based on m-valued Post logic is developed in Perkowska (1972).

Post algebras and logics of any finite type $(T, \leq)$ are considered in Nour (1997). They are also treated in Konikowska, Morgan and Orlowska (1998).

3. Post algebras of order ω^+

Let $(T_\omega, \leq)$ be a poset such that $T_\omega = \omega$ is the set of natural numbers and $\leq$ is the natural ordering of natural numbers. Then $ET_\omega = \{\emptyset, s(1), s(2), \ldots, T_\omega\}$. Clearly, $(ET_\omega, \subseteq)$ is isomorphic to $\{0, 1, 2, \ldots, \omega\}$ with the natural ordering. Hence, we can identify these two posets and assume that constants e_s are indexed with elements from $\{0, 1, \ldots, \omega\}$.

A Post algebra of order ω^+ is a psP-algebra of type $(T_\omega, \leq)$.

An example of a Post algebra of order ω^+ can be defined in a way similar to that developed in Example 2.1.

Theorems analogous to propositions 1.1,...,1.8 hold and constructions from examples 1.2 and 1.3 carry over to the case of type $(T_\omega, \leq)$. Moreover, the algebras $\mathbf{B_P}$ and $\mathbf{C_P}$ corresponding to a Post algebra of order ω^+ coincide.

Representation theory for Post algebras of order ω^+ has been also developed in Maksimova and Vakarelov (1974), Rasiowa (1985).

A Hilbert-style axiomatisation of ω^+-valued Post predicate logic and its completeness with respect to the algebraic semantics is presented in Rasiowa (1973). The other results on ω^+-valued Post logic include: Kripke style semantics (Maximova and Vakarelov 1974, Vakarelov 1977), Herbrand theorem and a resolution-style proof system (Orlowska 1977, 1978, 1980), relational semantics and a relational proof system (Orlowska 1991).

Applications of the ω^+-valued Post logic are concerned with the theory of programming. An algorithmic logic based on ω^+-valued Post logic is developed and investigated in Rasiowa (1974a, 1977, 1979).

4. Post algebras of order $\omega + \omega^*$ (in a strict sense)

These algebras are introduced and investigated in Epstein and Rasiowa (1990, 1991). Let $T = \{1, 2, \ldots, -2, -1\}$ and $E = \{0, 1, 2, \ldots, -2, -1\}$. A Post algebra of order $\omega + \omega^*$ is an algebra of the form (P) satisfying axioms (p0),...,(p6), where in (p0) $\mathbf{0} = e_0$ and $\mathbf{1} = e_{-1}$, and the following

(p7) $\quad d_1a = d_{-1}a \cup \bigcup\{d_sa \cap \neg d_{s+1}a : 1 \leq s \leq -1\}$ pivot elimination axiom

(p8) $\quad (a \to b) \cup (b \to a) = \mathbf{1}$

The axiom (p7) says that an element e such that $e_t \leq e$ $(d_te = \mathbf{1})$ for all positive t and $e < e_t$ $(d_te = \mathbf{0})$ for all negative t does not exist.

Propositions analogous to Propositions 1.1, 1.2, 1.3, 1.4 hold for Post algebras of order $\omega + \omega^*$. Moreover, the algebras $\mathbf{B_P}$ and $\mathbf{C_P}$ corresponding to a Post algebra of order $\omega + \omega^*$ coincide.

Example 4.1

A most natural example of a Post algebra of order $\omega + \omega^*$ is a linear Post algebra of order $\omega + \omega^*$ defined as follows:

> $P = \{e_s : s \in E\}$, and the operations in P are defined with conditions analogous to (ex1),...,(ex5) from Example 2.1.

Disjoint operations in Post algebras of order $\omega + \omega^*$ can be defined with conditions analogous to (c1), (c2), (c3) from section 2 by replacing $m - 1$ with -1. Then any element a of P has a disjoint representation given by condition (c4).

In Post algebras of order $\omega + \omega^*$ one can define arithmetic-like operations in the following way.

The successor sa (the predecessor pa) of an element a of P is an element given by the following disjoint representation

$$sa = \bigcup\{c_ta \cap e_{t+1} : t \in E\}$$

$$pa = \bigcup\{c_ta \cap e_{t-1} : t \in E\}$$

provided that either of these exist.

The inverse $-a$ of an element a is given by the disjoint representation

$$-a = \bigcup\{c_ta \cap e_{-1} : t \in T\}$$

provided that it exists.

Addition and multiplication operations have disjoint representations as follows

$$a + b = \bigcup\{c_t(a + b) \cap e_t : t \in T\}$$

where for each $t \in T$ the infinite join $c_t(a+b) = \bigcup\{c_ia \cap c_jb : i + j = t\}$ exists,

$$a \cdot b = \bigcup\{c_t(a \cdot b) \cap e_t : t \in T\}$$

where for each t there is the finite join $c_t(a \cdot b) = \bigcup\{c_ia \cap c_jb : ij = t\}$.

Proposition 4.1

A Post algebra of order $\omega + \omega^*$ with inverse, addition and multiplication is a commutative ring with unit, where the ring zero is e_0 and the ring unit is e_1.

These rings have the characteristic 0.

For a descending T-sequence $X = (X_t)_{t \in T}$ of subsets from the field $B(U)$ of all subsets of a nonempty set U we define

$$X^+ = \bigcap\{X_t : t \text{ positive}\}$$

$$X^- = \bigcup\{X_t : t \text{ negative}\}.$$

It can be shown that the algebra of descending T-sequences $X = (X_t)_{t \in T}$ of sets from $B(U)$ such that $X^+ = X^-$, with the operations defined as in Example 1.2 is a Post algebra of order $\omega + \omega^*$.

Representation theorem Post algebras of order $\omega + \omega^*$ has the following form.

Proposition 4.2 (Representation theorem)

For every denumerable Post algebra **P** of order $\omega + \omega^*$ there is a monomorphism h of **P** into a Post set algebra of order $\omega + \omega^*$ whose elements are descending T-sequences $X = (X_t)_{t \in T}$ of sets from the field $B(U)$ of all subsets of a nonempty set U such that $X^+ = X^-$. Moreover, h preserves a given denumerable set Q of infinite joins and meets of **P**.

Applications of the logic are concerned with approximation reasoning. An approximation reasoning to recognise a subset S of a nonempty universe U is understood as a process of gradual approximating S by

subsets of U $\ S \subseteq S_1 \subseteq S_2 \subseteq \ldots$ which cover S

and subsets $\ldots \subseteq S_{-2} \subseteq S_{-1} \subseteq S$ which are contained in S.

Then the approximations of set S are defined as follows:

$$S^+ = \bigcap \{S_t : t \text{ positive}\}$$

$$S^- = \bigcup \{S_t : t \text{ negative}\}.$$

In Epstein and Rasiowa (1991) a characterisation of sets S such that $S^+ = S^-$ is given.

Post algebras of order ϑ, where ϑ is an arbitrary ordinal number are introduced and investigated in Przymusinska (1980, 1980a, 1980b, 1980c).

Bibliography of Post algebras

Cat-Ho, Nguyen. (1973) Generalised Post algebras and their applications to some infinitary many-valued logics. Dissertationes Mathematicae 57, 1-76.

Cat-Ho, Nguyen and Rasiowa, H. (1987) Semi-Post algebras. Studia Logica 46, No 2, 147-158.

Cat-Ho, Nguyen and Rasiowa, H. (1989) Plain semi-Post algebras as a poset-based generalisation of Post algebras and their representability. Studia Logica 48, No 4, 509-530.

Cat-Ho, Nguyen and Rasiowa, H. (1992) LT-fuzzy logics. In: Zadeh, L. A. and Kacprzyk, J. (eds) Fuzzy Logic for Management of Uncertainty. J. Wiley, New York, 121-139.

Chang, C. C. and Horn, A. (1961) Prime ideals characterisation of generalised Post algebras. Proceedings of the Symposium in Pure Mathematics 2, 43-48.

Dwinger, Ph. (1966) Notes on Post algebras I, II. Indagationes Mathematicae 28, 462-478.

Dwinger, Ph. (1968) Generalised Post algebras. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 16, 559-563.

Dwinger, Ph. (1977) A survey of the theory of Post algebras and their generalisations. In: Dunn, J. M. and Epstein, G. (eds) Modern Uses of Multiple-Valued Logic. Reidel, Dordrecht, 53-75.

Epstein, G. (1960) The lattice theory of Post algebras. Transactions of the American Mathematical Society 95, 300-317.

Epstein, G. and Horn, A. (1974) P-algebras, An abstraction from Post algebras. Algebra Universalis 4, 195-206.

Epstein, G. and Rasiowa, H. (1990) Theory and uses of Post algebras of order $\omega + \omega^*$, Part I. Proceedings of the 20th International Symposium on Multiple-Valued Logic, Charlotte, NC, USA, 42-47.

Epstein, G. and Rasiowa, H. (1990) Theory and uses of Post algebras of order $\omega + \omega^*$, Part II. Proceedings of the 21th International Symposium on Multiple-Valued Logic, Victoria, Canada, 248-254.

Konikowska, B., Morgan, Ch. and Orlowska, E. (1998) Relational formalisation of arbitrary finite valued logics. Logic Journal of the Interest Group in Pure and Applied Logic, to appear.

Maksimova, L. and Vakarelov, D. (1974) Semantics for ω^+-valued predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 22, No 8, 765-771.

Maksimova, L. and Vakarelov, D. (1974) Representation theorems for generalised Post algebras of order ω^+. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 22, No 8, 757-764.

Nour, A. (1997) Etude de modeles logiques. Extensions de la logique intuitionniste. PhD dissertation, University Joseph Fourier, Grenoble.

Orlowska, E. (1977) The Herbrand theorem for ω^+-valued logic. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 25, No 11, 1069-1071.

Orlowska, E. (1978) The resolution principle for ω^+-valued logic. Fundamenta Informaticae II, 1-15.

Orlowska, E. (1980) Resolution systems and their applications II. Fundamenta Informaticae III, 333-362.

Orlowska, E. (1985) Mechanical proof methods for Post logics. Logique et Analyse 110-111, 173-192.

Orlowska, E. (1991) Post relation algebras and their proof system. Proceedings of the 21th International Symposium on Multiple-Valued Logic, Victoria, Canada, 298-305.

Perkowska, E. (1971) Herbrand theorem for theories based on m-valued logics. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 19, 893-899.

Perkowska, E. (1972) On algorithmic m-valued logics. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 20, 717-719.

Post, E. (1920) Introduction to a general theory of elementary propositions. Bulletin of the American Mathematical Society 26.

Post, E. (1921) Introduction to a general theory of elementary propositions. American Journal of Mathematics 43, 163-185.

Przymusinska, H. (1980) On ϑ-valued infinitary predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 28, 193-198.

Przymusinska, H. (1980a) Some results concerning ϑ-valued infinitary predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 28, 199-202.

Przymusinska, H. (1980b) Genzen type semantics for ϑ-valued infinitary predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 28, 203-206.

Przymusinska, H. (1980c) Craig interpolation theorem and Hanf number for ϑ-valued infinitary predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 28, 207-211.

Rasiowa, H. (1969) A theorem on existence of prime filters in Post algebras and the completeness theorem for some many-valued predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 17, 347-354.

Rasiowa, H. (1970) Ultraproducts of m-valued models and a generalisation of the Lowenheim-Skolem-Godel-Malcev theorem for theories based on m-valued logics. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 18, 415-420.

Rasiowa, H. (1972) The Craig interpolation theorem for m-valued predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 20, 341-346.

Rasiowa, H. (1973) On generalised Post algebras of order ω^+ and ω^+-valued predicate calculi. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 21, 209-219.

Rasiowa, H. (1973) Formalised ω^+-valued algorithmic systems. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 21, 559-565.

Rasiowa, H. (1974) An Algebraic Approach to Non-Classical Logics. North Holland and Polish Scientific Publishers, Amsterdam, Warsaw.

Rasiowa, H. (1974a) Extended ω^+-valued algorithmic logic (a formalised theory of programs with recursive procedures). Bulletin of the Polish Academy of Sciences, Mathematics, vol. 22, 605-610.

Rasiowa, H. (1977) Many-valued algorithmic logic as a tool to investigate programs. In: Epstein, G. and Dunn, J. M. (eds) Modern Uses of Multiple-Valued Logic, Reidel, Dordrecht, 77-102.

Rasiowa, H. (1979) Algorithmic logic. Multiple-valued extension. Studia Logica 38, No 4, 317-325.

Rasiowa, H. (1985) Topological representations of Post algebras of order ω^+ and open theories based on ω^+-valued Post logic. Studia Logica 44, No 4, 353-368.

Rasiowa, H. (1992) Towards fuzzy logic. In: Zadeh, L. A. and Kacprzyk, J. (eds) Fuzzy Logic for Management of Uncertainty. J. Wiley, New York, 5-25.

Rasiowa, H. (1994) Axiomatization and completeness of uncountably-valued approximation logic. Studia Logica 53, No 1, 137-160.

Rasiowa, H. and Epstein, G. (1987) Approximation reasoning and Scott's information systems. Proceedings of the 2nd International Symposium on Methodologies for Intelligent Systems, North Holland, 33-42.

Rosenbloom, P. C. (1942) Post algebras, I. Postulates and general theory. American Journal of Mathematics 64, 167-188.

Rousseau, G. (1969) Logical systems with finitely truth values. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 17, 189-194.

Rousseau, G. (1970) Post algebras and pseudo-Post algebras. Fundamenta Mathematicae 67, 133-145.

Saloni, Z. (1972) Gentzen rules for the m-valued logic. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 20, 819-826.

Traczyk, T. (1963) Some theorems on independence in Post algebras. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 10, 3-8.

Traczyk, T. (1964) An equational definition of a class of Post algebras. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 12, 147-149.

Traczyk, T. (1967) On Post algebras with uncountable chain of constants. Algebras and homomorphisms. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 15, 673-680.

Traczyk, T. (1968) Prime ideals in generalized Post algebras. Bulletin of the Polish Academy of Sciences, Mathematics, vol. 16, 369-373.

Urquhart, A. (1973) An interpretation of many- valued logic. Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik 19, 111-114.

Vakarelov, D. (1977) Lattices related to Post algebras and their applications to some logical systems. Studia Logica 36, 89-107.

Wade, L. I. (1960) Post algebras and rings. Duke Mathematical Journal 12, 389-395.

Zabel, N. (1993) New techniques for automated deduction in many--valued first order, finite and infinite logics. PhD thesis, Institut National Polytechnique de Grenoble.

Ewa Orłowska
Institute of Telecommunications
Warsaw
e-mail: orlowska@itl.waw.pl

Grzegorz Malinowski

MANY-VALUED POST LOGICS

Emil Post's doctoral dissertation appeared in print as Post (1921)[1] is one of the major logic works of the first half of the 20th Century. It's contribution to the semantic and metalogic of the classical propositional calculus can hardly be overestimated. Besides of that one of the parts of the dissertation contained the description of the n-valued ($n \geq 2$, n finite) functionally complete algera of logic and its interpretation in terms of the $(n-1)$ dimentional Euclidean space.

Emil Post independently from Łukasiewicz introduced his (finitely) many-valued logics in 1920[2]. The main motivation of the actual consideration of truth values next to the truth and falsity stemmed from his work on functional completeness properties of the classical propositional logic. Though apparently tailored algebraically, the fuctionally complete Post logic and algebras have been playing an important role in the area of philosophical logic as well as in advanced applications in Computer Science.

The aim of the paper is a concise presentation of Post work on many-valued logic and its evolution.

1. Post matrices

The fundamental, many–valued constructions of Post are connected with two, primitive in the Principia Mathematica, propositional connectives: negation ($\neg$) and disjunction ($\vee$). For any natural $n \geq 2$ Post builds an n–valued logical algebra on the linearly ordered set of objects

[1] Abstracted as Post (1920).

[2] The priority of Łukasiewicz is unquestioned, though it concerns merely the introduction of the three-valued logic in 1918, widely published as Łukasiewicz (1920), and thus the first construction many-valued logic at all. In Łukasiewicz (1929) one finds the generalization of the original concept onto the case of finite and infinite (countable and continuum) number of values.

$$P_n = \{t_1, t_2, \ldots, t_n\}$$

($t_1 < t_j$ iff $i < j$) equipped with two operations: unary cyclic *rotation* $\neg$ (*cyclic negation*) and binary disjunction $\vee$, defined in the following way:

$$\neg t_i = \begin{cases} t_{i+1} & \text{if } i \neq n \\ t_1 & \text{if } i = n \end{cases}; \qquad t_i \vee t_j = t_{max(i,j)}.$$

The disjunction function fixes on a natural and entirely intuitive meaning of the disjunction connective, typical at least for the most known many–valued (including infinite) constructions. In plain terms, the logical value of disjunctive proposition equals the greater of the values of its components. The function of cyclic rotation permutes, in some specified manner, the set P_n and the negation corresponding to it is, the case $n = 2$ being excluded, quite special – compare the beside table. It is just the fact of combining of the latter with an appropriate binary function of algebra on P_n that warrants the functional completeness of that algebra, i.e. it ensures that by means of the primitive functions, there can be defined every finitely–argument function on P_n, including constant functions and hence the objects $t_1, t_2, \ldots, t_n$. For a given finite $n \geq 2$ the algebraic structure:

a	$\neg a$
t_1	t_2
t_2	t_3
$\vdots$	$\vdots$
t_n	t_1

$$\mathcal{P}_n = (\{t_1, t_2, \ldots, t_n\}, \neg, \vee)$$

will be called *n–valued Post algebra.*

The matrix P_n naturally associated with the algebra $\mathcal{P}_n$:

$$P_n = (\{t_1, t_2, \ldots, t_n\}, \neg, \vee, \{t_n\})$$

will, in the sequel, be referred to as the (basic) *n–valued Post matrix.* It is easily seen that the two–valued Post matrix is isomorphic to the negation–disjunction matrix for the classical propositional calculus. To check it, one must replace t_1 in P_2 by the falsity (0) and t_2 by the symbol of truth 1. Simultaneously, however, the matrices P_n for $n > 2$ are totally incompatible to the mentioned classical matrix, which is the result of nonstandard mode of the negation connective. Hence, for instance, for $n = 3$ t_3 could be the only counterpart of "truth" respective the adopted interpretation of disjunction but then t_1 would have to correspond to "falsity" as $\neg t_3 = t_1$, which should not take place because $\neg\neg t_3 = \neg t_1 = t_2 \neq t_3$. A contradiction.

It is remarkable that among the laws of n–valued logics determined by Post matrices P_n there are many–valued counterparts of some significant

classical tautologies expressed in terms of negation and disjunction connectives, including the "generalized law of the excluded middle":

$$p \vee \neg p \vee \neg\neg p \vee \ldots \vee \underbrace{\neg\neg\ldots\neg}_{(n-1)\text{ times}} p.$$

On the other hand, however, the connectives of implication, conjunction and equivalence introduced up to the two–valued pattern

$$\alpha \to \beta = \neg\alpha \vee \beta,$$

$$\alpha \wedge \beta = \neg(\neg\alpha \vee \neg\beta) \quad \text{and} \quad \alpha \equiv \beta = (\alpha \to \beta) \wedge (\beta \to \alpha)$$

evidently stray away from their classical counterparts. Thus, e.g. the connective $\to$ resembles no "reasonable" connective of implication and $\wedge$ is neither associative nor symmetric. Accordingly, due to the second remark the formula

$$\neg(p \wedge \neg p \wedge \neg\neg p \wedge \ldots \wedge \underbrace{\neg\neg\ldots\neg}_{(n-1)\text{ times}} p)$$

is not among the laws of n-valued Post logic. One may also check that this formula as it stands, without internal parentheses, is even not well formed.

The property of functional completeness of Post logic algebras, i.e. the property that every finitary mapping $f\colon P_n^m \to P_n$ ($m \geq 0, m$ finite) can be represented as a composition of the operations $\neg$ and $\vee$, warrants that all counterparts of classical connectives are definable in $\mathcal{P}_n$. This implies that the whole classical logic may be interpreted within any many-valued system of Post logic.

2. Interpretation

Post did manage to present a semantical interpretation for his nonstandard matrices P_n providing the following, Euclidean in its spirit, construction of the "spaces" E^{n-1}:

(1) Elements $P \in E^{n-1}$ are $(n-1)$–element tuples of ordinary two–valued propositions (represented by small letters), $P = (p_1, p_2, \ldots, p_{n-1})$ subject to the condition that the true propositions are listed before the false.

(2) $\neg P$ is formed by replacing the first false element of P by its denial, $\neg P = (\neg(p_1 \wedge p_2 \wedge \ldots \wedge p_n), \neg(p_1 \wedge p_2 \wedge \ldots \wedge p_{n-1}) \wedge (p_1 \vee p_2), \ldots, \neg(p_1 \wedge p_2 \wedge \ldots \wedge p_{n-1}) \wedge (p_{n-2} \vee p_{n-1}))$, the connectives on the right–hand side

are the usual (classical) connectives; but if there is no false element in P, then all of them are to be denied, in which case $\neg P$ is a sequence of false propositions.

(3) When $P = (p_1, p_2, \ldots, p_{n-1})$ and $Q = (q_1, q_2, \ldots, q_{n-1})$, then $P \vee Q = (p_1 \vee q_1, p_2 \vee q_2, \ldots, p_{n-1} \vee q_{n-1})$ (with the right–hand side $\vee$ as above).

The mapping $\underline{i}: E^{n-1} \to P_n$:

$$\underline{i}(P) = t_i \text{ iff } P \text{ contains exactly } (i-1) \text{ true propositions}$$

establishes an isomorphism of $(E^{n-1}, \vee, \neg)$ onto the Post algebra P_n.

The discussed interpretation shows, among others, that similarly stated logical values appearing in diverse matrices P_n are, regarding the author's intention, objects different from each other. Therefore, for the sake of precision the symbols of logical values t_i should be always indexed by a parameter assigning them to a given matrix.

Example. Five-valued Post logic based on the set $P_5 = \{t_1, t_2, t_3, t_4, t_5\}$ of values may be interpreted in E^4 consisting of the following elements:

$$(0,0,0,0)$$
$$(1,0,0,0)$$
$$(1,1,0,0)$$
$$(1,1,1,0)$$
$$(1,1,1,1)$$

which correspond to t_1, t_2, t_3, t_4, t_5, respectively. While $\vee$ of E^4 "coincides" with Boolean sum on the axes, an application of $\neg$ results in descending step down on the array with the exception that $\neg(1,1,1,1) = (0,0,0,0)$.

Urquhart (1973) gave an interpretation of Post logics in terms of a Kripke-style semantics

$$K_n = (\, S_n\,,\, \leq\,,\, \vdash\,),$$

where $S_n = \{0, 1, ..., n-2\}$, $\vdash \subseteq S_n \times For$ and the relation $\leq$ is transitive:

(Tr) If $x \vdash \alpha$ and $x \leq y \in S_n$, then $y \vdash \alpha$.

The conditions stating the sense of Post connectives are the following:

$x \vdash \neg\alpha$ iff $y \vdash \alpha$ for no $y \in S_n$ or there is a $y \in S_n$ such that $y < x$ and $y \vdash \alpha$

$x \vdash \alpha \vee \beta$ iff $x \vdash \alpha$ or $x \vdash \beta$.

Several meanings may be attached to "reference points" $x \in S_n$. Urquhart suggests a temporal interpretation: 0 being the present moment, $x \neq 0$ a future moment, then "$x \vdash \alpha$" reads "α being true at (the moment) x". It is worth noting that the assumption (Tr) guarantees that any proposition true at x is also true at every moment y future to x. That obviously means that in the framework elaborated, propositions are treated as temporally definitive units and, as such, they must not contain any occasional, time–depending expressions (such as e.g. "now", "today" etc.). A quick reflection upon the Example would lead one to the conclusion that Urquhart's interpretation is entirely compatible with the original interpretation envisaged by Post himself.

Apart from the basic matrices P_n, Post considered matrices with more designated elements. In turn, he also defined a family of functionally incomplete *n–valued implicative matrices* with k designated values $(1 \leq k < n)$:

$$P_{nk} = (\{t_1, t_2, \ldots, t_n\}, \to_{nk}, \{t_{n-k+1}, \ldots, t_n\}),$$

where

$$t_i \to_{nk} t_j = \begin{cases} t_n & \text{if } i \leq j \\ t_j & \text{if } i > j \text{ and } i \geq n-k+1 \\ t_{n-i+j} & \text{if } i > j \text{ and } i < n-k+1. \end{cases}$$

The matrices of that family can serve as a tool for description of the implication connectives of other known many–valued logics. So, for instance, $\to_{n1}$ and $\to_{n\,n-1}$ are n–valued of Łukasiewicz and Gödel implication respectively (to obtain the implicative Gödel matrix one ought to reduce the set of distinguished values to $\{t_n\}$, previously having built the truth–table for implication).

The fact that Post has designated many (at a time) logical values induced a significant impulse for a just emerging, in the 1920's, theory of logical matrices. It seems that other originators of many–valued logics ignored that possibility, or, did not attach great importance to it.

3. Axiomatization of functionally complete systems of n–valued logic

The original $(\neg, \vee)$ systems of Post's logic are not axiomatized so far. However, the problem of their axiomatizability has been for years a foregone matter; hence Słupecki (1939) has constructed the largest possible class of functionally complete finite logics and gave a general method of their axiomatization. From this it evidently follows that also Post logics are axiomatizable albeit the problem of providing axioms for their original version still remains open.

Słupecki matrix S_{nk} (n being a given natural number, $1 \le k \le n$) is of the form:

$$S_{nk} = (\{1, 2, \ldots, n\}, \rightarrow, R, S, \{1, 2, \ldots, k\}).$$

where $\rightarrow$ is a binary (implication), and R, S unary operations defined in the following way:

$$x \rightarrow y = \begin{cases} y & \text{if } 1 \le x \le k \\ 1 & \text{if } k < x \le n \end{cases}$$

$$R(x) = \begin{cases} x+1 & \text{if } 1 \le x \le n-1 \\ 1 & \text{if } x = n \end{cases}$$

$$S(x) = \begin{cases} 2 & \text{if } x = 1 \\ 1 & \text{if } x = 2 \\ x & \text{if } 3 \le x \le n \end{cases}$$

Functional completeness of each of these matrices is based on Picard (1935): R and S are two of the Picard's functions, in order to define the third, it suffices to put:

$$Hx = (x \rightarrow R(x \rightarrow x)) \rightarrow Sx \quad \text{for } k = 1, \text{ then } Hx = \begin{cases} 1 & \text{if } x = 2 \\ x & \text{if } x \ne 2 \end{cases}$$

$$Hx = R(x \rightarrow x) \rightarrow x \quad \text{for } k > 1, \text{ then } Hx = \begin{cases} 1 & \text{if } x = k \\ x & \text{if } x \ne k \end{cases}$$

Słupecki produced an effective proof of axiomatizability of every logic determined by the matrix S_{nk} (any pair (n, k) as above) giving a long list of axioms formulated in terms of implication and special one–argument connectives defined through the superpositions of R, S, and H. The chief line of approach here is to make capital of the stand character of implication, which can be classically axiomatized using MP (the Detachment Rule). Słupecki extends MP onto the whole language, taking the Łukasiewicz's formula: $((p \rightarrow q) \rightarrow r)) \rightarrow ((r \rightarrow p) \rightarrow (s \rightarrow p))$ as the only axiom for implication and provides an inductive, combinatorial completeness proof.

In the end the method of axiomatization in Rosser, Turquette (1952) should at least be mentioned here. Rosser and Turquette determine the conditions that make finitely many-valued propositional logic resemble more

the Classical Propositional Calculus, CPC. and hence simplified problem of their axiomatization. The heart of the method is the use special j operators, which play a role of identifiers of respective logical values. Since Post matrices are functionally complete all necessary connectives are definable, including the counterparts of all classical connectives and j as well. Consequently, all Post logics are axiomatizable in the Rosser and Turquette style[3].

4. Algebraic counterparts and a new formalization

The concept of Post algebra of order n ($n \ge 2$) was introduced by Rosenbloom (1942) who defined Post algebras by means of the rotation $\neg$, the disjunction $\vee$ and some auxiliary functions. Subsequently, it has undergone several modifications resulting both from theoretical and practical reasons (see e.g. Dwinger (1977)). A particular importance among them has the lattice–theoretical characterization (Epstein (1960)) fixing a creative direction for the studies (see 14.4). The equational definition of these algebras is due to Traczyk (1964): *Post algebra of order* n ($n \ge 2$) can be presented as an algebra having two binary $\cup, \cap$, n unary operations $-, D_1, \ldots, D_{n-1}$ and constants $e_0, \ldots, e_{n-1}$

$$\mathcal{L} = (L, \cup, \cap, -, D_1, \ldots, D_{n-1}, e_0, \ldots, e_{n-1}),$$

satisfying the conditions:

(1) $(L, \cup, \cap)$ is a distributive lattice with zero, $0 = e_0$, and unit, $1 = e_{n-1}$

(2) $-(x \cup y) = -x \cap -y, \quad --x = x$

(3) $e_i \cap e_j = e_i$ if $i \le j$

(4) $D_i(x \cup y) = D_i(x) \cup D_i(y), \quad D_i(x \cap y) = D_i(x) \cap D_i(y)$

(5) $D_i(x) \cup -D_i(x) = 1, \quad D_i(x) \cap -D_i(x) = 0$

(6) $D_i(x) \cap D_j(x) = D_i(x)$ if $i \le j$

(7) $D_i(-x) = -D_{n-i}(x)$

(8) $D_i(e_j) = 1$ for $i \le j, \quad D_i(e_j) = 0$ for $j < i$

[3] An overall idea of axiomatization given in Rosser and Turquette coincides with that by Supecki since the central is here also the use of a "classical implication" and its rule, MP, as base for the axiom system.

(9) $x = (D_1(x) \cap e_1) \cup (D_2 \cap e_2) \cup \ldots \cup (D_{n-1}(x) \cap e_{n-1})$.

It can be proved that the set $C(L) = \{D_i(x) : x \in L, i \in \{1, \ldots, n-1\}\}$ is closed under lattice operations and that the structure $(C(L,n), \cup, \cap, -1, 0)$ is a Boolean algebra. Apparently, each Post algebra of order 2 is a Boolean algebra as well.

The simplest Post algebra of order n is the structure based on the set of logical values $\{t_1, \ldots, t_n\}$ having the operations $x \cup y = max\{x, y\}$, $x \cap y = min\{x, y\}$ and

$$D_i(x) = \begin{cases} t_n & \text{if } x > t_i \\ t_1 & \text{if } x \leq t_i \end{cases}; \qquad -t_i = t_{n-i+1}.$$

To the end, every Post algebra of order n is isomorphic to some field of sets $P(U, n)$ (Wade (1945)).

At present the attention is generally focused on the formalization of Post logics basing on Rousseau algebras. Rousseau (1969) noticed that any Post algebra of order n is a pseudo–Boolean algebra (see Rasiowa (1974)). Consequently, he proposed a definition of Post algebra (of order n) which turned out to be exceedingly important from the point of view of applications (see e.g. Rasiowa (1977)). A new operation appearing in this definitional version is a binary operation of *relative pseudocomplement* $\rightarrow$, which on the set of constants $\{e_0, \ldots, e_{n-1}\}$ can be described as follows:

$$e_i \rightarrow e_j = \begin{cases} e_{n-1} & \text{when } i \leq j \\ e_j & \text{otherwise.} \end{cases}$$

The system of n–valued propositional calculus corresponding to the Rousseau algebras (given n) is determined in the language with connectives $\neg$, $\rightarrow$, $\vee$, $\wedge$, $\equiv$, D_1, …, D_{n-1}, e_0, …, e_{n-1} (for the sake of brevity algebraic symbols used here bear new meaning, e.g. e_i now refers to logical constants i.e. zero–argument connectives). Its axioms are the schemes of complete derivational axiom system of intuitionistic logic[4] and, for every $i = 1, \ldots, n-1$,

(P11) $D_i(\alpha \vee \beta) \equiv (D_i\alpha \vee D_i\beta)$

(P12) $D_i(\alpha \wedge \beta) \equiv (D_i\alpha \wedge D_i\beta)$

(P13) $D_i(\alpha \rightarrow \beta) \equiv ((D_1\alpha \rightarrow D_1\beta) \wedge (D_2\alpha \rightarrow D_2\beta) \wedge \ldots \wedge (D_i\alpha \rightarrow D_i\beta))$

(P14) $D_i(\neg\alpha) \equiv \neg D_1\alpha$

[4] See e.g. Rasiowa (1974). p. 264

(P15) $D_iD_j\alpha \equiv D_j\alpha$

(P16) D_ie_j when $i \leq j$ and $\neg D_ie_j$ when $j > j$

(P17) $\alpha \equiv (D_1\alpha \wedge e_1) \vee \ldots \vee (D_{n-1}\alpha \wedge e_{n-1})$

(P18) $D_1\alpha \vee \neg D_1\alpha$.

And, apart from MP, an extra inference rule is

$$(r_n) \qquad \frac{\alpha}{D_{n-1}\alpha}$$

The predicate calculi for Post logics are built in a standard way on the basis of propositional calculi. The most systematic studies of them carried out so far are due to Rasiowa (1974).

Bibliography

Dunn, J. M. and Epstein, G. (eds.) (1977). *Modern uses of multiple--valued logic*. D. Reidel, Dordrecht, Holland.

Dwinger, Ph. (1977). A survey of the theory of Post algebras and their generalizations [in:] Dunn and Epstein (eds.) (1977), 53–75.

Epstein, G. (1960). The lattice theory of Post algebras. *Transactions of the American Mathematical Society*, **95**, 300–317.

Łukasiewicz, J. (1920). O logice trójwartościowej. *Ruch Filozoficzny*, **5**, 170–171. English tr. On three–valued logic [in:] *Selected works* – see Łukasiewicz (1961), 87–88.

Łukasiewicz, J. (1929). *Elementy logiki matematycznej*. Skrypt. Warszawa (II ed. Warszawa 1958, PWN); English tr. *Elements of Mathematical Logic* translated by Wojtasiewicz, O. Oxford, Pergamon Press, 1963.

Picard, S. (1935). Sur les fonctions définies dans les ensembles finis quelconques. *Fundamenta Mathematicae*, **24**, 198–302.

Post, E. L. (1920). Introduction to a general theory of elementary propositions. *Bulletin of the American Mathematical Society*, **26**, 437.

Post, E. L. (1921). Introduction to a general theory of elementary propositions. *American Journal of Mathematics*, **43**, 163–185.

Rasiowa, H. (1974). *An algebraic approach to non–classical logics.* North–Holland, PWN, Amsterdam, Warszawa.

Rasiowa, H. (1977). Many–valued algorithmic logic as a tool to investigate programs [in:] Dunn and Epstein (eds.) (1977), 79–102.

Rosenbloom, P. C. (1942). Post algebra. I. Postulates and general theory. *American Journal of Mathematics*, **64**, 167–188.

Rosser, J. B. and Turquette, A. R. (1952). *Many–valued logics*. North-Holland, Amsterdam.

Rousseau, G. (1969). Logical systems with finitely many truth–values. *Bulletin de l'Académie Polonaise des Sciences, Série des sciences mathématiques, astronomiques et physiques*, **17**, 189–194.

Słupecki, J. (1939b). Dowód aksjomatyzowalności pełnych systemów wielowartościowych rachunku zdań (Proof of the axiomatizability of full many–valued systems of propositional calculus). *Comptes rendus de séances de la Société des Sciences et des Lettres de Varsovie Cl. III*, **32**, 110–128.

Traczyk, T. (1964). An equational definition of a class of Post algebras. *Bulletin de l'Académie Polonaise des Sciences Cl. III*, **12**, 147–149.

Urquhart, A. (1973). An interpretation of many–valued logic. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, **19**, 111–114.

Wade, C.I. (1945). Post algebras and rings. *Duke Mathematical Journal*, **12**, 389–395.

Whitehead, A. N. and Russell, B. (1910). *Principia Mathematica*, vol. **I**. Cambridge U.P.

Grzegorz Malinowski
Department of Logic
University of Łódź
e-mail: gregmal@krysia.uni.lodz.pl

STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2 (15)

Zenon Sadowski

ON THE DEVELOPMENT OF EMIL POST'S IDEAS IN STRUCTURAL COMPLEXITY THEORY

Abstract

Structural complexity theory is one of the branches of computer science. In this survey we will show how notions of a complete r.e. set and many-one reducibility from Post's 1944 paper influenced the study of NP-complete languages in structural complexity theory.

1. Introduction

Emil Post's 1944 address to the American Mathematical Society, "Recursively enumerable sets of positive integers and their decision problems" [2], had a great impact on later research in recursive function theory and theoretical computer science. The line of research which started with this paper culminated in the invention of **NP**-completeness, the central notion of structural complexity theory and the favorite paradigm of the whole field of computer science.

In Post's paper, in recursive function theory and structural complexity theory the concept of decision problem is the main subject of research. Connected with every decision problem are an infinite family of instances and a question. For any instance we ask this question and expect the answer YES or NO.

An example of decision problem is the Hamilton cycle problem (see [1]).
Instance: A graph $G = (V, E)$ with a finite set of nodes and a set E of edges, which are pairs of nodes.
Question: Is there a cycle that visits all nodes of the graph G once?

Every problem is fully determined by the set of its YES instances. Coding these YES instances by strings or by natural numbers we can treat problems as sets of words (formal languages) or as subsets of natural numbers. The first approach appears in Post's paper and in recursive function theory, the second one is typical for complexity theory.

2. Classification of recursively enumerable sets

In his paper Emil Post introduced the concept of reductions between problems, defined complete problems and started the classification of recursively enumerable sets. In this section we will consequently code problems, following the approach from Post's paper, as subsets of natural numbers.

A nonempty set of natural numbers is recursively enumerable provided there is an automatic method (a recursive function) for listing out its members, one after the other. This does not imply that there is a decision method for determining membership in the set. Sets possessing such a decision method are called recursive.

There is another equivalent definition of recursively enumerable sets. A set of natural numbers is recursively enumerable if and only if every number belonging to this set has a certificate witnessing its membership in the set. Numbers which are not elements of the set possess no such certificates.

Let A, B be two sets of natural numbers. The set A is reducible to the set B if there exists an effective method which determines for each natural number n a natural number m such that m is or is not in B according as n is or is not in A. Emil Post studied reductions between recursively enumerable sets. He considered five types of reducibility: 1) many-one reducibility, 2) one-one reducibility, 3) truth-tables reducibility, 4) bounded truth-tables reducibility and 5) Turing reducibility.

The notion of reducibility orders recursively enumerable sets. The maximal elements of this partial order Post named complete recursively enumerable sets. Every recursively enumerable set is reducible to a complete set. Post proved the following theorem:

Theorem 1. (Post)

There exist complete recursively enumerable sets with respect to one-one reducibility (many-one, truth-tables, bounded truth-tables, Turing reducibility).

Whether there exist recursively enumerable sets which are not recursive and which are not complete was the main question considered by Post in paper [2]. He proved that with respect to many-one, one-one, truth-tables,

and bounded truth-tables reducibility such sets exist. The problem whether there exists a recursively enumerable set which is not recursive and which is not complete with respect to Turing reducibility was solved over ten years later by R. Friedberg and independently by A. Muchnik.

3. Classification of NP-languages

By an alphabet we mean any finite set of symbols, as denoted by Σ. A finite sequence of symbols from Σ is named a word. By a formal language we mean any set of words over Σ. In this section decision problems are coded as formal languages (sets of codes of their YES-instances).

In structural complexity theory the basic models of computation are deterministic and nondeterministic Turing machines (see [1]). In the deterministic Turing machine model every move is completely defined by the current situation. The state of the machine and the symbol currently scanned by the tape head completely determine the next state and the move of the tape head. Nondeterministic Turing machines may have choices in selecting the next move. Such a machine accepts input x if and only if there is a sequence of choices of the allowable moves which lead from the starting configuration to an accepting state.

In structural complexity theory decision problems are classified as tractable and intractable. Tractable problems are those which are feasibly computable (practically solvable). A problem is generally considered to be tractable if it can be solved by an algorithm with a time-complexity function which is bounded by a polynomial. Since problems are encoded as formal languages, the class **P** consisting of all languages accepted by deterministic Turing machines in polynomial time may be considered as a formal equivalent of tractable problems.

The class of problems which can be solved in polynomial time using a nondeterministic procedure is the second class central to structural complexity theory. Problems from this class have one common property: the certificate property. In each case, if a given input of the problem is a YES instance, then there is a short argument (a succint certificate) which convinces us that the input is indeed a YES instance. Naturally NO instances possess no such certificates. Certificates are small which means that their length is bounded by a polynomial in the length of the input. The formal counterpart of these problems is the class **NP** consisting of all languages accepted by a nondeterministic Turing machine in polynomial time. At the moment we do not know whether **NP=P**, and this open question is the central problem in structural complexity theory.

The polynomial time analogue of many-one reducibility from Post's paper is the notion of polynomial time reducibility (Karp reducibility) intensively studied in complexity theory. Just as the family of recursively enumerable sets possesses complete sets with respect to many-one reducibility, **NP**-languages possess complete languages with respect to Karp reducibility.

NP-complete languages capture the essence and difficulty of the whole **NP** class. They enjoy the following property: one of **NP**-complete languages belongs to **P** only if they all do, only if the whole **NP** class comes down to **P**.

The most important **NP**-complete language was discovered by Stephen Cook.

Theorem 2. (Cook)
SAT, the set of satisfiable Boolean formulas, is **NP***-complete.*

The importance of **NP**-completeness for combinatorial optimization was revealed by Richard Karp. He proved that the following graph theoretic problems are **NP**-complete: HAMILTONIAN CYCLE, CLIQUE, VERTEX COVER, TSP – the travelling salesman problem (for exact definitions see [1]). His results opened the floodgate to proofs that hundreds of different problems are **NP**-complete. A proof that a language is **NP**-complete is now considered strong evidence that the problem encoded by this language is not feasibly solvable.

Post's problem was considered in polynomial setting by Richard Ladner. He finished the classification of **NP** languages by proving the following theorem:

Theorem 3. (Ladner)
If $\mathbf{P} \neq \mathbf{NP}$*, then there is a language in* **NP** *which is neither in* **P** *nor is it* **NP***-complete.*

4. Isomorphism Conjecture

It was proved by John Myhill that all many-one complete recursively enumerable sets are recursively isomorphic, i.e., are identical up to a recursive permutation of their elements. **NP**-complete languages correspond to many-one complete recursively enumerable sets and this analogy together with Myhill's result led to the Isomorphism Conjecture: all

NP-complete languages are isomorphic under polynomial time computable permutations (polynomial time bijections).This conjecture was supported by Leonard Berman and Juris Hartmanis' result that all known-up-to-now **NP**-complete languages are polynomially isomorphic.

One way of disproving the Isomorphism Conjecture would be by showing that there exist **NP**-complete languages with sufficiently different densities, because if two languages are polynomially isomorphic, then their densities are polynomially related.

We say that a language is sparse if the number of its words up to length n is bounded by a polynomial in n. A sparse language can not be polynomially isomorphic to SAT, since SAT is too dense to be mapped by a bijection onto a sparse set. Thus sparse **NP**-complete languages are natural candidates to refute the Isomorphism Conjecture.

The possibility of finding sparse **NP**-complete languages was resolved by Stephen Mahaney's result.

Theorem 4. (Mahaney)
There exist sparse **NP***-complete languages if and only if* $\mathbf{P}=\mathbf{NP}$.

It is worth mentioning that the first step in this direction was done by Polish mathematician Piotr Berman. He proved that there exist sparse **NP**-complete languages over a single letter alphabet if and only if $\mathbf{P}=\mathbf{NP}$.

References

[1] Aho A., Hopcroft J. and Ullman D. (1977) *The Design and Analysis of Computer Algorithms*, Addison-Wesley.

[2] Post E. (1944) *Recursively Enumerable Sets of Positive Integers and their Decision Problems*, Bulletin AMS, 50, pp. 284-316.

Zenon Sadowski
Institute of Mathematics
University of Białystok
e-mail: sadowski@math.uw.bialystok.pl

STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2 (15)

Wiktor Dańko, Jolanta Koszelew

THE POST CORRESPONDENCE PROBLEM AS A TOOL FOR PROVING UNDECIDABILITY IN LOGICS OF PROBABILISTIC PROGRAMS

Abstract

We present a simple proof of undecidability of the set of tautologies of Probabilistic Algorithmic Logic (cf. [1]). It is based on the well-known Post Correspondence Problem (e.g., cf. [6]).

1. Introduction

Logics for probabilistic programs, e.g., Probabilistic Dynamic Logic (cf. [4]), Probabilistic Algorithnic Logic (cf. [1]), are highly undecidable formal systems. To establish their degrees of undecidability we need some advanced techniques of recursion theory (cf. [4,8]). Nevertheless, the Post Correspondence Problem is sufficient to prove undecidability of such systems.

2. The Post Correspondence Problem

We recall basic notions related to the problem mentioned in the title of this section (cf. [6]).

Let $A = \{\varphi_1, \ldots, \varphi_q\}$ be a finite alphabet and let S, T be two sequences of strings (words) on A, say

$$S = \langle w_1, \ldots, w_s \rangle,$$

$$T = \langle v_1, \ldots, v_s \rangle.$$

We say that there exists a Post correspondence solution (PC-solution) for the pair (S,T) if there is a nonempty sequence of integers $\langle m_1,\ldots,m_r\rangle$, $m_1,\ldots,m_r \in \{1,\ldots,s\}$, such that the strings

$$w_{m_1}\ldots w_{m_r},\quad v_{m_1}\ldots v_{m_r}$$

are identical.

The Post correspondence problem is to devise an algorithm that will tell us for any (S,T) whether or not there exists a PC-solution.

2.1. Theorem (cf. [6])

The Post correspondence problem is undecidable.

3. Probabilistic Algorithmic Logic

We shall use an abstract programming language L_P being an extension of a first order language L, based on a countable set $V = \{x_0, x_1, x_2, \ldots\}$ of individual variables (denoted shortly by $x, y, z, \ldots$), a set $F = \{f, g, \ldots\}$ of function symbols, a set $R = \{r, s, \ldots\}$ of predicate symbols and a set $C = \{c, d, \ldots\}$ of constant symbols.

Let $\mathfrak{I} = \langle U, \underline{f}, \underline{g}, \ldots, \underline{r}, \underline{s}, \ldots, \underline{c}, \underline{d}, =, \ldots\rangle$ be a structure for L with the universe U. By a valuation of variables from the set $X_k = \{x_0, x_1, \ldots, x_{k-1}\}$ we shall understand any mapping $w{:}\, X_k \to U$. The set of all valuations of variables from X_k we shall denote by w_k. The fact, that a formula $\alpha(x_0,\ldots,x_{k-1})$ of L is satisfied at a valuation w will be denoted by $\mathfrak{I}, w \models \alpha$ (we shall write $\alpha(x_0,\ldots,x_{k-1})$ instead of α provided that all free variables of α are among $\{x_0,\ldots,x_{k-1}\}$).

Probabilistic programs of L_P are understood as iterative programs with typical program constructions *begin* ...; ... *end*, *if* ... *then* ... *else* ..., *while* ... *do* ... (tests are assumed to be open formulas of L) and with two probabilistic constructions: x:=?, *either* ... *or* ... interpreted as follows: the first corresponds to a random generation of a value of the variable x, each part of the second one is choosen with the probability 0,5. We shall denote by ρ the probability distribution corresponding to random generation of elements in the realization of assignments of the form x:=?. We can write in an informal way:

$$\rho{:}\, U \to [0,1],\quad \sum_{u\in U} \rho(u) = 1.$$

By a probabilistic structure for L_P we shall understand the pair $\langle \mathfrak{I}, \rho\rangle$ and denote by $\mathfrak{I}_\rho$.

We recall that a deterministic iterative program $K(x_0,\ldots,x_{k-1})$ is

interpreted in $\mathfrak{I}$ as a (partial) function transforming (input) valuations into (output) valuations (cf. [1,4]):

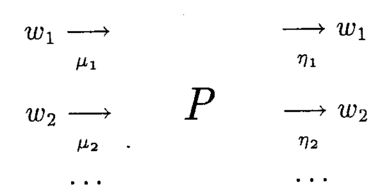
$$K_A{:}\, W_k \to W_k.$$

(Similarly, to the case of formulas, we shall write $K(x_0,\ldots,x_{k-1})$ instead of K, provided that all free variables of K are among $\{x_0,\ldots,x_{k-1}\}$; a formal definition of the set of free variables of a program is given in [7]).

In the case of probabilistic programs we shall assume that each possible valuation of program variables is assumed to appear with a probability. A probabilistic programs P is interpreted in the structure $\langle \mathfrak{I}, \rho\rangle$ as a (total) mapping transforming input (sub)distributions into output (sub)distributions (cf. [1,4]):

$$P_{\langle \mathfrak{I},\rho\rangle}{:}\, M \to M,$$

$P_{\langle \mathfrak{I},\rho\rangle}(\mu)$ will denote the output subdistribution realized by the program P in the structure $\langle \mathfrak{I}, \rho\rangle$ at the initial subdistribution μ. We can illustrate this situation as follows



For details we refer the reader to the paper [1]. An example illustrating the above introduced notions is given in Appendix (Example 6.2.). It remains to explain why we use the term "(sub)distribution" instead of "distribution":

3.1. Remark

Note, that in the case, where a program does not terminate any computation, then for the output subdistribution μ, $\mu(w) = 0$, for every valuation w.

The language L_p contains an additional set $V_{\mathfrak{R}}$ of variables ranging over the set $|\mathfrak{R}|$ of real numbers and the set of symbols $\{+,-,\times,/,<,0,1\}$ interpreted in the standard way in the ordered field $\mathfrak{R}$ of real numbers. These variables will be used to denote probabilities of transitions (in program computations) from an input valuations to an output valuation and/or probabilities that a formula is satisfied.

Thus L_p is a two sorted language with the set T of terms (defined in a standard way), the set $T_{\mathfrak{R}}$ of arithmetical terms defined as the least set satisfying

– $0, 1 \in T_{\mathfrak{R}}$ and each variable from $V_{\mathfrak{R}}$ belongs to $T_{\mathfrak{R}}$,

– if α is a formula of L then $P(\alpha)$ belongs to $T_{\mathfrak{R}}$,

– if t', t'' belong to $T_{\mathfrak{R}}$ then $t' + t'', t' - t'', t' \cdot t'', t'/t''$ belong to $T_{\mathfrak{R}}$,

The language L contains the set F_P of probabilistic algorithmic formulas which are used to express properties of probabilistic programs. F_P is defined as the least set of expressions satisfying:

– if K is a probabilistic program of L_P and A is a formula of F_P then KA belongs to F_P,

– if $A, B \in F_P$ then $\neg A, (A \wedge B), (A \vee B), (A \rightarrow B) \in F_P$,

– if $A \in F_P$ and $r \in V_{\mathfrak{R}}$ then $(\exists r)A, (\forall r)A \in F_P$.

The intuitive meaning of a formula of the form KA is the following:

> KA is satisfied in an initial state μ iff the program K end its computation at a final state μ' and the formula A is satisfied at the state μ'.

We omit here formal definitions of the interpretation of a language of Probabilistic Algorithmic (Dynamic) Logic and refer the reader to the papers [1,4]. In Section 4 we shall consider the case of finite interpretations, where the intuitive meaning of the interpretation is more simple.

4. Decidable and Undecidable Problems in Probabilistic Algorithmic Logic

The Probabilistic Dynamic Logic (PrDL) is the system for reasoning about probabilistic programs proposed by Feldman and Harell in [4]. This system differs from the sytem of the Probabilistic Algorithmic Logic:

– a language of PrDL contains disjoint set of variables for reals and integers;

– the results in [4] are presented under an additional assumption that the universe of interpretation is the set of real numbers or, more general, an arithmetical universe (cf. [4] for the definition of an arithmetical universe).

We would like to accent, that in the case of Probabilistic Algorithmic Logic, we do not use variables for integers, values of real variables are not changed during program computations, and the field of reals is an external tool for probabilistic estimation of behaviours of algorithms.

4.1. Remark (cf. [4])

In [4], Feldman and Harel have mentioned that the set of tautologies of the Probabilistic Dynamic Logic is highly undecidable (for information about the position of this set in the Kleene-Mostowski hierarchy, we refer the reader to [4,8]). In [4] Feldman and Harell present an axiom system for Probabilistic Dynamic Logic that is complete relative to an extension of first order analysis (second order arithmetic is definable in first order analysis, cf. [8]). Therefore the Probabilistic Dynamic Logic does not admit a complete axiomatization in the classical sense.

In the paper [4] a question is formulated:

> when the first-order analysis, without integer variables is sufficient to describe properties of probabilistic algorithms?

We answer this question in [1]:

4.2. Theorem (cf. [1])

The set of formulas of Probabilistic Algorithmic Logic, valid in a fixed finite structure $\mathfrak{I}$ is decidable with respect to the diagram of the structure $\mathfrak{I}$.

By a diagram of a structure $\mathfrak{I}$ for L we understand the set of all atomic formulas valid in $\mathfrak{I}$ (for each element of $U = |\mathfrak{I}|$ there is a constant symbol in L). The proof consists in reducing the validity of a formula α of L_P in $\mathfrak{I}$ to the validity of a corresponding formula α' in the ordered field of reals $\mathfrak{R}$ (cf. [1]). Some facts related to the proof of this theorem are given in Appendix.

Moreover, this theorem enables us to construct an effective axiom system sufficient to prove all valid sentences in a given finite structure $\mathfrak{I}$, provided that the diagram of $\mathfrak{I}$ is taken as the set of specific axioms.

Now, we shall demonstrate that the proof of undecidability of the set of tautologies of the Probabilistic Algorithmic Logic does not require advanced results of the Recursion Theory and can be based on the Post Correspondence Problem.

Let $S = \langle w_1, \ldots, w_s \rangle$, $T = \langle v_1, \ldots, v_s \rangle$ be two sequences of strings on $A = \{\varphi_1, \ldots, \varphi_q\}$. We shall treat the symbols of A as one-argument function symbols of a language L_P. Denote by $K_{(S,T)}$ the following program of L_P:

```
begin
  y:=x; z:=x;
  repeat
    either
      begin
        y:=w_1(y); z:=v_1(z);
      end
    or either
      begin
        y:=w_2(y); z:=v_2(z);
      end
     or either
        ⋮
           or either
             begin
               y:=w_{s-1}(y); z:=v_{s-1}(z);
             end
            or
             begin
               y:=w_s(y); z:=v_s(z);
             end;
  until y = z;
end.
```

Note, that the program construction *repeat ... until ...*, used in this program can be easily eliminated by the *while ... do ...* construction and therefore this program can be viewed as a program of L_P. The subprogram of $K_{(S,T)}$ contained between *repeat* and *until* we shall denote by K'.

The following facts reduce the halting problem for the program $K_{(S,T)}$ to the Post Correspondence Problem; the fact that the program $K_{(S,T)}$ halts with a positive probability can be expressed by the formula of L_P of the form $P(K_{(S,T)}(\gamma \vee \neg\gamma)) > 0$, where γ is an open formula of L.

(1) Suppose, that the sequence $\langle m_1, \ldots, m_r \rangle$, $m_1, \ldots, m_r \in \{1, \ldots, s\}$, is a PC-solution for the pair (S,T), $S = \langle w_1, \ldots, w_n \rangle$, $T = \langle v_1, \ldots, v_n \rangle$.

Let μ be a distribution such that a value u of the variable x appears with a positive probability,

From the definition of interpretation of the *either ... or ...* construction it follows that after r repetitions of the subprogram K' the probability of realization of the following sequence of assignments is positive:

$$y{:=}w_{m_1}(y);\ \ z{:=}v_{m_1}(z);$$
$$y{:=}w_{m_2}(y);\ \ z{:=}v_{m_2}(z);$$
$$\ldots$$
$$y{:=}w_{m_r}(y);\ \ z{:=}v_{m_r}(z);$$

Note that the at the begining of the realization of the instruction construction *repeat ... until ...* the initial values of the variables y, z are identical (the common value is the initial value of x). Since $\langle m_1, \ldots, m_r \rangle$ is a PC-solution for (S,T), then the strings

$$w_{m_1} \ldots w_{m_r}, \quad v_{m_1} \ldots v_{m_r}$$

are identical. Thus the values of the terms

$$w_{m_1} \ldots w_{m_r}(y), \quad v_{m_1} \ldots v_{m_r}(z)$$

are identical which means, that the probability that $K_{(S,T)}$ end its computation is positive.

(2) Now, suppose that for the pair (S,T), $S = \langle w_1, \ldots, w_n \rangle$, $T = \langle v_1, \ldots, v_n \rangle$, there is no PC-solution, i.e., for each sequence $\langle m_1, \ldots, m_r \rangle$, $m_1, \ldots, m_r \in \{1, \ldots, s\}$, the strings

$$w_{m_1} \ldots w_{m_r}, \quad v_{m_1} \ldots v_{m_r}$$

are different.

Let us consider a Herbrand interpretation $\mathfrak{I}$ of L_P such that its universe is A^* (i.e., consists of all strings on A) and for every function symbol φ of $A = \{\varphi_1, \ldots, \varphi_s\}$, the interpretation $\varphi\colon A^* \to A^*$ is defined as follows:

$$\varphi_{\mathfrak{I}}(w) \text{ is the string } \varphi w \text{ of } A^*.$$

It is easy to note, that for each initial value v of x, for each number r of repetitions of the subprogram K' during realization of $K_{(S,T)}$, and for the sequensce of assignments corresponding to these repetitions

$$y{:=}w_{m_1}(y);\ \ z{:=}v_{m_1}(z);$$
$$y{:=}w_{m_2}(y);\ \ z{:=}v_{m_2}(z);$$
$$\ldots$$
$$y{:=}w_{m_r}(y);\ \ z{:=}v_{m_r}(z);$$

the values of y, z after r repetitions of K' are the strings

$$w_{m_1} \ldots w_{m_r} v, \quad v_{m_1} \ldots v_{m_r} v$$

respectively. These strings are different, since the strings $w_{m_1} \dots w_{m_r}$, $v_{m_1} \dots v_{m_r}$ are different.

This means that the halting formula of the *repeat* ... *until* ... instruction of $K_{(S,T)}$ cannot be satisfied.

From the points 1, 2 it follows the equivalence:

4.4. Corollary

Let S, T be two sequences of strings on A, and let $K_{(S,T)}$ has the meaning as in the above. The halting formula of the program $K_{(S,T)}$ can be satisfied in an interpretation $\mathfrak{I}$ if and only if the Post Correspondence Problem has a PC-solution for (S, T)

Thus the undecidability of the Post Correspondence Problem induces the undecidability of validity of halting formulas of the Probabilistic Algorithmic Logic and therefore proves the following theorem:

4.5. Theorem

The set of tautologies of the Probabilistic Algorithmic Logic is undecidable.

We end this Section by a remark:

4.6. Remark

The Post Correspondence Problem can be used to prove undecidability of the set of tautologies in logics of nondeterministic programs (like the Algorithmic Logic of Nondeterministic Programs (cf. [8]).

5. References

[1] Dańko W., *The set of probabilistic algorithmic formulas valid in a finite structure is decidable with respect to its diagram*, Fundamenta Informaticae, vol. 19, 3-4, 1993, (417-431)

[2] Dańko W., Koszelew J., *Properties of Probabilistic Algorithms Provable in First-Order Analysis*, submitted to Fundamenta Informaticae,

[3] Dańko W., *A Criterion of Undecidability of Algorithmic Theories*, Fundamenta Informaticae vol. IV.3, 1981

[4] Feldman Y.A., Harel D., *A Probabilistic Dynamic Logic*, ACM Journal of Comp., 1982 (181-195),

[5] Feller W., *An Introduction to Probability Theory*, John Willey & Sons, Inc., New York, London, 1961,

[6] Linz P., *An Introduction to Formal Languages and Automata*, D.C.Heath and Co., Lexington Massachussets, Toronto 1990,

[7] Mirkowska G., Salwicki A., *Algorithmic Logic*, D. Reidel Publ. Co. & PWN Warsaw, 1987,

[8] Rogers H.J., *Theory of Recursive Functions and Effective Computability*, McGraw-Hill Book Co., New York Toronto, London Sydney, 1967

[9] Skowron A., *Data Filtration: A Rough Set Approach*, in: W. Ziarko (ed.), Roug Sets, Fuzzy Sets and Knowledge discovery, Workshop in Computing, Springer-Verlag & British Computer Society, London, Berlin, 1994, 108-118,

[10] Stapp, L., *The normal form theorem for nondeterministic programs*, University of Warsaw, 1979, unpublished manuscript,

6. Appendix

We recall here some facts related to the proof of the theorem 4.2. In particular we give a lemma proved in [1] on that the proof of Theorem 4.2 depends in an essential manner. For details we refer the reader to the paper [1].

Let $U = \{u_1, u_2, \dots, u_m\}$ and let us denote by ρ the probability distribution corresponding to the realization of random assignments of the form v:=?:

$$\rho: A \to [0, {}^*1], \quad \rho(a_0) + \rho(a_1) + \dots + \rho(a_m) = 1.$$

Let $P(x_0, \rho, x_m)$ be a program. Since $U = \{u_1, \dots, u_n\}$ and $X_k = \{x_0, x_1, \dots, x_{k-1}\}$, then W_k consists of $N = m^n$ elements, $W_m = \{w_1, w_2, \dots, w_N\}$. Thus each distribution μ can be viewed as a N-element vector $\mu = [\mu_1, \mu_2, \dots, \mu_N]$, where $\mu_i = \mu(w_i)$, $i = 1, \dots, N$. Denote by M the set of all such vectors of distributions. Then

$$P_{\langle \mathfrak{I}, \rho \rangle}: M \to M.$$

To avoid any misunderstanding, we shall write $[(w_1, \mu_1,), \dots, (w_N, \mu_N)]$ instead of $[\mu_1, \mu_2, \dots, \mu_N]$ if the ordering among the valuations of W_m is not obvious.

The following lemma shows that each program P can be interpreted as a linear operator $\mathbf{P}$ on the space M of vectors of distributions:

6.1. Lemma (cf. [1])

Let $\langle \mathfrak{J}, \rho \rangle$ be a structure for L_P with the universe $U = \{u_1, \ldots, u_n\}$. For every program $P(x_0, \ldots, x_m)$ we can construct, in an effective way, a $N \times N$ matrix $\mathsf{P} = [p_{ij}]_{i,j=1,\ldots,N}$ such that for every vectors $\mu = [\mu_1, \ldots, \mu_N]$, $\eta = [\eta_1, \ldots, \eta_N] = P_{\langle \mathfrak{J}, \rho \rangle}(\mu)$, the following holds:

$$\eta = \mu \cdot \mathsf{P}$$

($\cdot$ denotes here the product operation for matrices).

Moreover, an element p_{ij} of P describes the probability that w_j is the output valuation after computation of P, provided that the valuation w_i appears as the input valuation with the probability 1.

We illustrate the Lemma as follows:

$$[\mu_1 \quad \ldots \quad \mu_N] \cdot \begin{bmatrix} p_{11} & \ldots & p_{1N} \\ \ldots & \ldots & \ldots \\ p_{N1} & \ldots & p_{NN} \end{bmatrix} = [\eta_1 \quad \ldots \quad \eta_N]$$

where $\mu = [\mu_1, \ldots, \mu_N]$, $\eta = [\eta_1, \ldots, \eta_N] = P_{\langle \mathfrak{J} \rho \rangle}(\mu)$, $\mathsf{P} = [p_{ij}]_{i,j=1,\ldots,N}$

6.2. Example

Let K denote the program

while $x = b$ *do*
either x:=? *or* x:=a

and denote by K' the subprogram *either* x:=? *or* x:=a.

Assume that the universe U of the interpretation $\mathfrak{J}$ consists of two elements, $U = \{\underline{a}, \underline{b}\}$. Note, that in this case the above defined set W_1 (of valuations of the variable x consists of two valuations) w_1, w_2 satisfying:

$$w_1(x) = \underline{a}, \quad w_2(x) = \underline{b}.$$

Moreover, assume that the random generator distribution ρ is defined as follows:

$$\rho(\underline{a}) = p, \quad \rho(\underline{b}) = 1 - p$$

and the initial distribution μ satisfies

$$\mu(w_1) = q, \quad \mu(w_2) = r.$$

One can check that the output distribution μ' for the program K' at the initial distribution μ satisfies:

$$q' = \mu'(w_1) = 0.5 \cdot (1+p) \cdot q + 0.5 \cdot (1+p) \cdot r,$$
$$r' = \mu'(w_2) = 0.5 \cdot (1+p) \cdot q + 0.5 \cdot (1+p) \cdot r.$$

This may be written in the form

$$[q', r'] = [q, r] \cdot \begin{bmatrix} 0.5 \cdot (1+p) & 0.5 \cdot (1-p) \\ 0.5 \cdot (1+p) & 0.5 \cdot (1-p) \end{bmatrix}$$

that shows the transition matrix K' for the subprogram K'.

Using the method proposed in [1] we can check that the matrix K for the program K is $\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$.

Wiktor Dańko
Department of Computer Science
Technical University of Białystok
e-mail: danko@ii.pb.bialystok.pl

Jolanta Koszelew
Institute of Computer Science
Technical University of Białystok
e-mail: jolka@ii.pb.bialystok.pl

Jolanta Koszelew

SOME METHODS FOR QUALIFYING PROPERTIES OF PROBABILISTIC PROGRAMS

Key words. probabilistic programs, Markov chain, program termination, random variable, modelling stochastic processes

In this work, some methods and tools for a quantitative analysis of iterative probabilistic programs are proposed. The need for analyse of probabilistic programs arises in two main situations. The first is when we analyse a probabilistic program whose nondeterministic constructions are realised with known probability distribution, and we wish to infer some statistical property of the program, such as its average running time, the expected value of some output state, the probability of program termination, etc. Another situation is that of a probabilistic programs where we know a set of all possible distributions for realisation of nondeterministic constructions. For that case, the method of determining average probability of program termination in a fixed state is presented.

First situation: *Probabilistic Programs* are understood as iterative programs with two probabilistic constructions: x:=?, *either* ... *or* ... interpreted as follows: the first corresponds to a random generation of a value of the variable x, each part of the second one is choosen with the probability 1/2. Each possible valuation of program variables appears with known probability. Thus one can define distributions of probabilities on sets of input and output valuations and therefore *programs are interpreted as mappings transforming input distributions into output distributions.* In the case of probabilistic programs interpreted in a finite universe, there is an *effective algebraic method* (cf. [2]) for determining probabilities of transitions from an initial distribution to a to a final one.

The similarities between such programs and finite – state Markov chains enable to adopt the results for searching properties of probabilistic algorithms. On the other hand, the composiotin (decomposition) problem for the probabilistic programs can not be solved using Markov chains theory.

Contrary to Markov chains, probabilistic algorithms preserve an internal structure of stochastic processes which are modelled. For example: M is a program which simulates cooperation of two alarms: x and y. Each of them can work in a "good" state (denote by 1) or "bad" state (0).

```
M: begin
      while (x = 1) and (y = 1) do
        {M':} begin x:=?; y:=?; end;
    end;
```

If we treat such program as Markov chain we can analyse only one matrix $\mathcal{M}$ corresponding to whole program M. We can not determine a submatrix $\mathcal{M}'$ which is corresponding to a sub-program M'. The original method for composition (decomposition) of probabilistic programs is proposed.

Second situation: The exact distribution for nondeterministic construction of the form: x:=? is unknown but an information about the set Z of all admissible probability distributions is given.

Moreover, we known that each distribution $\rho_k \in Z$ appears with a probability f_k. Then we can determine *average case matrix*, which describes an average probabilities of transition from state to state.

Let us consider the same program M. Let us assume, that the set Z is the same for both devices and is defined as follows:

$$Z = \{\rho_k\colon \rho_k(1) = q_k \quad \rho_k(1) = 1 - q_k \text{ where } q_k \in \langle 0.7, 1\rangle\}.$$

Proposed method unable us to determine an *average probability* of event that alarm x and alarm y will be in "bad" state simultaneously ($x = 0$ *and* $y = 0$) for a given probability distribution f_k (for all elements of a set Z). If f_k is the following:

$$f_k = \begin{cases} 0 & \text{for } q_k \in \langle 0, 0.7) \\ 1 & \text{for } q_k \in \langle 0.7, 1\rangle \end{cases}$$

then the exact result is: 0.02504 and its practical interpretation is: An average risk, that this control system is deceptive is equal to 2,5% under condition that a probability of "bad" state for alarm x and y (independent) is not greater then 0.3.

A prototype of computer system for automatically analysis of statistical properties of probabilistic programs is prepared and implemented. The above problems: composition (decomposition) and "average case" of probabilistic programs arise during a work on this system.

References

[1] Bhararucha A. T., *Elements of the Theory of Markov Processes and Their Applications*, McGraw-Hill, New York 1960.

[2] Dańko W., *The set of probabilistic algorithmic formulas valid in a finite structure is decidable with respect to its diagram*, Fundamenta Informaticae, vol. 19, 3-4, 1993, (417-431)

[3] Koszelew J., *Analiza porównawcza algorytmów probabilistycznych i łancuchów Markowa*, Symulacja w badaniach i rozwoju, zbiór referatów Trzecich Warsztatów Naukowych PTSK Wigry, 26-28 września 1996 r.

Jolanta Koszelew
Institute of Computer Science
Technical University of Białystok
e-mail: jolka@ii.pb.bialystok.pl

Piotr Wojtylak

AXIOMATIZABILITY OF LOGICAL MATRICES

1. The completeness theorem for classical propositional logic was proved by Emil Post [1921]. Let $\mathfrak{M}_2$ be the classical two-valued matrix and let $E(\mathfrak{M}_2)$ denote the set of all its tautologies, that is propositional formulae valid in the matrix. The completeness theorem

$$E(\mathfrak{M}_2) = Cn(\{r_0, r_*\}, A_2)$$

says that any $\mathfrak{M}_2$ tautology can be derived by use of two inferential rules: substitution for propositional variables r_* and detachment (modus ponens) rule r_0 from a finite set A_2 of axioms (which are tautologies, too). This form of axiomatization (substitution, detachment and a finite set of axioms) became then a standard in propositional logic. Post referred to axioms given earlier by G. Frege and showed the completeness theorem using (conjunctive) normal forms of formulae. The completeness theorem, with the same argument, was then rediscovered by several logicians who were unaware of the earlier paper by E. Post. It shows the significance of the result and probably means that normal forms provide the most natural way of proving the theorem. Others, more interesting, methods of proving (for example by the construction of Lindenbaum algebra) were provided much later on. The completeness theorem is maintained as one of the most fundamental results in logic and is included in probably all elementary courses in logic. The possibility of elementary proving all classically valid formulae fascinates and attracts our attention. Nevertheless, let me make the following two general remarks.

a) We still seem to believe that any concept in science is accompanied with a finite set of natural axioms characterizing it. We seem to forget that the axiomatizabilty method, the ingenious discovery of ancient Greeks, shows only certain weaknesses of human nature and is not occurring immanently in the nature. The development of logic did not confirm the priority of one axiom system (for propositional logic) over others. Axioms may differ very roughly and I would even say that each logician has its own preferred axiom system for classical propositional logic. The choice of the method of proving the completeness theorem decides mostly on the choice of the set of axioms. Certainly, not any set equivalent (with respect to the substitution and detachment rule) to A_2 may be widely accepted as a set of axioms. For instance, the proposal of J. Łukasiewicz with a single axiom

$$((((p \to q) \to (\neg u \to \neg s)) \to u) \to t) \to ((t \to p) \to (s \to p)),$$

which is known to be the shortest formula with this property, has not been widely accepted. Any axiom system for classical propositional logic should be intuitively clear and should consist of relatively simple formulae. Despite some attempts, no formal general conditions obeyed by any acceptable set of axioms was given. In this situation it seems to be proper to say about axiomatizabilty of the matrix $\mathfrak{M}_2$ rather than the completeness theorem.

b) In case of propositional logic there raises a more general question if any axiom system is really needed. What are the adventages of the axiomatizability method there. This question disappears in predicate logic where the axiomatizability method is in fact the only possibility to introduce all classically valid laws. To support the significance of the axiomatizability method in propositional logic one could refer today to results in complexity theory. But this kind of argument could not be used in Post times though the completeness theorem was valued. Probably, the most important fact as concerns the result in questions is the reduction of possibly infinite the set of classical tautologies to its finite subset.

2. As soon as non-classical systems appeared in logic, the problem of their axiomatizability was posed immediately. It was natural as much as the first systems of non-classical logics were introduced by logical matrices. There were the so-called systems of Post and Łukasiewicz. Let me say that the main difference between these two groups of logics was the existence of a kind of philosophical motivation for Łukasiewicz logics whereas Post systems were motivated only formally. This philosophical motivation, despite the fact if it was correct or not, attracted an interest. Let me focus on Łukasiewicz logics. This is not due to the fact that any formal result confirms the philosophical significance of the systems. Contrary, it seems to me that it does not. But Łukasiewicz logics provide a nice illustration for the discussed in this lecture question of axiomatizability. So, similarly as for Post systems, the motivation for Łukasiewicz logics is only formal.

a) The problem of axiomatizability of Łukasiewicz logics was by no means one of the most important questions in the so-called Lwow-Warsaw school of logic in the twenties and thirties. The most important results

in this field were achieved by M. Wajsberg. In particular, he provided an axiomatization for the three-valued logic,

$$E(\mathfrak{M}_3) = Cn(\{r_0, r_*\}, A_3).$$

This axiomatization was widely accepted and was often referred to. Wajsberg also confirmed Łukasiewicz conjecture and showed the completeness theorem for the infinite valued logic

$$E(\mathfrak{M}_\infty) = Cn(\{r_0, r_*\}, A_\infty).$$

Let me recall that the set A_∞ consists of the formulae

$$p \to (q \to p)) \quad , \quad (p \to q) \to ((q \to s) \to (p \to s)),$$
$$((p \to q) \to q) \to ((q \to p) \to p) \quad , \quad (\neg p \to \neg q) \to (q \to p).$$

In Łukasiewicz conjecture there also occurred $((p \to q) \to (q \to p)) \to (q \to p)$ which was showed to be dependent by D. Meredith. Unfortunately, Wajsberg's proof has never been published and first (non-elementary) proof of this theorem was given by C. C. Chang [1955]. An elementary proof of this theorem may be found in R. Cignoli, D. Mundici [1997].

b) For remaining Łukasiewicz logics no sets of axioms was regarded in the thirties. Wajsberg [1935] showed a general theorem from which finite axiomatizability of any n-valued Łukasiewicz logic immediately follows.

If $\mathfrak{M}$ is a finite and normal (ie the detachment rule is normal) matrix and $A_w \subseteq E(\mathfrak{M})$, then $E(\mathfrak{M})$ finitely axiomatizable by use of substitution and detachment rule. The set A_w consists of

$$(p \to q) \to ((q \to s) \to (p \to s)) \quad , \quad (q \to s) \to ((p \to q) \to (p \to s)),$$
$$(p \to p) \to (q \to q)) \quad , \quad (p \to q) \to (\neg q \to \neg p) \quad , \quad \neg q \to ((p \to q) \to \neg p)).$$

The proof of this theorem contains an algorithm for searching of a finite set of axioms for any given finite matrix. But the algorithm is very difficult and useless for any practical reason. Apparently Wajsberg was unable to reduce the received, in the case of n-valued Łukasiewicz logics, sets of axioms to an acceptable form. Some solution of this problem can be found in W. A. Pogorzelski, P. Wojtylak [1993] and R. Tuziak [1988]. There is shown that

$$E(\mathfrak{M}_n) = Cn(\{r_0, r_*\}, A_n)$$

where A_n results from A_∞ by addoing formulae of the form (for appropriate k's)

$$(p \to^n q) \to (p \to^{n-1} p) \quad , \quad p \equiv (p \to^k \neg p) \to^{n-1} q.$$

3. Results on axiomatizability of logical matrices achieved in the thirties though deep and sophisticated showed certain difficulties. Logicians could not overcome problems raised by additional informal restrictions on intuitive meaning of axioms and formal restrictions to axiomatizations based on two inferential rules of inference: substitution and detachment. The further progress in the area was mostly due to the change of the point of view on logical systems. Logic was ceased to be seen as a set of logical laws (tautologies) and more attention was given to its inferential rules. Logical systems were described in terms of consequence operations where not only the set of tautologies plays a role but all theories over the logic are important. If one takes into account logical rules other than substitution and detachment, then any restriction to detachment-substitutionary axiomatizations should be neglected. The change of the point of view, together with additional concepts such as matrix consequence operation and structural completeness, creates a number of new problems and benefits with a number of interesting results. It also made the theory of axiomatizability of logical matrices more attractive and interesting.

a) In the theory of logical matrices, the change of the point of view realized by use of the concept of the matrix consequence operation; J. Łoś, R. Suszko [1958]. Let $\mathfrak{M} = \langle \mathcal{A}, D \rangle$ be a logical matrix in a propositional language S. The operation $\overrightarrow{\mathfrak{M}} : 2^S \to 2^S$ is defined as follows

$$\alpha \in \overrightarrow{\mathfrak{M}}(X) \equiv \forall_{v:At\to A}\ [h^v(X) \subseteq D \to h^v(\alpha) \subseteq D]$$

One easily shows that $\overrightarrow{\mathfrak{M}}$ is a structural consequence operation. It means that for any sets X, Y of formulae and any substitution e

$$X \subseteq \overrightarrow{\mathfrak{M}}(X)$$
$$X \subseteq Y \to \overrightarrow{\mathfrak{M}}(X) \subseteq \overrightarrow{\mathfrak{M}}(Y)$$
$$\overrightarrow{\mathfrak{M}}(\overrightarrow{\mathfrak{M}}(X)) \subseteq \overrightarrow{\mathfrak{M}}(X)$$
$$h^e(\overrightarrow{\mathfrak{M}}(X)) \subseteq \overrightarrow{\mathfrak{M}}(h^e X)$$

There immediately raises the question of axiomatizability of matrix consequences determined by certain known logical matrices, in particular those determined by Łukasiewicz matrices. It is known as the problem of strong axiomatization and it relies on searching for a set R of schematically defined rules and axioms A, such that for each set X of formulae

$$\overrightarrow{\mathfrak{M}}(X) = Cn(R, Sb(A) \cup X).$$

Schematically defined rules are rules defined by one of their sequents. So, schematically define rule is any inferential rules consisting of all (and only) instances of this basic sequent. The detachment rule is schematically defined and determined by its sequent $\langle \{p, p \to q\}, q \rangle$. The simplification rule is also schematically defined

$$\frac{\alpha}{\beta \to \alpha}$$

but the substitution rule is not. Let $Sb(A)$ be the set of all substitution instances of formulae in A. There is a clear coincidence between the schematically defined rules and axiom schemata. Axiom schemata may be seen as a special case of inferential rules in which the sets of premises is empty. Since for schematically defined (structural) rules $Cn(R \cup \{r_*\}, A) = Cn(R, Sb(A))$, the strong axiomatizability of a matrix implies its usual (weak) axiomatizability by use of the same rules together with the substitution rule.

b) Let me illustrate the concept of strong axiomatizability by referring to results on infinite valued Łukasiewcz logic and related matrices. I shall refer to results included in R. Wójcicki [1973] and P. Wojtylak [1978]. Clearly,

$$Cn(\{r_0\}, Sb(A_\infty)) = E(\mathfrak{M}_\infty) = \bigcap_{n=2}^{\infty} E(\mathfrak{M}_n) = E(\mathbf{P}_{n=2}^{\infty}\mathfrak{M}_n) = E(\mathfrak{L}_\infty)$$

where $\mathbf{P}_{n=2}^{\infty}\mathfrak{M}_n$ is the product of the involved matrices and $\mathfrak{L}_\infty$ is the Lindenbaum matrix for the infinite valued Łukasiewicz logic. For any set X of formulae the following inclusion hold and for certain sets they are proper

$$Cn(\{r_0\}, Sb(A_\infty) \cup X) \subseteq \overrightarrow{\mathfrak{M}_\infty}(X) \subseteq \bigcap_{n=2}^{\infty} \overrightarrow{\mathfrak{M}_n}(X) \subseteq \overrightarrow{\mathbf{P}_{n=2}^{\infty}\mathfrak{M}_n}(X) \subseteq \overrightarrow{\mathfrak{L}_\infty}(X)$$

So, the problem of strong axiomatizability of infinite valued Łukasiewicz logic becomes more complicated as it is not quite clear which matrix should be axiomatized and what is a matrix strongly adequate for the propositional system $\langle \{r_0\}, Sb(A_\infty) \rangle$. Moreover, one can easily extend the above sequence by adding new matrix consequences with the same set of tautologies; for instance the submatrix of $\mathfrak{M}_\infty$ determined by rational numbers. It is only known that the matrix consequence determined by the Lindenbaum matrix $\mathfrak{L}_\infty$ is structurally complete (cf. W. A Pogorzelski [1971]) and hence it is the greatest structural consequence with a given set of tautologies.

c) The view that any of the above consequences represents infinite valued Łukasiewicz logic could be questioned as the operations (except of the first one) are not finite. If one considers only finite sets X, then some of the inclusions can be replaced with equalities:

$$Cn(\{r_0\}, Sb(A_\infty) \cup X) \subseteq \overrightarrow{\mathfrak{M}_\infty}(X) = \bigcap_{n=2}^{\infty} \overrightarrow{\mathfrak{M}_n}(X) \subseteq \overrightarrow{\mathbf{P}_{n=2}^{\infty}\mathfrak{M}_n}(X) \subseteq \overrightarrow{\mathfrak{L}_\infty}(X).$$

In result, the matrix consequence operations should be axiomatized by use of inferential rules with infinite sets of premises and the problem is that there is no proper logical theory for such rules. The difference between finitary and infinitary rules are not so fundamental as one could expect. Let us consider, for instance, the substitution version of the above consequence operations. Then we get

$$Cn(\{r_0, r_*\}, A_\infty \cup X) \subseteq \overrightarrow{\mathfrak{M}_\infty}(Sb(X)) = \ldots = \overrightarrow{\mathbf{P}_{n=2}^{\infty}\mathfrak{M}_n}(Sb(X)) \subseteq \overrightarrow{\mathfrak{L}_\infty}(Sb(X))$$

The last, strongest consequence operation (the structural complete extension of the infinite valued logic) is finite and contains exactly three theories: $E(\mathfrak{M}_\infty)$, $E(\mathfrak{M}_2)$ and S. So, all finite valued logics (except of the classical one) can be rejected on the ground of the infinite valued one. The admittance of the substitution rule corresponds to the admittance of all instances of a given set of premises and is acceptable form a finitistic point of view.

d) Let us note that the above relations between consequence operations will change if we change (extend or restrict) the propositional language. The above inclusions concern the usual logical language with the usual logical operations including implications and negation. If we restrict to the positive fragment (that is the fragment without negation), then we get for any finite set X of formulae

$$Cn(\{r_0\}, Sb(A^p_\infty) \cup X) = \overrightarrow{\mathfrak{M}^p_\infty}(X) = \bigcap_{n=2}^{\infty} \overrightarrow{\mathfrak{M}^p_n}(X) = \overrightarrow{\mathbf{P}_{n=2}^{\infty}\mathfrak{M}^p_n}(X) = \overrightarrow{\mathfrak{L}^p_\infty}(X).$$

So, the logic $\langle\{r_0\}, Sb(A^p_\infty)\rangle$ is structurally complete for rules with finite sets of premises (but is not for arbitrary ones). If one considers, the extensions of the n-valued (for $n > 2$) logic $\langle\{r_0\}, Sb(A_n)\rangle$, then the situation changes a bit. The logic $\langle\{r_0\}, Sb(A_n)\rangle$ is strongly adequate for $\overrightarrow{\mathfrak{M}_n}$ but is not structurally complete. Its structural complete extension is $\overrightarrow{\mathfrak{M}_n \times \mathfrak{M}_2}$ and is received by extending $\langle\{r_0\}, Sb(A_n)\rangle$ with the rule

$$\frac{\alpha \vee \neg\alpha \rightarrow^{n+1} \alpha \wedge \neg\alpha}{\beta}$$

Some additional information on the lattice of strengthenings of any finite valued Łukasiewicz logic can be found in G. Malinowski [1977].

4. As one easily notice, the above does not concern at all the fundamental (as one might think) question as concern the problem of axiomatizability of logical matrices – namely, the question of finite axiomatizablity (that is axiomatizability with a finite set of schematically defined rules and axioms) of finite matrices (or, more specifically the sets of tautologies or matrix consequences determined by finite matrices). A partial solution to this question was given by M. Wajsberg and we discussed it above. This question become central in research programme initiated by R. Wójcicki [1977]. Let us call a consequence operation Cn strongly finite (or SF in short) iff $Cn = \overrightarrow{\mathfrak{N}_1} \cap \ldots \cap \overrightarrow{\mathfrak{N}_n}$ for some finite matrices $\mathfrak{N}_1, \ldots, \mathfrak{N}_n$. Then let us try to identify logical properties of strongly finite logics. If SF logics have some "nice" logical properties, then finitness will get some logical meaning and it will turn out that finitness is significant from a logical point of view.

a) The initial interest in SF logics was supported by the following two general results

(1) *Each strongly finite logic Cn is finite, that is $\alpha \in Cn(X)$ iff $\alpha \in Cn(Y)$ for some finite $Y \subseteq X$.*

(2) *If Cn is strongly finite, then its degree of completeness (that is the number of theories closed under substitution) is finite.*

As concerns (1) (see J. Łoś, R. Suszko [1958]) the argument used there is similar to that used if one wants to show the compactness theorem for satisfiability (ie. if one shows that a set is satisfiable iff all its finite subsets are satisfiable). One can simply make use of Boolean Prime Ideal Theorem or related (fundamental) results in set theory. In other words, our argument is set-theoretical, not logical, in nature. As concerns (2) the concept of the degree of completeness was introduced by A. Tarski and the result that any finite valued Łukasiewicz logic has a finite degree of completeness was noted in literature. From this point of view, the result (2) (by R. Wójcicki) was an essential strengthening of the known result. However, the result in question is an immediate corollary of the following characterization of the substitution version of the matrix consequence

$$\overrightarrow{\mathfrak{M}}(Sb(X)) = \bigcap\{E(\mathfrak{N}) : X \subseteq E(\mathfrak{N}) \text{ oraz } \mathfrak{N} \subseteq \mathfrak{M}\}$$

given by P. Wojtylak [1978]. It does not mater here if the matrix $\mathfrak{M}$ is finite or not. Since any finite matrix has only finitely many submatrices, the finitness of the degree of completeness of any strongly finite logic immediately follows. So, one can say that our argument with (2) has little to do with finitness of matrices and properties of strongly finite logics.

b) Nevertheless, the above "positive" results encouraged to make several conjectures on strongly finite logics. In particular,

(3) *Each structural strengthening of any SF logic is SF;*

$$Cn_1 \geq Cn \in SF \Rightarrow Cn_1 \in SF$$

(4) *The supremum of two SF logics is SF;*

$$Cn, Cn_1 \in SF \Rightarrow Cn \cup Cn_1 \in SF$$

(5) *Each strongly finite logic Cn is finitely based (FB in short) which means that for some finite set R of inferential rules (including axiom schemata)*

$$Cn(X) = Cn(R, X) \text{ for every } X$$

It soon appeared that all the above conjecture (and also some others) are false. Counterexamples were provided by M. Tokarz [1976], A. Wroński [1976] and [1979], P. Wojtylak [1979]. For the considered in this paper question of axiomatizability, the most important is the negative answer to (5) provided by A. Wroński [1979]. If one wants to get a positive result on finite axiomatizability of a finite matrix one has to assume certain logical properties of the logic determined by this matrix. To illustrate this question let us consider the following general (positive) results on axiomatizability of finite matrice from my paper [1979]:

If $\mathfrak{M}$ is a finite matrix and $\overrightarrow{\mathfrak{M}}$ is a consequence operation with equivalence and disjunction, then $\overrightarrow{\mathfrak{M}}$ jest finitely based.

Let me give here a sketch of its proof. Then, each finite structural consequence operation, in particular $\overrightarrow{\mathfrak{M}}$, can be given in the form $\bigcup_k Cn_k$ where Cn_k is the fragment of $\overrightarrow{\mathfrak{M}}$ determined by rules (and axioms) which can be defined by sequents containing at most k variables. Since each Cn_k can be easily shown to be finitely based, the by the known Tarski's criterion $\overrightarrow{\mathfrak{M}}$ is finitely based iff it collapses to one of its fragments Cn_k. Next, let us consired the following formulae

$$\bigvee\{p_i \equiv p_j : 0 \leq i < j \leq k\}$$

and let me notice that the same formulae were considered by K.Gödel when he showed that intuitionistic logic has no adequate finite matrix. According to our assumptions $\overrightarrow{\mathfrak{M}}$ is a consequence operation with equivalence and disjunction and hence the above formula is valid in $\mathfrak{M}$ if the number k is sufficiently large. Thus, we get

$$Cn_{k+1}(X) = \bigcap_{i<j\leq k} Cn_{k+1}(X, p_i \equiv p_j)$$

for each set X if k is sufficiently large. By use of this equation we can reduce rules of Cn_{k+1} to those of Cn_k and hence $\overrightarrow{\mathfrak{M}}$ collapses to Cn_k.

c) Let me make the following general comments. First, all counterexamples concerning strongly finite logics provided by many authors shows that, perhaps against some obvious suppositions, finiteness is not an assumption of a logical character. The finitness of semantics does not provide any interesting logical property unless the logic in question is sufficiently strong and regular. Finitness let alone means nothing in logic and it profits only if accompanied with additional assumptions of logical character. Second, the progress in formal sciences is a very complex subject. No progress is possible if one only reads works by one's masters without any attempt to improve them. Some arrogance or ignorance is sometimes needed to overcome old ruts and mistakes of our masters. It so happend with the considered subject when, against strong tradition in Polish logic, taking into account logical rules caused progress and originated interesting results and important questions. On the other hand, however, if we read more carefully works by our masters, we could avoid new mistakes and false conjectures. In particular, the old result by M. Wajsberg has the same form as the above one. Both theorems say that not all finite matrices enjoy finite axiomatizability (though finite axiomatizability is understand in both cases differently). Wajsberg's argument is very difficult to follow but in my opinion it resembles very much the argument sketched above.

5. Let me present some finite matrices considered in literature which do not enjoy finite axiomatizability. These counterexamples attracts much attention when one considers the question of finite axiomatizability.

a) I should begin with the oldest (known to me) counterexample by M. Wajsberg [1935]. It was intended to show the importance of the assumption of normality of the matrix considered. Normality of the matrix $\mathfrak{M}$ means that if the premises for the detachment rule are distinguished in the matrix, then the conclusion is distinguished too. Let $\mathfrak{M}$ be the matrix on $\{0, \ldots, k\}$ with $D = \{0\}$ as the set of distinguished elements and the logical operation (implication and negation) interpreted as

$$f^{\rightarrow}(x, y) = y \quad , \quad f^{\neg}(y) = 0.$$

One easily shows that $\mathfrak{M}$ tautologies are formulae

$$\neg\alpha \quad \text{or} \quad \alpha_1 \rightarrow (\alpha_2 \rightarrow (\ldots(\alpha_n \rightarrow \neg\alpha)\ldots)$$

So, it is clear that $E(\mathfrak{M})$ cannot be axiomatized by use by the detachment rule and a finite set of axioms. However, it does not settle the problem of finite axiomatizability if one allows other inferential rules. For instance, $\neg\alpha$ and the simplification rule suffice to derive all $\mathfrak{M}$ tautologies.

b) Now, let us consider the finite matrix given by Wroński [1979] the matrix consequence of which is not finitely based. The matrix is defined in a language with one binary operator. We will omit as usual the symbol of this operators and accept the convention of omitting brackets where xyz means $(xy)z$. The matrix is 3-element on $\{0, 1, 2\}$ and 2 is the only distinguished element and is defined by

	0	1	2
0	2	0	2
1	2	2	2
2	2	2	2

We will not repeat the proof of the fact that the matrix consequence is not finitely based. Let us only note that finite non-axiomatizability of the matrix consequence does not mean finite non-axiomatizability of the set of all tautologies. In fact, the set of all tautologies can be finitely axiomatized. As axioms we can take formulae of the form $\alpha(\beta\gamma)$ and $\alpha\alpha$ which are clearly valid in the matrix. Let us note that not only these formulae but also all formulae containing them as subformulae are valid in the matrix. So, we will also need the simplification rule in the form

$$\frac{\alpha}{\beta\ \alpha}$$

There remains to consider formulae of the form

$$p_{i_0}p_{i_1}\dots p_{i_k}$$

Any such formula is valid in the matrix iff $i_0 \in \{i_1, \dots, i_k\}$. To show all these valid instances it suffices to make use of the following additional rule

$$\frac{\alpha\ \beta}{\alpha\ \gamma\ \beta}.$$

Let us notice that the above rule, similarly as the above simplification rule, is normal in the matrix and hence we obtain a finite axiomatization of the set of all formulae valid in the matrix.

c) So, the above counterexample still leaves open the problem of finite (weak) axiomatizability of finite matrices. An example of a finite matrix

whose set of tautologies is not finitely axiomatizable was given in Wojtylak [1979]. It was a 5-element matrix with 2 elements distinguished. A better (in many respects) example was provied by W. Dziobiak [1991]. there was gives a 4-element matrix with one element distinguished with the same property. But the best result in this field is due to K. Pałasińska [1994]. She showed a 3-element matrix with one element distinguished which is not finitely axiomatizable in any reasonable way. The matrix is given by

	0	1	2
0	1(2)	2	2
1	2	2	2
2	1	2	2

where 2 is distinguished and 1(2) in the first row means that one can put 1 or 2 there. So, there are given two 3-element matrices with the same property. At first sight one easily notices a large number of 2's in the matrix. It must be so. If there were no 2's (or there were no enough 2's), then the set of all tautologies were empty and hence finitely axiomatizable. If we consider an operation with less number of 2's, but with nonempty set of tautologies, it usually happens that the operation is sufficiently strong to define disjunction and equivalence which gives finite axiomatizability of the matrix consequence. One should not expect that matrices which do not enjoy finite axiomatizability were natural and regular from the point of view of standards in logic.

d) The above results on finite non-axiomatizability also have certain consequences in universal algebra. Instead of axiomatizability of matrices we consider quasivarieties generated by a finite algebra and the results transforms. But these algebraic aspects are far beyond the scope of my lecture. At the end let me only say that the result by K. Pałasińska is by all means optimal and one should not even try to improve it. This is impossible as

If $|M| = 2$, *then* $\overrightarrow{M}$ *is finitely based.*

The proof of this result given by W. Rautenberg [1981] is long and not quite interesting as one has to consider a large number of cases. It gives me, however, an opportunity to get back to E. Post at the end. Namely, to show his result W. Rautenberg used Post's classification of all clones (that is sets of operations closed under substitions and projection) on a two element set. In consequence, the old result by E. Post can be used to support the view that, although finitness of semantics is not an assumption of a logical character, the two-valuedness does have a logical character. The admittance of a third logica value destoys more than one could even expect at the beginning.

References

C. C. Chang [1959], *A new proof of the completeness of the Łukasiewicz axioms*, Transactions of the American Mathematical Society 93, 74-80.

R. Cignoli, D. Mundici [1997], *An elementary proof of the Chang's completeness theorem for the infinite-valued calculus of Łukasiewicz*, Studia Logica 58, 79-97.

W. Dziobiak [1991], *A finite matrix whose set of tautologies is not finitely axiomatizable*, Reports on Mathematical Logic 25, 105-112.

J. Łoś, R. Suszko [1958], *Remarks in sentential logic*, Indagationes Mathematicae 20, 177-183.

G. Malinowski [1977], *Degrees of maximality of Łukasiewicz-like sentential calculi*, Studia Logica 34, 213-228.

K. Pałasińska [1994], *Three-element nonfinitely axiomatizable matrices*, Studia Logica 53, 361-372.

W. A. Pogorzelski [1971], *Structural completeness of the propositional calculus*, Bulletin de l'Académie Polonaise des Sciences, série des Sciences Mathématiques, Astronomiques et Physiques 19, 349-351.

W. A. Pogorzelski, P. Wojtylak [1982], *elements of the completeness in propositional logic*, Silesian University, Katowice.

E. Post [1921], *Introduction to a general theory of elementary proposition*, American Journal of Mathematics 43, 163-185.

W. Rautenberg [1981], *2-element matrices*, Studia Logica 40, 315-353.

M. Tokarz [1976], *A strongly finite logic with infinite degree of maximality*, Studia Logica 35, 447-451.

R. Tuziak [1988], *an axiomatization of the finitely-valued Łukasiewicz calculus*, Studia Logica 48, 49-56.

Wajsberg M. [1935], *Beiträge zum Metaaussagenkalkül I*, Monatshefte für Mathematik und Physik 42, 221-242; English translation *Contributions to meta-calculus of propositions I* in *Mordchaj Wajsberg logical works* ed. S. J. Surma, Ossolineum, Warszawa-Wrocław 1973.

P. Wojtylak [1978], *On structural completeness of many-valued logics*, Studia Logica 37, 139-147.

P. Wojtylak [1979], *Strongly finite logics: finite axiomatizability and the problem of supremum*, Bulletin of the Section of Logic, Polish Academy of Sciences 8, 99-111.

P. Wojtylak [1984], *An example of a finite though finitely non-axiomatizable matrix*, Reports on Mathematical Logic 17, 39-46.

R. Wójcicki [1973], *On matrix representations of consequence operations of Łukasiewicz sententail calculi*, Zeitrshrift für Mathematische Logik und Grundlagen der Mathematik 19, 230-247.

R. Wójcicki [1977] *Strongly finite sentential calculi*, in *Selected papers on Łukasiewicz sentential calculi* ed R. Wójcicki, G. Malinowski, Ossolineum, Warszawa-Wrocław.

A. Wroński [1976], *On finitely based consequence operations.* Studia Logica 35, 453-458.

A. Wroński [1979], *A three element matrix whose consequence is not finitely axiomatizable*, Bulletin of the Section of Logic, Polish Academy of Sciences 8, 68-71. Studia Logica 35, 453-458.

Piotr Wojtylak
Institute of Mathematics
Silesian University, Katowice
e-mail: wojtylak@ux2.math.us.edu.pl

Witold Marciszewski

POST'S PROBLEM OF CREATIVITY AND 'NATURE AS INFINITE INTELLIGENCE'*

The phrase quoted in the title is found in Emil L. Post's *Diary* which under the title "Time Accounts" was begun in the spring of 1916 and continued to the spring of 1922[1]. That time interval embraces the most productive time in Post's life when he prepared his doctoral dissertation (1920) "Introduction to a general theory of elementary propositions" (*The American Journal of Mathematics*, *43*, 1931, 163-185) – that so seminal work which (i) introduced the truth-table method, (ii) with generalizing that method put algebraic foundations of multi-valued logic, and (iii) provided a general framewok for systems of logic as means of deriving theorems through *finitary* symbol manipulation.

Like great predecessors being both mathematicians and philosophers (notably Pascal), Post carefully distinguished scientific results, to be made public, from incomplete projects and philosophical intuitions to be entertained in privacy until they mature enough. This in why in the time he wrote the dissertation he made notes of his problems and dawning ideas, somehow related to his 'public' research. Those thoughts did not come to light until Martin Davis as the editor of *The Collected Works of Emil L. Post* put them as Appendix to Post's paper on absolutely unsolvable problems.

Post contributed to grasping the essence of finitary mechanical operations on equal footing with Turing and Hilbert.

If someone asks how is Hilbert involved in mechanization research, this can be answered by quoting Turing's description of what he calls *paper machines*. It amounts to what Hilbert called formalization. This description runs as follows.

* The work presented here was supported by the Polish Ministry of Science ('Komitet Badań Naukowych'), Committee for Informatics [etc.], under the grant to the Chair of Logic, Informatics and Philosophy of Science at the University of Białystok, No. 8 T11C 018 12, for research concerning *AI vs. Natural Intelligence from the Point of View of Mechanization of Reasoning*.

[1] See "Absolutly Unsolvable Problems and Relatively Undecidable Propositions – Account of an Anticipation" in *Solvability, Provability, Definability. The Collected Works of Emil L. Post* edited by Martin Davis. Birkhäuser (Boston, etc.) 1994.

"It is possible to produce the effect of computing machine by writing down a set of rules of procedure and asking a man to carry them out. Such a combination of a man with written instructions will be called a 'Paper Machine'. A man provided with paper, pencil, and rubber, and subject to strict discipline, is in effect a universal machine. The expression 'paper machine' will be often used below." A. Turing, 'Intelligent Machinery' in *Collected works of A.M. Turing. Mechanical Intelligence*, ed. D.C. Ince, North-Holland 1992. See p. 113.

However, Post was not fully satisfied with his theory of such operations, since it did not explain the riddle of mathematical creativity. It was the phenomenon which astonished and intrigued him (like Pascal who wondered that our reason before starts a precise proof has to trust a 'feeling' that there are numbers, space, time, motion, etc).

The problem of creativity was seen by Post as closely tied to that of solvability. There are problems which can be solved by mere manipulating symbols in a finite chain of steps; these do not require a creative thought. However, people successfully deal with problems which are not solvable in that way. How is it possible? How to describe the mechanism being behind such processes? What kind of logic could render and guide them?

When reading Post's *Diary*, we find no definite answer to such questions. His notes express rather a dim anxiety than a conclusive line of thought. However, they should be appreciated as witnessing the struggles of a great mind at the peak of his creative powers. His helplessnes gives us a measure of the degree of objective difficulties in that part of philosophy of science which so much absorbed Post.

Is there any chance in attacking the problem some tens of years later? The answer seems to be in the affirmative since we had a lot of discussions on the issues raised by Post (a great deal of them due to Gödel and Turing). Moreover, we have some experiences with computing and reasonig machines, and with computational power of Nature, which were not available to Post (he died in 1954). A totality of such discussions may be covered by what nowadays is called *cognitive science*. The objective of this essay is to confront Post's questions with some ideas belonging to this new field.

1. Two Concepts of Solvability

There is a remarkable difference between the lexical meaning of the word *solvability* and its technical meaning which is in the focus of logic, philosophy of science, computer science, and cognitive science. To wit, the lexical meaning as defined, eg, by Webster's Third New International Dictionary, is as follows:

solve – to find an answer, explanation or remedy for; arrive at a clear, definite and satisfying answer.

solution – the fact or state of a problem's being solved.

solvable – suceptible of solution.

solvability – the quality or state of being solvable.

Obviously, in this context 'answer' means a true answer.

Now, take the famous Gödel sentence, G for short, and put the problem:

[P_G] *Is G true?*

Provided that G has number N at a list of propositions of a formalized system S, the sentence G, roughly, runs as follows.

[N] *The sentence no. N is not provable in S, provided that S is consistent.*

The answer YES to P_G results from the following consideration. The denial of G is to the effect that G is provable in S, and this means that a sentence which predicates non-provability about itself is provable. This is a contradiction. Since the denial of G implies contradiction, this denial must be false, hence G itself is true.

The answer we arrive at is clear, definite and satisfying, as required by Webster. Thus the problem answered by it enjoys the quality of being solvable, in the ordinary lexical meaning of this word.

However, in the technical meaning which holds in logic and in philosophy of science, the problem P_G is not solvable, since there is no mechanical procedure, such as a formalized proof, to check the truth of G.

Now, one may ask: *Which concept of solvability should be endorsed by cognitive science?* That being in use in ordinary English, or that defined in logic, philosophy of science and computer science? The latter option is highly plausible because of yielding a precise definition. On the other hand, cognitive science deals with the functioning of human intelligence in the conditions of everyday life, and its features are best rendered by our everyday language.

2. Solvability as the point where cognitive science meets philosophy of science

The science-philosophical term *Entscheidungsproblem* can be rendered as the *problem of solvability* as well as *decision problem.* The transitive verbs 'to solve' and 'do decide' enter a grammatical structure with different objects: it is a *problem* which is said to be solvable, while it is a *proposition* (or a set of them, as a theory) which is said to be decidable. This usage can be exemplified by the title of Post's study referred to in this discussion.

Thus, a problem is said to be solvable if the answer, either in the affirmative or in the negative, is decidable. A part of the technical sense of 'solvability' appears in Post's usage of the term *finiteness* as the property of a mechanical procedure which consists in reaching a solution in a finite number of steps.

At the same time, this term enters the definition of intelligence to mean the ability of problem solving. Since any procedures of representing and processing information by minds and organisms (the subject-matter of cognitive science) serve solving problems, the concept of solvability is as crucial for cognitive science as is for philosophy of science.

To solve problems was always the goal of science, and thus the subject of philosophical reflexion on science. However, there was no problem of solvability in the modern sense until Hilbert in 1900 announced his great programme, and then Gödel – precursed by Post – has demonstrated its limitations.

In modern philosophy of science, the problem of solvability emerged with dramatic strengthening of the rigour of proof. This new rigour is what is called *formalization*, and it amounts to a *mechanical* procedure in proving. It was Frege, Peano and Russell who paved the way to this new notion in their axiomatic approaches, while Hilbert was the one who put it into an explicit methodological doctrine. Now, to solve a problem means to support the answer with a formalized proof (or, at least, a proof liable to such enhancement).

Should the term *mechanical* be seen as a metaphor or be taken literally? Since the latter is the case – as shown by Turing, Post, and others – the theory of logic, branching into philosophy of science, meets both the theory of problem solving machines (computer science) and that of problem solving organisms (cognitive science).

It would be advantageous to have a more comprehensive concept to cover both computer science and cognitive science. The term 'informatics' seems to be a suitable candidate for that role, though sometimes it is used in a narrower sense, which amounts to that of 'computer science' (this terminological issue deserves a careful consideration).

The great problem of cognitive science is as follows. In the present state of research on information processing – as a procedure of solving problems – we have to distinguish between two kinds of systems, to wit mechanical and creative systems. Should this difference become less and less, as we shall gain ever more knowlwedge on the both kinds of systems? Or, is it fundamental, that is, not being likely to disappear?

Turing believed that the producing of creative machines was just a matter of time; thus the mentioned disparity should disappear with the progress of our knowledge and technology. In spite of obtaining similar technical results, Post entertained a much diferent feeling. He saw the phenomenon of mental creativity as something that hardly could be reduced to operations of machine, even a highly advanced one.

Obviously, such contrasting approaches produce two divergent perspectives on cognitive science (including a theory of mind) as well as philosophy of science. In that of Turing, both disciplines would tend to become parts of a general theory of machines. In that of Post, they will ever preserve their specific problem and methods, the theory of mind being concerned with the solving of problems by creative non-mechanical minds.

3. Post's engagement in the problem of creativity

Emil Post's concern with what we nowadays call cognitive science was greater than it can be judged when one reads his published metamathematical results. We learn about it owing to a ceratain coincidence of facts (partly reported by Roman Murawski in his contribution to this volume, Section 'Canonical systems').

To wit, Post anticipated Gödel's results on incompleteness in his unpublished Diary. He did not publish those notes for his having been aware that they needed a more detailed elaboration (moreover, he imagined how shocking such a highly unorthodox point would have been in the academic atmosphere of the twenties).

However, after Gödel's results had been published, Post desired to let people know about his own approach – as a methodological alternative worth to be discussed. To certify these claims, he was ready to make publicly available a diary and notes where his ideas were sketched. The full text of them has been edited by his pupil M. Davis in the *Collected Works*, and so we got a look into those private records. There we encounter remarks on human creativity, much to the point for cognitive science. These comments form the Appendix to the paper in question where Post wrote in Introduction with emphasis. (p. 378)

But perhaps the greatest service the present account could render would stem from its stressing of its final conclusion that *mathematical thinking is, and must be, essentially creative.* It is to the writer's continuing amazement that ten years after Gödel's remarkable achievement current views on the nature of mathematics are thereby affected only to the point of seeing the need of many formal systems, instead of a universal one. Rather has it seemed to us to be inevitable that these developments will result in a reversal of the entire axiomatic trend of the late 19th and early 20th centuries, with a return to meaning and truth. Postulational thinking will then remain as but one phase of mathematical thinking.

In the above passage, the italicized part is accompanied by the following footnote (no. 12).

Yet, as this account emphasises, the creativeness of human mathematics has a counterpart inescapable limitations thereof – witness the absolutly unsolvable (combinatory) problems. Indeed, with the bubble of symbolic logic as universal logical machine finally burst, a new future dawns for it as the indispensable means for revealing and developing those limitations. For, in the spirit of the Appendix, Symbolic Logic may be said to be Mathematics self-conscious. [Aactually, the old dream of symbolic logic is finding partial realization in Tarski's recent positive work on decision problems.]

Let us notice the expressive metaphor that "the bubble of symbolic logic as universal logical machine finally burst". On the basis of that conviction, Post claims the restatement of the goal of logic towards its becoming just self-consciousness of mathematics (instead of being its universal tool). When compared with the contention (of some cognitive scientists) that nothing essentially changed with discoveries of Gödel, Turing, Church, and Post himself, this attitude should draw close attention. Post may have been wrong, nevertheless his intellectual quests are worth to be traced.

In what follows those introductory comments, Post develops his ideas and results in two parts. Part I, entitled *Formal Transformations*, gives us an account of his theory of canonical forms (A, B, C), ie normal forms to which propositions of a logical system, namely *Principia Mathematica*, can be reduced. Part II *The Anticipation* serves the purpose mentioned in the title of his account, namely, to show how the results reported in Part I anticipated those of Gödel; these points are extensively disussed in Sections 2 and 3, respectively, of Roman Murawski's paper (this volume).

Part II concludes with the following statement:

A complete symbolic logic is impossible.

It follows from two previously proved theorems (whose content is hinted at in the footnote quoted below), and is commented as being in line with ideas of other authors, as Russell, C. I. Lewis, and (unexpectedly enough) Bergson as the author of *Creative Evolution.* The latter may prove an important hint in interpreting Post's attitude. This final statement is provided with an extensive footnote (no. 101, p. 428); it deserves to be cited as a whole,

being related to the distinction of two concepts of solvability discussed above (Section 1).

Mere incompleteness, as in the first of the two "Theorems" preceding, might not rule out the logic being as complete as it ever could be made. Fundamental, then, is the added effect of the second theorem, which rules out the posibility of a completed symbolic logic. That is, any symbolic logic can be made more complete. [Post, as seen from the context, means logic in the sense of such an extensive system as *Principia*, which in the sequel he calls 'upper reaches of symbolic logic"] It is doubtful if the writer ever paused note the mere incompleteness of a symbolic logic in the sense of existence of some undecidable propositions therein, for experience with Zermelo's axiom, the axiom of infinity, and the theory of types clearly leads one expect incompleteness in the upper reaches of symbolic logic. Rather was the emphasis placed on the stronger concept of incompleteness with respect to a fixed subject matter, in the present instance the propositions stating whether a given sequence is or not is generated by the productions in a given normal systems from its initial sequence. Likewise, Gödel would stress, for example, the incompleteness of any symbolic logic with respect to the class of arithmetical propositions. Where we say "symbolic logic" the tendency is now to say "finitary symbolic logic". However, it seems to the writer that logic should be considered essentially a human enterprise, and that when this is departed from, it is *then* incumbent on such a writer to add a qualifying "non-finitary".

After having so explained the concluding maxim that a complete symbolic logic is impossible, Post gives this thought still another formulation, when saying: "Better still, we may write

The Logical Process is essentially Creative.

This conclusion [...] makes of the mathematician much more than a kind of clever being who can do quickly what a *machine* could do ultimately. We see that a *machine* would never give a complete logic; for once the machine is made *we* could prove a theorem it does not prove."

This is the last passage of the study in question. Then follows Appendix including the *Diary.* In it various observations are recorded to explain the fact stated in the above conclusion (the pages below refer to *Collected Works*).

Post notices that in the mental process of proof creative and non-creative parts are intermingled (p. 433). Those creative ones are found in a stream of consciousness extended in time while the non-creative parts consist of symbols manipulated which are extended in space (p. 431). The creative parts are not expressed in symbols, and therefore the mind may be unaware of them (p. 434).

There are in the text comments on a connexion between the creative side in the process of proof and transfinite ordinals. For scarcity of a context such utterances are not easy for interpretation. They may mean that the discovery of transfinite induction, following the discovery of tranfinite ordinals, provides a most striking example of creativity which consists in a good vision of infinitude of elements which possess stateable properties. This, Post says (p. 56), cannot be unravelled by our logical process of syllogism etc. (if logical, then mechanical or combinatory, for logic is meant here as finitary logic).

Whatever this should mean, there arises the question concerning a source of that capability of 'seeing the infinitude' which transdends the abilities of a machine. There is a clue to a tentative answer if we take into account the phrase quoted in the title, but only when we accept an assumption which is by Post never mentioned (at the same time, there is no evidence that he would question it).

To wit, let us take literally his maxim that nature is (or possesses) an infinite intelligence. The brain of a human (or even a non-human animal) is a part of nature (unlike a machine made by humans), and so may participate in its infinite intelligence.

If someone feels this conjecture as a crazy metaphysics, let him take its weaker form. That there are giant computational powers in living beings is no crazy claim nowadays (though this fact was less conspicuous in Post's times). At the same time, owing to our experiences with computers, we know the enormous role of miniaturization for obtaining ever higher computational powers. This is why a vision of quantum computer is so promising for the increase of computational powers available to us. Now suppose that the human body, or mind (seen also as part of nature) has computational power even greater than quantum computer.

To put the thing in a nutshell, the greater is the complexity of computing, the greater complexity is demanded from the computing device, and in the existing physical conditions the latter is being increased through ever deeper miniaturization. This, in turn, is incomparably greater with minds than with machines.

Now the crucial issue is whether the complexity of nature can be ever matched by human technology. There is no ready answer yet. However, those who believe – as do some quantum physicists (eg, Louis de Broglie, David Bohm, Basil Hiley), some mathematicians (eg, Georg Cantor, Stanisław Ulam) and some classics of metaphysics (eg, Pascal and Leibniz) – in the infinitude of levels with increasing degrees of complexity in the physical world, should be ready to assume that the mind may be located at a level of complexity not to be matched by human technology. Owing to that, an internal mental code (a notion that should have been enjoyed by Post, had he learned it), if recorded at a sufficiently deep level of complexity (possibly in a continuous structure) might prove unsurpassable by any discrete symbolic code of a machine.

No one can now prove things like those, but no one can disprove either; this is an open option to be investigated. Those who would try to investigate, may find encouragement in Post's belief in the infinite intelligence of nature which inspired his vision of human creativity.

Witold Marciszewski
Chair of Logic, Informatics and Philosophy of Science
at the University of Białystok
e-mail: witmar@calculemus.org
home page: www.calculemus.org/wit.html

STUDIES IN LOGIC, GRAMMAR AND RHETORIC 2 (15)

Urszula Wybraniec-Skardowska

A FORMAL-LOGICAL APPROACH TO THE PROBLEM OF IMPRECISION

Abstract

The problems of the paper refer to the theory of knowledge and its representation, and may be used for the needs of semiotics and artificial intelligence. It contains a proposal of a formal-logical approach to the question of imprecision: incompleteness and uncertainty in information systems in the spirit of Z. Pawlak's conceptions [3,2,4,6]. It contains explication of the notions of empty, exact and inexact, i.e., imprecise: incomplete or vague information, it defines the accuracy coefficient of information in the process of communication and it shows the relationship of classical logic to reasoning based on incomplete or vague premises.

1. Introduction

The problems of the imprecision of knowledge and its representation has been the subject of reflection of logical semiotics, philosophy and linguistics for a long time. Recently it has become the subject of investigations of computer scientists interested in the problems of artificial intelligence, in particular in the questions of reasoning on the basis of incomplete or vague information and in the possibility of a representation of such information in computer memory.

In the process of cognition of a definite fragment of reality, the cognitive agent (a man or a group of men, some other living organism, or a group of organisms, or a robot) attempts to discover objective information contained in it. The agent's knowledge of the given reality, and of the components of knowledge, i.e. information about the particular objects of the reality, is the basis for reasoning and communication with other agents. This knowledge may be *empty, exact or imprecise: incomplete or vague.*

The subject of this paper is a formal-logical approach to the problem of the imprecision of information about objects of reality. The main ideas and intuitions are taken from the author's papers [8-11]. Reality will be understood as a relational structure. Reality discovered by the agent will be defined through an assigned information system in Z. Pawlak's sense [3,4,6].

2. Reality and information contained in it

Reality is complex and hierarchical. In the cognitive process the agent discovers its fragments. Each such fragment $\mathfrak{R}$ may be described as the following relational structure:

$$\mathfrak{R} = \langle U, \mathcal{R}_1, \ldots, \mathcal{R}_n \rangle,$$

where U, the *universe of objects of reality* $\mathfrak{R}$, is a nonempty set and $\mathcal{R}_i$, for $i = 1, \ldots, n$, is the set of i-ary relations on U. These relations have either a momentary, situational, unit character or a more steady, constituted character. In this way reality $\mathfrak{R}$ may be of a twofold kind:

$S\mathfrak{R}$ or $C\mathfrak{R}$

where

1° **Situational Reality**		2° **Constituted Reality**
$\mathcal{R}_1 \subseteq \{r : r = (o) = o \in U\} = U$	(1)	$\mathcal{R}_1 \subseteq \{R : R \subseteq U\} = P(U)$
$\mathcal{R}_{i+1} \subseteq \{r : r = (o_1, \ldots, o_{i+1}) \in U^{i+1}\} = U^{i+1}$	$(i+1)$	$\mathcal{R}_{i+1} \subseteq \{R : R \subseteq U^{i+1}\} = P(U^{i+1})$

(*) $\forall o \in U \; \exists r \in \mathcal{R}_{i+1} \; \exists o_1, \ldots, o_i \in U \quad r = (o_1, \ldots, o, \ldots, o_i)$.

In case 1°, for the situational reality $S\mathfrak{R}$, its 1-ary, *unit relation* r is a unit property of an object identifying this object – in a formal approach the unit relation r is identified with this object (see (1)), while multi-ary relation r is a mutual location, co-occurrence of objects of a system – in a formal approach it is identified with this system (see $(i+1)$). For the objects of the situational reality we assume additionally (see (*)) that for every object of the reality $S\mathfrak{R}$ there exists a certain multi-ary relation, determining its co-occurrence with some objects.

In case 2°, for the constituted reality $C\mathfrak{R}$, its unary relation R is a property of objects of the universe – in a formal approach identified with a subset of the objects of the universe which can be regarded as an equivalence

class of the *indiscernibility* objects – objects possessing this property, i.e. the same property (see (1)), while the multi-ary relation R is identified with the relationship occurring between systems of objects – in a formal approach this is a set-theoretical relation, i.e. a subset of the finite Cartesian product of the set U which as the set of ordered systems of objects of the universe can be regarded as equivalence classes of *indiscernibility* systems of objects – systems remaining in this relationship, i.e. the same relationship (see $(i+1)$).

The objective knowledge contained in reality consists of pieces of *unit information about the objects of* U which are determined by the relations defining $\mathfrak{R}$. They are the following image functions $\vec{\ }$ of objects with respect to relations in $\mathfrak{R}$:

1° for $S\mathfrak{R}$

(1) if $r \in \mathcal{R}_1$, then $\vec{r}: \{r\} \to \{r\}$ and $\vec{r}(o) = o$ for $o = r$;

$(i+1)$ if $r \in R_{i+1}$, $r = (o_1, \ldots, o_{i+1})$,
then $\vec{r}: \{o_l\} \to \{(o_1, \ldots, o_{l-1}, o_{l+1}, \ldots, o_{i+1})\}$
and $\vec{r}(o_l) = (o_1, \ldots, o_{l+1}, o_{l+1}, \ldots, o_{i+1})$
for $o_l \in U$, $l = 1, \ldots, i+1$

2° for $C\mathfrak{R}$

(1) if $R \in \mathcal{R}_1$, then $\vec{R}: R \to P(U)$ and $\vec{R}(o) = R$ for every $o \in R$;

$(i+1)$ if $R \in R_{i+1}$, $D_k(R)$ is k-th domain of R $(k = 1, \ldots, i+1)$,
then $\vec{R}: D_l(R) \to D_1 \times \ldots \times D_{l-1} \times D_{l+1} \times \ldots \times D_{i+1}$ and
$\vec{R}(o_l) = \{(o_1, \ldots, o_{l-1}, o_{l+1}, \ldots, o_{i+1}) : (o_1, \ldots, o_{l-1}, o_l, o_{l+1}, \ldots, o_{i+1}) \in R\}$
for all $o_l \in D_l(R)$, $l = 1, \ldots, i+1$.

In the situational reality $S\mathfrak{R}$ a piece of unit information about its object with respect to a unary relation r (individual property of this object) is this object, while with respect to a multi-ary relation r – ordered system of remaining objects constituting this relation together with this objects.

In the constituted reality $C\mathfrak{R}$ a piece of unit information about its object with respect to a unary relation R to which this object belongs is this relation, i.e. an ascribed property – the set to which this object belongs as possessing this property, while a piece of unit information about its object with respect to the multi-ary relation R is the set of all the ordered systems of objects with which the object remains in relation.

Information contained in reality consists of suitable sets of unit information determined by relations of this reality:

1° for $S\mathfrak{R}$ $\qquad$ 2° for $C\mathfrak{R}$

$IS\mathfrak{R} = \langle \vec{\mathcal{R}}_1, \ldots, \vec{\mathcal{R}}_n \rangle$ $\qquad$ $IC\mathfrak{R} = \langle \vec{\mathcal{R}}_1, \ldots, \vec{\mathcal{R}}_n \rangle$

where

$\vec{\mathcal{R}}_i = \{\vec{r} : r \in \mathcal{R}_i\}$ $\qquad$ $\vec{\mathcal{R}}_i = \{\vec{R} : R \in \mathcal{R}_i\}$

for $i = 1, \ldots, n$.

3. Reality determined by an information system and its information

We usally discover reality and the objective information contained in it, in a more or less exact way, with respect to definite attributes of the objects of its universe U. As the universe we usually choose a finite set of its objects Ob and we put it forward as a generalized *attribute-value system* Σ called also an *information system* (cf. [1] and [3,2,4,6]):

$$\Sigma = \langle Ob, A_1, \ldots, A_n \rangle,$$

where $Ob \subseteq U$, $|Ob| < \omega$ and A_i $(i = 1, \ldots, n)$ is the set of i-argument *attributes* understood as i-ary functions, i.e.

$$\forall a \in A_i \; a : Ob^i \to V_a,$$

where V_a is the set of all *values* of the attribute a.

In the case where Σ determines the situational reality, an additional condition is satisfied: a is a one to one function onto V_a, i.e.

$$\forall a \in A_i \; \forall v \in V_a \; \exists^1 (o_1, \ldots, o_i) \in Ob^i \; (a(o_1, \ldots, o_i) = v).$$

Every attribute of the information system Σ and every value of this attribute explicitly assigns a relation belonging to the reality $\mathfrak{R}(\Sigma)$. The fragment $\mathcal{R}(\Sigma)$ of the reality discovered by means of information system Σ can be described as follows:

1° **Situational Reality** $S\mathfrak{R}(\Sigma)$

$$S\mathfrak{R}(\Sigma) = \langle Ob, \overbrace{\{\{r_{a,v}\} v \in V_a : a \in A_1\}}^{\mathcal{R}_1}, \ldots, \overbrace{\{\{r_{a,v}\} v \in V_a : a \in A_n\}}^{\mathcal{R}_n} \rangle,$$

where

$$\forall i = 1, \ldots, n \; \forall a \in A_i \; \forall v \in V_a \; (r_{a,v} = (o_1, \ldots, o_i) \iff a(o_1, \ldots, o_i) = v);$$

2° **Constituted Reality** $C\mathfrak{R}(\Sigma)$

$$C\mathfrak{R}(\Sigma) = \langle Ob, \overbrace{\{\{R_{a,v}\} v \in V_a : a \in A_1\}}^{\mathcal{R}_1}, \ldots, \overbrace{\{\{R_{a,v}\} v \in V_a : a \in A_n\}}^{\mathcal{R}_n} \rangle,$$

where

$$\forall i = 1, \ldots, n \; \forall a \in A_i \; \forall v \in V_a \; (R_{a,v} = \{(o_1, \ldots, o_i) \in Ob^i : a(o_1, \ldots, o_i) = v\}).$$

Both realities consist of the set of objects Ob and relations determined by any attribute a and its values. In case $S\mathfrak{R}(\Sigma)$ any relation $r_{a,v}$ determined by an attribute a and its value v is an ordered system such that the attribute's value for it is equal to v. In case $C\mathfrak{R}(\Sigma)$ any relation $R_{a,v}$ determined by an attribute a and its value v is the set of all ordered systems such that the attribute's value for each of them equals v.

Each piece of *unit information* determining information $IS\mathfrak{R}(\Sigma)$ or $IC\mathfrak{R}(\Sigma)$ is the image function $\vec{\ }$ with respect to the relation in the form $r_{a,v}$ or $R_{a,v}$ assigning it.

Example 1. A "situational" information system is a concrete concert of a symphony orchestra, e.g. a concert CON of the New York Philharmonic Orchestra (NYPO) characterized as the following system:

$$\mathrm{CON} = \langle \{1, 2, 3\}, A_1, A_2, A_3 \rangle,$$

where: object 1 is the conductor, object 2 is NYPO and object 3 is the audience, while $A_1 = \{\text{LOCATION}, \text{POSITION}\}$, $A_2 = \{\text{MUTUAL LOCATION}\}$, $A_3 = \{\text{ACTIVITY}\}$. The set $V_{LOC.}$ is defined through assigning the values: *conductor's podium* for 1, *stage* for 2, *place for audience* for 3, and the set $V_{POS.}$ – the values: *standing* for 1, *sitting*^{2} for 2, *sitting*^{3} for 3. The set $V_{MUT.LOC.} = \{n_{11}, f_{12}, b_{13}, f_{21}, n_{22}, f_{23}, f_{31}, f_{32}, n_{33}\}$, where each of the values n_{kk} is *neutral to oneself* – k to k, f_{kj} are the values of *facing* – k facing j, for $k, j = 1, 2, 3$, and b_{13} is the value of *standing with his back to* – 1 to 3.

Fourth attribute – ACTIVITY does not refer to a potential activity of objects 1, 2, 3 but to the situational activity in which the subject performs a definite activity by referring to two remaining subjects, in particular to oneself. The domain of the attribute (function) ACTIVITY is compounded from 27 ordered triples of objects and from the same number of values $v_{jkl} = \mathrm{ACT}_{(j,k,l)}$ $(j, k, l = 1, 2, 3)$ of this attribute determining the set V_{ACT}. We can assume that its values in particular are the following:

v_{123} = *conducting* $.\overset{2}{.}.$ *in front of* $.\overset{3}{.}.$

v_{132} = *presenting* $.\overset{3}{.}.$ *the way of conducting* $.\overset{2}{.}.$

v_{111} = *making movements with one's (1) hands and body according to the score perceived by* $.\overset{1}{.}.$
v_{222} = *playing one's* $.\overset{2}{.}.$ *instruments for oneself (2)*
v_{333} = *perceiving the concert with one's (3) senses connected with its reception by* $.\overset{3}{.}.$
v_{213} = *playing under conduction of* $.\overset{1}{.}.$ *for* $.\overset{3}{.}.$
v_{223} = *playing one's* $.\overset{2}{.}.$ *instruments for others* $.\overset{3}{.}.$
v_{321} = *perceiving the play of the orchestra* $.\overset{2}{.}.$ *conducted by* $.\overset{1}{.}.$.

The situational reality determined by the information system CON is the relational structure:

$$\mathrm{S}\mathcal{R}(\mathrm{CON}) = \langle\{1,2,3\}, \mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3\rangle,$$

where
$\mathcal{R}_1 = \{r_{LOC.,v}\}v \in V_{LOC.} \cup \{r_{POS.,v}\}v \in V_{POS.}$,
$\mathcal{R}_2 = \{r_{MUT.LOC.,v}\}v \in V_{MUT.LOC.}$,
$\mathcal{R}_3 = \{r_{ACT.,v}\}v \in V_{ACT.}$.

The information contained in reality $\mathrm{S}\mathfrak{R}(\mathrm{CON})$ is the system

$$\mathrm{ISR}(\mathrm{CON}) = \langle\vec{\mathcal{R}}_1, \vec{\mathcal{R}}_2, \vec{\mathcal{R}}_3\rangle.$$

The unit information of set $\vec{\mathcal{R}}_1$ is $3+3=6$, $\vec{\mathcal{R}}_2$ – 9, $\vec{\mathcal{R}}_3$ – 27.

Example 2. A "constituted" information system is NYPO (the New York Philharmonic Orchestra) mentioned in Example 1 and understood as the following system:

$$\mathrm{NYPO} = \langle\{o_1, o_2, \ldots, o_{90}\}, \overset{A_1}{\{\text{PLAYING INSTRUMENT}, \text{PART}\}}, \overset{A_2}{\{\text{DEPENDENCE}\}}\rangle,$$

where the universe of the NYPO system is assumed to be a 90-element set of all the members of the New York Philharmonic Orchestra, where PLAYING INSTRUMENT is a one-argument attribute for which all the values will be playing a particular instrument (thus $V_{PL.INS.}$ = {*piano, harp, violin I, violin II, violoncello, doublebass, viola, flute, oboe, horn, clarinet, fagot, trumpet, trombone, tuba, percussion*}); PART is also a one-argument attribute, understood as a musical function in the orchestra in general; the set of values V_{PART} consists of particular principal instrumentalists (i.e., *concert master, principal violoncellist, etc.*), particular associate principal instrumentalists (i.e., *associate concert master, assistant principal cellist, etc.*) and *regular members*, DEPENDENCE is a two-argument attribute for which a set of all values $V_{DEP.} = \{id, indp, dep, ld\}$, is compounded from

the values: *identical, independent, dependent, leading* a particular instrumentalists of a given group.

The constituted reality determined by the NYPO system is as follows:

$$\mathrm{C}\mathfrak{R}(\mathrm{NYPO}) = \langle\{o_1, o_2, \ldots, o_{90}\}, \mathcal{R}_1, \mathcal{R}_2\rangle,$$

where
$\mathcal{R}_1 = \{R_{PL.INS.,v}\}v \in V_{PL.INS.} \cup \{R_{PART,v}\}v \in V_{PART}$
$\mathcal{R}_2 = \{R_{DEP.,v}\}v \in V_{DEP.}$

Information contained in the reality is

$$\mathrm{IC}\mathfrak{R}(\mathrm{NYPO}) = \langle\vec{\mathcal{R}}_1, \vec{\mathcal{R}}_2\rangle.$$

4. Complete information and incomplete information

The information $\mathrm{I}\mathcal{R}(\Sigma)$ contained in reality $\mathfrak{R}(\Sigma)$ is objective. A piece of unit information discovered by the cognitive agent is information about the objects of this reality. It is subjective information – it depends on the knowledge of this agent about the attributes of an information system Σ.

4.1. Knowledge about attributes of the system Σ

Let $a \in A_j$ $(j = 1, \ldots, n)$. The knowledge K_a about the attribute a of Σ is – according to Professor A. Skowron's suggestion – the following set:

$$K_a = \{(o, V_a^o) \colon o \in Ob^j\},$$

where $V_a^o \subseteq V_a$ and V_a^o is *the set of all possible values of the attribute a for the object $o \in Ob^j$* from the point of view of the agent discovering the reality $\mathfrak{R}(\Sigma)$. The knowledge of this agent corresponds to his ability to solve the following *equation of the agent's unknowing* for any object $o \in Ob^j$:

$$(e) \qquad a(o) = x,$$

where x is the unknown quantity the range of which is the set V_a^o. The knowledge K_a of this agent about the attribute a and its value for the object o can be either (0) *empty* or (1) *exact* or (>1) *inexact*, i.e. *imprecise*. In case (0) the equation (e) has no solution, in case (1) it has exactly one solution and in case (>1) it has more than one solution and we have, respectively

$$(0) \quad |V_a^o| = 0, \qquad (1) \quad |V_a^o| = 1, \qquad (>1) \quad |V_a^o| > 1.$$

The knowledge K_a about the attribute $a \in A_j$ of the system Σ is *totally empty*, if every object $o \in Ob^j$ $(j = 1, \ldots, n)$ satisfies the condition (0). It is *complete*, *exact* if every such object satisfies the condition (1). K_a is *incomplete*, if at least object $o \in Ob^j$ satisfies the condition (>1); then the knowledge K_a is *inexact*, *imprecise*.

From the point of view of the agent, the information system Σ is *complete* if K_a is exact for any of its attributes a and it is *incomplete* if K_a is inexact, imprecise for a certain attribute a of this system. The discovered system Σ includes *gaps* from the agent's point of view if the knowledge about an attribute of Σ is empty.

If, from the agent's point of view, the system Σ is complete, then it is identified with Σ. If such a system is for him incomplete, then more than one information system corresponds to it (there can be many such systems). Each such system is obtained by assigning to each attribute the knowledge about which is incomplete for a certain object, one of its possible values for this object.

The system discovered by the agent can be represented in the form of data tables as a syntactical representation of the agent's knowledge. If such a system is complete, then every place in these tables indicating the value of an attribute of this system is filled in by the name of this value. If the system determines a situational reality this name will be a singular term, and if it determines a constituted reality this name will be most often a non-singular sharp name. If it is incomplete then in a certain data table a variable (a vague name) corresponding to an unknown quantity of the value of an attribute for a certain object appears. If such a system includes gaps with respect to the attribute, then in its table representation, in place of the names of the values of this attribute, at least one place is empty.

Example 3. Referring to the information system CON from Example 1 the agent's complete knowledge about the one-argument attributes LOCATION and POSITION can be represented in Table 1, and his knowledge about the two-argument attribute MUTUAL LOCATION – in Table 2.

Ob	LOCATION	POSITION
1	*conduc. podium*	*standing*
2	*stage*	*sitting*^{2}
3	*place for audience*	*sitting*^{3}

Table 1

MUTUAL LOCATION	1	2	3	*Ob*
1	n_{11}	f_{12}	b_{13}	
2	f_{21}	n_{22}	f_{23}	
3	f_{31}	f_{32}	n_{33}	
Ob				

Table 2

The knowledge about the three-argument attribute ACTIVITY can be represented by means of a data table in which all the possible combinations of triads of objects are indicated in three columns and all the possible values of this attribute could be indicated in 27 rows. For any triad of objects the knowledge about this attribute can be either empty (in the table there will be one or more than one respective empty places) or incomplete (in the table there will appear at least one variable, e.g. instead of the name of the value v_{223} the variable representing the names of values v_{223} and v_{222} will appear).

Similarly, the knowledge about the attributes of the system NYPO from Example 2 can be represented in data tables. Fragments of such tables are shown by Table 3 and Table 4.

Ob	PLAYING INSTRT	PART
o_1	*piano*	*regul. member*
o_2	*harp*	*regul. member*
o_3	*violin I*	*concert master*
o_4	*violin I*	*associate concert master*
o_5	*violin II*	*regul. member*
o_6	*violin I*	*regul. member*
⋮	⋮	⋮
o_{90}	*percussion*	*regul. member*

Table 3

DEP	o_1	o_2	o_3	o_4	o_5	o_6	...	o_{90}
o_1	*id*	*ind*	*ind*	*ind*	*ind*	*ind*	...	*ind*
o_2	*ind*	*id*	*ind*	*ind*	*ind*	*ind*	...	*ind*
o_3	*ind*	*ind*	*id*	*ld*	*ld*	*ld*	...	*ind*
o_4	*ind*	*ind*	*dep*	*id*	*ind*	*ind*	...	*ind*
o_5	*ind*	*ind*	*dep*	*ind*	*id*	*ind*	...	*ind*
o_6	*ind*	*ind*	*dep*	*ind*	*ind*	*id*	...	*ind*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
o_{90}	*ind*	*ind*	*ind*	*ind*	*ind*	*ind*	...	*id*

Table 4

4.2. Discovering unit information

For a complete information system, information about the reality $\mathfrak{R}(\Sigma)$ is *complete*: it is identical with $I\mathfrak{R}(\Sigma)$. Then each piece of unit information about the objects of the reality $\mathfrak{R}(\Sigma)$ is *exact*. For an incomplete information system or a system containing a gap there exists *inexact*, i.e. *imprecise*, or *empty* unit information, respectively.

When the knowledge K_a of an agent about the attribute $a \in A_j$ and its value for its object o is not defined yet, then indefinite unit information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$, determined by the attribute a and its unknown discovered value x for o (see (e)), information about objects $o_1, \ldots, o_j$, such that $o = (o_1, \ldots, o_j) \in Ob^j$, is obtained by solving the following *information equations* with two unknown quantities x and x':

1° for $S\mathcal{R}(\Sigma)$

(e_1) if $r_{a,a(o)} \in \mathcal{R}_1$, then for $o = r_{a,a(o)}$

$$\vec{r}_{a,x}(o) = x'$$

where x' is unknown quantity the range of which is the set

$$O^1 = \{r_{a,v}\} v \in V_a^o \subseteq Ob;$$

(e_{i+1}) if $r_{a,a(o)} \in \mathcal{R}_{i+1}$ and $o = (o_1, \ldots, o_{i+1})$, then for o_l, $l = 1, \ldots, i+1$

$$\vec{r}_{a,x}(o_l) = x' = (x_1, \ldots, x_{l+1}, x_{l+1}, \ldots, x_{i+1})$$

where x' is an ordered system of unknown quantities the range of which for any $j = 1, \ldots, l-1, l+1, \ldots, i+1$ are the nonempty sets

$$O_j = \{r^j_{a,v}\} v \in V_a^o, \text{ where } r^j_{a,v} \text{ is } j\text{-th element of } r_{a,v}$$

2° for $C\mathfrak{R}(\Sigma)$

(e_1) if $R_{a,a(o)} \in \mathcal{R}_1$, then for every $o \in R_{a,a(o)}$

$$\vec{R}_{a,x}(o) = x'$$

where x' is unknown quantity the range of which is the family

$$\mathcal{F}^1 = \{R_{a,v}\} v \in V_a^o \subseteq P(Ob);$$

(e_{i+1}) if $R_{a,a(o)} \in \mathcal{R}_{i+1}$ and $o = (o_1, \ldots, o_{i+1})$, then for all $o_l \in D_l(R_{a,a(o)})$, $l = 1, \ldots, i+1$

$$\vec{R}_{a,x}(o_l) = \{(x_1, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{i+1}) = x' : (x_1, \ldots, x_{l-1}, o_l, x_{l+1}, \ldots, x_{i+1}) \in R_{a,x}\}$$

where x' is an ordered system of unknown quantities the range of which for any $j = 1, \ldots, l-1, l+1, \ldots, i+1$ are the nonempty sets

$$\mathcal{F}_j = \{D_j(R_{a,v})\} v \in V_a^o, \text{ where } D_j(R_{a,v}) \text{ is } j\text{-th domain of } R_{a,v}.$$

4.3. Exact information

Unit information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ of an agent about the object o or o_l with respect to his knowledge about the attribute a and its value for this object is *exact*, when the equation (e_1) or (e_{i+1}) has exactly one solution, respectively, and for the range of the unknown quantity x' we have, respectively

$$O^1 = \{o\}, \quad \mathcal{F}^1 = \{R_{a,a(o)}\}, \quad O_j = \{o_j\}, \quad \mathcal{F}_j = \{D_j(R_{a,a(o)}\}$$

for $j = 1, \ldots, l-1, l+1, \ldots, i+1$.

It occurs if and only if equality (e) has exactly one solution and $V_a^o = \{a(o)\}$. Then the information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about the object o is identical with the objective information $\vec{r}_{a,a(o)}$ or $\vec{R}_{a,a(o)}$ contained in the given reality $S\mathfrak{R}(\Sigma)$ or $C\mathfrak{R}(\Sigma)$.

Exact information $\vec{r}_{a,v}$ or $\vec{R}_{a,v}$ assigned by relations and so by attributes and their values has sign carriers and can be represented by means of data tables (cf. Table 1 – Table 3 for information systems CON and NYPO).

If such information has representations in a natural language then the atomic sentences there are representations describing relations according to the number of arguments of suitable relations determining several pieces of unit information. They have the following form $s(1)$ or $s(i+1)$:

1° $s(1)$ for $S\mathfrak{R}(\Sigma)$

(a) $o\ \underbrace{\text{has (corresponds to) } a \text{ with the value } v}_{P}$

or

(b) $o\ \underbrace{\text{with respect to the property assigned by the value } v \text{ of the attribute } a \text{ is identified with itself}}_{P}$

(P is a one-argument predicate)

2° $s(1)$ for $C\mathfrak{R}(\Sigma)$

(a) $o\ \underbrace{\text{is an object which has } a \text{ with the value } v}_{P}$

or

(b) $\underbrace{o \text{ is an object with the property assigned through the value } v \text{ of the attribute } a.}_{P}$

(P is a predicate noun)

In a natural language, equality in the form $(i+1)$ defining the unit information about object o_l determined by the $i+1$-argument relation $r_{a,v} = (o_1, \dots, o_l, \dots, o_{i+1})$ and so by the $i+1$-argument attribute a and its value v, is represented by the *relative atomic sentence* $s(i+1)$ stating that

1° $s(i+1)$ for $S\mathfrak{R}(\Sigma)$

(a) o_l $\underbrace{\text{together with objects } o_1, \dots, o_{l-1}, o_{l+1}, \dots, o_{i+1} \text{ creates the system (relation) } r_{a,v} \text{ which has the attribute } a \text{ with the value } v.}_{R}$

or

(b) o_l $\underbrace{\text{is in the relation } r_{a,v} \text{ with objects}}_{R}$ $o_1, \dots, o_{l-1}, o_{l+1}, \dots, o_{i+1}$

(R is a $i+1$-argument predicate)

The equality $(i+1)$ defining the unit information about object $o_l \in D_l(R_{a,v})$, where $R_{a,v}$ is the relation determined by the $i+1$-argument attribute a and its value v is represented by the class of all the relative atomic sentences $s(i+1)$ stating that

2° $s(i+1)$ for $C\mathfrak{R}(\Sigma)$

(a) o_l is an object such that $\underbrace{\text{system } (o_1, \dots, o_{l-1}, o_l, o_{l+1}, \dots, o_{i+1}) \text{ has the attribute } a \text{ with the value } v}_{R}$

(R is a relative i-argument predicate noun)

or

(b) o_l $\underbrace{\text{is in the relation } R_{a,v} \text{ with objects}}_{R}$ $o_1, \dots, o_{l-1}, o_{l+1}, \dots, o_{i+1}$

(R is a $i+1$-argument predicate).

Example 4. And here we have several pieces of unit information and their representation for the reality $S\mathfrak{R}$(CON) from Example 1 and for the reality $C\mathfrak{R}$(NYPO) from Example 2 in the form of atomic sentences assuming that the object 2 – NYPO, belongs to the first domain of multi-ary relations.

for $S\mathfrak{R}$(CON)

Unit information	**Language representation**
$\vec{r}_{LOC.,c.podium}(2) = 2$	2 has LOCATION on *the conductor's podium*
$\vec{r}_{POS.,sitt.^2}(2) = 2$	2 is in *sitting*² POSITION
$\vec{r}_{MUT.LOC.,f_{21}}(2) = 1$	2 has (MUTUAL) LOCATION *facing* 1
$\vec{r}_{MUT.LOC.n_{22}}(2) = 2$	2 has (MUTUAL) LOCATION *neutral* to oneself (2)
$\vec{r}_{MUT.LOC.,f_{23}}(2) = 3$	2 has (MUTUAL) LOCATION *facing* 3
$\vec{r}_{ACT.,v_{222}}(2) = (2,2)$	2 is *playing one's* (2) *instruments for oneself* (2)
$\vec{r}_{ACT.,v_{213}}(2) = (1,3)$	2 *is playing under conduction of* 1 *for* 3
$\vec{r}_{ACT.,v_{223}}(2) = (2,3)$	2 *is playing one's* (2) *instruments for* 3

for $C\mathfrak{R}$(NYPO)

Unit information	**Language representation**
$\vec{R}_{PL.INSTR,piano}(o_1) = R_{PL.INSTR,piano}$	o_1 is A PIANIST in the NYPO
$\vec{R}_{PL.INSTR,viol.I}(o_6) = R_{PL.INSTR,viol.I}$	o_6 is A VIOLINIST I in NYPO
$\vec{R}_{PART,con.master.}(o_3) = R_{PART,con.master}$	o_3 is A CONCERT MASTER in the NYPO
If $o_3 \in D_1(R_{DEP.,ld})$, then $\vec{R}_{DEP.,ld}(o_3) = \{o: (o_3, o) \in R_{DEP.,ld}\}$	For every $o \in D_2(R_{DEP.,ld})$ each sentences of such type: o_3 is A LEADER OF A GROUP WHOSE MEMBER IS o

4.4. Empty information

Unit information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about an object o or an object o_l as an element of the ordered system $o = (o_1, \dots, o_{i+1})$ is *empty*, with respect to the knowledge of the agent about the attribute a and its value for this object, if the information equation (e_1) or (e_{i+1}), respectively, has no solution; then the range of the unknown quantity x' is empty (sets O^1, $\mathcal{F}^1$ and O_j, $\mathcal{F}_j$ for $j = 1, \dots, l-1, l+1, \dots, i+1$ are empty). It holds if and only if the equation (e) of the agent's unknowing has no solution, i.e. $V_a^o = \emptyset$.

If the agent's unit information has a table representation and for the attribute a and the object o or o_l as an element of the system $(o_1, \dots, o_{i+1})$ the information about it with respect to this attribute is empty, in the data table, in the place corresponding to the value of a for o or o_l, a gap appears. If the pieces of unit information about the object o or o_l have representations in a natural language, then the representation of empty information is a sentence stating the lack of whichever data about the value of this attribute for o or o_l.

4.5. Imprecise information: incomplete and vague

Unit information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about an object o or an object o_l as an element of the ordered system $o = (o_1, \ldots, o_{i+1})$ is *imprecise*, i.e. *inexact*, with respect to the knowledge of the agent about the attribute a and its unknown value x for this object, if the information equation (e_1) or (e_{i+1}), respectively, has more than one solution; then the range of the unknown quality x' of the information equation (e_1) is the multielement set O^1 or $\mathcal{F}^1$ – the set of solutions of this information equation (e_1), and at least one of the sets O_j or $\mathcal{F}_j$ $(j = 1, \ldots, l-1, l+1, \ldots, i+1)$ creating the appropriate sets of solutions for information equation (e_{i+1}) is a multi-element set. The agent's imprecise unit information about o or o_l is called *incomplete* in case 1°, if it refers to the reality $S\mathcal{R}(\Sigma)$, and *vague* in case 2°, if it refers to the reality $C\mathcal{R}(\Sigma)$.

The counterparts of the equations (e_1) and (e_{i+1}) for imprecise information are the following relationships:

1° for $S\mathfrak{R}(\Sigma)$

(e_1') If $r_{a,a(o)} \in \mathcal{R}_1$, then

$$\vec{r}_{a,x}(o) \in O^1 \text{ for } o = r_{a,a(o)};$$

(e_{i+1}') If $r_{a,a(o)} \in \mathcal{R}_{i+1}$ and $o = (o_1, \ldots, o_{i+1})$, then for o_l, $l = 1, \ldots, i+1$

$$\vec{r}_{a,x}(o_l) \in O_1 \times \ldots \times O_{l-1} \times O_{l+1} \times \ldots \times O_{i+1}$$

where $\emptyset \neq O_j = \{r^j_{a,v}\} v \in V^o_a \subseteq Ob, \ j = 1, \ldots, l-1, l+1, \ldots, i+1$

2° for $C\mathfrak{R}(\Sigma)$

(e_1') If $R_{a,a(o)} \in \mathcal{R}_1$, then

$$\vec{R}_{a,x}(o) \in \mathcal{F}^1 \text{ for any } o \in R_{a,a(o)};$$

(e_{i+1}') If $R_{a,a(o)} \in \mathcal{R}_{i+1}$ and $o = (o_1, \ldots, o_{i+1})$, then for all $o_l \in D_l(R_{a,a(o)})$, $l = 1, \ldots, i+1$

$$\vec{R}_{a,x}(o_l) \in \mathcal{F}_1 \times \ldots \times \mathcal{F}_{l-1} \times \mathcal{F}_{l+1} \times \ldots \times \mathcal{F}_{i+1}$$

where $\emptyset \neq \mathcal{F}_j = \{D_j(R_{a,v})\} v \in V^o_a \subseteq P(Ob)$, $j = 1, \ldots, l-1, l+1, \ldots, i+1$

The above relationships hold if and only if equation (e) of the agent's unknowing about the attribute a has more than one solution, i.e. the condition (> 1) holds and $|V^o_a| > 1$. For every solution of the unknowing equation (e) and so for every possible value of the attribute a for the object o,

from the point of view of the agent, in cases (e_1') there exists one unit information about object o; then in the agent's estimation indefinite information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about the object o corresponds to as many possible unit information about o as many possible values are in the set V^o_a. There is also the same number of elements of the range O^1 or $\mathcal{F}^1$ of the unknown quantity x' in (e_1).

In the case of information equations (e_{i+1}) the unknown quantity x' is a properly ordered system of unknown quantities each of which has a range of cardinality less then or equal to the the cardinality of the set V^o_a. Thus the possible unit information about the object o_l as an element of the system $o = (o_1, \ldots, o_{i+1})$ is no more than $|V^o_a|^i$ (i-power of the cardinality of the set V^o_a).

4.6. The degree of inexactness of information

If information is imprecise with respect to the agent's knowledge about the attribute a for the object o or o_l, then the *risk of error*, and so the *degree of inexactness* of the agent's information about objects of reality, depends on the cardinality of the scope of the unknown value x of the attribute a for the object o or o_l, i.e. on the set V^o_a. The degree of inexactness of information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about the object o can be assigned by means of the accuracy measure α which we call the *coefficient of exactness of the agent's information*:

1° for $S\mathfrak{R}(\Sigma)$

(α_1) $$\alpha(\vec{r}_{a,x}(o)) = \frac{1}{|O^1|} \quad (|O^1| = |V^o_a|)$$

(α_{i+1}) $$\alpha(\vec{r}_{a,x}(o_l)) = \frac{1}{|O_1| \circ \ldots |O_{l-1}| \circ |O_{l+1}| \circ \ldots \circ |O_{i+1}|} \leq \frac{1}{|V^o_a|^i};$$

2° for $C\mathfrak{R}(\Sigma)$

(α_1) $$\alpha(\vec{R}_{a,x}(o)) = \frac{1}{|\mathcal{F}^1|} \quad (|\mathcal{F}^1| = |V^o_a|)$$

(α_{i+1}) $$\alpha(\vec{R}_{a,x}(o_l)) = \frac{1}{|\mathcal{F}_1| \circ \ldots |\mathcal{F}_{l-1}| \circ |\mathcal{F}_{l+1}| \circ \ldots \circ |\mathcal{F}_{i+1}|} \leq \frac{1}{|V^o_a|^i};$$

$(0 < \alpha \leq 1)$.

4.7. Representation of inexact information

Inexact information of the agent about the object o determined by the unary relation $r_{a,x}$ or $R_{a,x}$ is linguistically represented as a sentential function – given in the below form and stating that:

$sf(1)$

1° for $S\mathfrak{R}(\Sigma)$

(a) o has (corresponds to) a with the variable value x

or

(b) o – with respect to indefinite property assigned by the attribute a and its unknown value x – is identified with x';

2° for $C\mathfrak{R}(\Sigma)$

(a) o is an object x' which has the attribute a with the variable value x

or

(b) o is an object which with respect to indefinite property assigned by the attribute a and its unknown value x is an object x'.

The values of the variable "x'" in the sentential function are the same as the values of the unknown quantity x' in the information equation (e_1). In case 1° the value of the variable "x'" is each object which could be identified with o with respect to the unknown value x of the attribute a for o. The variable "x'" represents singular terms of these objects which for the attribute a have one of the possible values for x. In a natural language, an indefinite pronoun or indefinite description corresponds to the variable "x'", and expression (b) is an *incomplete utterance.* In case 2° the value of the variable "x'" is any set of objects (thus each unary relation – each subset of the universe of reality) which from the agent's point of view has some possible value of the attribute a for the object o. Each such set belongs to the range of the variable "x'" and is the range of a sharp name representing this variable. In a natural language a vague name corresponds to this variable. The expression (b) can be regarded as a *vague utterance.* Hence, the range of a vague name contained in it – regarded as the variable "x'" – is a family of all extensions of sharp names which are represented by this variable-name. Such a family can be regarded as a generalized rough set in Pawlak's sense [5,7] (cf. U. Wybraniec-Skardowska [8-11]), the lower and upper approximations of which are the greatest lower bound and the least upper bound, respectively.

The inexact information $\vec{r}_{a,x}$ or $\vec{R}_{a,x}$ about the object o_l as an element of the system $o = (o_1, \ldots, o_{i+1})$, determined by $i{+}1$-ary relation assigning by $i+1$-argument attribute a and an indefinite value x, is represented in language as a sentential function corresponding to indefinite, indetermined relative sentences or a class of such sentences; such sentences describe a $i+1$-ary relation between object o_l and i indefinite objects. And thus we have

$sf(i+1)$

1° for $S\mathfrak{R}(\Sigma)$

(a) o_l together with the indefinte system $x' = (x_1, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{i+1})$ of objects gives an object $o' = (x_1, \ldots, x_{l-1}, o_l, x_{l+1}, \ldots, x_{i+1})$ which has (coresponds to) a with the indefinite value x

or

(b) o_l with respect to the unknown value x of the attribute a for some objects of the form $o' = (x_1, \ldots, x_{l-1}, o_l, x_{l+1}, \ldots, x_{i+1})$ remains in relation to the indefinite objects $x_1, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{i+1}$ creating system o'.

2° for $C\mathfrak{R}(\Sigma)$

(a) o_l is such the object that the attribute a for an indefinite object $o' = (x_1, \ldots, x_{l-1}, o_l, x_{l+1}, \ldots, x_{i+1})$ has the indefinite value x

or

(b) o_l remains in relation to an indefinite system of objects $x' = (x_1, \ldots, x_{l-1}, x_{l+1}, \ldots, x_{i+1})$ such that x is the indefinite value of the attribute a for $o' = (x_1, \ldots, x_{l-1}, o_l, x_{l+1}, \ldots, x_{i+1})$.

The values of the variable "x'" in these sentential functions are values of the unknown quantity x' of the information equation (e_{i+1}). The variables "x_1", ..., "x_{l-1}", "x_{l+1}", ..., "x_{i+1}" represent singular terms of objects whose appropriate system with the object o_l takes on a possible value of the variable "x'". In colloquial language, indefinite pronouns or indefinite descriptions correspond to these variables. In sentential functions $sf(i+1)$ there appear words containing i-argument relative names (predicate nouns). These words are regarded as variables representing sharp names. So that in case 1° the expression representing incomplete information is an *incomplete utterance* and in case 2° – a *vague utterance*, which is a carrier of vague information.

When we represent inexact information about the object o or o_l as the element $o = (o_1, \ldots, o_{i+1})$ with respect to its attribute a by means of a data table, we put the variable "x" in the place corresponding to the unknow value x of a for the object o or o_l. Its role is fulfilled by either an indefinite pronoun, in case 1°, or an indefinite description, in case 2°.

Example 5. The example of incomplete information about the object 1 (the conductor) of the reality $S\mathfrak{R}$(CON) from Example 1, with respect to the attribute LOCATION and its unknown value *SOMEWHERE ON THE*

PLATFORM with the range {*conductor's podium, stage*}, is the information represented by an incomplete utterance (in the form $sf(1)$ for 1°):

– The conductor has LOCATION *SOMEWHERE ON THE PLATFORM.*

or more exactly

– The conductor with respect to LOCATION and his indefinite value *SOMEWHERE ON THE PLATFORM* is identified with A PERSON LOCATED SOMEWHERE ON THE PLATFORM.

Ob	LOCATION	POSITION
1	*SOMEWHERE*	*standing*
2	*conduc. podium*	*sitting*^{2}
3	*place for audience*	*sitting*^{3}

Table 1'

In Table 1 (see Table 1') from Example 3, in the first row and first column, instead of the name "*conductor's podium*" the indefinite pronoun-variable "*SOMEWHERE (ON THE PLATFORM)*" occurs, representing the unknown value *SOMEWHERE ON THE PLATFORM* of the attribute LOCATION.

The following incomplete utterance (in the form $sf(1+2)$) for 1°

– NYPO is *playing SOMEBODY'S instruments for SOMEBODY*,

in which the distinguished indefinite pronoun-variable "*SOMEBODY'S*" has only one value: NYPO (2), and the variable "*SOMEBODY*" has two values: NYPO (2) and the audience (3), represents incomplete information with respect to the attribute ACTIVITY about the object NYPO (2) which together with two indefinite objects: *SOMEBODY'S* and *SOMEBODY* creates an ordered triad for which the value of this attribute is the unknown quantity: *playing SOMEBODY'S instruments for SOMEBODY* with the scope $\{v_{222}, v_{223}\}$.

Example 6. Let us consider the information system NYPO from Example 2 (see Table 3' and Table 4').

When, e.g. the knowledge about the attribute PLAYING INSTRUMENT referring to o_4 is incomplete, then in the fourth line and first column of the Table 3 (see Table 3') there appears a variable, let us say, "*A STRING INSTRUMENT*" ("x") representing possible sharp names of some string instruments, e.g.: violin I, violin II, violoncello, obviously from the point of view of the agent.

Ob	PLAYING INSTR.	PART
o_1	*piano*	*regul. member*
o_2	*harp*	*regul. member*
o_3	*violin I*	*IMPORT. INSTR.*
o_4	*A STRING INSTR*	*IMPORT. INSTR.*
o_5	*violin II*	*regul. member*
o_6	*violin I*	*regul. member*
⋮	⋮	⋮
o_{90}	*percussion*	*regul. member*

Table 3'

DEP	o_1	o_2	o_3	o_4	o_5	o_6	...	o_{90}
o_1	*id*	*ind*	*ind*	*ind*	*ind*	*ind*	...	*ind*
o_2	*ind*	*id*	*ind*	*ind*	*ind*	*ind*	...	*ind*
o_3	*ind*	*ind*	*id*	*ld*	*UNK*	*UNK*	...	*ind*
o_4	*ind*	*ind*	*dep*	*id*	*ind*	*ind*	...	*ind*
o_5	*ind*	*ind*	*UNK*	*ind*	*id*	*ind*	...	*ind*
o_6	*ind*	*ind*	*UNK*	*ind*	*ind*	*id*	...	*ind*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	...	⋮
o_{90}	*ind*	*ind*	*ind*	*ind*	*ind*	*ind*	...	*id*

Table 4'

The following vague utterance (in the form $sf(1)$ for 2°) expresses imprecise information about the member o_4 of NYPO with respect to the attribute PLAYING INSTRUMENT with indefinite values *A STRING INSTRUMENT* (x):

– o_4 is A MEMBER OF NYPO WHO PLAYS A STRING INSTRUMENT IN NYPO (x')

or

– o_4 is A STRING INSTRUMENTALIST IN NYPO (x').

Let us notice that the family of three sets which are extensions of the sharp names: "violinist I", "violinist II", "violoncellist" is the extension of the vague name distinguished in the second utterance and representing these sharp names.

If the agent possesses inexact information about the attribute PART for the objects o_3 and o_4, in the third and fourth rows and second column of Table 3 (see Table 3') a variable appears, e.g. "*AN IMPORTANT INSTRUMENTALIST*" ("x") whose possible values, from the point of view the agent, can be: *concert master*, *associate concert maser*. The following vague utterance (in the form $sf(1)$) corresponds to the vague information about for instance o_3:

– o_3 is AN IMPORTANT INSTRUMENTALIST IN NYPO (x').

Then the vague name occurring in this utterance represents two sharp names which are a definite description: "the concert master in NYPO" and "the associate concert master in NYPO".

If the agent's knowledge about the attribute DEPENDENCE for an ordered pair (o_3, x_2) is inexact, in the third line and in a certain column of Table 4 (see Table 4') the variable "*UNKNOWN*" ("x") can appear, for instance ranging over three terms of values: *independent*, *dependent*, *leading*. Then the following vague utterance (in the form $sf(1+1)$) concluding a one-argument relative vague name (a two-argument vague predicate) represents the vague information about the object o_3 with respect to the attribute DEPENDENCE and its indefinite values *UNKNOWN* (x):

– o_3 remains in *UNKNOWN* DEPENDENCE on *SOMEBODY* (x_2) in NYPO

or

– o_3 is a MEMBER OF NYPO WITH *UNKNOWN* DEPENDENCE (x) ON *SOMBODY* (x_2) IN NYPO.

The vague name "MEMBER OF NYPO WITH DEPENDENCE ON" distinguished in the second utterance represents three relative sharp names: "independent", "dependent", "leading" the extensions of which are, respectively, three sets of an ordered pair in the form (o_3, x_2), where x_2 – *SOMEBODY* – is an unknown quantity, values of which are members of NYPO such that o_3 together with them has the attribute DEPENDENCE with the value either *dependence* or *independence* or *leading*, respectively. And so the family compounded from three sets of ordered pairs of members of NYPO is the extension of this relative vague name. Each of these sets is a set of pairs which have a certain value of the attribute DEPENDENCE, which was mentioned above. The whole utterance determines the class of three vague utterances with the indefinite pronoun-variable "*SOMEBODY*". All such utterances represent relative atomic sentences.

5. Incomplete and vague utterances and processes of communication and reasoning; observations and conclusions

(i) In the proposed approach an incomplete utterance representing incomplete information about an object of reality $S\mathfrak{R}(\Sigma)$ is assumed to be a sentential function in which variables represent singular terms, while a vague utterance representing vague information about an object of the reality $C\mathfrak{R}(\Sigma)$ is assumed to be a sentential function in which vague names are treated as variables representing sharp names. In both cases, the ranges of these variables are established in advance by the agent.
(ii) Substituting in the classical sentential calculus laws sentential functions understood as incomplete or vague utterances (representing true or false logical sentences) for propositional variables, we obtain true expressions – satisfying all the laws of this calculus.
(iii) On the basis of premises which are incomplete or vague utterances with the same established ranges for equiform variables, correct logical reasoning can be done.

Example 7. An illustration of the above remarks can be the following correct reasoning referring to $C\mathfrak{R}$(NYPO):

o_4 is A STRING INSRUMENTALIST IN NYPO (x')
o_7 is A STRING INSTRUMENTALIST IN NYPO (x')

and so o_4 and o_7 are STRING INSTRUMENTALISTS IN NYPO (x's)

This reasoning runs according to the schemes

$o_4 \; \varepsilon \; x'$	p
$o_7 \; \varepsilon \; x'$	q
o_4 and o_7 are x's iff $o_4 \; \varepsilon \; x'$ and $o_7 \; \varepsilon \; x'$	$r \iff p \wedge q$
and so o_4 and o_7 are x's	r

(iv) The extensions of the variables occurring in incomplete and vague utterances can be variously established by a user of information. It is essential especially in the process of verbal communication where for a sender s and a receiver r the variables occurring in incomplete or vague utterances can have different values. The lack of common agreement concerning the extensions of the variables occurring in these utterances is, as we know, one of the main causes of misunderstanding between a sender and a receiver.
(v) When in the discussed utterances the variable "y" occurs whose range for a sender s is $\text{Range}^s(y)$ and for a receiver r – $\text{Range}^r(y)$, then the *degree*

of exactness of understanding between s and r is determined by the following coefficients:

$$\beta(y) = \frac{|\mathrm{Range}^s(y) \cap \mathrm{Range}^r(y)|}{|\mathrm{Range}^s(y)|}; \quad 0 \leq \beta(y) \leq 1,$$

$$\gamma(y) = \frac{|\mathrm{Range}^s(y) \cap \mathrm{Range}^r(y)|}{|\mathrm{Range}^r(y)|}; \quad 0 \leq \gamma(y) \leq 1,$$

$\beta(y)$ – the coefficient of understanding between s and r,
$\gamma(y)$ – the coefficient of understanding between r and s.

Form the point of view of a sender and a receiver there can occur known relations between the ranges of the variable "y":

(1) If $\mathrm{Range}^s(y) = \mathrm{Range}^r(y)$, then $\beta(y) = \gamma(y) = 1$, and a complete understanding between s and r takes place,

(2) If $\mathrm{Range}^s(y) \cap \mathrm{Range}^r(y) = \emptyset$, in the process of communication a complete misunderstanding between s and r holds; then $\beta(y) = \gamma(y) = 0$,

(3) If $\mathrm{Range}^s(y)$ and $\mathrm{Range}^r(y)$ remain in different range relations: the relation of overlapping ($0 < \beta(y), \gamma(y) < 1$), of subordination ($\beta(y) = 1$, $\gamma(y) < 1$), of superordination ($0 < \beta(y) < 1 = \gamma(y)$), there occurs misunderstanding between s and r the degree of inexactness of which is determined by appropriate coefficients of understanding.

(vi) When imprecise information about an object is expressed by means of sentential functions with a greater number of variables, for such variables we count coefficients β and γ. If for every such variable both of them are equal to 1 in the process of communication, understanding between s and r is achieved; when they are all equal to 0, we have to do with a complete misunderstanding. In the remaining cases we obtain an understanding or a partial misunderstanding, the degree of inexactness of which can be determined by appropriate coefficients of understanding dependent on all the unknown quantities of the transmited information.

(vii) The unification of the extension of the variable-vague name in the process of transmission of vague information or in the process of reasoning, when the premise of reasoning is a carrier of vague information of many agents, can be achieved by its approximation (see Z. Pawlak [5,7]) – i.e. in our approach by establishing the greatest lower bound and the least upper bound (with respect to inclusion) of this extension, treated as a rough set and at the same time as a family of extensions of sharp names which can be represented by a variable. Then the degree of exactness of vague information about the object, when it is expressed by means of the vague

name "y", can be determined by the following *coefficient $\delta(y)$ of exactness of approximation*:

$$\delta(y) = \frac{|\underline{\mathrm{Range}}(y)|}{|\overline{\mathrm{Range}}(y)|}; \quad 0 \leq \delta(y) \leq 1,$$

where $\underline{\mathrm{Range}}(y)$ is the greatest lower bound of the family Range(y) which is being the range y, while $\overline{\mathrm{Range}}(y)$ is the least upper bound of this family.

Acknowledgment

I am grateful to Professors Zdzisław Pawlak and Andrzej Skowron for discussions on the subject.

References

[1] Codd, E. F.: A Relational Models of Data for Large Shared Data Banks, *Comm. ACM, June 1970*, 13, 6, pp. 377-387.

[2] Marek W., Pawlak Z.: Information Storage and Retrival Systems – Mathematical Foundations, *Theoretical Computer Science* vol. 1 (1976), pp. 331-354.

[3] Pawlak Z.: Mathematical Foundations of Information Retrieval. *CC PAS Report* (1973), 101, Warsaw.

[4] Pawlak, Z.: Information Systems – Theoretical Foundations, *Information Systems*, vol. 6, no. 3 (1981), pp. 205-218.

[5] Pawlak, Z.: Rough Sets, *International Journal of Computer Sciences*, vol. 11 (1982), pp. 341-356.

[6] Pawlak, Z.: Information System – Theoretical Foundations *(the book in Polish)*, PWN, Warsaw 1982.

[7] Pawlak, Z.: *Rough Sets – Theoretical Aspects of Reasoning about Data*, **Kluwer Academic Publishers**, Dordrecht 1991.

[8] Wybraniec-Skardowska, U.: A Logical Explication of The Concept of Incomplete and Uncertain Information, in: (Alagar V. S., S. Bergler, F. Q. Dang; eds.), *Incompletness and Uncertainty in Information Systems*, pp. 180-188, **Springer-Verlag**, London 1994.

[9] Wybraniec-Skardowska, U.: Status of Rough Information and The Problem of Vagueness (in Polish), in: (W. Omyła; ed.) *Nauka i Język*, pp. 420-442, **Biblioteka Myśli Semiotycznej** (J. Pelc; ed.), Warsaw 1995.

[10] Wybraniec-Skardowska, U.: O konceptualizacji wiedzy nieostrej, *Filozofia Nauki*, Rok IV, Nr 3 (15), (1996), 45-62.

[11] Wybraniec-Skardowska, U.: Logic in View of Imprecision, in: Semiotics around the World: Sythesis in Diversity. *Proceedings of The Fith Congress of The International Association for Semiotic Studies*, Section Computer/AI, *June 12-18*, 1994, *Berkeley, California, USA*; **Mouton de Gruyter**, Berlin-New York,1997, 817-819.

Urszula Wybraniec-Skardowska
Department of Psychology
University of Opole
e-mail: uws@uni.opole.pl