Kazimierz Trzęsicki
University of Białystok

POLISH LOGICIANS' CONTRIBUTION TO THE WORLD'S INFORMATICS*

The position of Polish informatics, both in research and teaching, in the world of informatics, has its roots in the achievements of the Polish mathematicians of the Warsaw School and the logicians of the Lvov-Warsaw School.
Jan Łukasiewicz is the most famous Polish logician in the world of computer science. Invented by him, the parenthesis-free notation is known as PN (Polish Notation) and RPN (Reverse Polish Notation). Łukasiewicz created multi-valued logic as a separate subject. The idea of multi-valuedness is applied to hardware design (many-valued or fuzzy switching, analogue computer). A many-valued approach to vague notions and commonsense reasoning is the method of expert systems, databases and knowledge-based systems, as well as data and knowledge mining.
Stanisław Jaśkowski's system of natural deduction is the basis of systems regarding automatic deduction and theorem proving. He created a system of paraconsistent logic. Such logics are used in AI.
Kazimierz Ajdukiewicz, with his categorial grammar, participated in the development of formal grammars, the field significant for programming languages.
Andrzej Grzegorczyk has made an important contribution to the development of the theory of recursiveness.
Alfred Tarski, and the significance of his work for informatics, is not under consideration in the paper. His achievements are the subject of S. Feferman's article "Tarski's Influence on Computer Science".

Keywords: parenthesis-free notation, many-valued logic, paraconsistent logic, categorial grammar, theory of recursiveness, Polish notation, fuzzy switching, analogue computer, AI

When we hear of the successes achieved by Polish students of informatics at the Students' World Championships in Programming, or the winners of competitions for young scientists in the European Union, or the fact that Warsaw University occupies a top position in the world rankings of informatics studies; we have to ask ourselves, why this is so and in which place we should look for the causes of this success. Undoubtedly, their achievements, and this includes those of students from other Polish universities as well,

* I would like to thank the anonymous reviewer for the comments which made this article better. The work was supported by KBN (The Commission for Scientific Research) Grant 3 T11F 01130.

are their very own successes – the result of their talents, hard work and ambition. All this however, would matter little without good teachers, who being scientists themselves, contribute significantly to the development of informatics. We can be so bold as to say that informatics stemming from Warsaw University leads the world. While commenting on the fact that Warsaw University was listed in 2003 as one of the top universities (this according to a number of the most often quoted publications), professor Damian Niwiński (2003) points out that informatics has been developing systematically at Warsaw University since 1960. He sees the cause of this state of affairs in the attitudes of great Polish mathematicians,[1] the successors of the Polish school of mathematics: Kazimierz Kuratowski, Stanisław Mazur, Wacław Sierpiński, Hugo Steinhaus,[2] and Helena Rasiowa.[3] At the same time he emphasizes the role of logicians. It was mainly in this field that the critical mass of today's successes was created. It could have been a matter of chance in a sense, but when in 1948 the first Polish institution dealing with computers came into existence (the Mathematical Apparatus Group), professor Kuratowski appointed the logician and statistician Henryk Greniewski[4] (1930–1972) as its first director. It was Greniewski who initiated the establishment of the Polish Cybernetics Society[5] in 1962. The 23rd of December, 1948 however can be considered the launch date regarding the history of Polish informatics. Romuald W. Marczyński remembers that six people met on that very day in the mathematics seminar room in the Institute of Physics. Those present were Professor Kazimierz Kuratowski, Professor Andrzej Mostowski (logician), Doctor Henryk Greniewski and the three engineers Krystyn Bochenek, Leon Łukaszewicz and Marczyński himself. During the meeting they discussed the possibilities of

[1] There are two other important names worth mentioning in this context: professor Oskar Lange – economist, professor Janusz Groszkowski – director of the State Institute of Telecommunications, later deputy chairman of the State Council of the Polish People's Republic.

[2] At first he was deputy director of the Mathematical Apparatus Group dealing with applications. Later this post was held by Professor Stanisław Turski (1906–1986).

[3] Professor Rasiowa was very dedicated to publishing *Fundamenta Informaticae*. This journal started to appear in 1977 mainly thanks to her efforts. She was its Editor-in-Chief until her death. She never ceased to deal with it, even when she was ill. Let us add that she was an active member of the editorial board of *Studia Logica* (from 1974) and *Journal of Approximate Reasoning* (from 1986).

[4] He was expelled from the Planning Commission for political reasons, as a result of the growing class struggle.

[5] The name was ideologically conditioned. This is how informatics was referred to in the Soviet Union as well.

constructing mathematical apparatuses. It should be added that the first GAM-1 machine was built in 1950 by Zdzisław Pawlak, but was not used for calculations.

One of the achievements on a world scale was the language *KLIPA*, created in the 1960s by a team headed by Professor Władysław M. Turski: Marek Greniewski, Jadwiga Empacher, Jadwiga Zdanowska and Ryszard Solich. *KLIPA* was the external language for the *URAL* digital computer (Greniewski, Turski 1963). In the 1970s Andrzej Salwicki created the object-oriented programming language *LogLan.* A few years before dynamic logic was appreciated in the West, the team headed by Salwicki – Grażyna Mirkowska, Antoni Kreczmar and others – created algorithmic logic as a tool for examining and describing problems connected with the verification of programmes. Following Niwiński, one should also mention the works of Jerzy Tiuryn and his successors concerning the place of logic in informatics (type theory, the lambda calculation, functional programming, programming logic, the computing power of programming languages, complexity issues in logic and finite model theory). Tiuryn is presently in charge of a bio informatics team. Professor Jan Madey, head of the Section of Operating Systems in the Institute of Informatics, Warsaw University, director of the Centre for Open Multimedia Education (COME) at Warsaw University and the author of the first Polish handbooks of the languages *Algol 60* and *Pascal*, conducted innovative classes for students at various levels. He is the author of the system OS Kit designed to examine operating systems, problems of parallel computation and the efficiency of information systems. This world class specialist in the field of software engineering, who is probably best known for the methodology he developed in co-operation with David Lorge Parnas known in the literature as "Parnas-Madey Four Variable Model", finds a great deal of satisfaction in the achievements of his students. He rates their academic successes as his greatest success. His supervisees won the Academic World Championships in Team Programming and European Union Contests for Young Scientists (Szumiec-Presch 2004). He himself studied with professors Andrzej Kiełbasiński (doctor at the time), Karol Borsuk, Kazimierz Kuratowski, Stanisław Mazur and Andrzej Mostowski. He points out how important it was for him to enjoy the confidence of Professor Andrzej Turski, the rector of Warsaw University, and at the same time one of the people who made the greatest contributions to Polish informatics: "He threw me into the deep end of the pool", he said, "but was always there keeping an eye on me and supporting me from a distance". From 1964–70, Jan Madey was deputy head of the Section of Numerical Calculations at Warsaw University under Professor Stanisław Turski. Professor Stanisław Turski was a very important figure in the history of informatics at Warsaw University, who as its rector established the first computing centre (the Section of Numerical Calculations), and in 1975 also launched the Institute of Informatics within the department bearing a new name: *the Department of Mathematics, Informatics and Mechanics*. These institutional changes were connected with the launch of full informatics MA programmes (in place of studies offered within the section of numerical methods). Other areas of research in the Institute comprised automata theory (Stanisław Waligórski and others) and applied linguistics (Leonard Bolc, Janusz Bień and others), especially in connection with issues concerning artificial intelligence and programming in logic.

It is undoubtedly difficult to answer the question as to how important Polish logic was for achieving the aforementioned results. Many apparently irrelevant ideas may have significance in creating the right climate. Many enterprises, regardless of the intentions of their architects, may have an unexpected importance in other areas. As far as logic is concerned, however,

> ... the best known Pole in informatics is the logician Jan Łukasiewicz (1876–1956), who in 1917[6] introduced a way of expressing arithmetic expressions which avoids the use of parentheses, known as the Polish Notation. This notation is now commonly applied to automatic calculations of the value of expressions, used, among other things, in various calculators. (Madey, Sysło 2000)

This is the reason why solving the problem regarding the economisation of the notation, not in the least inspired by informatics issues, gave Łukasiewicz a lasting position in informatics.

Logic, alongside algorithmics, is a component of theoretical informatics.[7] In this sense all output by Polish logicians would be significant for informatics, and would render my initial question regarding their contribution to the world's informatics irrelevant. Therefore I will point out only those ideas which seem to be more directly connected with informatics, as was the case of the aforementioned Polish notation. Accordingly, I will not comment on the authors' intentions, nor will I ponder over the fact as to

---

[6] For more information (on this subject) see the next page.

[7] Electronics should also be added to contemporary informatics as a whole. Electronic solutions have turned out to be more effective than mechanical ones. Perhaps in the future electronics will be replaced by some biotechnological solutions.

whether those ideas were taken from their authors directly or indirectly or whether, as it was in the case of multi-valued logic, the same idea occurred independently to different scholars at the same time. Their importance for informatics will be discussed in as much as it is deemed necessary for their comprehension.

Polish Notation (Parenthesis-Free Notation)

The idea of the notation which avoids the use of parentheses appeared in connection with examining formal systems. Polish logicians, alongside other current issues, found the independence of the set of primitive terms and axioms equally important. Subsequently the problem of 'economisation' arose; in particular, a system with the smallest possible number of primitive terms and one shortest axiom was sought out.

From the point of view of semiotics (and informatics) – due to an economy of expression – it was interesting to find out whether a language without punctuation marks in general, and parentheses in particular, was possible. This kind of notation was invented by Jan Łukasiewicz. Łukasiewicz (1931, p. 165), who states that he laid down the principles of parenthesis-free symbolism in 1924, used it for the first time in his article *O znaczeniu i potrzebach logiki matematycznej* (On the importance and needs of mathematical logic) published in 1929, and although it was Chwistek who at the beginning of the 1920s talked about placing conjunctions before arguments, as Woleński (1985, p. 93) writes, there is more to parenthesis-free symbolism than writing conjunctions in front of arguments, hence there is no conflict in attributing the creation of parenthesis-free symbolism to Łukasiewicz and the idea of placing conjunctions in front of arguments to Chwistek.

It turned out that whenever all the conjunctions were *prefixes* (i.e. when they were written before their arguments) or when all of them were *suffixes* (i.e. written after their arguments), it was possible to eliminate the parentheses. Łukasiewicz's notation, apart from the economisation of means of expression, has an additional advantage in that the structure of an expression is defined by the position of symbols of which it is built. This very feature displays an advantage from the viewpoint of informatics (and not only informatics).

The importance of Łukasiewicz's notation for informatics was noted by Turing, who met Łukasiewicz in 1949. According to Turing, it is more advantageous for mechanical devices to have function symbols at the beginning of formulas. In informatics it is a suffixal notation that is particularly important. It was Hamblin who found a way of applying it. According to Pearcey (1994), Hamblin, who had gained some experience from radar services during World War II, was employed to run a third university computer in Australia in 1956. He became aware of the problems connected with (a) computing mathematical formulas containing parentheses and (b) loading memory with proper names of memory resources. As a formal logician he knew Łukasiewicz's work.[8] The solution to the first problem was supplied by Łukasiewicz's notation. Instead of writing, for example: $(a+b) \times c$, one can write: $\times, +abc$. The other problem, enabling the machine to access resources which do not require an address (a current operation would be always carried out on the results of the operations immediately preceding it, left and always remaining in the resources), was solved by applying Łukasiewicz's reverse notation (**R**everse **P**olish **N**otation – RPN). Instead of writing: $\times + abc$, one writes: $ab + c\times$). This is how the idea of organising resources into a stack was born – *last-in, first-out* (LIFO). Humblin presented his results at the First Australian Conference on Computing and Data Processing (1957). Representatives of the English Electric Company who were present at the conference carried his ideas to England and the company used Hamblin's architecture (and even his terminology) (Lavington 1980). Hamblin presented his conception in (1962) as well. One of the designers of the American computer B5000 (announced in 1961 and produced in 1963), in which RPN was used, R. S. Burton, wrote (1970) that the idea had occurred to him independently of Hamblin, when he was reading a handbook of symbolic logic. 10 years after Hamblin's first publication, the RPN idea was used by engineers from Hewlett-Packard in a calculator which appeared on the market in 1968 and then in HP-35 from 1972. In this way RPN became popular in scientific and engineering circles.[9] It is worth adding at this point that Hamblin was the precursor of many conceptions, among others, the application of temporal logic in informatics (Allen 1984, Allen 1985, Hamblin 1987, Williams 1985).

[8] Łukasiewicz's notation was used by A. N. Prior, a logician from New Zealand, among others in a handbook of logic (1955), which 'hindered' readability (see a comment on this in Woleński 1985, pp. 94–95).

[9] Edsger W. Dijkstra, converting infix notation into RPN, invented an algorithm which due to its similarity to the way a railroad shunting yard operates was called the 'shunting yard'.

Multi-Valued Logic

Jan Łukasiewicz (1878–1956) is best known for his concept of multi-valued logic.[10] Łukasiewicz was convinced not only that it was a discovery comparable to non-euclidian geometry, when he wrote (1930, p. 161):

> It is not easy to predict the impact of non-chrysippian[11] systems of logic upon philosophical speculation. It seems however that the philosophical importance of the systems presented here may be at least as great as the importance of non-euclidian systems of geometry.

Łukasiewicz designed his systems as the basis for mathematical research in arithmetic and multiplicity theory.[12] As far as practical application is concerned, having cybernetics in mind, he wrote in a letter to Lejewski in 1951 (Woleński 2005, p. 261):

> Multi-valued systems already today have important practical applications and may become a source of significant income.

One may agree with Woleński when he writes that:

> At present it is beyond any doubt that Łukasiewicz's expectations have not been fulfilled. Multi-valued logics have not revolutionised either logic or mathematics, or philosophy. (Woleński 1985, pp. 122–123)

It should be added however that the thesis concerning the practical benefits from multi-valued logics, and I do not mean those used in metatheoretical research but those in widely understood informatics, seems to have a chance to be confirmed. It is worth noting however that both multi-valued logics and the concept of the notation were not created for the sake of informatics. In the case of multi-valued logic it was philosophical motives. There is a monograph devoted to the use of multi-valued logics in informatics (Rine 1977). Interest in using multi-valued logics in informatics is reflected in conferences devoted to this issue. In 2006 the 36$^{\text{th}}$ annual symposium organised by *The Multi-Valued Logic Technical Committee of the IEEE Computer Society* will be held in Singapur.

---

[10] Independently of Łukasiewicz (1920*c*, 1920*b*, 1920*a*) multi-valued logics were created by E. Post (1921), born in Augustów, Poland.

[11] For more on the history of creating Łukasiewicz's multi-valued logics, see (Woleński 1985, pp. 115–122).

[12] This is how Łukasiewicz referred to multi-valued logics.

The Multi-Valued Logic – An International Journal (www.csi.uottawa.ca/∼ivan/mvl.html), among issues in its sphere of interest, mentions the following:

> MVL[13] and Soft Computing: neural networks, evolutionary computation, fuzzy systems, computational intelligence cost-effectiveness;
>
> Engineering aspects of MVL: circuit design, programmable logic, hardware and software verification, testing, analog and digital VLSI and ULSI, new concept devices and architectures, carrier computing (biocomputing, optical computing, ...);
>
> MVL and Automated Reasoning: machine learning, reasoning, theorem proving, expert systems;
>
> Computer Science and MVL: databases, massively parallel systems, collision-based computing;
>
> Fuzzy Logic and MVL: theoretical and practical aspects;
>
> Philosophical aspects of MVL

Ewa Orłowska from the Institute of Telecommunications is a member of the journal's editorial board.

One can distinguish between the applications of multi-valued logic in designing informatics equipment and in the methods of artificial intelligence.

Engineering Applications

To put it simply, just as multi-valued logics are a generalisation of two-valued logics, so are electrical circuits where m states a generalisation of circuits with two states. This issue has been addressed for a long time. Henryk Greniewski, who, as already mentioned, was the first director of the Mathematical Apparatus Group, was interested in the technical application of multi-valued logics. Let me add in passing that his book *Elementy cybernetyki systemem niematematycznym wyłożone* (*Cybernetics without Mathematics*) (1959) was translated into German, English and French and is still available from Pergamon Press (1960). In the German Democratic Republic Greniewski was an authority, among other things, in the field of applying cybernetics (informatics) in planning economic development (Segal 1999). His views on this issue have been quoted to this day (Greniewski 1962). There is at least one major publication in Polish on the subject of the application

[13] Multi-Valued Logic

of Łukasiewicz's logics. It is a two-volume work by Moisil (1966, 1967).[14] Epstein (1993) is a good introduction to the problem of multi-valued (fuzzy) switches.[15]

Currently the binary standard is obligatory in informatics. It was Leibniz who already in his day opted for this solution, but – and one has to remember that it was the era of mechanics – based his 'computer' on the decimal system. The architecture for contemporary computers was postulated by Von Neumann in the report 'First Draft of a Report on the EDVAC' (1981) written in 1945 under the auspices of the University of Pennsylvania and the United States Army Ordinance Department. In his justification for the choice of the binary system, similarly to Leibniz, he points to the simplicity of the system. The report states (Von Newman 1981):

> 5.1 ... Since these tube arrangements are to handle numbers by means of their digits, it is natural to use a system of arithmetic in which the digits are also two valued. This suggests the use of the binary system.
>
> 5.2. A consistent use of the binary system is also likely to simplify the operations of multiplication and division considerably. Specifically it does away with the decimal multiplication table. ... In other words: Binary arithmetic has a simpler and more one-piece logical structure than any other, particularly than the decimal one.

The report emphasises that:

> An important part of the machine is not arithmetical, but logical in nature. Now logics, being a yes-no system, is fundamentally binary. Therefore, a binary arrangement of the arithmetical organs contributes very significantly towards a more homogenous machine, which can be better integrated and is more efficient.

If designing computers operating with the decimal system can be explained in terms of it being natural, for other systems it is theoretical and practical arguments that are important. Such arguments can be found not only for the binary system but also for the ternary system.

The first designer of the machine operating in the ternary system was Thomas Fowler. In May 1840 he demonstrated his wooden calculating machine. It was described by De Morgan (1840, (1837–1843)).[16]

---

[14] His earlier work (1959) was also published in English by Pergamon Press. See also (1972).

[15] Cf. (Gottwald Winter 2004).

[16] The bibliography concerning this subject can be found at: http://www.mortati.com/glusker/fowler/refslinks.htm.

In the Soviet Union 50 ternary computers were built – *Setun* and *Setun 70*. The designer of these computers, Brousentsov, writes as a co-author (Brousentsov, Maslov, Ramil, Zhogolev 2005):

> It is known that the ternary arithmetic has essential advantages as compared with the binary one that is used in present-day computers. In connection with this Donald Knuth assumed that the replacement of "flip-flop" for "flip-flap-flop" one a "good" day will nevertheless happen [1].[17] Now, when the binary computers predominate, it is hard to believe in a reality of such assumption, but if it would happen not only the computer arithmetic, but the informatics on the whole would become most simple and most perfect. The third value (Aristotle named it σνμβεβηκος – attendant) what is very actual but hidden in binary logic, will become obvious and direct manipulated. Ternary logic has better accordance with the Nature and human informal thinking [2]. Unfortunately the modern researches of the multi-valued (non-binary) logic are formal and not associated with practical requests.
>
> A remarkable exclusion is the experience of creating the ternary computers "Setun" and "Setun 70" at Moscow State University [...]. This experience convincingly confirms practical preferences of ternary digital technique.
>
> 1. Knuth D. E. The art of computer programming. Vol. 2. Seminumerical algorithms. – Addison-Wesley, 1969.
> 2. Brousentsov N. P. Origins of informatics. – Moscow, The New Millenium Foundation, 1994. (In Russian).

Brousentsov points out in an interview (Rumyantsev 2004) the technical sources of the idea. He noticed however the importance of three-valued logic, stating that these issues were not well thought out in his computers.

The Analogue Computer[18]

Nowadays digital computers predominate. It seems that the concept of the analogue computer has finally been abandoned. However, it would not be for the first time in informatics that predictions have turned out to be wrong. Suffice to recall the prognosis concerning the number of computers necessary for the United States. Professor Jonathan W. Mills from Indiana University Bloomington[19] believes in the success of such machi-

[17] Let me add that for Knuth this must be a balanced system, which the system based on $\{-1, 0, 1\}$ supposedly is. More on the advantages of such a system in (Hayes 2001).

[18] It was Professor Witold Marciszewski who pointed out this application of Łukasiewicz's logic to me.

[19] For his conception of the analogue computer see (Mills 1993) and (Mills, Walker, Himebaugh 2003).

nes. He is not alone in his beliefs. In 1995 Lee Rubel[20] wrote to him (Mills 2006):

> The future of analog computing is unlimited. As a visionary, I see it eventually displacing digital computing, especially, in the beginning, in partial differential equations and as a model in neurobiology. It will take some decades for this to be done. In the meantime, it is a very rich and challenging field of investigation, although (or maybe because) it is not in the current fashion.

Mills became interested in the idea of the analogue computer back in 1990 in connection with his studies of Łukasiewicz's multi-valued logic. He collaborated at the time with J. Michael Dunn, professor of philosophy, and Oscar R. Ewing, professor of informatics. Together with Ch. Daffinger and M. G. Beavers he started designing electric circuits based on infinitely valued Łukasiewicz's logic. Mills considered this logic suitable for describing analogue circuits. The construction of the machine was inspired by Kirchhoff's research on electricity. Mills writes the following about his Kirchhoff-Łukasiewicz machine:

> I'm thinking that within five to ten years, we will find a niche in which these processors are superior, efficient, and cost-effective. (Hedger 2006)

He predicted that:

> We may develop sensors that would detect chemicals in the environment or toxins within our bodies, such as life-threatening cholesterol levels. We might develop an implant that could predict heart attacks – sort of a biological beeper. (Hedger 2006)

Applications in AI

Applications in AI seem to be the most promising of all the possible applications of multi-valued logics.

Multi-valued logics form the basis for the description of vague concepts, which are characteristic of natural language and non-formal reasoning. This has an importance, among others, for expert systems.

The most famous conception is the theory of fuzzy sets developed in the 1960s by Lofti A. Zadeh (1965). He applied Łukasiewicz's logic to elements

[20] The author, among others, of the article *The Extended Analog Computer* (1993), which prompted the invention of Kirchhoff-Łukasiewicz Machine.

of a set, thereby creating an algebra of fuzzy sets. They were not put to use until the mid 1970s, when Ebrahim H. Mamdani of Queen Mary College in London designed a 'fuzzy' controller for a steam engine.

A similar solution in connection with the research on expert systems was worked out in Poland by Z. Pawlak. Rough set theory was developed in many publications, for example (Pawlak 1982, Pawlak 1991, Pawlak 1993) and (Komorowski, Pawlak, Polkowski, Skowron 1999).

The theories of fuzzy and rough sets are applied in artificial intelligence and expert systems. They are used for the automation of data and knowledge exploration. In connection with the applications of multi-valued logic in informatics the notion of fuzzy logic is used (see, for example, http://plato.stanford.edu/entries/logic-menyvalued/).

Natural Deduction

In contemporary informatics, natural logic is applied first of all in broadly understood issues relating to artificial intelligence. It is the starting point of the basic systems of theorem proving and/or proof verification. It was created independently by Genzten (1934) and Jaśkowski (1906–1965). Jaśkowski testifies that in 1926 Łukasiewicz formulated the problem of a logical system which would be in line with the practice of proving mathematical theorems. Gentzen approached the issue in a similar way (1934, p. 176):

> My initial point of view was as follows: The formalisation of logical reasoning, especially in the way it was developed by Frege, Russell and Hilbert, differs significantly from the way of reasoning practised in mathematical proofs. One gets significant formal advantages in return. I would like therefore to present a formalism, which is as close to real-life reasoning as it is possible.[21]

Jaśkowski published the solution to the problem in 1934, creating an assumption-based system. The first announcement concerning this issue appeared as early as 1929, in *The Commemorative Book of the Polish Mathematical Symposium*, 1927. In this announcement, Jaśkowski wrote about his results, which had been presented at Łukasiewicz's seminar in 1926.

[21] Mein erster Gesichtspunkt war folgender: Die Formalisierung des logischen Schließens, wie sie insbesondere durch Frege, Russell und Hilbert entwickelt worden ist, entfernt sich ziemlich weit von der Art des Schließens, wie sie in Wirklichkeit bei mathematischen Beweisen geübt wird. Dafür worden beträchtliche formale Vorteile erzielt. Ich wollte nun zunächst einmal einen Formalismus aufstellen, der dem wirklichen Schließen möglichst nahe kommt.

Andrzej Trybulec, the author of the MIZAR system used for verifying mathematical proofs, directly refers to Jaśkowski's system (and the non-Fregian logic of Roman Suszko). Witold Marciszewski (1994, 2005) holds the view that Jaśkowski's system is more useful in computer-assisted proof verification, while Gentzen's system is better in computer-assisted proving. As far as the problems of the mechanisation of reasoning are concerned, it is worth noting the work of (Marciszewski, Murawski 1995).

Temporal Logic

Undoubtedly the founder of temporal logic is Arthur Norman Prior. One should note however the influence of the Lvov-Warsaw School and especially that of Łukasiewicz upon the development of Prior as a logician and upon his early temporal considerations.[22] Among the works that are mentioned as important for its founding, one should mention the work of Jerzy Łoś (1948). Prior (1996, p. 46) was astonished by the usefulness of temporal logic, learning from, among others, Dov Gabbay and Dana Scott that:

> There are practical gains to be had from this study too, for example in the representation of time-delay in computer circuits.

Presently, temporal logic is a recognised and important subject from the point of view of informatics.

Paraconsistent Logics

Jaśkowski formulated and designed discursive logic. His work "Rachunek zdań dla systemów dedukcyjnych sprzecznych" (Propositional calculus for contradictory deductive systems) (1948, 1969) was written as a response to a political need. Marxists rejected the principle of inconsistency. Therefore there appeared a need in logicians' circles for a work which would show the rationality of such a stance. The concerns of logic for paraconsistent systems, however, are more deeply rooted. Contradiction on classical logic grounds leads to trivialisation and its rejection is one of the oldest postulates of logic, clearly formulated, for example, by Aristotle. We know, however, that in 'real-world' cognitive activity, despite hidden or revealed

[22] I write more on this subject in (2005).

inconsistencies, the complete rejection of the system does not have to take place. The importance of paraconsistent systems reveals itself in informatics in connection with the fact that data a computer programme has to deal with may be, in a sense, contradictory. Man can somehow deal with the inconsistency of his views. Therefore artificial intelligence should also have this ability. A programme may draw data from various resources and hence collect contradictory data. Since man can cope with this, a computer should too. Man, in his activity, can give up reason, while a computer has to run according to a formal programme. If a programme is to be able to deal with an inconsistency, it has to be based on systems of paraconsistent logic. It was Stanisław Jaśkowski who created an important system of this kind, which is commonly recognised and well known.

Categorial Grammar

The idea of categorial grammar was formulated by Kazimierz Ajdukiewicz (1890–1963) in his work (1935). Undoubtedly informatics used other grammars for its purposes. What is important, however, is the very fact that this is a formal grammar. It can be used in various applications of informatics, especially in linguistics, but not only (Park 2001). The theory of categorial grammars is developed in connection with the Lambek Calculus. The work is also carried out in Poland. Among major world-ranking publications one should mention (Buszkowski, Marciszewski, van Benthem 1988).

The Theory of Recursive Functions

Andrzej Grzegorczyk[23] joined in on the mathematical milieu at the time when "the political situation was conducive to staying in the safe circle of logical and mathematical speculations". In 1950 he received his doctoral degree and was promoted by Andrzej Mostowski. Three years later, on the basis of his work *Some Classes of Recursive Functions* (1953), he was promoted to the position of 'docent'. This very work is an important historical contribution to world informatics. It is his most frequently cited work in the area of broadly understood issues of theoretical informatics. He showed his interest in the problems of decidability in his works *Zagadnienia rozstrzy-*

[23] My text devoted to A. Grzegorczyk is almost entirely based on the information taken from Stanisław Krajewski's work *Andrzej Grzegorczyk*, (2005).

galności (issues of decidability) (Grzegorczyk 1957, Grzegorczyk 1961). The problems regarding the concept of decidability, the concept of computability and the concept of recursive function, which arose in connection to Hilbert's programme and eventually resulted in the creation of the theoretical foundations of informatics, were taken up by Gödl, Church, Turing and Kleene. The work of Grzegorczyk contributed in a significant way to a better understanding of them. As Krajewski writes (2005, p. 109): "Throughout the whole period of his scientific academic activity he was faithful to the problems of decidability and computable functions." Let us add that for him it was (Krajewski 2005, p. 109) "connected with an in-depth study of concrete, empirical, 'tangible' aspects of the world, which are approached mathematically."

The contribution of Polish logicians to the problems of decidability and computability is much greater if one considers the achievements of Alfred Tarski, one of the most remarkable logicians. The influence of Alfred Tarski and the Polish logicians collaborating with him (A. Mostowski, L. Szczerba and others) goes beyond the scope of the problems of decidability and computability. This issue calls for separate treatment. Finally, let us add that a Polish presence in the world as far as these issues are concerned has found its expression in publications like that by Roman Murawski (1999).

Bibliography

(1929) 'Księga pamiątkowa pierwszego polskiego zjazdu matematycznego, 1927', Kraków.

Ajdukiewicz K. (1935), 'Die syntaktische Konnexität', *Studia Philosophica, Leopoli* 1,

Ajdukiewicz K. (1960), *Język i poznanie. Wybór pism z lat 1920–1939*, t. 1, PWN, Warszawa.

Allen J. F. (1984), 'Towards a general theory of action and time', *Artificial Intelligence* 23 (2), 123–154.

Allen J. F. (1985), 'Charles Hamblin (1922–1985)', *The Australian Computer Journal* 17 (4), 194–195.

Barton R. S. (1970), Ideas for computer systems organization: a personal survey, *w* J. S. Jou, red., 'Proceedings of the Third Symposium on Computer and Information Sciences held in Miami Beach, Florida, December 1969', t. 1 *Software Engineering*, Academic Press, New York, 7–16.

Brousentsov, N. P., Maslov, S. P., Ramil, A. J., Zhogolev, E. (2005), 'Development of ternary computers at Moscow State University', Internet.

Buszkowski W., Marciszewski W., van Benthem J. (1988), 'Categorial Grammar', John Benjamins Publishing Company.

De Morgan, A. (1837–1843), 'Description of a calculating machine, invented by Mr. Thomas Fowler of Torrington in Devonshire', *Abstracts of the Papers Printed in the Philosophical Transactions of the Royal Society of London 4* 243–244. Abstract (De Morgan 1840).

De Morgan, A. (1840), 'Description of a calculating machine, invented by Mr Thomas Fowler of Torrington in Devonshire', AP.23.24., London: The Royal Society.

Epstein, G. (1993), *Multiple-Valued Logic Design*, Institute of Physics Publishing, Bristol.

Genzten, G. (1934), 'Untersuchungen über das logische Schliessen', *Mathematische Zeitschrift* 39, 176–210, 405–431. Translation into english: "Investigation into logical deduction", 68–131 of *The Collected Papers of Gerhard Gentzen*, red. M. E. Szabo. North-Holland, Amsterdam, 1969 Translation into polish: Trzęsicki, K. *Badania nad wnioskowaniem logicznym*, Białystok 1980.

Gottwald, S. (Winter 2004), Manyvalued logic, *w* E. N. Zalta, red., 'The Stanford Encyclopedia of Philosophy'.
http://plato.stanford.edu/archives/win2004/entries/logic-manyvalued

Greniewski, H. (1959), *Elementy cybernetyki systemem niematematycznym wyłożone*, Warszawa.

Greniewski, H. (1960), *Cybernetics without Mathematics*, Pergamon Press, New York.

Greniewski, H. (1962), 'Logique et cybernétique de la planification', *Cahiers du Séminaire d'Econométrie*, ss. 115–164.
http://www.globalsecurity.org/wmd/library/report/2005/art137-teheran2005.htm.

Greniewski, H. (1966), *Kybernetische Systemtheorie ohne Mathematik*, Dietz, Berlin.

Greniewski, M., Turski, W. (1963), 'The external language KLIPA for the URAL-2 digital computer', *Communications of the Association for Computing Machinery* 6 (6), 322–324.

Grzegorczyk, A. (1953), *Some Classes of Recursive Functions*, IV *w* 'Rozprawy Matematyczne', Instytut Matematyczny PAN, Warszawa.

Grzegorczyk, A. (1957), *Zagadnienia rozstrzygalności*, PWN, Warszawa.

Grzegorczyk, A. (1961), *Fonctions Récursives*, Gauthier-Villars, Paris.

Hamblin, C. L. (1957), An addressless coding scheme based on mathematical notation, *w* 'Proceedings of the First Australian Conference on Computing and Data Processing', Salisbury, South Australia: Weapons Research Establishment.

Hamblin, C. L. (1962), 'Translation to and from Polish notation', *Computing Journal* 5, 210–213.

Hamblin, C. L. (1987), *Imperatives*, Basil Blackwell, Oxford.

Hayes, B. (2001), 'Third base', *American Scientist* 89(6), 490-494. http://www.americanscientist.org/content/AMSCI/AMSCI/Article-AltFormat/20035214317_146.pdf.

Hedger, L. (2006), 'Analog computation: Everything old is new again', http://www.indiana.edu/rcapub/v21n2/p01.html.

Jaśkowski, S. (1934), 'On the rules of supposition in formal logic', *Studia Logica* 1.

Jaśkowski, S. (1948), 'Rachunek zdań dla systemów dedukcyjnych sprzecznych', *Studia Societatis Scientiarun Torunesis* 1 (5), 55–77.

Jaśkowski, S. (1969), 'Propositional calculus for contradictory deductive systems', *Studia Logica* 24, 143–157. Translation into english (Jaśkowski 1948).

Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A. (1999), Rough sets: A tutorial, *w* 'Rough Fuzzy Hybridization. A new Trend in Decision-Making', Springer Verlag, Singapore.

Krajewski, S. (2005), Andrzej Grzegorczyk, *w* W. Mackiewicz, red., 'Polska filozofia powojenna', t. 3, Agencja Wydawnicza Witmark, Warszawa, pp. 99–118.

Lavington, S. H. (1980), *Early British Computers: The Story of Vintage Computers and the People who Built Them*, Manchester University Press, Manchester.

Łoś, J. (1948), 'Podstawy analizy metodologicznej kanonów Milla (the foundations of the methodological analysis of Mill's canons)', *Annales Universitatis Mariae Curie-Skłodowska, Section FII (5)*, 269–301.

Łukasiewicz, J. (1920a), 'Logika trójwartościowa', *Ruch Filozoficzny* V, 166–171.

Łukasiewicz, J. (1920b), 'O pojęciu możliwości', *Ruch Filozoficzny* V.

Łukasiewicz, J. (1920c), 'On three-valued logic', *Ruch Filozoficzny* 5, 170–171. Translation into english in: Borkowski, L. (red.) 1970. Jan Łukasiewicz: Selected Works. Amsterdam: North Holland.

Łukasiewicz, J. (1929), 'O znaczeniu i potrzebach logiki matematycznej', *Nauka Polska* 10, 604–20.

Łukasiewicz, J. (1930), 'Philosophische Bemerkungen zu mehrwertigen Systemem des Aussgankalkül', *Sprawozdania Towarzystwa Naukowego Warszawskiego Wydział III 23*, 51–77.

Łukasiewicz, J. (1931), Uwagi o aksjomacie Nicoda i «dedukcji uogólniającej», *w* 'Księga Pamiątkowa Polskiego Towarzystwa Filozoficznego we Lwowie', Lwów.

Łukasiewicz, J. (1961), *Z zagadnień logiki i filozofii. Pisma wybrane*, Państwowe Wydawnictwo Naukowe, Warszawa. Choice of papers done by J. Słupecki.

Madey, J., Sysło, M. M. (2000), 'Początki informatyki w Polsce', *Informatyka* 9/10.

Marciszewski, W. (1994), A Jaśkowski-style system of computer-assisted logic, in J. Woleński, red., 'Philosophical Logic in Poland', Synthese Library, Kluwer, Dordrecht. Changed version: *A System of Suppositional Logic as Embodied in the Proof Checker Mizar MSE* http://www.calculemus.org/MathUniversalis/3/marc-jas.html.

Marciszewski, W. (2005), 'A system of suppositional logic as embodied in the proof checker Mizar MSE', http://www.calculemus. org/MathUniversalis/3/marc-jas.html.

Marciszewski, W., Murawski, R. (1995), *Mechanization of Reasoning in a Historical Perspective*, Rodopi, Amsterdam/Atlanta.

McCall, S., red. (1967), *Polish Logic in 1920–1939*, Clarendon Press, Oxford.

Mills, J. (1993), Lukasiewicz' insect: The role of continuous-valued logic in a mobile robot's sensors, control, and locomotion, *w* 'Proc. 23rd Int. Symp. On Multiple-Valued Logic', IEEE Computer Society.

Mills, J. W. (2006), 'Kirchhoff-Lukasiewicz machines', http://www.cs.india-na.edu/~jwmills/ANALOG.NOTEBOOK/klm/klm.html.

Mills, J., Walker, T., Himebaugh, B. (2003), 'Lukasiewicz' insect: Continuous-valued robotic control after ten years', *Int. Jour. Multiple-Valued Logic* 9 (2).

Moisil, G. C. (1959), *The Algebraic Theory of Switching Circuits (in Romanian)*, Bucharest. English translation: Pergamon Press 1969.

Moisil, G. C. (1966), *Zastosowanie algebr Łukasiewicza do teorii układów przekaźnikowo-stykowych*, t. 1, Wrocław – Warszawa – Kraków.

Moisil, G. C. (1967), *Zastosowanie algebr Łukasiewicza do teorii układów przekaźnikowo-stykowych*, t. 2, Wrocław – Warszawa – Kraków.

Moisil, G. C. (1972), *Essais sur les logiques non Chrysippiennes*, Editions de l'Acad. de la Rép. Soc. de Roumania, Bucharest.

Murawski, R. (1999), *Recursive Functions and Metamathematics. Problems of Completeness and Decidability, Gödel's Theorems*, Kluwer Academic Publishers, Dordrecht – Boston – London.

von Neuman, J. (1981), First draft of a report on the EDVAC, *w* N. Stern, red., 'From ENIAC to UNIVAC: An Appraisal of the Eckert-Mauchly Computers', Digital Press, Bedford, Massachusetts, ss. 177–246.

Niwiński, D. (2003), 'University of Warsaw', http://www.in-cites.com/institutions/UniversityofWarsaw.html.

Park, J. (2001), 'Using combinatory categorial grammar to extract biomedical information', *IEEE Intelligent Systems and Their Applications* 16, 62–67.

Pawlak, Z. (1982), 'Rough sets', *International Journal of Computer and Information Sciences* 11, 341–356.

Pawlak, Z. (1991), *Rough Sets – Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publisher, Dordrecht.

Pawlak, Z. (1993), 'Rough sets. Present state and the future', *Foundations of Computing and Decision Sciences* 18, 157–166.

Pearcey, T. (1994), Australian computing: The second generation, *w* J. M. Bennett, R. Broomham, P. M. Murton, T. Pearcey, R. W. Rutledge, red., 'Computing in Australia: The Development of a Profession', Australian Computer Society.

Post, E. (1921), 'Introduction to a general theory of elementary propositions', *American Journal of Mathematics* XLIII.

Prior, A. N. (1955), *Formal Logic*, Clarendon Press, Oxford. II wyd. 1962.

Prior, A. N. (1996), A statement of temporal realism, *w* B. J. Copeland, red., 'Logic and Reality: Essays on the Legacy of Arthur Prior', Oxford University Press.

Rescher, N. (1969), *Many-valued Logic*, New York.

Rine, D. C., red. (1977), *Computer Science and Multiple Valued Logic*, North-Holland Publ. Comp., Amsterdam. The second edition. 1984.

Rubel, L. A. (1993), 'The extended analog computer', *Adv. In Appl. Math.* 14, 39–50.

Rumyantsev, D. (2004), 'Interview s konstruktorom troicznoj ewm (interviews with the designer of the ternary computer)', *Upgrade* 33 (175).

Segal, J. (1999), L'introduction de la cybernétique en R.D.A. Rencontres avec l'idéologie marxiste, in 'Proceedings of the XXth International Congress of History of Science (Liege, 20–26 July 1997)', t. 1, pp. 67–80. Die Einführung der Kybernetik in der DDR. Begegnung mit der marxistischen Ideologie http://jerome-segal.de/Publis/Kyb-DDR.htm.

Szumiec-Presch, B. (2004), 'Madey: Najbardziej cieszą mnie sukcesy moich studentów i wychowanków', http://www.naukawpolsce.pap.pl/nauka/index.jsp?place=Lead20&news_cat_id=62&news_id=319&layout=1&page=text.

Trzęsicki, K. (2005), Arthura Normana Priora związki ze szkołą lwowsko-warszawską, in K. Trzęsicki, editor., 'Ratione et studio', Uniwersytet w Białymstoku, Białystok, pp. 269–288.

Williams, G. (1985), 'A shy blend of logic, maths and languages (obituary of Charles Hamblin)', Morning Herald, Sydney.

Woleński, J. (1985), *Filozoficzna szkoła lwowsko-warszawska*, PWN, Warszawa.

Woleński, J. (2005), Dzieje pewnego przypisu, *w* K. Trzęsicki, editor., 'Ratione et studio', Wydawnictwo Uniwersytetu w Białymstoku, Białystok, pp. 249–268.

Zadeh, L. (1965), 'Fuzzy sets', *Journal of Information and Control* 8, 338–353.