ASSOCIATION FOR AUTOMATED REASONING NEWSLETTER

No. 18 June 1991

From the AAR President, Larry Wos . . .

Our second newsletter for 1991 features two contributions from our readers. One is a "scholarly footnote" to an article that appeared in our last issue. The other is an article by Maria Paola Bonacina on Lukasiewicz logic. While this article is significantly longer than the usual contributions, we believe our readers will be interested in the problems she presents and will enjoy her challenge to seek other proofs and to answer open questions.

Conferences

TACS'91

The International Conference on THEORETICAL ASPECTS OF COMPUTER SOFTWARE (TACS'91) will take place on September 24–27, 1991, at Tohoku University, Sendai, Japan. The conference is sponsored by Tohoku University, Sendai, in cooperation with the Information Processing Society of Japan, the Japan Society for Software Science and Technology, the Association for Symbolic Logic, and ACM SIGACT; the cooperation of the IEEETC on Mathematical Foundations of Computing and ACM SIGPLAN is pending.

Presentations will include invited talks by Gordon D. Plotkin (Edinburgh University) on the completeness of type-checking, by Masahiko Sato (Tohoku University) on adding proof objects and inductive definition mechanisms to Frege structures, Albert R. Meyer (MIT Laboratory for Computer Science) on full abstraction and the context lemma, Robert L. Constable (Cornell University) on constructive aspects of classical logics, Amir Pnueli (Weizmann Institute of Science) on applying formal methods to software development–partially, Kazuhiro Fuchi (ICOT, Tokyo) on the role of logic programming in the Fifth Generation Computing project, Masami Hagiya (Kyoto University) on programming-by-example and proving-by-example, Jean-Louis Lassez (IBM Watson) on programming with constraints, Susumu Hayashi (Ryukoku University) on singleton union and intersection types in program extraction, and Dana S. Scott (Carnegie Mellon University) on replacing logicians by machines.

A proceedings, published as a volume in Springer-Verlag's Lecture Notes in Computer Science, will be given to registrants on arrival.

For further information, contact

Prof. Takayasu Ito
Department of Information Engineering
Tohoku University
Sendai, Japan 980
tacs91@ito.ecei.tohoku.ac.jp
FAX: 81 22 267 4404

Prof. Albert R. Meyer
MIT Laboratory for Computer Science
545 Technology Square, NE43-315
Cambridge, MA 02139, USA
tacs-request@theory.lcs.mit.edu
FAX: (617) 253 3480

2nd International Symposium on AI and Math

The Second International Symposium on Artificial Intelligence and Mathematics will be held at Fort Lauderdale, Florida, on January 5–8, 1992.

The symposium is the second of a biennial series featuring applications of mathematics in artificial intelligence, as well as artificial intelligence techniques and results in mathematics. There has always been a strong relationship between the two disciplines; however, the contact between practitioners of each has been limited, partly by the lack of a forum in which the relationship could grow and flourish. This symposium represents a step towards improving contacts and promoting cross-fertilization between the two areas.

Authors are invited to submit 5 copies of extended abstracts (up to 10 double-spaced pages) by July 31, 1991, to J.-L. Lassez, IBM T. J. Watson Research Center, H1-A12, P.O. Box 704, Yorktown Heights, NY 10598. Authors will be notified of acceptance on Sept. 10. Authors will be invited to submit within one month after the symposium a final full-length version of their paper, to be considered for inclusion in a refereed volume of the series Annals of Mathematics and Artificial Intelligence, J. C. Blatzer Scientific Publishing Co.

The symposium is sponsored by Florida Atlantic University and IJCAI. Partial travel subsidies may be available to young researchers.

International Conference on Fifth Generation Computer Systems 1992

The Fifth Generation Computer Systems (FGCS) project was started in 1982 to make a revolutionary new type of computer oriented to knowledge information processing in the 1990s The project is now in the final three-year stage of its ten-year duration. The purpose of the Internation Conference on Fifth Generation Computer Systems 1992 is to present the final results of the FGCS project, as well as to promote the exchange of new ideas in the fields of knowledge information processing, logic programming, and parallel processing. The conference will take place June 1–5, 1992, in Tokyo. The first two days will be devoted to the presentation of results of the FGCS project, as well as to invited lectures by leading researchers. The remaining three days will be devoted to technical sessions for invited and contributed papers and panel discussions. Throughout, a number of demonstrations will be given using the parallel inference machine prototype system now being completed.

Authors are invited to submit papers to Professor Hozumi Tanaka, FGCS'92 Program Chairperson, ICOT, Mita Kokusai Bldg. 21F, 4-28 Mita 1-chome, Minato-ku, Tokyo 108, Japan. The papers are restricted to 16 pages (1-1/2 spaced or 12/18), including figures and a 150–200 word abstract. The papers must be received by October 1, 1991; they must be written (and presented) in English.

Program areas include Foundations (e.g., automated reasoning; knowledge representation; theoretical foundations of logic programming and databases; and concurrent, distributed, and parallel computation); Architectures (e.g., parallel implementation techniques, performance modeling, and inference machines); Software (e.g., logic programming, deductive databases, reasoning about programs, and programming environments); and Applications and Social Impacts (e.g., robotics, knowledge-base systems, natural language, and database applications).

CADE-11

The 11th Conference on Automated Deduction (CADE) will be held at Saratoga Springs, New York, on June 15–18, 1992. It will be hosted by the State University of New York at Albany.

CADE is the major research forum covering all aspects of automated deduction. Original papers in automated deduction (for nonclassical as well as classical logics) are invited; specific topics of interest include (but are not limited to) the following:

Applications	Induction	Program Synthesis
Commonsense Reasoning	Inference Systems	Rewrite Rules
Deductive Databases	Logic Programming	Theorem Proving
Decision Procedures	Program Verification	Unification Theory

Original research papers, descriptions of working reasoning systems, and problem sets that provide innovative, challenging tests for automated reasoning systems, are solicited. Research papers should not exceed 7,000 words (15 proceedings pages, 5.5 by 8 inches, 11-point type, will be allotted). System descriptions and problem sets should be limited to 5 proceedings pages.

Authors should send 6 copies of their submission to the Program Chair Deepak Kapur (kapur@cs.albany.edu) at the Conference address. The title page of the submission should include author's name, address, phone number, and e-mail address. Papers must be unpublished and not submitted for publication elsewhere. Submissions that are late or too long or that require major revision will not be considered.

Submission deadline:	Nov. 8, 1991
Notification of acceptance:	Feb. 1, 1992
Camera-ready copy due:	March 16, 1992

Further information about the conference may be obtained from the Local Arrangements Chair Neil V. Murray (nvm@cs.albany.edu) at the Conference address.

Conference Address: CADE-11
Institute for Programming and Logics
Department of Computer Science LI67A
University at Albany - SUNY
Albany, NY 12222

The program committee is as follows:

Peter Andrews	Larry Henschen	William McCune
Wolfgang Bibel	Deepak Kapur	Grigori Mints
W. W. Bledsoe	Claude Kirchner	David Musser
Robert S. Boyer	Kurt Konolige	Hans-Juergen Ohlbach
Alan Bundy	Jean-Louis Lassez	David Plaisted
Edmund Clarke	Vladimir Lifschitz	Joerg Siekmann
Robert Constable	Donald Loveland	John Slaney
Ryuzo Hasegawa	Ewing Lusk	Mark Stickel

The Halting Problem: A Scholarly Footnote

Leslie Burkholder, CDEC, Carnegie Mellon University
(burkholder@andrew.cmu.edu)

Massimo Bruschi's recent article providing a mechanical proof of the unsolvability of the halting problem [2] allows me to correct a scholarly error and point to a relevant citation. The error concerns where the hand-produced proof of the result appeared. It was not in the *SIGCSE Bulletin* but in [3]. The relevant citation is to [1]. There, Woody Bledsoe and Richard Hodges give the formalization of the halting problem from [4] as an example of how complicated a problem for automated deduction might get.

References

[1] W. Bledsoe and R. Hodges, "A survey of automated deduction," in H. E. Shrobe and AAAI (eds.), *Exploring Artificial Intelligence* (Morgan Kaufmann: San Mateo, 1988).

[2] M. Bruschi, "The halting problem," *AAR Newsletter* 17, March 1991.

[3] L. Burkholder, "The halting problem," *SIGACT News,* vol. 18, no. 3 (Spring 1987).

[4] L. Burkholder, "A 76th automated theorem proving problem," *AAR Newsletter* 8, April 1987.

Problems in Lukasiewicz Logic

Maria Paola Bonacina
(bonacina@sbcs.sunysb.edu)

In the November 1990 issue of the *AAR Newsletter* [15], Larry Wos described a problem in Lukasiewicz logic as a challenge problem for theorem provers. This note is intended to provide additional information to anyone interested in attacking the problem with an automated prover. We present three problems in Lukasiewicz logic and the results obtained so far in proving them automatically.

Lukasiewicz logic is many-valued propositional logic, that is, logic with n truth values. This is actually a family of logics $L_1, L_2, \ldots, L_n, \ldots, L_{\aleph_0}$, where L_n, $n \geq 1$, is propositional logic with n truth values. $L_{\aleph_0}$ has infinitely many truth values. A logic L_n, $n \geq 1$, is the set of all sentences satisfied by the structure $L_n =< A_n, g, f >$ with domain

$A_n = \{\frac{k}{n-1} | 0 \leq k \leq n-1\}$

and two functions

$g : A_n \rightarrow A_n$, $g(x) = 1 - x$ and

$f : A_n \times A_n \rightarrow A_n$, $f(x, y) = min(1 - x + y, 1)$,

where $-$, $+$ and min are subtraction, addition and minimum on the rational numbers. The domain A_n is the set of truth values of the logic. The function g gives the complement of its argument with respect to 1, while $f(x, y)$ adds the complement of x to y, truncating it to 1 if it exceeds 1. L_2 is the classical two-valued propositional logic, with domain $A_2 = \{0, 1\}$, $g(x)$ is negation and $f(x, y)$ is implication. The functions $g(x)$ and $f(x, y)$ are a form of generalized negation and generalized implication, respectively, that specialize to the classical connectives only if $n = 2$. As n increases, the domain A_n grows, whereas the set L_n of true sentences becomes smaller and smaller. At the limit, the logic $L_{\aleph_0}$ has the interval $[0, 1]$ of rational numbers as its set of truth values.

Lukasiewicz conjectured that the following axioms, together with modus ponens, are an axiomatization for $L_{\aleph_0}$:

1. $p \Rightarrow (q \Rightarrow p)$
2. $(p \Rightarrow q) \Rightarrow ((q \Rightarrow r) \Rightarrow (p \Rightarrow r))$
3. $((p \Rightarrow q) \Rightarrow q) \Rightarrow ((q \Rightarrow p) \Rightarrow p)$
4. $(not(p) \Rightarrow not(q)) \Rightarrow (q \Rightarrow p)$
5. $((p \Rightarrow q) \Rightarrow (q \Rightarrow p)) \Rightarrow (q \Rightarrow p)$

where *not* and $\Rightarrow$ are interpreted as g and f in the model $L_{\aleph_0}$. These axioms and the original description of many-valued logic can be found in [14]. A more recent treatment is in [11]. The conjecture that the above axioms with modus ponens are an axiomatization of $L_{\aleph_0}$ has been proved first by Wajsberg, then independently by Rose and Rosser in [13] and by Chang in [4].

First Problem: Dependency of the Fifth Axiom (original presentation)

The first challenge problem is **to derive the fifth axiom in the above list from the other four.** This result has been proved independently by Meredith [9] and Chang [5]. It has been called, somewhat imprecisely, "Fifth Lukasiewicz Conjecture" in [1] and thus in [15]; we shall rather call it "Dependency of the Fifth Axiom." To our knowledge, no automated proof of this problem in this formulation has been obtained so far. Extensive experimentation with OTTER [8] has been and is currently being conducted at the Argonne National Laboratory.

Second Problem: Dependency of the Fifth Axiom (equational presentation)

Lukasiewicz logic is related to several families of algebra: the *MV-algebras*, introduced by Chang in [4] to prove Lukasiewicz's conjecture about the axiomatization of $L_{\aleph_0}$; the *AFC^*-algebras* (approximately finite dimensional C^*-algebras), with applications in quantum mechanics [10] and the *Wajsberg algebras* [6, 12]. The problem of proving the dependency of the fifth axiom can be reformulated as an equational problem in Wajsberg algebras. The following set of equations, which we call $\mathcal{W}$, is the axiomatization of Wajsberg algebras [6]:

1. $true \Rightarrow x == x$
2. $(x \Rightarrow y) \Rightarrow ((y \Rightarrow z) \Rightarrow (x \Rightarrow z)) == true$
3. $(x \Rightarrow y) \Rightarrow y == (y \Rightarrow x) \Rightarrow x$
4. $(not(x) \Rightarrow not(y)) \Rightarrow (y \Rightarrow x) == true$

Axioms 2, 3, and 4 correspond to axioms 2, 3, and 4 in the original presentation of $L_{\aleph_0}$ by Lukasiewicz. The fifth axiom

$$((p \Rightarrow q) \Rightarrow (q \Rightarrow p)) \Rightarrow (q \Rightarrow p) == true$$

can be written more conveniently as

$$(x \Rightarrow y) \vee (y \Rightarrow x) == true$$

by introducing the connective $\vee$ defined as

$$x \vee y == (x \Rightarrow y) \Rightarrow y.$$

Thus the problem is **to derive** $(x \Rightarrow y) \vee (y \Rightarrow x) == true$ **from** $\mathcal{W}$. The operation $x \vee y$ is interpreted as $max(x, y)$: if we replace $\Rightarrow$ by its interpretation, we get

$$min(1 - min(1 - x + y, 1) + y, 1) = \begin{cases} min(1 - 1 + y, 1) = y \ if \ y \geq x \\ min(1 - 1 + x - y + y, 1) = x \ if \ x \geq y \end{cases}$$

i.e. $max(x, y)$. A dual operator $\wedge$ can be defined as

$$x \wedge y == not(not(x) \vee not(y))$$

and its interpretation is $1 - max(1 - x, 1 - y))$, that is, $min(x, y)$. The connectives $\vee$ and $\wedge$ are not the only operators that can be defined starting from $\Rightarrow$ and *not*. Another way to introduce a connective for disjunction is

$$x \ or \ y == not(x) \Rightarrow y$$

with a dual operator *and* defined by

$x \; and \; y == not(not(x) \; or \; not(y))$.

By replacing $\Rightarrow$ by its interpretation $min(1 - x + y, 1)$, we can see that *or* is interpreted as $min(1-(1-x)+y, 1) = min(x+y, 1)$; that is, it is the rational sum of x and y truncated to 1 if it exceeds 1. Interestingly, x *and* y gives the difference between $x+y$ and $min(x+y, 1)$, that is, the information lost by using the truncated sum: the interpretation of x *and* y is $1 - (min(1 - x + 1 - y, 1))$. If $min(x+y, 1) = x+y$, then $min(1-x+1-y, 1) = 1$ and $x \; and \; y = 0$. If $min(x+y, 1) = 1$, i.e. $x+y = 1+a$ for some a, then $min(1-x+1-y, 1) = 1-x+1-y$ and $x \; and \; y = 1-1+x-1+y = x+y-1 = a$. The two pairs $\vee$ and $\wedge$ and *or* and *and* are two different pairs of connectives. Only if the domain of interpretation is $\{0, 1\}$ (i.e., the logic is two-valued), $\vee$ collapses onto *or* and $\wedge$ collapses onto *and*.

The theorem $(x \Rightarrow y) \vee (y \Rightarrow x) == true$ is interpreted as $max(min(1-x+y, 1), min(1-y+x, 1)) == 1$, that is intuitively true, since the left hand side evaluates to $max(1-x+y, 1) = 1$ if $x \geq y$ and to $max(1-y+x, 1) = 1$ if $y \geq x$.

The First Automated Proof of the Dependency of the Fifth Axiom (equational presentation)

The first automated proof of the dependency of the fifth axiom in Wajsberg algebras appeared in [1]. The proof has been obtained by using the theorem prover SBR3. SBR3 is based on the AC-Unfailing Knuth-Bendix completion procedure [7, 3] with several significant enhancements that are described in part in [1, 2]. In completion-based theorem proving, the principle of completion is applied not to generate a canonical system, but to prove refutationally a specific given theorem.

The proof in [1] also uses the knowledge that the following lemmas are true in any Wajsberg algebra [6]:

1. $x \Rightarrow x == true$
2. if $x \Rightarrow y == y \Rightarrow x == true$ then $x == y$
3. $x \Rightarrow true == true$
4. $x \Rightarrow (y \Rightarrow x) == true$
5. if $x \Rightarrow y == y \Rightarrow z == true$ then $x \Rightarrow z == true$
6. $(x \Rightarrow y) \Rightarrow ((z \Rightarrow x) \Rightarrow (z \Rightarrow y)) == true$
7. $x \Rightarrow (y \Rightarrow z) == y \Rightarrow (x \Rightarrow z)$
8. $x \Rightarrow false == x \Rightarrow not(true) == not(x)$
9. $not(not(x)) == x$
10. $not(x) \Rightarrow not(y) == y \Rightarrow x$

We list them here as additional, simpler problems for experimenting in Wajsberg algebras with an equational prover. All of them except lemma 7 have been derived by SBR3 from $\mathcal{W}$ in a few seconds [1]. Lemma 7 is treated below. The proof of the dependency of the fifth axiom in [1] is done incrementally through five executions:

1. Prove lemma 9 $not(not(x)) == x$ from $\mathcal{W}$.

   During the proof the lemmas 1, 3, 4 and 8 are also generated automatically. The running time is 58 secs.

2. Prove lemma 10 $not(x) \Rightarrow not(y) == y \Rightarrow x$ with $\mathcal{W}$ and lemmas 1, 3 and 9 as input. The running time is 11 secs.

3. Introduce the operator *and* defined implicitly by

   $(x \ and \ y) \Rightarrow z == (x \Rightarrow (y \Rightarrow z))$

   and prove that *and* is commutative from $\mathcal{W}$ and lemmas 1, 3, 4, 8, 9 and 10. Lemma 7 $x \Rightarrow (y \Rightarrow z) == y \Rightarrow (x \Rightarrow z)$ is an immediate consequence. The running time is 17 secs.

4. Introduce the operator *or* defined as $x \ or \ y == not(x) \Rightarrow y$, and prove that it is associative and commutative from $\mathcal{W}$ and lemmas 7, 9 and 10. This proof is very easy and can be done quickly also by hand. As a side-effect it produces the equation: $x \ and \ y = not(not(x) \ or \ not(y))$, that defines *and* in terms of *or* and shows that *and* is also AC. Note that lemma 9 implies that $x \ or \ y == not(x) \Rightarrow y$ is equivalent to $x \Rightarrow y == not(x) \ or \ y$ and thus allows us to express implication in terms of *or*.

5. Prove $(x \Rightarrow y) \vee (y \Rightarrow x) == true$ from $\mathcal{W}$, lemmas 1, 3, 9, $x \Rightarrow y == not(x) \ or \ y$, where *or* is AC, and $x \vee y == (x \Rightarrow y) \Rightarrow y$. The running time is 22 minutes and 30 secs.

This proof shows just one successful approach to the problem. Other proofs may be sought. In particular, there remains open the problem of finding a proof without resorting to the auxiliary operators *or* and *and*, that is, working only with the basic operators $\Rightarrow$ and *not*.

Another Approach to an Automated Proof of the Dependency of the Fifth Axiom (equational presentation)

The dependency of the fifth axiom in Wajsberg algebras has been proved by SBR3 in less than 1 minute, by using a different axiomatization of Wajsberg algebras, that we shall call $\mathcal{W}'$. The basic connectives in $\mathcal{W}'$ are *and* and *exclusive or*, that we denote by $\oplus$. The axiomatization $\mathcal{W}'$ has been generated and proved equivalent to $\mathcal{W}$ by SBR3 [2]. This experimentation has been conducted by Siva Anantharaman. We assume to have already performed all the steps of the previous proof but the last one: namely, we have lemmas 9, 10, 7, and the AC operators *and* and *or*, and we can express $\Rightarrow$ in terms of *or*. Then, we define

$x * y == x \ and \ y == not(not(x) \ or \ not(y))$ and

[1] All running times are for a Sun 3/260 and refer to the very first run of this proof in fall 1989. The current version of SBR3 is much more efficient.

$x \oplus y == (x \; and \; not(y)) \; or \; (not(x) \; and \; y)$.

If we add to $\mathcal{W}$ these two definitions, plus $x \Rightarrow y == not(x) \; or \; y$ and the knowledge that *and* and *or* are AC, we can prove by SBR3 all the theorems in the following set $\mathcal{W}'$:

1. $not(x) == x \oplus 1$
2. $x \oplus 0 == x$
3. $x \oplus x == 0$
4. $x * 1 == x$
5. $x * 0 == 0$
6. $(1 \oplus x) * x == 0$
7. $x \oplus (1 \oplus y) == (x \oplus 1) \oplus y$
8. $((1 \oplus x) * y) \oplus 1) * y == ((1 \oplus y) * x) \oplus 1) * x$

where $*$ is AC, while $\oplus$ is *commutative* only. Furthermore, the prover generates the definition of *or* in terms of $\oplus$:

$x \; or \; y = 1 \oplus ((1 \oplus x) * (1 \oplus y))$.

Inversely, if we start with $\mathcal{W}'$ as axiomatization and we add the definition:

$(x \Rightarrow y) = 1 \oplus (x * (1 \oplus y))$,

we obtain by SBR3 all the equations of $\mathcal{W}$ as theorems. This proves that the sets $\mathcal{W}$ and $\mathcal{W}'$ are equivalent axiomatizations. The axiomatization $\mathcal{W}'$ is partly resemblant of the system of axioms for the Boolean ring given by J. Hsiang. However, there are substantial differences, as $\mathcal{W}'$ is an axiomatization for many-valued logic, whereas the axioms for the Boolean ring apply to the Boolean case (i.e., two-valued logic). The product $*$ is *not* idempotent. This can be easily checked by recalling that $*$ is just an alias for *and* and thus is interpreted as $1 - min(1 - x + 1 - y, 1)$: for instance for $x = 0.3$, $x * x = 0$ and for $x = 0.7$, $x * x = 0.4$!

The most important property that is missing is distributivity:

$(x \; and \; y) \; or \; z == (x \; or \; z) \; and \; (y \; or \; z)$ and

$(x \; or \; y) \; and \; z == (x \; and \; z) \; or \; (y \; and \; z)$

do not hold, as can be easily seen by assigning for instance 0.05 to x, 0.2 to y and 0.9 to z. Similarly, distributivity does not hold if *or* is replaced by $\oplus$. Also, $\oplus$ is only commutative in $\mathcal{W}'$, whereas it is AC in the Boolean case. The above assignment to x, y and z is also a counterexample for associativity of $\oplus$. The absence of these properties is clearly related: if distributivity were true, associativity of $\oplus$ would follow and many-valued logic would collapse on two-valued logic.

The Lattice Structure of Wajsberg Algebras

A simple, manual proof of the dependency of the fifth axiom in Wajsberg algebras is sketched in [6]. We describe here this approach, as it may provide hints for other automated proofs. Also, the proof uses a second bunch of lemmas that may be used for further experiments. The proof is based on regarding Wajsberg algebras as lattices. The relation defined by

$x \leq y$ if and only if $x \Rightarrow y == true$

is a partial order: lemmas 1, 2 and 5 establish reflexivity, antisymmetry and transitivity of this relation. If we interpret as usual $x \Rightarrow y$ as $min(1-x+y,1)$ and *true* as 1 on the rational interval $[0,1]$, we see that this order is just the standard ordering on the rational numbers. Indeed, the connectives $\vee$ and $\wedge$, that are interpreted as *max* and *min* on the rational numbers, are the supremum and infimum with respect to this order. The following theorems are given in [6] and proved by using the properties of lattices:

1. if $x \leq y$ then $x \Rightarrow z \geq y \Rightarrow z$
2. if $x \leq y$ then $z \Rightarrow x \leq z \Rightarrow y$
3. $x \leq y \Rightarrow z$ if and only if $y \leq x \Rightarrow z$
4. $not(x \vee y) == not(x) \wedge not(y)$
5. $not(x \wedge y) == not(x) \vee not(y)$
6. $(x \vee y) \Rightarrow z == (x \Rightarrow z) \wedge (y \Rightarrow z)$
7. $x \Rightarrow (y \wedge z) == (x \Rightarrow y) \wedge (x \Rightarrow z)$
8. $(x \Rightarrow y) \vee (y \Rightarrow x) == true$
9. $x \Rightarrow (y \vee z) == (x \Rightarrow y) \vee (x \Rightarrow z)$
10. $(x \wedge y) \Rightarrow z == (x \Rightarrow z) \vee (y \Rightarrow z)$
11. $(x \wedge y) \vee z == (x \vee z) \wedge (y \vee z)$
12. $(x \wedge y) \Rightarrow z == (x \Rightarrow y) \Rightarrow (x \Rightarrow z)$

Theorem 8 is the dependency of the fifth axiom and theorem 11 is distributivity, that hold between $\vee$ and $\wedge$ whereas it does not for *or* and *and*. Assuming to have proved the theorems preceding it in the above list, the dependency of the fifth axiom can be proved as follows: by instantiating first z to y and then z to x in theorem 6, we obtain respectively

$x \Rightarrow y == (x \vee y) \Rightarrow y$ and $y \Rightarrow x == (x \vee y) \Rightarrow x.$

Then we have

$(x \Rightarrow y) \Rightarrow (y \Rightarrow x) == ((x \vee y) \Rightarrow y) \Rightarrow ((x \vee y) \Rightarrow x)$

by using the two above equations,

$((x \vee y) \Rightarrow y) \Rightarrow ((x \vee y) \Rightarrow x) == (not(y) \Rightarrow not(x \vee y)) \Rightarrow (not(x) \Rightarrow not(x \vee y))$

by lemma 10,

$(not(y) \Rightarrow not(x \vee y)) \Rightarrow (not(x) \Rightarrow not(x \vee y)) == not(x) \Rightarrow ((not(y) \Rightarrow not(x \vee y)) \Rightarrow not(x \vee y))$

by lemma 7,

$not(x) \Rightarrow ((not(y) \Rightarrow not(x \vee y)) \Rightarrow not(x \vee y)) == not(x) \Rightarrow (not(y) \vee not(x \vee y))$

by the definition of $\vee$,

$not(x) \Rightarrow (not(y) \vee not(x \vee y)) == not(not(y) \vee not(x \vee y)) \Rightarrow x$

by lemma 10,

$not(not(y) \vee not(x \vee y)) \Rightarrow x == (y \wedge (x \vee y)) \Rightarrow x$

by theorem 4 in the above list and lemma 9,

$(y \wedge (x \vee y)) \Rightarrow x == y \Rightarrow x$

by the absorption law, so that finally we have proved

$((x \Rightarrow y) \Rightarrow (y \Rightarrow x)) \Rightarrow (y \Rightarrow x) == true$

that is the fifth axiom.

Third Problem: A "One Variable Problem"

The problem is to prove from the four axioms of $\mathcal{W}$, the following theorem

$not((x * (2x)) \; or \; (x^2)) == not(x) * (2not(x)) \; or \; (not(x)^2)$,

where $*$ is an alias for *and*, $2x$ is a short hand for $x \; or \; x$ and x^2 is a short hand for $x * x$. We call it "one variable problem" because just one variable appears. A way to split this problem into easier tasks is as follows:

1. assume $x \; or \; x == true$ and prove the theorem from $\mathcal{W}$ and $x \; or \; x == true$,
2. assume $not(x) \; or \; not(x) == true$ and prove the theorem from $\mathcal{W}$ and $not(x) \; or \; not(x) == true$.

In principle, in order to have a fully automated proof, one should also prove

$(x \; or \; x == true) \vee (not(x) \; or \; not(x) == true)$

from $\mathcal{W}$. SBR3 has proved Step 1 in 19 sec and Step 2 in 15 sec from $\mathcal{W}$, lemmas 1, 3, 9, 10, $(x \; and \; y) \Rightarrow z == (x \Rightarrow (y \Rightarrow z))$ and $x \Rightarrow y == not(x) \; or \; y$ with *or* AC.

The theorem prover SBR3 is available through ftp: all interested readers may send mail to *bonacina@sbcs.sunysb.edu* or to *hsiang@sbcs.sunysb.edu* for instructions.

Acknowledgments

Prof. Daniele Mundici introduced me to Lukasiewicz logic and suggested the dependency of the fifth axiom and the "one-variable problem" as problems for theorem provers.

References

[1] S. Anantharaman and M. P. Bonacina, Automated Proofs in Lukasiewicz Logic, Technical Report, Department of Computer Science, SUNY at Stony Brook, November 1989.

[2] S. Anantharaman and M. P. Bonacina, An Application of the Theorem Prover SBR3 to Many-Valued Logic, to appear in M. Okada and S. Kaplan (eds.), *Proceedings of the Second International Workshop on Conditional and Typed Rewriting Systems*, Montreal, Canada, June 1990. LNCS 516

[3] S. Anantharaman and J. Mzali, Unfailing Completion Modulo a Set of Equations, Technical Report, LRI, Université de Paris Sud, 1989.

[4] C. C. Chang, in *Transactions American Mathematical Society*, No. 88, 467–490, 1958 and ibid. No. 93, 74–80, 1959.

[5] C. C. Chang, in *Transactions American Mathematical Society*, No. 87, 55–56, 1958.

[6] J. M. Font, A. J. Rodriguez, and A. Torrens, Wajsberg algebras, *Stochastica*, Vol. 8, no. 1, 5–31, 1984.

[7] J. Hsiang and M. Rusinowitch, On word problems in equational theories, in Th. Ottman (ed.), *Proceedings of the Fourteenth International Conference on Automata, Languages and Programming*, Karlsruhe, West Germany, July 1987, Springer Verlag, Lecture Notes in Computer Science 267, 54–71, 1987.

[8] W. W. McCune, OTTER 2.0 Users Guide, Technical Report ANL-90/9, Argonne National Laboratory, Argonne, Illinois, 1990.

[9] C. A. Meredith, in *Transactions American Mathematical Society*, No. 87, 54, 1958.

[10] D. Mundici, Interpretation of AFC^*-algebras in Lukasiewicz sentential calculus, *Journal of Functional Analysis*, No. 65, 15–63, 1986.

[11] N. Rescher, *Many Valued Logic*, McGraw-Hill, New York, 1969.

[12] A. J. Rodriguez, A. Torrens, and Verdu, Lukasiewicz logic and Wajsberg algebras, *Bulletin of the Polish Academy of Sciences*, Sect. Logic, 51–55, June 1990.

[13] A. Rose and J. B. Rosser, in *Transactions American Mathematical Society*, No. 87, Chapter 13, 1–53, 1958.

[14] A. Tarski and J. Lukasiewicz, Investigations into the sentential calculus, Chapter IV in A. Tarski, *Logic, Semantics and Metamathematics*, 38–56, Clarendon Press, Oxford, 1956.

[15] L. Wos, New Challenge Problem in Sentential Calculus, *AAR Newsletter*, No. 16, November 1990.