Gentzen-Type Refutation Systems for Three-Valued Logics with an Application to Disproving Strong Equivalence*

Johannes Oetsch and Hans Tompits

Technische Universität Wien, Institut für Informationssysteme 184/3,
Favoritenstraße 9-11, A-1040 Vienna, Austria
{oetsch,tompits}@kr.tuwien.ac.at

Abstract. While the purpose of conventional proof calculi is to axiomatise the set of valid sentences of a logic, *refutation systems* axiomatise the invalid sentences. Such systems are relevant not only for proof-theoretic reasons but also for realising deductive systems for nonmonotonic logics. We introduce Gentzen-type refutation systems for two basic three-valued logics and we discuss an application of one of these calculi for disproving strong equivalence between answer-set programs.

1 Introduction

In contrast to conventional proof calculi that axiomatise the valid sentences of a logic, *refutation systems*, also known as *complementary calculi* or *rejection systems*, are concerned with axiomatising the invalid sentences. Axiomatic rejection was introduced in modern logic by Jan Łukasiewicz in his well-known treatise on analysing Aristotle's syllogistic [1]. Subsequently, refutation systems have been studied for different logics [2–8] (for an overview, cf., e.g., the papers by Wybraniec-Skardowska [9] and by Caferra and Peltier [10]). Such systems are relevant not only for proof-theoretic reasons but also for realising deductive systems for nonmonotonic logics [11]. Moreover, axiomatic refutation provide the means for proof-theoretic investigations concerned with proof complexity, i.e., with the size of proof representations [12].

In this paper, we introduce analytic Gentzen-type refutation systems for two particular three-valued logics, $\mathcal{L}$ and $\mathcal{P}$, following Avron [13]. The notable feature of these logics is that they are *truth-functionally complete*, i.e., any truth-functional three-valued logic can be embedded into these logics. In particular, Gödel's three-valued logic [14] is expressible in $\mathcal{L}$, and since equivalence in this logic amounts to strong equivalence between logic programs under the answer-set semantics, in view of the well-known result by Lifschitz, Pearce, and Valverde [15], we can apply our refutation system for $\mathcal{L}$ to disprove strong equivalence between programs in a purely deductive manner, which will be briefly discussed in this paper as well. Finally, there is a Prolog implementation of our calculi available, which can be downloaded at

`www.kr.tuwien.ac.at/research/projects/mmdasp`.

* This work was partially supported by the Austrian Science Fund (FWF) under grant P21698. The authors would like to thank Valentin Goranko, Robert Sochacki, and Urszula Wybraniec-Skardowska for valuable support during the preparation of this paper.

2 Preliminaries

Unlike classical two-valued logic, three-valued logics admit a further truth value besides true and false. Let $\mathbf{t}$ and $\mathbf{f}$ be the classical truth values, representing true and false propositions, respectively, and $\mathbf{i}$ the third one. Semantically, there are only two major classes of three-valued logics: those where $\mathbf{i}$ is *designated*, i.e., associated with truth, and those where $\mathbf{i}$ is not designated. In this paper, we are concerned with two logics, $\mathcal{L}$ and $\mathcal{P}$ [13]. Logic $\mathcal{L}$ can be considered as a prototypical logic where $\mathbf{i}$ is not designated, whilst $\mathcal{P}$ is a prototypical logic where $\mathbf{i}$ is designated. Both logics are fully expressive, meaning that they allow to embed any truth-functional three-valued logic from the literature in it.

Both $\mathcal{L}$ and $\mathcal{P}$ are formulated over a countably infinite universe $\mathcal{U}$ of atoms including the truth constants T, F, and I. Based on the connectives $\neg$, $\vee$, $\wedge$, and $\supset$, the set of well-formed formulae is defined as usual. A set of literals is *consistent* if it does not contain both an atom and its negation. In $\mathcal{P}$, $\mathbf{t}$ and $\mathbf{i}$ are designated, while in $\mathcal{L}$, the only designated truth value is $\mathbf{t}$.

By an *interpretation*, we understand a mapping from $\mathcal{U}$ into $\{\mathbf{t}, \mathbf{f}, \mathbf{i}\}$. For any interpretation I, $I(\mathrm{T}) = \mathbf{t}$, $I(\mathrm{F}) = \mathbf{f}$, and $I(\mathrm{I}) = \mathbf{i}$. As usual, a *valuation* is a mapping from formulae into the set of truth values. We assume the ordering $\mathbf{f} < \mathbf{i} < \mathbf{t}$ on the truth values in what follows. The valuation $v^I_{\mathcal{L}}(\cdot)$ of a formula in $\mathcal{L}$ given an interpretation I is is inductively defined as follows: (i) $v^I_{\mathcal{L}}(\psi) = I(\psi)$, if ψ is an atomic formula; (ii) $v^I_{\mathcal{L}}(\neg\psi) = \mathbf{t}$ if $v^I_{\mathcal{L}}(\psi) = \mathbf{f}$, $v^I_{\mathcal{L}}(\neg\psi) = \mathbf{f}$ if $v^I_{\mathcal{L}}(\psi) = \mathbf{t}$, and $v^I_{\mathcal{L}}(\neg\psi) = \mathbf{i}$ otherwise; (iii) $v^I_{\mathcal{L}}(\psi \wedge \varphi) = \min(v^I_{\mathcal{L}}(\psi), v^I_{\mathcal{L}}(\varphi))$; (iv) $v^I_{\mathcal{L}}(\psi \vee \varphi) = \max(v^I_{\mathcal{L}}(\psi), v^I_{\mathcal{L}}(\varphi))$; and (v) $v^I_{\mathcal{L}}(\psi \supset \varphi) = v^I_{\mathcal{L}}(\varphi)$ if $v^I_{\mathcal{L}}(\psi) = \mathbf{t}$, and $v^I_{\mathcal{L}}(\psi \supset \varphi) = \mathbf{t}$ otherwise. The valuation $v^I_{\mathcal{P}}(\cdot)$ of a formula in $\mathcal{P}$ given an interpretation I is defined like $v^I_{\mathcal{L}}(\cdot)$ except for the condition of the implication: $v^I_{\mathcal{P}}(\psi \supset \varphi) = v^I_{\mathcal{P}}(\varphi)$ if $v^I_{\mathcal{P}}(\psi) = \mathbf{t}$ or $v^I_{\mathcal{P}}(\psi) = \mathbf{i}$, and $v^I_{\mathcal{P}}(\psi \supset \varphi) = \mathbf{t}$ otherwise.

A formula ψ is *true* under an interpretation I in $\mathcal{L}$ if $v^I_{\mathcal{L}}(\psi) = \mathbf{t}$. Likewise, ψ is *true* for I in $\mathcal{P}$ if $v^I_{\mathcal{P}}(\psi) = \mathbf{t}$ or $v^I_{\mathcal{P}}(\psi) = \mathbf{i}$. If ψ is true under I in $\mathcal{L}$ (resp., $\mathcal{P}$), I is a *model* of ψ in $\mathcal{L}$ (resp., $\mathcal{P}$). For a set Γ of formulae, I is a model of Γ in $\mathcal{L}$ (resp., $\mathcal{P}$) if I is a model in $\mathcal{L}$ (resp., $\mathcal{P}$) for each formula in Γ. A formula is *valid* in $\mathcal{L}$ (resp., $\mathcal{P}$) if it is true for each interpretation in $\mathcal{L}$ (resp., $\mathcal{P}$).

3 The Refutation Calculi SRCL and SRCP

Bryll and Maduch [16] axiomatised the invalid sentences of Łukasiewicz's many-valued logics including the three-valued case by means of a Hilbert-type calculus. Since their calculus is not analytic, its usefulness for proof search in practice is rather limited. In this paper, we aim at *analytic Gentzen-style refutation calculi* for three-valued logics. The first sequential refutation systems for classical propositional logic was introduced by Tiomkin [4]; equivalent systems were independently discussed by Goranko [6] and Bonatti [5]. We pursue this work towards similar refutation systems for the logics $\mathcal{L}$ and $\mathcal{P}$, which we will call **SRCL** and **SRCP**, respectively.

$$\frac{\Gamma \dashv \Delta, \psi}{\Gamma, \psi \supset \varphi \dashv \Delta}\ (\supset l)_1 \qquad \frac{\Gamma, \varphi \dashv \Delta}{\Gamma, \psi \supset \varphi \dashv \Delta}\ (\supset l)_2 \qquad \frac{\Gamma, \psi \dashv \Delta, \varphi}{\Gamma \dashv \Delta, \psi \supset \varphi}\ (\supset r)$$

$$\frac{\Gamma, \psi, \varphi \dashv \Delta}{\Gamma, \psi \wedge \varphi \dashv \Delta}\ (\wedge l) \qquad \frac{\Gamma \dashv \Delta, \psi}{\Gamma \dashv \Delta, \psi \wedge \varphi}\ (\wedge r)_1 \qquad \frac{\Gamma \dashv \Delta, \varphi}{\Gamma \dashv \Delta, \psi \wedge \varphi}\ (\wedge r)_2$$

$$\frac{\Gamma, \psi \dashv \Delta}{\Gamma, \psi \vee \varphi \dashv \Delta}\ (\vee l)_1 \qquad \frac{\Gamma, \varphi \dashv \Delta}{\Gamma, \psi \vee \varphi \dashv \Delta}\ (\vee l)_2 \qquad \frac{\Gamma \dashv \Delta, \psi, \varphi}{\Gamma \dashv \Delta, \psi \vee \varphi}\ (\vee r)$$

Fig. 1. Standard rules of $\mathbf{SRCL}$ and $\mathbf{SRCP}$.

By an *anti-sequent*, we understand a pair of form $\Gamma \dashv \Delta$, where Γ and Δ are finite sets of formulae.[1] Given a set Γ of formulas and a formula ψ, following custom, we write "Γ, ψ" as a shorthand for $\Gamma \cup \{\psi\}$. An interpretation I *refutes* $\Gamma \dashv \Delta$ in $\mathcal{L}$ (resp., $\mathcal{P}$) iff I is a model of Γ in $\mathcal{L}$ (resp., $\mathcal{P}$) and all formulae in Δ are false under I in $\mathcal{L}$ (resp., $\mathcal{P}$). Moreover, an anti-sequent is *refutable* in $\mathcal{L}$ (resp., $\mathcal{P}$) iff it is refuted by some interpretation in $\mathcal{L}$ (resp., $\mathcal{P}$).

The postulates of the calculi $\mathbf{SRCL}$ and $\mathbf{SRCP}$ are as follows: Let Γ and Δ be two disjoint sets of literals such that $\neg\mathrm{T}, \mathrm{F} \notin \Gamma$ and $\mathrm{T}, \neg\mathrm{F} \notin \Delta$. Then, $\Gamma \dashv \Delta$ is an axiom of $\mathbf{SRCL}$ iff $\{\mathrm{I}, \neg\mathrm{I}\} \cap \Gamma = \emptyset$ and Γ is consistent, and $\Gamma \dashv \Delta$ is an axiom of $\mathbf{SRCP}$ iff $\{\mathrm{I}, \neg\mathrm{I}\} \cap \Delta = \emptyset$ and Δ is consistent. The inference rules of $\mathbf{SRCL}$ and $\mathbf{SRCP}$ comprise the *standard rules* depicted in Fig. 1 and the *non-standard rules* depicted in Fig. 2. The standard rules introduce one occurrence of $\wedge$, $\vee$, or $\supset$ at a time. Note that they coincide with the respective introduction rules in the refutation systems for classical logic [4, 6, 5]. The non-standard rules introduce two occurrences of a connective at the same time, in particular this concerns negation in combination with all other connectives. Note that the logical rules of $\mathbf{SRCL}$ and $\mathbf{SRCP}$ coincide, so the difference between the two calculi lies only in their axioms.

Theorem 1 (Soundness and Completeness). *For any anti-sequent $\Gamma \dashv \Delta$, (i) $\Gamma \dashv \Delta$ is provable in $\mathbf{SRCL}$ iff $\Gamma \dashv \Delta$ is refutable in $\mathcal{L}$, and (ii) $\Gamma \dashv \Delta$ is provable in $\mathbf{SRCP}$ iff $\Gamma \dashv \Delta$ is refutable in $\mathcal{P}$.*

Note that our calculi are, in a sense, refutational counterparts of the Gentzen-type calculi of Avron [17] for axiomatising the valid sentences of $\mathcal{L}$ and $\mathcal{P}$. In fact, for each unary rule in Avron's systems, our system contains a respective rule were "$\vdash$" is replaced by "$\dashv$", whilst for each binary rule of form

$$\frac{\Gamma' \vdash \Delta' \quad \Gamma'' \vdash \Delta''}{\Gamma \vdash \Delta}$$

of Avron, our systems contain two rules

$$\frac{\Gamma' \dashv \Delta'}{\Gamma \dashv \Delta} \quad \text{and} \quad \frac{\Gamma'' \dashv \Delta'}{\Gamma \dashv \Delta}.$$

[1] The symbol "$\dashv$", the dual of Frege's assertion sign "$\vdash$", is due to Ivo Thomas.

$$\frac{\Gamma, \psi \dashv \Delta}{\Gamma, \neg\neg\psi \dashv \Delta}\ (\neg\neg\, l) \qquad \frac{\Gamma \dashv \Delta, \psi}{\Gamma \dashv \Delta, \neg\neg\psi}\ (\neg\neg\, r)$$

$$\frac{\Gamma, \neg\psi \dashv \Delta}{\Gamma, \neg(\psi \wedge \varphi) \dashv \Delta}\ (\neg\wedge\, l)_1 \qquad \frac{\Gamma, \neg\varphi \dashv \Delta}{\Gamma, \neg(\psi \wedge \varphi) \dashv \Delta}\ (\neg\wedge\, l)_2$$

$$\frac{\Gamma \dashv \Delta, \neg\psi, \neg\varphi}{\Gamma \dashv \Delta, \neg(\psi \wedge \varphi)}\ (\neg\wedge\, r) \qquad \frac{\Gamma, \neg\psi, \neg\varphi \dashv \Delta}{\Gamma, \neg(\psi \vee \varphi) \dashv \Delta}\ (\neg\vee\, l)$$

$$\frac{\Gamma \dashv \Delta, \neg\psi}{\Gamma \dashv \Delta, \neg(\psi \vee \varphi)}\ (\neg\vee\, r)_1 \qquad \frac{\Gamma \dashv \Delta, \neg\varphi}{\Gamma \dashv \Delta, \neg(\psi \vee \varphi)}\ (\neg\vee\, r)_2$$

$$\frac{\Gamma, \psi, \neg\varphi \dashv \Delta}{\Gamma, \neg(\psi \supset \varphi) \dashv \Delta}\ (\neg\supset l) \qquad \frac{\Gamma \dashv \Delta, \psi}{\Gamma \dashv \Delta, \neg(\psi \supset \varphi)}\ (\neg\supset r)_1$$

$$\frac{\Gamma \dashv \Delta, \neg\varphi}{\Gamma \dashv \Delta, \neg(\psi \supset \varphi)}\ (\neg\supset r)_2$$

Fig. 2. Non-Standard rules of $\mathbf{SRCL}$ and $\mathbf{SRCP}$.

Hence, as already remarked by Bonatti [5], exhaustive search in the standard system becomes non-determinism in the refutation system—a property that often allows for quite concise proofs and thus helps to reduce the size of proof representations.

Contrary to standard sequential systems, our systems do not contain binary rules. Hence, proofs in our systems are not trees but sequences, and consequently each proof has a single axiom. In fact, a proof of a formula ψ does not represent a single counter model for ψ, rather it represents an entire class of counter models for ψ, in view of the following property underlying the soundness of our calculi: each interpretation I that refutes the axiom $\Gamma \dashv \Delta$ in a proof of $\dashv \psi$ in $\mathbf{SRCL}$ (resp., in $\mathbf{SRCP}$), also refutes $\dashv \psi$ in $\mathbf{SRCL}$ (resp., in $\mathbf{SRCP}$).

4 An Application for Disproving Strong Equivalence

We outline an application scenario that is concerned with logic programs under the answer-set semantics [18]. In a nutshell, a (*disjunctive*) *logic program* is a set of rules of form $a_1 \vee \cdots \vee a_l \leftarrow a_{l+1}, \ldots, a_m, \mathrm{not}\ a_{m+1}, \ldots \mathrm{not}\ a_n$, where all a_i are atoms over some universe $\mathcal{U}$ and "not" denotes default negation. The answer-sets of a program are sets of atoms defined using a fixed-point construction based on the reduct of a program relative to an interpretation [18].

Two logic programs are *equivalent* if they have the same answer sets. In contrast to classical logic, equivalence between programs fails to yield a replacement property. The notion of *strong equivalence* circumvents this problem: two programs P and Q are strongly equivalent iff, for each program R, $P \cup R$ and $Q \cup R$ are equivalent. For instance, consider $P = \{a \leftarrow \mathrm{not}\ b,\ b \leftarrow \mathrm{not}\ a\}$ and $Q = \{a \vee b\}$. P and Q are equivalent but not strongly equivalent.

$$\frac{\Gamma, \neg\psi \dashv \Delta}{\Gamma, \sim\psi \dashv \Delta}\ (\sim l) \qquad \frac{\Gamma \dashv \Delta, \neg\psi}{\Gamma \dashv \Delta, \sim\psi}\ (\sim r)$$

$$\frac{\Gamma \dashv \Delta, \psi, \neg\varphi}{\Gamma, \psi \to_G \varphi \dashv \Delta}\ (\to_G l)_1 \qquad \frac{\Gamma, \varphi \dashv \Delta}{\Gamma, \psi \to_G \varphi \dashv \Delta}\ (\to_G l)_2$$

$$\frac{\Gamma, \neg\psi \dashv \Delta}{\Gamma, \psi \to_G \varphi \dashv \Delta}\ (\to_G l)_3 \qquad \frac{\Gamma, \psi \dashv \Delta, \varphi}{\Gamma \dashv \Delta, \psi \to_G \varphi}\ (\to_G r)_1$$

$$\frac{\Gamma, \neg\varphi \dashv \Delta, \neg\psi}{\Gamma \dashv \Delta, \psi \to_G \varphi}\ (\to_G r)_2 \qquad \frac{\Gamma \dashv \Delta, \neg\psi}{\Gamma, \neg\sim\psi \dashv \Delta}\ (\neg\sim l)$$

$$\frac{\Gamma, \neg\psi \dashv \Delta}{\Gamma \dashv \Delta, \neg\sim\psi}\ (\neg\sim r) \qquad \frac{\Gamma, \neg\varphi \dashv \Delta, \neg\psi}{\Gamma, \neg(\psi \to_G \varphi) \dashv \Delta}\ (\neg\to_G l)$$

$$\frac{\Gamma \dashv \Delta, \neg\varphi}{\Gamma \dashv \Delta, \neg(\psi \to_G \varphi)}\ (\neg\to_G r)_1 \qquad \frac{\Gamma, \neg\psi \dashv \Delta}{\Gamma \dashv \Delta, \neg(\psi \to_G \varphi)}\ (\neg\to_G r)_2$$

Fig. 3. Derived rules for three-valued Gödel logic.

The central observation connecting strong equivalence with three-valued logics is the well-known result [15] that strong equivalence between two programs P and Q holds iff P and Q, interpreted as theories, are equivalent in Gödel's three-valued logic [14]. The connectives of three-valued Gödel logic are $\wedge$, $\vee$, $\sim$, and $\to_G$, which can be defined in $\mathcal{L}$ as $\sim\psi = \neg(\neg\psi \supset \psi)$ and $\psi \to_G \varphi = ((\neg\varphi \supset \neg\psi) \supset \psi) \supset \varphi$. In view of this, we can extend $\mathbf{SRCL}$ by derived rules for $\sim$and $\to_G$, which are given in Fig. 3.

To verify that P and Q are not strongly equivalent, it suffices to give a proof of one of $P \dashv Q$ or $Q \dashv P$ in $\mathbf{SRCL}$.[2] While $Q \dashv P$ is not provable, there is a proof of $P \dashv Q$:

$$\cfrac{\cfrac{\cfrac{\cfrac{\dashv a, b, \neg a, \neg b}{\dashv a, b, \sim a, \sim b}\ (\sim r), (\sim r)}{\sim a \to_G b, \sim b \to_G a \dashv a, b}\ (\to_G l)_1, (\to_G l)_1}{\sim a \to_G b, \sim b \to_G a \dashv a \vee b}\ (\vee r)}{(\sim a \to_G b) \wedge (\sim b \to_G a) \dashv a \vee b}\ (\wedge l)$$

Hence, P and Q are indeed not strongly equivalent. In fact, as detailed below, a concrete program R such that $P \cup R$ and $Q \cup R$ have different answer sets, i.e., a *witness* that P and Q are not strongly equivalent, can be immediately constructed from the axiom $\dashv a, b, \neg a, \neg b$ of the above proof: $R = \{a \leftarrow b,\ b \leftarrow a\}$. Indeed, $P \cup R$ has no answer set while $Q \cup R$ yields $\{a, b\}$ as its unique answer set.

The general method to obtain a witness theory (as R above) from an axiom in $\mathbf{SRCL}$ is as follows: Given an axiom $\Gamma \dashv \Delta$, construct some interpretation I that refutes $\Gamma \dashv \Delta$. For the above example, an interpretation that assigns both a and b to $\mathbf{i}$ would refute the axiom already. Note that I then refutes $P \dashv Q$ as well. Based on I, a witness program

[2] We interpret programs as a theories, i.e., as the conjunctions of rules, where rules are interpreted as implications.

R can always be constructed by using the next proposition which immediately follows from the proof of the main theorem by Lifschitz, Pearce, and Valverde [15]:

Proposition 1. *Let P and Q be two programs such that an I is a model of P but not of Q in three-valued Gödel logic, and let J be the classical interpretation defined by setting $J(a) = \mathbf{f}$ iff $I(a) = \mathbf{f}$, and define $R' = \{a \mid I(a) = \mathbf{t}$ or $I(a) = \mathbf{i}\}$ and $R'' = \{a \mid I(a) = \mathbf{t}\} \cup \{a \rightarrow_G b \mid I(a) = I(b) = \mathbf{i}\}$. Then, $P \cup R$ and $Q \cup R$ are not strongly equivalent, where $R = R'$ if J is not a classical model of Q, and $R = R''$ otherwise.*

Note that a proof that two programs are not strongly equivalent represents, in general, not only a single witness program but an entire class of programs which distinguishes our axiomatic approach from approaches based on finding counter models.

References

1. Łukasiewicz, J.: Aristotle's syllogistic from the standpoint of modern formal logic. 2nd edn. Clarendon Press, Oxford (1957)
2. Kreisel, G., Putnam, H.: Eine Unableitbarkeitsbeweismethode für den Intuitionistischen Aussagenkalkül. Archiv für Mathematische Logik und Grundlagenforschung **3** (1957) 74–78
3. Wójcicki, R.: Dual counterparts of consequence operations. Bulletin of the Section of Logic **2** (1973) 54–57
4. Tiomkin, M.: Proving unprovability. In: 3rd Annual Symposium on Logics in Computer Science, IEEE (1988) 22–27
5. Bonatti, P.A.: A Gentzen system for non-theorems. Technical Report CD-TR 93/52, Christian Doppler Labor für Expertensysteme, Technische Universität Wien (1993)
6. Goranko, V.: Refutation systems in modal logic. Studia Logica **53** (1994) 299–324
7. Skura, T.: Refutations and proofs in S4. In: Proof Theory of Modal Logic. Kluwer (1996) 45–51
8. Skura, T.: A refutation theory. Logica Universalis **3** (2009) 293–302
9. Wybraniec-Skardowska, U.: On the notion and function of the rejection of propositions. Acta Universitatis Wratislaviensis Logika **23** (2005) 179–202
10. Caferra, R., Peltier, N.: Accepting/rejecting propositions from accepted/rejected propositions: A unifying overview. International Journal of Intelligent Systems **23** (2008) 999–1020
11. Bonatti, P.A., Olivetti, N.: Sequent calculi for propositional nonmonotonic logics. ACM Transactions on Computational Logic **3** (2002) 226–278
12. Egly, U., Tompits, H.: Proof-complexity results for nonmonotonic reasoning. ACM Transactions on Computational Logic **2** (2001) 340–387
13. Avron, A.: Natural 3-valued logics - Characterization and proof theory. Journal of Symbolic Logic **56 (1)** (1991) 276–294
14. Gödel, K.: Zum intuitionistischen Aussagenkalkül. Anzeiger Akademie der Wissenschaften Wien, mathematisch-naturwissenschaftliche Klasse **32** (1932) 65–66
15. Lifschitz, V., Pearce, D., Valverde, A.: Strongly equivalent logic programs. ACM Transactions on Computational Logic **2** (2001) 526–541
16. Bryll, G., Maduch, M.: Aksjomaty odrzucone dla wielowartościowych logik Łukasiewicza. Zeszyty Naukowe Wyższej Szkły Pedagogigicznej w Opolu, Matematyka VI, Logika i algebra (1968) 3–17
17. Avron, A.: Classical Gentzen-type methods in propositional many-valued logics. In: 31st IEEE International Symposium on Multiple-Valued Logic, IEEE (2001) 287–298
18. Gelfond, M., Lifschitz, V.: Classical negation in logic programs and disjunctive databases. New Generation Computing **9** (1991) 365–385