Logica Universalis

Tree-Like Proof Systems for Finitely-Many Valued Non-deterministic Consequence Relations

Pawel Pawlowski

Abstract. The main goal of this paper is to provide an abstract framework for constructing proof systems for various many-valued logics. Using the framework it is possible to generate strongly complete proof systems with respect to any finitely valued deterministic and non-deterministic logic. I provide a couple of examples of proof systems for well-known many-valued logics and prove the completeness of proof systems generated by the framework.

Mathematics Subject Classification. Primary 03B60, 03B50.

Keywords. Non-deterministic logics, Proof systems, Many valued logics.

1. Motivations

In this paper, we'll present an abstract framework for constructing tree-like (tableaux) proof systems for finitely-many valued deterministic and non-deterministic consequence relations.[1]

Non-deterministic logics were constructed[2] by Avron and Lev [5] as a generalization of many-valued deterministic logics. Non-deterministic logics are characterized by means of non-deterministic matrices. Roughly speaking,

---

[1] This proof systems are in the spirit of Carnielli [14]. For a very nice introduction to this method see Priest [27]. The whole framework can be seen as a generalization of the systems presented in Priest [27].

[2] To be fair, Rescher [29], Ivlev [21], Kearns [22], and Quine [28] suggested something similar to a non-deterministic matrix.

in such matrices an interpretation may ascribe sets of values instead of a single value. This means that these logics are not truth-functional. In their paper, Avron and Lev constructed an abstract framework for sequent-based proof systems which was further studied in [1–3].

A recipe for constructing tree-like (or tableaux) proof systems for non-deterministic semantics will be provided.[3] This is at least as interesting as providing alternative proof systems for any other kind of logic. Here the main advantage of the framework is that it is quite general and that we do not find sequent based proof systems particularly handy when it comes to actually reasoning within the system.

One of the other advantages of these systems over sequent-based systems is that there is a nice mechanical procedure for finding a counter-model for any invalid inference. In this setting it is possible to keep track of values of formulas that appear on a tree. This allows to find valuations under which the premises have the designated value and the conclusion doesn't. Tree-like proof systems are also easy to handle for both humans and computers and they have a nice visual representation.

Moreover, it seems that some logics that use intensional operators such as modal logics or some paraconsistent logics despite not having finitely-many valued deterministic semantics, may have finite non-deterministic semantics.[4]

There is a very nice and general algebraic proof framework for non-deterministic consequence relations called *The method of Polynomial Ring Calculus*.[5] The method is mainly used in automated theorem proving. Since in that context the user-friendliness of the method is not important unsurprisingly, the proof system is not easy to work with. On the other hand, tableaux method is known to be very user-friendly and pedagogically interesting, and as such deserves attention.

The paper is structured as follows. The first section sets the notation and some necessary definitions. The second section is a presentation of the framework. The third section describes a couple of examples of proof systems generated using the framework for various well-known logics. Next, a brief section that elaborates on extensions of the framework and also comments on some issues with complexity will be presented. The last section is a sketch of a strong completeness proof.

2. Technical Preliminaries

Let $\mathcal{L}$ be a propositional language understood as a set of propositional variables $W = \{p, q, \ldots\}$ closed under connectives from the set

$$F = \left\{ \circ_1^1, \circ_2^1, \ldots, \circ_1^2, \circ_2^2, \ldots, \circ_m^n \right\}$$

---

[3]Roughly speaking, non-deterministic semantics is a semantics in which the value of a complex formula is not uniquely determined by the values of its sub-formulas.
[4]Some logicians are interested in finding non-deterministic semantics for such logics [15,16, 24].
[5]See Carnielli and Matulovic [13].

where the upper script is the arity of a given functor. Greek letters $\varphi, \psi, \ldots$ stand for meta-variables for formulas. Capital Greek letters denotes sets of formulas.

Definition 1. (*n-matrix*) An *n-matrix* for a propositional language $\mathcal{L}$ is a triple $\mathtt{M} = \langle T, D, O \rangle$, where:

- T is a non-empty set of truth values.
- $\emptyset \neq D \subseteq T$ is a set of designated values.
- O is a set of functions, which for any n-ary functor $\circ^n$ of the language contains a corresponding n-ary function $\overline{\circ^n}$, $\overline{\circ^n} \colon T^n \mapsto 2^T \setminus \{\emptyset\}$.

It is quite easy to see that the above notion is a generalization of deterministic matrices. The set O consists of functions called truth-functions which assign sets of possible values to complex formulas, given assignments of values to their components. In the case of deterministic matrices, these sets are singletons. In a proper n-matrix functions from O can assign non-empty sets of values, including those which are not singletons. In such case every valuation picks exactly one value of a formula from the sets of possible values ascribed to it by an appropriate function from O.

Definition 2. (*Valuation*) A *valuation* $v \colon \mathcal{L} \mapsto T$ in an n-matrix $\mathtt{M}$ is a function such that for any functor $\circ^n$ and any sequence of formulas $\varphi_1, \varphi_2 \ldots, \varphi_n$, $v(\circ^n(\varphi_1, \varphi_2, \ldots, \varphi_n)) \in \overline{\circ^n}(v(\varphi_1), v(\varphi_2), \ldots, v(\varphi_n))$. A *valuation* v *satisfies* φ in $\mathtt{M}$ ($v \models_{\mathtt{M}} \varphi$) iff $v(\varphi) \in D$. We say that φ *follows from* Γ [$\Gamma \models_{\mathtt{M}} \varphi$] iff every valuation that satisfies all elements of Γ satisfies φ. An n-matrix $\mathtt{M}$ is k-valued if $|T| = k$.

3. Trees

Suppose we have an n-valued non-deterministic consequence relation, whose semantics is given by an n-matrix $\mathtt{M}$, where $T = \{0, 1, \ldots, n-2, d\}$ and d is its designated value.[6]

Definition 3 (*Root appropriate* Γ, ψ). Let $\Gamma = \{\varphi_1, \varphi_2, \ldots, \varphi_i\}$, be a set of formulas and ψ a single formula. By the *root appropriate* for Γ, ψ we mean the following construction:

---

[6] We stick to the case where there is only one designated value. In order to deal with a logic whose matrix has more designated values, the validity of an inferences is checked by constructing a tree for each combination of values under which all the premises have labels corresponding to different designated values and the conclusion's label corresponds to different non-designated values. So for a reasoning with m premises in a logic with d designated values one has to construct m^d different trees. If all of them are closed, then the inference is valid. If at least one is open, it is not.

$$\begin{array}{c}
\varphi_1, d \\
\varphi_2, d \\
\vdots \\
\varphi_i, d
\end{array}$$

$$\psi, 0 \quad \psi, 1 \quad \ldots \quad \psi, n-1$$
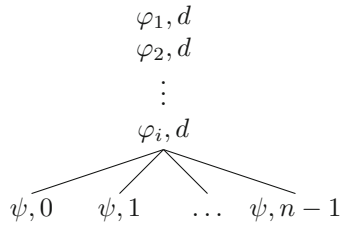
The idea here is to list all the premises and ascribe to them index d. Roughly speaking, it means that under all the valuations corresponding to this root all premises have the designated value. Next, we split the construction into $n-1$ branches. On each branch we list the conclusion with different indices. These indices represent all non-designated values. Intuitively, each branch corresponds to any valuation under which all the premises and the conclusion values are its own indices determined by the branch.
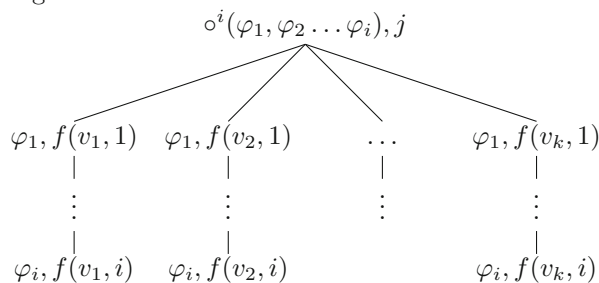
The root is further extended into a tree by means of syntactical rules appropriate for a given logic. Before defining what a syntactical rule is we need to introduce some additional notions.

Definition 4 (*ith combination*). Let φ be a formula. The set of all possible valuations under which φ has value i is called *an ith combination* of φ and is abbreviated as $CO_i(\varphi)$. Let $\circ^j(\varphi_1, \ldots, \varphi_j)$ be a formula built from $\circ^j$ and $\varphi_1 \ldots, \varphi_j$, suppose that $v \in CO_i(\circ^j(\varphi_1, \ldots, \varphi_j))$. By $f(v, i)$ we mean the value ascribed to φ_i by the valuation v.

Definition 5 (*Syntactical rule for $\circ^i, j$*). Let M be an n-matrix and

$$CO_j(\circ^i(\varphi_1, \varphi_2, \ldots, \varphi_i)) = \{v_1, v_2, \ldots, v_k\}.$$

By a *syntactical rule* for $\circ^i(\varphi_1, \varphi_2, \ldots, \varphi_i), j$ appropriate for an n-matrix M we mean the following construction:

$$\circ^i(\varphi_1, \varphi_2 \ldots \varphi_i), j$$

$$\begin{array}{cccc}
\varphi_1, f(v_1, 1) & \varphi_1, f(v_2, 1) & \ldots & \varphi_1, f(v_k, 1) \\
| & | & | & | \\
\vdots & \vdots & \vdots & \vdots \\
| & | & | & | \\
\varphi_i, f(v_1, i) & \varphi_i, f(v_2, i) & & \varphi_i, f(v_k, i)
\end{array}$$

Definition 6 (*Functor described by a set*). Let $M = \langle T, D, O \rangle$ be an n-matrix and $\circ$ be a functor from the language of the matrix. We say that a set $A_\circ$ of syntactical rules *describes* $\circ$ iff for any $t \in T$, $A_\circ$ contains a syntactical rule $\circ, t$ appropriate for M and nothing else is in $A_\circ$.

Definition 7 (*Matrix described by a set*). An n-matrix M is *described by a set of syntactical rules A* iff $A = \bigcup_i A_i$ where A_i describes the i-th functor in the language of n-matrix M.

Suppose that a consequence relation is given by an n-matrix M. In order to check any given inference in our framework we start with a root appropriate for it and then extend it into a tree by a set of syntactical rules describing M. Next, in order to see if an inference is indeed valid in a consequence relation described by M, we need to introduce some additional definitions.

Definition 8 (*Tree appropriate for* Γ, ψ). A *tree is appropriate* for Γ, ψ if it starts with a root appropriate for Γ, ψ.

Definition 9 (*Complete tree*). We say that a *tree is complete* iff it is impossible to extend the tree further using syntactical rules appropriate for a given logic.

Definition 10 (*Closed branch*). We say that a *branch on a tree is closed* iff for some formula φ occurring on the branch it occurs with two different numbers(values).

Definition 11 (*Closed tree*). A *tree is closed* iff all of its branches are closed.

Definition 12 (*Valid inference*). We say that an inference from Γ to φ is *valid on a tree* iff the tree starts with a root appropriate for Γ, φ and is complete and all of its branches are closed. We will denote this by $\Gamma \vdash \varphi$. Moreover if a tree was constructed only by using rules from a set A, we will denote this by $\vdash_A$.

4. A Handful of Examples

In this section we will show how the framework works.[7] We will give four examples of proof systems generated by our framework. Let's start with weak Kleene logic.[8]

4.1. Deterministic Examples

In our case we will be interested in a propositional language with three connectives $\neg, \vee, \wedge$. In a weak Kleene logic[9] these are characterized by the following truth-tables:

$\neg$	value
0	1
n	n
1	0

$\wedge$	0	n	1
0	0	n	0
n	n	n	n
1	0	n	1

$\vee$	0	n	1
0	0	n	1
n	n	n	n
1	1	n	1

by $\models_{Kl} \varphi$ we mean that any valuation respecting the above truth-tables ascribes value 1 to φ. By $\Gamma \models_{Kl} \varphi$ we mean that any valuation v such that for all $\psi \in \Gamma$ $v(\psi) = 1$ ascribe value 1 to φ.

---

[7]Note that the results presented for deterministic consequence relations are not original. They are here for pedagogical reasons. For an extensive study of tableaux method see Baaz et al. [6] and Fitting [18].

[8]This logic sometimes goes by the name of Bochvar logic [12]. One of the main application of this logic is in truth theories [19,20].

[9]Known also by the name Bochvar internal logic. See Bergmann [11] for an extensive overview of many-valued logics.

The above characterization of weak Kleene logic can be easily rephrased in terms of n-matrix.[10] Let A_{Kl} be a set of rules describing Kleene logics. It means that A_{Kl} consists of:

Negation 1:
$$\neg\varphi, 0$$
$$|$$
$$\varphi, 1$$

Negation 2:
$$\neg\varphi, 1$$
$$|$$
$$\varphi, 0$$

Negation 3:
$$\neg\varphi, n$$
$$|$$
$$\varphi, n$$

Disjunction 1:
$$\varphi \vee \psi, 1$$

$$\begin{array}{ccc} \varphi,1 & \varphi,0 & \varphi,1 \\ \psi,0 & \psi,1 & \psi,1 \end{array}$$

Disjunction 2:
$$\varphi \vee \psi, n$$

$$\begin{array}{ccccc} \varphi,n & \varphi,n & \varphi,0 & \varphi,1 & \varphi,n \\ \psi,n & \psi,0 & \psi,n & \psi,n & \psi,1 \end{array}$$

Disjunction 3:
$$\varphi \vee \psi, 0$$
$$|$$
$$\varphi,0$$
$$\psi,0$$

Conjunction 1:
$$\varphi \wedge \psi, 1$$
$$|$$
$$\varphi,1$$
$$\psi,1$$

Conjunction 2:
$$\varphi \wedge \psi, n$$

$$\begin{array}{ccccc} \varphi,n & \varphi,n & \varphi,0 & \varphi,1 & \varphi,n \\ \psi,n & \psi,0 & \psi,n & \psi,n & \psi,1 \end{array}$$

Conjunction 3:
$$\varphi \wedge \psi, 0$$

$$\begin{array}{ccc} \varphi,0 & \varphi,0 & \varphi,1 \\ \psi,0 & \psi,1 & \psi,0 \end{array}$$

It is rather easy to see that the following system is sound (since all the abstract rules are based on truth-tables). A completeness proof is in the last section.

4.2. Non-deterministic examples

The first candidate is a paraconsistent logic CLuN [7,8,10].[11] This logic has both a non-deterministic and a deterministic characterization (see Batens [8,

---

[10] Observe, that truth-tables define interpretations of connectives, the designated value is 1 and $T = \{0, n, 1\}$.
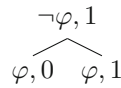
[11] One might be interested in a comparison of CluN with da Costa and Alves [17].

9]). We will focus on the non-deterministic one. CLuN is a two valued logic characterized by the truth-tables of classical logic for all the connectives except negation. The negation is characterized by:

$\neg$	Value
{0,1}	1
1	0

where $0, 1$ means that the interpretation of $\neg$ ascribes two values, so negation is strictly non-deterministic. In other words, there are valuations under which a formula and its negation can be true. Again, it is rather straightforward what the n-matrix for CLuN looks like. The set of rules appropriate for CluN is defined as:
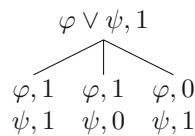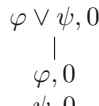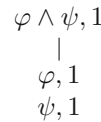
Negation 1:
$$\neg\varphi, 1$$
$$\varphi, 0 \quad \varphi, 1$$

Negation 2:
$$\neg\varphi, 0$$
$$|$$
$$\varphi, 1$$

Disjunction 1:
$$\varphi \lor \psi, 1$$
$$\begin{array}{ccc} \varphi, 1 & \varphi, 1 & \varphi, 0 \\ \psi, 1 & \psi, 0 & \psi, 1 \end{array}$$

Disjunction 2:
$$\varphi \lor \psi, 0$$
$$|$$
$$\varphi, 0$$
$$\psi, 0$$

Conjunction 1:
$$\varphi \land \psi, 1$$
$$|$$
$$\varphi, 1$$
$$\psi, 1$$

Conjunction 2:
$$\varphi \land \psi, 0$$
$$\begin{array}{ccc} \varphi, 0 & \varphi, 1 & \varphi, 0 \\ \psi, 0 & \psi, 0 & \psi, 1 \end{array}$$
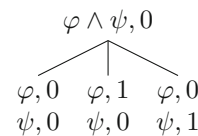
The last example of a proof system that we will formulate more interesting, since it contains a modal operator B. The logic **BAT** was presented in [26] as a logic of *informal provability*. In this logic, informal provability is treated similarly to the notion of truth in [23]. It is partial in the sense that some sentences are informally provable, other are informally refutable but this division is not exhaustive. There are sentences which are neither informally provable nor informally refutable. To indicate this **BAT** logic has three values: 0,n,1. The intended interpretation of 0 is *informally refutable*, 1 stands for *informally provable* and n stands for *neither*. **BAT** connectives are characterized by the following truth-tables:

$\overline{\neg}$	value
0	1
n	n
1	0

$\overline{\wedge}$	0	n	1
0	0	n	0
n	n	n	n
1	0	n	1

$\overline{\vee}$	0	n	1
0	0	n	1
n	n	n	n
1	1	n	1

the provability operator is characterized by the following table:

$\overline{\mathrm{B}}$	Value
1	1
{0,n}	n
0	0

Since in this case we have multiple non-deterministic connectives,[12] we will split the presentation of syntactical rules for **BAT** into subsets. **BAT** negation is described by the following:

Negation 1:
$$\neg\varphi, 1$$
$$|$$
$$\varphi, 0$$

Negation 2:
$$\neg\varphi, n$$
$$|$$
$$\varphi, n$$

Negation 3:
$$\neg\varphi, 0$$
$$|$$
$$\varphi, 1$$

As one can imagine, rules for disjunction and conjuction are somehow complicated (both connectives are non-deterministic[13]). For readability we will provide a minor simplification. Suppose that we have a formula $\varphi_1 \circ \varphi_2$. If it is clear from the truth-table that for $v(\varphi_1 \circ \varphi_2) = i$ it is sufficient that either $v(\varphi_1) = i$ or $v(\varphi_2) = i$ instead of listing all the options we will only list two options: either φ_1, i or φ_2, i. Thus, the non-deterministic connectives of **BAT** are described by the following rules.

Disjunction 1:
$$\varphi \vee \psi, 1$$

$$\varphi, 1 \quad \psi, 1 \quad \varphi, n$$
$$\psi, n$$

Disjunction 2:
$$\varphi \vee \psi, n$$

$$\varphi, n \quad \psi, n \quad \varphi, n$$
$$\psi, 0 \quad \varphi, 0 \quad \psi, n$$

---

[12]By a non-deterministic connective we mean a connective whose truth-function at least for one value is not deterministic.

[13]Some disjunctions of two undecided claims can be informally provable whereas some other remains undecided. Similar considerations apply to conjuctions. For more details see Pawlowski and Urbaniak [26].

Disjunction 3:
$$(\varphi \vee \psi), 0$$
$$|$$
$$\varphi, 0$$
$$\psi, 0$$

Conjunction 1:
$$\varphi \wedge \psi, 1$$
$$|$$
$$\varphi, 1$$
$$\psi, 1$$

Conjunction 2:
$$(\varphi \wedge \psi), n$$

$$\varphi, n \quad \varphi, n \quad \varphi, 1$$
$$\psi, n \quad \psi, 1 \quad \psi, n$$

Conjunction 3:
$$(\varphi \wedge \psi), 1$$

$$\varphi, 0 \quad \psi, 0 \quad \varphi, n$$
$$\psi, n$$

Provability 1:
$$B\varphi, 1$$
$$|$$
$$\varphi, 1$$

Provability 2:
$$B\varphi, 0$$

$$\varphi, 0 \quad \varphi, n$$

Provability 3:
$$B\varphi, n$$
$$|$$
$$\varphi, n$$

5. Reductions and Generalizations

The main feature of the proposed framework is its generality. Yet, one might worry that some proof systems generated by the framework are more complex than needed. Some rules in those proof systems, when applied to a branch on a tree, lead to multiple extensions of this branch, and some of those extensions are redundant. Consider a proof system generated by the framework for propositional logic. The application of the rule for classical disjunction results in three different extensions of a branch: one where the first disjunct is true, one where the second disjunct is true, and one where both disjuncts are true. It is fairly easy to see that the third extension of the branch does not provide any new information and it is repetitive and redundant.

The most straightforward way to solve the above issue is to provide proof-theoretical reductions. However, since the proposed framework is quite general, it is not that simple. First, for deterministic proof systems, such proof-theoretic reductions are well-known. Second, for non-deterministic cases, it is quite hard to provide any exiting reductions—non-deterministic consequence relations are too general and too weak. To be fair, there are some partial results for the labeled sequent-calculus (see Avron [1] and Avron and Lev [4], for a discussion),

but those results concern particular examples of proof systems, and not general results on classes of them.

Logicians are well-aware that most of non-deterministic consequence relations are quite weak. This is the main reason for supplementing some of non-deterministic logics with further semantical conditions put on the top of valuations. For instance, Avron and Lev [5] introduced the notion of static valuations, Kearns [22], Omori and Skurt [24], Pawlowski and Urbaniak [26] used different versions of filtration of the set of valuations to strengthen the consequence relation.

An interesting example of a logic that results from non-deterministic logic **BAT** by filtration is **CABAT**. One will see that the framework can be used to generate a proof system for **CABAT**. Moreover, it is quite easy to see how one could go about adapting the framework for different kinds of filtration procedure.[14]

Let's start with a couple of definitions. We say that a **BAT** valuation e respects the *closure condition* iff

For any $\mathcal{L}_{\mathtt{B}}$-formulas $\phi_1, \phi_2, \ldots, \phi_n, \psi$ such that

$$\phi_1, \phi_2, \ldots, \phi_n \models \psi,$$

where $\models$ is the classical consequence relation in the language of **BAT**, if $v(\mathtt{B}\phi_i) = 1$ for any $0 < i \leq n$, then $v(\mathtt{B}\psi) = 1$.

CABAT is the logic resulting from **BAT** by not considering **BAT**-valuations which do not respect the closure condition. The closure condition is quite important since **BAT** is a very weak logic, for instance disjunction in **BAT** is not symmetric.

We will use $\Gamma \twoheadrightarrow_C \phi$ to denote the consequence relation of **CABAT**. Alternatively, we can characterize **CABAT** by using filtration:

Definition 13. Let $\models$ be the classical propositional consequence relation in the language with $\mathtt{B}$. We say that a **BAT**-valuation e belongs to the filtered set of **BAT** valuations just in case the following conditions hold:

1. For any two formulas ϕ, ψ, if $\models \phi \equiv \psi$ then $v(\phi) = v(\psi)$,
2. For any tautology ϕ, $v(\phi) = 1$,
3. For any counter tautology ϕ, $v(\phi) = 0$.

Let $\Gamma \models_{Lf} \varphi$ mean that for any L-filtrated **BAT** valuation v such that $v(\psi) = 1$ for any $\psi \in \Gamma$ it follows that $v(\varphi) = 1$. Then the following holds:

Fact 1. *For any Γ, φ, $\Gamma \twoheadrightarrow_C \varphi$ iff $\Gamma \models_{Lf} \varphi$*

How to adapt the proposed framework to construct a complete proof system with respect to **CABAT**? The answer is quite straightforward. One needs to define a procedure of selecting appropriate branches on **BAT** tree, that will mimic the conditions put on the set of **BAT** valuations.

To do so, we need to introduce some additional definitions.

---

[14]It is worth pointing out that it is way harder to adapt labeled sequent calculi for this task.

Definition 14. (*Filtrated branch*) Let b be a complete open branch in a **BAT** tree. We will say that b is filtrated iff for all formulas ϕ, ψ on b the following hold:

1. If ϕ is a classical tautology, then it doesn't appear with a signature or in a negated form on b.
2. If ϕ is a classical countertautology, it appears in negated form.
3. If ϕ, ψ are classically equivalent then they appear in the same form: either both with signatures, or both negated, or both in the standard form: ϕ, ψ.

A tree appropriate for **CABAT**, is a **BAT** tree whose open not filtrated branches are removed.[15] This method can be also applied to generate proof systems for logics defined in a similar fashion [22, 24]. This shows that the framework is a bit more flexible than the labeled sequent calculus.

6. Soundness and Completeness

Suppose that we have a consequence relation L defined in terms of an n-matrix M. Let A be a set of rules describing M. In order to prove that $\vdash_A$ is sound and strongly complete with respect to L we need some further definitions.

Definition 15. We say that a valuation v is faithful to a branch b iff for all formulas φ occurring on the branch, if φ occurs with a label k, then $v(\varphi) = k$.

Fact 2. *Let v be an L-valuation and b a branch in a tree. If v is faithful to b, then for any rule of A that can be applied to b, there is an extension b' of b such that v is faithful to b'.*

Proof. by cases. Let $\varphi = \circ_i(\varphi_1, \varphi_2, \ldots, \varphi_i)$ and suppose φ, k appears on b. Let $CO_k(\varphi) = \{v_1, \ldots, v_l\}$ for some l. Thus, the application of a rule appropriate for $\circ, i$ generates l many extensions of b. By construction, each extension is correlated with a valuation from $CO_k(\varphi)$ and these valuations constitute all the possible ways of having $v(\varphi) = k$. By the definition, since v is faithful to b, $v(\varphi) = k$, so $v \in CO_k(\varphi)$ which means that v is faithful to at least one extension of b. $\square$

Fact 3. *If $\Gamma \vdash_A \varphi$ then $\Gamma \models_M \varphi$.*

Proof. By contraposition, suppose $\Gamma \not\models_M \varphi$. Then there is an L-valuation v, such that for any $\psi \in \Gamma$, $v(\psi) \in D$ and $v(\varphi) \notin D$. This valuation is faithful to at least one extension of a root of a tree for Γ, φ. By a multiple application of the above lemma we can find a valuation faithful to some branch b. It means that this branch must be open, so $\Gamma \not\vdash_A \varphi$. $\square$

Definition 16 (*Evaluation induced by b*). Let b be an open branch. An L-valuation v is induced by b iff for all propositional variables p, if they occur in the form p, k, then $v(p) = k$.

---

[15]The proof of the completeness of this proof system can be found in [25].

We proceed with a lemma which will enable us to sketch a proof of the completeness theorem.

Fact 4. *Let a branch b be open and complete. Let E be the set of L- valuations induced by b. Then there is an L-valuation $v \in E$ such that if φ occurs on b with a label k, then $v(\varphi) = k$.*

Proof. The proof is by induction on the complexity of φ. If φ is a propositional parameter, we are done by the definition of v being induced. Suppose that the theorem works for $\varphi_1, \varphi_2, \ldots \varphi_m$. We will show that it also works for $\circ^m(\varphi_1, \varphi_2 \ldots \varphi_m)$. Assume that $\circ^m(\varphi_1, \varphi_2 \ldots \varphi_m), j$ occurs on the branch. Since, b is complete a rule for $\circ^m, j$ must have been applied to it. Let $o = \mid CO_j(\circ^m(\varphi_1, \varphi_2 \ldots \varphi_m)) \mid$. Thus, we have o many extensions of b where indexes of $\varphi_1 \ldots \varphi_m$ on each extension correspond uniquely to some valuation $v \in CO_j(\circ^m(\varphi_1, \varphi_2 \ldots \varphi_m))$. By induction hypothesis if for some formula φ_i, $i < m+1$, it occurs with a number h, there is a valuation v such that $v(\varphi_i) = h$. It is clear that $v \in CO_j(\circ^m(\varphi_1, \varphi_2 \ldots \varphi_m))$, so $v(\varphi) = j$. □

Fact 5. *If $\Gamma \models_M \varphi$ then $\Gamma \vdash_A \varphi$.*

Proof. By contraposition, suppose $\Gamma \nvdash_A \varphi$. Then there is a complete open tree for Γ, φ. Let b be an open branch on that tree. Take valuation $v \in E$ which is induced by b. Under this valuation all elements of Γ have the designated value and φ does not, meaning $\Gamma \nvDash_L \varphi$. □

7. Conclusions

We presented a simple framework for constructing proof systems for finitely-valued deterministic and non-deterministic logics. Using the framework we can construct, for any finite n-matrix, a strongly complete proof system.

References

[1] Avron, Z.: Non-deterministic semantics for logical systems. In: Gabbay, D.M., Guenthner, F. (eds.) Handbook of Philosophical Logic, vol. 16. Springer, Amsterdam (2011)

[2] Avron, A., Ben-Naim, J., Konikowska, B.: Cut-free ordinary sequent calculi for logics having generalized finite-valued semantics. Log. Univ. **1**(1), 41–70 (2007)

[3] Avron, A., Konikowska, B.: Multi-valued calculi for logics based on non-determinism. Log. J. IGPL **13**(4), 365–387 (2005)

[4] Avron, A., Lev, I.: Canonical propositional Gentzen-type systems. In: Automated Reasoning: First International Joint Conference, IJCAR 2001 Siena, Italy, June 18–23, 2001 Proceedings (Lecture Notes in Computer Science), pp. 529–544. Springer, Berlin (2001)

[5] Avron, A., Lev, I.: Non-deterministic multiple-valued structures. J. Log. Comput. **15**(3), 241–261 (2005)

[6] Baaz, M., Fermüller, C.G., Salzer, G., Zach, R.: Multlog 1.0: towards an expert system for many-valued logics. In: International Conference on Automated Deduction, pp. 226–230. Springer, Berlin (1996)

[7] Batens, D.: A dynamic semantics for inconsistency-adaptive logics. Bull. Sect. Log. **27**(15–18), 5l (1998)

[8] Batens, D.: Inconsistency-adaptive logics. In: Logic at Work. Essays Dedicated to the Memory of Helena Rasiowa, pp. 445–472. Springer, Berlin (1999)

[9] Batens, D.: A survey of inconsistency-adaptive logics. In: Frontiers of Paraconsistent Logic, pp. 49–73. Research Studies Press (2000)

[10] Batens, D., De Clercq, K.: A rich paraconsistent extension of full positive logic. Log. Anal. 185–188 (2004)

[11] Bergmann, M.: An Introduction to Many-Valued and Fuzzy Logic: Semantics, Algebras, and Derivation Systems. Cambridge University Press, Cambridge (2008)

[12] Bochvar, D.A.: On a three valued calculus and its application to the analysis of contradictories. Mat. Sb. **4**(2), 287–308 (1939)

[13] Carnielli, W., Matulovic, M.: The method of polynomial ring calculus and its potentialities. Theor. Comput. Sci. **606**, 42–56 (2015)

[14] Carnielli, W.A.: Systematization of finite many-valued logics through the method of tableaux. J. Symb. Log. **52**(2), 473–493 (1987)

[15] Coniglio, M.E., Fariñas del Cerro, L., Peron, N.M.: Finite non-deterministic semantics for some modal systems. J. Appl. Non-Class. Log. **25**(1), 20–45 (2015)

[16] Coniglio, M.E., Peron, N.M.: Dugundji's theorem revisited. Log. Univ. **8**(3–4), 407–422 (2014)

[17] da Costa, N.C.A., Alves, E.H.: A semantical analysis of the calculi c_n. Notre Dame J. Form. Log. **18**(4), 621–630 (1977)

[18] Fitting, M.: A Semantics for the Logic of Proofs. Technical report, CUNY Ph.D. Program in Computer Science, TR-2003012 (2003)

[19] Halbach, V.: Axiomatic Theories of Truth. Cambridge University Press, Cambridge (2011)

[20] Horsten, L.: The Tarskian Turn. Deflationism and Axiomatic Truth. MIT Press, Cambridge (2011)

[21] Ivlev, Y.V.: A semantics for modal calculi. Bull. Sect. Log. **17**(3/4), 114–121 (1988)

[22] Kearns, J.T.: Modal semantics without possible worlds. J. Symb. Log. **46**(1), 77–86 (1981)

[23] Kripke, S.A.: Outline of a theory of truth. J. Philos. **72**(19), 690–716 (1975)

[24] Omori, Skurt: More modal semantics without possible worlds. IfCoLog J. Log. Appl. **3**(5), 815–845 (2016)

[25] Pawlowski, P.: Proof systems for BAT consequence relations. Log. J. IGPL **26**(1), 96–108 (2018)

[26] Pawlowski, P., Urbaniak, R.: Many-valued logic of informal provability: a non-deterministic strategy. Rev. Symb. Log. **11**(2), 207–223 (2018)

[27] Priest, G.: An Introduction to Non-classical Logic, 2nd edn. Cambridge University Press, Cambridge (2008)

[28] Quine, W.V.: The Roots of Reference. Lasalle, Ill., Open Court (1974)

[29] Rescher, N.: Quasi-truth-functional systems of propositional logic. J. Symb. Log. **27**(1), 1–10 (1962)

Pawel Pawlowski
University of Gdansk
Gdańsk
Poland
e-mail: `haptism89@gmail.com`

and

Ghent University
Ghent
Belgium