

Tableau Proof Systems for Justification Logics

Meghdad Ghari

School of Mathematics, Institute for Research in Fundamental Sciences (IPM),
P.O.Box: 19395-5746, Tehran, Iran
ghari@ipm.ir

Abstract. In this paper we present tableau proof systems for various justification logics. We establish that the tableau systems are sound and complete with respect to Mkrtychev models. We show the subformula property for our tableaux, and prove the decidability of justification logics with finite constant specifications. Also sequent calculi with analytic cuts are introduced and the subformula property are shown.

Keywords: Justification logics, Tableaux, Subformula property, decidability, sequent calculus, Analytic cut

1 Introduction

Justification logics are modal-like logics that provide a framework for reasoning about epistemic justifications (see [3, 4, 11] for a survey). The language of justification logics extends the language of propositional logic by justification terms and expressions of the form $t : A$, with the intended meaning “$t$ is a justification for $A$”. Justification terms are constructed from variables and constants by means of various operations. The first logic in the family of justification logics, the Logic of Proofs $LP$, was introduced by Artemov in [1, 2]. The logic of proofs is a counterpart of modal logic $S4$. Other logics of this kind have been introduced so far (cf. [13]). In this paper we deal only with those justification logics which are counterparts of normal modal logics between $K$ and $S5$.

Various tableau proof systems have been developed for the logic of proofs (see [8, 9, 15, 16]). However, it seems the only analytic tableau proof system is Finger’s KE tableaux for the logic of proofs [8]. Moreover, most of the justification logics still lack tableau proof systems.

The aim of this paper is to present tableau proof systems for various justification logics. For each justification logic we present three tableau proof systems. The difference between these systems is that they use different tableau rules for the basic justification logic $J$. All tableau proof systems are sound and complete with respect to Mkrtychev models of justification logics.

In the first formulation (see Section 3.1), the rules of the tableau system for $J$ is similar to the ($J$-part) tableau rules given by Renne in [16] for $LP$. Renne’s tableaux corresponds to the Artemov’s sequent calculus for $LP$ in [2]. The subformula property fails for both the tableaux and the sequent calculus of $LP$, and also fails for the tableaux of justification logics introduced in this section.

In the second formulation (see Section 3.2), the rules of the tableau system for $J$ is similar to the ($J$-part) tableau rules given by Finger in [8] for $LP$. Finger’s tableau
system has KE tableau rules (cf. [7]) in its propositional part. KE tableaux have linear tableau rules for propositional connectives (different from ordinary propositional tableau rules of Smullyan [17]), and the cut rule (PB). Following Finger [8], by restricting the applications of (PB) to analytic ones, we obtain analytic tableaux for justification logics. We give a definition of subformulas in the context of justification logics, and prove that our tableau systems enjoy the subformula property. Finally, using KE tableaux, decidability of justification logics (with finite constant specifications) are shown.

In Section 3.3 we present a tableau system for J, the third formulation, which is similar to its KE tableau system but with ordinary propositional rules. Our propositional tableau rules are the ordinary ones given by Smullyan [17], and justification tableau rules are similar to those introduced by Finger [8]. This enables us to have analytic tableaux for justification logics with ordinary propositional tableau rules, which transforms into analytic sequent calculi in Section 3.4. Our sequent calculi have an analytic cut rule, and thus the subformula property holds.

2 Justification logics

The language of justification logics is an extension of the language of propositional logic by the formulas of the form \( t : F \), where \( F \) is a formula and \( t \) is a justification term. Justification terms (or terms for short) are built up from (justification) variables \( x, y, z, \ldots \) and (justification) constants \( a, b, c, \ldots \) using several operations depending on the logic: (binary) application ‘·’, (binary) sum ‘+’, (unary) verifier ‘!', (unary) negative verifier ‘?', and (unary) weak negative verifier ‘¯?’. Subterms of a term are defined in the usual way: \( s \) is a subterm of \( s, s + t, t + s, s \cdot t, !s, ?s, \) and \( ?s \).

Justification formulas are constructed from a countable set of propositional variables, denoted \( \mathcal{P} \), by the following grammar:

\[
A ::= p \mid \bot \mid \neg A \mid A \rightarrow A \mid t : A,
\]

where \( p \in \mathcal{P} \) and \( t \) is a justification term. Other Boolean connectives are defined as usual.

We now begin with describing the axiom schemes and rules of the basic justification logic \( J \), and continue with other justification logics. The basic justification logic \( J \) is the weakest justification logic we shall be discussing. Other justification logics are obtained by adding certain axiom schemes to \( J \).

**Definition 2.1.** Axioms schemes of \( J \) are:

- **Taut.** All propositional tautologies,
- **Sum.** \( s : A \rightarrow (s + t) : A \), \( s : A \rightarrow (t + s) : A \),
- **jK.** \( s : (A \rightarrow B) \rightarrow (t : A \rightarrow (s \cdot t) : B) \).

Other justification logics are obtained by adding the following axiom schemes to \( J \) in various combinations:

- **jT.** \( t : A \rightarrow A \).
- **jD.** \( t : \bot \rightarrow \bot \).
j4. \( t : A \rightarrow !t : t : A \),
jB. \( \neg A \rightarrow ?t : \neg t : A \),
j5. \( \neg t : A \rightarrow ?t : \neg t : A \).

All justification logics have the inference rule Modus Ponens, and the Iterated Axiom Necessitation rule:

\[ \vdash c_i : c_{i-1} : \ldots : c_1 : A, \text{ where } A \text{ is an axiom instance of the logic, } c_i \text{'s are arbitrary justification constants and } n \geq 1. \]

In what follows, JL denotes any of the justification logics defined in Definition 2.1, unless stated otherwise. The language of each justification logic JL includes those operations on terms that are present in its axioms. \( Tm_{JL} \) and \( Fm_{JL} \) denote the set of all terms and the set of all formulas of JL respectively. Moreover, the name of each justification logic is indicated by the list of its axioms. For example, JT4 is the extension of J by axioms jT and j4, in the language containing term operations \( \cdot, +, \) and \( ! \). JT4 is usually called the logic of proofs LP.

Definition 2.2. A constant specification CS for JL is a set of formulas of the form \( c_i : c_{i-1} : \ldots : c_1 : A \), where \( n \geq 1 \), \( c_i \)'s are justification constants and \( A \) is an axiom instance of JL.

Let \( JL_{CS} \) be the fragment of JL where the Iterated Axiom Necessitation rule only produces formulas from the given CS.

In the remaining of this section, we recall the definitions of M-models for justification logics (see [14, 13]).

Definition 2.3. An M-model \( M = (\mathcal{E}, \mathcal{V}) \) for justification logic \( JL_{CS} \) consists of a valuation \( \mathcal{V} : \mathcal{P} \rightarrow \{0, 1\} \) and an admissible evidence function \( \mathcal{E} : Tm_{JL} \rightarrow 2^{Fm_{JL}} \) meeting the following conditions:

\[ \mathcal{E}_1. A \rightarrow B \in \mathcal{E}(s) \text{ and } A \in \mathcal{E}(t) \text{ implies } B \in \mathcal{E}(s \cdot t). \]
\[ \mathcal{E}_2. \mathcal{E}(s) \cup \mathcal{E}(t) \subseteq \mathcal{E}(s + t). \]
\[ \mathcal{E}_3. c : F \in CS \text{ implies } F \in \mathcal{E}(c). \]

Definition 2.4. For an M-model \( M = (\mathcal{E}, \mathcal{V}) \) the forcing relation \( \models \) is defined as follows:

1. \( M \not\models \bot \),
2. \( M \models p \text{ iff } \mathcal{V}(p) = 1, \text{ for } p \in \mathcal{P} \),
3. \( M \models \neg A \text{ iff } M \not\models A \),
4. \( M \models A \rightarrow B \text{ iff } M \not\models A \text{ or } M \models B \),
5. \( M \models t : A \text{ iff } A \in \mathcal{E}(t) \).

If \( M \models F \) then it is said that \( F \) is true in \( M \) or \( M \) satisfies \( F \).

In order to define M-models for other justification logics of Definition 2.1 certain additional conditions should be imposed on the M-model.

Definition 2.5. An M-model \( M = (\mathcal{E}, \mathcal{V}) \) for justification logic \( JL_{CS} \) is an M-model for \( JL_{CS} \) such that:
– if \( JL \) contains axiom \( jT \), then for all \( t \in \text{m}_{\text{JL}} \) and \( A \in \text{F}_{\text{m}_{\text{JL}}} \):
\[ E4. \ A \in \mathcal{E}(t) \implies M \models A. \]

– if \( JL \) contains axiom \( jD \), then for all \( t \in \text{m}_{\text{JL}} \):
\[ E5. \ \bot \notin \mathcal{E}(t). \]

– if \( JL \) contains axiom \( j4 \), then for all \( t \in \text{m}_{\text{JL}} \) and \( A \in \text{F}_{\text{m}_{\text{JL}}} \):
\[ E6. \ A \in \mathcal{E}(t) \implies t : A \in \mathcal{E}(\bar{t}). \]

– if \( JL \) contains axiom \( jB \), then for all \( t \in \text{m}_{\text{JL}} \) and \( A \in \text{F}_{\text{m}_{\text{JL}}} \):
\[ E7. \ M \not\models A \implies \neg t : A \in \mathcal{E}(\bar{t}). \]

– if \( JL \) contains axiom \( j5 \), then for all \( t \in \text{m}_{\text{JL}} \) and \( A \in \text{F}_{\text{m}_{\text{JL}}} \):
\[ E8. \ A \notin \mathcal{E}(t) \implies \neg t : A \in \mathcal{E}(\bar{t}). \]

By a \( \text{JL}_{\text{CS}} \)-model we mean an \( M \)-model for justification logic \( \text{JL}_{\text{CS}} \). A \( \text{JL} \)-formula \( F \) is \( \text{JL}_{\text{CS}} \)-valid if it is true in every \( \text{JL}_{\text{CS}} \)-model. For a set \( S \) of formulas, \( M \models S \) provided that \( M \models F \) for all formulas \( F \) in \( S \). Note that given a constant specification \( \mathcal{CS} \) for \( JL \), and a model \( M \) of \( \text{JL}_{\text{CS}} \) we have \( M \models \mathcal{CS} \) (in this case it is said that \( M \) respects \( \mathcal{CS} \)).

The proof of soundness and completeness theorems for all justification logics of Definition 2.1 are given in [13].

**Theorem 2.1.** Let \( JL \) be one of the justification logics of Definition 2.1, and \( \mathcal{CS} \) be a constant specification for \( JL \). Then a \( JL \)-formula \( F \) is provable in \( JL_{\text{CS}} \) iff \( F \) is \( JL_{\text{CS}} \)-valid.

### 3 Tableaux

In this section we present three tableau proof systems for each justification logic of Definition 2.1. The difference between these systems is that they use different tableau rules for the basic justification logic \( J \). The rules of our tableau system for \( J \) in Section 3.1 is similar to that given in [16], and in Section 3.2 is similar to KE tableaux given in [8]. In Section 3.3 we present a tableau system for \( J \) which is similar to its KE tableau system but with ordinary propositional rules. In Section 3.4 we transform these latter tableau systems into sequent calculi.

#### 3.1 Tableaux I

Tableau proof systems for the logic of proofs are given in [9, 15, 16]. In this section we present similar tableaux for all justification logics.

A \( \text{JL}_{\text{CS}} \)-tableau for a formula is a binary tree with that formula at the root constructed by applying \( \text{JL}_{\text{CS}} \)-tableau rules from Table 1. For extensions of \( J \), tableau rules corresponding to axioms from Table 2 should be added to \( \text{JL}_{\text{CS}} \)-tableau rules.

For example, the tableau proof system of the logic of proofs \( \text{LP} \) is obtained by adding the rules \((T:)\) and \((F!)\) to the tableau rules of \( J \). For a justification logic \( JL \), a tableau branch of a \( \text{JL}_{\text{CS}} \)-tableau closes if one of the following holds:

1. Both \( A \) and \( \neg A \) occurs in the branch, for some formula \( A \).
2. \( \bot \) occurs in the branch.
3. \( \neg c : F \) occurs in the branch, for some \( c : F \in \mathcal{CS} \).
A tableau closes if all branches of the tableau close. A $\mathcal{JL}_{CS}$-tableau proof for formula $F$ is a closed tableau beginning with $\neg F$ (the root of the tableau) using only tableau rules of $\mathcal{JL}_{CS}$. A $\mathcal{JL}_{CS}$-tableau for a finite set $S$ of $\mathcal{JL}$-formulas begins with a single branch whose nodes consist of the formulas of $S$ as roots.

Example 3.1. We give a $\mathcal{JL}_{CS}$-tableau proof of $x : A \rightarrow c \cdot x : (B \rightarrow A)$, where $CS$ contains $c : (A \rightarrow (B \rightarrow A))$.

\begin{align*}
1. & \neg(x : A \rightarrow c \cdot x : (B \rightarrow A)) \\
2. & x : A \\
3. & \neg c \cdot x : (B \rightarrow A) \\
4. & \neg c : (A \rightarrow (B \rightarrow A)) \\
5. & \neg x : A
\end{align*}

Formulas 2 and 3 are from 1 by rule $(F \rightarrow)$, 4 and 5 are from 3 by rule $(F \cdot)$.

Remark 3.1. For those justification logics that contain axiom jD, instead of the rule $(T : \bot)$ the following closing condition can be used: a tableau branch is closed if $t : \bot$, for some term $t$, occurs in it.

| Propositional rules: |
|----------------------|
| $\neg\neg A \rightarrow A$ \hspace{1cm} \begin{array}{c} (F\neg) \end{array} |
| $\neg(A \rightarrow B) \rightarrow (A \rightarrow B)$ \hspace{1cm} \begin{array}{c} (F \rightarrow) \end{array} |
| $\neg B \rightarrow A$ \hspace{1cm} \begin{array}{c} (T \rightarrow) \end{array} |

| Justification rules: |
|----------------------|
| $\neg t + s : A \rightarrow \neg t : A$ \hspace{1cm} \begin{array}{c} (F+) \end{array} |
| $\neg t : A \rightarrow \neg s : (A \rightarrow B) \rightarrow \neg t : A$ \hspace{1cm} \begin{array}{c} (F\cdot) \end{array} |

Table 1. Tableau rules for basic justification logic $J$.

Let us show the soundness and completeness of tableau systems with respect to M-models. The following lemma is a consequence of the definition of M-models.

Lemma 3.1. Let $\pi$ be any branch of a $\mathcal{JL}_{CS}$-tableau and $\mathcal{M}$ be a $\mathcal{JL}_{CS}$-model that satisfies all the formulas occur in $\pi$. If a $\mathcal{JL}_{CS}$-tableau rule is applied to $\pi$, then it produces at least one extension $\pi'$ such that $\mathcal{M}$ satisfies all the formulas occur in $\pi'$.

Theorem 3.1 (Soundness). If $A$ has a $\mathcal{JL}_{CS}$-tableau proof, then it is $\mathcal{JL}_{CS}$-valid.
Table 2. Justification axioms with corresponding tableau rules.

| Justification axiom | Tableau rule |
|---------------------|--------------|
| jT. \( t : A \to A \) | \( t : A \) (T :) |
| jD. \( t : \bot \to \bot \) | \( t : \bot \) (T :\) |
| j4. \( t : A \to !t : A \) | \( \neg !t : A \) (F !) |
| jB. \( \neg A \to ?t : \neg A \) | \( \neg ?t : \neg A \) (F ?) |
| j5. \( \neg t : A \to ?t : \neg t : A \) | \( \neg ?t : \neg t : A \) (F ?) |

Proof. If \( A \) is not \( \mathcal{JCS} \)-valid, then there is a \( \mathcal{JCS} \)-model \( \mathcal{M} \) such that \( \mathcal{M} \models \neg A \). Thus by Lemma 3.5, there is no closed \( \mathcal{JCS} \)-tableau beginning with \( \neg A \). Therefore, \( A \) does not have a \( \mathcal{JCS} \)-tableau proof. \( \square \)

Next we shall prove the completeness theorem, by making use of maximal consistent sets.

Definition 3.1. Suppose \( \Gamma \) is a set of \( \mathcal{JL} \)-formulas. \( \Gamma \) is tableau \( \mathcal{JCS} \)-consistent if there is no closed tableau beginning with any finite subset of \( \Gamma \). \( \Gamma \) is maximal if for every \( \mathcal{JL} \)-formula \( A \) either \( A \) or \( \neg A \) is in \( \Gamma \).

It is known that every \( \mathcal{JCS} \)-consistent set has a maximally \( \mathcal{JCS} \)-consistent extension (Lindenbaum Lemma).

It is easy to show that maximally \( \mathcal{JCS} \)-consistent sets are downward closed, that is maximally \( \mathcal{JCS} \)-consistent sets are closed under \( \mathcal{JCS} \)-tableau rules. For a non-branching rule like

\[
\frac{\alpha_1 \alpha_2}{\alpha}
\]

this means that if \( \alpha \) is in a maximally \( \mathcal{JCS} \)-consistent set \( \Gamma \), then both \( \alpha_1 \in \Gamma \) and \( \alpha_2 \in \Gamma \). For a branching rule like

\[
\frac{\beta}{\beta_1 | \beta_2}
\]

this means that if \( \beta \) is in a maximally \( \mathcal{JCS} \)-consistent set \( \Gamma \), then \( \beta_1 \in \Gamma \) or \( \beta_2 \in \Gamma \).

Lemma 3.2. Suppose \( \Gamma \) is a maximally \( \mathcal{JCS} \)-consistent set. Then \( \Gamma \) is closed under \( \mathcal{JCS} \)-tableau rules.

Proof. The proof for propositional rules \( (F \neg) \), \( (F \to) \), and \( (T \to) \) are standard. For justification rules, we detail the proof only for the rule \( (F \cdot) \). The proof for the other tableau rules is similar.

Suppose \( \Gamma \) is a maximally \( \mathcal{JCS} \)-consistent set and \( \neg s \cdot t : B \in \Gamma \). Suppose towards a contradiction that \( \neg s : (A \to B) \notin \Gamma \) and \( \neg t : A \notin \Gamma \). Since \( \Gamma \) is maximal, we
have \( s : (A \rightarrow B) \) and \( t : A \) are in \( \Gamma \). Now it is easy to see that the \( \mathcal{L}_{CS} \)-tableau beginning with the finite subset \( \{ \neg s \cdot t : B, s : (A \rightarrow B), t : A \} \) of \( \Gamma \) closes, and hence \( \Gamma \) is not \( \mathcal{L}_{CS} \)-consistent, contra with the assumption. □

**Definition 3.2.** Given a maximally \( \mathcal{L}_{CS} \)-consistent set \( \Gamma \), the canonical model \( \mathcal{M} = (\mathcal{E}, \mathcal{V}) \) with respect to \( \Gamma \) is defined as follows:

- \( \mathcal{E}(t) = \{ A \mid t : A \in \Gamma \} \).
- \( \mathcal{V}(p) = 1 \) iff \( p \in \Gamma \), where \( p \in \mathcal{P} \).

**Lemma 3.3 (Truth Lemma).** Suppose \( \Gamma \) is a maximally \( \mathcal{L}_{CS} \)-consistent set and \( \mathcal{M} = (\mathcal{E}, \mathcal{V}) \) is the canonical model with respect to \( \Gamma \). Then for every \( \mathcal{L} \)-formula \( F \):

\[
\mathcal{M} \vDash F \iff F \in \Gamma.
\]

*Proof.* By induction on the complexity of \( F \). The base case and the propositional inductive cases are standard. The proof for the case that \( F = t : A \) is as follows. By definition of forcing and definition of evidence function \( \mathcal{E} \) in canonical model we have \( \mathcal{M} \vDash t : A \) if and only if \( A \in \mathcal{E}(t) \) if and only if \( t : A \in \Gamma \). □

**Lemma 3.4.** Given a maximally \( \mathcal{L}_{CS} \)-consistent set \( \Gamma \), the canonical model \( \mathcal{M} = (\mathcal{E}, \mathcal{V}) \) with respect to \( \Gamma \) is a \( \mathcal{L}_{CS} \)-model.

*Proof.* Suppose \( \Gamma \) is a maximally \( \mathcal{L}_{CS} \)-consistent set and \( \mathcal{M} = (\mathcal{E}, \mathcal{V}) \) is the canonical model with respect to \( \Gamma \). We shall show that the admissible evidence function \( \mathcal{E} \) satisfies the corresponding conditions stated in the definition of \( \mathcal{L}_{CS} \)-models.

For \( \mathcal{E}1 \), suppose that \( A \in \mathcal{E}(t) \) and \( A \rightarrow B \in \mathcal{E}(s) \). We have to show that \( B \in \mathcal{E}(s \cdot t) \). By definition of \( \mathcal{E} \), \( t : A \in \Gamma \) and \( s : (A \rightarrow B) \in \Gamma \). By maximality of \( \Gamma \), \( \neg t : A \notin \Gamma \) and \( \neg s : (A \rightarrow B) \notin \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \( (F^+) \), and hence \( \neg s \cdot t : B \notin \Gamma \). Therefore, \( s \cdot t : B \in \Gamma \). Hence, by definition of \( \mathcal{E} \), \( B \in \mathcal{E}(s \cdot t) \).

For \( \mathcal{E}2 \), suppose that \( A \in \mathcal{E}(s) \cup \mathcal{E}(t) \). We have to show that \( A \in \mathcal{E}(s + t) \). If \( A \in \mathcal{E}(s) \), then \( s : A \in \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \( (F+) \), and hence \( \neg s + t : A \notin \Gamma \). Therefore, \( s + t : A \in \Gamma \), and hence \( A \in \mathcal{E}(s + t) \). The case that \( A \in \mathcal{E}(t) \) is similar.

For \( \mathcal{E}3 \), suppose that \( c : F \in \mathcal{CS} \). We have to show that \( F \in \mathcal{E}(c) \). Since \( \Gamma \) is \( \mathcal{L}_{CS} \)-consistent, \( \neg c : F \notin \Gamma \) and hence \( c : F \in \Gamma \). Thus \( F \in \mathcal{E}(c) \).

For \( \mathcal{E}4 \), in the case that \( \mathcal{J} \) contains axiom \( jT \), suppose that \( A \in \mathcal{E}(t) \). We have to show that \( \mathcal{M} \vDash A \). From \( A \in \mathcal{E}(t) \) we have \( t : A \in \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \( (T^+) \), and hence \( A \in \Gamma \). By Truth Lemma, we have \( \mathcal{M} \vDash A \).

For \( \mathcal{E}5 \), in the case that \( \mathcal{J} \) contains axiom \( jD \), note that \( t : \bot \notin \Gamma \), for any term \( t \in \mathcal{T}m_{\mathcal{J}} \). Otherwise, by Lemma 3.6, \( \Gamma \) is closed under rule \( (T^+) \), and hence \( \bot \notin \Gamma \), which would contradict the \( \mathcal{L}_{CS} \)-consistency of \( \Gamma \). Thus \( \bot \notin \mathcal{E}(t) \).

For \( \mathcal{E}6 \), in the case that \( \mathcal{J} \) contains axiom \( j4 \), suppose that \( A \in \mathcal{E}(t) \). We have to show that \( t : A \in \mathcal{E}(t) \). By definition of \( \mathcal{E} \), \( t : A \in \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \( (F!) \), and hence \( \neg \neg t : t : A \notin \Gamma \). Therefore, \( t : A \in \Gamma \), and hence \( t : A \in \mathcal{E}(t) \).

For \( \mathcal{E}7 \), in the case that \( \mathcal{J} \) contains axiom \( jB \), suppose that \( \mathcal{M} \not\vDash A \). We have to show that \( \neg t : A \in \mathcal{E}(t) \). By Truth Lemma, \( A \notin \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \( (F^?) \), and hence \( \neg \neg t : \neg t : A \notin \Gamma \). Therefore, \( t : \neg t : A \in \Gamma \), and hence \( \neg t : A \in \mathcal{E}(t) \).
For \( E_8 \), in the case that \( \mathcal{J} \) contains axiom j5, suppose that \( A \notin E(\ell) \). We have to show that \( \neg t : A \in E(\ell) \). By definition of \( E \), \( t : A \notin \Gamma \). By Lemma 3.6, \( \Gamma \) is closed under rule \((F?)\), and hence \( \neg \ell t : \neg t : A \notin \Gamma \). Therefore, \( \ell t : \neg t : A \in \Gamma \), and hence \( \neg t : A \in E(\ell) \).

\[ \Box \]

**Theorem 3.2 (Completeness).** If \( A \) is \( \mathcal{J}_{CS} \)-valid, then it has a \( \mathcal{J}_{CS} \)-tableau proof.

**Proof.** If \( A \) does not have a \( \mathcal{J}_{CS} \)-tableau proof, then \( \{ \neg A \} \) is a \( \mathcal{J}_{CS} \)-consistent set and can be extended to a maximal \( \mathcal{J}_{CS} \)-consistent set \( \Gamma \). Since \( \neg A \in \Gamma \), by Truth Lemma, \( M \not\models A \), where \( M \) is the canonical model of \( \mathcal{J}_{CS} \) with respect to \( \Gamma \). Therefore \( A \) is not \( \mathcal{J}_{CS} \)-valid.

Clearly the rule \((F?)\) in any \( \mathcal{J}_{CS} \)-tableau system

\[
\begin{array}{c}
\neg s \cdot t : B \\
\hline
\neg s : (A \rightarrow B) | \neg t : A
\end{array}
\]

is not analytic, because the formula \( A \) in the conclusion of the rule could be a new formula from the outside of the proof. In the following sections we replace this rule with an analytic rule.

### 3.2 Tableaux II

A tableau proof system for the logic of proofs based on KE tableaux (cf. [5, 6, 7]) are presented in [8]. In this section we give KE tableaux for all justification logics.

Let us first recall the definition of subformulas of a formula from [8].

**Definition 3.3.** Let \( CS \) be a constant specification for \( \mathcal{J} \), and let \( A \) and \( B \) be \( \mathcal{J} \)-formulas. \( A \) is a \( \mathcal{J}_{CS} \)-subformula of \( B \) if one of the following clauses holds:

1. \( A = B \),
2. \( B = \neg F \), and \( A \) is a \( \mathcal{J}_{CS} \)-subformula of \( F \),
3. \( B = F \rightarrow G \), and \( A \) is a \( \mathcal{J}_{CS} \)-subformula of \( F \) or \( G \),
4. \( B = \ell : F \), and \( A \) is a \( \mathcal{J}_{CS} \)-subformula of \( F \),
5. \( A = \ell : F \), where \( \ell \) is a subterm of a term in \( B \) and \( F \) is either a \( \mathcal{J}_{CS} \)-subformula of \( B \).
6. \( A \) is a \( \mathcal{J}_{CS} \)-subformula of \( F \), for some \( c_{i_n} : c_{i_{n-1}} : \ldots : c_i : F \in CS \) and constants \( c_{i_n}, c_{i_{n-1}}, \ldots, c_i \) occurring in \( B \).

\( A \) is a weak \( \mathcal{J}_{CS} \)-subformula of \( B \) if \( A \) is either a \( \mathcal{J}_{CS} \)-subformula of \( B \) or the negation of a \( \mathcal{J}_{CS} \)-subformula of \( B \).

KE tableau rules for basic justification logic \( J \) are given in Table 3. Other KE tableau systems for justification logics are obtained by adding rules from Table 2 to the KE tableau rules of \( J \) depending on the axioms of the logic. For a justification logic \( \mathcal{J} \) its KE tableau proof system is denoted by \( KEJ \). Note that in KE tableaux the only branching rule is \((PB)\), and all other rules are linear. The formula \( A \) in the conclusion of \((PB)\) is called the \( PB \)-formula. Moreover, the rules \((T^\cdot)\) and \((PB)\) have restrictions on their applications (see Table 3).

From Definition 3.3 it is obvious that the following rule is admissible in \( KEJ_{CS} \):
Propositional rules:

\[ \neg \neg A \quad A \quad \neg (A \rightarrow B) \quad A \quad \neg B \quad A \rightarrow B \]

Justification rules:

\[ \neg t + s : A \quad (F+) \quad s : (A \rightarrow B) \quad \neg s : A \quad t : A \quad s \cdot t : B \quad (T\cdot) \]

Principle of Bivalence:

\[ A \mid \neg A \quad (PB) \]

In \((T \rightarrow)\) the term \(s \cdot t\) should occur in the root of the branch.

In \((PB)\) the \(PB\)-formula \(A\) is a \(JL_{CS}\)-subformula of the root of the branch.

| Table 3. KE Tableau rules for basic justification logic J. |
|----------------------------------------------------------|
| \[ c_{i_n} : c_{i_{n-1}} : \ldots : c_{i_1} : A \mid \neg c_{i_n} : c_{i_{n-1}} : \ldots : c_{i_1} : A \quad (PB) \] |

where \(c_{i_n} : c_{i_{n-1}} : \ldots : c_{i_1} : A \in CS\) and all constants \(c_{i_n}, c_{i_{n-1}}, \ldots, c_{i_1}\) occur in the root of the tableau.

The original formulation of KE tableaux (see [7]) has in addition the following rule for implication:

\[ A \rightarrow B \quad \neg B \quad \neg A \]

This rule is derivable from the other rules as the following derivation shows:

\[ A \rightarrow B \quad \neg B \quad / \quad \backslash \quad \neg A \quad A \quad B \quad \times \]

(Note that the displayed application of rule \((PB)\) on the formula \(A\) in the above derivation is justified by the fact that \(A\) is a \(JL_{CS}\)-subformula of the premise \(A \rightarrow B\) which in turn is a \(JL_{CS}\)-subformula of the root of the tableau, by Theorem 3.5.)

Example 3.2. We give a KE\(_{JCS}\)-tableau proof of \(x : A \rightarrow c \cdot x : (B \rightarrow A)\), where \(CS\) contains \(c : (A \rightarrow (B \rightarrow A))\).
1. \( \neg(x : A \to c \cdot x : (B \to A)) \)

2. \( x : A \)

3. \( \neg c \cdot x : (B \to A) \)

4. \( c : (A \to (B \to A)) \)

5. \( \neg c : (A \to (B \to A)) \)

Formulas 2 and 3 are from 1 by rule \((F \to)\), 4 and 5 are obtained by \((PB)\), and 5 from 2 and 4 by rule \((T \cdot)\). Note that in the application of \((PB)\) the \(PB\)-formula \(c : (A \to (B \to A))\) is a \(JCS\)-subformula of the root, and in the application of \((T \cdot)\) the term \(c \cdot x\) occurs in the root of the branch.

Soundness of KE tableau systems is shown similar to that given in the previous section.

**Lemma 3.5.** Let \( \pi \) be any branch of a \(KEJLCS\)-tableau and \( M \) be a \(JLCS\)-model that satisfies all the formulas occur in \( \pi \). If a \(KEJLCS\)-tableau rule is applied to \( \pi \), then it produces at least one extension \( \pi' \) such that \( M \) satisfies all the formulas occur in \( \pi' \).

**Theorem 3.3 (Soundness).** If \( A \) has a \(KEJLCS\)-tableau proof, then it is \(JLCS\)-valid.

Completeness is shown by means of the same canonical model construction as the previous section. The definition of the canonical model is similar to Definition 3.2, and the Truth Lemma can be proved similarly. It is easy to show that maximally \(JLCS\)-consistent sets are closed under \(KEJL\)-tableau rules. For the rule \((T \cdot)\) (although this rule has a condition on its applications) this means that if \( s : (A \to B), t : A \in \Gamma \), then \( s \cdot t : B \in \Gamma \).

**Lemma 3.6.** Suppose \( \Gamma \) is a maximally \(JLCS\)-consistent set. Then \( \Gamma \) is closed under \(KEJLCS\)-tableau rules.

**Proof.** We detail the proof only for the rule \((T \cdot)\). Suppose \( \Gamma \) is a maximally \(JLCS\)-consistent set, and \( s : (A \to B), t : A \in \Gamma \). We have to show that \( s \cdot t : B \in \Gamma \). Suppose towards a contradiction that \( s \cdot t : B \notin \Gamma \). Since \( \Gamma \) is maximal, we have \( \neg s \cdot t : B \in \Gamma \). Thus, \( s \cdot t \) occurs in \( \Gamma \). Now it is easy to see that the \(JLCS\)-tableau beginning with the finite subset \( \{ s : (A \to B), t : A, \neg s \cdot t : B \} \) of \( \Gamma \) closes, and hence \( \Gamma \) is not \(JLCS\)-consistent, contra with the assumption. \(\square\)

**Lemma 3.7.** Given a maximally \(JLCS\)-consistent set \( \Gamma \), the canonical model \( M = (E, V) \) with respect to \( \Gamma \) is a \(JLCS\)-model.

**Proof.** Suppose \( \Gamma \) is a maximally \(JLCS\)-consistent set and \( M = (E, V) \) is the canonical model with respect to \( \Gamma \). We only show that the admissible evidence function \( E \) satisfies the condition \(E1\).
Suppose that $A \in \mathcal{E}(t)$ and $A \rightarrow B \in \mathcal{E}(s \cdot t)$. We have to show that $B \in \mathcal{E}(s \cdot t)$. By definition of $\mathcal{E}$, $t : A \in \Gamma$ and $s : (A \rightarrow B) \in \Gamma$. By Lemma 3.6, $\Gamma$ is closed under rule $(T \cdot)$, hence $s \cdot t : B \in \Gamma$. Therefore, by definition of $\mathcal{E}$, $B \in \mathcal{E}(s \cdot t)$. \hfill \Box

**Theorem 3.4 (Completeness)**. If $A$ is $\mathcal{JL}_{CS}$-valid, then it has a $\mathcal{JL}_{CS}$-tableau proof.

The completeness theorem also shows that the unrestricted cut rule, i.e. $(PB)$ without any condition on the $PB$-formula, is admissible in all $\mathcal{KEJL}_{CS}$-tableau systems. Note that the unrestricted cut rule violates the subformula property.

Inspection of all KE tableau rules in Tables 2 and 3 shows that in a $\mathcal{KEJL}_{CS}$-tableau every expanded formula of a rule is a weak $\mathcal{JL}_{CS}$-subformula of the root of the tableau.

**Theorem 3.5 (Subformula property)**. Every formula in a $\mathcal{KEJL}_{CS}$-tableau proof is a weak $\mathcal{JL}_{CS}$-subformula of the root of the tableau.

Note that for a finite constant specification $CS$, the set of all $\mathcal{JL}_{CS}$-subformulas of a formula is finite. Thus, in order to prove a given formula we need to search for a finite number of formulas. Furthermore, the number of applications of each tableau rule is finite too. In fact, the only tableau rules that increase the complexity of formulas are $(PB)$ and $(T \cdot)$. For a finite constant specification, the set of all $\mathcal{JL}_{CS}$-subformulas of a formula is finite, and hence $(PB)$ could apply only finitely many times. Moreover, as shown in [8], the condition given in Table 1 for the rule $(T \cdot)$ avoids infinite applications of this rule. Thus we have

**Theorem 3.6**. Given any justification logic $\mathcal{JL}$ and finite constant specification $CS$ for $\mathcal{JL}$, $\mathcal{KEJL}_{CS}$-tableaux always terminate. Therefore justification logic $\mathcal{JL}_{CS}$ is decidable.

### 3.3 Tableaux III

In this section we give a tableau proof system for $\mathcal{J}$ which is similar to its KE tableaux, with the difference that its propositional logic rules is the same as Smullyan’s rules [17]. Tableau rules for $\mathcal{J}$ are given in Table 4. We denote this tableau system by $\mathcal{JT}$. For extensions of $\mathcal{J}$, tableau rules corresponding to axioms from Table 2 should be added to the rules of $\mathcal{JT}$. For a justification logic $\mathcal{JL}$, the resulting tableau system is denoted by $\mathcal{JL}_{T}$.

The proof of soundness and completeness theorems and the subformula property for these tableaux are similar to those of KE tableaux.

**Theorem 3.7 (Soundness and Completeness)**. $A$ is $\mathcal{JL}_{CS}$-valid if and only if it has a $\mathcal{JL}_{T}$-tableau proof.

**Theorem 3.8 (Subformula property)**. Every formula in a $\mathcal{JL}_{T}$-tableau proof is a weak $\mathcal{JL}_{CS}$-subformula of the root of the tableau.

**Theorem 3.9**. Given any justification logic $\mathcal{JL}$ and finite constant specification $CS$ for $\mathcal{JL}$, $\mathcal{JL}_{T}$-tableaux always terminate.
Propositional rules:

- \( \neg \neg A \rightarrow A \) 
- \( A \rightarrow \neg (A \rightarrow B) \)
- \( \neg B \rightarrow \neg (A \rightarrow B) \)
- \( A \rightarrow \neg B \)
- \( \neg A \rightarrow B \rightarrow T \rightarrow \neg A \rightarrow B \)

Justification rules:

- \( \neg t + s : A \rightarrow (F+ \rightarrow \neg t : A \rightarrow \neg t : A \rightarrow s : (A \rightarrow B) \rightarrow t : A \rightarrow s \cdot t : B \rightarrow (T \cdot) \)

Principle of Bivalence:

- \( A \rightarrow \neg A \) 

In \((T \cdot)\) the term \( s \cdot t \) should occur in the root of the branch.

In \((PB)\) the \( PB \)-formula \( A \) is a \( JLCS \)-subformula of the root of the branch.

| Table 4. Tableau rules for basic justification logic \( J \). |

3.4 Sequent calculi

In this section we present sequent calculi that corresponds to our tableau systems \( JLCS \). In order to show the correspondence we use \textit{unsigned formulas} in our presentation of tableau proof systems \( JLCS \). Corresponding signed tableau system \( JT \) and other signed tableau rules are shown in Tables 5 and 6.

Sequents are expressions of the form \( \Gamma \Rightarrow \Delta \), where \( \Gamma \) and \( \Delta \) are sets of \( JL \)-formulas (thus structural rules of the sequent calculus are not needed).

Smullyan in [17] give a mapping from sets of signed formulas to sequents. Let \( S \) be a set \{\( TA_1, \ldots, TA_n, FB_1, \ldots, FB_m \)\} of signed formulas. Then \(|S|\) denotes the sequent \( A_1, \ldots, A_n \Rightarrow B_1, \ldots, B_m \). Keeping this translation in mind, it is known that tableau proofs are the same as sequent proofs but in the reverse direction. Thus by turning tableau rules of Table 5 upside down, and using Smullyan’s mapping, we get the sequent calculus of Table 7 for basic justification logic \( J \). In this table the formula \( A \) in the initial sequent \((Ax)\) is an arbitrary \( JL \)-formula. There is a condition on the application of rule \((cut)\) (in Table 7) which is similar to that given for rule \((PB)\) in Table 5. The condition on the rule \((L \cdot)\) is stronger than that of tableau rule \((T \cdot)\), but it helps us to show the subformula property.

Other sequent calculi for justification logics are obtained by adding rules from Table 8 to the sequent calculus of \( J \) depending on the axioms of the logic. In Table 8, axiom \((Ax : \bot)\) corresponds to the closure condition given in Remark 3.1.

If \( CS \) is a constant specification for \( JL \), then \( JLCS \) denotes the sequent calculus of \( JLCS \). A \( JL \)-formula \( A \) is provable in \( JLCS \) if the sequent \( \Rightarrow A \) is provable.

\footnote{Some of the rules of Table 8 are already introduced in [2, 12].}
Propositional rules:

| Rule | Premise | Conclusion |
|------|---------|------------|
| $F \neg A \vdash T A$ | ($F \neg$) | $T \neg A \vdash F A$ | ($T \neg$) |
| $F A \rightarrow B \vdash T A$ | ($F \rightarrow$) | $T A \rightarrow B \vdash F A \| T B$ | ($T \rightarrow$) |
| Justification rules: | | |
| $F t + s : A$ | ($F +$) | $T s : (A \rightarrow B)$ | ($T \cdot$) |
| $F t : A$ | ($F$) | $T t : A$ | ($T$) |
| $F s : A$ | ($F$) | $T s \cdot t : B$ | ($T$) |

Principle of Bivalence:

| Rule | Premise | Conclusion |
|------|---------|------------|
| $F A \| F A$ | ($\bot$) | $T A \| F A$ | ($TPB$) |

In ($T \cdot$) the term $s \cdot t$ should occur in the root of the branch.
In ($PB$) the $PB$-formula $A$ is a $J\text{LS}$-subformula of the root of the branch.

| Table 5. Signed tableau rules for basic justification logic $J\text{LS}$. |

**Example 3.3.** We give a sequent proof of $x : A \rightarrow c \cdot x : (B \rightarrow A)$ in $J\text{LS}_C$, where $CS$ contains $c : (A \rightarrow (B \rightarrow A))$.

$$
\begin{array}{c}
(Ax) \\
x : A \Rightarrow c \cdot x : (B \rightarrow A), c : (A \rightarrow (B \rightarrow A)) \\
\end{array} \quad
\begin{array}{c}
\Rightarrow c \cdot x : (B \rightarrow A) \\
\Rightarrow x : A \Rightarrow c \cdot x : (B \rightarrow A) \\
\Rightarrow x : A \rightarrow c \cdot x : (B \rightarrow A)
\end{array}
$$

As usual the interpretation of sequents is as follows:

$$
(\Gamma \Rightarrow \Delta)^I = \bigwedge \Gamma \rightarrow \bigvee \Delta,
$$

where $\bigwedge \Gamma$ and $\bigvee \Delta$ are respectively conjunction of all formulas of $\Gamma$ and disjunction of all formulas of $\Delta$. Now it is easy to show that the interpretation of initial sequents ($Ax$), ($Ax \bot$) are $J\text{LS}$-valid, for all justification logics $J\text{LS}$. And moreover, the interpretation of ($Ax : \bot$) is $J\text{LS}$-valid, in those justification logics that contain axiom $jD$. Furthermore, if the interpretation of the premise(s) of a rule of $J\text{LS}_C$ is $J\text{LS}$-valid, then so is the interpretation of its conclusion.

**Lemma 3.8.** If the sequent $\Gamma \Rightarrow \Delta$ is provable in $J\text{LS}_C$, then $(\Gamma \Rightarrow \Delta)^I$ is $J\text{LS}$-valid.

Now soundness is easily established.

**Theorem 3.10 (Soundness).** If $A$ has a proof in $J\text{LS}_C$, then it is $J\text{LS}$-valid.

**Proof.** Suppose $A$ has a proof in $J\text{LS}_C$. Then the sequent $\Rightarrow A$ is provable in $J\text{LS}_C$. Hence, by Lemma 3.8, its interpretation, that is $A$, is $J\text{LS}$-valid. $\square$
Completeness is shown by making use of the completeness of tableaux (Theorem 3.4) and the Smullyan’s mapping (our method is similar to that given in [10]). First we need a lemma.

**Lemma 3.9.** Let $S$ be a finite set of signed $\mathcal{JL}$-formulas. If there is a closed $\mathcal{JL}_{CS}$-tableau for $S$, then the sequent $\{S\}$ is provable in $\mathcal{JL}_{CS}^0$.

**Proof.** We say that $S$ closes with depth $d$ if $d$ is the smallest number of applications of rules we need to construct a closed tableau for $S$. The proof is by induction on the depth $d$. If $S$ closes with depth 0, then it should contain $T\bot$, or $TA$ and $FA$, for some formula $A$. In both cases the sequent $\{S\}$ is an initial sequent. For the induction step, we only consider one case.

Suppose that $T s : (A \rightarrow B), T t : A \in S$ and in a closed $\mathcal{JL}_{CS}$-tableau for $S$ the first rule application is the rule $(T \cdot)$ on these formulas. By the subformula property (Theorem 3.5), the formula $s \cdot t : B$ is a subformula of a formula in the root $S$ of the tableau. There is a closed $\mathcal{JL}_{CS}$-tableau for $S \cup \{T s \cdot t : B\}$ with smaller depth. By the induction hypothesis, the sequent $\{S \cup \{T s \cdot t : B\}\}$ is provable in $\mathcal{JL}_{CS}^0$. Let $S_L$ and $S_R$ be respectively the set of all formulas of $S$ with sign $T$ and sign $F$. Then $\{S \cup \{T s \cdot t : B\}\}$ is $S_L, s \cdot t : B \Rightarrow S_R$, and thus this sequent is provable in $\mathcal{JL}_{CS}^0$. Since $s \cdot t : B$ is a subformula of a formula in $S$, by an application of rule $(L \cdot)$, the sequent $S_L, s : (A \rightarrow B), t : A \Rightarrow S_R$ is provable in $\mathcal{JL}_{CS}^0$. Therefore, $\{S\}$ is provable in $\mathcal{JL}_{CS}^0$.

□

**Theorem 3.11 (Completeness).** If $A$ is $\mathcal{JL}_{CS}$-valid, then it has a proof in $\mathcal{JL}_{CS}^0$.

**Proof.** Suppose $A$ is $\mathcal{JL}_{CS}$-valid. Then, by Theorem 3.4, is has a $\mathcal{JL}_{CS}$-tableau proof. Thus, there is a closed $\mathcal{JL}_{CS}$-tableau for $\{FA\}$. Hence, by Lemma 3.9, the sequent $\{\{FA\}\}$, or $A$, is provable in $\mathcal{JL}_{CS}^0$. Therefore, $A$ is provable in $\mathcal{JL}_{CS}^0$. □

Since the cut rule (without restriction on the cut-formula) is validity-preserving, by the soundness and completeness theorems, it is admissible in all sequent calculi $\mathcal{JL}_{CS}^0$.

Inspection of all sequent rules of $\mathcal{JL}_{CS}^0$ shows that every formula in the premise(s) of a rule is a $\mathcal{JL}_{CS}$-subformula of a formula in the conclusion. This implies the subformula property.

Table 6. Justification axioms with corresponding signed tableau rules.

| Justification axiom | Signed tableau rule |
|---------------------|---------------------|
| JT: $t : A \rightarrow A$ | $\frac{T t : A}{T A} (T :)$ |
| JD: $t : \bot \rightarrow \bot$ | $\frac{T t : \bot}{T \bot} (T : \bot)$ |
| J4: $t : A \rightarrow \neg t : A$ | $\frac{F \neg t : t : A}{F t : A} \quad (F!)$ |
| JB: $\neg A \rightarrow \neg t : A$ | $\frac{F \neg t : \neg t : A}{T \neg t : A} \quad (F?)$ |
| J5: $\neg t : A \rightarrow \neg \neg t : A$ | $\frac{F \neg t : \neg t : A}{T \neg t : A} \quad (F?)$ |
Theorem 3.12 (Subformula property). Every formula in a sequent proof in \( \mathcal{JL}_{CS} \) is a \( \mathcal{JL}_{CS} \)-subformula of a formula in the endsequent.

Remark 3.2. As an another alternative to formulate our sequent calculi we could define sequents using multisets of formulas instead of sets. In this case, we should add the following rules of contraction to our sequent calculi:

\[
\frac{\Gamma, A, A \Rightarrow \Delta}{\Gamma, A \Rightarrow \Delta} \quad (LC) \quad \frac{\Gamma \Rightarrow A, A, \Delta}{\Gamma \Rightarrow A, \Delta} \quad (RC)
\]

All the results of this section hold for these sequent calculi as well.

4 Conclusion

We introduced tableau proof systems for various justification logics, and proved its soundness and completeness theorems. Using the subformula property, we also show that our tableau systems gives decision procedure for justification logics with finite constant specifications. Furthermore, we presented sequent calculus systems for justification logics, and show also the subformula property.

Acknowledgments

This research was in part supported by a grant from IPM. (No. 93030416)
### Table 8. Justification axioms with corresponding sequent calculus rules.

| Justification axiom | Sequent calculus rule |
|---------------------|-----------------------|
| jT. \( t : A \to A \) | \[ \Gamma, A \Rightarrow \Delta \quad \frac{\Gamma, A \Rightarrow \Delta}{\Gamma, t : A \Rightarrow \Delta} \quad (\text{L} : ) \] |
| jD. \( t : \bot \to \bot \) | (Ax : \( \bot \)) \[ \frac{\Gamma, t : \bot \Rightarrow \Delta}{\Gamma, t : \bot \Rightarrow \Delta} \] |
| j4. \( t : A \Rightarrow !t : t : A \) | \[ \Gamma \Rightarrow t : A, \Delta \quad \frac{\Gamma \Rightarrow t : A, \Delta}{\Gamma \Rightarrow !t : t : A, \Delta} \quad (\text{R}!) \] |
| jB. \( \neg A \Rightarrow ?t : \neg t : A \) | \[ \Gamma, A \Rightarrow \Delta \quad \frac{\Gamma \Rightarrow \neg t : \neg t : A, \Delta}{\Gamma \Rightarrow \Delta} \quad (\text{R}?) \] |
| j5. \( \neg t : A \Rightarrow ?t : \neg t : A \) | \[ \Gamma, t : A \Rightarrow \Delta \quad \frac{\Gamma \Rightarrow \neg t : \neg t : A, \Delta}{\Gamma \Rightarrow \Delta} \quad (\text{R}?) \] |

### References

[1] S. Artemov, Operational modal logic, Technical Report MSI 95–29, Cornell University, 1995.

[2] S. Artemov, Explicit provability and constructive semantics, The Bulletin of Symbolic Logic, 7(1), 1–36, 2001.

[3] S. Artemov, The logic of justification, The Review of Symbolic Logic, 1(4), 477–513, 2008.

[4] S. Artemov and M. Fitting, Justification logic, In Edward N. Zalta, editor, The Stanford Encyclopedia of Philosophy, 2012.

[5] M. D’Agostino, Are tableaux an improvement on truth-tables? Cut-free proofs and bivalence, Journal of Logic, Language and Information, 1, 1–36, 1992.

[6] M. D’Agostino, Tableau methods for classical propositional logic, In Handbook of Tableau Methods, M. D’Agostino, D. Gabbay, R. Haehnle, and J. Posegga, eds, pp. 45–124. Kluwer, Dordrecht/Boston/London, 1999.

[7] M. D’Agostino and M. Mondadori, The taming of the cut. Classical refutations with analytic cut, Journal of Logic and Computation, 4, 285–319, 1994.

[8] M. Finger, Analytic methods for the logic of proofs. Journal of Logic and Computation, 20(1), 167–188, 2010.

[9] M. Fitting, The logic of proofs, semantically, Annals of Pure and Applied Logic, 132(1), 1–25, 2005.

[10] M. Fitting, Modal proof theory, in Handbook of Modal Logic, P. Blackburn, J. van Benthem, and F. Wolter (editors), Chapter 2, pages 85–138, Elsevier, 2007.

[11] M. Fitting, Reasoning with justifications, In David Makinson, Jacek Malinowski, and Heinrich Wansing, editors, Towards Mathematical Philosophy, Papers from the Studia Logica conference Trends in Logic IV, volume 28 of Trends in Logic, chapter 6, pages 107–123. Springer, 2009.

[12] R. Goetschi, On the realization and classification of justification logics. PhD thesis, Universität Bern, 2012.

[13] Roman Kuznets and Thomas Studer, Justifications, ontology, and conservativity, In Thomas Bolander, Torben Braauer, Silvio Ghilardi, and Lawrence Moss, editors, Advances in Modal Logic, Volume 9, pages 437–458. College Publications, 2012.
[14] A. Mkrtychev, Models for the logic of proofs, In S. I. Adian, A. Nerode (Eds.), Logical Foundations of Computer Science, Vol. 1234 of Lecture Notes in Computer Science, Springer, 1997, pages 266–275.

[15] B. Renne, Tableaux for the Logic of Proofs, Technical Report TR–2004001, CUNY Ph.D. Program in Computer Science, March 2004.

[16] B. Renne, Semantic cut-elimination for two explicit modal logics, In Janneke Huitink and Sophia Katrenko, editors, Proceedings of the Eleventh ESSLLI Student Session, 18th European Summer School in Logic, Language and Information (ESSLLI’06), pages 148–158, 2006.

[17] R. M. Smullyan, First-Order Logic, Dover Publications, Inc., New York, 1995.