Terminating Calculi for Propositional Dummett Logic with Subformula Property

Guido Fiorino
Dipartimento di Metodi Quantitativi per le Scienze Economiche ed Aziendali,
Università di Milano-Bicocca, Piazza dell'Ateneo Nuovo, 1, 20126 Milano, Italy.
guido.fiorino@unimib.it

September 29, 2018

Abstract

In this paper we present two terminating tableau calculi for propositional Dummett logic obeying the subformula property. The ideas of our calculi rely on the linearly ordered Kripke semantics of Dummett logic. The first calculus works on two semantical levels: the present and the next possible world. The second calculus employs the usual object language of tableau systems and exploits a property of the construction of the completeness theorem to introduce a check which is an alternative to loop check mechanisms.

1 Introduction

In this paper we present two terminating tableau calculi for propositional Dummett logic obeying the subformula property. The depth of the deductions of the first calculus is quadratic and allow to extract a counter model whose depth is $n+1$ at most, with n the number of propositional variables in the formula to be decided. The depth of the deductions of the second calculus is linear. To avoid the introduction of loop check mechanisms, our calculi exploit the linearly ordered Kripke semantics of Dummett logic. The first calculus uses the ideas presented in paper [11], and works on two semantical levels: the present and the next possible world. The second calculus uses the usual **T** and **F** signs and exploits a property of the construction of the completeness theorem to introduce a check which is an alternative to loop check mechanisms.

Dummett logic has been extensively investigated both by people working in computer science and in logic. The history of this logic starts with Gödel, who studied the family of logics semantically characterised by a sequence of n-valued ($n > 2$) matrices ([12]). In paper [6] Dummett studied the logic semantically characterized by an infinite valued matrix which is included in the family of logics studied by Gödel and proved that such a logic is axiomatizable by adding to any Hilbert system for propositional intuitionistic logic the axiom scheme $(p \to q) \lor (q \to p)$. Moreover, it is well-known that such a logic is semantically characterised by linearly ordered Kripke models. Dummett logic also appear in investigations related to the relevance logics [7] and Heyting provability [16]. Dummett logic has been studied also in recent years for its relationships with computer science ([2]) and fuzzy logics ([13]). For a survey of proof theory for Gödel-Dummett logics we cite [5].

To perform automated deduction both tableau and sequent calculi have been proposed. To get a terminating calculus for Dummett logic obeying to the subformula property, the main problem is how to handle formulas of the kind $\mathbf{T}(A \to B)$ (left-implicative formulas, in the sequent terminology). A terminating calculus can be achieved by introducing specialized rules based on the main connective of A. In the conclusions of such specialized rules some formulas are not subformulas of the premise. Calculi of this nature are provided in [1, 3, 8, 9, 10, 11, 14]. The specialized rules used in [1, 8, 9, 10, 11] are based on the rules proposed by Vorobiev [17] to handle formulas of the kind $\mathbf{T}(A \to B)$ in propositional Intuitionistic logic. Papers [3, 14] decompose implicative formulas by rules whose correctness is justified by the semantics of Dummett logic.

In this work we present calculi whose deductions have, respectively, linear and quadratic depth in the size of the formula to be decided and the subformula property, a feature that the calculi in the above quoted papers fail.

Papers [4] and [15] provide calculi with the subformula property. Work [4] provides a calculus based on sequents called *sequent of relations calculus* whose deductions have exponential depth in the formula to be proved, because in the premise of some rules can occur multiple copies of a subformula of the conclusion. Moreover, the nodes of a proof with such a system are more cumbersome than the nodes of a tableau proof, because every node of the deduction expresses the relation order between the subformulas of the formula to be proved. Thus every node has a quadratic number of formula occurrences. Paper [15] provides two goal-oriented calculi, one based on hypersequents and one on labelled sequents. The systems are restricted to the implicative fragment. The first advantage is that our results are given for the full language. Although a translation from the full language to the implicative fragment is possible but, in the case of disjunction the cost is an exponential blow-up in the size of the formula. Moreover, we do not need the more expressive power of hypersequents and differently from the labelled sequents, the object language of our calculi does not depend on the input.

As regards our results, it is worth to remark that our final calculus is a genuine tableau calculus only employing the usual two signs $\mathbf{T}$ and $\mathbf{F}$ corresponding respectively to the left-hand side and right-hand side of the sequent systems.

As regards the techniques used in the paper, we have deliberately chosen to employ tableaux as proof-systems. Our choice is justified by the fact that the rules of our calculi can be easily explained by semantical considerations based on Kripke models. For this reason we do not use the sequent systems, whose behaviour is upside-down with respect to a semantical characterization. However, since tableau and sequent calculi are related, it is an easy exercise to translate our calculi into sequents. Moreover, correctness and completeness are proved always taking the Kripke models for propositional Dummett logic as semantical reference. Following the proofs of the completeness theorems, the procedures we provide can be modified to return a proof or a counter model.

2 Basic definitions and a terminating tableau calculus with the subformula property

We consider the propositional language based on a denumerable set of propositional variables $\mathcal{PV}$, the boolean constants $\top$ and $\bot$ and the logical connectives $\wedge, \vee, \rightarrow$. We call *atoms* the elements of $\mathcal{PV} \cup \{\top, \bot\}$. In the following, formulas (respectively set of formulas and propositional variables) are denoted by letters A, B, C... (respectively S, T, U,... and p, q, r,...) possibly with subscripts or superscripts.

From the introduction we recall that *Dummett Logic* (**Dum**) can be axiomatized by adding to any axiom system for propositional intuitionistic logic the axiom scheme $(p \rightarrow q) \vee (q \rightarrow p)$ and a well-known semantical characterization of **Dum** is by *linearly ordered Kripke models*. In the paper *model* means a linearly ordered Kripke model, namely a structure $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$, where $\langle P, \leq, \rho \rangle$ is a linearly ordered set with ρ minimum with respect to $\leq$ and $\Vdash$ is the *forcing relation*, a binary relation on $P \times (\mathcal{PV} \cup \{\top, \bot\})$ such that: (i) if $\alpha \Vdash p$ and $\alpha \leq \beta$, then $\beta \Vdash p$; (ii) for every $\alpha \in P$, $\alpha \Vdash \top$ holds and $\alpha \Vdash \bot$ does not hold. Hereafter we denote the members of P by means of lowercase letters of the Greek alphabet.

The forcing relation is extended in a standard way to arbitrary formulas of our language as follows:

1. $\alpha \Vdash A \wedge B$ iff $\alpha \Vdash A$ and $\alpha \Vdash B$;
2. $\alpha \Vdash A \vee B$ iff $\alpha \Vdash A$ or $\alpha \Vdash B$;
3. $\alpha \Vdash A \rightarrow B$ iff, for every $\beta \in P$ such that $\alpha \leq \beta$, $\beta \Vdash A$ implies $\beta \Vdash B$;

We write $\alpha \nVdash A$ when $\alpha \Vdash A$ does not hold. It is easy to prove that for every formula A the *persistence* property holds: If $\alpha \Vdash A$ and $\alpha \leq \beta$, then $\beta \Vdash A$. We say that β is *immediate successor* of α iff $\alpha < \beta$ and there is no $\gamma \in P$ such that $\alpha < \gamma < \beta$. A formula A *is valid in a model* $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ if and only if $\rho \Vdash A$. It is well-known that **Dum** coincides with the set of formulas valid in all models.

In Figures 1 and 2 are given the rules of $\mathbf{D_1}$, a terminating tableau calculus exploiting the truth at present and next possible world in the Kripke semantics.

The calculus $\mathbf{D_1}$ works on signed formulas, that is well-formed formulas prefixed with one of the *signs* $\mathbf{T}$ (with $\mathbf{T}A$ to be read "the fact A is known at the present state of knowledge"), $\mathbf{F}$ (with $\mathbf{F}A$ to be read "the fact A is not known at the present state of knowledge"), $\mathbf{F_l}$ (with $\mathbf{F_l}A$ to be read "this is the last state of knowledge where A is not known"), $\mathbf{F_n}$ (with $\mathbf{F_n}A$ to be read

$$\frac{S, \mathbf{T}(A \wedge B)}{S, \mathbf{T}A, \mathbf{T}B}\,\mathbf{T}\wedge \qquad \frac{S, \mathbf{F_l}(A \wedge B)}{S, \mathbf{F_l}A, \mathbf{T_n}B | S, \mathbf{T}A, \mathbf{F_l}B}\,\mathbf{F_l}\wedge$$

$$\frac{S, \mathbf{T}(A \vee B)}{S, \mathbf{T}A | S, \mathbf{T}B}\,\mathbf{T}\vee \qquad \frac{S, \mathbf{F_l}(A \vee B)}{S, \mathbf{F}A, \mathbf{F_l}B | S, \mathbf{F}B, \mathbf{F_l}A}\,\mathbf{F_l}\vee$$

$$\frac{S, \mathbf{T}(A \to B)}{S, \mathbf{T}B | S, \mathbf{F_l}A, \mathbf{T_n}B | S, \widetilde{\mathbf{T}}(A \to B)}\,\mathbf{T}\to \qquad \frac{S, \mathbf{F_l}(A \to B)}{S, \mathbf{T}A, \mathbf{F_l}B}\,\mathbf{F_l}\to$$

$$\frac{S, \mathbf{F}A}{S, \mathbf{F_l}A | S, \mathbf{F_n}A}\,\mathbf{F}{-decide} \qquad \frac{S, \widehat{\mathbf{T}}(A \to B)}{S, \mathbf{F_l}A, \mathbf{T_n}B | S, \widetilde{\mathbf{T}}(A \to B)}\,\widehat{\mathbf{T}}{-decide}$$

Figure 1: The invertible rules of $\mathbf{D_1}$.

$$\frac{S, \widetilde{\mathbf{T}}(A_1 \to B_1), \ldots, \widetilde{\mathbf{T}}(A_n \to B_n), \mathbf{F_n}C_{n+1}, \ldots, \mathbf{F_n}C_u}{S_c, V_1, S_{\mathbf{F}} | \ldots | S_c, V_n, S_{\mathbf{F}} | S_c, S_{\widehat{\mathbf{T}}}, V_{n+1} | \ldots | S_c, S_{\widehat{\mathbf{T}}}, V_u}\,\mathbf{F_n}\widetilde{\mathbf{T}}$$

where: $S_{\mathbf{F}} = \{\mathbf{F}C_{n+1}, \ldots, \mathbf{F}C_u\}$, $S_{\widehat{\mathbf{T}}} = \{\widehat{\mathbf{T}}(A_1 \to B_1), \ldots, \widehat{\mathbf{T}}(A_n \to B_n)\}$,
for $j = 1, \ldots, n$,
$V_j = \{\widehat{\mathbf{T}}(A_1 \to B_1), \ldots, \widehat{\mathbf{T}}(A_{j-1} \to B_{j-1}), \mathbf{F_l}A_j, \mathbf{T_n}B_j, \widehat{\mathbf{T}}(A_{j+1} \to B_{j+1}), \ldots, \widehat{\mathbf{T}}(A_n \to B_n)\}$,
for $j = n+1, \ldots, u$, $V_j = \{\mathbf{F}C_{n+1}, \ldots, \mathbf{F}C_{j-1}, \mathbf{F_l}C_j, \mathbf{F}C_{j+1}, \ldots, \mathbf{F}C_u\}$ and
$S_c = \{\mathbf{T}A | \mathbf{T}A \in S\} \cup \{\mathbf{T}A | \mathbf{F_l}A \in S\} \cup \{\mathbf{T}A | \mathbf{T_n}A \in S\}$;

Figure 2: The non-invertible rule $\mathbf{F_n}\widetilde{\mathbf{T}}$.

"A is not known in the next state of knowledge"), $\mathbf{T_n}$ (with $\mathbf{T_n}A$ to be read "A will be known in the next state of knowledge"), $\widehat{\mathbf{T}}$ (with $\widehat{\mathbf{T}}A$ to be read as "$\mathbf{T}$A holds and if A is of the kind $B \to C$, then $\mathbf{F}B$ holds") and $\widetilde{\mathbf{T}}$ (with $\widetilde{\mathbf{T}}A$ to be read as "$\mathbf{T}$A holds and if A is of the kind $B \to C$, then $\mathbf{F_n}B$ holds") and on sets of signed formulas (hereafter we omit the word "signed" in front of "formula" in all the contexts where no confusion arises). Formally, the meaning of the signs is provided by the relation *realizability* ($\rhd$) defined as follows: Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a model, let $\alpha \in P$, let H be a signed formula and let S be a set of signed formulas. We say that α *realizes* H (respectively α *realizes* S and $\underline{K}$ *realizes* S), and we write $\alpha \rhd H$ (respectively $\alpha \rhd S$ and $\underline{K} \rhd S$), if the following conditions hold:

1. $\alpha \triangleright \mathbf{T}A$ iff $\alpha \Vdash A$;
2. $\alpha \triangleright \mathbf{F}A$ iff $\alpha \nVdash A$;
3. $\alpha \triangleright \mathbf{F_n}A$ iff there exists $\beta > \alpha$, $\beta \triangleright \mathbf{F}A$;
4. $\alpha \triangleright \mathbf{T_n}A$ iff for every $\beta > \alpha$, $\beta \triangleright \mathbf{T}A$;
5. $\alpha \triangleright \mathbf{F_l}A$ iff $\alpha \triangleright \mathbf{F}A$ and $\alpha \triangleright \mathbf{T_n}A$;
6. $\alpha \triangleright \widetilde{\mathbf{T}}A$ iff $A \equiv B \to C$ and $\alpha \triangleright \mathbf{T}A$ and $\alpha \triangleright \mathbf{F_n}B$;
7. $\alpha \triangleright \widehat{\mathbf{T}}A$ iff $A \equiv B \to C$ and $\alpha \triangleright \mathbf{T}A$ and $\alpha \triangleright \mathbf{F}B$;
8. $\alpha \triangleright S$ iff α realizes every formula in S.

By inspecting the rules of the calculus we have that signs $\widetilde{\mathbf{T}}$ and $\widehat{\mathbf{T}}$ are used for implicative formulas only.

From the meaning of the signs we get the conditions that make a set of formulas inconsistent. A set S is *inconsistent* if one of the following conditions holds:
-$\mathbf{T}\bot \in S$;
-$\{\mathbf{T}A, \mathbf{F}A\} \subseteq S$;
-$\{\mathbf{T}A, \mathbf{F_l}A\} \subseteq S$.

It is easy to prove the following proposition:

Proposizione 1 *If a set of formulas S is inconsistent, then for every Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ and for every $\alpha \in P$, $\alpha \not\triangleright S$.*

A proof table (or proof tree) for S is a tree, rooted in S and obtained by the subsequent instantiation of the rules of the calculus. A *closed proof table* is a proof table whose leaves are all inconsistent sets. A closed proof table is a proof of the calculus and a formula A is provable iff there exists a closed proof table for $\{\mathbf{F_l}A\}$.

The premise of the rules are instantiated in a duplication-free style: in the application of the rules we always consider that the formulas in evidence in the premise are not in S. We say that a rule $\mathcal{R}$ applies to a set U when it is possible to instantiate the premise of $\mathcal{R}$ with the set U and we say that a rule $\mathcal{R}$ applies to a formula $H \in U$ (respectively the set $\{H_1, \ldots, H_n\} \subseteq U$) to mean that it is possible to instantiate the premise of $\mathcal{R}$ taking S as $U \setminus \{H\}$ (respectively $U \setminus \{H_1, \ldots, H_n\}$). As an example, given the set $U = \{\mathbf{T}(B \wedge C), \mathbf{T}(A \wedge C), \mathbf{F}(A \vee B)\}$, by *applying the rule* $\mathbf{T}\wedge$ *taking* $\mathbf{T}(A \wedge C)$ *as main formula* means to instantiate the premise of $\mathbf{T}\wedge$ taking $S = \{\mathbf{T}(B \wedge C), \mathbf{F}(A \vee B)\}$ and $H = \mathbf{T}(A \wedge C)$.

Before going into technical details we give an informal description of the whole machinery. First, note that there are no rules for sign $\mathbf{T_n}$. The sign $\mathbf{T_n}$ aims to mark formulas that will be signed with $\mathbf{T}$ after an application of $\mathbf{F_n}\widetilde{\mathbf{T}}$. Moreover, $\mathbf{F}$-formulas are handled by $\mathbf{F}$-decide. In semantical terms of counter model construction, given the formula $\mathbf{F}A$, rule $\mathbf{F}$-decide decides if in

$$\frac{S, \widetilde{\mathbf{T}}(A_1 \to B_1), \ldots, \widetilde{\mathbf{T}}(A_n \to B_n), \mathbf{F_n} C_{n+1}, \ldots, \mathbf{F_n} C_u}{S_c, V_1, S_{\mathbf{F}} | \ldots | S_c, V_n, S_{\mathbf{F}} | S_c, S_{\widetilde{\mathbf{T}}}, V_{n+1} | \ldots | S_c, S_{\widetilde{\mathbf{T}}}, V_u} \mathbf{F_n}\widetilde{\mathbf{T}}{-opt}$$

where: $S_{\mathbf{F}} = \{\mathbf{F}C_{n+1}, \ldots, \mathbf{F}C_u\}$, $S_{\widetilde{\mathbf{T}}} = \{\widetilde{\mathbf{T}}(A_1 \to B_1), \ldots, \widetilde{\mathbf{T}}(A_n \to B_n)\}$,
for $j = 1, \ldots, n$,
$V_j = \{\widetilde{\mathbf{T}}(A_1 \to B_1), \ldots, \widetilde{\mathbf{T}}(A_{j-1} \to B_{j-1}), \mathbf{F_l} A_j, \mathbf{T_n} B_j, \widehat{\mathbf{T}}(A_{j+1} \to B_{j+1}), \ldots, \widehat{\mathbf{T}}(A_n \to B_n)\}$,
for $j = n+1, \ldots, u$, $V_j = \{\mathbf{F_n} C_{n+1}, \ldots, \mathbf{F_n} C_{j-1}, \mathbf{F_l} C_j, \mathbf{F} C_{j+1}, \ldots, \mathbf{F} C_u\}$ and
$S_c = \{\mathbf{T}A | \mathbf{T}A \in S\} \cup \{\mathbf{T}A | \mathbf{F_l} A \in S\} \cup \{\mathbf{T}A | \mathbf{T_n} A \in S\}$;

Figure 3: The non-invertible rule $\mathbf{F_n}\widetilde{\mathbf{T}}$.

the next state of knowledge the formula A will be a known or an unknown fact. Similarly, rule $\widehat{\mathbf{T}}$-decide decides the semantical status of the antecedent A for formulas of the kind $\widehat{\mathbf{T}}(A \to B)$. By the rules of the calculus, if $\mathbf{T}(A \to B)$ becomes $\widetilde{\mathbf{T}}(A \to B)$, then in the subsequent sets the sign of $A \to B$ can only be $\widehat{\mathbf{T}}$ or $\widetilde{\mathbf{T}}$ and rules $\widehat{\mathbf{T}}$-decide and $\mathbf{F_n}\widetilde{\mathbf{T}}$ are the only rules where the sign can be switched. Rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ is the only non-invertible rule of $\mathbf{D_1}$, thus to devise a complete strategy that does not require backtracking it is sufficient that rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ is applied when no other rule is applicable. Finally, the following features of $\mathbf{D_1}$ allow us to prove the termination: every node of the proof table contains at least a $\mathbf{F_l}$-formula and an application of $\mathbf{F_n}\widetilde{\mathbf{T}}$ increases the number of $\mathbf{T}$-signed formulas. Also note that, if $\mathbf{F_n}\widetilde{\mathbf{T}}$ is applied only if no other rule is applicable, which is the way we want to use $\mathbf{F_n}\widetilde{\mathbf{T}}$, then the premise is always instantiated to a set containing at least an $\mathbf{F_l}$-atomic formula. This implies that in the conclusion at least one new $\mathbf{T}$-signed atomic formula is introduced. Summarizing, in spite of the rightmost set in the conclusion of rules $\mathbf{T} \to$, $\mathbf{F} - decide$ and $\widehat{\mathbf{T}} - decide$, any sequence of application of rules ends in a set containing signed atomic formulas only and we do not have infinite loops.

Remark 1 *The presentation of the calculus is without efficiency in mind. We could exploit the meaning of signs* $\mathbf{F_l}$ *and* $\mathbf{T_n}$ *to introduce more rules and checks that allow us to reduce the size of the proofs. As an example we could extend the notion of inconsistent set by adding to those given above the following conditions:* $\{\mathbf{F}A, \widehat{\mathbf{T}}A\} \subseteq S$; $\{\mathbf{F}A, \widetilde{\mathbf{T}}A\} \subseteq S$; $\{\mathbf{T}A, \mathbf{F_n}A\} \subseteq S$; $\{\mathbf{T}A, \widetilde{\mathbf{T}}(A \to B)\} \subseteq S$; $\{\mathbf{T}A, \widehat{\mathbf{T}}(A \to B)\} \subseteq S$; $\{\mathbf{T_n}A, \widetilde{\mathbf{T}}(A \to B)\} \subseteq S$; $\{\mathbf{T_n}\bot, \mathbf{F_n}A\} \subseteq S$; $\{\mathbf{T_n}A, \mathbf{F_n}A\} \subseteq S$; $\{\mathbf{F_l}A, \widetilde{\mathbf{T}}(A \to B)\} \subseteq S$; $\{\mathbf{F_l}\bot, \mathbf{F_n}A\} \subseteq S$. *This would avoid to perform useless deduction steps all ending in inconsistent sets. The rule* $\mathbf{F_n}\widetilde{\mathbf{T}} - opt$ *given in Figure 3 is an optimization of rule* $\mathbf{F_n}\widetilde{\mathbf{T}}$ *of Figure 2. This rule avoids useless applications of* $\mathbf{F}$*-decide and* $\widehat{\mathbf{T}}$*-decide which are rules that introduce branching points.*

3 Correctness

To obtain the correctness of $\mathbf{D_1}$ with respect to Dummett logic, we proceed by showing that the existence of a proof table for $\{\mathbf{F_l}A\}$, implies the validity of A in Dummett logic. The main step consists in establishing that the rules of the calculus preserve realizability:

Proposizione 2 *For every rule of* $\mathbf{D_1}$*, if a world* α *of a model* $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ *realizes the premise, then* α *realizes at least one of the conclusions.*

Proof 1 *We consider only two rules:*
Rule $\mathbf{F_l}\wedge$*. Let us suppose that* $\alpha \rhd \mathbf{F_l}(A \wedge B)$*. By the meaning of* $\mathbf{F_l}$ *we have that* $\alpha \nVdash A \wedge B$ *and for every world* $\beta \in P$*, if* $\alpha < \beta$*, then* $\beta \Vdash A \wedge B$*. This implies that* $\alpha \nVdash A$ *or* $\alpha \nVdash B$ *and* $\beta \Vdash A$ *and* $\beta \Vdash B$*. We have two main cases on* A*: if* $\alpha \nVdash A$ *holds, then, since* $\beta \Vdash A$*, we get* $\alpha \rhd \mathbf{F_l}A$*. Moreover, from* $\beta \Vdash B$*,* $\alpha \rhd \mathbf{T_n}B$ *follows; if* $\alpha \Vdash A$ *holds, then* $\alpha \rhd \mathbf{T}A$ *holds and, by* $\alpha \nVdash A \wedge B$*, we have* $\alpha \nVdash B$ *and since* $\beta \Vdash B$ *it follows that* $\alpha \rhd \mathbf{F_l}B$ *holds;*
Rule $\mathbf{F_n}\widetilde{\mathbf{T}}$*. The correctness of the rule can be explained following [1]. Let us suppose that* $\alpha \rhd S, \widetilde{\mathbf{T}}(A_1 \to B_1), \ldots, \widetilde{\mathbf{T}}(A_n \to B_n), \mathbf{F_n}A_{n+1}, \ldots, \mathbf{F_n}A_u$*. By the meaning of* $\widetilde{\mathbf{T}}$ *we have that* $\alpha \rhd \mathbf{F_n}A_1, \mathbf{T}(A_1 \to B_1), \ldots, \mathbf{F_n}A_n, \mathbf{T}(A_n \to B_n)$*. Thus there exists* β_i *such that* $\alpha < \beta_i$ *and* $\beta_i \rhd \mathbf{F_l}A_i$*, for* $i = 1, \ldots, u$*. We notice that* β_i *realizes all the* $\mathbf{T}$ *formulas in* S *and* $\beta_i \rhd \mathbf{T}C$ *if* $\mathbf{F_l}C \in S$*. Moreover, if* $\beta_i = \min\{\beta_1, \ldots, \beta_u\}$*, then* $\beta_i \rhd \mathbf{F}A_1, \ldots, \mathbf{F}A_{i-1}, \mathbf{F_l}A_i, \mathbf{F}A_{i+1}, \ldots, \mathbf{F}A_u$*. By the meaning of* $\mathbf{T}, \widehat{\mathbf{T}}$ *and* $\mathbf{T_n}$ *we conclude that if* $i \in \{1, \ldots, n\}$*, then*

$$\beta_i \rhd \{\widehat{\mathbf{T}}(A_1 \to B_1), \ldots, \widehat{\mathbf{T}}(A_n \to B_n), \mathbf{F}A_{n+1}, \ldots \mathbf{F}A_u\} \cup \{\mathbf{F_l}A_i, \mathbf{T_n}B_i\},$$

otherwise $\beta_i \rhd \{\widehat{\mathbf{T}}(A_1 \to B_1), \ldots, \widehat{\mathbf{T}}(A_n \to B_n), \mathbf{F}A_{n+1}, \ldots, \mathbf{F}A_u\} \cup \{\mathbf{F_l}A_i\}$.

From the proposition above we get

Theorem 1 *If there exists a closed proof table for* A*, then* A *is valid in Dummett logic.*

4 Completeness

We describe a procedure using the rules of the calculus to return a proof or a counter model for a given set of signed formulas S.

In the following we sketch the recursive procedure D(S). Given a set S of formulas, D(S) returns either a closed proof table for S or `NULL` (if there exists a model realizing S). To describe D we use the following definitions and notations. We call α-rules and β-rules the rules of Figure 1 with one conclusion and with two conclusions, respectively. The α-formulas and β-formulas are the kind of the signed formulas in evidence in the premise of the α-rules and β-rules, respectively (e.g. $\mathbf{T}(A \wedge B)$ is an α-formula and $\mathbf{T}(A \vee B)$ is a β-formula). Let S be a set of formulas, let $H \in S$ be an α or β-formula. With $Rule(H)$

we denote the rule corresponding to H in Figure 1. Let S_1 or $S_1|S_2$ be the nodes of the proof tree obtained by applying to S the rule $Rule(H)$. If Tab_1 and Tab_2 are closed proof tables for S_1 and S_2 respectively, then $\frac{S}{Tab_1}Rule(H)$ or $\frac{S}{Tab_1|Tab_2}Rule(H)$ denote the closed proof table for S defined in the obvious way. Moreover, $\mathcal{R}_i(H)$ $(i = 1, 2)$ denotes the set containing the formulas of S_i which replaces H. For instance:
$\mathcal{R}_1(\mathbf{T}(A \wedge B)) = \{\, \mathbf{T}A, \mathbf{T}B \,\}$,
$\mathcal{R}_1(\mathbf{T}(A \vee B)) = \{\mathbf{T}A\}$, $\mathcal{R}_2(\mathbf{T}(A \vee B)) = \{\mathbf{T}B\}$,
In the case of $\mathbf{F_n}\widetilde{\mathbf{T}}$ we generalize the above notation. Let $S_{\mathbf{F_n}\widetilde{\mathbf{T}}}$ be the set of all the $\mathbf{F_n}$-formulas of S. Let $S_1|\ldots|S_n$ be the nodes of the proof tree obtained by applying to S the rule $\mathbf{F_n}\widetilde{\mathbf{T}}$. If $Tab_1 \ldots, Tab_n$ are closed proof tables for $S_1, \ldots, S_n$, respectively, then $\frac{S}{Tab_1|\ldots|Tab_n}\mathbf{F_n}\widetilde{\mathbf{T}}$ is the closed proof table for S. With $\mathcal{R}_i(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$ we denote the set of formulas that replaces the set $S_{\mathbf{F_n}\widetilde{\mathbf{T}}}$ in the i-th conclusion of $\mathbf{F_n}\widetilde{\mathbf{T}}$. For example, given $S_{\mathbf{F_n}\widetilde{\mathbf{T}}} = \{\widetilde{\mathbf{T}}(A_1 \to B_1), \mathbf{F_n}A_2, \mathbf{F_n}A_3\}$, $\mathcal{R}_2(S_{\mathbf{F_n}\widetilde{\mathbf{T}}}) = \{\widehat{\mathbf{T}}(A_1 \to B_1), \mathbf{F_l}A_2, \mathbf{F}A_3\}$.

Function D(S)

1. If S is an inconsistent set, then D returns the proof S;

2. If an α-rule applies to S, then let H be a α-formula of S. If $\mathrm{D}((S \setminus \{H\}) \cup \mathcal{R}_1(H))$ returns a proof π, then D returns the proof $\frac{S}{\pi}Rule(H)$, otherwise D returns `NULL`;

3. If a β-rule applies to S, then let H be a β-formula of S. Let $\pi_1 = \mathrm{D}((S \setminus \{H\}) \cup \mathcal{R}_1(H))$ and $\pi_2 = \mathrm{D}((S \setminus \{H\}) \cup \mathcal{R}_2(H))$. If π_1 or π_2 is `NULL`, then D returns `NULL`, otherwise D returns $\frac{S}{\pi_1|\pi_2}Rule(H)$;

4. If the rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ applies to S, then let $S_{\mathbf{F_n}\widetilde{\mathbf{T}}} = \{\mathcal{S}A \in S | \mathcal{S} \in \{\widetilde{\mathbf{T}}, \mathbf{F_n}\}\}$ and $n = |S_{\mathbf{F_n}\widetilde{\mathbf{T}}}|$. If there exists $i \in \{1, \ldots, n\}$, such that $\pi_i = \mathrm{D}((S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c \cup \mathcal{R}_i(S_{\mathbf{F_n}\widetilde{\mathbf{T}}}))$ is `NULL`, then D returns `NULL`. Otherwise $\pi_1, \ldots, \pi_n$ are proofs and D returns $\frac{S}{\pi_1|\ldots|\pi_n}\mathbf{F_n}\widetilde{\mathbf{T}}$;

5. If none of the previous points apply, then D returns `NULL`.

end function D.

We emphasize that function D respects a particular sequence in the application of the rules: $\mathbf{F_n}\widetilde{\mathbf{T}}$ is applied if no other rule is applicable. As a result no backtracking step is necessary. Moreover, to decide A, the function call $\mathrm{D}(\{\mathbf{F_l}A\})$ is performed. By the rules handling $\mathbf{F_l}$-formulas we have that when rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ is applied the formal parameter S contains at least a $\mathbf{F_l}$-atomic formula and by rule $\mathbf{F_n}\widetilde{\mathbf{T}}$, every actual parameter of the recursive call performed in Step 4, contains a $\mathbf{F_l}$-formula and a $\mathbf{T}$-atomic formula not occurring in S. This implies that every application of $\mathbf{F_n}\widetilde{\mathbf{T}}$ introduces a new $\mathbf{T}$-atomic formula and thus we can have at most n applications of rule $\mathbf{F_n}\widetilde{\mathbf{T}}$, where n is the number of propositional variables in A. Since between two applications of rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ we cannot have an infinite sequence of rule applications, we conclude that the function call $\mathrm{D}(\{\mathbf{F_l}A\})$ always terminates. More formally, we can define a binary relation $\prec$ on sets of formulas defined as follows: $S' \prec S$ iff (i) the set of $\mathbf{T}$-atomic formulas in S' includes the set of atomic formulas in S, or (ii) the set of $\mathbf{T}$-atomic formulas in S' coincides with the set of atomic formulas in S and the number of connectives in S' is lower than in S or (*iii*) the sets S and S' contain the same $\mathbf{T}$-atomic formulas, the same number of connectives and they differ for the sign of a single formula A such that $\mathbf{T}A \in S$ and $\widetilde{\mathbf{T}}A \in S'$ or $\widetilde{\mathbf{T}}A \in S$ and $\widehat{\mathbf{T}}A \in S'$ or $\mathbf{F}A \in S$ and $\mathbf{F_l}A \in S'$ or $\mathbf{F}A \in S$ and $\mathbf{F_n}A \in S'$. By inspecting the rules of the calculus it follows that every recursive call is performed on a actual parameter S' such that $S' \prec S$. By definition of $\prec$ and the fact that the sets only contain subformulas of the formula to be decided, every chain of recursive calls on non-inconsistent sets ends in a set only containing signed atomic formulas. This implies that function D terminates.

In order to get the completeness of D, in the following it is proved that given a set of formulas S, if the call of $\mathrm{D}(S)$ returns `NULL`, then there is enough information to build a model $\underline{K} = \langle P,\leq,\rho,\Vdash\rangle$ such that $\rho \rhd S$.

Theorem 2 (Completeness of D) *Let A be a formula. If A is valid in propositional Dummett logic, then $\mathrm{D}(\{\mathbf{F_l}A\})$ returns a proof.*

Proof 2 *To prove the theorem, we consider a set S of formulas and we prove that if $\mathrm{D}(S)$ returns `NULL`, then there exists a Kripke model $\underline{K} = \langle P,\leq,\rho,\Vdash\rangle$ such that $\rho \rhd S$. We get the statement of the theorem by setting $S = \{\mathbf{F_l}A\}$ and using the contrapositive.*

We proceed by induction on the number of nested recursive calls. It is worth to note that the construction of $\underline{K}$ uses the sets of formulas involved in Step 4 or 5 of function D as elements of P.
Basis: *There are no recursive calls. Then Step 5 has been performed. We notice that S is not inconsistent (otherwise Step 1 would have been performed). Indeed, S only contains atomic formulas signed with $\mathbf{T}$, $\mathbf{T_n}$, $\mathbf{F_l}$. It is easy to prove that the model $\underline{K} = \langle P,\leq,\rho,\Vdash\rangle$, where $\rho = S$, $P = \{\rho\}$, $\rho \leq \rho$ and $\rho \Vdash p$ iff $\mathbf{T}p \in S$, realizes S.*

Step: *By induction hypothesis we assume that the proposition holds for all sets S' such that $\mathrm{D}(S')$ requires less than n recursive calls. We prove the proposition holds for a set S such that $\mathrm{D}(S)$ requires n recursive calls by inspecting all the possible cases where the procedure returns the `NULL` value.*
`NULL` value returned performing Step 4. *By induction hypothesis there exists a model $\underline{K}' = \langle P', \leq', \rho', \Vdash'\rangle$ such that $\rho' \rhd (S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c \cup \mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$. We have two cases: if the j-th formula in the enumeration of $S_{\mathbf{F_n}\widetilde{\mathbf{T}}}$ is $\widetilde{\mathbf{T}}(A_j \to B_j)$, then $\mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}}) = (\{\widehat{\mathbf{T}}(A \to B)|\widetilde{\mathbf{T}}(A \to B) \in S_{\mathbf{F_n}\widetilde{\mathbf{T}}}\}\setminus\{\widehat{\mathbf{T}}(A_j \to B_j)\}) \cup \{\mathbf{F_l}A_j, \mathbf{T_n}B_j\} \cup \{\mathbf{F}C|\mathbf{F_n}C \in S_{\mathbf{F_n}\widetilde{\mathbf{T}}}\}$. By $\rho' \rhd \mathbf{F_l}A_j, \mathbf{T_n}B$, we have $\rho \rhd \widehat{\mathbf{T}}(A_j \to B_j)$ and $\rho \nVdash A_j$. We also have that for every $\widehat{\mathbf{T}}(A \to B) \in \mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$, $\rho' \nVdash' A$ and for every $\mathbf{F}C \in \mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$, $\rho' \nVdash C$. We build the following structure*

$\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ *such that*

$$\begin{array}{rcl} P & = & P' \cup \{\rho\}, \\ \leq & = & \leq' \cup \{(\rho, \alpha) | \alpha \in P'\}, \\ \Vdash & = & \Vdash' \cup \{(\rho, p) | \mathbf{T}p \in S\}, \end{array}$$

where we set $\rho = S$*. Since* $\underline{K}'$ *is a Dummett model realizing* $(S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c$*, it follows that* $\underline{K}$ *is a Dummett model. As a matter of fact,* ρ' *is the immediate successor of* ρ *and* $\mathbf{T}A \in S$ *implies* $\mathbf{T}A \in (S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c$*, thus the forcing relation is preserved. This also implies that:* $\rho \nVdash A_j$ *holds, that, together with the facts* $\rho' \Vdash' A_j \to B_j$ *and* $\rho' \nVdash' A_j$*, implies* $\rho \rhd \widetilde{\mathbf{T}}(A_j \to B_j)$*; for every* $\widehat{\mathbf{T}}(A \to B) \in \mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$*,* $\rho \nVdash A$ *holds, that together with the facts* $\rho' \Vdash' A \to B$ *and* $\rho' \nVdash' A$*, implies that* $\rho \rhd \widetilde{\mathbf{T}}(A \to B)$*; for every* $\mathbf{F}C \in \mathcal{R}_j(S_{\mathbf{F_n}\widetilde{\mathbf{T}}})$*,* $\rho \nVdash C$ *holds, that together with the fact* $\rho' \nVdash C$ *implies* $\rho \rhd \mathbf{F_n}C$*. Thus we have proved that* $\rho \rhd S_{\mathbf{F_n}\widetilde{\mathbf{T}}}$*. As regard the other formulas in S: if* $\mathbf{F_l}A \in S$*, then* A *is an atomic formula and* $\mathbf{T}A \in (S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c$*. Since* S *is not inconsistent (otherwise Step 1 would have been performed)* $\mathbf{T}A \notin S$ *holds, this implies* $\rho \nVdash A$*; if* $\mathbf{T_n}A \in S$ *holds, then* $\mathbf{T}A \in (S \setminus S_{\mathbf{F_n}\widetilde{\mathbf{T}}})_c$ *and hence* $\rho' \Vdash A$*. Summarizing we conclude that* $\rho \rhd S$*.*

We remark that following the construction of Theorem 2, it is straightforward how to modify function D to get a function returning a proof or a counter model. In particular, the proof puts in evidence that a counter model can be extracted by any branch of a tableau proof ending in a non-contradictory set to which no further rule is applicable. By the fact that every application of $\mathbf{F_n}\widetilde{\mathbf{T}}$ introduces in the conclusion a new propositional variable, it follows that if a formula A is realizable, then D returns a counter model for A having $n+1$ elements at most, where n is the number of propositional variables of A. Finally, note that the elements of the counter model $\underline{K}$ are sets of formulas only with the aim to simply the discussion in next section.

5 Handling F-formulas

Now we start to discuss a calculus handling formulas signed with $\mathbf{T}$ and $\mathbf{F}$ only. We present our ideas in two steps. First we introduce calculus $\mathbf{D_2}$ having rules to handle the main connective of $\mathbf{F}$-formulas, this allow us to get rid of $\widehat{\mathbf{T}}$-decide rule. Then we go a step further to get our final calculus $\mathbf{D_3}$.

To handle $\mathbf{F}$-formulas by rules based on the main connective, it is necessary to introduce a machinery to determine, given $\widetilde{\mathbf{T}}(A \to B)$, if A is forced. Such a machinery is based on a notion similar to the boolean satisfiability of a formula in a model. Let S be a non-inconsistent set of signed formulas and let A be a formula, we write $S \models A$ iff $\mathbf{T}A \in S$, $\widetilde{\mathbf{T}}A \in S$ or one of the following conditions holds: (i) $A = \top$; (ii) $A = B \wedge C$, $S \models B$ and $S \models C$; (iii) $A = B \vee C$, $S \models B$ or $S \models C$; (iv) $A = B \to C$ and $\mathbf{F_n}A \notin S$ and if $S \models B$ then $S \models C$.

We are interested to check if $S \models A$ holds when $\mathbf{F_n}\widetilde{\mathbf{T}}$ is the only rule applicable to S. The relation $\models$ aims to express via syntax the semantical notion of realizability. In other words, we are looking for a syntactical checking for forcing and non-forcing of a formula in a world of the Kripke model built in the proof of Theorem 2. Relation $\models$ allows us to express such a checking via the way the formulas are handled in the construction of the counter model. The construction has the properties suggesting that a new calculus managing $\mathbf{T}$ and $\mathbf{F}$-formulas and a syntactical checking based on $\models$ can be given. We start to show a relation between $\Vdash$ and $\models$ in the construction given in Theorem 2:

Lemma 1 *Let S be a set occurring in the construction of the model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ in the proof of Theorem 2 and let $\alpha \in P$. Let us suppose that $\alpha \rhd S$. Then:*
(i) if $\mathcal{S}A \in S$, with $\mathcal{S}A \in \{\mathbf{T}, \widetilde{\mathbf{T}}, \widehat{\mathbf{T}}\}$, then for every $\beta \in P$ such that $\alpha \leq \beta$, $\beta \models A$ and
(ii) if $\mathcal{S}A \in S$, with $\mathcal{S}A \in \{\mathbf{F}, \mathbf{F_l}, \mathbf{F_n}\}$, then $\alpha \not\models A$.

Proof 3 *Note that (i) states that $\models$ is persistent. The proof proceeds by induction on A.*

Basis: A is an atomic formula. We have the cases $\mathbf{T}A$, $\mathbf{F_l}A$, $\mathbf{F_n}A$ and $\mathbf{F}A$.
Case $\mathbf{T}A$. By construction of $\underline{K}$, for every $\beta \in P$ such that $\alpha \leq \beta$, $\mathbf{T}A \in \beta$ holds and we get $\beta \models A$ by definition of $\models$.
Case $\mathbf{F_l}A$. By construction $\mathbf{F_l}A \in \alpha$ and $\mathbf{T}A \notin \alpha$. By definition of $\models$ we get $\alpha \not\models A$.
Case $\mathbf{F_n}A$. By construction $\mathbf{F_n}A \in \alpha$. Since by construction there exists a subsequent set S' of α such that $\mathbf{F_l}A \in S'$, it follows that $\mathbf{T}A \notin S$, thus, by definition of $\models$ we get $\alpha \not\models A$.
Case $\mathbf{F}A$. By construction $\mathbf{F_l}A \in \alpha$ or $\mathbf{F_n}A \in \alpha$ and we immediately get that $\alpha \not\models A$.

Step: we proceed according to the outer connective of A.
Case $\mathbf{T}A = \mathbf{T}(B \to C)$. By construction of proof in Theorem 2 we have three cases: (i) there is a subsequent set S' of S such that $\mathbf{T}C \in S'$ and $\alpha \rhd S'$. For every $\beta \in P$ such that $\alpha \leq P$, $\beta \rhd \mathbf{T}C$ and by induction hypothesis we conclude $\beta \models C$; (ii) there exists a subsequent set S' such that $\mathbf{F_l}B, \mathbf{T_n}C \in S$. Thus $\alpha \rhd \mathbf{F_l}B, \mathbf{T_n}C$ and by induction hypothesis applied to $\mathbf{F_l}B$ we get $\alpha \not\models B$. Moreover by the construction we have that there exists a set S'' such that $\mathbf{T}C \in S''$ and for every $\beta \in P$ such that $\alpha < \beta$, $\beta \rhd \mathbf{T}C$. By induction hypothesis on C we get that $\beta \models C$ that together $\alpha \not\models B$ proves that $\alpha \models B \to C$; (iii) by construction there exist a subsequent set S' of S and α such that $\mathbf{F_l}B, \mathbf{T_n}C \in S'$ and $\beta \in P$ such that $\alpha < \beta$ and $\beta \rhd S'$. By proceeding as in Point (ii) we get that for every $\gamma \in P$ such that $\beta \leq \gamma$, $\gamma \models B \to C$. Moreover for every $\gamma \in P$ such that $\alpha \leq \gamma$ and $\gamma < \beta$, $\widetilde{\mathbf{T}}A \in \gamma$. By definition of $\models$ we immediately get that $\gamma \models \widetilde{\mathbf{T}}(B \to C)$. Thus we have proved that for every $\beta \in P$ such that $\alpha \leq \beta$, $\beta \models B \to C$.

Next Proposition 3 is the main step to introduce our new calculus. We express the relationship between $\Vdash$ and $\models$ in the construction of the counter model given in Theorem 2:

Proposizione 3 *Let S be a set and let us suppose that $\mathbf{F_l}A \in S$, the call $\mathrm{D}(S)$ returns* `NULL` *and in the counter model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ built in Theorem 2 there exists an element of P forcing A. Let $\alpha \in P$ be the minimum world such that $\alpha \Vdash A$. We have that $\alpha \models A$ and for every $\beta \in P$ such that $\beta < \alpha$, $\beta \not\models A$.*

Proof 4 *By the construction given in Theorem 2, the hypothesis $\mathbf{F_l}A \in S$ implies $\rho \nVdash A$. Moreover, by the meaning of the sign $\mathbf{F_l}$ we have that α is the immediate successor of ρ. What we are going to prove is that if in the construction of the counter model $\underline{K}$ the formula $\mathbf{F_l}A$ is occurred and there exists $\alpha \in P$ such that $\alpha \Vdash A$, then the syntactical information in α allows us to prove $\alpha \Vdash A$ via $\alpha \models A$ also when $\mathbf{T}A \notin \alpha$. We proceed by induction on A.*
Basis: A is a propositional variable. We have that $\alpha \Vdash A$ iff (by definition of $\Vdash$) $\mathbf{T}A \in \alpha$ iff $\alpha \models A$ (by definition of $\models$). Moreover, for every $\beta < \alpha$, since $\beta \nVdash A$ we have that $\mathbf{T}A \notin \beta$, thus $\beta \not\models A$.
Step:
Case $A = B \to C$. In the stack of the recursive calls of $\mathrm{D}(S)$ there exists a subsequent set S' of S such that $\mathbf{T}B \in S'$ and $\mathbf{F_l}C \in S'$. By the completeness theorem we have $\rho \rhd S'$, thus $\rho \Vdash B$ and $\rho \nVdash C$. Since $\alpha \Vdash B \to C$, from $\rho \Vdash B$ it follows that $\alpha \Vdash C$. By induction hypothesis on C, $\alpha \models C$. Thus we conclude that $\alpha \models B \to C$. Since $\mathbf{T}B \in S'$, $\mathbf{F_l}C \in S'$ and $\rho \rhd S'$, by Lemma 1 we get that $\rho \not\models B \to C$ holds.

Case $A = B \wedge C$. We have three cases: (i) in the stack of the recursive calls there exists a subsequent set S' of S such that $\mathbf{F_l}B, \mathbf{F_l}C \in S'$. By Theorem 2 $\rho \rhd \mathbf{F_l}B, \mathbf{F_l}C$, thus $\rho \nVdash B$ and $\rho \nVdash C$. Since $\alpha \Vdash A \wedge B$ we have $\alpha \Vdash A$ and $\alpha \Vdash B$. By induction hypothesis applied to B and C we get $\alpha \models B$ and $\alpha \models C$, thus $\alpha \models B \wedge C$, $\rho \not\models B$ and $\rho \not\models C$, thus $\rho \not\models B \wedge C$; (ii) $\mathbf{F_l}B, \mathbf{T}C \in S'$. By Theorem 2, $\rho \rhd \mathbf{F_l}B, \mathbf{T}C$. Since $\alpha \Vdash B \wedge C$, we have $\alpha \Vdash B$ and $\alpha \Vdash C$. By induction hypothesis and Lemma 1, $\alpha \models B \wedge C$. Moreover, since $\rho \rhd S'$ and $\mathbf{F_l}B \in S'$ we get $\rho \not\models B$. (iii) $\mathbf{T}B, \mathbf{F_l}C \in S$. The case goes as (ii).
Case $A = B \vee C$. We have two cases: (i) $\mathbf{F}C, \mathbf{F_l}B \in S'$. By completeness theorem $\rho \rhd S'$. By hypothesis, $\alpha \rhd B$, thus by induction hypothesis applied to B, $\alpha \models B$ and $\rho \not\models B$. This implies $\alpha \models B \vee C$. Now, since $\mathbf{F}C \in S'$ we have that there exists a set S'' subsequent to S' such that $\rho \rhd S''$ and $\mathbf{F_l}C \in S''$ or $\mathbf{F_n}C \in S''$. In both cases we get $\rho \not\models C$ and thus $\rho \not\models B \vee C$. (ii) $\mathbf{F}B, \mathbf{F_l}C \in S'$. The case goes as (i).

Note that in the proof above, we take advantage from the fact that the world α is the immediate successor of ρ and, as in Case $A = B \vee C$, we appeal to the fact $\rho \rhd S'$. The difficult part will come when, by construction, we cannot say that the world α is the immediate successor of ρ. We will face this problem with our final calculus $\mathbf{D_3}$. The strategy employed by function D implies that a $\mathbf{F}$-formula sooner or later become a $\mathbf{F_l}$-formula. We can use the result above to get calculus $\mathbf{D_2}$, which represents a first slight change to calculus $\mathbf{D_1}$:

$$\frac{S, \mathbf{T}(A \to B)}{S, \mathbf{T}A | S, \mathbf{F}A, \overline{\mathbf{T}}(A \to B)}\mathbf{T}{\to_1} \qquad \frac{S, \overline{\mathbf{T}}(A \to B)}{S, \mathbf{T}B}\overline{\mathbf{T}}\text{, provided } S \models A$$

Figure 4: Rules for $\mathbf{D_2}$

- We leave out the signs $\widetilde{\mathbf{T}}$ and $\widehat{\mathbf{T}}$ and the rule $\widehat{\mathbf{T}}$-decide.
- the new calculus $\mathbf{D_2}$ has the rules $\mathbf{T}\wedge$, $\mathbf{T}\vee$, $\mathbf{F_l}\wedge$, $\mathbf{F_l}\vee$, $\mathbf{F_l} \to$ and $\mathbf{F}$-decide of $\mathbf{D_1}$. Rule $\mathbf{F_n}\widetilde{\mathbf{T}}$ now becomes a rule handling $\mathbf{F_n}$-formulas only, thus we refer to it with the name of $\mathbf{F_n}$. Finally, $\mathbf{D_2}$ has the rules in Figure 4;
- relation $\models$ needs to be redefined according to the syntax of the new calculus: Let S be a set of signed formulas and A a formula, we write $S \models A$ iff $\mathbf{T}A \in S$, $\overline{\mathbf{T}}A \in S$ or one of the following conditions hold: (i) $A = \top$; (ii) $A = B \wedge C$ and $S \models B$ and $S \models C$; (iii) $A = B \vee C$ and $S \models B$ or $S \models C$; (iv) $A = B \to C$, $\mathbf{F_n}A \notin S$ and if $S \models B$ then $S \models C$.
- the sign $\overline{\mathbf{T}}$ is introduced to mark forced formulas of the kind $A \to B$ that are not at disposal of the rule $\mathbf{T} \to$ because already handled previously in the branch. By the propositions given above, if $S \not\models A$ holds, then meaning of $\overline{\mathbf{T}}(A \to B)$ is exactly the same of $\widetilde{\mathbf{T}}(A \to B)$.

By using previous results it is not difficult design a decision procedure based on $\mathbf{D_2}$ and to prove correctness and completeness. In such a procedure rule $\overline{\mathbf{T}}$ is possibly applied if no other rule but $\mathbf{F_n}$ is applicable.

Now we can do another step and get rid of sign $\mathbf{F_l}$ and rule $\mathbf{F}$-decide. The propositions given above use the fact that the information about an $\mathbf{F}$-formula is not syntactically lost. As a matter of fact, every $\mathbf{F}$-formula is handled by $\mathbf{F}$-decide and sooner or later a $\mathbf{F}$-formula is turned into a $\mathbf{F_l}$-formula and in the meantime the $\mathbf{F}$-formula has become a $\mathbf{F_n}$-formula.

The rules of this new calculus $\mathbf{D_3}$ are given in Figure 5. The calculus works on the signs $\mathbf{T}$ and $\mathbf{F}$. The sign $\overline{\mathbf{T}}$ labels formulas that are not at disposal of deduction, thus it is not part of the object language. The signs $\mathbf{F_l}$, $\mathbf{T_n}$ and $\mathbf{F_l}$ are no longer necessary to get a calculus obeying the subformula property. A set S is inconsistent iff $\mathbf{T}\bot \in S$ or $\{\mathbf{T}A, \mathbf{F}A\} \subseteq S$. Note rule $\mathbf{F}\wedge$ where both A and B occur. This is necessary to get for $\mathbf{D_3}$ the analogous of Proposition 3. For this calculus relation $\models$ is defined as follows: $S \models A$ iff $\mathbf{T}A \in S$, $\overline{\mathbf{T}}A \in S$ or one of the following conditions hold: (i) $A = B \wedge C$ and $S \models B$ and $S \models C$; (ii) $A = B \vee C$ and $S \models B$ or $S \models C$; (iii) $A = B \to C$, $\mathbf{F}A \notin S$ and if $S \models B$ then $S \models C$. As for the rules of the calculus, $\overline{\mathbf{T}}$ is the only rule requiring a proof of correctness. Moreover, for every rule of $\mathbf{D_3}$ but $\mathbf{F} \to$, it is immediate to check that if an element $\alpha \in P$ of a model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ realizes one of the sets in the conclusion, then α also realizes the premise. The following Function G uses calculus $\mathbf{D_3}$ to decide a set S. We recall that the formulas in S can be written only using $\mathbf{T}$ and $\mathbf{F}$, since $\overline{\mathbf{T}}$ is a private labelling of the deduction and as far

$$\frac{S, \mathbf{T}(A \wedge B)}{S, \mathbf{T}A, \mathbf{T}B}\mathbf{T}\wedge \quad \frac{S, \mathbf{F}(A \wedge B)}{S, \mathbf{F}A, \mathbf{F}B | S, \mathbf{F}A, \mathbf{T}B | S, \mathbf{T}A, \mathbf{F}B}\mathbf{F}\wedge \quad \frac{S, \mathbf{T}(A \vee B)}{S, \mathbf{T}A | S, \mathbf{T}B}\mathbf{T}\vee \quad \frac{S, \mathbf{F}(A \vee B)}{S, \mathbf{F}A, \mathbf{F}B}\mathbf{F}\vee$$

$$\frac{S, \mathbf{T}(A \to B)}{S, \mathbf{T}B | S, \mathbf{F}A, \overline{\mathbf{T}}(A \to B)}\mathbf{T}\!\to_1 \quad \frac{S, \overline{\mathbf{T}}(A \to B)}{S, \mathbf{T}B}\overline{\mathbf{T}} \text{ provided } S \models A$$

$$\frac{S, \mathbf{F}(A_1 \to B_1), \ldots, \mathbf{F}(A_u \to B_u)}{S_c, V_1 | \ldots | S_c, V_u}\mathbf{F}\!\to$$

where:

$$\begin{aligned} &\text{for } j = 1, \ldots, u \\ V_j = \quad &(\{\mathbf{F}(A_1 \to B_1), \ldots, \mathbf{F}(A_u \to B_u)\} \setminus \{\mathbf{F}(A_j \to B_j)\}) \cup \{\mathbf{T}A_j, \mathbf{F}B_j\} \\ S_c = \quad &\{\mathbf{T}A \in S\} \cup \{\overline{\mathbf{T}}A \in S\} \end{aligned}$$

Figure 5: The calculus $\mathbf{D_3}$

as concerns the deduction $\overline{\mathbf{T}}$-formulas are $\mathbf{T}$ formulas which are not at disposal of deduction.

Function G(S)

1. If S is an inconsistent set, then G returns the proof S;

2. If an α-rule applies to S, then let H be a α-formula of S. If $\mathrm{G}((S \setminus \{H\}) \cup \mathcal{R}_1(H))$ returns a proof π, then G returns the proof $\frac{S}{\pi}Rule(H)$, otherwise G returns `NULL`;

3. If a β-rule applies to S, then let H be a β-formula of S. Let $\pi_1 = \mathrm{G}((S \setminus \{H\}) \cup \mathcal{R}_1(H))$ and $\pi_2 = \mathrm{G}((S \setminus \{H\}) \cup \mathcal{R}_2(H))$. If π_1 or π_2 is `NULL`, then G returns `NULL`, otherwise G returns $\frac{S}{\pi_1|\pi_2}Rule(H)$;

4. If rule $\mathbf{F}\wedge$ applies to S, then let $H = \mathbf{F}(A \wedge B)$ be a formula in S. Let $\pi_1 = \mathrm{G}((S \setminus \{H\}) \cup \{\mathbf{F}A, \mathbf{F}B\})$, $\pi_2 = \mathrm{G}((S \setminus \{H\}) \cup \{\mathbf{F}A, \mathbf{T}B\})$ and $\pi_3 = \mathrm{G}((S \setminus \{H\}) \cup \{\mathbf{T}A, \mathbf{F}B\})$. If π_1, π_2 or π_3 is `NULL`, then G returns `NULL`, otherwise G returns $\frac{S}{\pi_1|\pi_2|\pi_3}\mathbf{F}\wedge$;

5. If $\overline{\mathbf{T}}(A \to B) \in S$ and $S \models A$, then let $\pi_1 = \mathrm{G}((S \setminus \{\overline{\mathbf{T}}(A \to B)\}) \cup \{\mathbf{T}B\})$. If π_1 is `NULL` then G returns `NULL`, otherwise G returns $\frac{S}{\pi_1}\overline{\mathbf{T}}$.

6. If the rule $\mathbf{F} \to$ applies to S, then let $S_{\mathbf{F}\to} = \{\mathbf{F}(A \to B) \in S\}$ and $n = |S_{\mathbf{F}\to}|$. If there exists $i \in \{1, \ldots, n\}$, such that $\pi_i = \mathrm{G}(S_c \cup \mathcal{R}_i(S_{\mathbf{F}\to}))$ is `NULL`, then G returns `NULL`. Otherwise $\pi_1, \ldots, \pi_n$ are proofs and G returns $\frac{S}{\pi_1|\ldots|\pi_n}\mathbf{F}\!\to$;

7. If none of the previous points apply, then G returns `NULL`.

end function .

We need to prove that the properties of $\models$ still hold in the construction of G. Following the lines of Lemma 1, we can prove that relation $\models$ is persistent:

Lemma 2 *Let us suppose that $\mathbf{T}X \in S$. Then in the construction, for every subsequent set S' of S, we have that $S' \models X$.*

In the following lemma we sketch correctness and completeness of G.

Theorem 3 *Let S be a set of formulas. We have that:*
(i) if $\mathrm{G}(S)$ returns `NULL`*, then there exists a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ such $\rho \rhd S$;*
(ii) if $\mathrm{G}(S)$ returns a proof, then for every Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ and for every $\alpha \in P$, $\alpha \not\rhd S$.

Proof 5 *We proceed by induction on the number of nested recursive calls. Note that if function G returns* `NULL`*, the elements of the Kripke model we build are the sets of formulas involved in Steps 6 and 7.*
Basis: *There are no recursive calls.*
(i) If $G(S)$ returns `NULL`*, then Step 7 has been performed. We notice that S is not inconsistent (otherwise Step 1 would have been performed). Indeed, S only contains atomic formulas signed with $\mathbf{T}$ or $\mathbf{F}$. It is easy to prove that the model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$, where $\rho = S$, $P = \{\rho\}$, $\rho \leq \rho$ and $\rho \Vdash p$ iff $\mathbf{T}p \in S$, realizes S.*
(ii) If $G(S)$ returns a proof, then Step 1 is performed, thus S is inconsistent and an inconsistent set is not realizable.
Step: *By induction hypothesis we assume that the proposition holds for all sets S' such that $\mathrm{G}(S')$ requires less than n recursive calls. To prove the proposition holds for a set S such that $\mathrm{G}(S)$ requires n recursive calls, one has to inspect all the possible steps of* G*.*
Let us suppose that Step 5 is performed. *Thus we have that $\overline{\mathbf{T}}(X \to Y) \in S$ and $S \models X$. The call $\mathrm{G}((S \setminus \{\overline{\mathbf{T}}(X \to Y)\}) \cup \{\mathbf{T}Y\})$ is performed. We have to analyze two main cases:*
(i) The call $\mathrm{G}((S \setminus \{\overline{\mathbf{T}}(X \to Y)\}) \cup \{\mathbf{T}Y\})$ returns `NULL`*. By induction hypothesis there is a model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ such that $\rho \rhd S \setminus \{\overline{\mathbf{T}}(X \to Y)\}) \cup \{\mathbf{T}Y\}$, thus $\rho \rhd S$;*
(ii) The call $\mathrm{G}(S \setminus \{\overline{\mathbf{T}}(X \to Y)\}) \cup \{\mathbf{T}Y\})$ returns a proof. We have to show that the rule application is correct. We want to prove that if $S \models X$, α is an element of a Kripke model such that $\alpha \Vdash p$ iff $\mathbf{T}p \in S$ and $\alpha \rhd S$, then $\alpha \rhd \mathbf{T}X$. By construction we have that in the stack of the recursive calls there exists a set S_0 such that $\mathbf{T}(X \to Y) \in S_0$ and a subsequent set S_1 of S_0 such that $\mathbf{F}X \in S_1$. This means that $S_1 \not\models X$.

Claim 1 *Let U be a set of the construction, Z a formula and β an element of a Kripke model respectively meeting the conditions of S, X and α. We claim that:*
(i) if $U \not\models Z$ and $\beta \rhd U$, then $\beta \rhd \mathbf{F}Z$;
(ii) if $U \models Z$ and $\beta \rhd U$, then $\beta \rhd \mathbf{T}Z$.

Proof 6 *The proof of the claim goes by induction on Z:*
Basis: *Z is an atomic formula.*
(i) If $U \not\models Z$, then $\mathbf{T}Z \notin U$ and by the relation of forcing defined on β we have $\beta \rhd \mathbf{F}Z$;

(ii) if $U \models Z$, then $\mathbf{T}Z \in U$, thus $\beta \rhd \mathbf{T}Z$.
Step: *we only prove the case $Z = K \to H$.*
(i) $U \not\models K \to H$. We have two cases: (a) $\mathbf{F}(K \to H) \in U$, thus we immediately get $\beta \rhd \mathbf{F}(K \to H)$; (b) $\mathbf{F}(K \to H) \notin U$. Thus $U \models K$ and $U \not\models H$. By induction hypothesis $\beta \rhd \mathbf{T}K$ and $\beta \rhd \mathbf{F}H$ and we get $\alpha \rhd \mathbf{F}(K \to H)$;
(ii) $U \models K \to H$. Thus $\mathbf{F}(K \to H) \notin U$. Since in the stack of the recursive calls there exists a set S_1 such that $\mathbf{F}(K \to H) \in S_1$, then there exists a subsequent set S_2 of S_1 such that $\mathbf{T}K, \mathbf{F}H \in S_2$. Thus $S_2 \models K$. By Lemma 2, $U \models K$ and thus $U \models H$. By induction hypothesis $\beta \rhd \mathbf{T}H$ and thus $\beta \rhd \mathbf{T}(K \to H)$.

Now, since $\alpha \rhd \overline{\mathbf{T}}(X \to Y)$ means $\alpha \Vdash X \to Y$, by the claim we get $\alpha \Vdash X$ and thus $\alpha \Vdash Y$, that is $\alpha \rhd \mathbf{T}Y$ (note that by construction α meets the conditions of the claim).
Let us suppose that Step 6 is performed. *Note that in this case S contains atomic formulas, formulas of the kind $\overline{\mathbf{T}}(A \to B)$, with $S \not\models A$, and $\mathbf{F}(A \to B)$. Point (ii) is an easy task, since it is based on the fact that rule $\mathbf{F} \to$ preserves the realizability (Point (ii) corresponds to the proof correctness of rule $\mathbf{F} \to$). As for Point (i), by induction hypothesis there exists a Kripke model $\underline{K}' = \langle P', \leq', \rho', \Vdash' \rangle$ such that ρ' realizes one of the set in the conclusion of the rule. We build the following structure $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ such that*

$$\begin{array}{rcl} P & = & P' \cup \{\rho\}, \\ \leq & = & \leq' \cup \{(\rho, \alpha) | \alpha \in P'\}, \\ \Vdash & = & \Vdash' \cup \{(\rho, p) | \mathbf{T}p \in S\}, \end{array}$$

where we set $\rho = S$. The difficult part in proving $\rho \rhd S$ is to show that if $\overline{\mathbf{T}}(A \to B) \in S$, then $\rho \rhd \overline{\mathbf{T}}(A \to B)$. Since if Step 6 is performed and $\overline{\mathbf{T}}(A \to B) \in S$ then $S \not\models A$. Note that by construction, in the stack of recursive calls, there exists a previous set S_0 of S such that $\mathbf{F}A \in S$. Now by proceeding as in the claim above we can prove that $\rho \nVdash A$ and this allow us to get that $\rho \Vdash A \to B$. An analogous argument has to be applied when Step 7 is performed, since in this case S can contain $\overline{\mathbf{T}}$-formulas.

By inspecting the rules of the calculus, it is easy to prove that the procedure terminates and the depth of the deductions is linear in the size of the formula to be decided.

The check to decide if rule $\overline{\mathbf{T}}$ has to be applied is performed on every $\overline{\mathbf{T}}$-formula when no other rule but $\mathbf{F} \to$ or possibly $\overline{\mathbf{T}}$ is applicable. Thus before every application of $\mathbf{F} \to$ or $\mathbf{T} \to$ the check is performed. Note that every application of $\mathbf{F} \to$ and $\mathbf{T} \to$ erases at least an implication, thus along a branch the number of times that the check is performed is linear in the length of the proof. A single check requires a linear number of steps in the number of connectives in the antecedent. Summarizing, along a branch to check if $\models$ holds requires a quadratic number of steps in the size of the formula to be proved.

6 Conclusions

In this paper we have presented two tableau calculi for propositional Dummett logic obeying to the subformula property and whose deductions have respectively quadratic and linear depth in the size of the formula to be decided. The papers presented in literature lack of fulfilling all these features.

Both calculi do not require backtracking and are based on a multiple premise rule. The object language of calculus $\mathbf{D_1}$ contains signs to characterize the semantical status of "forced/non-forced in the next possible world" or "this is last possible world where the formula is not known", which are also employed in [11]. Calculus $\mathbf{D_3}$ uses the signs $\mathbf{T}$ and $\mathbf{F}$, that is the semantics of the signed formulas is restricted to the forcing or non-forcing, and the proof is built-up without the necessity of any particular labelling. Calculus $\mathbf{D_3}$ has a straightforward translation into a sequent calculus.

Our completeness theorems prove that calculi $\mathbf{D_1}$ and $\mathbf{D_3}$ allow to provide a procedure returning a counter model or a proof. In particular, a feature of $\mathbf{D_1}$ is that from a failed proof of a formula A it is possible to extract a counter model for A whose depth is $n+1$ at most, with n the number of propositional variables occurring in A. From a remark on the completeness of $\mathbf{D_1}$ we get calculus $\mathbf{D_3}$. Calculus $\mathbf{D_3}$ shows that the semantics of Dummett logic implies that deduction conveys syntactical information about implicative formulas that can be used to drive the deduction by means of a fast computational check on some formulas which are possibly not at disposal of the deduction.

The multiple premise rules such as $\mathbf{F_n}\widetilde{\mathbf{T}}$ and $\mathbf{F}\rightarrow$, which are analogous to the multiple premise rule introduced in [1], have been criticized because they have an arbitrary number of premises and thus they are supposed not to be suitable for automated deduction. In papers [10, 11] we showed that implementations of systems equipped with a rule analogous to $\mathbf{F_n}\widetilde{\mathbf{T}}$ and $\mathbf{F}\rightarrow$ are far better than the implementation based on decomposition systems of [3, 14], which reduce the formulas to implicative atomic formulas and then applies transitivity rules or procedures based on graph reachability.

We note that it is possible to add some rules to optimize the proof search. As an example, by refining the completeness theorem for $\mathbf{D_3}$, follows that given $\mathbf{T}(A \rightarrow B)$, if A does not contain implications, then we can turn $\mathbf{T}(A \rightarrow B)$ into $\overline{\mathbf{T}}(A \rightarrow B)$, thus saving an application of $\mathbf{T}\rightarrow$ still preserving the completeness. We believe that there are more general cases on the syntax on A that allow to avoid an useless application of rule $\mathbf{T}\rightarrow$. Moreover, since the sign of the occurrence of A in $\mathbf{T}(A \rightarrow B)$ is $\mathbf{F}$, it could be possible to apply our check to $\mathbf{F}$-formulas in order to avoid also useless applications of $\mathbf{F}$-rules.

As a future work, the first question is an investigation along the above line, that could be useful both to deepen the understanding of the proof theory of Dummett logic and to design more efficient decision procedures. Another question is to extend, if possible, the same technique to the first-order case of Dummett logic. Finally, currently we are investigating how to adapt these techniques employed for $\mathbf{D_3}$ to propositional intuitionistic logic, whose Kripke semantics is more complicated than Dummett logic. Our preliminary results show that both the syntactical check and the strategy are more involved than those given for $\mathbf{D_3}$.

References

[1] A. Avellone, M. Ferrari, and P. Miglioli. Duplication-free tableau calculi and related cut-free sequent calculi for the interpolable propositional intermediate logics. *Logic Journal of the IGPL*, 7(4):447–480, 1999.

[2] A. Avron. Simple consequence relations. *Journal of Information and Computation*, 92:276–294, 1991.

[3] A. Avron and B. Konikowska. Decomposition proof systems for gödel-dummett logics. *Studia Logica*, 69(2):197–219, 2001.

[4] M. Baaz and C.G. Fermüller. Analytic calculi for projective logics. In Neil V. Murray, editor, *Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, TABLEAUX '99*, volume 1617 of *Lecture Notes in Computer Science*, pages 36–50. Springer, 1999.

[5] M. Baaz, A. Ciabattoni, and C. G. Fermüller. Hypersequent calculi for Gödel logics – a survey. *J. of Logic and Computation*, 13(6):835–861, 2003.

[6] M. Dummett. A propositional calculus with a denumerable matrix. *Journal of Symbolic Logic*, 24:96–107, 1959.

[7] J. M. Dunn and R. K. Meyer. Algebraic completeness results for Dummett's LC and its extensions. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 17:225–230, 1971.

[8] R. Dyckhoff. A deterministic terminating sequent calculus for Gödel-Dummett logic. *Logic Journal of the IGPL*, 7(3):319–326, 1999.

[9] G. Fiorino. An $O(n \log n)$-SPACE decision procedure for the propositional Dummett Logic. *Journal of Automated Reasoning*, 27(3):297–311, 2001.

[10] G. Fiorino. Fast decision procedure for propositional Dummett logic based on a multiple premise tableau calculus. *Information Sciences*, 180(19):3633–3646, 2010.

[11] G. Fiorino. Refutation in dummett logic using a sign to express the truth at the next possible world. In Toby Walsh, editor, *IJCAI*, pages 869–874. IJCAI/AAAI, 2011.

[12] K. Gödel. On the intuitionistic propositional calculus. In S. Feferman et al, editor, *Collected Works*, volume 1. Oxford University Press, 1986.

[13] P. Hajek. *Metamathematics of Fuzzy Logic*. Kluwer, 1998.

[14] D. Larchey-Wendling. Graph-based decision for Gödel-Dummett logics. *J. Autom. Reasoning*, 38(1-3):201–225, 2007.

[15] G. Metcalfe, N. Olivetti, and D. M. Gabbay. Goal-directed calculli for Gödel-Dummett logics. In Matthias Baaz and Johann A. Makowsky, editors, *Computer Science Logic, 17th International Workshop, CSL 2003, 12th Annual Conference of the EACSL, and 8th Kurt Gödel Colloquium, KGC 2003, Vienna, Austria, August 25-30, 2003, Proceedings*, volume 2803 of *Lecture Notes in Computer Science*, pages 413–426. Springer, 2003.

[16] A. Visser. On the completenes principle: A study of provability in heyting's arithmetic and extensions. *Annals of Mathematical Logic*, 22(3):263 – 295, 1982.

[17] N. N. Vorob'ev. A new algorithm of derivability in a constructive calculus of statements. In *Sixteen papers on logic and algebra*, volume 94 of *American Mathematical Society Translations, Series 2*, pages 37–71. American Mathematical Society, Providence, R.I., 1970.

A My review of CLS reviewers

BERTRAND MEYER:

> "Refereeing should be what it was before science publication turned into a business: scientists giving their polite but frank opinion on the work of other scientists." (CACM, Vol. 54 No. 11).

I submitted this paper to IJCAR 2012 and, in the present form to CSL 2012. In both cases it was rejected. Now it is my turn to give a review of reviewers and spend some words about my experience as an author in proof-theory.
I start with the facts: at CLS the paper had three reviewers. The first gave an accept and was the only reviewer to read the paper. Reviewers 2 and 3 clearly read the introduction, at most, as anyone can understand from the general comments they give.

```
----------------------- REVIEW 2 ---------------------
PAPER: 49
TITLE: Terminating Calculi for Propositional Dummett Logic with Subformula Property
AUTHORS: Guido Fiorino

OVERALL RATING: -3 (strong reject)

This paper is presenting two new calculi for propositional Dummet logic aka
Goedel logic. This logic can be viewed both as an intermediate logic
(intuitionistic logic + axiom scheme (a->b)\/(b->a) ) or as a fuzzy logic
with operators over the unit interval.
```

```
While this is a nice paper in pure logic, it is not clear to me why
this paper is submitted to CSL.

1) The paper contains no motivation that relates to computer science
except for a reference to a famous 1991 paper [2] on simple consequence
relations. It is neither clear why [2] is called "recent" nor why it
is considered a CS motivation.

2) There already exist many calculi for this logic including
[1,3,8,9,10,11,14]

3) The paper contains no generic discussion why tableau calculi are
the right approach for Dummett logic. Given the simplicity of
the logic considered, and its simple semantic characterization in
terms of the unit interval (i.e., Goedel logic), one would expect that a
DPLL style procedure similar to standard SAT and SMT solving is more
efficient in practice. Reductions of fuzzy logics to arithmetic
solvers have been proposed by Haehnle and others in the 90ies.

4) There also is no methodological breakthrough which can be
generalized to other logics.

In conclusion I think the paper is lacking motivation.
----------------------- REVIEW 3 ---------------------
PAPER: 49
TITLE: Terminating Calculi for Propositional Dummett Logic with Subformula Property
AUTHORS: Guido Fiorino

OVERALL RATING: 1 (weak accept)

This paper describes two terminating calculi for propositional Goedel
Dummett logicwith subformula property which is not the important point as
subformula property can be always obtained by suitable choice of the
linguistic frame. The first calculus is completely straightforwardly
obtained from valuations in linearly ordered Kripke semantics, the claim on
the size of models is however trivial as only valuations of variables count
in Goedel-Dummett logics, they are projective. The second calculus is much
more interesting and the paper should concentrate on this. Furthermore the
paper has the deficiency for non-experts of providing no single example.
------------------------------------------------------
```

In my opinion the reviewers have a conflict interest and want to make space for their papers, thus I consider them in bad faith.
As regard review 3, he/she gives a borderline rating. Here we meet the first characteristic of many reviews in proof-theory: “the topic is not important”. The reviewer pretend of ignoring that there are many papers about calculi

with the subformula property and many authors consider this property important. Statement "suitable choice of the linguistic frame", means to have hypertableaux/hypersequents and/or labelled systems. The advantage of my systems is in evidence in the introduction (see paragraph starting with "Papers [4] and [15] provide calculi ...") but the reviewer has ignored my considerations. This is one of the behaviours that I observed by reviewers in proof-theory: minimize the idea and the interest of the problem, in order not to give importance to the whole paper, even if there are many papers along the same line (note that at CLS 2003 a paper addressing the same question was proposed and in all the quoted papers the efficiency or the subformula property or the proof-system or the termination is addressed).
Reviewer 2 is the typical coward that hide himself under anonymous review to make nasty statements and to give a very bad mark without entering into technical details. The aim of the reviewer is clear: to be sure that the paper is rejected, independently of the others reviews. A strong reject implies that the paper contains technical errors that cannot be clearly fixed. But here the review is not scientific and the program committee is responsible for this (I wonder if the reviewer has read the whole introduction or at least the abstract).
The reviewer states that he/she does not understand my submission to the conference. To understand the submission he/she should read CSL call for paper. The paper perfectly matches the topic both in proof-theory and automated deduction. Point 2) is perfect to understand the bad faith of the reviewer: the argument is that there are enough papers on Dummett logic thus we do not need more. It's a pity, my paper is late! On this base, I aspect that in the future CSL will reject papers on Dummett/Goedel logic, **independently of the name(s) of the author(s)**. Also Point 3) deserves attention, because it is another typical scheme to reject a/my paper: "why to provide a calculus when there is a translation into another logic?" On this base we cannot have calculi for propositional intuitionistic logic, since there exist translations in S4 or classical logic and so on for many other logics. Variants of this are "why do you use semantical techniques?" and "I don't like the presentation" and, following the Point 4) "the result is not interesting because it cannot be generalized".

I charge the reviewers to have used anonymous review to be unfair, biased and in bad faith instead of giving a frank scientific opinion.

The problem is not the content of the paper but the name of the author. Proof-theory is a close world, a kind of private club made of some schools and newcomers are not welcome. Thus can happen that also a trivial mistake as a typo is used as an excuse to give the minimum rate and the original ideas are ignored. The result is that for authors that are not part of the club it is almost impossible to have a paper accepted to a conference, the timings to have a paper accepted on a journal are amplified and when the papers is published it is not cited, also if pertinent.

For these reasons I support the statement of Bertrand Meyer.