View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by AIR Universita degli studi di Milano

Simplification Rules for Intuitionistic Propositional Tableaux

MAURO FERRARI, Università degli Studi dell'Insubria
CAMILLO FIORENTINI, Università degli Studi di Milano
GUIDO FIORINO, Università degli Studi di Milano-Bicocca

The implementation of a logic requires, besides the definition of a calculus and a decision procedure, the development of techniques to reduce the search space. In this article we introduce some simplification rules for Intuitionistic propositional logic that try to replace a formula with an equi-satisfiable "simpler" one with the aim to reduce the search space. Our results are proved via semantical techniques based on Kripke models. We also provide an empirical evaluation of their impact on implementations.

Categories and Subject Descriptors: F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*Proof theory; Mechanical theorem proving*

General Terms: Theory

Additional Key Words and Phrases: Decision procedures, intuitionistic logic, simplification rules, tableau calculi

ACM Reference Format:
Ferrari, M., Fiorentini, C., and Fiorino, G. 2012. Simplification rules for intuitionistic propositional tableaux. ACM Trans. Comput. Logic 13, 2, Article 14 (April 2012), 23 pages.
DOI = 10.1145/2159531.2159536 http://doi.acm.org/10.1145/2159531.2159536

1. INTRODUCTION

It is well known that the effective implementation of a logic requires, besides the definition of a calculus and a decision procedure, the development of *simplification/optimization* techniques to reduce the search space. In the case of Classical Logic the development of simplification/optimization techniques has a long history that goes back to the very beginning of Davis-Putnam procedure implementation [Davis and Putnam 1960; Davis et al. 1962] and resolution implementation [Robinson 1965] and continues nowadays. The situation is quite different in the case of tableau calculi for Intuitionistic Logic, indeed in this framework very little work has been done in this direction despite the interest for this logic. As far as we know, the only works that address these issues in the context of tableau calculi are Massacci [1998] and Hustadt and Schmidt [1998]; however, these papers essentially refer to classical and modal logics, even if some of their ideas can be adapted to the case of Intuitionistic Logic.

Before entering into the details of our work, we remark that in the setting of automated deduction the word *optimization* refers to different aspects. To clarify our discussion we introduce the following distinction: we call *simplification technique* a

Authors' addresses: M. Ferrari, Department of Information and Communication, University of Insubria; email: mauro.ferrari@uninsubria.it; C. Fiorentini, Department of Information Science, University of Milan; G. Fiorino, Department of Quantitative Methods, University of Milan-Bicocca.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from the Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701, USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2012 ACM 1529-3785/2012/04-ART14 $10.00
DOI 10.1145/2159531.2159536 http://doi.acm.org/10.1145/2159531.2159536

method that enables to replace a formula with an equi-satisfiable one with the aim to reduce the search space; we call *optimization technique* a method to visit the search space that tries to avoid inessential branches or useless backtracking. Examples of the former technique are *unit propagation* [Dowling and Gallier 1984; Zhang and Stickel 2000], Boolean simplification rules [Massacci 1998], the Syntactic Trees Transformations [Aguilera et al. 2001] and the simplification techniques described in Hustadt and Schmidt [1998] for the modal logic **K**. A well-known example of the latter are backjumping rules; see, for instance, the one described in Hustadt and Schmidt [1998] for the modal logic **KE** and, in the framework of Intuitionistic Logic, the strategy to reduce backtracking presented in Weich [1998].

In this article we concentrate our attention on simplification rules for tableau calculi for Intuitionistic propositional Logic. The aim of these rules is to reduce the formulas to be analyzed as much as possible before applying a branch or a noninvertible rule (which requires backtracking). Our starting point is the simplification technique presented in Massacci [1998], which consists in replacing every occurrence of a formula assumed to be true with the logical constant $\top$ and every occurrence of a formula assumed to be false with $\bot$. As an example, if A can be replaced with $\top$, we can rewrite the formula $A \vee B$ as $\top \vee B$, which can be simplified to $\top$. In the tableau systems for classical logic the notions of provable and unprovable are codified by means of the *signs* **T** and **F** [Smullyan 1968]. It is well known that the sign (polarity) of a formula determines also the sign (polarity) of every occurrence of its subformulas in a proof [Kleene 1967, Lemma 14, page 337]. If the sign of a propositional variable occurring in a set of signed formulas is always **T** (respectively **F**), then such a variable is equivalent to $\top$ (respectively $\bot$). This technique can be applied also in the intuitionistic setting, but, differently from classical logic, here the signs **T** and **F** are not dual, in particular $\mathbf{F}A$ does not imply that A is equivalent to $\bot$. Thus, replacement can be performed only if further conditions are satisfied. In this article we present several criteria under which such a replacement can be applied. In Sections 4 and 5 we introduce the rules **T**-permanence, **T**$\neg$-permanence and **F**-permanence that allow us to replace, under suitable conditions, propositional variables with $\top$ and $\bot$. After the replacements, we can apply the Boolean simplification rules defined in Section 3 to reduce the size of the set of formulas to be decided. The results in Sections 6 and 7 extend those presented in previous sections. In Section 8 we discuss the impact of our simplification rules on the performances of a theorem prover. Finally, in Section 9 we discuss some possible extensions of our results.

We remark that our simplification rules are independent from the tableau calculus at hand. Moreover, these rules are invertible. As discussed in Section 2, this means that we can apply them at any point of a proof-search strategy without affecting its completeness. Finally, via the usual translation [Avellone et al. 1999], these rules can also be applied in implementations based on sequent calculi. We conclude noticing that all the results are proved by means of the Kripke semantics for Intuitionistic Logic, which is the guide to understand our simplification rules.

2. NOTATION AND PRELIMINARIES

We consider the propositional language $\mathcal{L}$ based on a denumerable set of propositional variables $\mathcal{PV}$, the logical connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$ and the logical constants $\top$ and $\bot$. Writing formulas we assume that $\neg$ binds stronger than $\wedge$ and $\vee$, which in turn are stronger than $\rightarrow$.

We recall the main definitions about Kripke semantics (see, e.g., Chagrov and Zakharyaschev [1997] for more details). An *(intuitionistic) Kripke model* for $\mathcal{L}$ is a

structure $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$, where $\langle P, \leq, \rho \rangle$ is a poset with minimum ρ and the *forcing* relation $\Vdash$ is a binary relation on $P \times \mathcal{PV}$ such that $\alpha \Vdash p$ and $\alpha \leq \beta$ imply $\beta \Vdash p$ (monotonicity property). The forcing relation extends to arbitrary formulas of $\mathcal{L}$ as follows:

— $\alpha \Vdash \top$;
— $\alpha \Vdash \bot$ does not hold;
— $\alpha \Vdash A \wedge B$ iff $\alpha \Vdash A$ and $\alpha \Vdash B$;
— $\alpha \Vdash A \vee B$ iff $\alpha \Vdash A$ or $\alpha \Vdash B$;
— $\alpha \Vdash A \rightarrow B$ iff, for every $\beta \in P$ such that $\alpha \leq \beta$, $\beta \Vdash A$ implies $\beta \Vdash B$;
— $\alpha \Vdash \neg A$ iff, for every $\beta \in P$ such that $\alpha \leq \beta$, $\beta \Vdash A$ does not hold.

With the notation $\alpha \nVdash A$ we mean that $\alpha \Vdash A$ does not hold. It is easy to prove that the monotonicity property holds for arbitrary formulas, that is, $\alpha \Vdash A$ and $\alpha \leq \beta$ imply $\beta \Vdash A$. A formula A *is valid in a Kripke model* $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ iff $\rho \Vdash A$. It is well-known that Intuitionistic propositional Logic **Int** coincides with the set of formulas valid in all Kripke models [Chagrov and Zakharyaschev 1997].

A tableau calculus $\mathcal{T}$ works on *signed formulas*, namely formulas of $\mathcal{L}$ prefixed with one of the signs **T** or **F**. The semantics of formulas extends to signed formulas as follows. Given a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$, $\alpha \in P$ and a signed formula H, α *realizes* H *in* $\underline{K}$ ($\underline{K}, \alpha \rhd H$) iff:

— $H \equiv \mathbf{T}A$ and $\alpha \Vdash A$;
— $H \equiv \mathbf{F}A$ and $\alpha \nVdash A$.

$\underline{K}$ realizes H ($\underline{K} \rhd H$) iff $\underline{K}, \alpha \rhd H$ for some $\alpha \in P$. H is *realizable* iff $\underline{K} \rhd H$ for some Kripke model $\underline{K}$. These definitions extend in the obvious way to sets Δ of signed formulas: for instance, $\underline{K}, \alpha \rhd \Delta$ means that $\underline{K}, \alpha \rhd H$, for every $H \in \Delta$; Δ is realizable iff $\underline{K}, \alpha \rhd H$ for some Kripke model $\underline{K}$ and α in $\underline{K}$. By definition, $A \in \mathbf{Int}$ iff $\mathbf{F}A$ is not realizable.

We remark that, by the monotonicity property, the **T**-signed formulas are *persistent*, namely: $\underline{K}, \alpha \rhd \mathbf{T}A$ and $\alpha \leq \beta$ imply $\underline{K}, \beta \rhd \mathbf{T}A$. On the other hand, **F**-signed formulas are not persistent.

In general, a tableau calculus $\mathcal{T}$ consists of a set of rules of the form:

$$\frac{\Delta}{\Delta_1 \mid \cdots \mid \Delta_n}\, r,$$

where Δ (the *premise* of r) and $\Delta_1, \ldots, \Delta_n$ (the *consequences* of r) are nonempty sets of signed formulas of $\mathcal{L}$. A *proof-table* for Δ is a tree τ such that:

— the root of τ is Δ;
— for every node Δ' in τ, if $\Delta_1, \ldots, \Delta_n$ are the immediate successors of Δ', then there exists an instance of a rule r of $\mathcal{T}$ having Δ' as premise and $\Delta_1, \ldots, \Delta_n$ as consequences.

A set Δ of signed formulas is *contradictory* if either $\mathbf{T}\bot \in \Delta$ or $\mathbf{F}\top \in \Delta$. When all the leaves of a proof-table τ are contradictory, we say that τ is *closed*. A finite set of signed formulas Δ is *provable* in $\mathcal{T}$ iff there exists a closed proof-table for Δ.

A tableau calculus $\mathcal{T}$ is *complete* for **Int** iff, for every finite set of signed formulas Δ, Δ is provable in $\mathcal{T}$ iff Δ is not realizable (hence, $A \in \mathbf{Int}$ iff $\mathbf{F}A$ is provable in $\mathcal{T}$). Let $\mathcal{T}$ be a complete calculus; to decide the realizability of Δ we have to search for a closed

proof-table τ for Δ. A proof-search algorithm tries to build τ by applying the rules of $\mathcal{T}$ in all possible ways: whenever a node Δ' of the proof-table is generated, a rule r of $\mathcal{T}$ is applied to Δ' and, recursively, the search proceeds on the new nodes. Let r be a rule with premise Δ and consequences $\Delta_1, \ldots, \Delta_n$:

— r is *sound* iff Δ realizable implies that there exists $k \in \{1, \ldots, n\}$ such that Δ_k is realizable;
— r is *invertible* iff r is sound and, for every $k \in \{1, \ldots, n\}$, if Δ_k is realizable then Δ is realizable.

In proof-search, the application of an invertible rule r of a complete calculus $\mathcal{T}$ does not require backtracking. Indeed, suppose to apply r to Δ and let $\Delta_1, \ldots, \Delta_n$ be the consequences of r. If, for some $k \in \{1, \ldots, n\}$, Δ_k is not provable, by the completeness of $\mathcal{T}$ it follows that Δ_k is realizable hence, being r invertible, Δ is realizable. We can conclude that Δ is not provable, and there is no need to try the application of another rule to Δ.

A complete tableau calculus is *terminating* if proof-search can be accomplished in finite time. Accordingly, for every finite set Δ, proof-search terminates in finite time and, if no closed-proof table is found, Δ is realizable.

The simplification rules introduced in this article are intended to be added to a tableau calculus $\mathcal{T}$ which is complete for **Int** and terminating. The aim of these rules is to narrow the proof-search space. An essential feature we require for simplification rules is invertibility, so to avoid backtracking on their application. Note that the proof of the invertibility of a rule r only depends on its definition (and not on $\mathcal{T}$); in many cases such a proof is immediate, sometimes it is rather involved (as for the rules in Section 6).

Simplification rules act on a set Δ by replacing subformulas occurring in formulas of Δ with "simpler" ones. An example of invertible simplification rule is the rule substituting any occurrence of $A \wedge \bot$ in Δ with $\bot$. One expects that such a replacement narrows the search-space (indeed, complex formulas collapse to $\bot$) and has not side effects on the termination of proof-search. Clearly, the impact of simplification rules on proof-search must be checked case-by-case, depending on the calculus $\mathcal{T}$ at hand; we trust that the known calculi for **Int** should take advantage of adopting the simplification rules introduced in the article. For the sake of concreteness, in the rest of the article we study the benefits of simplification rules on the tableau calculus $\mathbf{T}_{\mathrm{Int}}$, a slight variant of the calculus **Tab** [Avellone et al. 2008].

2.1. The Calculus $\mathbf{T}_{\mathrm{Int}}$

The rules of the calculus $\mathbf{T}_{\mathrm{Int}}$ are presented in Figure 1. In the formulation of the rules, we use the notation Δ, H as a shorthand for $\Delta \cup \{H\}$. In the premise of a rule, writing Δ, H we assume that $H \notin \Delta$. $\mathbf{T}_{\mathrm{Int}}$ essentially coincides with the terminating calculus **Tab** for **Int** presented in Avellone et al. [2008]. More in details, **Tab** uses the sign $\mathbf{F_c}$ besides the usual signs **T** and **F**; in $\mathbf{T}_{\mathrm{Int}}$ we do not use the sign $\mathbf{F_c}$ and the rules for $\mathbf{F_c}$ are translated by substituting $\mathbf{F_c}A$ with the equivalent signed formula $\mathbf{T}\neg A$. We made this choice to ease the presentation: the sign $\mathbf{F_c}$, which is not standard in tableau calculi, would require ad-hoc simplification rules. The rules $contr_1$ and $contr_2$ of $\mathbf{T}_{\mathrm{Int}}$ are introduced to represent the closure rules of **Tab** in our setting. The rule MP is a generalization of the rule $\mathbf{T} \to Atom$ of **Tab** ($\mathbf{T} \to Atom$ can be applied only if A is a propositional variable). Thus, one can define a one-to-one translation between proof-tables of **Tab** and proof-tables of $\mathbf{T}_{\mathrm{Int}}$. Due to this correspondence, the results

$$\frac{\Delta, \mathbf{T}(A \wedge B)}{\Delta, \mathbf{T}A, \mathbf{T}B}\,\mathbf{T}\wedge \qquad \frac{\Delta, \mathbf{F}(A \wedge B)}{\Delta, \mathbf{F}A \mid \Delta, \mathbf{F}B}\,\mathbf{F}\wedge \qquad \frac{\Delta, \mathbf{T}\neg(A \wedge B)}{\Delta_{\mathbf{T}}, \mathbf{T}\neg A \mid \Delta_{\mathbf{T}}, \mathbf{T}\neg B}\,\mathbf{T}\neg\wedge$$

$$\frac{\Delta, \mathbf{T}(A \vee B)}{\Delta, \mathbf{T}A \mid \Delta, \mathbf{T}B}\,\mathbf{T}\vee \qquad \frac{\Delta, \mathbf{F}(A \vee B)}{\Delta, \mathbf{F}A, \mathbf{F}B}\,\mathbf{F}\vee \qquad \frac{\Delta, \mathbf{T}\neg(A \vee B)}{\Delta, \mathbf{T}\neg A, \mathbf{T}\neg B}\,\mathbf{T}\neg\vee$$

$$\frac{\Delta, \mathbf{T}A, \mathbf{T}(A \to B)}{\Delta, \mathbf{T}A, \mathbf{T}B}\,MP$$

$$\frac{\Delta, \mathbf{F}(A \to B)}{\Delta_{\mathbf{T}}, \mathbf{T}A, \mathbf{F}B}\,\mathbf{F}\to \qquad \frac{\Delta, \mathbf{T}\neg(A \to B)}{\Delta_{\mathbf{T}}, \mathbf{T}A, \mathbf{T}\neg B}\,\mathbf{T}\neg\to$$

$$\frac{\Delta, \mathbf{F}\neg A}{\Delta_{\mathbf{T}}, \mathbf{T}A}\,\mathbf{F}\neg \qquad \frac{\Delta, \mathbf{T}\neg\neg A}{\Delta_{\mathbf{T}}, \mathbf{T}A}\,\mathbf{T}\neg\neg$$

$$\frac{\Delta, \mathbf{T}((A \wedge B) \to C)}{\Delta, \mathbf{T}(A \to (B \to C))}\,\mathbf{T}\to\wedge \qquad \frac{\Delta, \mathbf{T}(\neg A \to B)}{\Delta_{\mathbf{T}}, \mathbf{T}A \mid \Delta, \mathbf{T}B}\,\mathbf{T}\to\neg$$

$$\frac{\Delta, \mathbf{T}((A \vee B) \to C)}{\Delta, \mathbf{T}(A \to p), \mathbf{T}(B \to p), \mathbf{T}(p \to C)}\,\mathbf{T}\to\vee \quad \text{with } p \text{ a new atom}$$

$$\frac{\Delta, \mathbf{T}((A \to B) \to C)}{\Delta_{\mathbf{T}}, \mathbf{T}A, \mathbf{F}p, \mathbf{T}(p \to C), \mathbf{T}(B \to p) \mid \Delta, \mathbf{T}C}\,\mathbf{T}\to\to \quad \text{with } p \text{ a new atom}$$

$$\frac{\Delta, \mathbf{T}A, \mathbf{F}A}{\Delta, \mathbf{T}\bot}\,contr_1 \qquad \frac{\Delta, \mathbf{T}A, \mathbf{T}\neg A}{\Delta, \mathbf{T}\bot}\,contr_2$$

$$\text{where } \Delta_{\mathbf{T}} = \{\mathbf{T}A \mid \mathbf{T}A \in \Delta\}$$

Fig. 1. The $\mathbf{T}_{\mathrm{Int}}$ calculus.

proved in Avellone et al. [2008] for **Tab** also apply to $\mathbf{T}_{\mathrm{Int}}$. In particular, to prove the termination of $\mathbf{T}_{\mathrm{Int}}$ we need the following degree functions [Avellone et al. 2008]:

$$\begin{aligned}
\mathrm{dg}_{\to}(\bot) &= \mathrm{dg}_{\to}(\top) = \mathrm{dg}_{\to}(p) = 0 \text{ with } p \in \mathcal{PV}\\
\mathrm{dg}_{\to}(A \odot B) &= \mathrm{dg}_{\to}(A) + \mathrm{dg}_{\to}(B) \text{ with } \odot \in \{\wedge, \vee\}\\
\mathrm{dg}_{\to}(\neg A) &= \mathrm{dg}_{\to}(A) + 1\\
\mathrm{dg}_{\to}(A \to B) &= \mathrm{dg}_{\to}(A) + \mathrm{dg}_{\to}(B)\\
\mathrm{dg}(\bot) &= \mathrm{dg}(\top) = 1\\
\mathrm{dg}(p) &= 2 \text{ with } p \in \mathcal{PV}\\
\mathrm{dg}(A \wedge B) &= \mathrm{dg}(A) + \mathrm{dg}(B) + 2\\
\mathrm{dg}(A \vee B) &= \mathrm{dg}(A) + \mathrm{dg}(B) + 9\\
\mathrm{dg}(A \to B) &= \mathrm{dg}(A) + \mathrm{dg}(B) + \mathrm{dg}_{\to}(A) + 1\\
\mathrm{dg}(\neg A) &= \mathrm{dg}(A) + 1\\
\mathrm{dg}(\mathcal{S}A) &= \mathrm{dg}(A) \text{ with } \mathcal{S} \in \{\mathbf{T}, \mathbf{F}\}\\
\mathrm{dg}(\Delta) &= \sum_{H \in \Delta} \mathrm{dg}(H).
\end{aligned}$$

$$\frac{\Delta}{\Delta[\bot/A \wedge \bot]}\mathrm{S}\wedge\bot \qquad \frac{\Delta}{\Delta[\bot/\bot \wedge A]}\mathrm{S}\bot\wedge \qquad \frac{\Delta}{\Delta[A/A \wedge \top]}\mathrm{S}\wedge\top \qquad \frac{\Delta}{\Delta[A/\top \wedge A]}\mathrm{S}\top\wedge$$

$$\frac{\Delta}{\Delta[A/A \vee \bot]}\mathrm{S}\vee\bot \qquad \frac{\Delta}{\Delta[A/\bot \vee A]}\mathrm{S}\bot\vee \qquad \frac{\Delta}{\Delta[\top/A \vee \top]}\mathrm{S}\vee\top \qquad \frac{\Delta}{\Delta[\top/\top \vee A]}\mathrm{S}\top\vee$$

$$\frac{\Delta}{\Delta[\top/\bot \to A]}\mathrm{S}\bot\to \qquad \frac{\Delta}{\Delta[\neg A/A \to \bot]}\mathrm{S}\to\bot \qquad \frac{\Delta}{\Delta[A/\top \to A]}\mathrm{S}\top\to \qquad \frac{\Delta}{\Delta[\top/A \to \top]}\mathrm{S}\to\top$$

$$\frac{\Delta}{\Delta[\bot/\neg\top]}\mathrm{S}\neg\top \qquad \frac{\Delta}{\Delta[\top/\neg\bot]}\mathrm{S}\neg\bot$$

Fig. 2. Boolean simplification rules.

One can easily check that all the rules r of $\mathbf{T}_{\mathrm{Int}}$ are decreasing w.r.t. the function dg, namely: if Δ' is any of the consequences obtained by applying r to a finite set Δ, then $\mathrm{dg}(\Delta') < \mathrm{dg}(\Delta)$. As a consequence, any proof-table for Δ has height bounded by $\mathrm{dg}(\Delta)$, and this implies that $\mathbf{T}_{\mathrm{Int}}$ is terminating. In Avellone et al. [2008], an efficient proof-search algorithm for **Tab** is presented and an efficient implementation, called PITP, is discussed.

All the simplification rules introduced in this article are decreasing w.r.t. dg (the proofs are immediate and will be omitted), thus their addition to $\mathbf{T}_{\mathrm{Int}}$ does not affect the termination of proof-search. We show by means of significant examples that such rules actually reduce the proof-search space.

3. REPLACEMENT AND BOOLEAN SIMPLIFICATION RULES

In this section we recall the simplification rules introduced in Avellone et al. [2008]. Such rules allow us to simplify signed formulas by replacing some of their subformulas either with $\bot$ or $\top$. First of all we introduce the notion of *(signed) formula substitution*: given a signed formula H and two formulas A and B, we denote with $H[B/A]$ the signed formula obtained by replacing every occurrence of A in H with B. If Δ is a set of signed formulas, $\Delta[B/A]$ is the set of signed formulas $H[B/A]$ such that $H \in \Delta$.

It is easy to prove the following facts:

LEMMA 3.1. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ be a Kripke model, H a signed formula, A a formula, and $\alpha \in P$.*

(i) If $\underline{K}, \alpha \rhd \mathbf{T}A$, then $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd H[\top/A]$.
(ii) If $\underline{K}, \alpha \rhd \mathbf{T}\neg A$, then $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd H[\bot/A]$.

Let us consider the following rules:

$$\frac{\Delta, \mathbf{T}A}{\Delta[\top/A], \mathbf{T}A}\text{Replace-}\mathbf{T} \qquad \frac{\Delta, \mathbf{T}\neg A}{\Delta[\bot/A], \mathbf{T}\neg A}\text{Replace-}\mathbf{T}\neg.$$

By Lemma 3.1, Theorem 3.2 immediately follows.

THEOREM 3.2. *The rules* Replace-**T** *and* Replace-**T**$\neg$ *are invertible.*

These rules are the intuitionistic version of the analogous rules for classical tableaux discussed in Massacci [1998]. After having applied a replacement rule, we can simplify the formulas by means of the invertible Boolean simplification rules in Figure 2.

Now, we present the analogous rule for **F**-signed formulas introduced in Avellone et al. [2008]. We remark that, differently from Classical Logic, where the signs **F** and

T are dual, in Intuitionistic Logic **T**-signed formulas are persistent while **F**-signed formulas are not. Due to this asymmetry the replacement rule for **F**-signed formulas involves a notion of *partial substitution* which is weaker than the "full" substitution used so far. Formally, given the formulas Z, A and B, we denote with $Z\{B/A\}$ the *partial substitution of A with B in Z* defined as follows:

— if $Z = A$, then $Z\{B/A\} = B$;
— if $Z = (X \odot Y)$, then $Z\{B/A\} = X\{B/A\} \odot Y\{B/A\}$, where $\odot \in \{\wedge, \vee\}$;
— if $Z = X \to Y$ or $Z = \neg X$ or Z is a propositional variable different from A, then Z{B/A} = Z.

We remark that partial substitutions do not act on subformulas with main connective $\to$ or $\neg$. For instance, while $((X \to Y) \vee Y)[\bot/Y]$ produces $(X \to \bot) \vee \bot$, the partial substitution $((X \to Y) \vee Y)\{\bot/Y\}$ yields $(X \to Y) \vee \bot$. Given a signed formula $\mathcal{S}Z$ with $\mathcal{S} \in \{\mathbf{T}, \mathbf{F}\}$, we denote with $\mathcal{S}Z\{B/A\}$ the signed formula $\mathcal{S}(Z\{B/A\})$. Given a set of signed formulas Δ, $\Delta\{B/A\}$ is the set containing $K\{B/A\}$ for every $K \in \Delta$.

Proceeding by induction on the structure of the signed formula H it is easy to prove the following result.

LEMMA 3.3. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model, let $\alpha \in P$ and let H and $\mathbf{F}A$ be two signed formulas. If $\underline{K}, \alpha \rhd \mathbf{F}A$, then $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd H\{\bot/A\}$.*

Now, let us consider the rule:

$$\frac{\Delta, \mathbf{F}A}{\Delta\{\bot/A\}, \mathbf{F}A}\text{Replace-}\mathbf{F}.$$

By the previous lemma, Theorem 3.4 immediately follows.

THEOREM 3.4. *The rule* Replace-**F** *is invertible.*

As discussed in Avellone et al. [2008], the rules Replace-**T**, Replace-**T**$\neg$ and Replace-**F** together with the Boolean simplification rules can considerably reduce the search space.

4. RULES FOR PROPOSITIONAL VARIABLES WITH CONSTANT SIGN

The rules Replace-**T** and Replace-**T**$\neg$ of Section 3 can be applied whenever a signed formula $\mathbf{T}A$ or $\mathbf{T}\neg A$ occurs in Δ. In this section we exploit some conditions under which we can replace a propositional variable p applying the rules of Section 3 also when $\mathbf{T}p$ or $\mathbf{T}\neg p$ does not explicitly occur in Δ. The condition for the applicability of these rules is based on the notion of *polarity* of p: p can be eliminated from Δ (replaced with $\top$ or $\bot$) if all the occurrences of p in Δ have the same polarity. Our notion of positive and negative polarity of a propositional variable p in a signed formula H is formalized by the relations $p \preceq^+ H$ (*p positively occurs in H*) and $p \preceq^- H$ (*p negatively occurs in H*). Hereafter we use $\mathcal{S}$ to denote either **T** or **F**. The definition of $p \preceq^l H$, with $l \in \{+, -\}$, is by induction on the structure of H:

— $p \preceq^- \mathbf{F}p$ and $p \preceq^+ \mathbf{T}p$
— $p \preceq^l \mathcal{S}\top$ and $p \preceq^l \mathcal{S}\bot$
— $p \preceq^l \mathcal{S}q$, where q is any propositional variable such that $q \neq p$
— $p \preceq^l \mathcal{S}(A \odot B)$ iff $p \preceq^l \mathcal{S}A$ and $p \preceq^l \mathcal{S}B$, where $\odot \in \{\wedge, \vee\}$
— $p \preceq^l \mathbf{F}(A \to B)$ iff $p \preceq^l \mathbf{T}A$ and $p \preceq^l \mathbf{F}B$
— $p \preceq^l \mathbf{T}(A \to B)$ iff $p \preceq^l \mathbf{F}A$ and $p \preceq^l \mathbf{T}B$

— $p \preceq^l \mathbf{F}\neg A$ iff $p \preceq^l \mathbf{T}A$
— $p \preceq^l \mathbf{T}\neg A$ iff $p \preceq^l \mathbf{F}A$.

Given a set of signed formulas Δ, $p \preceq^l \Delta$ iff, for every $H \in \Delta$, $p \preceq^l H$.

Now, let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model and p a propositional variable. We define the Kripke models $\underline{K}_p^+$ and $\underline{K}_p^-$ as follows:

— $\underline{K}_p^+ = \langle P, \leq, \rho, \Vdash' \rangle$, where $\Vdash' = \Vdash \cup \{(\alpha, p) \mid \alpha \in P\}$;
— $\underline{K}_p^- = \langle P, \leq, \rho, \Vdash' \rangle$, where $\Vdash' = \Vdash \setminus \{(\alpha, p) \mid \alpha \in P\}$.

Note that, for every $\alpha \in P$, $\underline{K}_p^+, \alpha \rhd \mathbf{T}p$ and $\underline{K}_p^-, \alpha \rhd \mathbf{T}\neg p$. We prove that, moving from the model $\underline{K}$ to the model $\underline{K}_p^l$, with $l \in \{+, -\}$, the validity of signed formulas H such that $p \preceq^l H$ is preserved.

LEMMA 4.1. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model, let H be a signed formula, and let p be a propositional variable.*

(1) *If $p \preceq^+ H$ then, for every $\alpha \in P$, $\underline{K}, \alpha \rhd H$ implies $\underline{K}_p^+, \alpha \rhd H$.*
(2) *If $p \preceq^- H$ then, for every $\alpha \in P$, $\underline{K}, \alpha \rhd H$ implies $\underline{K}_p^-, \alpha \rhd H$.*

PROOF. The proof easily goes by structural induction on H. As an example, we prove Point (1) for $H = \mathbf{T}(A \to B)$. Let us assume that $p \preceq^+ \mathbf{T}(A \to B)$ and $\underline{K}, \alpha \rhd \mathbf{T}(A \to B)$. Let $\beta \in P$ such that $\alpha \leq \beta$ and $\underline{K}_p^+, \beta \rhd \mathbf{T}A$. To prove $\underline{K}_p^+, \alpha \rhd \mathbf{T}(A \to B)$ we have to show that $\underline{K}_p^+, \beta \rhd \mathbf{T}B$. Since $p \preceq^+ \mathbf{F}A$ we have $\underline{K}, \beta \not\rhd \mathbf{F}A$, otherwise, by the induction hypothesis, $\underline{K}_p^+, \beta \rhd \mathbf{F}A$, in contradiction with the given assumption. Thus $\underline{K}, \beta \rhd \mathbf{T}A$ and, since $\underline{K}, \alpha \rhd \mathbf{T}(A \to B)$ and $\alpha \leq \beta$, it follows that $\underline{K}, \beta \rhd \mathbf{T}B$. Since $p \preceq^+ \mathbf{T}B$, by the induction hypothesis we get $\underline{K}_p^+, \beta \rhd \mathbf{T}B$. □

Now, let us consider the following rules:

$$\frac{\Delta}{\Delta[\top/p]}\ \mathbf{T}\text{-permanence} \qquad \text{provided that } p \preceq^+ \Delta$$

$$\frac{\Delta}{\Delta[\bot/p]}\ \mathbf{T}\neg\text{-permanence} \quad \text{provided that } p \preceq^- \Delta.$$

Intuitively, these rules state that, if $p \preceq^+ \Delta$ (resp. $p \preceq^- \Delta$), then we can consistently add $\mathbf{T}p$ (resp. $\mathbf{T}\neg p$) to Δ and replace every occurrence of p in Δ with $\top$ (resp. $\bot$). From the previous lemma, we have the following.

THEOREM 4.2. *The rules* $\mathbf{T}$-permanence *and* $\mathbf{T}\neg$-permanence *are invertible.*

PROOF. Let us consider the case of the rule $\mathbf{T}$-permanence. We have to show that, if $p \preceq^+ \Delta$, then Δ is realizable iff $\Delta[\top/p]$ is realizable. Let us assume that Δ is realizable. Then, there exists a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ and $\alpha \in P$ such that $\underline{K}, \alpha \rhd \Delta$. Since $p \preceq^+ \Delta$, by Point (1) of Lemma 4.1, $\underline{K}_p^+, \alpha \rhd \Delta$ and, by definition of its forcing relation, $\underline{K}_p^+, \alpha \rhd \mathbf{T}p$. By Lemma 3.1(i), we get $\underline{K}_p^+, \alpha \rhd \Delta[\top/p]$, hence $\Delta[\top/p]$ is realizable. Conversely, let us suppose that $\Delta[\top/p]$ is realizable and let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model and $\alpha \in P$ such that $\underline{K}, \alpha \rhd \Delta[\top/p]$. Since p does not occur in $\Delta[\top/p]$, it holds that $p \preceq^+ \Delta[\top/p]$. By Point (1) of Lemma 4.1, $\underline{K}_p^+, \alpha \rhd \Delta[\top/p]$. Since $\underline{K}_p^+, \alpha \rhd \mathbf{T}p$, by Lemma 3.1(i) $\underline{K}_p^+, \alpha \rhd \Delta$, hence Δ is realizable. The case of the rule $\mathbf{T}\neg$-permanence is similar. □

Example 4.3. We show how the rule **T**-permanence can improve proof-search. Let

$$A = ((p \to q) \wedge ((\neg\neg r \to s) \to t) \wedge ((\neg\neg s \to t) \to p)) \to q.$$

The formula A is classically valid but not intuitionistically valid[1]. To decide A, we have to search for a proof of $\mathbf{F}A$. Since $r \preceq^+ \mathbf{F}A$, we can apply the rule **T**-permanence to get the set

$$\Delta_1 = \{\mathbf{F}(((p \to q) \wedge ((\neg\neg\top \to s) \to t) \wedge ((\neg\neg s \to t) \to p)) \to q)\}$$

and, simplifying $\neg\neg\top \to s$ to s with the rules of Figure 2, we get:

$$\Delta_2 = \{\mathbf{F}(((p \to q) \wedge (s \to t) \wedge ((\neg\neg s \to t) \to p)) \to q)\}.$$

Now, we proceed applying the rules $\mathbf{F} \to$ and $\mathbf{T}\wedge$ of $\mathbf{T}_{\mathrm{Int}}$ and we get:

$$\Delta_3 = \{\mathbf{T}(p \to q),\ \mathbf{T}(s \to t),\ \mathbf{T}((\neg\neg s \to t) \to p),\ \mathbf{F}q\}.$$

The only rule applicable to Δ_3 is the branching rule $\mathbf{T} \to\to$ and we obtain the nodes

$$\Delta_4 = \{\mathbf{T}(p \to q),\ \mathbf{T}(s \to t),\ \mathbf{T}\neg\neg s,\ \mathbf{F}a,\ \mathbf{T}(a \to p),\ \mathbf{T}(t \to a)\}$$
$$\Delta_5 = \{\mathbf{T}(p \to q),\ \mathbf{T}(s \to t),\ \mathbf{T}p,\ \mathbf{F}q\},$$

where a is a new propositional variable. Applying rules MP and $contr_1$ to Δ_5 we get a contradictory set. As for Δ_4, we have that $q \preceq^+ \Delta_4$, hence, applying **T**-permanence we get:

$$\Delta_6 = \{\mathbf{T}(p \to \top),\ \mathbf{T}(s \to t),\ \mathbf{T}\neg\neg s,\ \mathbf{F}a,\ \mathbf{T}(a \to p),\ \mathbf{T}(t \to a)\}.$$

Simplifying we obtain

$$\Delta_7 = \{\mathbf{T}\top,\ \mathbf{T}(s \to t),\ \mathbf{T}\neg\neg s,\ \mathbf{F}a,\ \mathbf{T}(a \to p),\ \mathbf{T}(t \to a)\}.$$

Now, $p \preceq^+ \Delta_7$ hence, by **T**-permanence and simplification, $\mathbf{T}(a \to p)$ reduces to $\mathbf{T}\top$ and we get:

$$\Delta_8 = \{\mathbf{T}\top,\ \mathbf{T}(s \to t),\ \mathbf{T}\neg\neg s,\ \mathbf{F}a,\ \mathbf{T}(t \to a)\}.$$

Now, we can only apply the $\mathbf{T}\neg\neg$ rule and we obtain the set

$$\Delta_9 = \{\mathbf{T}\top,\ \mathbf{T}(s \to t),\ \mathbf{T}s,\ \mathbf{T}(t \to a)\}.$$

Applying MP twice we get the set

$$\Delta_{10} = \{\mathbf{T}\top,\ \mathbf{T}t,\ \mathbf{T}s,\ \mathbf{T}a\},$$

which is clearly not contradictory and cannot be treated by any rule of the calculus. Since in our derivation there is no backtrack point, we conclude that $\mathbf{F}A$ is not provable.

If we disregard the rule **T**-permanence, that is we only apply then rules of $\mathbf{T}_{\mathrm{Int}}$, we have to begin the proof of $\mathbf{F}A$ by applying the rules $\mathbf{F} \to$ and $\mathbf{T}\wedge$ obtaining

$$\{\mathbf{T}(p \to q),\ \mathbf{T}((\neg\neg r \to s) \to t),\ \mathbf{T}((\neg\neg s \to t) \to p),\ \mathbf{F}q\}.$$

[1] A is the formula SYJ211+1.001 of ILTP Library [Raths et al. 2007].

At this point we have a backtrack point since the rule $\mathbf{T} \to\to$ can be applied to $\mathbf{T}((\neg\neg r \to s) \to t)$ or to $\mathbf{T}((\neg\neg s \to t) \to p)$.

5. THE RULE F-PERMANENCE

In this section we discuss the conditions under which it is correct to apply the partial substitution of a propositional variable p with $\bot$ also in the case the formula $\mathbf{F}p$ is not explicitly at hand.

Given a propositional variable p and a signed formula H, the relation $p \preceq_w^- H$ (p weakly negatively occurs in H) is defined by induction on the structure of H:

— $p \preceq_w^- \mathcal{S}\top$ and $p \preceq_w^- \mathcal{S}\bot$
— $p \preceq_w^- \mathbf{F}A$ and $p \preceq_w^- \mathbf{T}\neg A$ for every A
— $p \preceq_w^- \mathbf{T}q$ if $q \neq p$
— $p \preceq_w^- \mathbf{T}(A \odot B)$ iff $p \preceq_w^- \mathbf{T}A$ and $p \preceq_w^- \mathbf{T}B$, where $\odot \in \{\wedge, \vee\}$
— $p \preceq_w^- \mathbf{T}(A \to B)$ iff $p \preceq_w^- \mathbf{T}B$.

We remark that $p \preceq^- H$ implies $p \preceq_w^- H$, but the converse does not hold. Given a set Δ of signed formulas, we say that $p \preceq_w^- \Delta$ iff, for every $H \in \Delta$, $p \preceq_w^- H$.

Now, let us consider the following construction over Kripke models. Given $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ and a propositional variable p, let $\rho' \notin P$. By $\underline{K}_p^w$ we denote the structure $\langle P', \leq', \rho', \Vdash' \rangle$ such that:

$$P' = P \cup \{\rho'\} \qquad \leq' = \leq \cup \{(\rho', \alpha) \mid \alpha \in P'\}$$
$$\Vdash' = \Vdash \cup \{(\rho', q) \mid \rho \Vdash q \text{ and } q \neq p\}.$$

It is easy to check that $\underline{K}_p^w$ is a Kripke model. Note that $\underline{K}_p^w, \rho' \rhd \mathbf{F}p$ and, for every signed formula H and every $\alpha \in P$, $\underline{K}, \alpha \rhd H$ iff $\underline{K}_p^w, \alpha \rhd H$.

LEMMA 5.1. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model, let H be a signed formula, let p be a propositional variable such that $p \preceq_w^- H$ and let $\underline{K}_p^w = \langle P', \leq', \rho', \Vdash' \rangle$ the model defined previously. Then, $\underline{K}, \rho \rhd H$ implies $\underline{K}_p^w, \rho' \rhd H$.*

PROOF. Let us assume $\underline{K}, \rho \rhd H$. We prove $\underline{K}_p^w, \rho' \rhd H$ by induction on H. If $H = \mathbf{F}A$, with A any formula, then $\underline{K}_p^w, \rho \rhd \mathbf{F}A$, hence $\underline{K}_p^w, \rho' \rhd \mathbf{F}A$. The case $H = \mathbf{T}\neg A$ is similar. If $H = \mathbf{T}q$ then $q \neq p$ (indeed $p \preceq_w^- \mathbf{T}p$ does not hold) and hence, by definition of $\Vdash'$, $\underline{K}_p^w, \rho' \rhd H$. The cases $H = \mathbf{T}(A \wedge B)$ and $H = \mathbf{T}(A \vee B)$ easily follow by the induction hypothesis. Let $H = \mathbf{T}(A \to B)$ and let us suppose that $\underline{K}, \rho \rhd \mathbf{T}(A \to B)$. We immediately have $\underline{K}_p^w, \rho \rhd \mathbf{T}(A \to B)$; thus, to prove that $\underline{K}_p^w, \rho' \rhd \mathbf{T}(A \to B)$ we have only to show that $\underline{K}_p^w, \rho' \rhd \mathbf{T}A$ implies $\underline{K}_p^w, \rho' \rhd \mathbf{T}B$. If $\underline{K}_p^w, \rho' \rhd \mathbf{T}A$, then $\underline{K}_p^w, \rho \rhd \mathbf{T}A$, and this implies $\underline{K}, \rho \rhd \mathbf{T}A$. Since $\underline{K}, \rho \rhd \mathbf{T}(A \to B)$, we get $\underline{K}, \rho \rhd \mathbf{T}B$. Since $p \preceq_w^- \mathbf{T}B$, by induction hypothesis we conclude $\underline{K}_p^w, \rho' \rhd \mathbf{T}B$. □

Now, let us consider the rule:

$$\frac{\Delta}{\Delta\{\bot/p\}}\ \mathbf{F}\text{-permanence} \qquad \text{provided that } p \preceq_w^- \Delta.$$

Along the lines of the proof of Theorem 4.2, from Lemma 5.1 we get the following.

THEOREM 5.2. *The rule* **F**-permanence *is invertible.*

Example 5.3. As an application of this rule, let us consider the set

$$\Delta_1 \;=\; \{\,\mathbf{T}(p \vee q),\ \mathbf{F}(q \wedge r),\ \mathbf{F}(p \wedge r),\ \mathbf{F}(r \to q)\,\}$$

First of all, we notice that the propositional variables p, q and r do not occur in Δ_1 with constant sign, that is $x \not\preceq^+ \Delta_1$ and $x \not\preceq^- \Delta_1$ for every $x \in \{p, q, r\}$, thus the replacement rules discussed Sections 4 cannot be applied to Δ_1. On the other hand $r \preceq_w^- \Delta_1$, hence we can apply **F**-permanence and we get the set

$$\Delta_2 \;=\; \{\,\mathbf{T}(p \vee q),\ \mathbf{F}(q \wedge \bot),\ \mathbf{F}(p \wedge \bot),\ \mathbf{F}(r \to q)\,\}.$$

Applying the Boolean simplification rules to Δ_2 we get:

$$\Delta_3 \;=\; \{\,\mathbf{T}(p \vee q),\ \mathbf{F}\bot,\ \mathbf{F}(r \to q)\,\}.$$

We remark that the negative occurrence of p has disappeared, and now $p \preceq^+ \Delta_3$. We can apply the rule **T**-permanence and the Boolean simplification rules, obtaining the set

$$\Delta_4 \;=\; \{\,\mathbf{T}\top,\ \mathbf{F}\bot,\ \mathbf{F}(r \to q)\,\},$$

which does not have a closed proof-table. Since the derivation does not contain any backtrack point we conclude that Δ_1 is not provable.

6. PERMANENCE RULES UNDER CONTEXTS

Permanence rules described in Sections 4 and 5 act on propositional variables with constant polarity in a set Δ of signed formulas. In this section we describe a generalization of these rules allowing us to apply replacement on propositional variables which occur with constant polarity in opportune subformulas of formulas in Δ.

A *context* is a particular formula containing only one occurrence of a dedicated propositional variable (not used elsewhere in sets of formulas) and denoted by $*$. Formally, we consider contexts $\Theta[\![*]\!]$ defined as follows:

$$\Theta[\![*]\!] \;::=\; * \;\mid\; C \odot \Theta[\![*]\!] \;\mid\; \Theta[\![*]\!] \odot C \;\mid\; C \to \Theta[\![*]\!],$$

where $\odot \in \{\wedge, \vee\}$ and C is any formula. Given a context $\Theta[\![*]\!]$ and a formula A, by $\Theta[\![A]\!]$ we denote the formula obtained by replacing $*$ with A in $\Theta[\![*]\!]$. In the following we always consider contexts under the sign **F** so that $*$ has a negative polarity. The *set of antecedents* $\mathrm{Ant}(\Theta[\![*]\!])$ *of a context* $\Theta[\![*]\!]$ is inductively defined as follows.

$$\begin{aligned}
&\mathrm{Ant}(*) \;=\; \emptyset \\
&\mathrm{Ant}(C \odot \Theta[\![*]\!]) \;=\; \mathrm{Ant}(\Theta[\![*]\!] \odot C) \;=\; \mathrm{Ant}(\Theta[\![*]\!]) \qquad \odot \in \{\wedge, \vee\} \\
&\mathrm{Ant}(C \to \Theta[\![*]\!]) \;=\; \{C\} \cup \mathrm{Ant}(\Theta[\![*]\!]).
\end{aligned}$$

Now, let us consider the following rules:

$$\frac{\Delta, \mathbf{F}\Theta[\![A \to B]\!]}{\Delta, \mathbf{F}\Theta[\![\, A[\top/p] \to B[\top/p] \,]\!]}\ \mathbf{T}\text{-cperm}$$

where $p \preceq^+ \mathbf{F}(A \to B)$, $p \preceq^+ \Delta_{\mathbf{T}}$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^+ \mathbf{T}Z$.

$$\frac{\Delta, \mathbf{F}\Theta[\![A \to B]\!]}{\Delta, \mathbf{F}\Theta[\![\, A[\bot/p] \to B[\bot/p] \,]\!]}\ \mathbf{T}\neg\text{-cperm}$$

where $p \preceq^- \mathbf{F}(A \to B)$, $p \preceq^- \Delta_{\mathbf{T}}$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^- \mathbf{T}Z$.

$$\frac{\Delta, \mathbf{F}\Theta[\![\neg A]\!]}{\Delta, \mathbf{F}\Theta[\![\, \neg A[\top/p] \,]\!]}\ \mathbf{T}\text{-cperm}'$$

where $p \preceq^+ \mathbf{F}(\neg A)$, $p \preceq^+ \Delta_{\mathbf{T}}$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^+ \mathbf{T}Z$.

$$\frac{\Delta, \mathbf{F}\Theta[\![\neg A]\!]}{\Delta, \mathbf{F}\Theta[\![\, \neg A[\bot/p] \,]\!]}\ \mathbf{T}\neg\text{-cperm}'$$

where $p \preceq^- \mathbf{F}(\neg A)$, $p \preceq^- \Delta_{\mathbf{T}}$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^- \mathbf{T}Z$.

$$\frac{\Delta, \mathbf{F}\Theta[\![B]\!]}{\Delta, \mathbf{F}\Theta[\![\, B\{\bot/p\} \,]\!]}\ \mathbf{F}\text{-cperm}$$

where $p \preceq_w^- \Delta_{\mathbf{T}}$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq_w^- \mathbf{T}Z$.

We remark that the rules $\mathbf{T}$-cperm$'$ and $\mathbf{T}\neg$-cperm$'$ can be seen as a particular case of $\mathbf{T}$-cperm and $\mathbf{T}\neg$-cperm respectively, via the intuitionistic equivalence $\neg A \equiv A \to \bot$. In Section 6.1 we prove that these rules are invertible. The side condition on $\mathrm{Ant}(\Theta[\![*]\!])$ is essential to guarantee the invertibility. For instance, let us consider the context $\Theta[\![*]\!] = \neg p \to *$. If we drop out the condition on $\mathrm{Ant}(\Theta[\![*]\!])$, we could apply $\mathbf{T}$-cperm to $\mathbf{F}\Theta[\![p \to q]\!] = \mathbf{F}(\neg p \to (p \to q))$ to replace p with $\top$ in $p \to q$ (indeed, $p \preceq^+ \mathbf{F}(p \to q)$), and we would obtain $\mathbf{F}\Theta[\![\top \to q]\!] = \mathbf{F}(\neg p \to (\top \to q))$ which is realizable, whereas $\mathbf{F}\Theta[\![p \to q]\!] = \mathbf{F}(\neg p \to (p \to q))$ is not.

We give some examples of application.

Example 6.1. Let us consider the signed formula

$$H_1 = \mathbf{F}(\ \neg q \vee ((q \to \neg p) \to ((\neg q \wedge p) \vee (p \to q)))\ ).$$

First of all we notice that, for $x \in \{p, q\}$, neither $x \preceq^+ H_1$ nor $x \preceq^- H_1$ holds; moreover, the partial substitution of x with $\bot$ in H_1 has no effect, hence the application of the rule $\mathbf{F}$-permanence is useless. Now, let us consider the context $\Theta[\![*]\!] = \neg q \vee ((q \to \neg p) \to *)$. We have that $H_1 = \mathbf{F}\Theta[\![(\neg q \wedge p) \vee (p \to q)]\!]$. Since $\mathrm{Ant}(\Theta[\![*]\!]) = \{q \to \neg p\}$ and $p \preceq_w^- \mathbf{T}(q \to \neg p)$, we can apply the rule $\mathbf{F}$-cperm to H_1 and we get

$$\mathbf{F}\Theta[\![(\neg q \wedge \bot) \vee (p \to q)]\!],$$

which simplifies to

$$H_2 \ = \ \mathbf{F}\Theta[\![p \to q]\!] \ = \ \mathbf{F}(\ \neg q \vee ((q \to \neg p) \to (p \to q))\ ).$$

Now, let us consider the context $\Theta'[\![*]\!] = \neg q \vee *$. Then

$$H_2 \ = \ \mathbf{F}\Theta'[\![(q \to \neg p) \to (p \to q)]\!].$$

Since $\mathrm{Ant}(\Theta'[\![*]\!]) = \emptyset$ and $q \preceq^- \mathbf{F}((q \to \neg p) \to (p \to q))$, we can apply the rule $\mathbf{T}\neg$-cperm obtaining

$$\mathbf{F}\Theta'[\![(\bot \to \neg p) \to (p \to \bot)]\!],$$

which simplifies to

$$\mathbf{F}\Theta'[\![\neg p]\!] \;=\; \mathbf{F}(\neg q \vee \neg p),$$

which is not provable. Since all the applied rules are invertible, we conclude that H_1 is not provable.

Example 6.2. As another example, let us consider the signed formula

$$K_1 \;=\; \mathbf{F}(\; a \to (\,(p \vee q) \to (q \wedge r) \vee (p \wedge r) \vee (r \to q)\,) \vee p \;\;)$$

and let us consider the context $\Theta[\![*]\!] = a \to ((p \vee q) \to *) \vee p$. We have that

$$K_1 \;=\; \mathbf{F}\Theta[\![(q \wedge r) \vee (p \wedge r) \vee (r \to q)]\!].$$

Since $\mathrm{Ant}(\Theta[\![*]\!]) = \{a, p \vee q\}$, $r \preceq_w^- \mathbf{T}a$ and $r \preceq_w^- \mathbf{T}(p \vee q)$ we can apply the rule $\mathbf{F}$-cperm and we get:

$$\mathbf{F}\Theta[\![(q \wedge \bot) \vee (p \wedge \bot) \vee (r \to q)]\!].$$

This signed formula simplifies to

$$K_2 \;=\; \mathbf{F}\Theta[\![r \to q]\!] \;=\; \mathbf{F}(\; a \to (\,(p \vee q) \to (r \to q)\,) \vee p\;).$$

Let us consider the context $\Theta'[\![*]\!] = a \to * \vee p$. Then

$$K_2 \;=\; \mathbf{F}\Theta'[\![(p \vee q) \to (r \to q)]\!].$$

Since $\mathrm{Ant}(\Theta'[\![*]\!]) = \{a\}$, $p \preceq^+ \mathbf{T}a$ and $p \preceq^+ \mathbf{F}((p \vee q) \to (r \to q))$, we can apply the rule $\mathbf{T}$-cperm and we get

$$\mathbf{F}\Theta'[\![(\top \vee q) \to (r \to q)]\!],$$

which simplifies to

$$\mathbf{F}\Theta'[\![r \to q]\!] \;=\; \mathbf{F}(\; a \to (r \to q) \vee p\;),$$

which is not provable. Since this proof only consists of invertible rules, we deduce that K_1 is unprovable. We remark that proof-search for K_1 in $\mathbf{T}_{\mathrm{Int}}$ is more expensive since one has to apply several noninvertible rules.

To support the usefulness of the simplification rules described in this section, we point out that the $\mathbf{T}$-cperm and $\mathbf{F}$-cperm rules reduce backtracking required to decide the family of de Bruijn formulas (the family SYJ207+1 of ILTP library [Raths et al. 2007]).

6.1. Proof of Invertibility

The proof of invertibility of the rules $\mathbf{T}$-cperm, $\mathbf{T}\neg$-cperm, $\mathbf{T}$-cperm$'$, $\mathbf{T}\neg$-cperm$'$ and $\mathbf{F}$-cperm is not trivial and requires some machinery. We start by stating some properties about contexts which can be easily proved by induction on the structure of $\Theta[\![*]\!]$.

LEMMA 6.3. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ be a Kripke model, $\alpha \in P$, and let $\Theta[\![*]\!]$ be a context. If $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![A \wedge C \to B \vee D]\!]$, then $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![A \to B]\!]$.*

The following lemma extends lemmas 3.1 and 3.3 to contexts.

LEMMA 6.4. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ be a Kripke model and $\alpha \in P$. Then:*

(i) $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, A \wedge C \to B \,]\!]$ *iff* $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, A[\top/C] \wedge C \to B[\top/C] \,]\!]$;
(ii) $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, A \wedge \neg C \to B \,]\!]$ *iff* $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, A[\bot/C] \wedge \neg C \to B[\bot/C] \,]\!]$;
(iii) $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, B \vee C \,]\!]$ *iff* $\underline{K}, \alpha \rhd \mathbf{F}\Theta[\![\, B\{\bot/C\} \vee C \,]\!]$.

Let $\underline{K}_1 = \langle P_1, \leq_1, \rho_1, \Vdash_1 \rangle$ and $\underline{K}_2 = \langle P_2, \leq_2, \rho_2, \Vdash_2 \rangle$ be two Kripke models; $\underline{K}_1$ and $\underline{K}_2$ are *disjoint* if $P_1 \cap P_2 = \emptyset$. Let $\underline{K}_1$ and $\underline{K}_2$ be disjoint and let $\rho \notin P_1 \cup P_2$. By $\underline{K}_1 \biguplus_\rho \underline{K}_2$ we denote the Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ such that:

$$P = P_1 \cup P_2 \cup \{\rho\} \qquad \leq = \leq_1 \cup \leq_2 \cup \{(\rho, \alpha) \mid \alpha \in P\}$$
$$\Vdash = \Vdash_1 \cup \Vdash_2 \cup \{(\rho, p) \mid \rho_1 \Vdash_1 p \text{ and } \rho_2 \Vdash_2 p\}.$$

(see, for instance, the models in Figures 3 and 4). One can easily check that, for every signed formula H, every $i \in \{1, 2\}$ and every $\alpha \in P_i$, it holds that $\underline{K}, \alpha \rhd H$ iff $\underline{K}_i, \alpha \rhd H$.

We show that in the model $\underline{K}_1 \biguplus_\rho \underline{K}_2$ we have a sort of downward preservation of **T**-signed formulas realizability: under suitable conditions, the realizability of a signed formula $\mathbf{T}D$ in the root of $\underline{K}_1$ implies the realizability of $\mathbf{T}D$ in ρ.

LEMMA 6.5. *Let $\underline{K}_1 = \langle P_1, \leq_1, \rho_1, \Vdash_1 \rangle$ and $\underline{K}_2 = \langle P_2, \leq_2, \rho_2, \Vdash_2 \rangle$ be two disjoint Kripke models and let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be the model $\underline{K}_1 \biguplus_\rho \underline{K}_2$. Let p be a propositional variable, let $\preceq$ be one of the relations $\preceq^+$, $\preceq^-$ or $\preceq^-_w$ and assume that, for every formula D such that $p \preceq \mathbf{T}D$, $\underline{K}, \rho_1 \rhd \mathbf{T}D$ implies $\underline{K}, \rho_2 \rhd \mathbf{T}D$. Then, for every formula D such that $p \preceq \mathbf{T}D$, $\underline{K}, \rho_1 \rhd \mathbf{T}D$ implies $\underline{K}, \rho \rhd \mathbf{T}D$.*

PROOF. Let us assume $p \preceq \mathbf{T}D$ and $\underline{K}, \rho_1 \rhd \mathbf{T}D$. By the hypothesis of the lemma, we get $\underline{K}, \rho_2 \rhd \mathbf{T}D$. We prove $\underline{K}, \rho \rhd \mathbf{T}D$ by induction on D.
Let $D = q$ be a propositional variable. Since $\underline{K}, \rho_1 \rhd \mathbf{T}q$ and $\underline{K}, \rho_2 \rhd \mathbf{T}q$, by definition of $\Vdash$ we get $\underline{K}, \rho \rhd \mathbf{T}q$.
The cases $D = X \wedge Y$ and $D = X \vee Y$ follow by the fact that $p \preceq \mathbf{T}X$ and $p \preceq \mathbf{T}Y$ and by the induction hypothesis.
Let $D = X \to Y$ and let $\alpha \in P$ such that $\underline{K}, \alpha \rhd \mathbf{T}X$; we have to show that $\underline{K}, \alpha \rhd \mathbf{T}Y$. If $\alpha \in P_1 \cup P_2$, by the fact that $\underline{K}, \rho_1 \rhd \mathbf{T}(X \to Y)$ and $\underline{K}, \rho_2 \rhd \mathbf{T}(X \to Y)$, we get $\underline{K}, \alpha \rhd \mathbf{T}Y$. It remains to consider the case $\alpha = \rho$. Since $\rho \leq \rho_1$, we have $\underline{K}, \rho_1 \rhd \mathbf{T}X$, which implies $\underline{K}, \rho_1 \rhd \mathbf{T}Y$. Since $p \preceq \mathbf{T}(X \to Y)$, it holds that $p \preceq \mathbf{T}Y$; by the induction hypothesis we conclude $\underline{K}, \rho \rhd \mathbf{T}Y$.
Let $D = \neg X$. Since $\underline{K}, \rho_1 \rhd \mathbf{T}\neg X$ and $\underline{K}, \rho_2 \rhd \mathbf{T}\neg X$, we get $\underline{K}, \rho \rhd \mathbf{T}\neg X$. □

To prove the invertibility of the rule **T**-cperm, the crucial point is to build a model $\underline{K}^*$ realizing $\mathbf{F}\Theta[\![A \wedge p \to B]\!]$ given a model $\underline{K}$ for $\mathbf{F}\Theta[\![A \to B]\!]$. The next lemma shows how to build $\underline{K}^*$.

LEMMA 6.6. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model, p a propositional variable, $\Theta[\![A \to B]\!]$ a formula such that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A \to B]\!]$, $p \preceq^+ \mathbf{F}(A \to B)$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^+ \mathbf{T}Z$. There exists a model $\underline{K}^* = \langle P^*, \leq^*, \rho^*, \Vdash^* \rangle$ such that:*

(i) $\underline{K}^*, \rho^* \rhd \mathbf{F}\Theta[\![A \wedge p \to B]\!]$;
(ii) For every formula D, if $\underline{K}, \rho \rhd \mathbf{F}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{F}D$;
(iii) For every formula D such that $p \preceq^+ \mathbf{T}D$, if $\underline{K}, \rho \rhd \mathbf{T}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{T}D$.

PROOF. By induction on the structure of $\Theta[\![*]\!]$.
Let $\Theta[\![A \to B]\!] = A \to B$. There is $\alpha \in P$ such that $\underline{K}, \alpha \rhd \mathbf{T}A$ and $\underline{K}, \alpha \rhd \mathbf{F}B$. By Lemma 4.1, there exists a Kripke model $\underline{K}' = \langle P', \leq', \alpha', \Vdash' \rangle$ such that:

(P1) $\underline{K}', \alpha' \rhd \mathbf{T}p$;
(P2) For every signed formula H such that $p \preceq^+ H$, if $\underline{K}, \alpha \rhd H$ then $\underline{K}', \alpha' \rhd H$.

Since $p \preceq^+ \mathbf{T}A$ and $p \preceq^+ \mathbf{F}B$, by (P2) we get $\underline{K}', \alpha' \rhd \mathbf{T}A$ and $\underline{K}', \alpha' \rhd \mathbf{F}B$. Without loss of generality we can assume that $\underline{K}$ and $\underline{K}'$ are disjoint. Let $\rho^* \notin P \cup P'$ and let $\underline{K}^* = \langle P^*, \leq^*, \rho^*, \Vdash^* \rangle$ be the model $\underline{K} \biguplus_{\rho^*} \underline{K}'$ (see Figure 3). Since $\rho^* \leq^* \alpha'$, we get $\underline{K}^*, \rho^* \rhd \mathbf{F}(A \wedge p \to B)$ and Point (i) is proved. Point (ii) follows by the fact that

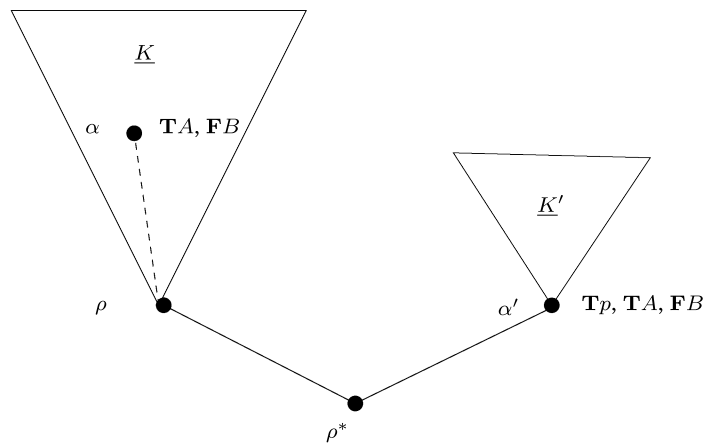

Fig. 3. The model $\underline{K}^* = \underline{K} \uplus_{\rho^*} \underline{K}'$ in the case $\Theta[\![A \to B]\!] = A \to B$.

$\underline{K}, \rho \rhd \mathbf{F}D$ implies $\underline{K}^*, \rho \rhd \mathbf{F}D$ and $\rho^* \leq^* \rho$. To prove Point (iii), we observe that, for every D such that $p \preceq^+ \mathbf{T}D$, $\underline{K}, \rho \rhd \mathbf{T}D$ implies $\underline{K}', \alpha' \rhd \mathbf{T}D$ (this follows by the fact that $\rho \leq \alpha$ and (P2)). Now, we can apply Lemma 6.5, and Point (iii) is proved.

Let $\Theta[\![A \to B]\!] = C \wedge \Theta'[\![A \to B]\!]$. Since $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A \to B]\!]$, either $\underline{K}, \rho \rhd \mathbf{F}C$ or $\underline{K}, \rho \rhd \mathbf{F}\Theta'[\![A \to B]\!]$. In the former case, the model $\underline{K}^* = \underline{K}$ meets points (i)–(iii). In the latter case, by induction hypothesis there exists a model $\underline{K}^*$ such that $\underline{K}^*, \rho^* \rhd \mathbf{F}\Theta'[\![A \wedge p \to B]\!]$ and points (ii), (iii) are satisfied. Hence, $\underline{K}^*, \rho^* \rhd \mathbf{F}(C \wedge \Theta'[\![A \wedge p \to B]\!])$, and the assertion is proved.
Let $\Theta[\![A \to B]\!] = C \vee \Theta'[\![A \to B]\!]$. Since $\underline{K}, \rho \rhd \mathbf{F}\Theta'[\![A \to B]\!]$, by induction hypothesis there is a model $\underline{K}^*$ such that $\underline{K}^*, \rho^* \rhd \mathbf{F}\Theta'[\![A \wedge p \to B]\!]$ and points (ii), (iii) are satisfied. Since $\underline{K}, \rho \rhd \mathbf{F}C$, by (ii), we have $\underline{K}^*, \rho^* \rhd \mathbf{F}C$, hence $\underline{K}^*, \rho^* \rhd \mathbf{F}(C \vee \Theta'[\![A \wedge p \to B]\!])$.
Finally, let $\Theta[\![A \to B]\!] = C \to \Theta'[\![A \to B]\!]$ and let $\beta \in P$ such that $\underline{K}, \beta \rhd \mathbf{T}C$ and $\underline{K}, \beta \rhd \mathbf{F}\Theta'[\![A \to B]\!]$. By the induction hypothesis, there exists a model $\underline{K}' = \langle P', \leq', \beta', \Vdash' \rangle$ such that:

(P3) $\underline{K}', \beta' \rhd \mathbf{F}\Theta'[\![A \wedge p \to B]\!]$;
(P4) For every D such that $p \preceq^+ \mathbf{T}D$, if $\underline{K}, \beta \rhd \mathbf{T}D$ then $\underline{K}', \beta' \rhd \mathbf{T}D$.

Since $C \in \mathrm{Ant}(\Theta[\![*]\!])$, by the hypothesis of the lemma it holds that $p \preceq^+ \mathbf{T}C$; hence, by (P4), we get $\underline{K}', \beta' \rhd \mathbf{T}C$. We can reason as in the base case taking the model $\underline{K} \uplus_{\rho^*} \underline{K}'$ as $\underline{K}^*$, where $\rho^* \notin P \cup P'$ (see Figure 4). Indeed, since $\rho^* \leq^* \beta'$, we get $\underline{K}^*, \rho^* \rhd \mathbf{F}(C \to \Theta'[\![A \wedge p \to B]\!])$. Point (ii) is immediate. Point (iii) follows by (P4) and Lemma 6.5. □

Lemma 6.6 can be restated for the relations $\preceq^-$ and $\preceq^-_w$ as follows.

LEMMA 6.7. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash \rangle$ be a Kripke model, p a propositional variable, $\Theta[\![A \to B]\!]$ a formula such that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A \to B]\!]$, $p \preceq^- \mathbf{F}(A \to B)$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^- \mathbf{T}Z$. There exists a model $\underline{K}^* = \langle P^*, \leq^*, \rho^*, \Vdash^* \rangle$ such that:*

(i) $\underline{K}^*, \rho^* \rhd \mathbf{F}\Theta[\![A \wedge \neg p \to B]\!]$;
(ii) For every formula D, if $\underline{K}, \rho \rhd \mathbf{F}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{F}D$;
(iii) For every formula D such that $p \preceq^- \mathbf{T}D$, if $\underline{K}, \rho \rhd \mathbf{T}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{T}D$.

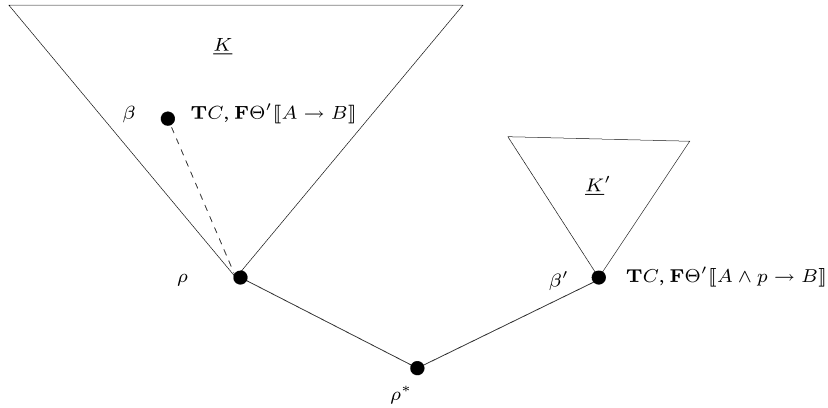

Fig. 4. The model $\underline{K}^* = \underline{K} \uplus_{\rho^*} \underline{K}'$ in the case $\Theta[\![A \to B]\!] = C \to \Theta'[\![A \to B]\!]$.

LEMMA 6.8. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ be a Kripke model, p a propositional variable, $\Theta[\![B]\!]$ a formula such that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![B]\!]$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq_w^- \mathbf{T}Z$. There exists a model $\underline{K}^* = \langle P^*, \leq^*, \rho^*, \Vdash^*\rangle$ such that:*

(i) $\underline{K}^*, \rho^* \rhd \mathbf{F}\Theta[\![B \vee p]\!]$;
(ii) For every formula D, if $\underline{K}, \rho \rhd \mathbf{F}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{F}D$;
(iii) For every formula D such that $p \preceq_w^- \mathbf{T}D$, if $\underline{K}, \rho \rhd \mathbf{T}D$ then $\underline{K}^*, \rho^* \rhd \mathbf{T}D$.

Finally, we can prove the main theorem of this section.

THEOREM 6.9. *The rules* **T**-cperm*,* $\neg$**T**-cperm*,* **T**-cperm$'$*,* **T**$\neg$-cperm$'$ *and* **F**-cperm *are invertible.*

PROOF. Let us consider the rule **T**-cperm. Let Δ be a set of signed formulas such that $p \preceq^+ \Delta_{\mathbf{T}}$ and $\mathbf{F}\Theta[\![A \to B]\!]$ a formula such that $p \preceq^+ \mathbf{F}(A \to B)$ and, for every $Z \in \mathrm{Ant}(\Theta[\![*]\!])$, $p \preceq^+ \mathbf{T}Z$. We have to prove that $\Delta \cup \{\mathbf{F}\Theta[\![A \to B]\!]\}$ is realizable iff $\Delta \cup \{\mathbf{F}\Theta[\![A[\top/p] \to B[\top/p]]\!]\}$ is. If the set $\Delta \cup \{\mathbf{F}\Theta[\![A \to B]\!]\}$ is realizable, by Lemma 6.6 there exists a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ such that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A \wedge p \to B]\!]$ and $\underline{K}, \rho \rhd \Delta$. By Lemmas 6.4(i) and 6.3, it follows that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A[\top/p] \to B[\top/p]]\!]$. This means that $\Delta \cup \{\mathbf{F}\Theta[\![A[\top/p] \to B[\top/p]]\!]\}$ is realizable, and this proves the soundness of the rule **T**-cperm.

Conversely, let us assume that the set $\Delta \cup \{\mathbf{F}\Theta[\![A[\top/p] \to B[\top/p]]\!]\}$ is realizable. Since p does not occur in $\mathbf{F}(A[\top/p] \to B[\top/p])$, it holds that $p \preceq^+ \mathbf{F}(A[\top/p] \to B[\top/p])$. By Lemma 6.6, there exists a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ such that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A[\top/p] \wedge p \to B[\top/p]]\!]$ and $\underline{K}, \rho \rhd \Delta$. By lemmas 6.4(i) and 6.3, it follows that $\underline{K}, \rho \rhd \mathbf{F}\Theta[\![A \to B]\!]$, thus $\Delta \cup \{\mathbf{F}\Theta[\![A \to B]\!]\}$ is realizable.

Invertibility of **T**-cperm$'$ immediately follows from the invertibility of **T**-cperm. Indeed, $\mathbf{F}\Theta[\![\neg A]\!]$ is realizable iff $\mathbf{F}\Theta[\![A \to \bot]\!]$ is realizable. By the previous discussion $\mathbf{F}\Theta[\![A \to \bot]\!]$ is realizable iff $\mathbf{F}\Theta[\![A[\top/p] \to \bot]\!]$ is realizable, iff $\mathbf{F}\Theta[\![\neg A[\top/p]]\!]$ is realizable.

The other cases are similar. □

7. PRUNING OVER CONJUNCTIVE CONTEXTS

In this section we define a "strong" simplification technique we call *pruning*, which is based on the *conjunctive context* $\mathrm{cc}(H)$ of a signed formula H. By $\mathrm{cc}(H)$ we denote a set of signed formulas built from the subformulas of H with the following property: for every Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ and every $\alpha \in P$, $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd \mathrm{cc}(H)$.

Formally, the conjunctive context of H is defined as follows.

— $\mathrm{cc}(\mathcal{S}p) = \{\mathcal{S}p\}$, with p a propositional variable
— $\mathrm{cc}(\mathbf{T}\neg p) = \{\mathbf{T}\neg p\}$, with p propositional variable
— $\mathrm{cc}(\mathbf{T}(A \wedge B)) = \mathrm{cc}(\mathbf{T}A) \cup \mathrm{cc}(\mathbf{T}B)$
— $\mathrm{cc}(\mathbf{F}(A \wedge B)) = \{\mathbf{F}(A \wedge B)\} \cup (\mathrm{cc}(\mathbf{F}A) \cap \mathrm{cc}(\mathbf{F}B))$
— $\mathrm{cc}(\mathbf{T}(A \vee B)) = \{\mathbf{T}(A \vee B)\} \cup (\mathrm{cc}(\mathbf{T}A) \cap \mathrm{cc}(\mathbf{T}B))$
— $\mathrm{cc}(\mathbf{F}(A \vee B)) = \mathrm{cc}(\mathbf{F}A) \cup \mathrm{cc}(\mathbf{F}B)$
— $\mathrm{cc}(\mathbf{T}\neg A) = \{\mathbf{T}\neg A\}$, if A is of the kind $(B \wedge C)$ or $\neg B$
— $\mathrm{cc}(\mathbf{T}\neg(A \vee B)) = \mathrm{cc}(\mathbf{T}\neg A) \cup \mathrm{cc}(\mathbf{T}\neg B)$
— $\mathrm{cc}(\mathbf{T}\neg(A \to B)) = \{\mathbf{T}\neg(A \to B)\} \cup \mathrm{cc}(\mathbf{T}\neg B)$
— $\mathrm{cc}(\mathbf{F}\neg A) = \{\mathbf{F}\neg A\}$
— $\mathrm{cc}(\mathbf{T}(A \to B)) = \{\mathbf{T}(A \to B)\}$
— $\mathrm{cc}(\mathbf{F}(A \to B)) = \{\mathbf{F}(A \to B)\} \cup \mathrm{cc}(\mathbf{F}B)$.

It is easy to check that, given a formula A, the signed formulas in $\mathrm{cc}(\mathcal{S}A)$ have sign $\mathcal{S}$. Moreover, all the signed formulas in $\mathrm{cc}(\mathbf{T}\neg A)$ are of the kind $\mathbf{T}\neg A'$. It is easy to prove the following by induction on the structure of H.

LEMMA 7.1. *Let $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ be a Kripke model and $\alpha \in P$. Then, $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd \mathrm{cc}(H)$.*

Now, let us consider the `pruning` function, described in Figure 5, which calls the following functions.

— `simpl`. It takes as input a set of signed formulas Δ and returns the set of signed formulas obtained by applying the Boolean simplification rules of Figure 2 and the replacement rules Replace-**T**, Replace-**T**$\neg$ and Replace-**F** as long as possible. By the results given in Section 3, $\underline{K}, \alpha \rhd \Delta$ iff $\underline{K}, \alpha \rhd \mathtt{simpl}(\Delta)$. We remark that the permanence rules cannot be applied in `simpl` because it works on subformulas of the premise, while the applicability condition of permanence rules involve the whole premise.
— `extract`. It takes a nonempty set of signed formulas Δ as input and works as follows:

```
let Δ1 = {A | TA ∈ Δ}, Δ2 = {A | FA ∈ Δ}
if Δ1 = ∅ then return F(⋁ Δ2)
else if Δ2 = ∅ then return T(⋀ Δ1)
else return F(⋀ Δ1 → ⋁ Δ2)
```

The pruning algorithm satisfies the following properties.

LEMMA 7.2. *Given a signed formula H:*

(i) $\mathtt{pruning}(H)$ *has the same sign of H, moreover if $H = \mathbf{T}\neg A'$, then* $\mathtt{pruning}(H)$ *has the form $\mathbf{T}\neg A''$.*
(ii) Given a Kripke model $\underline{K} = \langle P, \leq, \rho, \Vdash\rangle$ and $\alpha \in P$, $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd \mathtt{pruning}(H)$.

PROOF. The proof goes by induction on the structure of H. We only treat some relevant cases. If $H = \mathcal{S}p$ or $H = \mathbf{T}\neg p$, then the assertions immediately follow.
If $H = \mathbf{T}(X \wedge Y)$ then, by induction hypothesis, $\mathtt{pruning}(\mathbf{T}X) = \mathbf{T}X'$ and $\mathtt{pruning}(\mathbf{T}Y) = \mathbf{T}Y'$. Since $\mathrm{cc}(\mathbf{T}X') \cup \mathrm{cc}(\mathbf{T}Y')$ only contains **T**-signed formulas and the simplification rules do not act on the sign, by definition of `extract`, the sign of the returned formula is **T**. This proves Point (i). Now, $\underline{K}, \alpha \rhd H$ iff $\underline{K}, \alpha \rhd \mathbf{T}X$ and $\underline{K}, \alpha \rhd \mathbf{T}Y$. By induction hypothesis this holds iff $\underline{K}, \alpha \rhd \mathbf{T}X'$ and $\underline{K}, \alpha \rhd \mathbf{T}Y'$ iff, by Lemma 7.1,

```
function pruning(H) {
  case H = Sp or H = S¬p: return H
  case H = T(X ∧ Y) or H = F(X ∨ Y):
    let S be the sign of H and SX' = pruning(SX) and SY' = pruning(SY)
    return extract( simpl( cc(SX') ∪ cc(SY') ) )
  case H = T(X ∨ Y) or H = F(X ∧ Y):
    let S be the sign of H and SX' = pruning(SX) and SY' = pruning(SY)
    if H = T(X ∨ Y) then K = T(X' ∨ Y') else K = F(X' ∧ Y')
    return extract( simpl( (cc(SX') ∩ cc(SY')) ∪ {K} ) )
  case H = F(X → Y):
    TX' = pruning(TX), FY' = pruning(FY),
    return extract( simpl( cc(TX') ∪ cc(FY') ) )
  case H = T(X → Y):
    FX' = pruning(FX), TY' = pruning(TY)
    return T(X' → Y')
  case H = F¬X:
    TX' = pruning(TX)
    return F¬X'
  case H = T¬(X ∨ Y):
    T¬X' = pruning(T¬X), T¬Y' = pruning(T¬Y)
    Δ = simpl( cc(T¬X') ∪ cc(T¬Y') )
    return T¬(⋁_{T¬Z∈Δ} Z)
  case H = T¬(X → Y):
    TX' = pruning(TX), T¬Y' = pruning(T¬Y)
    Δ = simpl( cc(TX') ∪ cc(T¬Y') )
    Δ_{T¬} = {T¬Z | T¬Z ∈ Δ}, Δ_T = Δ \ Δ_{T¬}
    return T¬(⋀_{TZ∈Δ_T} Z → ⋁_{T¬Z∈Δ_{T¬}} Z)
  case H = T¬(X ∧ Y):
    T¬X' = pruning(T¬X), T¬Y' = pruning(T¬Y)
    Δ = simpl( (cc(T¬X') ∩ cc(T¬Y')) ∪ {T¬(X' ∧ Y')} )
    return T¬(⋁_{T¬Z∈Δ} Z)
  case H = T¬¬X:
    TX' = pruning(TX)
    return T¬¬X'
  }
```

Fig. 5. The pruning function.

$\underline{K}, \alpha \rhd \mathrm{cc}(\mathbf{T}X') \cup \mathrm{cc}(\mathbf{T}Y')$. By the properties of simpl and by the fact that $\mathtt{simpl}(\mathrm{cc}(\mathbf{T}X') \cup \mathrm{cc}(\mathbf{T}Y'))$ only consists of $\mathbf{T}$-signed formulas, this holds iff $\underline{K}, \alpha \rhd \mathtt{extract}(\mathtt{simpl}(\mathrm{cc}(\mathbf{T}X') \cup \mathrm{cc}(\mathbf{T}Y')))$.

Let $H = \mathbf{T}\neg(X \wedge Y)$. Point (i) immediately follows by construction of the returned formula. As for Point (ii), let us suppose that $\underline{K}, \alpha \rhd \mathbf{T}\neg(X \wedge Y)$ and let ϕ be a final element of $\underline{K}$[2] such that $\alpha \leq \phi$. Then, $\underline{K}, \phi \rhd \mathbf{T}\neg X$ or $\underline{K}, \phi \rhd \mathbf{T}\neg Y$. Let $\mathbf{T}\neg X' = \mathtt{pruning}(\mathbf{T}\neg X)$ and $\mathbf{T}\neg Y' = \mathtt{pruning}(\mathbf{T}\neg Y)$. By induction hypothesis $\underline{K}, \phi \rhd \mathbf{T}\neg X'$ or $\underline{K}, \phi \rhd \mathbf{T}\neg Y'$. By Lemma 7.1, $\underline{K}, \phi \rhd \mathrm{cc}(\mathbf{T}\neg X') \cap \mathrm{cc}(\mathbf{T}\neg Y')$. Since this holds for every final element ϕ of $\underline{K}$ such that $\alpha \leq \phi$ and all the signed formulas in $\mathrm{cc}(\mathbf{T}\neg X') \cap \mathrm{cc}(\mathbf{T}\neg Y')$ are of the kind $\mathbf{T}\neg A$, we get that $\underline{K}, \alpha \rhd \mathrm{cc}(\mathbf{T}\neg X') \cap \mathrm{cc}(\mathbf{T}\neg Y')$. This immediately implies that $\underline{K}, \alpha \rhd (\mathrm{cc}(\mathbf{T}\neg X') \cap \mathrm{cc}(\mathbf{T}\neg Y')) \cup \{\mathbf{T}\neg(X' \wedge Y')\}$. Let $\Delta = \mathtt{simpl}((\mathrm{cc}(\mathbf{T}\neg X') \cap \mathrm{cc}(\mathbf{T}\neg Y')) \cup \{\mathbf{T}\neg(X' \wedge Y')\})$ (note that all the formulas in Δ are of the kind $\mathbf{T}\neg Z$). We have $\underline{K}, \alpha \rhd \Delta$, which implies $\underline{K}, \alpha \rhd \mathbf{T}\neg(\bigvee_{\mathbf{T}\neg Z \in \Delta} Z)$. For the converse, the proof is similar. □

[2]Namely, ϕ is a maximal element of $\underline{K}$ w.r.t. $\leq$.

Given a set of signed formulas Δ, $\texttt{pruning}(\Delta)$ is the set obtained by applying $\texttt{pruning}$ to every element in Δ. Now, let us consider the rule

$$\frac{\Delta}{\texttt{pruning}(\Delta)}\texttt{pruning}.$$

By Lemma 7.2, we immediately get the following.

THEOREM 7.3. *The rule* $\texttt{pruning}$ *is invertible.*

Example 7.4. As an example let us consider the signed formula

$$H \equiv \mathbf{F}(\,(p \vee (p \wedge q)) \to ((p \vee q) \to (p \wedge q))\,).$$

It is easy to check that for every $v \in \{p, q\}$, $v \not\preceq^{+} H$, $v \not\preceq^{-} H$ and $v \not\preceq^{-}_{w} H$. Moreover, also the rules given in Section 6 are not applicable. Let $X = p \vee (p \wedge q)$ and $Y = (p \vee q) \to (p \wedge q)$. We have:

$$\begin{aligned}
\texttt{pruning}(\mathbf{T}p) &= \mathbf{T}p \\
\texttt{pruning}(\mathbf{T}(p \wedge q)) &= \mathbf{T}(p \wedge q) \\
\texttt{pruning}(\mathbf{T}(p \vee q)) &= \mathbf{T}(p \vee q) \\
\texttt{pruning}(\mathbf{F}(p \wedge q)) &= \mathbf{F}(p \wedge q) \\
\texttt{pruning}(\mathbf{T}X) &= \texttt{extract}(\,\texttt{simpl}((\mathrm{cc}(\mathbf{T}p) \cap \mathrm{cc}(\mathbf{T}(p \wedge q))) \cup \{\mathbf{T}X\})\,) \\
&= \texttt{extract}(\,\texttt{simpl}((\{\mathbf{T}p\} \cap \{\mathbf{T}p,\ \mathbf{T}q)\}) \cup \{\mathbf{T}X\})\,) \\
&= \texttt{extract}(\,\texttt{simpl}(\{\mathbf{T}p,\ \mathbf{T}X\})\,) \\
&= \texttt{extract}(\,\{\,\mathbf{T}p\,\}\,) = \mathbf{T}p \\
\texttt{pruning}(\mathbf{F}Y) &= \texttt{extract}(\,\texttt{simpl}(\,\mathrm{cc}(\mathbf{T}(p \vee q)) \cup \mathrm{cc}(\mathbf{F}(p \wedge q)))\,) \\
&= \texttt{extract}(\,\texttt{simpl}(\,\{\mathbf{T}(p \vee q)\} \cup \{\mathbf{F}(p \wedge q)\}\,)\,) \\
&= \texttt{extract}(\,\{\,\mathbf{T}(p \vee q),\ \mathbf{F}(p \wedge q)\,\}\,) \\
&= \mathbf{F}((p \vee q) \to (p \wedge q)) = \mathbf{F}Y \\
\texttt{pruning}(H) &= \texttt{extract}(\,\texttt{simpl}(\,\mathrm{cc}(\mathbf{T}p) \cup \mathrm{cc}(\mathbf{F}(Y)))\,) \\
&= \texttt{extract}(\,\texttt{simpl}(\,\{\mathbf{T}p,\ \mathbf{F}(p \wedge q),\ \mathbf{F}Y\,\}\,)) \\
&= \texttt{extract}(\,\texttt{simpl}(\,\{\mathbf{T}p,\ \mathbf{F}(\top \wedge q),\ \mathbf{F}((\top \vee q) \to (\top \wedge q))\}\,)) \\
&= \texttt{extract}(\{\mathbf{T}p,\ \mathbf{F}\bot,\ \mathbf{F}q\}) \\
&= \mathbf{F}(p \to (q \vee \bot)).
\end{aligned}$$

By Lemma 7.2 we conclude that H is equivalent to $\mathbf{F}(p \to q)$, hence H is not provable.

Example 7.5. As another example let us consider the set

$$\Delta = \Gamma \cup \{\ \mathbf{T}(a \vee b)\,, \mathbf{F}(x \to ((x \vee z) \to y))\ \}$$

and let us suppose that $v \not\preceq^{+} \Gamma$, $v \not\preceq^{-} \Gamma$ and $v \not\preceq^{-}_{w} \Gamma$ for every $v \in \{a, b, x, y\}$, whereas $z \preceq^{-} \Gamma$. Note that $z \preceq^{+} \mathbf{F}(x \to ((x \vee z) \to y))$, thus $z \not\preceq^{+} \Delta$ and $z \not\preceq^{-} \Delta$; hence no rule of previous sections is applicable. Moreover, since

$$\begin{aligned}
z &\not\preceq^{-} \mathbf{F}(x \to ((x \vee z) \to y)) \\
z &\not\preceq^{-}_{w} \mathbf{F}(x \to ((x \vee z) \to y)),
\end{aligned}$$

we cannot apply the rules **T**-permanence, **T**¬-permanence and **F**-permanence. By applying the rule `pruning` to Δ we get the set

$$\Delta_1 = \texttt{pruning}(\Gamma) \cup \{\mathbf{T}(a \vee b), \mathbf{F}(x \to y)\}.$$

Since $z \preceq^- \Gamma$, it follows that $z \preceq^- \texttt{pruning}(\Gamma)$, thus $z \preceq^- \Delta_1$. Now, we can apply the rule **T**¬-permanence replacing z with $\bot$ in Δ_1 and then the Boolean simplification rules. This sequence of applications of invertible rules erases z and reduces the size of the set Δ. Without the rule `pruning`, such a result is obtainable by instantiating the premise of the noninvertible rule $\mathbf{F} \to$ with Δ, but this application, differently from the sequence of rules applied previously, introduces a backtrack point.

8. EMPIRICAL EVALUATION

Here we discuss how the rules described in the previous sections influence the performances of a theorem prover. First of all we notice that, implementing formulas by graphs where all occurrences of a given subformula are represented by pointers to the same node, and sets of formulas as pointers to formulas, the rules Replace-**T** and Replace-**T**¬ of Section 3 can be implemented in constant time. More than this, note that, given a set Δ where no replacement rule is applicable, after an application of a rule r of $\mathbf{T}_{\mathrm{Int}}$ the only formulas candidated to be replaced are those in evidence in the conclusion of r, hence it requires constant time to decide if one of the rules Replace-**T** and Replace-**T**¬ is applicable. As for the rule Replace-**F**, both to decide its applicability and to apply it requires time linear w.r.t. the size of the premise. Boolean simplifications can be applied only if a logical constant occurs in the set Δ (e.g., after a replacement application) and every single application of a Boolean simplification rule requires constant time; hence the whole Boolean simplification step requires linear time. To check the polarity of a propositional variable (i.e., to decide $p \preceq^+ H$, $p \preceq^- H$ or $p \preceq^-_w H$) requires linear time in the length of H; as a consequence, the application of permanence rules of Sections 4 and 5 requires linear time in the size of the premise. As for the context-permanence rules of Section 6, the number of contexts is linear in the length of the premise of the rule and, given a context, the application of the rule requires linear time in the size of the premise. Finally, the complexity of the pruning rule requires quadratic time in the size of the premise.

To get an empirical evidence of the impact of our simplification techniques, we realized a Prolog theorem prover, called BPPI (Basic Prolog Prover for Intuitionism), based on the calculus $\mathbf{T}_{\mathrm{Int}}$ extended with the rules of Section 3 and on the proof-search strategy described in Avellone et al. [2008]. According to the discussion of Section 2.1 on the relations between $\mathbf{T}_{\mathrm{Int}}$ and **Tab**, BPPI turns out to be a Prolog implementation of PITP [Avellone et al. 2008]. We point out that BPPI already implements replacement and Boolean simplifications rules of Section 3. We tested how performances of BPPI are affected by the optimizations introduced in this article. In particular we compare BPPI with the following theorem provers.

— EPPI (Extended Prolog Prover for Intuitionism) extends BPPI with the rules **T**-permanence, **T**¬-permanence of Section 4 and **F**-permanence of Section 5.
— EPPI+ extends EPPI with context rules of Section 6. An implementation of EPPI+ is presented in Ferrari et al. [2010], where the proof-search strategy is described in full details.
— EPPI++ extends EPPI+ with the rule `pruning` of Section 7.

Prover	0-1s	1-10s	10-100s	100-600s	>600s
BPPI	1025(101.4)	51(158.9)	11(281.8)	4(972.3)	9(n.a.)
EPPI	859(161.9)	226(710.7)	11(392.7)	4(1607.9)	0(0.0)
EPPI+	1049(65.2)	36(112.2)	11(238.2)	4(809.1)	0(0.0)
EPPI++	1063(95.1)	32(100.5)	3(55.7)	2(275.0)	0(0.0)

Fig. 6. Timings on random formulas.

SYJ formula	BPPI	EPPI	EPPI+	EPPI++	PITP	Imogen
201+1.018	11.83	14.09	16.26	33.86	0.01	11.32
201+1.019	14.97	17.70	21.07	42.74	0.01	16.28
201+1.020	19.14	22.78	26.26	53.54	0.01	17
202+1.005	0.99	0.83	0.98	0.90	0.01	179.063
202+1.006	9.37	6.51	7.41	7.20	0.03	timeout
202+1.007	100.46	53.85	62.55	61.07	0.3	timeout
205+1.011	27.70	0.02	0.02	0.03	0.0	0.028
205+1.012	84.59	0.03	0.03	0.03	0.0	0.032
205+1.013	242.33	0.04	0.03	0.05	0.0	0.020
206+1.018	0.0	0.0	0.0	120.28	0.0	2.26
206+1.019	0.0	0.0	0.0	119.48	0.0	2.12
206+1.020	0.0	0.0	0.0	116.92	0.0	2.14
207+1.003	0.32	0.42	0.50	0.62	0.0	0.024
207+1.004	5.08	6.45	7.98	10.86	0.01	0.056
207+1.005	102.10	118.11	139.45	195.29	2.54	0.116
208+1.014	121.60	105.50	116.80	135.36	0.0	timeout
208+1.015	203.35	179.18	201.18	226.92	0.04	timeout
208+1.016	333.04	291.36	342.54	373.71	0.05	timeout
209+1.006	2.13	2.85	3.51	3.92	0.01	0.012
209+1.007	19.64	26.82	33.81	33.42	0.22	0.008
209+1.008	198.94	269.41	328.47	336.97	1.950	0.016
211+1.009	27.19	0.01	0.01	0.01	0.050	0.020
211+1.010	84.76	0.01	0.01	0.02	0.120	0.020
211+1.011	253.94	0.02	0.02	0.02	0.260	0.020
212+1.018	0.01	0.02	0.04	200	0.0	0.036
212+1.019	0.02	0.04	0.04	203	0.0	0.048
212+1.020	0.02	0.04	0.05	196.14	0.0	0.064

Fig. 7. Timings on ILTP library.

Experiments have been carried out along the lines of Raths et al. [2007] and their results are summarized in Figures 6 and 7.[3]

In Figure 6 we report the timings of experiments performed on random generated formulas with 1024 connectives and a number of variables ranging from 1 to 1024. In every entry we indicate the number of formulas decided in the specified time range and between brackets we put the total time required to decide them; "9(n.a.)" in the last column means that 9 formulas require more than 600secs to be decided. The results emphasize that for formulas decidable in few steps, the overhead of the simplification rules slows down the prover, but when the formula requires a lot of computation optimizations are effective. As a matter of fact EPPI, EPPI+ and EPPI++ decide all the formulas within 600secs.

[3]Experiments have been carried out on a 3.00GHz Intel Xeon CPU computer with 2MB cache size and 2GB RAM.

We remark that we tested our provers on random generated formulas because simplification rules have little impact on the formulas of the ILTP-library, which is the standard benchmark for intuitionistic theorem provers. This is due to the fact that the ILTP library only includes 12 families of formulas and on many of them simplifications are not activated. For the sake of completeness in Figure 7 we show the timings of experiments performed on the ILTP Library [Raths et al. 2007] for the families with a significant computation time. First we notice that BPPI outperforms EPPI, EPPI+ and EPPI++ on the families SYJ201, SYJ207, SYJ209 and SYJ212, this because of the overhead required to check the applicability of simplifications while they are never applied. Nevertheless, we note that on these formulas the performances of EPPI and EPPI+ are not far from those of BPPI. As for the families where the sign permanence rules work, we notice a sharp gain of EPPI and EPPI+ with respect to BPPI, in particular this happens on the families SYJ205 and SYJ211. We remark that the sign permanence rules are applied also to decide the families SYJ202 and SYJ208. The family SYJ207 (de Bruijn formulas) is the only one where the context rules actually work. Although the timings show a slow-down due to the overhead needed to decide the application of the context rules we remark that their application reduce backtracking in proof-search. The performances could be surely improved by a more accurate implementation of data structures. Finally EPPI++ is far slower than EPPI and EPPI+ because the pruning rules are never applied to these families and the check of the applicability of the `pruning` rule is time consuming.

The last two columns of Figure 7 report the timings of PITP [Avellone et al. 2008] and Imogen [McLaughlin and Pfenning 2008], the fastest provers on the ILTP-library available so far. We remark that, in general, these provers outperform our implementation. This is not surprising since our Prolog prototypes lack of the graph formulas representation mentioned at the beginning of this section. Nevertheless, in the case of family SYJ211 our provers outperform PITP and Imogen.

9. CONCLUSIONS

In this article we have presented rules based on replacement of formulas with logical constants, with the aim to speed-up deductions. To prove correctness and invertibility we employ semantical techniques based on Kripke models. Invertibility of the rules is an important point, since enables us to apply them at any step of the deduction without requiring backtracking. As discussed in the previous section, to establish the conditions of the applicability of the rules low overhead is required. Moreover, these techniques improve the performances of theorem provers for Intuitionistic Logic.

We remark that, as far as we know, in the framework of tableau/sequent calculi for Intuitionistic Logic there is no general investigation on simplification/optimization techniques. In the literature on theorem provers for Intuitionistic Logic the only reported optimization techniques are the structural sharing of STRIP [Larchey-Wendling et al. 2001] and the focusing employed in Imogen [McLaughlin and Pfenning 2008]. Both these optimizations have a very different flavor with respect to our rules; indeed, the former is essentially an optimization regarding implementation issues, while the latter is an optimization strategy to control the nondeterminism in proof-search.

We point out that our rules can be adapted to sequent systems via the usual translation between tableau and sequent calculi. We refer the reader to Avellone et al. [1999], where such a translation is discussed also for tableau calculi using signs other than **T** and **F**.

The semantical techniques used in this article can be applied to design similar rules for intermediate and modal logics with Kripke semantics. In particular the rules **T**-permanence, **T**$\neg$-permanence and `pruning` can be straightforwardly adapted to first

order Intuitionistic Logic and to intermediate logics. These rules can also be reformulated for modal logics via a suitable definition of positive and negative occurrence of a propositional variable in a modal formula. As for the rules **F**-permanence and the context rules, the definitions of partial substitution and context strictly depend on the notion of validity in intuitionistic Kripke models. Hence their extension to other logics requires an appropriate reformulation for the logic at hand.

REFERENCES

AGUILERA, G., DE GUZMÁN, I. P., OJEDA-ACIEGO, M., AND VALVERDE, A. 2001. Reductions for non-clausal theorem proving. *Theor. Comput. Sci. 266*, 1–2, 81–112.

AVELLONE, A., FERRARI, M., AND MIGLIOLI, P. 1999. Duplication-free tableau calculi and related cut-free sequent calculi for the interpolable propositional intermediate logics. *Logic J. IGPL 7*, 4, 447–480.

AVELLONE, A., FIORINO, G., AND MOSCATO, U. 2008. Optimization techniques for propositional intuitionistic logic and their implementation. *Theor. Comput. Sci. 409*, 1, 41–58.

CHAGROV, A. AND ZAKHARYASCHEV, M. 1997. *Modal Logic*. Oxford University Press.

DAVIS, M., LOGEMANN, G., AND LOVELAND, D. 1962. A machine program for theorem-proving. *Comm. ACM 5*, 394–397.

DOWLING, W. AND GALLIER, J. 1984. Linear time algorithms for testing the satisfiability of propositional horn formulae. *J. Logic Program. 1*, 267–284.

FERRARI, M., FIORENTINI, C., AND FIORINO, G. 2010. Fcube: An efficient prover for intuitionistic propositional logic. In *Proceedings of the 17th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*. C. G. Fermüller and A. Voronkov Eds., Lecture Notes in Computer Science, vol. 6397, Springer, 294–301.

HUSTADT, U. AND SCHMIDT, R. A. 1998. Simplification and backjumping in modal tableau. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX'98)*. H. C. M. de Swart Ed., Lecture Notes in Computer Science, vol. 1397, 187–201.

KLEENE, S. 1967. *Mathematical Logic*. John Wiley & Sons, Inc.

LARCHEY-WENDLING, D., MÉRY, D., AND GALMICHE, D. 2001. Strip: Structural sharing for efficient proof-search. In *Proceedings of the 1st International Joint Conference on Automated Reasoning*. R. Goré, A. Leitsch, and T. Nipkow Eds., Lecture Notes in Computer Science, vol. 2083, Springer, 696–700.

DAVIS, M. AND PUTNAM, H. 1960. A computing procedure for quantification theory. *J. ACM 7*, 201–215.

MASSACCI, F. 1998. Simplification: A general constraint propagation technique for propositional and modal tableaux. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX '98)*. H. C. M. de Swart Ed., Lecture Notes in Computer Science, vol. 1397, 217–231.

MCLAUGHLIN, S. AND PFENNING, F. 2008. Imogen: Focusing the polarized inverse method for intuitionistic propositional logic. In *Proceedings of the International Conference on Logic Programming and Automated Reasoning (LPAR)*. I. Cervesato, H. Veith and A. Voronkov Eds., Lecture Notes in Computer Science, vol. 5330, Springer, 174–181.

RATHS, T., OTTEN, J., AND KREITZ, C. 2007. The ILTP problem library for intuitionistic logic. *J. Automat. Reason. 31*, 261–271.

ROBINSON, J. A. 1965. A machine-oriented logic based on the resolution principle. *J. ACM 12*, 1, 23–41.

SMULLYAN, R. 1968. *First-Order Logic*. Springer.

WEICH, K. 1998. Decision procedures for intuitionistic propositional logic by program extraction. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX'98)*. H. C. M. de Swart Ed., Lecture Notes in Computer Science, vol. 1397, Springer, 292–306.

ZHANG, H. AND STICKEL, M. E. 2000. Implementing the Davis-Putnam method. *J. Automat. Reason. 24*, 1/2, 277–296.

Received February 2010; revised December 2010; accepted March 2011