An Evaluation-Driven Decision Procedure for G3i

MAURO FERRARI, Dipartimento di Scienze Teoriche e Applicate, Università degli Studi dell’Insubria, mauro.ferrari@uninsubria.it
CAMILLO FIORENTINI, Dipartimento di Informatica, Università degli Studi di Milano, fiorentini@di.unimi.it
GUIDO FIORINO, Dipartimento di Informatica Sistemistica e Comunicazione, Università degli Studi di Milano-Bicocca, guido.fiorino@unimib.it

It is well-known that G3i, the sequent calculus for intuitionistic propositional logic where weakening and contraction are absorbed into the rules, is not terminating. Indeed, due to the contraction in the rule for left implication, the naive goal-oriented proof-search strategy, consisting in applying the rules of the calculus bottom-up until possible, can generate branches of infinite length. The usual solution to this problem is to support the proof-search procedure with a loop-checking mechanism which prevents the generation of infinite branches by storing and analyzing some information regarding the branch under development.

In this paper we propose a new technique based on evaluation functions. An evaluation function is a lightweight computational mechanism which, analyzing only the current goal of the proof-search, allows one to drive the application of rules so to guarantee termination and to avoid useless backtracking. We describe an evaluation-driven proof-search procedure that given a sequent $\sigma$ returns either a G3i-derivation of $\sigma$ or a counter-model for $\sigma$. We prove that such a procedure is terminating and correct and that the depth of the G3i-trees generated during proof-search is quadratic in the size of $\sigma$. Finally, we discuss the overhead time introduced by evaluation functions in the proof-search procedure.

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: proof-search procedures, intuitionistic propositional logic, sequent calculi

1. INTRODUCTION

The research on the design of efficient decision procedures for intuitionistic propositional logic has a long and articulated history. In the context of sequent calculi the main concern is the treatment of implicative formulas occurring in the left-hand side of a sequent. Let us consider the variant G3i [Troelstra and Schwichtenberg 2000] of the Gentzen’s sequent calculus LJ [Gentzen 1969] for intuitionistic propositional logic. In G3i weakening and contraction are absorbed into the rules and this makes this calculus more adequate for bottom-up proof-search than LJ. Since the rule $\rightarrow L$ for left implication retains the main formula $A \rightarrow B$ in the left premise, the naive proof-search strategy, consisting in applying the rules of the calculus bottom-up until possible, is not terminating since the formula $A \rightarrow B$ might be always selected for application. This problem originated two lines of research: one devoted to the identification of extra-logical conditions allowing one to cut proof-search so to avoid the generation of non-terminating branches; the second one concerns the design of structural calculi that do not need the re-use of formulas.

In the first line, we cite loop-checking [Gabbay and Olivetti 2000; Heuerding et al. 1996; Howe 1997; 1998]: whenever the “same” sequent occurs twice along a branch of the proof under construction, the search is cut. An efficient implementation of loop-checking exploits histories [Heuerding et al. 1996; Howe 1997]. In the construction of a branch, the formulas decomposed by right rules are stored in the history; loops are avoided by preventing the application of some right rules to formulas already in the history. The second line of research, which in proof-theoretical terms concerns the identification of contraction-free calculi, generated several calculi where the re-use
of implicative formulas is prevented by replacing $A \rightarrow B$ on the left with “simpler” formulas or adopting a non-standard notion of sequent [Vorob’ev 1970; Dyckhoff 1992; Hudelmaier 1993; Miglioli et al. 1997; Ferrari et al. 2009; 2013a].

In this paper we approach the problem from the first perspective: we show that terminating backward proof-search in G3i can be performed controlling termination by means of an evaluation function defined on sequents. Such a function $E$ “evaluates” a formula $A$ in a sequent $\sigma$ and yields a value $E(A, \sigma)$ in $\{F, T, X\}$; the outcome of the evaluation is exploited to drive the proof-search and to avoid useless applications of rules.

Let us better define the picture. In the paper, we refer to the version G3i (see Figure 3) where in addition to the standard rule for right implication, here called $\rightarrow$, we introduce the rule $\rightarrow$ evaluated to drive the proof-search and to avoid useless applications of formulas or adopting a non-standard notion of sequent [Vorob’ev 1970; Dyckhoff 1992; Hudelmaier 1993; Miglioli et al. 1997; Ferrari et al. 2009; 2013a].

Proof-search is performed by applying backward the rules of G3i. In this process, we alternate two phases: an unblocked phase (u-phase), where all the rules of G3i can be (backward) applied, and a blocked phase (b-phase), where only right rules of G3i can be used. The proof-search strategy is described by the recursive function $F$ at page 13; given $l \in \{b, u\}$ and a sequent $\sigma$, the invocation $F(l, \sigma)$ searches for a G3i-derivation of $\sigma$ starting from phase $l$. The pair $(l, \sigma)$ identifies the current state of the computation; a search for a derivation of $\sigma$ starts from state $(u, \sigma)$.

Proof-search is performed by applying backward the rules of G3i. In this process, we alternate two phases: an unblocked phase (u-phase), where all the rules of G3i can be (backward) applied, and a blocked phase (b-phase), where only right rules of G3i can be used. The proof-search strategy is described by the recursive function $F$ at page 13; given $l \in \{b, u\}$ and a sequent $\sigma$, the invocation $F(l, \sigma)$ searches for a G3i-derivation of $\sigma$ starting from phase $l$. The pair $(l, \sigma)$ identifies the current state of the computation; a search for a derivation of $\sigma$ starts from state $(u, \sigma)$. Only the rules for left implication, right disjunction and right implication can affect the phase. As an example of a transition that does not change the phase, let us suppose that the current state is $(u, \sigma = [A \rightarrow B, \Gamma \Rightarrow H])$ and that the rule $\lor L$ is applied with main formula $A \lor B$. Then, the next states are $(u, \sigma_A = [A, \Gamma \Rightarrow H])$ and $(u, \sigma_B = [B, \Gamma \Rightarrow H])$; this corresponds to the fact that, to build a G3i-derivation with root sequent $\sigma$ and root rule $\lor L$, we need a derivation of $\sigma_A$ and a derivation of $\sigma_B$.

A u-phase ends when we apply the rule for right disjunction or the rule for left implication. In detail, let us suppose that the current state is $(u, [A \rightarrow B, \Gamma \Rightarrow H])$ and that rule $\rightarrow L$ is applied; by definition of $\rightarrow L$, the next sequents to be proved are $\sigma_A = [A \rightarrow B, \Gamma \Rightarrow H]$ and $\sigma_B = [B, \Gamma \Rightarrow H]$. While the latter maintains the u-phase, the former leads to the state $(b, \sigma_A)$, with phase changed to b. If rule $\rightarrow B$ is applied in $(b, [\Gamma \Rightarrow H_0 \lor H_1])$, the next state is $(b, [\Gamma \Rightarrow H_i])$, where $i \in \{0, 1\}$. Note that the choice of $i$ determines a backtrack point in proof-search. The evaluation function $E$ prevents useless applications of $\rightarrow L$ and $\lor R_i$. Indeed, in $(u, \sigma = [A \rightarrow B, \Gamma \Rightarrow H])$ the application of $\rightarrow L$ is allowed only if $E(A, \sigma) \neq F$; the transition from $(u, \sigma = [\Gamma \Rightarrow H_0 \lor H_1])$ to $(b, [\Gamma \Rightarrow H_i])$ is performed only if $E(H_i, \sigma) \neq F$. In both cases the evaluation function avoids the application of a rule that would lead to a failure in proof-search.

Now, let us consider the b-phase. As mentioned above, a b-phase only permits the application of right rules, so that it resembles a right-focused phase of a focused calculus (see, e.g., [Dyckhoff and Lengrand 2006]). A b-phase terminates when a state $(b, \sigma = [\Gamma \Rightarrow A \rightarrow B])$ such that $E(A, \sigma) \neq T$ is reached: in this case, rule $\rightarrow R_2$ is applied, the phase reverts to u and the next state is $(u, [A, \Gamma \Rightarrow B])$.

Whenever we reach a state $(l, [A, \Gamma \Rightarrow A])$ or a state $(l, [\bot, \Gamma \Rightarrow H])$, we have successfully built a branch of a G3i-derivation for the initial sequent. The search fails when we reach a state $(u, \sigma = [\Gamma \Rightarrow H])$ where $\Gamma$ only contains propositional variables or implicative formulas $A \rightarrow B$ such that $E(A, \sigma) = F$, and $H$ is either $\bot$ or a propositional variable not in $\Gamma$ or a disjunction $H_0 \lor H_1$ such that $E(H_0, \sigma) = E(H_1, \sigma) = F$; we call irreducible a sequent $\sigma$ of this kind. We show that irreducible sequents are not intuitionistically valid, so there would be no advantage in applying the rules for left
implication or right disjunction. In this case, to continue the proof-search a backtrack step is needed.

Rules for right implication have a crucial role in the proof of termination of the proof-search procedure. In a state \((I, \sigma = [\Gamma \Rightarrow A \Rightarrow B])\), the choice between \(\rightarrow R_1\) and \(\rightarrow R_2\) is determined by the value of \(\mathcal{E}(A, \sigma)\): if \(\mathcal{E}(A, \sigma) = T\), we have to apply \(\rightarrow R_1\) and the next state is \((I, [\Gamma \Rightarrow B])\); otherwise, rule \(\rightarrow R_2\) must be applied and proof-search carries on with \((u, [A, \Gamma \Rightarrow B])\). While in the former case the phase does not change, in the latter the phase is set to \(u\). We stress that \(\rightarrow R_2\) is the only rule that can conclude a \(b\)-phase. By definition of \(\rightarrow R_2\) and the properties of \(\mathcal{E}\), it follows that, along a branch of a derivation, we cannot apply twice the rule \(\rightarrow R_2\) with the same main formula \(A \Rightarrow B\). Since \(G3i\) has the subformula property, the number of possible main formulas \(A \Rightarrow B\) is linear in the size of the sequent to be proved. Thus, we get a bound on the number of changes from \(b\) to \(u\) phases and this implies the termination of the procedure. As discussed in Section 8, the depth of the \(G3i\)-trees generated by our decision procedure is quadratic in the size of the sequent to be proved. We remark that such a bound is optimal as discussed, e.g., in [Buss and Iemhoff 2003]

To prove the correctness of our decision procedure we must show that \(\mathcal{F}(u, \sigma)\) returns a \(G3i\)-derivation of \(\sigma\) iff \(\sigma\) is intuitionistically valid. The tricky part of the proof is to show that, if \(\mathcal{F}(u, \sigma)\) fails to return a \(G3i\)-derivation of \(\sigma\), then \(\sigma\) is not intuitionistically valid. To prove this fact we define the function \(\mathcal{F}\) so that, if the the search for a \(G3i\)-derivation of \(\sigma\) fails, it returns a structure which codifies the failed proof-search and that can be used to build a counter-model for \(\sigma\). To this aim, following a standard approach (see, e.g., [Ferrari et al. 2013a; Pinto and Dyckhoff 1995]), we introduce a calculus \(RG3i\) for asserting intuitionistic unprovability (Section 6) which is dual to the direct calculus \(G3i\). Differently from \(G3i\), sequents of \(RG3i\) are labelled to record the phase information so to get a compact representation of a failed proof-search. Actually, \(\mathcal{F}(u, \sigma)\) returns either a \(G3i\)-derivation of \(\sigma\) or an \(RG3i\)-derivation of \(\sigma^u\) (\(\sigma\) with label \(u\)). As shown in Section 9, from an \(RG3i\)-derivation of a sequent \(\sigma^u = [\Gamma \Downarrow B, H]\), we can extract an intuitionistic Kripke model \(K\) such that, at its root, all the formulas in \(\Gamma\) are forced while \(H\) is not; this means that \(K\) is a counter-model for \(\sigma\), hence \(\sigma\) is not intuitionistically valid.

To maintain the presentation as general as possible, we do not bind ourselves to a specific evaluation function but we use an abstract notion of evaluation (see Section 3). The concrete evaluation \(\mathcal{E}\) we use in the examples is presented in Section 3.1. In Section 10, we discuss the complexity of computing evaluations and we prove that the overhead time required to compute \(\mathcal{E}\) in the construction of a branch is cubic in the size of the sequent \(\sigma\) to be decided. In the conclusions, we show that, considering a slight modification of the proof-search procedure, this overhead can be reduced so to be quadratic in the size of \(\sigma\). In Section 11 we show another example of evaluation function related to the implication-locking technique presented in [Franzén 1988].

To summarize our achievements, in this paper we show a terminating proof-search procedure for \(G3i\) where loops are avoided by a local analysis of the current goal of the proof-search, exploiting evaluation functions. Hence, differently from approaches based on loop-checking, we do not need to store any information about the development of a branch. From a worst-case analysis perspective, our approach and those based on histories [Heuerding et al. 1996; Howe 1997; 1998] have the same computational properties. Indeed, in both approaches the length of the branches generated during the proof-search is quadratic in the size of the sequent to be proved. As we discuss in the Section 11, the search space of our strategy is never larger than the one generated by history based mechanisms and there are cases where our approach outperforms the one based on histories. To prove this, we compare the behaviour of
the two approaches on the family $\sigma_S_n$ of intuitionistically valid sequents introduced in [Franzén 1988]. We show that, while our proof-search procedure builds a derivation of $\sigma_S_n$ without performing any backtrack step, those based on histories [Heuerding et al. 1996; Howe 1997; 1998] might require an high number of backtrack steps depending on the size of $\sigma_S_n$. As a consequence of this comparison, evaluation functions provide a valuable tool to exploit logical properties of sequents which are relevant to proof-search and that are not fully exploited by the logical rules of the calculus and by history based mechanisms.

In the conclusions we also make a comparison between our approach and those based on implication-locking discussed in [Franzén 1988; Buss and Iemhoff 2003; Corsi and Tassi 2007] and we briefly discuss the relations with contraction-free calculi [Vorob’ev 1970; Dyckhoff 1992; Hudelmaier 1993; Miglioli et al. 1997; Ferrari et al. 2009; 2013a], and focusing [Dyckhoff and Lengrand 2006; Miller and Pimentel 2013].

A first application of evaluation functions as a tool to avoid loop-checking is presented in [Ferrari et al. 2013b]. In that paper, we use a labelled variant of G3i, instead of the original G3i used in this paper, and a weaker notion of evaluation only aimed at guaranteeing termination. Indeed, the evaluations in [Ferrari et al. 2013b] only consider left-hand sides of sequents (they lack the value $F$) and are only used in the application of rules for right implication. Here, we refine the definition, so that evaluations are also used to avoid useless applications of $\to L$ and $\lor R_i$, and hence to reduce backtrack in proof-search.

The prover $g3ied$, available at http://www.dista.uninsubria.it/~ferram, implements the proof-search procedure $F$ and its variants discussed in the conclusions.

2. PRELIMINARIES

We consider the propositional language $\mathcal{L}$ based on a denumerable set of propositional variables $V$, the logical connectives $\land$, $\lor$, and the logical constant $\bot$. Writing formulas we assume that $\land$ and $\lor$ bind stronger than $\to$. We use $\neg A$ as a shorthand for $A \to \bot$ and $A \leftrightarrow B$ as a shorthand for $(A \to B) \land (B \to A)$. Given a formula $A$, $|A|$ denotes the size of $A$, that is the number of symbols occurring in $A$. The set $\text{Sf}(A)$ of subformulas of $A$ is defined as follows:

$$\text{Sf}(A) = \begin{cases} \{A\} & \text{if } A \in V \text{ or } A = \bot \\ \{A\} \cup \text{Sf}(B) \cup \text{Sf}(C) & \text{if } A = B \cdot C \text{ with } \cdot \in \{\land, \lor, \to\} \end{cases}$$

A (finite) Kripke model for $\mathcal{L}$ is a structure $\mathcal{K} = \langle P, \leq, \rho, V \rangle$, where:

- $\langle P, \leq, \rho \rangle$ is a finite partially ordered set with minimum $\rho$;
- $V$ is a function mapping every $\alpha \in P$ to a subset of $V$ such that $\alpha \leq \beta$ implies $V(\alpha) \subseteq V(\beta)$.

We write $\alpha < \beta$ to mean $\alpha \leq \beta$ and $\alpha \neq \beta$. The forcing relation $\models P \times \mathcal{L}$ of $\mathcal{K}$ is defined as follows:

- $\mathcal{K}, \alpha \models \bot$ does not hold and, for every $p \in V$, $\mathcal{K}, \alpha \models p$ iff $p \in V(\alpha)$;
- $\mathcal{K}, \alpha \models A \land B$ iff $\mathcal{K}, \alpha \models A$ and $\mathcal{K}, \alpha \models B$;
- $\mathcal{K}, \alpha \models A \lor B$ iff $\mathcal{K}, \alpha \models A$ or $\mathcal{K}, \alpha \models B$;
- $\mathcal{K}, \alpha \models A \to B$ iff, for every $\beta \in P$ such that $\alpha \leq \beta$, $\mathcal{K}, \beta \models A$ implies $\mathcal{K}, \beta \models B$.

We write $\mathcal{K}, \alpha \not\models A$ to mean that $\mathcal{K}, \alpha \models A$ does not hold. Monotonicity property holds for arbitrary formulas, i.e.: $\mathcal{K}, \alpha \models A$ and $\alpha \leq \beta$ imply $\mathcal{K}, \beta \models A$. A formula $A$ is valid in $\mathcal{K}$ iff $\mathcal{K}, \rho, A$. It is well-known that intuitionistic propositional logic coincides with the set of formulas valid in all (finite) Kripke models [Chagrov and Zakharyaschev 1997].
In this paper we use three kinds of sequents: unlabelled sequents and labelled sequents, where the label can be either \( b \) (blocked sequent) or \( u \) (unblocked sequent). Labelled sequents are used in the refutation calculus \( \text{RG3i} \) described in Section 6 and are needed to encode in the structure of the calculus some properties of the proof-search procedure of Section 7. We point out that labels do not have any particular semantical meaning. The notions and properties introduced hereafter apply in the same way to labelled and unlabelled sequents.

Formally, given a finite, possibly empty, set of formulas \( \Gamma \) and a formula \( H \), a sequent is an expression of the kind \([\Gamma \Rightarrow H]\) (unlabelled sequent), \([\Gamma \Rightarrow^* H]\) (blocked sequent) or \([\Gamma \Rightarrow^u H]\) (unblocked sequent). With \([\Gamma \Rightarrow^* H]\) we denote a sequent of any of the above kinds (* is either the empty label or a label in \{\( b, u \}\}). Writing sequents, we adopt the usual concise notational conventions, where \( \Gamma \) and \( \Delta \) are finite sets of formulas and \( A \) and \( H \) are formulas:

- \([\Gamma, \Delta \Rightarrow^* H]\) stands for \([\Gamma \cup \Delta \Rightarrow^* H]\);
- \([A, \Gamma \Rightarrow^* H]\) stands for \([\{A\} \cup \Gamma \Rightarrow H]\);
- \([\Rightarrow^* H]\) stands for \([\emptyset \Rightarrow^* H]\).

The size of \( \sigma = [\Gamma \Rightarrow^* H] \), denoted by \(|\sigma|\), is the total number of symbols occurring in the formulas of \( \sigma \), that is \(|\sigma| = \sum_{A \in \Gamma} |A| + |H|\). The set of subformulas of \( \sigma = [\Gamma \Rightarrow^* H] \) is \( \text{Sf}(\sigma) = \bigcup_{A \in \Gamma \cup \{H\}} \text{Sf}(A) \).

The semantics of formulas extends to sequents as follows. Given a Kripke model \( \mathcal{K} = (P, \leq, \rho, V) \), \( \alpha \in P \) and a sequent \( \sigma = [\Gamma \Rightarrow H] \), \( \alpha \) refutes \( \sigma \) in \( \mathcal{K} \), written \( \mathcal{K}, \alpha \not\models \sigma \), iff:

- \( \mathcal{K}, \alpha \models A \) for every \( A \in \Gamma \);
- \( \mathcal{K}, \alpha \not\models H \).

We say that \( \sigma \) is refutable if there exists a Kripke model \( \mathcal{K} \) with root \( \rho \) such that \( \mathcal{K}, \rho \not\models \sigma \); in this case \( \mathcal{K} \) is a counter-model for \( \sigma \).

A sequent \( \sigma \) is intuitionistically valid iff it is not refutable; a formula \( H \) is intuitionistically valid iff the sequent \([\Rightarrow H]\) is intuitionistically valid. It is easy to check that \([\Gamma \Rightarrow^* H]\) is intuitionistically valid iff the formula \( \bigwedge \Gamma \rightarrow H \) is intuitionistically valid.

3. EVALUATIONS

Let \( \text{Seq}_\mathcal{L} \) denote the set of all the sequents on \( \mathcal{L} \). An evaluation is a function \( \mathcal{E} : \mathcal{L} \times \text{Seq}_\mathcal{L} \rightarrow \{T, F, X\} \). Intuitively:

- \( \mathcal{E}(A, \sigma) = T \) means that \( \sigma \) contains enough information to assert the “satisfiability” of \( A \) in the following sense: \( \mathcal{E}(A, \sigma) = T \) and \( \mathcal{K}, \alpha \not\models \sigma \) implies \( \mathcal{K}, \alpha \models A \).
- \( \mathcal{E}(A, \sigma) = F \) means that \( \sigma \) contains enough information to assert that \( A \) is not forced in a world \( \alpha \) “strictly refuting” \( \sigma \), i.e.: if \( \mathcal{K}, \alpha \not\models \sigma \) and \( V(\alpha) \) coincides with the set of propositional variables in the left-hand side of \( \sigma \), then \( \mathcal{K}, \alpha \not\models A \).
- \( \mathcal{E}(A, \sigma) = X \) means that nothing can be stated about the semantical relation between \( A \) and \( \sigma \).

Given an evaluation \( \mathcal{E} \), the left-valuation \( \mathcal{E}^L \) of \( \mathcal{E} \) is defined as:

\[
\mathcal{E}^L(A, [\Gamma \Rightarrow^* H]) = \mathcal{E}(A, [\Gamma \Rightarrow^* \bot]) .
\]

\( \mathcal{E}^L \) only depends on \( \Gamma \), indeed: \( \mathcal{E}^L(A, [\Gamma \Rightarrow^* H_1]) = \mathcal{E}^L(A, [\Gamma \Rightarrow^* H_2]) \) for every \( H_1 \) and \( H_2 \). In Condition (C1) below we require that the evaluation of a formula \( A \) in \( \sigma \) only depends on the subformulas of \( A \) occurring in \( \sigma \). To formalize this fact we define the restriction
\[ B(A) = A \quad \text{if } A \in \mathcal{V} \cup \{\bot, \top\} \]
\[ B(A_0 \land A_1) = \begin{cases} \bot & \text{if } B(A_i) = \bot \text{ for some } i \in \{0, 1\} \\ B(A_i) & \text{with } i \in \{0, 1\} \text{ if } B(A_{1-i}) = \top \\ B(A_0) \land B(A_1) & \text{otherwise} \end{cases} \]
\[ B(A_0 \lor A_1) = \begin{cases} \top & \text{if } B(A_i) = \top \text{ for some } i \in \{0, 1\} \\ B(A_i) & \text{with } i \in \{0, 1\} \text{ if } B(A_{1-i}) = \bot \\ B(A_0) \lor B(A_1) & \text{otherwise} \end{cases} \]
\[ B(A \rightarrow B) = \begin{cases} \top & \text{if } B(A) = \bot \text{ or } B(B) = \top \\ B(B) & \text{if } B(A) = \top \\ B(A) \rightarrow B(B) & \text{otherwise} \end{cases} \]

**Fig. 1.** Boolean simplification

of \( \sigma = [\Gamma \models H] \) to \( A \) as the sequent:

\[
\text{Restr}(\sigma, A) = \begin{cases} [\Gamma \cap Sf(A) \models H] & \text{if } H \in Sf(A) \\ [\Gamma \cap Sf(A) \models \bot] & \text{otherwise} \end{cases}
\]

Let \( \sigma = [\Gamma \models H] \); the evaluation function \( \mathcal{E} \) must satisfy the following conditions:

(E1) \( \mathcal{E}(A, \sigma) = \mathcal{E}(A, \text{Restr}(\sigma, A)) \).
(E2) \( A \in \mathcal{V} \cup \{\bot\} \) implies

\[
\mathcal{E}(A, [\Gamma \models H]) = \begin{cases} \top & \text{if } A \in \Gamma \\ \bot & \text{otherwise} \end{cases}
\]

(E3) \( \mathcal{E}(A, \sigma) = \top \) iff \( \mathcal{E}^L(A, \sigma) = \top \).
(E4) \( \mathcal{E}^L(A, \sigma) = \bot \) implies \( \mathcal{E}(A, \sigma) = \bot \).
(E5) \( A \in \Gamma \) implies \( \mathcal{E}^L(A, \sigma) = \top \).
(E6) \( \mathcal{E}^L(A, \sigma) = \top \) and \( \mathcal{E}^L(B, \sigma) = \top \) imply \( \mathcal{E}^L(A \land B, \sigma) = \top \).
(E7) \( \mathcal{E}^L(A, \sigma) = \top \) or \( \mathcal{E}^L(B, \sigma) = \top \) implies \( \mathcal{E}^L(A \lor B, \sigma) = \top \).
(E8) \( \mathcal{E}^L(B, \sigma) = \bot \) implies \( \mathcal{E}^L(A \rightarrow B, \sigma) = \top \).
(E9) If \( A_0 \land A_1 \notin \Gamma \) and \( \mathcal{E}^L(A_i, \sigma) = \bot \) for some \( i \in \{0, 1\} \), then \( \mathcal{E}^L(A_0 \land A_1, \sigma) = \bot \).
(E10) If \( A \lor B \notin \Gamma \) and \( \mathcal{E}^L(A, \sigma) = \bot \) and \( \mathcal{E}^L(B, \sigma) = \bot \) then \( \mathcal{E}^L(A \lor B, \sigma) = \bot \).
(E11) If \( A \rightarrow B \notin \Gamma \) and \( \mathcal{E}^L(A, \sigma) = \top \) and \( \mathcal{E}^L(B, \sigma) = \bot \), then \( \mathcal{E}^L(A \rightarrow B, \sigma) = \bot \).
(E12) Let \( \mathcal{K} = (P, \leq, \rho, \mathcal{V}) \) be a Kripke model and \( \alpha \in P \) such that \( \mathcal{K}, \alpha \vdash \sigma \).

(E12.1) \( \mathcal{E}^L(A, \sigma) = \top \) implies \( \mathcal{K}, \alpha \models A \);
(E12.2) \( \mathcal{E}(A, \sigma) = \bot \) and \( V(\alpha) = \Gamma \cap \mathcal{V} \) imply \( \mathcal{K}, \alpha \not\models A \).

We remark that conditions (E1)–(E11) concern syntactical properties while Condition (E12) is semantical.

### 3.1. A concrete evaluation

Here we introduce the concrete evaluation function \( \tilde{\mathcal{E}} \) we use in the examples. The proof that \( \tilde{\mathcal{E}} \) satisfies properties (E1)–(E12) is lengthy and it is postponed to Section 10.

Let \( \mathcal{L}_\top \) be the language extending \( \mathcal{L} \) with the constant \( \top \) \( (\mathcal{K}, \alpha \models \top \text{ for every Kripke model } \mathcal{K} \text{ and for every } \alpha \text{ in } \mathcal{K}) \). The evaluation function is defined by means of the simplifications displayed in figures 1 and 2. The function \( B : \mathcal{L}_\top \rightarrow \mathcal{L}_\top \) defined in Figure 1 is the usual boolean simplification of formulas [Massacci 1998; Ferrari et al. 2012]. The functions \( \mathcal{R}^L, \mathcal{R}^H \) and \( \mathcal{R} \) defined in Figure 2 take as arguments a formula in \( \mathcal{L}_\top \) and a
An Evaluation-Driven Decision Procedure for G3i

\[ R^L(A, \sigma) = \begin{cases} \top & \text{if } A \in \Gamma \\ A & \text{if } A \notin \Gamma \text{ and } A \in \mathcal{V} \cup \{\bot, \top\} \\ B(R^L(A_0, \sigma) \cdot R^L(A_1, \sigma)) & \text{if } A \notin \Gamma \text{ and } A = A_0 \cdot A_1 \\ \end{cases} \]

\[ \text{with } \cdot \in \{\land, \lor, \to\} \]

\[ R^R(A, \sigma) = \begin{cases} \bot & \text{if } A = H \\ A & \text{if } A \neq H \text{ and } \\ B(R^R(A_0, \sigma) \cdot R^R(A_1, \sigma)) & \text{if } A \neq H \text{ and } A = A_0 \cdot A_1 \\ \end{cases} \]

\[ \text{with } \cdot \in \{\land, \lor\} \]

\[ R(A, \sigma) = R^R(R^L(A, \sigma), \sigma) \]

\[ \text{Fig. 2. Simplifications } (\sigma = [\Gamma \Rightarrow H]) \]

sequent in SeqL and yield a formula in L\(\tau\). These functions perform a simplification of a formula A w.r.t. a sequent \(\sigma = [\Gamma \Rightarrow H]\); \(R^L(A, \sigma)\) simplifies the formula A according with the information contained in \(\Gamma\) while \(R^R(A, \sigma)\) simplifies A with respect to H.

We define the function \(\hat{E} : L \times SeqL \rightarrow \{T, F, X\}\) as:

\[ \hat{E}(A, \sigma) = \begin{cases} \top & \text{if } R^L(A, \sigma) = \top \\ F & \text{if } R(A, \sigma) \text{ is a local formula} \\ X & \text{otherwise} \end{cases} \]

where a local formula [Ferrari et al. 2010] is a formula L of L\(\tau\) defined according with the following grammar:

\[ L ::= \top \mid p \mid L \land A \mid A \land L \mid L \lor L \mid L \to L \mid p \in \mathcal{V} \text{ and } A \text{ any formula} \]

Note that \(\top\) is not a local formula.

In Section 10 we show that \(\hat{E}\) meets properties (E1)–(E12) and we discuss its impact on the time complexity of our proof-search procedure.

4. FORMALIZATION OF CALCULI AND DERIVATIONS

In this paper we introduce two sequent calculi, one for proving sequents and one for refuting them. To treat these calculi in a uniform way we introduce the following definitions. A rule has the form:

\[ \sigma_1 \ldots \sigma_n \rightarrow^R \]

where \(\sigma, \sigma_1, \ldots, \sigma_n (n \geq 0)\) are sequents and \(R\) is the name of the rule. The sequent \(\sigma\) is the conclusion of the rule, while \(\sigma_1, \ldots, \sigma_n\) are its premises. An axiom rule is a rule with no premises; we call initial sequent its conclusion. A sequent calculus C is a finite set of rules.

Given a tree T, we denote with root(T) the root of T. If a is a node in T, children(a) denotes the set of the immediate successors of a in T. A leaf is any node a of T such that children(a) = \emptyset.

Given a sequent calculus C, a C-tree is a triple \(\pi = \langle T, s, r \rangle\) where:
In this case we say that $\pi$ is a $\pi$-tree where $T$ is the subtree of $T$ having root $a$ and the functions $s'$ and $r'$ are the restriction of $s$ and $r$ to the nodes in $T'$; we call $\pi'$ an immediate subtree of $\pi$. It is easy to check that $\pi'$ is a C-derivation if $\pi$ is; in this case we say that $\pi'$ is an immediate subderivation of $\pi$.

5. THE SEQUENT CALCULUS G3I

In Figure 3 we present the rules of the calculus G3i [Troelstra and Schwichtenberg 2000] for intuitionistic propositional logic. This calculus corresponds to the calculus $LJ$ [Gentzen 1969] with internalized weakening and contraction. We notice that contraction is explicitly introduced in the rule $\rightarrow L$. Moreover, as discussed in the introduction, to simplify the proof of results in Section 9, we add the rule $\rightarrow R_1$ for right implication to the formulation given in [Troelstra and Schwichtenberg 2000]. We remark that in [Troelstra and Schwichtenberg 2000] the rules of G3i act on sequents where the left-hand side consists of a multiset. Here we use sequents based on sets because this simplifies our discussion, however all our results can be reformulated considering multisets instead of sets.

The calculus consists of left (L) and right (R) introduction rules for the logical constants plus the axiom rules $\bot L$ and $\text{Id}$. Given a sequent $[\Gamma \Rightarrow H]$, left rules act on formulas in $\Gamma$ and right rules act on $H$. The main formula of a rule is the formula put in evidence in the conclusion of the rule. In the conclusion of left rules, we write $K, \Gamma$ to display the main formula $K$. Applying the rules of the calculus bottom-up we assume that 

\[ [\bot, \Gamma \Rightarrow H] \Downarrow L \quad [H, \Gamma \Rightarrow H] \text{Id} \]

\[ [A, B, \Gamma \Rightarrow H] \land L \quad [\Gamma \Rightarrow A, \Gamma \Rightarrow B] \land R \]

\[ [A \land B, \Gamma \Rightarrow H] \lor L \quad [\Gamma \Rightarrow A_i] \lor R_i \quad i \in \{0, 1\} \]

\[ [A \lor B, \Gamma \Rightarrow H] \rightarrow L \quad [\Gamma \Rightarrow B, \Gamma \Rightarrow A] \rightarrow R_1 \quad [\Gamma \Rightarrow A \rightarrow B, \Gamma \Rightarrow A] \rightarrow R_2 \]

Fig. 3. The sequent calculus G3i.
that \( K \not\in \Gamma \); this guarantees that the size of the premises is lower than the size of the conclusion, apart the case of the left premise of the \( \rightarrow L \) rule.

A sequent \( \sigma \) is provable in G3i iff there exists a G3i-derivation of \( \sigma \).

The following notions are standard. Let \( R \) be a rule of G3i:

- \( R \) is sound if the refutability of the conclusion of \( R \) implies the refutability of at least one of its premises;
- a premise of \( R \) is invertible if its refutability implies the refutability of the conclusion;
- \( R \) is invertible if all its premises are invertible.

From the soundness and completeness of G3i [Troelstra and Schwichtenberg 2000], we get:

**THEOREM 5.1.** A sequent is provable in G3i iff it is intuitionistically valid. \( \square \)

As for invertibility of the rules we note that:

- The rule \( \rightarrow L \) is not invertible since its leftmost premise is not invertible; note that the rightmost premise of \( \rightarrow L \) is invertible.
- The rules \( \lor R_i \) and \( \rightarrow R_1 \) are not invertible.
- All the other rules are invertible.

We recall that G3i has the subformula property, that is: if \( \sigma' \) is a sequent occurring in a G3i-tree with root sequent \( \sigma \), then every formula in \( \sigma' \) belongs to \( Sf(\sigma) \).

6. THE REFUTATION CALCULUS RG3I

RG3i is a refutation calculus for intuitionistic propositional logic, that is a calculus to derive the unprovability of a sequent. Differently from G3i, the calculus RG3i acts on labelled sequents. Given a sequent \( \sigma = [\Gamma \Rightarrow H] \), \( \sigma^b \) and \( \sigma^u \) denote \( [\Gamma_b \Rightarrow H] \) and \( [\Gamma_u \Rightarrow H] \) respectively.

The formulation of the calculus depends on the choice of an evaluation function \( E \) as defined in Section 3. \( E \) is used to state the side conditions on the applicability of some of the rules of RG3i.

Before commenting on the rules, we introduce some terminology. We denote with \( \Gamma^\land \) a finite set of propositional variables and with \( \Gamma^\lor \) a finite set of implicative formulas.

Let \( \sigma = [\Gamma \Rightarrow H] \), we define the set \( \text{act}^\land(\sigma) \) of active antecedents of implicative formulas in the left-hand side of \( \sigma \) and the set \( \text{act}^\lor(\sigma) \) of active disjuncts in the right-hand side of \( \sigma \) as follows:

\[
\text{act}^\land(\sigma) = \{ A \mid A \rightarrow B \in \Gamma \text{ and } E(A, \sigma) \neq F \}
\]

\[
\text{act}^\lor(\sigma) = \begin{cases} 
\{ H_i \mid i \in \{0,1\} \text{ and } E(H_i, \sigma) \neq F \} & \text{if } H = H_0 \lor H_1 \\
\emptyset & \text{otherwise}
\end{cases}
\]

A sequent \( \sigma^u \) is irreducible if it is an unblocked sequent of the form \( [\Gamma^\lor, \Gamma^\land \cup \Rightarrow H] \), where:

- \( H = \bot \) or \( H \in \forall \setminus \Gamma^\land \) or \( H = H_0 \lor H_1 \);
- \( \text{act}^\land(\sigma) = \emptyset \) and \( \text{act}^\lor(\sigma) = \emptyset \).

Thus, for every \( A \rightarrow B \in \Gamma^\lor \), it holds that \( E(A, \sigma^u) = F \) and, if \( H = H_0 \lor H_1 \), then \( E(H_0, \sigma^u) = E(H_1, \sigma^u) = F \). In Lemma 9.3 we show that irreducible sequents are refutable.

The rules of the calculus RG3i are given in Fig. 4. The only axiom rule is Irr and the initial sequents of RG3i are the irreducible sequents. We notice that the rules \( \land R_i \) and \( \rightarrow R_1 \) can be applied to blocked and unblocked sequents; in an application of
In the formulation of the rules $l \in \{u, b\}$.

\[ \frac{\Gamma \vdash H}{\Gamma, \text{At } u \vdash H} \quad \text{Irr} \quad \text{if } [\Gamma \vdash H] \text{ is irreducible} \]

\[ \frac{\Gamma \vdash H}{A \land B, \Gamma \vdash H} \land L \]

\[ \frac{\Gamma \vdash H_i}{\Gamma \vdash H_0 \land H_1} \land R_i \quad i \in \{0, 1\} \]

\[ \frac{\Gamma \vdash H}{A_0 \lor A_1, \Gamma \vdash H} \lor L_i \quad i \in \{0, 1\} \]

\[ \frac{\Gamma \vdash B}{A \rightarrow B, \Gamma \vdash H} \rightarrow L \quad \text{if } [A \rightarrow B, \Gamma \vdash H] \text{ is not irreducible} \]

\[ \frac{\Gamma \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow R_1 \quad \text{if } E(A, [\Gamma \vdash A \rightarrow B]) = T \]

\[ \frac{\Gamma \vdash B}{\Gamma \vdash A \rightarrow B} \rightarrow R_2 \quad \text{if } E(A, [\Gamma \vdash A \rightarrow B]) \neq T \]

\[ \frac{\{ \Gamma \vdash H_i \}_{H_i \in \text{act}^\vee(\sigma)}}{\Gamma \vdash H_0 \lor H_1} S_b \quad \text{if } \text{act}^\vee(\sigma) \neq \emptyset, \text{ where } \sigma = [\Gamma \vdash H_0 \lor H_1] \]

\[ \frac{\{ \Gamma \vdash A \}_A \in \text{act}^\rightarrow(\sigma)}{\{ \Gamma \vdash H_i \}_{H_i \in \text{act}^\vee(\sigma)}} S_u \]

if $\text{act}^\rightarrow(\sigma) \cup \text{act}^\vee(\sigma) \neq \emptyset$, where $\sigma = [\Gamma \vdash H] \text{ and } (H = \bot \text{ or } H \in \mathcal{V} \setminus \text{At} \text{ or } H = H_0 \lor H_1)$

Fig. 4. The sequent calculus RG3i.

such a rule the premise and the conclusion have the same label. The rule $\rightarrow R_2$ has an unblocked sequent as premise while the conclusion can be either a blocked or an unblocked sequent. Note that, when rule $\rightarrow R_2$ is applied, we have that $A \notin \Gamma$; indeed, by (E5), $A \in \Gamma$ implies $E(A, [\Gamma \vdash A \rightarrow B]) = T$.

As we discuss in Section 9, rules $S_b$ and $S_u$ (where S stands for successors) generate the successors of a world in counter-model construction. The conclusion of the rule $S_b$ is a blocked sequent of the form $\sigma = [\Gamma \vdash H_0 \lor H_1]$ where $H_0 \lor H_1$ is the main formula of the rule application. The side condition states that the rule can be applied only if $\text{act}^\vee(\sigma) \neq \emptyset$ (namely, $E(\sigma, H_0) \neq F$ or $E(\sigma, H_1) \neq F$). If this holds, the rule requires a premise $[\Gamma \vdash H_i]$ for every $H_i \in \text{act}^\vee(\sigma)$; thus, $S_b$ has one or two premises.
The conclusion of the rule \( S_u \) is an unblocked sequent of the form \( \sigma = [\Gamma \Rightarrow \Gamma^u] \), where \( H \) can be either \( \bot \) or a propositional variable not occurring in \( \Gamma^u \) or a disjunction. The rule can be applied only if \( \text{act}^{-}(\sigma) \neq \emptyset \) or \( \text{act}^{+}(\sigma) \neq \emptyset \); the rule has a premise of the kind \( [\Gamma \Rightarrow \Gamma^u] \rightarrow A \) for every \( A \in \text{act}^{-}(\sigma) \) and a premise \( [\Gamma \Rightarrow \Gamma^u] \rightarrow H_i \) for every \( H_i \in \text{act}^{+}(\sigma) \). By the side condition, \( S_u \) has at least one premise.

In Section 9 we prove that RG3i is a sound refutation calculus in the following sense:

**Theorem 6.1 (Soundness of RG3i).** If there exists an RG3i-derivation of an unblocked sequent \( \sigma^n \), then \( \sigma^n \) is refutable. □

In the next section we describe a terminating proof-search procedure \( F \) that, given a sequent \( \sigma \), returns either a G3i-derivation of \( \sigma \) or an RG3i-derivation of \( \sigma^n \). From the soundness of G3i and RG3i, the correctness of \( F \) follows (see Theorem 8.11).

### 7. THE PROOF-SEARCH PROCEDURE

The proof-search procedure is described by the function \( F \) displayed below. This function takes as input an unlabelled sequent \( \sigma \) and a label \( l \in \{b, u\} \) and returns either a G3i-tree or an RG3i-tree.

The proof-search for \( \sigma \) is performed by the invocation \( F(u, \sigma) \). The function \( F \) searches for a proof or a refutation of \( \sigma \) by applying backward the rules of G3i. In the invocation of \( F(l, \sigma) \), the argument \( l \) determines the rules that can be backward applied to the sequent \( \sigma \):

- if \( l = b \) only right rules of G3i can be applied to \( \sigma \) (b-phase);
- if \( l = u \) any rule of G3i can be applied to \( \sigma \) (u-phase).

Let \( F(l', \sigma') \) be a recursive invocation occurring during the execution of \( F(l, \sigma) \), then \( \sigma' \) is a premise of some rule of G3i having \( \sigma \) as conclusion.

The function \( F \) uses a constructor \( \text{Build} \) on \( C \)-trees, where \( C \in \{\text{G3i, RG3i}\} \), which glues together a set of \( C \)-trees by means of a rule of \( C \). Let \( \{\pi_1, \ldots, \pi_n\} \) be a set of \( C \)-trees, where \( \pi_i = (T_i, s_i, r_i) \); we assume without loss of generality that the \( T_i \)'s are pairwise disjoint. Let \( \sigma \) be a sequent and let \( R \) be a rule of \( C \), we denote with \( \text{Build}(C, \sigma, \{\pi_1, \ldots, \pi_n\}, R) \) the structure \( \pi = (T, s, r) \) done as follows:

- let \( t \) be a node not occurring in \( T_1, \ldots, T_n \), \( T \) is the tree having \( t \) as root and \( T_1, \ldots, T_n \) as its immediate subtrees, that is \( \text{children}(t) = \{\text{root}(T_1), \ldots, \text{root}(T_n)\} \);
- \( s(t) = \sigma \) and \( r(t) = R \);
- for every \( i \in \{1, \ldots, n\} \) and for every \( a \in T_i \), \( s(a) = s_i(a) \) and \( r(a) = r_i(a) \).

We notice that, if

\[
\frac{s(\text{root}(T_1)) \ldots s(\text{root}(T_n))}{\sigma} R
\]

is an instance of \( R \), then \( \text{Build}(C, \sigma, \{\pi_1, \ldots, \pi_n\}, R) \) is the \( C \)-tree having root sequent \( \sigma \), root rule \( R \) and \( \pi_1, \ldots, \pi_n \) as immediate subtrees.

The function \( F \) performs the following steps:

- In the recursive invocation at line 4 we have \( \sigma = [A \land B, \Gamma' \Rightarrow H] \), where \( A \land B \notin \Gamma' \), and \( \sigma' = [A, B, \Gamma' \Rightarrow H] \). This corresponds to the application of the rule \( \land L \) of G3i to \( \sigma \) with \( A \land B \) as main formula.
- The recursive invocations at lines 8 and 10 correspond respectively to the construction of the first and the second premise of the rule \( \lor L \) of G3i with \( A_0 \lor A_1 \) as main formula.
The recursive invocations at lines 14 and 15 correspond to the construction of the premise of the rules \( \rightarrow R_1 \) and \( \rightarrow R_2 \) of G3i respectively, with \( A \rightarrow B \) as main formula.

The recursive invocations at lines 19 and 21 correspond to the construction of the two premises of the rule \( \land R \) of G3i with \( H_0 \land H_1 \) as main formula.

The recursive invocation at line 28 corresponds to the construction of the premise of the rule \( \lor R_i \) of G3i with \( H_0 \lor H_1 \) as main formula.

The recursive invocations at lines 34 and 36 correspond to the construction of the two premises of the rule \( \rightarrow L \) of G3i with \( A \rightarrow B \) as main formula.

It is easy to check that properties stated at lines 24, 32 and 40 hold. In Section 8.2 we show that performing the proof-search for a sequent \( \sigma_0 \), whenever in a recursive invocation the return instruction at line 41 is executed, it holds that \( l = u \) and \( \sigma' \) is irreducible.

In the rest of this section we provide some examples of derivations generated by the above proof-search procedure using the evaluation function \( \hat{E} \) described in Section 3.1. Termination and correctness of \( F \) are discussed later in Section 8.

**Example 7.1.** Let us consider the formula \( W = (((p \rightarrow q) \rightarrow p) \rightarrow q) \rightarrow q \), which is a variant of Peirce Law [Chagrov and Zakharyaschev 1997]. This principle is intuitionistically valid. The following is the G3i-derivation of \( \lceil \rightarrow W \rceil \) generated by \( F(u, \lceil \rightarrow W \rceil) \). Sequent are indexed by integers; we denote with \( \sigma_i \) the sequent with index \( i \) and with \( \pi_i \) the subderivation with root \( \sigma_i \). When ambiguities can arise, we underline the main formula of a rule application.

We use the following abbreviations:

\[
W = A \rightarrow q \quad A = (B \rightarrow p) \rightarrow q \quad B = (p \rightarrow q) \rightarrow p
\]

\[
\begin{align*}
[p, B, A \Rightarrow p]_8 & \quad \text{Id} \\
[p, B, A \Rightarrow B \rightarrow p]_7 & \quad \rightarrow R_1 \\
[p, B, A \Rightarrow q]_9 & \quad \text{Id} \\
[B, A \Rightarrow p \rightarrow q]_5 & \quad \rightarrow R_2 \\
[p, A \Rightarrow p]_{10} & \quad \text{Id} \\
[B, A \Rightarrow p]_4 & \quad \rightarrow R_2 \\
[A \Rightarrow ((p \rightarrow q) \rightarrow p) \rightarrow p]_3 & \quad \rightarrow R_2 \\
[q \Rightarrow q]_{11} & \quad \text{Id} \\
[A \Rightarrow q]_2 & \quad \rightarrow R_2 \\
[\Rightarrow (((p \rightarrow q) \rightarrow p) \rightarrow q) \rightarrow q]_1 & \quad \rightarrow R_2
\end{align*}
\]

The main formulas of \( \rightarrow L \) applications are \( A \) and \( B \): \( B \) is the main formula of the \( \rightarrow L \) application with conclusion \( \sigma_4 \), while \( A \) is the main formula of the \( \rightarrow L \) applications with conclusions \( \sigma_2 \) and \( \sigma_6 \). Let us discuss the most relevant steps in the execution of \( F(u, \lceil \rightarrow W \rceil) \).

(1) The application of the rule \( \rightarrow L \) with conclusion \( \sigma_2 \) is performed by the execution of \( F(u, \sigma_2) \). In this case \( \sigma_2 = [A \Rightarrow q] \) and \( A = (B \rightarrow p) \rightarrow q \) is the main formula of the \( \rightarrow L \) application. The G3i-derivation \( \pi_3 \) is returned by the recursive invocation \( F(b, \sigma_3) \) at line 36 of \( F \), while the G3i-derivation \( \pi_{11} \) is returned by the recursive

---

1The derivations and their \( \texttt{HypX} \) rendering are generated with g3i4ed, an implementation of our proof-search procedure available at [http://www.distu.uninsubria.it/~ferram/](http://www.distu.uninsubria.it/~ferram/).
invocation $F(u, \sigma_1)$ at line 34 of $F$. These recursive invocations are executed since
$\text{act}^-(\sigma_2) = \{B \rightarrow p\}$; indeed $R(B \rightarrow p, \sigma_2) = B \rightarrow p$ which is not a local formula
and hence $\mathcal{E}(B \rightarrow p, \sigma_2) = X$.

(2) The application of the rule $\rightarrow L$ with conclusion $\sigma_4$ is performed by the execution
of $F(u, \sigma_4)$. In this case $\sigma_4 = \{B, A \rightarrow p\}$ and $B = (p \rightarrow q) \rightarrow p$ is the main formula
of the application of $\rightarrow L$. Indeed, $\text{act}^{-}(\sigma_{4}) = \{p \rightarrow q\}$ since:

$$
\tilde{E}(p \rightarrow q, \sigma_{4}) = \begin{cases} 
\mathcal{R}^{L}(p \rightarrow q, \sigma_{4}) = p \rightarrow q \\
\mathcal{R}^{R}(p \rightarrow q, \sigma_{4}) = p \rightarrow q \\
\mathcal{R}(p \rightarrow q, \sigma_{4}) = p \rightarrow q \quad \text{and} \quad p \rightarrow q \text{is not a local formula}
\end{cases}
$$

$$
\tilde{E}(B \rightarrow p, \sigma_{4}) = \begin{cases} 
\mathcal{R}^{L}(B \rightarrow p, \sigma_{4}) = p \\
\mathcal{R}^{R}(p, \sigma_{4}) = \bot \\
\mathcal{R}(B \rightarrow p, \sigma_{4}) = \bot \quad \text{and} \quad \bot \text{is a local formula}
\end{cases}
$$

$\pi_{5}$ is the result of $\mathcal{F}(b, \sigma_{5})$ invoked at line 36 of $\mathcal{F}$, while $\pi_{10}$ is the result of $\mathcal{F}(u, \sigma_{10})$ invoked at line 34 of $\mathcal{F}$.

(3) The application of the rule $\rightarrow L$ with conclusion $\sigma_{6}$ is performed by the execution of $\mathcal{F}(u, \sigma_{6})$. In this case $\sigma_{6} = \{b, B, A \Rightarrow q\}$ and $A = (B \rightarrow p) \Rightarrow q$ is the main formula of the application of $\rightarrow L$. Indeed, $\text{act}^{-}(\sigma_{6}) = \{B \rightarrow p\}$ since:

$$
\tilde{E}(B \rightarrow p, \sigma_{6}) = \top \quad \text{and} \quad \mathcal{R}^{L}(B \rightarrow p, \sigma_{6}) = \top
$$

$$
\tilde{E}(p \rightarrow q, \sigma_{6}) = \begin{cases} 
\mathcal{R}^{L}(p \rightarrow q, \sigma_{6}) = q \\
\mathcal{R}^{R}(q, \sigma_{6}) = \bot \\
\mathcal{R}(p \rightarrow q, \sigma_{6}) = \bot
\end{cases}
$$

$\pi_{7}$ is the result of $\mathcal{F}(b, \sigma_{7})$ invoked at line 36 of $\mathcal{F}$, while $\pi_{9}$ is the result of $\mathcal{F}(u, \sigma_{9})$ invoked at line 34 of $\mathcal{F}$.

We remark that our mechanism avoids some applications of $\rightarrow L$ and this prevents the generation of infinite branches. As an example, in Point (2) $A$ cannot be selected as main formula of an $\rightarrow L$ application to $\sigma_{4}$ since $\tilde{E}(B \rightarrow p, \sigma_{4}) = F$. Similarly, in Point (3) we cannot apply $\rightarrow L$ to $\sigma_{6}$ with main formula $B$ since $\tilde{E}(p \rightarrow q, \sigma_{6}) = F$.

**Example 7.2.** Let us consider the Scott principle [Chagrov and Zakharyaschev 1997], that is the formula $S = (((\neg p \rightarrow p) \rightarrow (\neg p \lor p)) \rightarrow (\neg p \lor \neg p))$. We recall that $\neg Z = Z \rightarrow \bot$. Scott Principle is not intuitionistically valid and $\mathcal{F}(u, [\Rightarrow S])$ returns the following RG3i-derivation of $[\equiv S]$.

$$
A = (\neg\neg p \rightarrow p) \rightarrow (\neg p \lor p)
$$

$$
\begin{array}{c}
[p, \neg\neg p, \equiv, \bot]_{8} \quad \text{Irr} \\
[p \lor p, \neg\neg p, \equiv, \bot]_{7} \quad \text{L} \\
[p, \neg p, A \equiv, \bot]_{6} \quad \text{S}_{u} \\
[p, \neg p]_{5} \quad \text{R}_{2} \\
[p, A]_{4} \quad \text{R}_{2} \\
[p \lor p, \equiv, \bot]_{3} \quad \text{S}_{u} \\
[
eg\neg p \lor \neg p]_{2} \\
\equiv (\neg\neg p \rightarrow p) \rightarrow (\neg p \lor p) \\
\end{array}
$$

$$
\begin{array}{c}
[\neg p, \equiv, \bot]_{12} \quad \text{Irr} \\
[\neg p \lor p, \equiv, \bot]_{11} \quad \text{L} \\
[p, A, \equiv, \bot]_{10} \quad \text{S}_{u} \\
[p]_{9} \quad \text{R}_{2} \\
[p, A]_{8} \quad \text{R}_{2} \\
[p]_{7} \quad \text{L} \\
\end{array}
$$

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
transition between states. Formally, a recursive calls

— The application of \( S_u \) with conclusion \( \sigma_2 = [A_u \Rightarrow \neg p \lor \neg p] \) has three premises, indeed

\[
\hat{E}(\neg p \rightarrow p, \sigma_2) = \hat{E}(\neg p, \sigma_2) = \hat{E}(\neg p, \sigma_2) = X
\]

Hence \( \text{act}^+(\sigma_2) = \{\neg p \rightarrow p\} \) and \( \text{act}^\forall(\sigma_2) = \{\neg p, \neg p\} \).

— The application of \( S_u \) with conclusion \( \sigma_4 = [\neg p, A_u \Rightarrow p] \) has just one premise, indeed

\[
\hat{E}(\neg p \rightarrow p, \sigma_4) = F
\]

\[
\hat{E}(\neg p, \sigma_4) = X \quad (\neg p \text{ is the antecedent of } \neg p = \neg p \rightarrow \bot)
\]

Hence \( \text{act}^+(\sigma_4) = \{-p\} \) while \( \text{act}^\forall(\sigma_4) = \emptyset \).

Finally, \( \sigma_8 = [p, \neg p \Rightarrow \bot] \) is irreducible, indeed \( \hat{E}(\neg p, \sigma_8) = F \). Similarly, \( \sigma_{12} \) and \( \sigma_{16} \) are irreducible.

8. TERMINATION AND CORRECTNESS OF THE PROOF-SEARCH PROCEDURE

In this section we prove that \( \mathcal{F} \) is terminating and correct. To prove termination we show that every possible sequence of recursive calls of \( \mathcal{F} \) is finite. As for correctness, we show that, for every sequent \( \sigma \), the invocation \( \mathcal{F}(u, \sigma) \) returns either a G3i-derivation of \( \sigma \) or an RG3i-derivation of \( \sigma^u \). From this and the soundness of RG3i (Theorem 6.1), it follows that \( \mathcal{F}(u, \sigma) \) returns a G3i-derivation of \( \sigma \) if \( \sigma \) is intuitionistically valid.

To discuss the above properties of \( \mathcal{F} \) we need to formalize the notion of chain of recursive calls (CRC for short) generated by an execution of \( \mathcal{F} \). We call state of the invocation \( \mathcal{F}(l, \sigma) \) the input pair \((l, \sigma)\) and we describe a recursive invocation of \( \mathcal{F} \) as a transition between states. Formally, a CRC-transition is an expression of the kind

\[
(l, \sigma = [\Gamma \Rightarrow H]) \stackrel{K}{\rightarrow} (l', \sigma' = [\Gamma' \Rightarrow H'])
\]

where:

— \((l, \sigma)\) is the state of the current invocation of \( \mathcal{F} \);
— \((l', \sigma')\) is the state corresponding to the invocation \( \mathcal{F}(l', \sigma') \) directly called in \( \mathcal{F}(l, \sigma) \);
— \(K\) is the main formula of the rule application determining the recursive invocation \( \mathcal{F}(l', \sigma') \).

The possible CRC-transitions are described in Table I. For every CRC-transition, we indicate the line of \( \mathcal{F} \) where the related recursive invocation occurs and the evaluation condition determining its execution, if any. We point out some important properties of CRC-transitions.

— If \( H \in \Gamma \), the execution of \( \mathcal{F}(l, [\Gamma \Rightarrow H]) \) terminates without performing any recursive invocation (see line 2 of \( \mathcal{F} \)); for this reason we can assume that in every CRC-transition \( H \notin \Gamma \).
— There are two possible forms of the transition corresponding to the application of the rule \( \rightarrow R_2 \) that differ in the label of left state. We notice that (T5\( \text{in} \)) is the only transition that can turn the label \( b \) into \( u \); it has a crucial role in the proof of termination.
— For all CRC-transitions but (T10), it holds that \(|\sigma| > |\sigma'|\).

A CRC \( \delta(l, \sigma) \) for \((l, \sigma)\) is a sequence of CRC-transitions starting from \((l, \sigma)\). Formally,

\[
\delta(l, \sigma) = (l_0, \sigma_0) \stackrel{K_0}{\rightarrow} (l_1, \sigma_1) \ldots \stackrel{K_n}{\rightarrow} (l_n, \sigma_n) \ldots
\]

where:

— \((l_0, \sigma_0) = (l, \sigma)\);
— for every \( i \geq 0 \), \((l_i, \sigma_i) \stackrel{K}{\rightarrow} (l_{i+1}, \sigma_{i+1})\) is a CRC-transition.

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
In general, there exist different possible CRCs for $(l, \sigma)$. The sequence of sequents $\sigma_0, \sigma_1, \ldots, \sigma_n, \ldots$ in $\delta(l, \sigma)$ corresponds to a branch of a G3i-tree. Since G3i has the subformula property, $\delta(l, \sigma)$ has the subformula property: every formula occurring in $\delta(l, \sigma)$ belongs to $\text{St}(\sigma)$.

The length of $\delta(l, \sigma)$, denoted by $|\delta(l, \sigma)|$, is the number of states occurring in $\delta(l, \sigma)$. If the function $\mathcal{F}(l, \sigma)$ terminates without performing recursive invocations, then $\delta(l, \sigma) = (l, \sigma)$ and $|\delta(l, \sigma)| = 1$.

8.1. Termination

In this section we prove that every CRC $\delta(u, \sigma)$ for $(u, \sigma)$ has finite length. First of all we prove that, if a formula $A$ belongs to the left-hand side of a sequent $\sigma'$ in $\delta(l, \sigma)$, then $\mathcal{E}^l(A, \sigma) = T$ for every sequent $\sigma'$ following $\sigma'$.

**Lemma 8.1.** Let $\delta(l, \sigma)$ be a CRC for $(l, \sigma)$ and let $(l_h, \sigma_h)$ be a state in $\delta(l, \sigma)$. If $A \in \bigcup_{0 \leq i \leq l} \Gamma_i$, then $\mathcal{E}^l(A, \sigma_h) = T$.

**Proof.** Let $\Delta_h = \bigcup_{0 \leq i \leq l} \Gamma_i$ and $A \in \Delta_h$. The proof goes by induction on the structure of $A$. If $A \in \Gamma_h$, by (E5) we immediately get $\mathcal{E}^l(A, \sigma_h) = T$. If $A \notin \Gamma_h$, then $\delta(l, \sigma)$ contains a CRC-transition $\tau$ of the form

$$\tau = (l_j, \sigma_j = [\Gamma_j \Rightarrow H_j]) \xrightarrow{A_j} (l_{j+1}, \sigma_{j+1} = [\Gamma_{j+1} \Rightarrow H_{j+1}])$$

such that $0 \leq j < h$, $A \in \Gamma_j$, and $A \notin \Gamma_{j+1}$; thus, $\tau$ is one of the CRC-transitions (T1), (T2), (T3) and (T9), corresponding to the application of a left rule of G3i. Note that $\Gamma_{j+1} \subseteq \Delta_h$.

— If $\tau$ is the CRC-transition (T1), then $A = B \land C$ and

$$\tau = (u, \sigma_j = [B \land C, \Gamma' \Rightarrow H]) \xrightarrow{B \land C} (u, \sigma_{j+1} = [B, C, \Gamma' \Rightarrow H])$$

Thus, $B \in \Delta_h$ and $C \in \Delta_h$. By induction hypothesis, $\mathcal{E}^l(B, \sigma_h) = \mathcal{E}^l(C, \sigma_h) = T$; by (E6), we get $\mathcal{E}^l(B \land C, \sigma_h) = T$.

— If $\tau$ is (T2) or (T3), then $A = A_0 \lor A_1$ and

$$\tau = (u, \sigma_j = [A_0 \lor A_1, \Gamma' \Rightarrow H]) \xrightarrow{A_0 \lor A_1} (u, \sigma_{j+1} = [A_i, \Gamma' \Rightarrow H])$$

<table>
<thead>
<tr>
<th>Possible CRC-transitions $(l, \sigma = [\Gamma \Rightarrow H]) \xrightarrow{\Delta} (l', \sigma')$ where $H \notin \Gamma$</th>
<th>Evaluation condition</th>
<th>Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>(T1) $(u, [A \land B, \Gamma' \Rightarrow H]) \xrightarrow{A \land B} (u, [A, B, \Gamma' \Rightarrow H])$</td>
<td>$\mathcal{E}(A, \sigma) = T$</td>
<td>4</td>
</tr>
<tr>
<td>(T2) $(u, [A_0 \lor A_1, \Gamma' \Rightarrow H]) \xrightarrow{A_0 \lor A_1} (u, [A_0, \Gamma' \Rightarrow H])$</td>
<td>$\mathcal{E}(A, \sigma) = T$</td>
<td>8</td>
</tr>
<tr>
<td>(T3) $(u, [A_0 \lor A_1, \Gamma' \Rightarrow H]) \xrightarrow{A_0 \lor A_1} (u, [A_1, \Gamma' \Rightarrow H])$</td>
<td>$\mathcal{E}(A, \sigma) = T$</td>
<td>10</td>
</tr>
<tr>
<td>(T4) $(l, [\Gamma \Rightarrow A \rightarrow B]) \xrightarrow{\Gamma} (l, [\Gamma \Rightarrow B])$</td>
<td>$\mathcal{E}(A, \sigma) = T$</td>
<td>14</td>
</tr>
<tr>
<td>(T5$^{+\omega}$) $(u, [A \land B, \Gamma \Rightarrow A]) \xrightarrow{A \land B} (u, [A, \Gamma \Rightarrow A])$</td>
<td>$\mathcal{E}(A, \sigma) = T$</td>
<td>15</td>
</tr>
<tr>
<td>(T6) $(l, [\Gamma \Rightarrow H_0 \land H_{i+1}]) \xrightarrow{H_0 \land H_{i+1}} (l, [\Gamma \Rightarrow H_{i+1}])$</td>
<td>$\mathcal{E}(H_0, \sigma) = F$</td>
<td>19</td>
</tr>
<tr>
<td>(T7) $(l, [\Gamma \Rightarrow H_0 \land H_0]) \xrightarrow{H_0 \land H_0} (l, [\Gamma \Rightarrow H_0])$</td>
<td>$\mathcal{E}(H_0, \sigma) = F$</td>
<td>21</td>
</tr>
<tr>
<td>(T8) $(l, [\Gamma \Rightarrow H_0 \lor H_{i+1}]) \xrightarrow{H_0 \lor H_{i+1}} (l, [\Gamma \Rightarrow H_{i+1}])$</td>
<td>$\mathcal{E}(H_0, \sigma) = F$</td>
<td>28</td>
</tr>
<tr>
<td>(T9) $(u, [A \land B, \Gamma \Rightarrow H]) \xrightarrow{A \land B} (u, [B, \Gamma \Rightarrow H])$</td>
<td>$\mathcal{E}(A, \sigma) = F$</td>
<td>34</td>
</tr>
<tr>
<td>(T10) $(u, [A \land B, \Gamma \Rightarrow H]) \xrightarrow{A \land B} (u, [A \land B, \Gamma \Rightarrow H])$</td>
<td>$\mathcal{E}(A, \sigma) = F$</td>
<td>36</td>
</tr>
</tbody>
</table>

Table I. CRC-transitions
with \( i \in \{0, 1\} \). Since \( A_i \in \Delta_h \), by induction hypothesis we get \( \mathcal{E}^L(A_i, \sigma_h) = T \), which implies, by (\( \mathcal{E}7 \)), \( \mathcal{E}^L(A_0 \lor A_1, \sigma_h) = T \).

If \( \tau \) is (T9), then \( A = B \rightarrow C \) and

\[
\tau = (u, \sigma_j = [B \rightarrow C, \Gamma' \Rightarrow H]^{\mathcal{L}}B \rightarrow C(u, \sigma_{j+1} = [C, \Gamma' \Rightarrow H])
\]

Since \( C \in \Delta_h \), by induction hypothesis \( \mathcal{E}^L(C, \sigma_h) = T \), which implies, by (\( \mathcal{E}8 \)), \( \mathcal{E}^L(B \rightarrow C, \sigma_h) = T \). □

The following lemma provides a finite upper bound on the number of transitions of the kind (T5\textsubscript{bu}) that can occur in a CRC \( \delta(l, \sigma) \).

**Lemma 8.2.** Let \( \delta(l, \sigma) \) be a CRC for \((l, \sigma)\). Then, the number of applications of (T5\textsubscript{bu}) in \( \delta(l, \sigma) \) is at most \(|\sigma|\).

**Proof.** Firstly, we show that \( \delta(l, \sigma) \) cannot contain two distinct applications of (T5\textsubscript{bu}) with the same main formula. Indeed, let us assume that (T5\textsubscript{bu}) is applied to a state \((b, \sigma_i)\) with main formula \( A \rightarrow B \) and let \((b, \sigma_j)\) be a state in \( \delta(l, \sigma) \) such that \( i < j \). Since \( A \in \Gamma_i+1 \) and \( i + 1 \leq j \), by Lemma 8.1 it follows that \( \mathcal{E}^L(A, \sigma_j) = T \), which implies, by (\( \mathcal{E}3 \)), \( \mathcal{E}(A, \sigma_j) = T \). Hence (T5\textsubscript{bu}) cannot be applied to \((b, \sigma_j)\) with main formula \( A \rightarrow B \). Since CRCs fulfill the subformula property, the possible main formulas \( A \rightarrow B \) belong to \( SF(\sigma) \), hence they are at most \(|\sigma|\). Thus, we cannot have more than \(|\sigma|\) applications of (T5\textsubscript{bu}) in \( \delta(l, \sigma) \). □

Since (T10) turns label \( u \) into \( b \) and (T5\textsubscript{bu}) is the only transition which turns a label \( b \) into \( u \), between two successive applications of (T10) an application of (T5\textsubscript{bu}) must occur. This implies:

**Lemma 8.3.** Let \( \delta(l, \sigma) \) be a CRC for \((l, \sigma)\). Then, the number of applications of (T10) in \( \delta(l, \sigma) \) is at most \(|\sigma| + 1 \). □

**Theorem 8.4.** Let \( \delta(l, \sigma) \) be a CRC for \((l, \sigma)\). Then, the length of \( \delta(l, \sigma) \) is \( O(|\sigma|^2) \).

**Proof.** By Lemma 8.3 \( \delta(l, \sigma) \) contains \(|\sigma| + 1 \) applications of transition (T10) at most. For every application \((u, \sigma')^{\mathcal{L}}B(b, \sigma'') \) of (T10), it holds that \(|\sigma''| \leq |\sigma'| + |\sigma|\). Thus the applications of (T10) introduce in \( \delta(l, \sigma) \) at most \(|\sigma| \cdot (|\sigma| + 1)\) connectives.

Since, for every transition \((l', \sigma')^{\mathcal{L}}B(l'', \sigma'') \) different from (T10) \(|\sigma''| < |\sigma'|\), the length of \( \delta(l, \sigma) \) is \( O(|\sigma|^2) \). □

By the above theorem, we get:

**Corollary 8.5.** For every sequent \( \sigma \) and \( l \in \{b, u\} \), the execution of \( \mathcal{F}(l, \sigma) \) terminates. □

8.2. Correctness

To prove the correctness of \( \mathcal{F} \) we show that, for every sequent \( \sigma \), \( \mathcal{F}(u, \sigma) \) returns either a \( G3i \)-derivation of \( \sigma \) or an \( RG3i \)-derivation of \( \sigma^a \).

Let \((l, \sigma = [\Gamma \Rightarrow H])\) be a state of a CRC; we say that \((l, \sigma)\) has the property (INV) if:

(INV) \( l = b \) implies \( \mathcal{E}^L(H, \sigma) \neq F \).

We show that CRC-transitions preserve (INV).

**Lemma 8.6.** Let

\[
\tau = (l, \sigma = [\Gamma \Rightarrow H])^{\mathcal{L}}B(l', \sigma' = [\Gamma' \Rightarrow H'])
\]

be a CRC-transition. If \((l, \sigma)\) satisfies (INV), then \((l', \sigma')\) satisfies (INV).
PROOF. If \( l' = u \), we immediately have that \((l', \sigma')\) satisfies (INV).

Let \( l' = b \) and let us assume that \((l, \sigma)\) satisfies (INV); by a case analysis on the form of \( \tau \) (see Table I), we show that \( \mathcal{E}^L(H', \sigma') \neq F \). We point out that \( H \not\in \Gamma \).

- If \( \tau \) is (T4), then
  \[
  \tau = (b, \sigma = [\Gamma \Rightarrow A \rightarrow B]) \xrightarrow{A \rightarrow B} (b, \sigma' = [\Gamma \Rightarrow B])
  \]
  where \( \mathcal{E}(A, \sigma) = T \). By (E3), it follows that \( \mathcal{E}^L(A, \sigma) = T \). Moreover, we know that \( A \rightarrow B \not\in \Gamma \) and \( \mathcal{E}^L(A \rightarrow B, \sigma) \neq F \) (indeed, \( (b, \sigma) \) satisfies (INV)). By (E11) we get \( \mathcal{E}^L(B, \sigma) \neq F \). Since \( \sigma \) and \( \sigma' \) have the same left-hand side, we have \( \mathcal{E}^L(B, \sigma) = \mathcal{E}^L(B, \sigma') \), hence \( \mathcal{E}^L(B, \sigma') \neq F \).

- If \( \tau \) is (T6) or (T7), then
  \[
  \tau = (b, \sigma = [\Gamma \Rightarrow H_0 \land H_1]) \xrightarrow{H_0 \land H_1} (b, \sigma' = [\Gamma \Rightarrow H_i])
  \]
  where \( i \in \{0, 1\} \). Since \( H_0 \land H_1 \not\in \Gamma \) and \( \mathcal{E}^L(H_0 \land H_1, \sigma) \neq F \), by (E9) we get \( \mathcal{E}^L(H_i, \sigma) \neq F \), which implies \( \mathcal{E}^L(H_i, \sigma') \neq F \).

- If \( \tau \) is (T10), then
  \[
  \tau = (u, \sigma = [A \rightarrow B, \Gamma \Rightarrow H]) \xrightarrow{A \rightarrow B} (u, \sigma' = [A \rightarrow B, \Gamma \Rightarrow A])
  \]
  where \( \mathcal{E}(A, \sigma) \neq F \). By (E4), \( \mathcal{E}^L(A, \sigma) \neq F \), hence \( \mathcal{E}^L(A, \sigma') \neq F \).

Since \( \mathcal{E}^L(H', \sigma') \neq F \), we conclude that \((b, \sigma')\) satisfies (INV). \( \square \)

By the previous lemma, it follows that (INV) is an invariant property for a CRC, namely:

**Lemma 8.7.** Let \( \delta(l, \sigma) \) be a CRC for \((l, \sigma)\) and let us assume that \((l, \sigma)\) satisfies (INV). Then, for every state \((l', \sigma')\) in \( \delta(l, \sigma) \), \((l', \sigma')\) satisfies (INV). \( \square \)

Now, we prove the correctness of \( \mathcal{F} \) when no recursive invocation occurs during the execution of \( \mathcal{F}(l, \sigma) \).

**Lemma 8.8.** Let \( \sigma \) be a sequent and \( l \in \{u, b\} \) such that \((l, \sigma)\) satisfies (INV). If \( \mathcal{F}(l, \sigma) \) terminates without performing any recursive invocation, then it returns either a \( \mathcal{G}3i \)-derivation of \( \sigma \) or an \( \mathcal{R}G3i \)-derivation of \( \sigma' \).

**Proof.** It is immediate to check that, if one of the return instructions at lines 1 or 2 is executed, then the assertion holds. Let us assume that none of these return instructions have been executed. This means that the conditions at lines 1 and 2 are false. Moreover, since no recursive call is performed, the conditions at lines 3, 7, 13, 18 and 25 are false and the return instruction at line 41 is executed. To prove the assertion we must show that \( l = u \) and \( \sigma' \) is an irreducible sequent. Let \( \sigma = [\Gamma \Rightarrow H] \).

By the above discussion we deduce that the following hold:

1. \( \bot \not\in \Gamma \) and \( H \not\in \Gamma \) (since conditions at lines 1 and 2 are false);
2. \( H = \bot \) or \( H \in \mathcal{V} \) or \( H = H_0 \lor H_1 \) (since conditions at lines 13 and 18 are false);
3. \( l = u \) implies \( \sigma = [\Gamma^=, \Gamma^= \Rightarrow H] \) (since conditions at lines 3 and 7 are false);
4. \( l = b \) or \( \text{act}^- \sigma = \emptyset \) (since condition at line 25 is false);
5. \( \text{act}^\land \sigma = \emptyset \) (since condition at line 25 is false).

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
Now we prove that points (i)–(v) imply \( l = u \). Let us assume, by absurd, that \( l = b \). If \( H = \bot \) or \( H \in V \), by (2) and (i) it follows that \( E^l(H, \sigma) = F \), in contradiction with the hypothesis that \( (l, \sigma) \) satisfies (INV). Hence, by (ii), we deduce that \( H = H_0 \lor H_1 \). By (v), it holds that \( E(H_0, [\Gamma \Rightarrow H_0 \lor H_1]) = F \). Since \( H_0 \lor H_1 \not\in S(H_0) \), by (E1) we get that \( E(H_0, [\Gamma \Rightarrow \bot]) = F \); hence \( E^l(H_0, [\Gamma \Rightarrow \bot]) = F \). Similarly, \( E^l(H_1, [\Gamma \Rightarrow \bot]) = F \). By (i) and (E10), it follows that \( E^l(H_0 \lor H_1, [\Gamma \Rightarrow \bot]) = F \). This implies that \( E^l(H_0 \lor H_1, \sigma) = F \), against the hypothesis that \( (l, \sigma) \) satisfies (INV). Thus, \( l = b \) cannot hold, and this proves that \( l = u \). By (ii), (iv) and (v) we get:

- (vi) \( \sigma = [\Gamma \Rightarrow, \Gamma^{AT} \Rightarrow H] \);
- (vii) \( \text{act}^\leftarrow(\sigma) = \text{act}^\rightarrow(\sigma) = \emptyset \).

By (i), (ii), (vi) and (vii) we conclude that \( \sigma^u = [\Gamma \Rightarrow, \Gamma^{AT} \Rightarrow H] \) is irreducible, hence the value returned at line 41 is an RG3i-derivation of \( \sigma^u \). □

Note that the crucial point of the above proof is to show that, if line 41 is reached, then \( l = u \); to guarantee this, property (INV) is needed.

Given a state \((l, \sigma)\), the CRC-depth of \((l, \sigma)\) is the length of the longest CRC for \((l, \sigma)\).

**THEOREM 8.9.** Let \( \sigma \) be a sequent and \( l \in \{u, b\} \) such that \((l, \sigma)\) satisfies (INV). Then, \( F(l, \sigma) \) returns either a G3i-derivation of \( \sigma \) or an RG3i-derivation of \( \sigma^u \).

**PROOF.** The proof goes by induction on the CRC-depth \( k \) of \((l, \sigma)\). If \( k = 1 \) then \( F(l, \sigma) \) does not perform any recursive invocation and the assertion follows by Lemma 8.8.

Let \( k > 1 \) and \( \sigma = [\Gamma \Rightarrow H] \). We have to check that, whenever the auxiliary function \( \text{Build} \) is called, the arguments are correctly instantiated. Note that, when a recursive call \( F(l', \sigma') \) is executed, the CRC-depth of \((l', \sigma')\) is less than \( k \). Moreover, by Lemma 8.7 \((l', \sigma')\) satisfies (INV). Thus we can apply the induction hypothesis to \( F(l', \sigma') \).

Let us assume that one of the recursive calls at lines 8 and 10 is executed. In this case \( l = u \) and \( \sigma = [A_0 \lor A_1, \Gamma' \Rightarrow H] \). By induction hypothesis, the value \( \pi_0 \) returned at line 8 is either a G3i-derivation of \([A_0, \Gamma' \Rightarrow H]\) or an RG3i-derivation of \([A_0, \Gamma'^u \Rightarrow H]\). Similarly, \( \pi_1 \) is either a G3i-derivation of \([A_1, \Gamma' \Rightarrow H]\) or an RG3i-derivation of \([A_1, \Gamma'^u \Rightarrow H]\) (see line 10).

- If \( \pi_0 \) is an RG3i-derivation of \([A_0, \Gamma'^u \Rightarrow H]\), the return instruction at line 9 is executed and \( \text{Build}(\text{RG3i}, \sigma^u, \{\pi_0\}, \{L_0\}) \) returns an RG3i-derivation of \( \sigma^u \).
- Similarly, if \( \pi_1 \) is an RG3i-derivation of \([A_1, \Gamma'^u \Rightarrow H]\), then an RG3i-derivation of \( \sigma^u \) is returned at line 11.
- If both \( \pi_0 \) and \( \pi_1 \) are G3i-derivations, then \( \text{Build}(\text{G3i}, \sigma, \{\pi_0, \pi_1\}, \{L\}) \) returns a G3i-derivation of \( \sigma \) (line 12).

Now, let us consider the case where the return instruction at line 39 is executed. In this case, the condition at line 25 holds, hence

\[
(l = u \text{ and } \text{act}^\leftarrow(\sigma) \neq \emptyset) \text{ or } \text{act}^\rightarrow(\sigma) \neq \emptyset
\]

(1)

We have two cases.

- Let us assume \( l = b \). By (1), we have that \( \text{act}^\leftarrow(\sigma) \neq \emptyset \). Moreover, for every \( H_i \in \text{act}^\rightarrow(\sigma) \), \( \text{Refs} \) contains an RG3i-derivation of \([\Gamma^u \Rightarrow H_i]\) (namely, the RG3i-derivation \( \pi_i \) added to \( \text{Refs} \) at line 30), hence the value returned at line 39 is an RG3i-derivation of \( \sigma^b \).
- Let \( l = u \). By (1), \( \text{act}^\leftarrow(\sigma) \cap \text{act}^\rightarrow(\sigma) = \emptyset \). Moreover, for every \( H_i \in \text{act}^\rightarrow(\sigma) \), \( \text{Refs} \) contains an RG3i-derivation of \([\Gamma^u \Rightarrow H_i]\) (the RG3i-derivation \( \pi_i \) added to \( \text{Refs} \) at
line 30); for every $A \rightarrow B \in \Gamma$ such that $A \in \text{act}^{-}(\sigma)$, Refs contains an RG3i-derivation of $[\Gamma \vdash_{b} A]$ (the RG3i-derivation $\pi_{2}$ added to Refs at line 38). This implies that the value returned at line 39 is an RG3i-derivation of $\sigma_{u}$.

The other cases can be easily checked. □

Since a state $(u, \sigma)$ trivially meets property (INV), by the above theorem we get:

**Corollary 8.10.** Let $\sigma$ be a sequent. Then $F(u, \sigma)$ returns either a G3i-derivation of $\sigma$ or an RG3i-derivation of $\sigma_{u}$. □

Finally, we get the correctness of $F$:

**Theorem 8.11.** Let $\sigma$ be a sequent. Then $F(u, \sigma)$ returns a G3i-derivation of $\sigma$ iff $\sigma$ is intuitionistically valid.

**Proof.** By Corollary 8.10, $F(u, \sigma)$ returns either a G3i-derivation of $\sigma$ or an RG3i-derivation of $\sigma_{u}$. If $F(u, \sigma)$ returns a G3i-derivation of $\sigma$ then, by the soundness of G3i (Theorem 5.1), $\sigma$ is intuitionistically valid. Conversely, let us assume that $\sigma$ is intuitionistically valid. Then $F(u, \sigma)$ cannot return an RG3i-derivation, otherwise by Theorem 6.1 (Soundness of RG3i) $\sigma$ should be refutable. Hence $F(u, \sigma)$ returns a G3i-derivation of $\sigma$. □

## 9. Soundness of RG3i

In this section we prove the soundness of RG3i stated in Theorem 6.1.

First of all we discuss some properties of RG3i-trees. An RG3i-tree having root sequent $\sigma^{b} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H]$ and only containing blocked sequents has the form:

$$
\sigma^{b} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H_{1}] \quad \ldots \quad \sigma^{b}_{n} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H_{n}]
$$

$$
\sigma^{b} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H]
$$

(PB)

Indeed only the rules $\land R_{i}$, $\rightarrow R_{1}$ and $S_{b}$ can be applied in (PB), hence all the sequents have the same left-hand side. The following holds:

**Lemma 9.1.** Let $\pi$ be an RG3i-tree only containing blocked sequents of the form (PB) having root $\sigma^{b} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H]$ and leaves $\sigma^{b}_{1} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H_{1}], \ldots, \sigma^{b}_{n} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models H_{n}]$. Let $\mathcal{K} = (P, \leq, \rho, V)$ be a Kripke model and $\alpha \in P$ such that:

(H1) $\mathcal{K}, \alpha \nvdash H_{i}$ for every $i \in \{1, \ldots, n\}$;

(H2) $\mathcal{K}, \alpha \models Z$ for every $Z \in \Gamma^{\rightarrow} \cap \text{St}(H)$;

(H3) $V(\alpha) = \Gamma^{\text{At}}$.

Then, $\mathcal{K}, \alpha \nvdash H$.

**Proof.** The proof goes by induction on $\text{depth}(\pi)$. If $\text{depth}(\pi) = 0$, the root $\sigma^{b}$ of $\pi$ coincides with a leaf of $\pi$ and the assertion immediately follows by (H1). Let us assume that $\text{depth}(\pi) > 0$ and let $R$ be the rule applied at the root of $\pi$. Since the premises and the conclusion of $R$ are blocked sequents, $R$ is one of the rules $\land R_{i}$, $\rightarrow R_{1}$ and $S_{b}$. We proceed by cases on $R$.

If $R$ is $\land R_{i}$ with $i \in \{0, 1\}$, then $\pi$ has the form

$$
\pi' \quad \frac{[\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models A_{1}]}{\sigma^{b} = [\Gamma^{\rightarrow}, \Gamma^{\text{At}^{b}} \models A_{0} \land A_{1}]} \land R_{i}
$$

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
By induction hypothesis we get \( K, \alpha \not\vdash A_i \), which implies \( K, \alpha \not\vdash A_0 \land A_1 \).

If \( R \) is \( \rightarrow R_1 \), then \( \pi \) has the form

\[
\frac{\vdots \pi' [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} B] \sigma^b = [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A \rightarrow B]}{R_1}
\]

where \( \mathcal{E}(A, \sigma^b) = T \). By induction hypothesis, it follows that \( K, \alpha \not\vdash B \). To prove that \( K, \alpha \not\vdash A \rightarrow B \), we have to show that \( K, \alpha \vdash A \). Let \( \sigma = [\Gamma \rightarrow \cap \text{Sf}(A), \Gamma \vdash \frac{b}{b} \bot] \). Since

\[
\text{Restr}(\sigma, A) = \text{Restr}(\sigma^b, A),
\]

by (E1) we get \( \mathcal{E}(A, \sigma) = T \) which implies, by (E3), \( \mathcal{E}^L(A, \sigma) = T \). Now we can apply (E1.1) to deduce that

\[
K, \alpha \vdash A; \quad \text{indeed, hypothesis (H2) and (H3) of the lemma immediately imply that} \quad K, \alpha \vdash \sigma.
\]

Thus \( K, \alpha \vdash A \) and \( K, \alpha \not\vdash B \), which implies \( K, \alpha \not\vdash A \rightarrow B \). This concludes the proof in the case where \( R \) is \( \rightarrow R_1 \).

If \( R \) is \( S_b \), we have two cases depending on the number of premises of the rule.

— If \( \pi \) has the form

\[
\vdots \pi_0 \vdots \pi_1 \frac{[\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_0] [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_1]}{\sigma^b = [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_0 \lor A_1]} S_b
\]

by induction hypothesis, we get \( K, \alpha \not\vdash A_0 \) and \( K, \alpha \not\vdash A_1 \). Therefore, \( K, \alpha \not\vdash A_0 \lor A_1 \).

— Let us assume that \( \pi \) has the form

\[
\vdots \pi' \frac{[\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_i]}{\sigma^b = [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_0 \lor A_1]} S_b
\]

where \( i \in \{0, 1\}, \mathcal{E}(A_i, \sigma^b) \neq F \) and \( \mathcal{E}(A_j, \sigma^b) = F \), with \( j = 1 - i \). By induction hypothesis, we get \( K, \alpha \not\vdash A_i \). To prove that \( K, \alpha \not\vdash A_0 \lor A_1 \), we must show that

\[
K, \alpha \vdash \sigma.
\]

Let \( \sigma = [\Gamma \rightarrow \cap \text{Sf}(A_j), \Gamma \vdash \frac{b}{b} \bot] \). By hypothesis (H2) and (H3) of the lemma, we get that \( K, \alpha \vdash \sigma \). Since

\[
\text{Restr}(\sigma, A_j) = \text{Restr}(\sigma^b, A_j),
\]

by (E1) we get \( \mathcal{E}(A_j, \sigma) = F \). Since all the hypothesis of (E1.2) are satisfied, we conclude \( K, \alpha \not\vdash A_j \). \( \square \)

Now, let \( \pi \) be an RG3i-derivation of a sequent \( \sigma^b = [\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} H] \) and let \( \pi' \) be a subtree of \( \pi \) having root \( \sigma^b \) and only containing blocked sequents of the form (PB). We point out that none of the sequents \( \sigma^b_i \) at the leaves of \( \pi' \) can be a leaf of \( \pi \), indeed \( \pi \) is an RG3i-derivation and the initial sequents of RG3i are unblocked sequents. By \( \Pi(\pi, \sigma^b) \) we denote the maximal subtree of \( \pi \) having root \( \sigma^b \) and only containing blocked sequents. That is, any subtree of \( \pi \) having root \( \sigma^b \) and extending \( \Pi(\pi, \sigma^b) \) contains at least one unblocked sequent. Since rule \( \rightarrow R_2 \) is the only rule of RG3i which, read bottom-up, turns a blocked sequent into an unblocked one, every leaf \( \sigma^b_i \) of \( \Pi(\pi, \sigma^b) \) must be the conclusion of an application of \( \rightarrow R_2 \). Thus, every leaf \( \sigma^b_i \) of \( \Pi(\pi, \sigma^b) \) has the form

\[
[\Gamma \rightarrow, \Gamma \vdash \frac{b}{b} A_i \rightarrow B_i],
\]

where \( \mathcal{E}(A_i, \sigma^b_i) \neq T \). Hence \( \pi \) can be displayed as in Figure 5. We call unblocked successors of \( \sigma^b \) in \( \pi \) the unblocked sequents \( \sigma^b_1, \ldots, \sigma^b_n \).

Now, let us consider an RG3i-derivation \( \pi_u \) of an unblocked sequent \( \sigma^u \) having root rule \( S_u \). Since every premise \( \sigma' \) of \( S_u \) is a blocked sequent, the subderivation of \( \pi_u \) having root \( \sigma' \) has the form displayed in Figure 5. The set of the unblocked successors of \( \sigma^u \) in \( \pi_u \) is the union of the sets of unblocked successors of the premises of \( S_u \) in \( \pi_u \).
If \( \pi \) is one of the rules of \( \Gamma \) and \( \Lambda \), then \( \pi \) is an RG3i-derivation of \( \sigma^u_i \); since the root sequent of \( \pi \) is unblocked, the rule \( R \) is any of the rules of RG3i but \( S_b \).

(K1) If \( R \) is Irr, then \( \text{depth}(\pi) = 0 \) and \( \sigma^u = [\Gamma \vdash \Lambda, \Gamma^\Lambda \vdash H] \) is an irreducible sequent. We set \( \text{Mod}(\pi) = \langle \{ \rho \}, \leq, \rho, V \rangle \) where \( \leq = \{ (\rho, \rho) \} \) and \( V(\rho) = \Gamma^\Lambda \).

(K2) If \( R \) is one of the rules \( \Lambda L, \Lambda R, \vee L_1, \rightarrow L_1 \rightarrow R_1, \rightarrow R_2 \), let \( \pi' \) be the immediate subderivation of \( \pi \). Then \( \text{Mod}(\pi) = \text{Mod}(\pi') \).

(K3) If \( R \) is \( S_u \), then \( \sigma^u = [\Gamma \vdash \Lambda, \Gamma^\Lambda \vdash H] \) and, as discussed above, \( \pi \) has the form

\[
\begin{array}{c}
\pi_1 \\
\vdots \\
\pi_n \\
\end{array}
\]

and

\[
\begin{array}{c}
\sigma^u_1 \\
\vdots \\
\sigma^u_n \\
\end{array}
\]

For every \( i \in \{1, \ldots, n\} \), let \( K_i = \{ P_i, \leq_i, \rho_i, V_i \} \) be the model \( \text{Mod}(\pi_i) \). Without loss of generality, we can assume that the \( P_i \)’s are pairwise disjoint. Let \( \rho \) be an element not in \( \bigcup_{i \in \{1, \ldots, n\}} P_i \) and let \( K = \{ P, \leq, \rho, V \} \) be the model such that:

- \( P = \{ \rho \} \cup \bigcup_{i \in \{1, \ldots, n\}} P_i \);
- \( \leq = \{ (\rho, \alpha) \mid \alpha \in P \} \cup \bigcup_{i \in \{1, \ldots, n\}} \leq_i ;\)
- \( V(\rho) = \Gamma^\Lambda \) and for every \( i \in \{1, \ldots, n\} \) and \( \alpha \in P_i \), \( V(\alpha) = V_i(\alpha) \).

Then \( \text{Mod}(\pi) = K \). The structure of \( \text{Mod}(\pi) \) is displayed in Figure 6.

We notice that in the model \( K \) defined in Point (K3) we have that, if \( \alpha \in P_i \), then \( K_i, \alpha \vdash A \) iff \( K, \alpha \vdash A \).

Before proving the main result about \( \text{Mod}(\pi) \) we provide an example.
Let us consider the RG3i-derivation $\pi_1$ for Scott principle $S$ displayed in Example 7.2. Using the notation in Figure 6, we can rewrite $\pi_1$ as follows:

\[
\begin{array}{c}
\frac{[p \land \lnot p \not\vdash \bot]}{\text{Irr}} \quad \frac{[\lnot p \land p, \lnot p \not\vdash \bot]}{\text{Irr}} \quad \frac{[\lnot p, \lnot p \not\vdash \bot]}{\text{Irr}} \quad \frac{[\lnot p, \lnot p \not\vdash \bot]}{\text{Irr}} \\
L_1 \quad L_1 \quad L_0 \quad L_1 \\
S_u \quad S_u \quad S_u \quad S_u
\end{array}
\]

The model $K = \text{Mod}(\pi_1)$ extracted from $\pi_1$ is displayed below. Models are represented as trees with the convention that $w < w'$ if the world $w$ is drawn below $w'$. In each world $w_i$, we list the propositional variables in $V(w_i)$. We inductively define the models $\text{Mod}(\pi_i)$ for every $i$ such that $\sigma_i = [\rho_i \vDash H_i]$ in an unblocked sequent. At each step one can check that $\text{Mod}(\pi_1), \rho_i \vDash \sigma_i$, where $\rho_i$ is the root of $\text{Mod}(\pi_i)$. Hence $\text{Mod}(\pi_1), w_2 \vDash [\lnot \vdash S]$ and $\text{Mod}(\pi_1)$ is a counter-model for $[\Rightarrow S]$.

We denote with $\mathcal{K}_w$ the submodel of $\mathcal{K}$ with root $w$.

- The subderivations of depth zero are $\pi_8, \pi_{12}$ and $\pi_{16}$. By Point (K1) they generate models consisting of one world:
  \[
  \begin{align*}
  \text{Mod}(\pi_8) &= \mathcal{K}_{w_8}, \text{ where } V(w_8) = \{p\} \text{ (} p \text{ is the only propositional variable in } \Gamma_8 \} \\
  \text{Mod}(\pi_{12}) &= \mathcal{K}_{w_{12}}, \text{ where } V(w_{12}) = \emptyset \\
  \text{Mod}(\pi_{16}) &= \mathcal{K}_{w_{16}}, \text{ where } V(w_{16}) = \{p\}
  \end{align*}
  \]

- By Point (K2), $\text{Mod}(\pi_6) = \text{Mod}(\pi_7) = \text{Mod}(\pi_8) = \mathcal{K}_{w_8}$.

- By Point (K3) to build $\text{Mod}(\pi_4)$ we have to consider the unblocked successors of $\sigma_4$; here the only unblocked successor of $\sigma_4$ is $\sigma_6$. Hence $\text{Mod}(\pi_4) = \mathcal{K}_{w_4}$ is obtained.
by extending $K_{w_4}$ with the new root element $w_4$. Since no propositional variable belongs to $\Gamma_4$, $V(w_4) = \emptyset$.

— By Point (K2), $\text{Mod}(\pi_{10}) = \text{Mod}(\pi_{11}) = \text{Mod}(\pi_{12}) = K_{w_{12}}$.

— By Point (K2), $\text{Mod}(\pi_{14}) = \text{Mod}(\pi_{15}) = K_{w_{16}}$.

— By Point (K3) to build $\text{Mod}(\pi_2)$ we have to consider the unblocked successors $\sigma_4, \sigma_{10}$ and $\sigma_{14}$ of $\sigma_2$ and glue the models $\text{Mod}(\pi_4) = K_{w_4}, \text{Mod}(\pi_{10}) = K_{w_{14}}$ and $\text{Mod}(\pi_{14}) = K_{w_{16}}$ by means of the new root $w_2$ such that $V(w_2) = \emptyset$. Hence $\text{Mod}(\pi_2)$ is $K_{w_2}$.

Finally, $\text{Mod}(\pi_1) = \text{Mod}(\pi_2) = K_{w_2}$.

In the next lemma we show that the model described in (K1) is a counter-model for the irreducible sequent $\sigma^u$. We notice that the proof requires property (E1), stating that $E(A, \sigma^u)$ only depends on the formulas of $\sigma^u$ which are subformulas of $A$, and the semantical property (E12.2).

**Lemma 9.3.** Let $\sigma^u = [\Gamma \to, \Gamma^\text{At} \Downarrow H]$ be an irreducible sequent and let $K = \langle \{\rho\}, \{\rho, \rho\}, \rho, V\rangle$, where $V(\rho) = \Gamma^\text{At}$. Then $K, \rho \not\models \sigma^u$.

**Proof.** By definition of irreducible sequent, either $H = \bot$ or $H \in \mathcal{V} \setminus \Gamma^\text{At}$ or $H = H_0 \lor H_1$. We prove that the following hold by induction on the structure of formulas:

- (B1) $K, \rho \not\models A$, for every $A \to B \in \Gamma^{-\bot}$;
- (B2) $K, \rho \models A \to B$, for every $A \to B \in \Gamma^{-\bot}$;
- (B3) $K, \rho \not\models H$.

As for Point (B1) let $A \to B \in \Gamma^{-\bot}$. Since $\sigma^u$ is irreducible, we know that $E(A, \sigma^u) = F$. Now, let us consider the sequent

$$\mathfrak{s} = \begin{cases} \left[\Gamma \to \cap Sf(A), \Gamma^\text{At} \Downarrow H\right] & \text{if } H \in Sf(A) \\ \left[\Gamma \to \cap Sf(A), \Gamma^\text{At} \Downarrow \bot\right] & \text{otherwise} \end{cases}$$

Since $\text{Restr}(\sigma^u, A) = \text{Restr}(\mathfrak{s}, A)$, by property (E1), we get $E(A, \mathfrak{s}) = E(A, \sigma^u) = F$. We prove that $K, \rho \not\models A$ by applying (E12.2) w.r.t. $\mathfrak{s}$. To this aim we show that $K, \rho \not\models \mathfrak{s}$. By hypothesis, we know that $V(\rho) = \Gamma^\text{At}$. It remains to show that:

1. $K, \rho \not\models Z$, for every $Z \in \Gamma^{-\bot} \cap Sf(A)$;
2. $\text{if } H \in Sf(A)$ (where $H$ is the formula in the right-hand side of $\sigma^u$) then $K, \rho \not\models H$.

Let $Z \in \Gamma^{-\bot} \cap Sf(A)$; since $|Z| < |A \to B|$, by the induction hypothesis on Point (B2) we get $K, \rho \not\models Z$, hence (1) holds. Let us assume $H \in Sf(A)$. We have $|H| < |A \to B|$ hence, by the induction hypothesis on Point (B3), $K, \rho \not\models H$. By points (1), (2) and the definition of $V$, we get $K, \rho \not\models \mathfrak{s}$. Since $E(A, \mathfrak{s}) = F$, by (E12.2) we deduce $K, \rho \not\models A$, and this concludes the proof of Point (B1).

Point (B2) immediately follows from Point (B1) and the fact that $\rho$ does not have proper successors in $K$.

Finally, we prove Point (B3). If $H = \bot$ or $H \in \mathcal{V} \setminus \Gamma^\text{At}$, we immediately get $K, \rho \not\models H$. It remains to consider the case $H = H_0 \lor H_1$. Let $i \in \{0, 1\}$. Since $\sigma^u$ is irreducible, $E(H_i, \sigma^u) = F$. Let $\mathfrak{s}_i = \left[\Gamma^{-\bot} \cap Sf(H_i), \Gamma^\text{At} \Downarrow \bot\right]$. We have $\text{Restr}(\mathfrak{s}_i) = \text{Restr}(\sigma^u, H_i)$; by (E1), it follows that $E(H_i, \mathfrak{s}_i) = F$. Let $Z \in \Gamma^{-\bot} \cap Sf(H_i)$. Since $|Z| < |H_0 \lor H_1|$, by induction hypothesis on Point (B2) we get $K, \rho \not\models Z$; this implies that $K, \rho \not\models \mathfrak{s}_i$. By (E12.2), it follows that $K, \rho \not\models H_i$. Thus, $K, \rho \not\models H_0 \lor H_1$, and this concludes the proof of (B3). By points (B2), (B3) and the definition of $V$, we conclude $K, \rho \not\models \sigma^u$. □

Now we show that, for every RG3i-derivation $\pi$ of an unblocked sequent $\sigma^u$, $\text{Mod}(\pi)$ is a counter-model for $\sigma^u$.

**Theorem 9.4.** Let $\pi$ be an RG3i-derivation of an unblocked sequent $\sigma^u$ and let $\rho$ be the root of $\text{Mod}(\pi)$. Then $\text{Mod}(\pi), \rho \not\models \sigma^u$. 

**Proof.** The proof goes by induction on \( \text{depth}(\pi) \). If \( \text{depth}(\pi) = 0 \), then \( \sigma^u \) is an irreducible sequent and the assertion follows by Lemma 9.3.

Now let \( \text{depth}(\pi) > 0 \). The proof goes by cases on the rule \( R \) applied at the root of \( \pi \), that can be any of the rules of \( \text{RG3i} \) but \( \text{Irr} \) and \( S_u \). If \( R \neq \to R_1 \) and \( R \neq S_u \), the proof immediately follows by the induction hypothesis.

If \( R = \to R_1 \), then \( \pi \) has the form:

\[
\begin{align*}
\vdots & \quad \pi_1 \quad \vdots \\
\sigma_1^u &= [\Gamma \to, \Gamma^A \vDash B] \\
\sigma^u &= [\Gamma \to, \Gamma^A \vDash A \to B] \rightarrow R_1 \\
\end{align*}
\]

where \( \mathcal{E}(A, \sigma^u) = \mathcal{T} \), which implies by (E3) that \( \mathcal{E}^\downarrow(A, \sigma^u) = \mathcal{T} \). By induction hypothesis, \( \text{Mod}(\pi_1), \rho_1 \triangleright \sigma_1^u \), where \( \rho_1 \) is the root of \( \text{Mod}(\pi_1) \). By Property (E12.1) it follows that \( \text{Mod}(\pi_1), \rho_1 \models A \), hence \( \text{Mod}(\pi_1), \rho_1 \not\models A \rightarrow B \). Since \( \text{Mod}(\pi) = \text{Mod}(\pi_1) \) we conclude that \( \text{Mod}(\pi), \rho \triangleright \sigma^u \).

Let \( R = S_u \), let \( \sigma^u = [\Gamma \to, \Gamma^A \vDash H] \) and let \( K = \langle P, \leq, \rho, V \rangle \) be the model \( \text{Mod}(\pi) \).

We prove, by a secondary induction hypothesis on the structure of formulas, that the following hold:

- (B1) \( K, \rho \not\models A \), for every \( A \rightarrow B \in \Gamma \to \);
- (B2) \( K, \rho \models A \rightarrow B \), for every \( A \rightarrow B \in \Gamma \to \);
- (B3) \( K, \rho \not\models H \).

To prove Point (B1), let \( A \rightarrow B \in \Gamma \rightarrow \). If \( \mathcal{E}(A, \sigma^u) = \mathcal{F} \), we can proceed as in the proof of (B1) in Lemma 9.3. Let us assume that \( \mathcal{E}(A, \sigma^u) \neq \mathcal{F} \). By definition of \( S_u \), \( \sigma^u \) has an immediate subderivation \( \pi_A \) of \( \sigma^u = [\Gamma \rightarrow, \Gamma^{A} \vDash A] \) having the form described in Figure 5, that is \( \pi_A \) is:

\[
\begin{align*}
\vdots & \quad \pi_1 \quad \vdots \\
\sigma_A^1 &= [\Gamma \rightarrow, \Gamma^A \vDash A_1 \rightarrow B_1] \\
\sigma^u &= [\Gamma \rightarrow, \Gamma^A \vDash A \rightarrow B] \\
\end{align*}
\]

where \( \Pi(\pi_A, \sigma_b^1) \) only contains blocked sequents. Now we show that \( \Pi(\pi_A, \sigma_b^1) \) meets hypothesis (H1)–(H3) of Lemma 9.1 w.r.t. the root \( \rho \) of \( K \), and we apply the lemma to deduce that \( K, \rho \not\models A \). To prove (H1) let \( i \in \{1, \ldots, n\} \), we must show that \( K, \rho_i \not\models A_i \rightarrow B_i \). By the structure of \( \pi \) we know that \( \sigma_i^u \) is an unblocked successor of \( \sigma^u \). By construction of \( K \), the root \( \rho_i \) of \( K_i = \text{Mod}(\pi_i) \) is an immediate successor of \( \rho \) in \( K \) (see Figure 6). By the main induction hypothesis \( K_i, \rho_i \triangleright \sigma_i^u \); this implies that \( K_i, \rho_i \models A_i \) and \( K_i, \rho_i \not\models B_i \). Since \( K_i \) is a submodel of \( K \), we get \( K, \rho_i \models A_i \) and \( K, \rho_i \not\models B_i \), which implies \( K, \rho \not\models A_i \rightarrow B_i \). Since this holds for every \( i \in \{1, \ldots, n\} \), hypothesis (H1) of Lemma 9.1 follows. To prove that hypothesis (H2) of Lemma 9.1 holds, let \( Z \in \Gamma \to \cap \text{S}(A) \). Since \( |Z| < |A \rightarrow B| \), by the secondary induction hypothesis on Point (B2), we get \( K, \rho \models Z \).

The hypothesis (H3) of Lemma 9.1 follows by the definition of \( V \) in \( K \). Therefore, we can apply Lemma 9.1 to deduce \( K, \rho \not\models A \), and this proves Point (B1).

Let us consider Point (B2). The \( \text{RG3i} \)-derivation \( \pi \) can be displayed as (see Figure 6):

\[
\begin{align*}
\vdots & \quad \pi_1 \quad \vdots \\
\sigma_A^1 &= [\Gamma \rightarrow, \Gamma^A \vDash A_1 \rightarrow B_1] \\
\vdots & \quad \pi_n \quad \vdots \\
\sigma^u &= [\Gamma \rightarrow, \Gamma^A \vDash A \rightarrow B] \\
\end{align*}
\]

\[
\sigma^u = S_u
\]
Lemma 9.1, we get hypothesis (H1)–(H3) of Lemma 9.1 with respect to the root \( \rho \). Proceeding as in the proof of Point (B1), we can show that \( A \) is refutable; this proves the soundness of \( \mathcal{K} \).

Given a formula \( B \), if \( \mathcal{E} \) is an evaluation function, namely, \( \mathcal{E}(\mathcal{K},\rho) \neq \mathcal{F} \). Then, by definition of \( S_u \), \( \pi \) has an immediate subderivation \( \pi_H \) of \( \sigma_H^u = [\mathcal{G} \rightarrow \mathcal{G} \vdash \rho H_i] \) having the form (see Figure 5):

\[
\frac{\pi_1}{[\mathcal{G} \rightarrow \mathcal{G} \vdash \rho A_1 \vdash B_1]} \quad \cdots \quad \frac{\pi_n}{[\mathcal{G} \rightarrow \mathcal{G} \vdash \rho A_n \vdash B_n]} \quad \cdots \quad \frac{\Pi(\mathcal{G} \vdash \rho \mathcal{H})}{\sigma_H^u = [\mathcal{G} \rightarrow \mathcal{G} \vdash \rho H_i]}
\]

Proceeding as in the proof of Point (B1), we can show that \( \Pi(\mathcal{G} \vdash \rho \mathcal{H}) \) meets the hypothesis (H1)–(H3) of Lemma 9.1 with respect to the root \( \rho \) of \( \mathcal{K} \). Therefore, by Lemma 9.1, we get \( \mathcal{K}, \rho \not\vdash H_i \). This proves \( \mathcal{K}, \rho \not\vdash \rho H \) and hence Point (B3) holds.

By points (B2) and (B3) and the fact that \( V(\rho) = \mathcal{G} \), we conclude \( \mathcal{K}, \rho \vdash \sigma_H^u \). □

By Theorem 9.4, if there exists an RG3i derivation of an unblocked sequent \( \sigma,H \), then \( \sigma,H \) is refutable; this proves the soundness of RG3i stated in Theorem 6.1.

10. \( \mathcal{E} \) IS AN EVALUATION FUNCTION

Here we prove that the function \( \mathcal{E} \) defined in Section 3.1 satisfies Properties (E1)–(E12) of Section 3, and we discuss its time complexity and its impact on the performance of our proof-search procedure.

Inspecting the definition of the functions displayed in figures 1 and 2, one can check that they do not increase the size of formulas, namely, \(|\mathcal{B}(A)| \leq |A|, |\mathcal{R}^R(A,\sigma)| \leq |A|, |\mathcal{R}^L(A,\sigma)| \leq |A|, |\mathcal{R}(A,\sigma)| \leq |A|\). Moreover, \( \mathcal{B}(\mathcal{B}(A)) = \mathcal{B}(A) \).

First of all, let us prove some general facts about local formulas defined in Section 3.1. Given a formula \( A \), we denote with \( \mathcal{V}(A) \) the set of propositional variables occurring in \( A \).

**Lemma 10.1.** Let \( L \) be a local formula, \( \mathcal{K} = \langle \mathcal{P}, \mathcal{L}, \mathcal{V} \rangle \) a Kripke model and \( \alpha \) a world of \( \mathcal{K} \). If \( \mathcal{V}(\alpha) \cap \mathcal{V}(L) = \emptyset \), then \( \mathcal{K}, \alpha \not\vdash L \).

**Proof.** By induction on the structure of \( L \). If \( L \in \mathcal{V} \cup \{ \bot \} \), then the assertion immediately follows. Let \( L = L_0 \wedge A \), with \( L_0 \) a local formula and \( A \) any formula. By induction hypothesis, \( \mathcal{K}, \alpha \not\vdash L_0 \), which implies \( \mathcal{K}, \alpha \not\vdash L_0 \wedge A, \) the case \( L = A \) \wedge \( L_0 \) is similar. If \( L = L_0 \leftarrow L_1 \) then, by induction hypothesis, \( \mathcal{K}, \alpha \not\vdash L_0 \) and \( \mathcal{K}, \alpha \not\vdash L_1 \), hence \( \mathcal{K}, \alpha \not\vdash L_0 \leftarrow L_1 \). □

**Lemma 10.2.** Let \( L \) be a local formula and \( \sigma = [\mathcal{G} \vdash \rho H] \) any sequent. Then:

1. \( \mathcal{B}(L) \) is a local formula;
2. \( \mathcal{R}^H(L,\sigma) \) is a local formula.

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
PROOF. We prove Point (1) by induction on the structure of \( L \). If \( L \in \mathcal{V} \cup \{\bot\} \), then \( B(L) = L \), hence (1) holds. Let \( L = L_0 \land A \), where \( L_0 \) is a local formula and \( A \) any formula. By induction hypothesis, \( B(L_0) \) is a local formula; in particular, \( B(L_0) \neq \top \). It follows that \( B(L) \) is one of the formulas \( \bot \), \( B(L_0) \) or \( B(L_0) \land B(A) \); in any case, \( B(L) \) is a local formula. The case \( L = A \land L_0 \) is similar. Finally, let \( L = L_0 \lor L_1 \). By the induction hypothesis, both \( B(L_0) \) and \( B(L_1) \) are local formulas. This implies that \( B(L) \) is one of the formulas \( B(L_0) \lor B(L_1) \), hence \( B(L) \) is a local formula.

We prove Point (2) by induction on the structure of \( L \), through a case analysis on the definition of \( R^R(L, \sigma) \). If \( R^R(L, \sigma) = \bot \) or \( R^R(L, \sigma) = L \), Point (2) immediately follows. Let \( L = A_0 \lor A_1 \), where \( \cdot \in \{\land, \lor\} \), and let \( B_0 = R^R(A_0, \sigma) \) and \( B_1 = R^R(A_1, \sigma) \). Firstly, we show that \( B_0 \) \& \( B_1 \) is a local formula. If \( L = A_0 \land A_1 \), then, for some \( i \in \{0,1\} \), \( A_i \) is a local formula. By induction hypothesis, \( B_i \) is a local formula, hence \( B_0 \land B_1 \) is a local formula. If \( L = A_0 \lor A_1 \), then both \( A_0 \) and \( A_1 \) are local formulas. By induction hypothesis, \( B_0 \) and \( B_1 \) are local formulas, hence \( B_0 \lor B_1 \) is a local formula. Since \( B_1 \) \& 0 \& \( B_1 \) is a local formula, by Point (1) it follows that \( R^R(A_0 \lor A_1, \sigma) = B(B_0 \lor B_1) \) is a local formula. □

**Lemma 10.3.** Let \( \sigma = [\Gamma \vdash^* H] \) and \( \sigma_\bot = [\Gamma \vdash^* \bot] \) be sequents and let \( A \) be any formula. Then:

1. \( R^L(A, \sigma) = R^L(A, \sigma_\bot) \);
2. \( B(A) = A \) implies \( R^R(A, \sigma_\bot) = A \);
3. \( R^R(A, \sigma_\bot) = R^L(A, \sigma) \).

**Proof.** Point (1) follows by the fact that the value of \( R^L(A, \sigma) \) only depends on \( \Gamma \).

We prove Point (2) by induction on the size of \( A \) through a case analysis on the definition of \( R^R(A, \sigma_\bot) \). If \( A \in \mathcal{V} \cup \{\land, \lor\} \) or \( A = B \rightarrow C \), then \( R^R(A, \sigma_\bot) = A \), hence (2) holds. Let \( A = A_0 \land A_1 \), with \( \cdot \in \{\land, \lor\} \). Since \( B(A) = A \), we must have \( B(A_0) = A_0 \) and \( B(A_1) = A_1 \). By induction hypothesis, \( R^R(A_0, \sigma_\bot) = A_0 \) and \( R^R(A_1, \sigma_\bot) = A_1 \). We get:

\[
R^R(A_0 \land A_1, \sigma_\bot) = B(R^R(A_0, \sigma_\bot) \& R^R(A_1, \sigma_\bot)) = B(A_0 \land A_1) = A_0 \land A_1 .
\]

Now let us consider Point (3). By the idempotence of \( B \), we get \( B(R^L(A, \sigma_\bot)) = R^L(A, \sigma_\bot) \). Therefore, by Point (2) \( R(A, \sigma_\bot) = R(R^L(A, \sigma_\bot), \sigma_\bot) = R^L(A, \sigma_\bot) \). Since by Point (1) \( R^R(A, \sigma_\bot) = R^L(A, \sigma) \), we get Point (3). □

Let \( \tilde{\mathcal{E}}^L(A, [\Gamma \vdash^* H]) = \tilde{\mathcal{E}}(A, [\Gamma \vdash^* \bot]) \) be the left-evaluation of \( \mathcal{E} \) (see Section 3). By the above lemma we can explicitly define \( \tilde{\mathcal{E}}^L(A, \sigma) \) as follows:

\[
\tilde{\mathcal{E}}^L(A, \sigma) = \begin{cases} T & \text{if } R^L(A, \sigma) = T \\ F & \text{if } R^L(A, \sigma) \text{ is a local formula} \\ X & \text{otherwise} \end{cases}
\]

**Lemma 10.4.** Let \( \sigma = [\Gamma \vdash H] \) be a sequent, \( A \) and \( K \) formulas, and let \( \Gamma' \) be any finite set of formulas. Then:

1. If \( \sigma' = [\Gamma' \vdash K] \) and \( \Gamma' \cap \mathsf{Sf}(A) = \Gamma \cap \mathsf{Sf}(A) \), then \( R^L(A, \sigma') = R^L(A, \sigma) \);
2. If \( \sigma' = [\Gamma' \vdash H] \) then \( R^R(A, \sigma') = R^R(A, \sigma) \);
3. If \( H \not\in \mathsf{Sf}(A) \) and \( \sigma' = [\Gamma' \vdash \bot] \), then \( R^R(A, \sigma') = R^R(A, \sigma) \).

**Proof.** Let \( \Gamma' \cap \mathsf{Sf}(A) = \Gamma \cap \mathsf{Sf}(A) \); we prove Point (1) by induction on the size of \( A \), through a case analysis on the definition of \( R^L(A, \sigma') \). If \( A \in \mathcal{V} \), then \( R^L(A, \sigma') = \top \). By the assumption on \( \Gamma \), it holds that \( A \in \Gamma \), hence \( R^L(A, \sigma) = \top \).

Let us assume \( A \not\in \Gamma' \), which implies \( A \not\in \Gamma \). If \( A \in \mathcal{V} \cup \{\bot, \top\} \), then \( R^L(A, \sigma') = A \) and \( R^L(A, \sigma) = A \). Let \( A = A_0 \land A_1 \), with \( \cdot \in \{\land, \lor, \rightarrow\} \). For every \( i \in \{0,1\} \), we have...
Points (2) and (3) immediately follow from the definition of \( R^R \).

**PROPOSITION 10.5.** The function \( \tilde{\mathcal{E}} \) satisfies properties (E1)-(E11).

**PROOF.**

- Property (E1): \( \tilde{\mathcal{E}}(A, \sigma) = \tilde{\mathcal{E}}(A, \text{Restr}(\sigma, A)) \).
  
  Let \( \sigma = [\Gamma \Rightarrow H] \) and let \( \sigma' = \text{Restr}(\sigma, A) = [\Gamma' \Rightarrow K] \), where
  
  - \( \Gamma' = \Gamma \cap \text{Sf}(A) \);
  
  - \( K = H \) if \( H \in \text{Sf}(A) \) and \( K = \bot \) otherwise.
  
  By Lemma 10.4, the following equivalences hold:
  
  \[
  \mathcal{R}^L(A, \sigma) = \mathcal{R}^L(A, \sigma')
  \]
  
  \[
  \mathcal{R}(A, \sigma) = \mathcal{R}^R(\mathcal{R}^L(A, \sigma), \sigma') = \mathcal{R}^R(\mathcal{R}^L(A, \sigma'), \sigma') = \mathcal{R}(A, \sigma')
  \]
  
  This implies that \( \tilde{\mathcal{E}}(A, \sigma) = \tilde{\mathcal{E}}(A, \sigma') \).

- Property (E2): Let \( \sigma = [\Gamma \Rightarrow H] \) and \( A \in \mathcal{V} \cup \{\bot\} \). Then, \( \tilde{\mathcal{E}}(A, \sigma) = T \) if \( A \in \Gamma \) and \( \tilde{\mathcal{E}}(A, \sigma) = F \) otherwise.
  
  If \( A \in \Gamma \) then \( \mathcal{R}^L(A, \sigma) = T \), hence \( \tilde{\mathcal{E}}(A, \sigma) = T \). Let \( A \notin \Gamma \). We have \( \mathcal{R}^L(A, \sigma) = A \), hence \( \mathcal{R}(A, \sigma) = \mathcal{R}^R(A, \sigma) \). If \( A = H \), then \( \mathcal{R}(A, \sigma) = \bot \); otherwise, being \( A \in \mathcal{V} \cup \{\bot\}, \mathcal{R}(A, \sigma) = A \) and \( A \) is a local formula, hence \( \tilde{\mathcal{E}}(A, \sigma) = F \).

- Property (E3): \( \tilde{\mathcal{E}}(A, \sigma) = T \) iff \( \tilde{\mathcal{E}}^L(A, \sigma) = T \).
  
  It immediately follows by the definition of \( \tilde{\mathcal{E}}^L \).

- Property (E4): \( \tilde{\mathcal{E}}^L(A, \sigma) = F \) implies \( \tilde{\mathcal{E}}(A, \sigma) = F \).
  
  Let \( \tilde{\mathcal{E}}^L(A, \sigma) = F \); this means that \( \mathcal{R}^L(A, \sigma) \) is a local formula. By Point (2) of Lemma 10.2, \( \mathcal{R}(A, \sigma) = \mathcal{R}^R(\mathcal{R}^L(A, \sigma), \sigma) \) is a local formula, which implies \( \tilde{\mathcal{E}}(A, \sigma) = F \).

- Property (E5): \( \tilde{\mathcal{E}}^L(A, [A, \Gamma \Rightarrow H]) = T \).
  
  It immediately follows from the fact that \( \mathcal{R}^L(A, [A, \Gamma \Rightarrow H]) = T \).

- Property (E6): \( \tilde{\mathcal{E}}^L(A, \sigma) = T \) and \( \tilde{\mathcal{E}}^L(B, \sigma) = T \) imply \( \tilde{\mathcal{E}}^L(A \land B, \sigma) = T \).
  
  Let \( \sigma = [\Gamma \Rightarrow H] \). If \( A \land B \in \Gamma \) then \( \mathcal{R}^L(A \land B, \sigma) = \top \) and \( \tilde{\mathcal{E}}^L(A \land B, \sigma) = T \). Otherwise, since by the hypothesis, \( \tilde{\mathcal{E}}^L(A, \sigma) = \tilde{\mathcal{E}}^L(B, \sigma) = T \), by definition of \( \tilde{\mathcal{E}}^L \) we get \( \mathcal{R}^L(A, \sigma) = \mathcal{R}^L(B, \sigma) = \top \). Hence:
  
  \[
  \mathcal{R}^L(A \land B, \sigma) = B(\mathcal{R}^L(A, \sigma) \land \mathcal{R}^L(B, \sigma)) = B((\top \land \top) = \top
  \]
  
  which implies \( \tilde{\mathcal{E}}^L(A \land B, \sigma) = T \).

- Property (E7): \( \tilde{\mathcal{E}}^L(A_i, \sigma) = T \), with \( i \in \{0, 1\} \), implies \( \tilde{\mathcal{E}}^L(A_0 \lor A_1, \sigma) = T \).
  
  Similar to the proof of (E6).

- Property (E8): \( \tilde{\mathcal{E}}^L(B, \sigma) = T \) implies \( \tilde{\mathcal{E}}^L(A \rightarrow B, \sigma) = T \).
  
  Similar to the proof of (E6).
— Property (E9): if $A_0 \land A_1 \not\in \Gamma$ and $\mathcal{E}^L(A_i, \Gamma \models H) = F$ for some $i \in \{0, 1\}$, then $\mathcal{E}^L(A_0 \land A_1, \Gamma \models H) = F$.

Let $\sigma = [\Gamma \models H]$ and let us suppose that $\mathcal{E}^L(A_0, \sigma) = F$. This means that $\mathcal{R}^L(A_0, \sigma) = L_0$ is a local formula. Since $A_0 \land A_1 \not\in \Gamma$, we have:

$$\mathcal{R}^L(A_0 \land A_1, \sigma) = B(L_0 \land \mathcal{R}^L(A_1, \sigma)) = B$$

Since $L_0 \land \mathcal{R}^L(A_1, \sigma)$ is a local formula, by Point (1) of Lemma 10.2 it follows that $B$ is a local formula, hence $\mathcal{E}^L(A_0 \land A_1, \sigma) = F$. The case $\mathcal{E}^L(A_1, \sigma) = F$ is similar.

— Property (E10): if $A \lor B \not\in \Gamma$, $\mathcal{E}^L(A, \Gamma \models H) = F$ and $\mathcal{E}^L(B, \Gamma \models H) = F$, then $\mathcal{E}^L(A \lor B, \Gamma \models H) = F$.

Similar to the proof of (E9).

— Property (E11): if $A \rightarrow B \not\in \Gamma$ and $\mathcal{E}^L(A, \Gamma \models H) = T$ and $\mathcal{E}^L(B, \Gamma \models H) = F$, then $\mathcal{E}^L(A \rightarrow B, \Gamma \models H) = F$.

Similar to the proof of (E9). 

In order to prove Property (E12), we show some semantical properties on $\mathcal{R}(A, \sigma)$.

**Lemma 10.6.** Let $\sigma = [\Gamma \models H]$, let $\mathcal{K} = \langle P, \leq, \rho, V \rangle$ be a Kripke model and $\alpha \in P$ such that $\mathcal{K}, \alpha \models \sigma$. Then, for every formula $A$:

1. $\mathcal{K}, \alpha \models A \leftrightarrow B(A)$;
2. $\mathcal{K}, \alpha \models A \leftrightarrow \mathcal{R}^L(A, \sigma)$;
3. $\mathcal{K}, \alpha \models A$ iff $\mathcal{K}, \alpha \models \mathcal{R}^R(A, \sigma)$;
4. $\mathcal{K}, \alpha \models A$ iff $\mathcal{K}, \alpha \models \mathcal{R}(A, \sigma)$.

**Proof.** The proof of Point (1) is trivial.

As for Point (2), we proceed by induction on the size of $A$. If $A \in \Gamma$ then $\mathcal{R}^L(A, \sigma) = T$. Since $\mathcal{K}, \alpha \models A$, it holds that $\mathcal{K}, \alpha \models A \leftrightarrow \mathcal{R}^L(A, \sigma)$. Let $A \not\in \Gamma$. If $A \in \langle, \bot, \top \rangle$ then $\mathcal{R}^L(A, \sigma) = A$, hence Point (2) holds. Let $A = A_0 \cdot A_1$, where $\cdot \in \{\land, \lor, \rightarrow\}$. By induction hypothesis we have

$$\mathcal{K}, \alpha \models A_0 \leftrightarrow \mathcal{R}^L(A_0, \sigma) \quad \text{and} \quad \mathcal{K}, \alpha \models A_1 \leftrightarrow \mathcal{R}^L(A_1, \sigma)$$

hence $\mathcal{K}, \alpha \models (A_0 \cdot A_1) \leftrightarrow (\mathcal{R}^L(A_0, \sigma) \cdot \mathcal{R}^L(A_1, \sigma))$. By Point (1), we get

$$\mathcal{K}, \alpha \models (A_0 \cdot A_1) \leftrightarrow B(\mathcal{R}^L(A_0, \sigma) \cdot \mathcal{R}^L(A_1, \sigma))$$

and this concludes the proof of Point (2).

Let us consider Point (3). If $A = H$, then $\mathcal{R}^R(A, \sigma) = \bot$. Since $\mathcal{K}, \alpha \not\models A$ and $\mathcal{K}, \alpha \not\models \bot$ the assertion holds. Let $A \neq H$. If $A \in \langle, \bot, \top \rangle$ or $A = B \rightarrow C$, then $\mathcal{R}^R(A, \sigma) = A$ and the assertion immediately follows. Let $A = A_0 \cdot A_1$, with $\cdot \in \{\land, \lor\}$. By induction hypothesis we have

$$\mathcal{K}, \alpha \models A_0 \iff \mathcal{K}, \alpha \models \mathcal{R}^R(A_0, \sigma)$$

$$\mathcal{K}, \alpha \models A_1 \iff \mathcal{K}, \alpha \models \mathcal{R}^R(A_1, \sigma).$$

Since $\cdot \in \{\land, \lor\}$, it follows that

$$\mathcal{K}, \alpha \models A_0 \cdot A_1 \iff \mathcal{K}, \alpha \models \mathcal{R}^R(A_0, \sigma) \cdot \mathcal{R}^R(A_1, \sigma).$$

By Point (1), we get

$$\mathcal{K}, \alpha \models A_0 \cdot A_1 \iff \mathcal{K}, \alpha \models B(\mathcal{R}^R(A_0, \sigma) \cdot \mathcal{R}^R(A_1, \sigma))$$
and this concludes the proof of Point (3).

Point (4) immediately follows by points (2) and (3). □

It is easy to prove that the following holds:

**Lemma 10.7.** Let \([\Gamma \Rightarrow H]\) be a sequent, \(A\) a formula and let \(B = \mathcal{R}(A, [\Gamma \Rightarrow H])\). Then \(\forall(B) \cap \Gamma = \emptyset\). □

**Proposition 10.8.** The function \(\mathcal{E}\) satisfies \((\mathcal{E}12)\).

**Proof.** Let \(\sigma = [\Gamma \Rightarrow H]\), let \(\mathcal{K} = \langle P, \leq, \rho, V \rangle\) be a Kripke model and \(\alpha \in P\) such that \(\mathcal{K}, \alpha \models \sigma\). We have to show that:

(P1) \(\mathcal{E}^L(A, \sigma) = T\) implies \(\mathcal{K}, \alpha \models A\).

(P2) Let \(\mathcal{E}(A, \sigma) = F\) and \(V(\alpha) = \Gamma \cap V\). Then, \(\mathcal{K}, \alpha \not\models A\).

If \(\mathcal{E}^L(A, \sigma) = T\) then \(\mathcal{R}^L(A, \sigma) = T\). By Point (2) of Lemma 10.6 we get \(\mathcal{K}, \alpha \models A \leftrightarrow T\), hence \(\mathcal{K}, \alpha \models A\), and this proves (P1).

Now let us consider Point (P2). Since \(\mathcal{E}(A, \sigma) = F\), then \(\mathcal{R}(A, \sigma) = L\) is a local formula. By Lemma 10.7, \(V(L) \cap \Gamma = \emptyset\). Since by the hypothesis of Point (P2), \(V(\alpha) = \Gamma \cap V\), we get \(V(\alpha) \cap V(L) = \emptyset\). Applying Lemma 10.1, we get \(\mathcal{K}, \alpha \not\models L\) and, by Point (4) of Lemma 10.6, we conclude \(\mathcal{K}, \alpha \not\models A\). □

By propositions 10.5 and 10.8 we get:

**Theorem 10.9.** \(\mathcal{E}\) is an evaluation function. □

### 10.1. Complexity issues

We briefly discuss the time complexity of \(\mathcal{E}\) and the overhead time required by evaluations in the development of a branch. Let us suppose to search for a G3i-derivation of a sequent \(\sigma\) and let \(\mathcal{F}(u, \sigma)\) be the main call; without loss of generality, we assume that \(\sigma = [\Rightarrow H]\). To efficiently implement the evaluation function \(\mathcal{E}\), we can represent the formula \(H\) by a binary tree, where the subformulas (the nodes of the tree) are indexed. Since the calculus G3i has the subformula property, any sequent \(\sigma'\) generated during the computation of \(\mathcal{F}(u, \sigma)\) can be represented by an array of \(|Sf(H)|\) (cardinality of \(Sf(H)\)) elements: the \(i\)-th element of the array specifies if the \(i\)-th subformula of \(H\) occurs in the left-hand side, in the right-hand side, in both sides or does not occur in \(\sigma\). With such a representation, the membership of a formula to \(\sigma'\) can be checked in constant time, thus \(\mathcal{R}(A, \sigma')\) and \(\mathcal{R}(A, \sigma')\) can be computed in linear time in the size of \(A\) (for more details see [Ferrari et al. 2012]). Accordingly, any invocation of \(\mathcal{E}(A, \sigma')\) occurring during the execution of \(\mathcal{F}(u, \sigma)\) requires time \(O(|\sigma|)\); since \(A\) must be a subformula of \(H\), \(\mathcal{E}(A, \sigma')\) can be computed in time \(O(|\sigma|)\).

Now, let us consider the overhead time required by evaluations in the development of a branch (a CRC). Inspecting the code of \(\mathcal{F}\) one can check that line 25 is the only point where the number of evaluations to compute depends on the sequent \(\sigma' = [\Gamma' \Rightarrow H']\) at hand. Indeed, in line 25 we have to build \(\text{act}^{-1}(\sigma')\), hence we must compute \(\mathcal{E}(A, \sigma')\) for every \(A\) in \(\Theta = \{ A \mid A \rightarrow B \in \Gamma' \}\). Obviously, \(|\Theta| \leq |\sigma|\). To efficiently perform all these evaluations, a dynamic programming approach can be used: for every subformula \(A\) of \(H\), we store in a table \(T\) the value of \(\mathcal{E}(A, \sigma')\). Proceeding bottom-up on the structure of \(H\), the table \(T\) can be built in time \(O(|\sigma|)\). The value of \(\mathcal{E}(A, \sigma')\) can be now retrieved in constant time by a query to \(T\), hence the computation of \(\text{act}^{-1}(\sigma')\) requires time \(O(|\sigma|)\).

Since the length of a CRC starting from \((u, \sigma)\) is \(O(|\sigma|^2)\) (Theorem 8.4), we conclude that in the execution of \(\mathcal{F}(u, \sigma)\) the overall time needed to compute evaluations during the construction of a branch is \(O(|\sigma|^3)\).
11. CONCLUSIONS

First of all we discuss some variants of the decision procedure presented in this paper, then we provide a comparison between our approach and those based on histories. Finally we discuss related works and some directions for future work.

The original G3i calculus

To keep the presentation as simple as possible we based this paper on the calculus G3i, which is a variant of the one presented in [Troelstra and Schwichtenberg 2000] that we denote with G3iTS; G3i has two rules → R1 and → R2 to handle right implications whereas G3iTS has only rule → R2. The results of this paper can be reformulated using G3iTS along the following lines. First of all, we have to consider the refutation calculus RG3iTS obtained by replacing the rule → R1 of G3i with

\[
\frac{[A, \Gamma \vdash B]}{[\Gamma \vdash A \rightarrow B]} \quad \text{if } E(A, [\Gamma \vdash A \rightarrow B]) = T.
\]

The proof-search procedure for G3iTS and RG3iTS is obtained by substituting the recursive invocation \(\mathcal{F}(l, [\Gamma \Rightarrow B])\) at line 14 of \(\mathcal{F}\) with \(\mathcal{F}(l, [A, \Gamma \Rightarrow B])\). Let us denote with \(\mathcal{F}^{TS}\) such a variant of \(\mathcal{F}\). We can prove that \(\mathcal{F}^{TS}\) is a correct and terminating proof-search procedure along the lines of the proofs given for \(\mathcal{F}\). The main complication concerns the proof of the soundness of \(\mathcal{F}^{TS}\), since \(\mathcal{F}^{TS}\)-trees only containing blocked sequents have the form:

\[
\sigma_1^b = [\Gamma \Rightarrow, \Gamma^{At}, \Theta_1 \vdash H_1] \quad \ldots \quad \sigma_n^b = [\Gamma \Rightarrow, \Gamma^{At}, \Theta_n \vdash H_n]
\]

\[
\sigma^b = [\Gamma \Rightarrow, \Gamma^{At} \vdash H] \quad \text{(PBTS)}
\]

Differently from the analogous \(\mathcal{F}^{TS}\)-trees described in (PB) at page 20, we have to consider the sets \(\Theta_1, \ldots, \Theta_n\) containing the formulas introduced by possible applications of \(\rightarrow R_1'\) in \(\pi\). This complicates the proof of Theorem 9.4 and the proofs of related lemmas.

Refining \(\mathcal{F}\)

Still to avoid inessential complications we did not introduce in \(\mathcal{F}\) refinements that, exploiting evaluations, can reduce the search space. The first possible refinement consists in replacing the definition of \(act^\rightarrow([\Gamma \vdash H])\) at page 9 with:

\[
act_{LR}^\rightarrow([\Gamma \vdash H]) = \{ A | A \rightarrow B \in \Gamma \text{ and } E(A, [\Gamma \vdash H]) \neq F \text{ and } E(B, [\Gamma \vdash H]) \neq T \}
\]

Let \(\mathcal{F}^{LR}\) denote the version of \(\mathcal{F}\) using \(act_{LR}^\rightarrow\) instead of the original \(act^\rightarrow\). Differently from \(\mathcal{F}\), in \(\mathcal{F}^{LR}\) the application of the left rule for implication to \(\sigma = [A \rightarrow B, \Gamma \vdash H]\) is prevented also in the case where \(E(B, \sigma) = T\). We can prove that \(\mathcal{F}^{LR}\) is a correct and terminating proof-search procedure; in this case the complication is introduced in the proof of the correctness. The function \(\mathcal{F}^{LR}\) improves \(\mathcal{F}\) in the following sense: using the evaluation \(\hat{E}\), the overall time required by \(\mathcal{F}^{LR}(l, \sigma)\) to compute evaluations in the construction of a branch is \(O(|\sigma|^2)\) instead of \(O(|\sigma|^3)\). To give more details, we can prove that every CRC for \((l, \sigma)\) generated by \(\mathcal{F}^{LR}(l, \sigma)\) contains \(|\sigma|\) applications of (T9) and \(|\sigma| + 1\) applications of (T10) at most. Since, during the execution of \(\mathcal{F}^{LR}(l, \sigma)\), \(act_{LR}^\rightarrow\) is computed only before a (T9) or a (T10) application or before generating the last state of the CRC, we get that, in the construction of a branch, \(act_{LR}^\rightarrow\) is computed \(O(|\sigma|)\).
times. Hence, implementing $\hat{E}$ as discussed in Section 10.1, the overall time required to compute $\text{act}_{\text{LR}}^{\Rightarrow}$ during the development of a branch is $O(|\sigma|^2)$.

The second refinement we discuss is important for the following comparison with history based calculi. Let $\mathcal{F}^{\text{Eref}}$ be the version of $\mathcal{F}$ where a left rule is applied to $\sigma = [A, \Gamma \Rightarrow H]$ with main formula $A$ only if $E(A, [\Gamma \Rightarrow H]) \neq T$. As an example, the if-condition at line 3 of $\mathcal{F}$ must be rewritten as:

$$l = u \text{ and } \sigma = [A \land B, \Gamma' \Rightarrow H] \text{ where } \Gamma' = \Gamma \setminus \{A \land B\} \text{ and } E(A \land B, [\Gamma' \Rightarrow H]) \neq T$$

$\mathcal{F}^{\text{Eref}}$ is a correct and terminating proof-search procedure. To prove this we have to modify the proof of correctness. Even if this refinement allows us to prune the search space, it does not improve the asymptotic behaviour of $\mathcal{F}$.

Comparison with loop-checking and history mechanisms

Loop-checking is a typical solution to get termination of proof-search procedures based on G3i. Efficient implementations of loop-checking are the history-based calculi discussed in [Heuerding et al. 1996; Howe 1997; 1998]. To direct and possibly stop the proof-search, histories require space to store the right-hand side formulas already used. A strict comparison between the two approaches is hard, to stress the differences we provide some examples. For the sake of concreteness, we refer to the calculus G3$^H$ Swiss style presented in [Howe 1998] and inspired by [Heuerding et al. 1996].

Let us consider the non intuitionistically valid sequent

$$\sigma_{\text{ex}} = [p_1 \rightarrow \bot, \ldots, p_n \rightarrow \bot \Rightarrow \bot] \quad (\text{Ex})$$

where the $p_i$’s are distinct propositional variables. The call $\mathcal{F}(u, \sigma_{\text{ex}})$ immediately stops signaling that $\sigma_{\text{ex}}$ is not intuitionistically valid. Indeed, $\text{act}^{\Rightarrow} (\sigma_{\text{ex}}) = \emptyset$ (for every $i \in \{1, \ldots, n\}$, $E(p_i, \sigma_{\text{ex}}) = F$) and $\sigma_{\text{ex}}$ is an irreducible sequent. On the other hand, with the calculus G3$^H$, to assert the unprovability of $\sigma_{\text{ex}}$ one has to chain up to $n$ applications of $\rightarrow L$ and build an history set containing all the $p_i$’s. Hence, the trees generated by our approach are shorter than those generated by G3$^H$. We note that $\sigma_{\text{ex}}$ is not even classically valid; however, a similar analysis applies to the classically valid but not intuitionistically valid sequent $[p_1 \rightarrow \bot, \ldots, p_n \rightarrow \bot \Rightarrow q \lor (q \rightarrow \bot)]$.

As another example, let us consider the following family of intuitionistically valid sequents ($n \geq 1$):

$$\sigma_{S_n} = [(\bigvee_{i=1}^n p_i \rightarrow \bot) \lor (\bigwedge_{i=1}^n p_i) \rightarrow \bot \Rightarrow \bot]$$

This family has been introduced in [Franzén 1988] and it has been used in [Buss and Ilemhoff 2003] to prove that the quadratic bound on the depth of derivations in Gentzen’s sequent calculi is optimal. Indeed, any G3i-derivation of $\sigma_{S_n}$ has a branch of length $\Omega(n^2)$ containing at least $n + 1$ applications of $\rightarrow L$. The following is the G3i-derivation of $\sigma_{S_2}$ generated by $\mathcal{F}(u, \sigma_{S_2})$ (we recall that $\neg Z = Z \rightarrow \bot$).

$$\sigma_{S_2} = [\neg A \Rightarrow \bot] \quad A = (\neg p_1 \lor \neg p_2) \lor (p_1 \land p_2)$$
In the history, we can only apply
in the expansion of a sequent of the kind $G_3$, be considered. Clearly, also in $G_3i$ the application of
sequent $\sigma$ backtracks, since the evaluation function cuts the failing branches; this holds for every
$p \lor$ sequent $\sigma$. — By properties of evaluation functions (in particular, by $R_i$, $E_j$, $H_k$),
the potential backtrack points generated during the construction of the above derivation correspond to the treatment of sequents $\sigma_2$, $\sigma_3$, $\sigma_6$, $\sigma_7$ and $\sigma_{10}$. However, independently of the used evaluation function $E$, the execution of $F(u, \sigma_{S_2})$ never has to backtrack in these points, indeed:

— The (backward) application of rule $\rightarrow L$ with main formula $\neg A$ to $\sigma_2$, $\sigma_3$, $\sigma_6$, $\sigma_7$ and $\sigma_{10}$ is forbidden since these sequents are processed in a blocked phase (they follow the left-premise of an $\rightarrow L$ application). Thus, we can only apply either $\lor R_0$ or $\lor R_1$.

— By properties of evaluation functions (in particular, by $(E_2)$, $(E_4)$, $(E_9)$, $(E_10)$, $(E_11)$),
it holds that:

$$E(p_1 \land p_2, \sigma_2) = F \quad E(p_1 \land p_2, \sigma_6) = F \quad E(\neg p_1, \sigma_7) = F \quad E(\neg p_1 \lor \neg p_2, \sigma_{10}) = F$$

Accordingly, to the sequents $\sigma_2$, $\sigma_6$, $\sigma_7$ and $\sigma_{10}$ we are forced to apply the rule $\lor R_k$
displayed in the derivation.

We point out that both $\lor R_0$ and $\lor R_1$ can be applied to $\sigma_3$. The procedure $F$ selects
$\lor R_0$, but the choice of $\lor R_1$ does not affect the behaviour; simply, the atoms $p_1$ and
$p_2$ are transferred to the left in a different order. We conclude that $F(u, \sigma_{S_2})$ never
backtracks, since the evaluation function cuts the failing branches; this holds for every
sequent $\sigma_{S_2}$ of the family.

The analysis of the $G_3^i$ procedure is more involved and a plethora of cases must
be considered. Clearly, also in $G_3^i$ the sequents $\sigma_2$, $\sigma_3$, $\sigma_6$, $\sigma_7$ and $\sigma_{10}$ are potential
backtrack points, since one of the rules $\rightarrow L$ (with main formula $\neg A$), $\lor R_0$ and $\lor R_1$
can be applied. Some applications of $\rightarrow L$ are forbidden by the history mechanism (we
cannot apply $\rightarrow L$ to a sequent $[\neg A, \Gamma \Rightarrow H]$ if $H$ is already in the history), but there is
no means to discard an application of $\lor R_0$ which would lead to a failure. For instance,
in the expansion of a sequent of the kind $\sigma' = [\neg A \Rightarrow A]$, with $A = (\neg p_1 \lor \neg p_2) \land (p_1 \lor p_2)$
in the history, we can only apply $\lor R_0$ or $\lor R_1$; if we select the latter rule, we get the
sequent $[\neg A \Rightarrow p_1 \land p_2]$ and, being the application of $\rightarrow L$ inhibited by the history, after
the application of $\lor R$ the search fails and it must backtrack. If $A$ is not in the history,
it is possible to build a $G_3^i$-derivation of $\sigma'$ with root rule $\lor R_1$. If this happens (for
instance, if $\sigma'$ is obtained at the first step after the application of $\rightarrow L$ to $\sigma_{S_2}$), we
everget a derivation of $\sigma_{S_2}$ with some redundancies. In general, the proof-search
procedure based on $G_3^i$ for $\sigma_{S_2}$ has to consider $n^2$ possible backtrack points. A similar
analysis applies to the version of $G_3^i$ using Scottish style histories of [Howe 1998];
however, due to a more clever management of histories, some useless branches are cut and the search space is in general narrower.

To conclude this comparison we notice that there are cases where the trees generated by $G^3_H$ are shorter than those generated by $\mathcal{F}$. This comes from the fact that the left rules of $G^3_H$ can be applied only if their application (read bottom-up) actually extends the left-hand side of the sequent. As an example, the rules for left conjunction of $G^3_H$ are

$\frac{[A_i, A_0 \land A_1, \Gamma \Rightarrow H]}{[A_0 \land A_1, \Gamma \Rightarrow H] \land L_i \text{ if } A_i \not\in \Gamma}$

Now, let $\sigma = [\Gamma, A_0, A_1, A_0 \land A_1 \Rightarrow H]$. In $G^3_H$ the application of the $\land L_i$ rules is prevented since both $A_i$’s belong to the left-hand side of the sequent. On the other hand, the invocation of $F(u, \sigma)$ develops the branch applying the rule for left conjunction. We can get the same behaviour of $G^3_H$ considering the refined proof-search procedure $F^Eref$ discussed above. Indeed, by properties ($E3$), ($E5$) and ($E6$), we get $E(A_0 \land A_1, [A_0, A_1, \Gamma \Rightarrow H]) = T$, hence $F^Eref(u, \sigma)$ does not apply the rule for left conjunction with main formula $A_0 \land A_1$. One can check that $F^Eref$ subsumes all the checks on rule applications of $G^3_H$. Moreover, there are cases where $F^Eref$ generates shorter trees than $G^3_H$; e.g., in the case where $E(A_0 \land A_1, [\Gamma \Rightarrow H]) = T$ even if $\{A_0, A_1\} \not\subseteq \Gamma$.

**Comparison with implication-locking**

Methods to control termination of proof-search based on “blocking” of formulas have been already investigated in the literature. In [Franzén 1988] it is introduced an implication-locking technique to limit two consecutive applications of $\Rightarrow L$ with the same main formula. After having backward applied $\Rightarrow L$ with main formula $A \Rightarrow B$, the formula $A \Rightarrow B$ is locked, i.e., it is not available for further applications of $\Rightarrow L$, until new information is transferred to the left-hand side of a sequent. This approach is extended to the first-order case in [Sahlin et al. 1992].

We can rephrase Franzén’s conditions as follows. Given a set of formulas $\Gamma$ and a formula $P$, $\Gamma$ covers $P$, written Cov$(\Gamma, P)$, iff $P$ has the form

$$P := G | P \land P | P \lor A | A \lor P | A \Rightarrow P$$

where $G \in \Gamma$ and $A$ is any formula. Rules for right implication are backward applied to $[\Gamma \Rightarrow C \Rightarrow D]$ as follows: if Cov$(\Gamma, C)$, then we apply $\Rightarrow R_1$, otherwise we apply $\Rightarrow R_2$.

In the latter case, the transfer of the formula $C$ to the left-hand side conveys new information (this is the major intuition beyond the cover relation), and this permits the unlocking of a left implication. To sum up, along a branch of a derivation, between two applications of $\Rightarrow L$ with the same main formula $A \Rightarrow B$, at least one application of $\Rightarrow R_2$ must occur. This yields a linear upper bound on the number of times that a left implication can be used in a branch of a derivation.

We can mimic this kind of implication locking in our framework defining an evaluation function $E_{cov}$ based on the cover relation Cov and its negative counterpart Cov$^-$ (the latter is needed to properly account for the value $F$). Let $\Gamma$ be a set of formulas and $N$ a formula; $\Gamma$ negatively covers $N$, written Cov$^-$$(\Gamma, N)$, iff $N \not\in \Gamma$ and $N$ has the form

$$N := \bot | q | N \land A | A \land N | N \lor N | P \Rightarrow N$$

where $q \in \mathcal{V}$, Cov$(\Gamma, P)$ and $A$ is any formula. Let $E_{cov}$ be defined as follows:

$$E_{cov}(A, [\Gamma \Rightarrow H]) = \begin{cases} 
T & \text{if Cov}(\Gamma, A) \\
F & \text{if Cov}^-(\Gamma, A) \\
X & \text{otherwise}
\end{cases}$$
It is easy to check that \( \varepsilon_{\text{cov}} \) satisfies properties (E1)–(E12) of Section 3, hence \( \varepsilon_{\text{cov}} \) is an evaluation function. Using \( \varepsilon_{\text{cov}} \), the blocking/unblocking actions on left implications performed by the function \( \mathcal{F} \) are similar to those described in [Franzén 1988]. Note however that in [Franzén 1988] blocking is applied to individual implicative formulas, whereas in our approach all the formulas in the left-hand side are simultaneously blocked and unblocked. We point out that \( \varepsilon_{\text{cov}} \) can be computed in linear time. Moreover, one can easily prove that \( \varepsilon_{\text{cov}} \) is a minimal evaluation function, i.e., for every evaluation function \( \varepsilon \), every sequent \( \sigma \) and every formula \( A \), \( \varepsilon_{\text{cov}}(A, \sigma) \neq \varepsilon(A, \sigma) \) implies \( \varepsilon_{\text{cov}}(A, \sigma) = \varepsilon(A, \sigma) \). It is easy to check that \( \bar{\varepsilon} \) is more powerful than \( \varepsilon_{\text{cov}} \) and it allows us to better drive proof-search. As an example, let us consider the non intuitionistically valid sequent \( \sigma \equiv [(a \rightarrow b) \lor p] \rightarrow q \Rightarrow (a \rightarrow b) \lor p \). We have that

\[
\begin{align*}
\varepsilon_{\text{cov}}((a \rightarrow b) \lor p, \sigma) &= X \\
\varepsilon_{\text{cov}}(a \rightarrow b, \sigma) &= X \\
\varepsilon_{\text{cov}}(p, \sigma) &= F
\end{align*}
\]

Hence, using the evaluation function \( \varepsilon_{\text{cov}} \), \( \mathcal{F} \) has two possible rules to apply to \( \sigma \): rule \( \rightarrow L \) with main formula \( ((a \rightarrow b) \lor p) \rightarrow q \) and rule \( \lor R_0 \). On the other hand, using evaluation \( \bar{\varepsilon} \), \( \mathcal{F} \) can only apply rule \( \lor R_0 \).

A mechanism similar to the one described in [Franzén 1988] to freeze formulas in the left-hand side of the sequents is also used in [Buss and Imhof 2003]. Here, the aim is to show that intuitionistic valid sequents admit G3i-derivations having quadratic depth in the size of the root sequent and that such a bound is optimal. Also in this case, the main issue consists in providing upper (and lower) bounds on the reuse of left implications.

An application of implication-locking techniques to mG3i, the multi-succedent version of G3i, is discussed in [Corsi and Tassi 2007]. The proof-search strategy of [Corsi and Tassi 2007] is similar to [Buss and Imhof 2003; Franzén 1988] and can be informally summarized as follows (below, \( \rightarrow R_1 \), \( \rightarrow R_2 \) and \( \rightarrow L \) refer to the multi-succedent presentation of the rules). In expanding a branch: (1) between two applications of \( \rightarrow L \) with the same main formula at least one application of \( \rightarrow R_2 \) must occur; (2) two applications of \( \rightarrow R_2 \) with the same main formula \( C \rightarrow D \) are not allowed. Point (2) prevents the transfer of the antecedent \( C \) more than once; thus, after the first application of \( \rightarrow R_2 \), we must use rule \( \rightarrow R_1 \) (called a fortiori rule in [Corsi and Tassi 2007]), which maintains \( D \) on the right and discards \( C \). Using this strategy, we again get a linear bound on the use of left implications. We remark that the proof-search strategies for G3i and mG3i are quite different since the two calculi have different sets of non-invertible rules. Thus, to apply our framework to mG3i, we should heavily modify both the proof-search and the counter-model construction strategy.

**Focusing**

As for a comparison with focusing techniques [Dyckhoff and Lengrand 2006; Miller and Pimentel 2013] we remark that, even if the b-phase of our decision procedure resembles a right-focused phase of a focused calculus, there are few relations between the two approaches. Indeed, focusing techniques reduce the search space limiting the use of contraction, but they do not guarantee termination of proof-search. As for the right-focused calculus \( LJQ \) of [Dyckhoff and Lengrand 2006], its focused phase ends when a right rule for implication is applied; in our approach a b-phase terminates only if, applying the right rule for implication to \( \sigma = [\Gamma \Rightarrow A \rightarrow B], \varepsilon(A, \sigma) \neq T \) holds. As a result, our proof-search procedure is a refinement of the one supported by \( LJQ \).
Contraction free calculi
As discussed in the introduction, contraction-free calculi [Vorob’ev 1970; Dyckhoff 1992; Hudelmaier 1993; Miglioli et al. 1997; Ferrari et al. 2009; 2013a] are the alternative approach to get terminating calculi for intuitionistic propositional logic. We remark, that differently from $G3i$, these calculi either lack the subformula property, as those in [Vorob’ev 1970; Dyckhoff 1992; Hudelmaier 1993; Miglioli et al. 1997; Ferrari et al. 2009], or are based on a non-standard notion of sequent, as the one in [Ferrari et al. 2013a]. The calculi in [Vorob’ev 1970; Dyckhoff 1992; Miglioli et al. 1997] generate derivations that can have exponential depth in the size of the formula to be proved (see the discussion in [Hudelmaier 1993]) and hence are not adequate for proof-search. On the other hand, the calculi in [Hudelmaier 1993; Ferrari et al. 2009; 2013a], which generate derivations of linear depth, lack a well-developed proof-theory (e.g., they do not have a natural computational interpretation) and are difficult to treat in proof-standardization frameworks as the one based on focusing [Chihani et al. 2013].

The calculus $Gbu$
A first application of evaluation functions as a tool to avoid loop-checking is presented in [Ferrari et al. 2013b], however the results of this paper are essentially different. First of all, the calculus $Gbu$ used in [Ferrari et al. 2013b] is a labelled variant of $G3i$. Secondly, the notion of evaluation used in that paper is weaker than the one presented here. Indeed, the evaluations discussed in [Ferrari et al. 2013b] only consider left-hand sides of sequents (they lack the value $F$) and are only used in the application of rules for right implication. Here, we refine the definition, so that evaluations are also used to avoid useless applications of $	o L$ and $\lor R$. To give an example, let us consider the sequent $\sigma_{ex}$ displayed in (Ex). The proof-search procedure for $Gbu$ must try an application of $\to L$ for every implication $p_i \to \bot$ in $\sigma_{ex}$ before discovering that $\sigma_{ex}$ is unprovable, while, as discussed above, $F(u, \sigma_{ex})$ immediately stops returning an RG3i-derivation. For other examples, one can compare the trace of the execution of the proof-search procedure described in examples 7.1 and 7.2 with the corresponding examples in [Ferrari et al. 2013b]. In both cases $F$ cuts some backtrack points and, in general, the search-space of $F$ is narrower than that of [Ferrari et al. 2013b].

Future work
We plan to study the application of evaluation functions to the first-order case, where they can be used to reduce the proof-search space. In the propositional setting, an interesting question is if evaluation functions can be defined by purely syntactical conditions. To answer this question, we are analyzing the relations between evaluation functions and the proof-theoretical properties of $G3i$. We are also investigating the role of evaluation functions in the context of natural deduction calculi. Another interesting issue is if evaluation functions can be used to get terminating decision procedures for other intermediate and modal logics requiring loop-checking as, e.g., $S4$ [Troelstra and Schwichtenberg 2000].

We have implemented the decision procedure $F$ and its variants discussed in the conclusions in the prover g3ied available at http://www.dista.uninsubria.it/~ferram.

ACKNOWLEDGMENTS
We are grateful to the anonymous referees for their helpful suggestions and comments.

REFERENCES

ACM Transactions on Computational Logic, Vol. V, No. N, Article A, Publication date: January YYYY.
An Evaluation-Driven Decision Procedure for G3i


