Duality between Unprovability and Provability in Forward Refutation-search for Intuitionistic Propositional Logic

CAMILLO FIORENTINI, Dipartimento di Informatica, Università degli Studi di Milano, Italy
MAURO FERRARI, Dipartimento di Scienze Teoriche e Applicate, Università degli Studi dell’Insubria, Italy

The inverse method is a saturation-based theorem-proving technique; it relies on a forward proof-search strategy and can be applied to cut-free calculi enjoying the subformula property. Here, we apply this method to derive the unprovability of a goal formula \( G \) in Intuitionistic Propositional Logic. To this aim we design a forward calculus \( \text{FRJ}(G) \) for Intuitionistic unprovability, which is appropriate for constructively ascertaining the unprovability of a formula \( G \) by providing a concise countermodel for it; in particular, we prove that the generated countermodels have minimal height. Moreover, we clarify the role of the saturated database obtained as result of a failed proof-search in \( \text{FRJ}(G) \) by showing how to extract from such a database a derivation witnessing the Intuitionistic validity of the goal.

CCS Concepts: • Theory of computation → Automated reasoning; Proof theory;

Additional Key Words and Phrases: Proof-search procedures, intuitionistic propositional logic, sequent calculi

ACM Reference format:
Camillo Fiorentini and Mauro Ferrari. 2020. Duality between Unprovability and Provability in Forward Refutation-search for Intuitionistic Propositional Logic. ACM Trans. Comput. Logic 21, 3, Article 22 (March 2020), 47 pages.
https://doi.org/10.1145/3372299

1 INTRODUCTION
The inverse method, introduced in the 1960s by Maslov [23], is a saturation-based theorem-proving technique closely related to (hyper)resolution [7]; it relies on a forward proof-search strategy and can be applied to cut-free calculi enjoying the subformula property. Given a goal, a set of instances of the rules of the calculus at hand is selected; such specialized rules are repeatedly applied in the forward direction, starting from the axioms (i.e., the rules without premises). Proof-search terminates if either the goal is obtained or the database of proved facts saturates (no new fact can be added). As pointed out by Lifschitz [22], “The role of the inverse method in the Soviet work on proof procedures for predicate logic can be compared to the role of resolution method in theorem proving projects in the West.” But, he regrets, “for a number of reasons, this work has not been duly appreciated outside a small circle of Maslov’s associates.” The method has been popularized by Degtyarev and Voronkov [7], who provide the general recipe to design forward calculi, with applications to

Authors’ addresses: C. Fiorentini, Dipartimento di Informatica, Università degli Studi di Milano, Via Celoria 18, Milano 20133, Italy; email: fiorentini@di.unimi.it; M. Ferrari, Dipartimento di Scienze Teoriche e Applicate, Università degli Studi dell’Insubria, Padiglione Rossi, Via O. Rossi 9, Varese, 21100, Italy; email: mauro.ferrari@uninsubria.it.
Permissions to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2020 Association for Computing Machinery.
1529-3785/2020/03-ART22 $15.00
https://doi.org/10.1145/3372299

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
Classical Predicate Logic and some non-classical logics. Further extensions can be found in References [2, 8, 21]. A significant investigation is presented in References [4, 5], where focused calculi and polarization of formulas are exploited to reduce the search space in forward proof-search for Intuitionistic Logic. These techniques are at the heart of the design of the prover Imogen [24].

In all the mentioned papers, the inverse method has been exploited to prove the validity of a goal in a specific logic. Here, we follow the dual approach, namely: We design a forward calculus to derive refutations asserting the unprovability of a goal formula in Intuitionistic Propositional Logic (IPL). Our motivation is twofold. First, we aim to define a refutation calculus that constructively ascertains the unprovability of a formula by providing a concise countermodel for it. The second motivation is to clarify the role of the saturated database obtained when the search for a refutation (refutation-search) fails. In the case of the usual forward calculi for Intuitionistic provability, if proof-search fails, than a saturated database is generated that “may be considered a kind of countermodel for the goal sequent” [24]. However, as far as we know, no method has been proposed to effectively extract it. Actually, the main problem comes from the high level of non-determinism involved in the construction of countermodels. Here, assuming the dual approach, the saturated database generated by a failed refutation-search can be considered as “a kind of proof of the goal”; we give evidence of this by showing how to extract from such a database a derivation witnessing the Intuitionistic validity of the goal.

Our different perspectives requires a deep adjustment of the inverse method itself. Sequent $\Gamma \vdash C$ of standard forward calculi encode the fact that the right formula $C$ is provable from the set of left formulas $\Gamma$ in the understood logic. In our approach, a sequent $\Gamma \not\vdash C$ signifies the unprovability of $C$ from $\Gamma$ in IPL. From a semantic viewpoint, this means that, in some world of a Kripke model, all the formulas in $\Gamma$ are forced and $C$ is not forced. In standard forward reasoning, axioms have the form $p \vdash p$, where $p$ is a propositional variable. In our approach, axioms have the form $\Gamma_{At} \not\vdash p$ or $\Gamma_{At} \not\vdash \bot$, where $\Gamma_{At}$ is a set of propositional variables and $p$ is a propositional variable not belonging to $\Gamma_{At}$. Rules must preserve (top-down) unprovability. Examples of sound rules for unprovability are

$\frac{\Gamma \not\vdash A}{\Gamma \not\vdash A \land B} \quad R\land, \quad \frac{A, \Gamma \not\vdash C}{A \lor B, \Gamma \not\vdash C} \quad L \lor.
$

The former rule states that if $A$ is not provable from $\Gamma$, then $A \land B$ is not provable from $\Gamma$. The latter corresponds to the contrapositive of Inversion Principle for left $\lor$: if $C$ is not provable from $\{A\} \cup \Gamma$, then $C$ is not provable from $\{A \lor B\} \cup \Gamma$. The tricky point is how to cope with rules having more than one premise. In direct forward calculi, left formulas must be gathered. For instance, in sequent calculi with independent contexts [29], the rule for right $\land$,

$\frac{\Gamma_1 \vdash A \quad \Gamma_2 \vdash B}{\Gamma_1, \Gamma_2 \vdash A \land B} \quad R\land,$

encodes the property that if $A$ is provable from $\Gamma_1$ and $B$ is provable from $\Gamma_2$, then $A \land B$ is provable from $\Gamma_1 \cup \Gamma_2$. Note that we restrict our attention to the case of independent contexts, since in the perspective of applying forward reasoning, where premises are generated before the application of the rule, a context-sharing rule requiring $\Gamma_1 = \Gamma_2$ is not adequate. To handle rules having more than one premise in the forward refutation calculus, at a first sight, we should follow the dual approach and intersect left formulas. Thus, the rule $L\lor$ should be

$\frac{\Gamma_1 \not\vdash A \quad \Gamma_2 \not\vdash B}{\Gamma_1 \cap \Gamma_2 \not\vdash A \lor B} \quad R\lor,$

Note that our use of the term refutation is different from the one in the context of resolution, where it is about establishing that False is entailed in all models.
Duality between Unprovability and Provability in Forward Refutation-search for IPL

22:3

to be interpreted as “if A is not provable from \( \Gamma_1 \) and B is not provable from \( \Gamma_2 \), then \( A \lor B \) is not provable from \( \Gamma_1 \cap \Gamma_2 \).” But the alleged rule \( R \lor \) does not preserve unprovability, as shown by this trivial counterexample:

\[
\frac{q_2, p, H \not\Rightarrow q_1}{p, H \not\Rightarrow q_1 \lor q_2} \quad \frac{p, H \not\Rightarrow q_2}{R \lor, \quad H = p \supset q_1 \lor q_2.}
\]

Here, \( q_1 \) is not provable from \( \Gamma_1 = \{q_2, p, H\} \) and \( q_2 \) is not provable from \( \Gamma_2 = \{q_1, p, H\} \), but the right formula of the conclusion \( q_1 \lor q_2 \) is provable from \( \Gamma_1 \cap \Gamma_2 = \{p, H\} \). The drawback is that the conclusion cannot retain both \( p \) and \( H \) in left. To get a sound rule, we have to select a suitable subset \( \Gamma_0 \) of \( \Gamma_1 \cap \Gamma_2 \): the possible choices are \( \Gamma_0 = \{p\} \) or \( \Gamma_0 = \{H\} \) or \( \Gamma_0 = \emptyset \). Thus, we need a more clever strategy to join sequents and to treat left formulas in multi-premise rules. In addition to the sequents mentioned so far, we call regular sequents, we introduce irregular sequents of the kind \( \Sigma: \Theta \rightarrow C \), where the left formulas are partitioned into two sets \( \Sigma \) and \( \Theta \).

The definition of the rules of a forward calculus depends on the formula to be proved (the goal formula). The calculus we define is parametrized by the goal formula \( G \) (where the goal is to prove that \( G \) is not valid in IPL). We call the related calculus \( \text{FRJ}(G) \) (Forward Refutation calculus for IPL parametrized by \( G \)); formulas occurring in the sequents of \( \text{FRJ}(G) \) are suitable subformulas of \( G \). The rules of the calculus are shown in Figure 1 and discussed in Section 3. In Section 4, we define a forward refutation-search procedure to build an \( \text{FRJ}(G) \)-refutation of a goal formula \( G \), namely, an \( \text{FRJ}(G) \)-refutation of a regular sequent of the form \( \Gamma \not\Rightarrow G \). This is a standard saturation procedure where the derivable sequents of \( \text{FRJ}(G) \) are collected step-by-step in a database \( D_G \). To avoid redundancies and maintain \( D_G \) compact, we introduce a subsumption relation between sequents; for instance, if at some step \( \sigma \) is proved and \( \sigma \) is subsumed by a sequent already in \( D_G \), then \( \sigma \) is discarded and not added to \( D_G \) (forward subsumption).

If the formula \( G \) is valid in IPL, then refutation-search for \( G \) fails (indeed, no \( \text{FRJ}(G) \)-refutation of \( G \) can be built) and we eventually get a saturated database \( D_G \) for \( G \). This means that for every sequent \( \sigma \) derivable in \( \text{FRJ}(G) \), \( D_G \) contains a sequent \( \sigma' \) that subsumes \( \sigma \); thus, \( D_G \) is in some sense representative of all the sequents derivable in \( \text{FRJ}(G) \). We can exploit \( D_G \) to build a derivation of \( G \) in a sequent calculus for IPL. To this aim, in Section 5 we introduce the sequent calculus \( \text{Gbu}(G) \) (see Figure 9), a variant of the well-known sequent calculus \( \text{G3i} \) [29]. From a \( \text{Gbu}(G) \)-derivation of \( G \), we can immediately obtain a \( \text{G3i} \)-derivation of \( G \). Differently from \( \text{G3i} \), backward proof-search in \( \text{Gbu}(G) \) always terminates; indeed, we can define a weight function on sequents such that, after the backward application of a rule of \( \text{Gbu}(G) \) to a sequent, the weight of the sequents decreases. Nonetheless, backward-proof search in \( \text{Gbu}(G) \) might present several backtrack points, in correspondence of the applications of rules for left implication and right disjunction. The crucial point is that we can remove backtracking by exploiting the database \( D_G \): in presence of multiple non-deterministic choices, we query \( D_G \) as an oracle to select the right way so to successfully continue proof-search. Thus, we can consider \( D_G \) as a proof-certificate of the validity of \( G \), in the sense that it contains enough information to reconstruct a derivation of \( G \) in the sequent calculus \( \text{G3i} \). If we eliminate from \( D_G \) all the redundancies (if \( \sigma \) belongs to \( D_G \), then remove from \( D_G \) all the sequents subsumed by \( \sigma \)), then we get a saturated database \( D_G^* \) that is the minimum among the saturated databases of \( G \); hence, we can consider \( D_G^* \) as the canonical proof-certificate of the validity of \( G \). To get the minimum saturated database, we have to enhance the refutation-search procedure by implementing backward subsumption. As we discuss on a concrete example at the end of Section 5, the backtracking-free proof-search in \( \text{Gbu}(G) \) driven by a saturated database can be more efficient than the usual backward proof-search procedure in \( \text{G3i} \).

The rules of \( \text{FRJ}(G) \) are inspired by Kripke semantics. In Section 3.2, we show that, from a refutation of \( G \), we can extract a countermodel for \( G \), namely, a Kripke model such that, at its root,
the formula $G$ is not forced, witnessing that $G$ is not valid in IPL [3]. Actually, there is a close correspondence between a refutation and the related Kripke model. Thus, our forward refutation-search procedure can be understood as a top-down method to build a countermodel for $G$, starting from the final worlds down to the root. This original approach is dual to the standard one, where countermodels are built bottom-up, mimicking the backward application of rules (see, e.g., References [1, 6, 9, 10, 12, 17, 18, 25, 26]). This different viewpoint has a significant impact in the outcome. Indeed, the countermodels generated by a backward procedure are always trees, which might contain some redundancies. Instead, forward methods re-use sequents and do not replicate them; thus, the generated models are DAGs (Direct Acyclic Graphs) not containing duplications and are in general very concise (see the compact models in Figures 6 and 12). In Section 3 we show that FRJ($G$)-refutations have height quadratic in $|G|$ (the size of $G$). Moreover, if $G$ is not valid, then the countermodel extracted from an FRJ($G$)-refutation of $G$ has height at most $|G|$.

The relationship between a non-valid formula $G$ and the height of the countermodel extracted from an FRJ($G$)-refutation of $G$ is deeply investigated in Section 6. There we show that, given a countermodel for $G$ of height $h$, we can build an FRJ($G$)-refutation of $G$ having height at most $h$. By this fact, we conclude that, if $G$ is not valid in IPL, we can build an FRJ($G$)-refutation of $G$ such that the height $h$ of the extracted countermodel is minimal (namely, there exists no countermodel for $G$ having height less than $h$). We can tweak the refutation-search procedure so that, if $G$ is not valid, then it yields an FRJ($G$)-refutation of $G$ such that the extracted countermodel has minimal height.

To evaluate the potential of our approach, we have implemented frj, a Java prototype of our refutation-search procedure based on the JTabWb framework [13]. frj implements term-indexing, forward and backward subsumption, and it allows the user to generate the rendering of proofs and of the extracted countermodels.

The idea of applying forward-refutation to prove the non-validity of formulas has been introduced in Reference [16]. There we present the calculus FRJ($G$), we sketch a proof of soundness and a semantic proof of completeness, moreover, we introduce a preliminary version of the implementation frj, using backward and forward subsumption. The first part of the present article (Sections 2 and 3) recalls and improves Reference [16], providing more insights and more detailed proofs of the properties of FRJ($G$). Saturated databases (Section 4) and their use in proof-search (Section 5) is one of the original part of the present article; the dual calculus Gbu($G$) described in Section 5 is closely related to the calculus Gbu introduced in Reference [11] to get a terminating proof-search procedure for the sequent calculus G3i. The discussion in Section 6 about the bounds of the height of refutations and the height of countermodels extracted from refutations is a new contribution of the present article.

2 PRELIMINARIES

We consider the propositional language $\mathcal{L}$ based on a denumerable set of propositional variables $\mathcal{V}$, the connectives $\land$, $\lor$, $\rightarrow$ (as usual, $\land$ and $\lor$ bind stronger than $\rightarrow$) and the logical constant $\bot$. $\neg A$ is a shorthand for $A \rightarrow \bot$. If $\odot$ is a logical connective, then we call $\odot$-formula a formula with top-level connective $\odot$. By $\mathcal{V}^\odot$, we denote the set $\mathcal{V} \cup \{\bot\}$ and by $\mathcal{L}^\rightarrow$ the set of $\rightarrow$-formulas of $\mathcal{L}$. Capital Greek letters $\Gamma$, $\Sigma, \ldots$ denote sets of formulas; we use notations like $\Gamma^{\land}$ and $\Gamma^{\rightarrow}$ to mean that $\Gamma^{\land} \subseteq \mathcal{V}$ and $\Gamma^{\rightarrow} \subseteq \mathcal{L}^\rightarrow$. Given a formula $G$, $\text{Sf}(G)$ is the set of all subformulas of $G$ (including $G$ itself). By $\text{Sl}(G)$ and $\text{Sr}(G)$, we denote the subsets of left and right subformulas of $G$ (a.k.a. negative/positive subformulas of $G$ [29]). Formally, $\text{Sl}(G)$ and $\text{Sr}(G)$ are the smallest subsets of $\text{Sf}(G)$ such that:

--frj is available at http://github.com/ferram/jtabwb_provers/.

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
• $G \in \text{Sr}(G)$;
• $A \otimes B \in \text{Sx}(G)$ implies $\{A, B\} \subseteq \text{Sx}(G)$, where $\otimes \in \{\land, \lor\}$ and $\text{Sx} \in \{\text{Sl}, \text{Sr}\}$;
• $A \supset B \in \text{St}(G)$ implies $B \in \text{St}(G)$ and $A \in \text{Sr}(G)$;
• $A \supset B \in \text{Sr}(G)$ implies $B \in \text{Sr}(G)$ and $A \in \text{St}(G)$.

A Kripke model is a structure $\mathcal{K} = (P, \leq, \rho, V)$, where $(P, \leq)$ is a finite poset with minimum $\rho$ (the root of $\mathcal{K}$) and $V : P \rightarrow 2^V$ is a function such that $\alpha \leq \beta$ implies $V(\alpha) \subseteq V(\beta)$. The forcing relation $\models$ is defined as follows:

- $\mathcal{K}, \alpha \not\models \bot$;
- 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 $\mathcal{K}, \alpha \models B$;
- $\mathcal{K}, \alpha \models A \supset B$ iff, for every $\beta \in P$ such that $\alpha \leq \beta$, $\mathcal{K}, \beta \not\models A$ or $\mathcal{K}, \beta \models B$.

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 $\mathcal{K}, \rho \models A$; we say that $A$ is valid if $A$ is valid in all the Kripke models; Intuitionistic Propositional Logic IPL coincides with the set of valid formulas [3]. If $\mathcal{K}, \rho \not\models A$, then we say that $\mathcal{K}$ is a countermodel for $A$. A final world $\gamma$ of $\mathcal{K}$ is a maximal world in $(P, \leq)$; for every final world $\gamma$ and every classically valid formula $A$, we have $\mathcal{K}, \gamma \models A$. Let $\Gamma$ be a set of formulas, by $\mathcal{K}, \alpha \models \Gamma$, we mean that $\mathcal{K}, \alpha \models A$ for every $A \in \Gamma$. Using the above notation, we avoid to mention the model $\mathcal{K}$ whenever it is understood (e.g., we write $\alpha \models A$ instead of $\mathcal{K}, \alpha \models A$); moreover, by “model” we mean “Kripke model.”

The closure of $\Gamma$, denoted by $\text{Cl}(\Gamma)$, is the smallest set containing the formulas $X$ defined by the following grammar:

$$X ::= C \mid X \land X \mid A \lor X \mid X \lor A \mid A \supset X, \quad C \in \Gamma, A \text{ any formula}.$$  

The following properties of closures can be easily proved:

(C1) $\mathcal{K}, \alpha \models \Gamma$ implies $\mathcal{K}, \alpha \models \text{Cl}(\Gamma)$.
(C2) $A \in \text{Cl}(\Gamma)$ implies $A \in \text{Cl}(\Gamma \cap \text{St}(A))$.
(C3) $\Gamma \subseteq \text{Cl}(\Gamma)$ and $\text{Cl}(\text{Cl}(\Gamma)) = \text{Cl}(\Gamma)$.
(C4) $\Gamma_1 \subseteq \Gamma_2$ implies $\text{Cl}(\Gamma_1) \subseteq \text{Cl}(\Gamma_2)$.
(C5) $\text{Cl}(\Gamma) \cap \text{V} = \Gamma \cap \text{V}$.
(C6) $\Gamma_1 \subseteq \text{Cl}(\Gamma_2)$ implies $\text{Cl}(\Gamma_1) \subseteq \text{Cl}(\Gamma_2)$ (this follows from (C3) and (C4)).

3 THE CALCULUS FRJ(G)

The Forward Refutation calculus FRJ(G) is a calculus to infer the unprovability of a formula $G$ (the goal formula) in IPL and it is designed to support forward refutation-search. The calculus acts on FRJ(G)-sequents that depend on the subformulas of $G$. Throughout the article, we use the following notation:

$$\Gamma^A = \text{St}(G) \cap \text{V}, \quad \Gamma^C = \text{St}(G) \cap \text{L}^C, \quad \Gamma = \Gamma^A \cup \Gamma^C.$$

There are two types of FRJ(G)-sequents, we call regular (arrow $\Rightarrow$) and irregular (arrow $\rightarrow$), defined as follows:

- **regular sequents** have the form $\Gamma \Rightarrow C$, where $\Gamma \subseteq \overline{\Gamma}$ and $C \in \text{Sr}(G)$;
- **irregular sequents** have the form $\Sigma ; \Theta \rightarrow C$, where $\Sigma \cup \Theta \subseteq \overline{\Gamma}$ and $C \in \text{Sr}(G)$.

---

3In refutation calculi sequents are sometimes called anti-sequents (see, e.g., Reference [26]).
Given a sequent $\sigma$, the set $\text{Lhs}(\sigma)$ of left formulas of $\sigma$ and the right formula of $\sigma$ are defined as follows:

$$\text{Lhs}(\sigma) = \begin{cases} 
\Gamma & \text{if } \sigma \text{ is regular} \\
\Sigma \cup \Theta & \text{if } \sigma \text{ is irregular}
\end{cases}, \quad \text{Rhs}(\sigma) = C.$$ 

We remark that the left formulas of $\sigma$ are left subformulas of $G$ and the right formula of $\sigma$ is a right subformula of $G$; accordingly, the number of $\text{FRJ}(G)$-sequents is finite. Left formulas of irregular sequents $\sigma$ are partitioned into the sets $\Sigma$, the stable set of $\sigma$, and $\Theta$ the non-stable set of $\sigma$.

Rules of the calculus $\text{FRJ}(G)$ are collected in Figure 1. They consist of two axiom rules, some right introduction rules for the connectives $\land$, $\lor$, $\supset$ and the rules $\triangleright_{\text{At}}$ and $\triangleright_{\text{V}}$ to join sequents. In forward refutation-search, rules are applied top-down in all possible ways to the sequents at hand until no new sequent can be generated. To avoid that the number of proved sequents blows-up, we have designed the calculus so to minimize the number of rules. In particular, we have circumvented the use of left rules, which are in essence simulated by the side conditions involving the closure operator $\text{Cl}(.)$. For instance, let us consider the following applications of rule $\triangleright_{\epsilon}$, which introduce...
the formula $A_1 \land A_2 \supset B$ in the right-hand side of the conclusion:

\[
\frac{\sigma_1 = A_1, A_2; \vdash B}{A_1; A_2 \vdash A_1 \land A_2 \supset B} \quad \vdash_e, \quad \frac{\sigma_2 = \cdot; A_1, A_2 \supset B}{A_1, A_2; \cdot \vdash A_1 \land A_2 \supset B} \quad \vdash_e, \quad \frac{\sigma_3 = A_1, A_2; \vdash B}{A_1, A_2; \vdash \vdash A_1 \land A_2 \supset B} \quad \vdash_e.
\]

In all the three cases there is an implicit application of a left rule for $\land$ to get $A_1 \land A_2$ in the left. Since $A_1 \land A_2 \in C(l((A_1, A_2)))$, the formula $A_1 \land A_2$ is implicitly contained in the left of the premises and an explicit application of rule $L \land$ can be avoided. The use of explicit left rules would require the introduction of three rules for $L \land$ to cover the above displayed applications: a $L \land$ rule acting on stable sets in the case of $\sigma_1$, one acting on the non-stable set in the case of $\sigma_2$, and one acting on the mixing of the two in the case of $\sigma_3$. Note that, being the set of $FRJ(G)$-sequents finite, the calculus $FRJ(G)$ satisfies the Finite Rule Property [7], namely: It has a finite number of axioms and, given a finite number of $FRJ(G)$-sequents, there is only a finite number of inferences of $FRJ(G)$ applicable to these sequents.

We introduce the following notation:

- $D$ is an $FRJ(G)$-refutation of $\sigma$ iff $D$ is a formal structure built up according with the rules of $FRJ(G)$ having $\sigma$ as root sequent;
- $D$ is an $FRJ(G)$-refutation of $G$ if there exists a (possibly empty) set of formulas $\Gamma$ such that $D$ is an $FRJ(G)$-refutation of $\Gamma \Rightarrow G$;
- We write $\vdash_{FRJ(G)} \sigma$ iff there exists an $FRJ(G)$-refutation of $\sigma$;
- We write $\vdash_{FRJ(G)} G$ iff there exists an $FRJ(G)$-refutation of $G$.

Soundness of $FRJ(G)$ is stated as follows:

**Theorem 3.1 (Soundness of $FRJ(G)$).** $\vdash_{FRJ(G)} G$ implies $G \notin IPL$.

Theorem 3.1 is a consequence of the fact that the calculus $FRJ(G)$ satisfies the following soundness property concerning regular sequents:

- Property $(SREG)$ if $\vdash_{FRJ(G)} \Gamma \Rightarrow C$, then there exists a world $\alpha$ of a model such that $\alpha \models \Gamma$ and $\alpha \not\models C$.

Property $(SREG)$ follows by Lemma 3.10 of Section 3.3. By $(SREG)$, if there exists a refutation of the sequent $\Gamma \Rightarrow C$, then the formula $(\land \Gamma) \supset C$ is not valid in IPL. Note that soundness of $FRJ(G)$ hides subsumption, which is typical of forward reasoning. Indeed, let us assume that $\vdash_{FRJ(G)} G$. Then, there exists an $FRJ(G)$-refutation of a sequent $\Gamma \Rightarrow G$; hence, by $(SREG)$, the formula $(\land \Gamma) \supset G$ is not valid; obviously, this implies that $G$ is not valid.

Property $(SREG)$ explains the role of regular sequents: $\Gamma \Rightarrow C$ directly corresponds to a world $\alpha$ of a model $\mathcal{K}$ such that $\alpha \models \Gamma$ and $\alpha \not\models C$. Since refutations are built top-down, starting from the axioms, models are built top-down as well, starting from the final worlds. A regular axiom $\Gamma^A \setminus \{F\} \Rightarrow F$ corresponds to a final world $\alpha$ of the model such that $V(\alpha) = \Gamma^A \setminus \{F\}$, hence $\alpha \models \Gamma^A \setminus \{F\}$ and $\alpha \not\models F$. The role of the join rules of the calculus is to formalize top-down expansion of a model, which is the crucial point of the forward countermodel construction. Let us assume that the model already contains a set of worlds $P$ (the upper fragment of the countermodel), and we have to add a new world $\alpha$ closer to the root to be found (the world of the countermodel falsifying $G$). This task is far from being trivial, and the standard techniques used in bottom-up construction of countermodels cannot be adapted to this scope. In our calculus a top-down expansion step is performed by the join rules, which “join” some of the worlds $\alpha_1, \ldots, \alpha_n$ already in $P$ with a new world $\alpha$: $\alpha$ is the world represented by the conclusion of the rule application and $\alpha_1, \ldots, \alpha_n$ become the immediate successors of $\alpha$. Only using regular sequents, the definition of join rules would be very cumbersome and unmanageable. To get a sensible presentation, we introduce irregular sequents, which do not directly correspond to worlds of models, but represent intermediate steps.
towards a join operation. We highlight some of the critical points in join operations. Let \( \sigma = \Gamma \not\not\not\not\not\not\not\not C \) be the conclusion of a join rule. To guarantee the existence of a new world \( \alpha \) associated with \( \sigma \) (see Property (SREG)), we need that, for every \( A \supset B \in \Gamma, \alpha \not\not\not\not\not\not\not B \). To get this, the condition (J2) of join rules dictates that at least one of the premises has the form \( \sigma_1 = \Sigma_1; \Theta_1 \rightarrow A \). If \( A \) does not contain \( \supset \), then by considering the chain of irregular sequents used to prove \( \sigma_1 \), one can easily prove that \( \alpha \not\not\not\not\not\not\not B \). The critical case is when \( A = B \supset C \) and \( \sigma_1 \) is the conclusion of an application of \( \supset \). Indeed, it is not sufficient to guarantee that \( \alpha \not\not\not\not\not\not\not C \), but we have also to impose that \( \alpha \not\not\not\not\not\not\not B \), a demanding requirement. This is why rule \( \supset \) moves in the stable part \( \Sigma \) of the conclusion some of the formulas in the non-stable region \( \Theta \) of the premise: formulas in \( \Sigma \) are the formulas that must be forced in the new state \( \alpha \) generated by a join rule to guarantee the non-forcing of \( A \supset B \) in \( \alpha \). Moreover, the formulas in \( \Sigma \) must be downward preserved until the next application of a join rule. We also point out that, since forcing is monotone w.r.t. \( \leq \), formulas moved in the stable part of sequents must be forced in all the successors of \( \alpha \) (which, in the top-down process, have already been generated). The role of the non-stable part \( \Theta \) of irregular sequents is to provide a sound source from which formulas can be picked. A formal proof that join rules preserve the semantics of regular sequents is given in Lemma 3.10.

We provide some insights on the peculiar rules of \( \text{FRJ}(G) \).

**Rule \( \supset \) (Regular Sequents).** In standard refutation calculi (see, e.g., Reference [27]), the rule for right implication is

\[
\frac{\Gamma \not\not\not\not\not\not\not B}{\Gamma \not\not\not\not\not\not\not A \supset B} \quad R \quad A \in \Gamma.
\]

The side condition \( A \in \Gamma \) is needed to guarantee that, assuming that \( B \) is not provable from \( \Gamma \), then \( A \supset B \) is not provable from \( \Gamma \) as well. As discussed above, we impose \( A \in Cl(\Gamma) \) as side condition, instead of the usual \( A \in \Gamma \), to make up for the lack of left introduction rules.

**Rule \( \supset \) (Irregular Sequents).** In the case of an irregular sequent \( \sigma_1 = \Sigma; \Theta' \rightarrow B \) as premise, the rule \( \supset \) can be applied if the antecedent \( A \) of the implication belongs to \( Cl(\Sigma') \), where \( \Sigma' = \Sigma \cup \Lambda \) is obtained by extending \( \Sigma \) with a (possibly empty) subset \( \Lambda \) of \( \Theta' \). The application of the rule can be displayed as follows:

\[
\frac{\sigma_1 = \Sigma; \Theta', \Lambda \rightarrow B}{\sigma = \Sigma, \Lambda; \Theta \rightarrow A \supset B} \quad \supset \quad \Theta \cap \Lambda = \emptyset \quad A \in Cl(\Sigma \cup \Lambda)
\]

The set \( \Theta' \) has been partitioned as \( \Theta \cup \Lambda \), where the (possibly empty) set \( \Lambda \) is shifted to the left of semicolon; note that \( \text{Lhs}(\sigma) = \text{Lhs}(\sigma_1) \). As sketched above, the set \( \Sigma \cup \Lambda \) constrains the forcing of \( A \) in the new world \( \alpha \) generated by a join application below \( \sigma \), so to guarantee that \( \alpha \not\not\not\not\not\not\not B \). We remark that, to get the formula \( A \supset B \) in the conclusion, in general many choices of \( \Lambda \) are possible, as illustrated in the next example.

**Example 3.2.** Let \( \sigma_1 = \vdots; p, q \rightarrow B \) be an \( \text{FRJ}(G) \)-sequent such that the formula \( p \lor q \supset B \) belongs to \( \text{Sr}(G) \). To apply rule \( \supset \) to \( \sigma_1 \) so to get a sequent with \( p \lor q \supset B \) in the right, we have to select a subset \( \Lambda \) of \( \{p, q\} \) satisfying the side condition \( p \lor q \in Cl(\Lambda) \). The following three choices are possible:

\[
\begin{align*}
\Lambda_1 &= \{p\} & \Lambda_2 &= \{q\} & \Lambda_3 &= \{p, q\} \\
\vdots; p, q \rightarrow B & \rightarrow \vdots; p, q \rightarrow B & \rightarrow \vdots; p, q \rightarrow B & \rightarrow
\end{align*}
\]

\[
\begin{align*}
p; q \rightarrow p \lor q \supset B & \rightarrow p; q, \vdots; p \lor q \supset B & \rightarrow p; q, \vdots; p \lor q \supset B & \rightarrow
\end{align*}
\]

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
We point out that $\Lambda_1$ and $\Lambda_2$ are minimal sets satisfying the side condition, while $\Lambda_3$ is not. Indeed, the empty set is the only proper subset of $\Lambda_1$ and $\Lambda_2$ and $p \lor q \notin Cl(\emptyset)$. However, $\Lambda_3$ is not minimal, since both $\Lambda_1$ and $\Lambda_2$ are proper subsets of $\Lambda_3$.

In Section 3.1, we restrict the application of $\vDash_e$ to the cases where $\Sigma$ is extended with a minimal set $\Lambda$ such that $A \in Cl(\Sigma \cup \Lambda)$, and we show that this does not affect completeness. This choice reduces the number of possible instances of rule $\vDash_e$, hence it shrinks the refutation-search space.

Rule $\vDash_e$. The rule $\vDash_e$ has premise $\sigma_1 = \Gamma \not\vdash B$ and conclusion $\sigma = \ldots \Theta \not\vdash A \supset B$; this is the only rule of FRJ($G$) turning a regular sequent into an irregular one. We require that $A \in Cl(\Gamma)$ and $\Theta$ is any subset of $Cl(\Gamma) \cap \overline{\Gamma}$ such that $A \notin Cl(\Theta)$. This entails that, in the world $\alpha_1$ associated with $\sigma_1$, $\alpha_1 \models \Theta$ and $\alpha_1 \not\models A \supset B$. The condition $A \notin Cl(\Theta)$ (whence the name $\vDash_e$) is essentially needed to introduce a weight function on sequents such that the weight of the conclusion of a rule is smaller than the weight of each premise (see Section 3.3). We have

\[
\frac{\sigma_1 = \Gamma \not\vdash B}{\sigma = \ldots \Theta \not\vdash A \supset B \quad \vDash_e \quad A \in Cl(\Gamma)}
\]

Also in this case, many choices of $\Theta$ are possible in general, as shown in the next example.

**Example 3.3.** Let $\sigma_1 = p, q \not\vdash B$ be an FRJ($G$)-sequent and let assume that

\[
\overline{\Gamma} = St(G) \cap (\forall \cup \neg\forall) = \{p, q, r, r \supset p, p \supset r\}, \quad C = p \land q \supset B \in Sr(G).
\]

We show all the possible applications of $\vDash_e$ to $\sigma_1$ yielding an irregular sequent of the form $\ldots \Theta \not\vdash C$. By the side conditions, $\Theta$ must be a subset of $Cl((p, q)) \cap \overline{\Gamma} = \{p, q, r \supset p\}$, which gives rise to eight possible choices:

\[
\begin{align*}
\Theta_1 &= \emptyset, & \Theta_2 &= \{p\}, & \Theta_3 &= \{q\}, & \Theta_4 &= \{r \supset p\}, \\
\Theta_5 &= \{p, r \supset p\}, & \Theta_6 &= \{q, r \supset p\}, & \Theta_7 &= \{p, q\}, & \Theta_8 &= \{p, q, r \supset p\}.
\end{align*}
\]

By the side conditions, we need $p \land q \notin Cl(\Theta)$, hence $\Theta_7$ and $\Theta_8$ must be ruled out. This leads to six possible applications of rule $\vDash_e$:

\[
\begin{align*}
\Theta_1 &= \emptyset & \frac{p, q \not\vdash B}{\ldots \Theta_1 \not\vdash C} & \vDash_e, & \Theta_2 &= \{p\} & \frac{p, q \not\vdash B}{\ldots \Theta_2 \not\vdash C} & \vDash_e, \\
\Theta_3 &= \{q\} & \frac{p, q \not\vdash B}{\ldots \Theta_3 \not\vdash C} & \vDash_e, & \Theta_4 &= \{r \supset p\} & \frac{p, q \not\vdash B}{\ldots \Theta_4 \not\vdash C} & \vDash_e, \\
\Theta_5 &= \{p, r \supset p\} & \frac{p, q \not\vdash B}{\ldots \Theta_5 \not\vdash C} & \vDash_e, & \Theta_6 &= \{q, r \supset p\} & \frac{p, q \not\vdash B}{\ldots \Theta_6 \not\vdash C} & \vDash_e.
\end{align*}
\]

We point out that $\Theta_3$ is a maximal set satisfying the side condition. Indeed, the only proper superset $\Theta'$ of $\Theta_3$ such that $\Theta' \subseteq Cl((p, q)) \cap \overline{\Gamma}$ is $\Theta_8$ and $p \land q \in Cl(\Theta_8)$. Similarly, $\Theta_6$ is maximal as well. However, sets $\Theta_1, \ldots, \Theta_4$ are not maximal, since each of them is a proper subset of $\Theta_3$ or $\Theta_6$.

In Section 3.1, we restrict the application of $\vDash_e$ to maximal $\Theta$.

Rule $\lor$. The rule $\lor$ has premises $\sigma_1 = \Sigma_1 ; \Theta_1 \supset C_1$ and $\sigma_2 = \Sigma_2 ; \Theta_2 \supset C_2$ and conclusion $\sigma = \Sigma ; \Theta \supset C_1 \lor C_2$ introducing a disjunction in the right. As discussed in the Introduction, some care is required in defining the left formulas of $\sigma$. The leading idea is that the $\Sigma$-sets of premises must be preserved in the conclusion, since are commitments about the validity of formulas, while the
\(\Theta\)-sets are intersected. We have to guarantee that \(\text{Lhs}(\sigma) \subseteq \text{Lhs}(\sigma_1) \cap \text{Lhs}(\sigma_2)\), hence some side conditions on the \(\Sigma\)-sets are needed. We define

\[
\sigma_1 = \Sigma_1; \Theta_1 \vdash C_1 \quad \sigma_2 = \Sigma_2; \Theta_2 \vdash C_2 \quad \sigma = \Sigma_1, \Sigma_2; \Theta_1 \cap \Theta_2 \vdash C_1 \lor C_2 \quad \Sigma_1 \subseteq \Sigma_2 \cup \Theta_2 \quad \Sigma_2 \subseteq \Sigma_1 \cup \Theta_1.
\]

**Join rules.** The join rules \(\bowtie^{\text{Alt}}\) and \(\bowtie^\lor\) apply to \(n \geq 1\) irregular sequents \(\sigma_1 = \Sigma_1; \Theta_1 \rightarrow A_1, \ldots, \sigma_n = \Sigma_n; \Theta_n \rightarrow A_n\) and yield a regular sequent \(\sigma = \Gamma \not\vdash C\); this is the only way to obtain a regular sequent from irregular ones. These two rules have a similar structure and only differ in the form of \(C\): in rule \(\bowtie^{\text{Alt}}\), \(C \in \mathcal{V}^\perp\) while in \(\bowtie^\lor\), \(C\) is an \(\lor\)-formula. For every \(1 \leq j \leq n\), we write the premise \(\sigma_j\) as follows:

\[
\sigma_j = \Sigma_j^{\text{Alt}}, \Theta_j^{\text{Alt}} \vdash A_j, \quad \Sigma_j^{\text{Alt}} \cup \Theta_j^{\text{Alt}} \subseteq \mathcal{V}, \quad \text{and } \Sigma_j \cup \Theta_j \subseteq \mathcal{L}^\supseteq.
\]

Similar to rule \(\lor\), formulas in \(\Sigma_j\) must be preserved in the conclusion, and we need \(\text{Lhs}(\sigma) \subseteq \text{Lhs}(\sigma_1) \cap \cdots \cap \text{Lhs}(\sigma_n)\). Thus, we require the side condition of rule \(\lor\) for every pair of distinct premises; this is formalized by the side condition (J1) of join rules.

From a semantic perspective, the role of join rules is to downward expand a countermodel under construction; the conclusion \(\sigma = \Gamma \not\vdash C\) directly corresponds to a new world \(\alpha\) of the countermodel such that all the formulas in \(\Gamma\) are forced in \(\alpha\) and \(C\) is not forced in \(\alpha\). To perform this, we need a further side condition on the sets \(\Sigma_j\). Let

\[
\Sigma^{\text{Alt}} = \bigcup_{1 \leq j \leq n} \Sigma_j^{\text{Alt}}, \quad \Sigma^\supseteq = \bigcup_{1 \leq j \leq n} \Sigma_j^\supseteq.
\]

Both \(\Sigma^{\text{Alt}}\) and \(\Sigma^\supseteq\) must be kept in the conclusion \(\sigma = \Gamma \not\vdash C\) (namely, \(\Sigma^{\text{Alt}} \cup \Sigma^\supseteq \subseteq \Gamma\)), since all the formulas in \(\Sigma^{\text{Alt}} \cup \Sigma^\supseteq\) must be forced in the new world \(\alpha\). A formula \(Y \supseteq Z\) is **supported** if there exists a premise \(\sigma_k\) (\(1 \leq k \leq n\)) such that \(Y = A_k\) (\(A_k\) is the right formula of \(\sigma_k\)). Intuitively, if a formula \(Y \supseteq Z \in \Sigma^\supseteq\) is supported, then \(Y\) is not forced in \(\alpha\), and this allows us to conclude that \(Y \supseteq Z\) is forced in \(\alpha\). Condition (J2) on join rules ensure that all the \(\supseteq\)-formulas in the sets \(\Sigma_j\) are supported.

**Example 3.4.** Let \(\sigma_1 = \Sigma_1; \Theta_1 \rightarrow B\), where \(\Sigma_1 = \{B \supseteq X_1, B \supseteq X_2, C \supseteq X_3\}\) and \(C \neq B\). We cannot apply a join rule having \(\sigma_1\) as only premise, since the formula \(C \supseteq X_3\) would not be supported (while both \(B \supseteq X_1\) and \(B \supseteq X_2\) are, being \(B\) the right formula of \(\sigma_1\)). Thus, to apply a join rule, we need a further premise \(\sigma_2 = \Sigma_2; \Theta_2 \rightarrow C\) such that \(\Sigma_1 \subseteq \Sigma_2 \cup \Theta_2\) and \(\Sigma_2 \subseteq \Sigma_1 \cup \Theta_1\) (see condition (J1)). In turn, \(\Sigma_2\) might contain some non-supported formulas; e.g., a formula \(D \supseteq X_4\) such that \(D \neq B\) and \(D \neq C\); in this case, we need a further premise \(\sigma_3\) such that \(\text{Rhs}(\sigma_3) = D\) to support it.

Side conditions do not concern the sets \(\Theta_j\), and in the conclusion, we keep some of the formulas in the intersection of all the \(\Theta_j\). More precisely, the formulas \(\Theta^{\text{Alt}}\) and \(\Theta^\supseteq\) to be kept in the conclusion are defined as follows:

\[
\Theta^{\text{Alt}} = \bigcap_{j=1}^n \Theta_j^{\text{Alt}}, \quad \Theta^\supseteq = \left\{ Y \supseteq Z \in \bigcap_{j=1}^n \Theta_j^\supseteq \mid Y \in \{A_1, \ldots, A_n\} \right\}.
\]

We can now define the conclusion \(\sigma\) of a join rule having premises \(\sigma_1, \ldots, \sigma_n\) matching the side conditions (J1) and (J2). In rule \(\bowtie^{\text{Alt}}\), we can choose as right formula of \(\sigma\) any formula \(F \in (\mathcal{V} \cup \{\perp\}) \cap \text{Rhs}(G)\) such that \(F \notin \Sigma^{\text{Alt}}\) (see Condition (J3)). We set

\[
\sigma = \Sigma^{\text{Alt}}, \Theta^{\text{Alt}} \setminus \{F\}, \Theta^\supseteq, \Theta^\supseteq \not\vdash F.
\]

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
In rule \( \bowtie^\vee \), we can choose as right formula of \( \sigma \) any \( \lor \)-formula \( C_1 \lor C_2 \in \text{Rhs}(G) \) such that both \( C_1 \) and \( C_2 \) are among \( A_1, \ldots, A_n \) (see Condition (J4)). We set
\[
\sigma = \Sigma^\text{At}, \Theta^\supseteq, \Theta^\supseteq \not\Rightarrow C_1 \lor C_2, \quad \{C_1, C_2\} \subseteq \{A_1, \ldots, A_n\}.
\]
We note that rule \( \bowtie^\vee \) is similar to the rule \( r_n \) presented in Reference [27].

We point out the following special case of application of \( \bowtie^\text{At} \) having only one premise (\( \Sigma^\supseteq \) is possibly empty):
\[
\frac{\Sigma^\text{At}, \Sigma^\supseteq; \Theta^\supseteq, \Theta^\supseteq \not\Rightarrow A}{\Sigma^\text{At}, \Sigma^\supseteq, \Theta^\text{At} \setminus \{F\}, \Theta^\supseteq \not\Rightarrow F} \bowtie^\text{At} \quad \Sigma^\supseteq = \{A \supset B_1, \ldots, A \supset B_m\} \quad \Theta^\supseteq = \{Y \supset Z \in \Theta^\supseteq \mid Y = A\} \quad F \in (\mathcal{V} \setminus \Sigma^\text{At}) \cup \{\bot\}
\]

We formally discuss the properties of \( \text{FRJ}(G) \)-refutations. Let us introduce the following relations between sequents occurring in an \( \text{FRJ}(G) \)-refutation:

- \( \sigma_1 \Rightarrow_0 \sigma_2 \) iff \( \mathcal{R} \) is a rule of \( \text{FRJ}(G) \) such that \( \sigma_2 \) is the conclusion of \( \mathcal{R} \) and \( \sigma_1 \) is one of its premises;
- \( \sigma_1 \equiv_0 \sigma_2 \) iff there exists a rule \( \mathcal{R} \) of \( \text{FRJ}(G) \) such that \( \sigma_1 \Rightarrow_0 \sigma_2 \);
- \( \Rightarrow \) is the transitive closure of \( \Rightarrow_0 \);
- \( \Rightarrow^* \) is the reflexive closure of \( \Rightarrow \).

The following lemma presents some properties of the above relations that have an essential role in proving soundness and termination of \( \text{FRJ}(G) \). These properties can be easily proved by inspecting the rules of \( \text{FRJ}(G) \) and exploiting properties (C13) and (C16) of closures:

**Lemma 3.5.**

(i) \( \sigma_1 \Rightarrow_0 \sigma_2 \) and \( \mathcal{R} \not\supseteq \equiv \) imply \( \text{Lhs}(\sigma_2) \subseteq \text{Lhs}(\sigma_1) \).

(ii) \( \sigma_1 \equiv_0 \sigma_2 \) implies \( \text{Lhs}(\sigma_2) \subseteq \text{Cl}(\text{Lhs}(\sigma_1)) \).

(iii) \( \sigma_1 \Rightarrow^* \sigma_2 \) implies \( \text{Lhs}(\sigma_2) \subseteq \text{Cl}(\text{Lhs}(\sigma_1)) \).

In the next example, exploiting Lemma 3.5, we show that the unsound refutation discussed in the Introduction cannot be simulated in \( \text{FRJ}(G) \).

**Example 3.6.** In the Introduction, by applying the unsound rule \( R \lor \), we got a wrong refutation of the sequent \( \sigma = p, H \not\Rightarrow q_1 \lor q_2 \), where \( H = p \supset q_1 \lor q_2 \). By the soundness property (SREG), if \( \sigma \) could be proved in \( \text{FRJ}(G) \), there should be a world \( \alpha \) of a model such that both \( p \) and \( H \) are forced in \( \alpha \) and \( q_1 \lor q_2 \) is not forced in \( \alpha \), a contradiction. We show that it is not possible to build an \( \text{FRJ}(G) \)-refutation \( \mathcal{D} \) of \( \sigma \). Indeed, the root rule of \( \mathcal{D} \) should be \( \bowtie^\vee \), the only rule of \( \text{FRJ}(G) \) having as conclusion a regular sequent with an \( \lor \)-formula in the right. Since the formula \( H \) must be supported (see the side condition (J2) of join rules), rule \( \bowtie^\vee \) should have a premise \( \sigma' = \Sigma'; \Theta' \not\Rightarrow p \). Since the only irregular sequents having a formula in \( \mathcal{V}^\perp \) (where \( \mathcal{V}^\perp = \mathcal{V} \cup \{\bot\} \)) in the right are the irregular axioms, \( \mathcal{D} \) should have the following shape:

\[
\mathcal{D} = \cdots \sigma' = \cdots; \Theta' \not\Rightarrow p \quad \text{Ax}_{\ldots} \cdots \bowtie^\vee, \quad p \not\in \Theta'.
\]

Hence, \( \sigma' \not\Rightarrow^*_0 \sigma \), which implies, by Lemma 3.5(i), \( \{p, H\} \subseteq \Theta' \). Thus, both \( p \in \Theta' \) and \( p \not\in \Theta' \), a contradiction. We conclude that there is no \( \text{FRJ}(G) \)-refutation of \( \sigma \).

However, it is possible to have applications of rule \( \bowtie^\vee \) having conclusion \( p \not\Rightarrow q_1 \lor q_2 \) or \( H \not\Rightarrow q_1 \lor q_2 \). For instance, let
\[
G = (p \land H) \supset (q_1 \lor q_2), \quad H = p \supset q_1 \lor q_2.
\]
We have
\[ \text{Sl}(G) = \{ p \land H, H, q_1 \lor q_2, p, q_1, q_2 \}, \quad \text{Sr}(G) = \{ G, q_1 \lor q_2, p, q_1, q_2 \}. \]

We can build the following FRJ(G)-refutation:
\[
\frac{\vdots \vdash p, q_2, H \rightarrow q_1 \quad \text{Ax} \rightarrow}{p \nRightarrow q_1 \lor q_2} \quad \frac{\vdots \vdash p, q_1, H \rightarrow q_2 \quad \text{Ax} \rightarrow}{p \nRightarrow q_1 \lor q_2} \quad \frac{\vdots \vdash p, q_2, H \rightarrow q_1 \quad \text{Ax} \rightarrow}{p \nRightarrow q_1 \lor q_2}.
\]

We point out that the side conditions (J1) and (J2) of the join rules trivially hold, since the \( \Sigma \)-sets of the premises are empty. In the conclusion, \( H \) is left out, since it is not supported (no premise has \( p \) in the right). Similarly, we can build the following FRJ(G)-refutation:
\[
\frac{\vdots \vdash p, q_2, H \rightarrow q_1 \quad \text{Ax} \rightarrow}{H \nRightarrow q_1 \lor q_2} \quad \frac{\vdots \vdash p, q_1, H \rightarrow q_2 \quad \text{Ax} \rightarrow}{q_1, q_2, H \rightarrow p \quad \text{Ax} \rightarrow}.
\]

In the conclusion, \( p \) is omitted, since it does not occur as left formula in the right-most premise.

### 3.1 Restrictions (RS1)–(RS4) on the Application of FRJ(G) Rules

We formulate restrictions on the application of the rules \( \supseteq_e \), \( \supseteq_q \), \( \bowtie^\mathsf{At} \) and \( \bowtie^\mathsf{V} \) of FRJ(G) with the aim to shrink the refutation-search space. Hereafter, we assume that FRJ(G)-refutations comply with the following restrictions (RS1)–(RS4) (in particular, completeness of FRJ(G) is proved assuming these restrictions):

- **RS1** In rule \( \supseteq_e \), we require that \( \Lambda \) is a minimal set satisfying the side condition, namely:
  \[ \Lambda' \subseteq \Lambda \text{ implies } A \notin \text{Cl}(\Sigma \cup \Lambda'). \]

- **RS2** In rule \( \supseteq_q \), we require that \( \Theta \) is a maximal set satisfying the side condition, namely:
  \[ \Theta \subseteq \Theta' \subseteq \text{Cl}(\Gamma) \cap \overline{\text{P}} \text{ implies } A \in \text{Cl}(\Theta'). \]

- **RS3** In \( \bowtie^\mathsf{At} \), for every \( Y \in \{ A_1, \ldots, A_n \} \) there is \( Y \supseteq Z \in \text{Sl}(G) \).

- **RS4** In \( \bowtie^\mathsf{V} \), for every \( Y \in \{ A_1, \ldots, A_n \} \), either there is \( Y \supseteq Z \in \text{Sl}(G) \) or there is \( Y \lor Z \in \text{Sr}(G) \) or there is \( Z \lor Y \in \text{Sr}(G) \).

In Example 3.2, the applications of \( \supseteq_e \) complying with (RS1) are the ones concerning \( \Lambda_1 \) and \( \Lambda_2 \). In Example 3.3, the applications of \( \supseteq_q \) satisfying (RS2) are the ones related to \( \Theta_3 \) and \( \Theta_6 \).

We provide some significant examples of refutations.

**Example 3.7.** Let us consider the following instances \( S \) and \( T \) of Scott and Anti-Scott principles, which are equivalent to Nishimura formulas \( N_{10} \) and \( N_9 \), respectively [3] (the schema generating \( N_i \) is given in Section 7):
\[ S = ((\neg p \supset p) \supset \neg p \lor p) \supset \neg p \lor \neg p, \quad T = S \supset (\neg \neg p \supset p) \lor \neg \neg p. \]

Both formulas are valid in Classical Logic but not in IPL. Figures 2 and 3 show an FRJ(S)-refutation \( \mathcal{D}_S \) of \( S \) and an FRJ(T)-refutation \( \mathcal{D}_T \) of \( T \), respectively, in linear representation. We populate the database of proved sequents according with the naive recipe of [7]: We start by inserting the axioms; then we enter a loop where, at each iteration, we apply the rules to the sequents collected in previous steps. For the sake of conciseness, we only show the sequents needed to get the goal. We denote with \( \sigma(k) \) the sequent derived at line \( (k) \). The tree-like structure of refutations \( \mathcal{D}_S \) and \( \mathcal{D}_T \) are displayed in Figures 5 and 6, respectively. We point out that \( \mathcal{D}_T \) contains an application of
\[ S = H \cup \neg \neg \neg \neg p \lor \neg \neg p \quad H = (\neg \neg \neg \neg p \cup p) \cup \neg p \lor p \]
\[ \text{St}(S) = \{ H, \neg p \lor p, \neg \neg p, \neg p, p \} \quad \text{Sr}(S) = \{ S, \neg \neg \neg \neg p \lor p, \neg \neg \neg \neg p, p, \bot \} \]

1. \[ \vdash p, H, \neg \neg \neg \neg p \lor \neg \neg p \rightarrow \bot \quad \text{Ax}_{\neg \neg} \quad // \text{Start} \]
2. \[ \vdash H, \neg \neg \neg \neg p \lor \neg p \rightarrow p \quad \text{Ax}_{\neg \neg} \]
3. \[ \vdash p ; H, \neg \neg \neg \neg p \lor \neg p \rightarrow \neg \neg p \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 1} \]
4. \[ \vdash \neg \neg p ; H, \neg p \rightarrow \neg \neg \neg \neg p \lor p \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 2} \]
5. \[ \vdash \neg p \rightarrow \bot \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 3} \]
6. \[ \vdash p, \neg \neg \neg \neg p \lor \neg p \rightarrow \bot \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 4} \]
7. \[ \vdash H \rightarrow \neg \neg \neg \neg p \lor p \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 5} \]
8. \[ \vdash H \rightarrow \neg \neg \neg \neg p \lor \neg p \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 6} \]
9. \[ \vdash H \rightarrow S \quad \text{Ax}_{\neg \neg} \quad // \text{Iteration 7} \]

Fig. 2. The \text{FRJ}(S)-refutation \( D_S \) of \( S \).

\( \bowtie \lor \) having four premises, namely:

\[
\begin{array}{cccc}
\sigma(2) & & \sigma(7) & & \sigma(8) & & \sigma(11) \\
& & & & & & \text{Ax}_{\lor} \\
\sigma(13) & & & & & & \text{Ax}_{\lor} \\
\end{array}
\]

\[
\sigma(2) = \vdash ; S, \neg \neg \neg \neg p \lor p, \neg \neg \neg \neg p, \neg p \rightarrow p \\
\sigma(7) = \vdash ; S, \neg \neg \neg \neg p \lor p, \neg \neg \neg \neg p, \neg \neg \neg \neg p \rightarrow \neg \neg p \\
\sigma(8) = \vdash ; S, \neg \neg \neg \neg p \lor p, \neg \neg \neg \neg p, \neg p \rightarrow \neg \neg p \\
\sigma(11) = \vdash ; S, \neg \neg \neg \neg p \lor p, \neg \neg \neg \neg p, \neg \neg \neg \neg p \rightarrow H \\
\sigma(13) = \neg \neg \neg \neg p \lor p, S \rightarrow \neg \neg p \lor p
\]

The sequent \( \sigma(13) \) is essential to build the countermodel, since it corresponds to a world where both \( \neg \neg \neg \neg p \lor p \) and \( S \) are forced, while \( \neg \neg \neg \neg p \lor p \) is not; to get \( \Gamma = \{ \neg \neg \neg \neg p \lor p, S \} \) in the left of \( \sigma(13) \), we have to use premises \( \sigma' \) such that \( \Gamma \subseteq \text{Lhs}(\sigma') \) (see Lemma 3.5(i)). Sequents \( \sigma(8) \) and \( \sigma(2) \) are needed to obtain \( \neg \neg \neg \neg p \lor p \) in the right of \( \sigma(13) \), while sequents \( \sigma(7) \) and \( \sigma(11) \) are needed to support \( \neg \neg \neg \neg p \lor p \) and \( S \), respectively. One can easily check that the side conditions (J1) and (J2) hold, thus the displayed application of rule \( \bowtie \lor \) is sound.

Example 3.8. Another significant example is the \text{FRJ}(K)-refutation \( D_K \) of the instance \( K = (\neg a \lor c) \lor (\neg a \lor b) \lor (\neg a \lor c) \) of Kreisel-Putnam principle [3] shown in Figure 4. Differently from the formulas \( S \) and \( T \) in the previous example, the formula \( K \) contains three propositional variables, which give rise to eight axioms. In the figure, we only consider the sequents needed to prove the goal.

3.2 Countermodels and Soundness of \text{FRJ}(G)

We show that we can extract from an \text{FRJ}(G)-refutation \( D \) of \( G \) a countermodel for \( G \). We call \( p \)-sequent (prime sequent) of \( D \) any regular sequent occurring in \( D \), which is either an axiom or the conclusion of a join rule. Given an \text{FRJ}(G)-refutation \( D \) of \( G \), let \( \text{Mod}(D) = \langle P(D), \leq, \rho, V \rangle \) be the structure where:

- \( P(D) \) is the set of all \( p \)-sequents occurring in \( D \);
- for every \( \sigma_1, \sigma_2 \in P(D) \), \( \sigma_1 \leq \sigma_2 \iff \sigma_2 \rightarrow \sigma_1 \);
- \( \rho \) is the minimum of \( P(D) \) w.r.t. \( \leq \);
- \( V \) maps \( \sigma \in P(D) \) to the set \( V(\sigma) = \text{Lhs}(\sigma) \cap V \).

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
\( T = S \supset (\neg p \supset p) \vee \neg p \quad S = H \supset \neg p \vee p \quad H = (\neg p \supset p) \supset \neg p \vee p \)

\[
\begin{align*}
\text{St}(T) & = \{ S, \neg p \supset p, \neg p \vee p, \neg p, p \} \\
\text{Sn}(T) & = \{ T, H, (\neg p \supset p) \vee \neg p, \neg p \supset p, \neg p \vee p, \neg p, p, \bot \}
\end{align*}
\]

(1) \quad \vdots ; \ p, \ S, \neg p \supset p, \neg p, p \Rightarrow \bot \quad \text{Ax}_{\neg p}

(2) \quad \vdots ; \ S, \neg p \supset p, \neg p \Rightarrow p \quad \text{Ax}_{\neg p}

(3) \quad p; \ S, \neg p \supset p, \neg p, p \Rightarrow \neg p \quad \supset_{\epsilon} (1)

(4) \quad \neg p; \ p, \ S, \neg p \supset p, \neg p \Rightarrow \neg p \quad \supset_{\epsilon} (1)

(5) \quad \neg p, \neg p \supset p \Rightarrow \bot \quad \supset_{\epsilon} (2), (4)

(6) \quad p, \neg p \Rightarrow \bot \quad \supset_{\epsilon} (3)

(7) \quad \vdots ; \ S, \neg p \supset p \Rightarrow \neg p \quad \supset_{\epsilon} (5)

(8) \quad \vdots ; \ S, \neg p \supset p, \neg p \Rightarrow \neg p \quad \supset_{\epsilon} (6)

(9) \quad \vdots ; \ S, \neg p \supset p, \neg p \Rightarrow \neg p \vee p \quad \lor (2), (8)

(10) \quad \neg p \Rightarrow p \quad \supset_{\epsilon} (8)

(11) \quad \neg p \supset p; \ S, \neg p \Rightarrow H \quad \supset_{\epsilon} (9)

(12) \quad \vdots ; S \Rightarrow \neg p \supset p \quad \supset_{\epsilon} (10)

(13) \quad \neg p \supset p, S \Rightarrow \neg p \vee p \quad \supset_{\epsilon} (2), (7), (8), (11)

(14) \quad \vdots ; S \Rightarrow H \quad \supset_{\epsilon} (13)

(15) \quad S \Rightarrow (\neg p \supset p) \vee \neg p \quad \supset_{\epsilon} (7), (12), (14)

(16) \quad S \Rightarrow T \quad \supset_{\epsilon} (15)

Fig. 3. The FRJ(T)-refutation \( D_T \) of \( T \).

One can check that \( \text{Mod}(D) \) is a model. The proof that \( \leq \) is reflexive and transitive is trivial; as for the antisymmetry (which is not essential to prove completeness) is an immediate consequence of Lemma 3.16 given in Section 3.3. Since the root sequent of \( D \) is regular, there exists \( \rho \in P(D) \) such that \( \sigma_p \mapsto_* \rho \), for every \( \sigma_p \in P(D) \), hence \( \rho \) is the minimum of \( P(D) \) w.r.t. \( \leq \). Moreover, by Lemma 3.5(iii) and (C15), \( \sigma_1 \leq \sigma_2 \) implies \( V(\sigma_1) \subseteq V(\sigma_2) \), hence the definition of \( V \) is sound. We call \( \text{Mod}(D) \) the model extracted from \( D \).

For every regular sequent \( \sigma \) occurring in \( D \), let \( \phi(\sigma) \) be the p-sequent in \( D \) immediately above \( \sigma \), namely:

\[ \phi(\sigma) = \sigma_p \quad \text{iff} \quad \sigma_p \in P(D) \text{ and } \sigma_p \mapsto_* \sigma \text{ and for every } \sigma'_p \in P(D), \sigma_p \mapsto_* \sigma'_p \mapsto_* \sigma \text{ implies } \sigma'_p = \sigma_p. \]

Note that \( \phi \) is a well-defined function, since every regular sequent occurring in \( D \) is either a p-sequent or the conclusion of a one-premise rule. It is easy to check that:

- p-sequents are fixed points of \( \phi \), i.e., \( \sigma_p \in P(D) \) implies \( \phi(\sigma_p) = \sigma_p \);
- \( \phi \) is a surjective map from the set of regular sequents of \( D \) onto \( \text{Mod}(D) \);
- if \( \sigma_1 \) and \( \sigma_2 \) are regular and \( \sigma_1 \mapsto_* \sigma_2 \), then \( \phi(\sigma_2) \leq \phi(\sigma_1) \).

We call \( \phi \) the map associated with \( D \).

The following lemma is the key to prove the soundness property \( \text{SREG} \) of FRJ(G) stated at the beginning of Section 3. The proof goes by induction on the height of a sequent \( \sigma \) in an FRJ(G)-refutation. Formally, let \( D \) be an FRJ(G)-refutation and let \( \sigma \) be a sequent occurring in \( D \). The height of \( \sigma \) (in \( D \)), denoted by \( h(\sigma) \), is the maximum distance from \( \sigma \) to an axiom sequent of \( D \),
\[ K = K_0 \supset K_1 \quad K_0 = \neg a \supset b \vee c \quad K_1 = (\neg a \supset b) \vee (\neg a \supset c) \]
\[ \text{St}(K) = \{ K_0, \neg a, a, b, c \} \quad \text{Sr}(K) = \{ K, K_1, \neg a \supset b, \neg a \supset c, a, b, c, \bot \} \]

\begin{align*}
(1) & \vdots ; a, b, c, K_0, \neg a \rightarrow \bot & \text{Ax}_\rightarrow \\
(2) & \vdots ; b, c, K_0, \neg a \rightarrow a & \text{Ax}_\neg \\
(3) & a; b, c, K_0, \neg a \rightarrow \neg a & \supset_e (1) \\
(4) & c, \neg a \Rightarrow b & \supset^\text{At} (2) \\
(5) & b, \neg a \Rightarrow c & \supset^\text{At} (2) \\
(6) & a, b, c, K_0 \Rightarrow \bot & \supset^\text{At} (3) \\
(7) & \vdots ; c, K_0 \Rightarrow \neg a \supset b & \supset_g (4) \\
(8) & \vdots ; b, K_0 \Rightarrow \neg a \supset c & \supset_g (5) \\
(9) & \vdots ; b, c, K_0 \Rightarrow \neg a & \supset_g (6) \\
(10) & K_0 \Rightarrow K_1 & \supset^\text{'} (7) (8) (9) \\
(11) & K_0 \Rightarrow K & \supset_e (10) \\
\end{align*}

Fig. 4. The FRJ(K)-refutation \( D_K \) of \( K \).

namely:

\[
\text{h}(\sigma) = \begin{cases} 0 & \text{if } \sigma \text{ is an axiom sequent} \\
1 + \max \{ \text{h}(\sigma') \mid \sigma' \text{ occurs in } D \text{ and } \sigma' \Rightarrow \sigma \} & \text{otherwise} \end{cases}
\]

The \textit{height} of \( D \), denoted by \( \text{h}(D) \), is the height of the root sequent of \( D \).

Let \( |A| \) denote the \textit{size} of \( A \), namely, the number of symbols in \( A \). First, we prove a property concerning the size of formulas in the stable part of irregular sequents.

\textbf{Lemma 3.9.} Let \( \vdash_{\text{FRJ}(G)} \Sigma ; \Theta \Rightarrow C \). For every \( H \in \Sigma, |H| < |C| \).

\textbf{Proof.} Let \( \sigma = \Sigma ; \Theta \Rightarrow C \) and let \( R \) be the rule applied to get \( \sigma \); we prove the assertion by induction on \( \text{h}(\sigma) \). If \( R = \text{Ax}_\neg \) or \( R = \supset_g \), then \( \Sigma \) is empty, hence the assertion trivially holds. The cases \( R = \land \) and \( R = \lor \) immediately follow by the induction hypothesis. Finally, let \( R = \supset_e \) and let us assume that the rule application is

\[
\frac{\sigma_1 = \Sigma_1 \land \Lambda ; \Theta \Rightarrow A \supset B \quad \supset_e \Sigma = \Sigma_1 \cup \Lambda \quad A \in C(\Sigma_1 \cup \Lambda)}{\sigma = \Sigma_1, \Lambda ; \Theta \Rightarrow A \supset B}.
\]

Let \( H \in \Sigma \). If \( H \in \Sigma_1 \), then by the induction hypothesis, we have \( |H| < |B| \), which implies \( |H| < |A \supset B| \). Let us assume \( H \in \Lambda \). Since \( \Lambda \) is a minimal set such that \( A \in C(\Sigma_1 \cup \Lambda) \) (see the restriction (RS1)), by property (C12) of closures, we get \( \Lambda \subseteq \text{Sf}(A) \). This implies \( |H| \leq |A| \), hence \( |H| < |A \supset B| \). \( \square \)

Let \( \sigma_0 \) be an irregular sequent and \( \sigma_p \) a \( p \)-sequent. We write \( \sigma_0 \Rightarrow \cdots \Rightarrow \sigma_n \Rightarrow \sigma_p \) to mean that \( \sigma_0 \Rightarrow_0 \cdots \Rightarrow_0 \sigma_n \Rightarrow_0 \sigma_p \) \((n \geq 0)\), where the sequents \( \sigma_0, \ldots, \sigma_n \) are irregular and \( \sigma_n \) is a premise of a join rule with conclusion \( \sigma_p \).

\textbf{Lemma 3.10.} Let \( D \) be an \( \text{FRJ}(G) \)-refutation of \( G \), let \( \text{Mod}(D) \) be the model extracted from \( D \) and \( \phi \) the map associated with \( D \). For every sequent \( \sigma \) occurring in \( D \):

\begin{enumerate}
\item[(i)] if \( \sigma = \Gamma \Rightarrow C \), then \( \phi(\sigma) \models \Gamma \) and \( \phi(\sigma) \not\models C \).
\item[(ii)] Let \( \sigma = \Sigma ; \Theta \Rightarrow C \). For every \( \sigma_p \in P(D) \) such that \( \sigma \Rightarrow \sigma_p, \sigma_p \models \Sigma \) implies \( \sigma_p \not\models C \).
\end{enumerate}


\textbf{Proof.} We prove the assertions by a main induction (IH1) on \( h(\sigma) \). Let \( \text{Mod}(D) = \langle P(D), \leq, \rho, V \rangle \) and let \( R \) be the rule applied to get \( \sigma \); we proceed by a case analysis on \( R \).

\begin{itemize}
  \item \( R = \text{Ax}_{\neg \phi} \).
  \end{itemize}

We have \( \sigma = \Gamma^A \setminus \{ C \} \not\models C \), with \( C \in V^L \). Since \( \phi(\sigma) \models \sigma \) and \( V(\sigma) = \Gamma^A \setminus \{ C \} \), (i) immediately follows.

\begin{itemize}
  \item \( R = \text{Ax}_{\neg \phi} \).
  \end{itemize}

We have \( \sigma = \cdot ; \Gamma^A \setminus \{ C \}, \Gamma^\Rightarrow \not\models C \), with \( C \in V^L \). Let \( \sigma_p \in P(D) \) such that \( \sigma \models \models \sigma_p \); note that \( \sigma \) is a premise of an application of a join rule with conclusion \( \sigma_p \). Let \( \Gamma^A = V(\sigma_p) \); by Lemma 3.5(i), \( \Gamma^A \subseteq \Gamma^A \setminus \{ C \} \). It follows that \( C \notin \Gamma^A \), hence \( \sigma_p \not\models C \), and (ii) holds.

\begin{itemize}
  \item \( R = \land \).
  \end{itemize}

Both (i) and (ii) easily follow by (IH1).

\begin{itemize}
  \item \( R = \lor \).
  \end{itemize}

We have

\[
\sigma = \Sigma_1 \cup \Sigma_2 ; \Theta_1 \Rightarrow C_1 \quad \sigma = \Sigma_2 \cup \Theta_2 \Rightarrow C_2, \quad \Sigma = \Sigma_1 \cup \Sigma_2.
\]

Let \( \sigma_p \in P(D) \) such that \( \sigma \models \models \sigma_p \); then, both \( \sigma_1 \models \models \sigma_p \) and \( \sigma_2 \models \models \sigma_p \). Let us assume \( \sigma_p \models \Sigma \). By (IH1) applied to \( \sigma_1 \) and \( \sigma_2 \), we get \( \sigma_p \not\models C_1 \) and \( \sigma_p \not\models C_2 \), hence \( \sigma_p \not\models C_1 \lor C_2 \), which proves (ii).

\begin{itemize}
  \item \( R = \exists \).
  \end{itemize}

If \( \sigma \) is regular, then we have

\[
\sigma = \Gamma \not\models B \quad \exists \phi, \quad A \in \text{Cl}(\Gamma).
\]

By (IH1) applied to \( \sigma_1 \), it holds that \( \phi(\sigma_1) \models \Gamma \) and \( \phi(\sigma_1) \not\models B \). Since \( A \in \text{Cl}(\Gamma) \), by (C1), we get \( \phi(\sigma_1) \models A \). Since \( \phi(\sigma) = \phi(\sigma_1) \), we get \( \phi(\sigma) \not\models A \lor B \), which implies (i).

Let \( \sigma \) be irregular and let us assume

\[
\sigma = \Sigma_1, \cdot \Rightarrow B \quad \exists \phi, \quad A \in \text{Cl}(\Sigma), \quad \text{where } \Sigma = \Sigma_1 \cup \Lambda.
\]

Let \( \sigma_p \in P(D) \) such that \( \sigma \models \models \sigma_p \); then, \( \sigma_1 \models \models \sigma_p \). Let us assume \( \sigma_p \models \Sigma \). By (IH1) applied to \( \sigma_1 \), we get \( \sigma_p \not\models B \). Since \( A \in \text{Cl}(\Sigma) \), by (C1), we get \( \sigma_p \models A \). We conclude \( \sigma_p \not\models A \lor B \), which proves (ii).

\begin{itemize}
  \item \( R = \forall \).
  \end{itemize}

We have

\[
\sigma = \cdot ; \Theta \Rightarrow A \supset B \quad \forall \phi, \quad A \in \text{Cl}(\Gamma).
\]

By (IH1) applied to \( \sigma_1 \), we have \( \phi(\sigma_1) \models \Gamma \) and \( \phi(\sigma_1) \not\models B \). By (C1), \( \phi(\sigma_1) \models A \). Let \( \sigma_p \in P(D) \) such that \( \sigma \models \models \sigma_p \). Then, \( \sigma_1 \models \sigma_p \), which implies \( \sigma_p \leq \phi(\sigma_1) \). Thus, \( \sigma_p \not\models A \lor B \), and this proves (ii).

\begin{itemize}
  \item \( R = \text{At}^A \).
  \end{itemize}
We have

\[
\begin{align*}
\ldots \sigma_j &= \Sigma_j^{\Lambda}, \Sigma_j^\Sigma; \Theta_j^{\Lambda}, \Theta_j^\Sigma \not\rightarrow A_j \ldots \\
\sigma &= \Sigma^{\Lambda}, \Theta^{\Lambda} \setminus \{C\}, \Sigma^\Sigma, \Theta^\Sigma \not\rightarrow C \\
\Gamma^{\Lambda} &= \Sigma^{\Lambda} \cup (\Theta^{\Lambda} \setminus \{C\}) \\
\Gamma^\Sigma &= \Sigma^\Sigma \cup \Theta^\Sigma \\
\Gamma &= \Gamma^{\Lambda} \cup \Gamma^\Sigma.
\end{align*}
\]

Note that \(\sigma \in P(D)\), \(\phi(\sigma) = \sigma\) and \(V(\sigma) = \Gamma^\Lambda\). Since \(C \notin \Gamma^{\Lambda}\), we get

\[
(P1) \quad \sigma \models \Gamma^{\Lambda} \text{ and } \sigma \not\models C.
\]

To complete the proof of (i), it remains to show that \(\sigma \models \Gamma^\Sigma\). To this aim, we prove that, for every formula \(H\), the following properties hold:

\[
\begin{align*}
(P2) & \quad H \in \{A_1, \ldots, A_n\} \text{ implies } \sigma \not\models H. \\
(P3) & \quad H \in \Gamma^\Sigma \text{ implies } \sigma \not\models H.
\end{align*}
\]

To prove (P2) and (P3), we introduce a secondary induction hypothesis (IH2) on \(|H|\). Let \(H = A_j\), where \(j \in \{1, \ldots, n\}\). Note that \(\sigma_j \not\rightarrow \sigma\). Since \(h(\sigma_j) < h(\sigma)\), we can prove \(\sigma \not\models A_j\) by applying (IH1) on \(\sigma_j\); we have to check that

\[
(\dagger) \quad \sigma \models \Sigma_j^{\Lambda} \cup \Sigma_j^\Sigma.
\]

Let \(K \in \Sigma_j^{\Lambda} \cup \Sigma_j^\Sigma\). If \(K \in \Sigma_j^{\Lambda}\), then since \(\Sigma_j^{\Lambda} \subseteq V(\sigma)\), we immediately get \(\sigma \models K\). Let \(K \in \Sigma_j^\Sigma\). By Lemma 3.9, \(|K| < |A_j|\). Since \(K \in \Gamma^\Sigma\), we can apply (IH2) on (P3), and we get \(\sigma \models K\), and this proves (\dagger); this concludes the proof of (P2).

Let \(H \in \Gamma^\Sigma\). Then, there is \(k \in \{1, \ldots, n\}\) such that \(H = A_k \supset B\). To prove \(\sigma \models H\), we have to show that

\[
(\ddagger) \quad \text{for every } \sigma'_p \in P(D) \text{ such that } \sigma \leq \sigma'_p, \sigma'_p \models A_k \text{ implies } \sigma'_p \models B.
\]

Let \(\sigma'_p \in P(D)\) such that \(\sigma \leq \sigma'_p\) and \(\sigma'_p \models A_k\) and let \(\Gamma'_p = \text{Lhs}(\sigma'_p)\). Since \(H \in \text{Lhs}(\sigma)\) and \(\sigma'_p \not\rightarrow \sigma\) (indeed, \(\sigma \leq \sigma'_p\)), by Lemma 3.5(iii) we get \(H \in C(\Gamma'_p)\). Since \(|A_k| < |H|\), we can apply (IH2) on (P2) and claim that \(\sigma \not\models A_k\), hence \(\sigma'_p \not\models \sigma\), which implies \(h(\sigma'_p) < h(\sigma)\). By (IH1) applied to \(\sigma'_p\), we have \(\sigma'_p \models \Gamma'_p\) and, by (C1), \(\sigma'_p \models H\), namely, \(\sigma'_p \models A_k \supset B\). Since \(\sigma'_p \models A_k\), we get \(\sigma'_p \models B\). This concludes the proof of (\ddagger) and of (P3). By (P1) and (P3), Point (i) follows.

- \(\mathcal{R} = \models^{V}\).

Similar to the case \(\models^{\Lambda}\). Here, \(\sigma = \Sigma^{\Lambda}, \Theta^{\Lambda}, \Sigma^\Sigma, \Theta^\Sigma \not\rightarrow C_1 \cup C_2\), where \(\{C_1, C_2\} \subseteq \{A_1, \ldots, A_n\}\), hence \(\sigma \not\models C_1 \cup C_2\) immediately follows by (P2).

By Lemma 3.10, we get:

**Lemma 3.11.** Soundness property (**SREG**) holds.

**Proof.** Let \(\sigma = \Gamma \not\models C\) be provable in **FRJ**(G) and let \(D\) be an **FRJ**(G)-refutation of \(\sigma\). By Lemma 3.10(i), the world \(\phi(\sigma)\) of the model \(\text{Mod}(D)\) satisfies \(\phi(\sigma) \models \Gamma\) and \(\phi(\sigma) \not\models C\). Setting \(\alpha = \phi(\sigma)\), (**SREG**) holds in \(\text{Mod}(D)\).

As an immediate consequence, we get:

**Theorem 3.12.** Let \(D\) be an **FRJ**(G)-refutation of G. Then, \(\text{Mod}(D)\) is a countermodel for G.

**Proof.** By definition, \(D\) is an **FRJ**(G)-refutation of a regular sequent \(\sigma = \Gamma \not\models G\). Let \(\phi\) be the map associated with \(D\). Note that, for every \(\sigma_p \in P(D)\), it holds that \(\sigma_p \not\rightarrow \sigma\), hence \(\phi(\sigma)\) is the root of \(\text{Mod}(D)\). By Lemma 3.10(i), we have \(\phi(\sigma) \not\models G\). We conclude that \(\text{Mod}(D)\) is a countermodel for G.
Fig. 5. The model $\text{Mod}(\mathcal{D}_S)$ (see Figure 2).

$\sigma_k$ refers to the sequent at line $(k)$ in Figure 2

$\phi(\sigma) = \sigma$, for every p-sequent $\sigma$

$\phi(\sigma_{12}) = \sigma_{11}$

Fig. 6. The model $\text{Mod}(\mathcal{D}_T)$ (see Figure 3).

$\sigma_k$ refers to the sequent at line $(k)$ in Figure 3

$\phi(\sigma) = \sigma$, for every p-sequent $\sigma$

$\phi(\sigma_{16}) = \sigma_{15}$

Accordingly, if $\vdash_{\text{FRJ}(G)} G$ then $G$ is not valid, and this proves the Soundness of $\text{FRJ}(G)$ as stated in Theorem 3.1.

Example 3.13. Let us consider the formulas $S$, $T$ and $K$ in examples 3.7 and 3.8. The models $\text{Mod}(\mathcal{D}_S)$, $\text{Mod}(\mathcal{D}_T)$, $\text{Mod}(\mathcal{D}_K)$ and the related maps $\phi$ are shown in Figures 5, 6, and 7, respectively. The bottom world is the root and $\sigma < \sigma'$ iff the world $\sigma$ is drawn below $\sigma'$. For each $\sigma$, we display the set $V(\sigma)$. As an example, in Figure 5, we have $V(\sigma_{6}) = \{p\}$, while $V(\sigma_{9}) = V(\sigma_{11}) = \emptyset$.

We conclude by stating the main property of irregular sequents. As mentioned above, an irregular sequent $\sigma = \Sigma; \Theta \rightarrow C$ does not correspond to a world of a model, but it represents an intermediate state to get a new world $\alpha$ such that $\alpha \models \Sigma$ and $\alpha \not\models C$ via an application of a join rule. This is formalized by the next theorem:
THEOREM 3.14. Let $\mathcal{D}$ be an FRJ($G$)-refutation and $\text{Mod}(\mathcal{D})$ the model extracted from $\mathcal{D}$, let $\sigma = \Sigma; \Theta \vdash C$ be an irregular sequent occurring in $\mathcal{D}$. For every $\sigma_p \in \text{P}(\mathcal{D})$ such that $\sigma \vdash \sigma_p$, $\sigma_p$ is a world of $\text{Mod}(\mathcal{D})$ such that $\sigma_p \models \Sigma$ and $\sigma_p \not\models C$.

PROOF. Let $\sigma_p \in \text{P}(\mathcal{D})$ such that $\sigma \vdash \sigma_p$, and let $\sigma_p = \Gamma_p \nvdash C_p$. Since $\phi(\sigma_p) = \sigma_p$, by Lemma 3.10(i), we get $\sigma_p \models \Gamma_p$. By Lemma 3.5(i) we get $\Sigma \subseteq \Gamma_p$, hence $\sigma_p \models \Sigma$. By Lemma 3.10(ii), we get $\sigma_p \not\models C$, and this concludes the proof. \qed

Example 3.15. Let $G$ be the goal formula defined in Example 3.6: $G = (p \land H) \supset (q_1 \lor q_2)$, $H = p \supset q_1 \lor q_2$.

We can build the following FRJ($G$)-refutation of an irregular sequent $\sigma_1$ having $G$ in the right:

\[
\begin{array}{c}
\vdash p, q_2, H \Rightarrow q_1 & \text{Ax}_\lor & \vdash p, q_1, H \Rightarrow q_2 & \text{Ax}_\lor \\
\\hline
\vdash p, H \Rightarrow q_1 \lor q_2 & \lor & \vdash (p \land H) \supset (q_1 \lor q_2) & \supset_e
\end{array}
\]

At first sight, this contradicts the soundness of the calculus, since we have proved a sequent with a valid formula in the right. Actually, soundness property (SREG) only concerns regular sequents. By Theorem 3.14, we conclude that there is no regular sequent $\sigma$ such that $\sigma_1 \vdash \sigma$. Thus, $\sigma_1$ is a sort of dead sequent that cannot be used to generate new worlds of the model.

3.3 Termination

To conclude the presentation of FRJ($G$), we exhibit a weight function on sequents of FRJ($G$) such that, after the application of a rule, the weight of sequents decreases; accordingly, the naive refutation-search procedure always terminates, even if we do not implement any redundancy check. Let $\sigma_1$ and $\sigma_2$ be two FRJ($G$)-sequents and, for $k \in \{1, 2\}$, let $\Gamma_k = \text{Lhs}(\sigma_k)$ and $C_k = \text{Rhs}(\sigma_k)$; by $||\_||$, we denote the cardinality function. We show that the following properties hold:

1. $\sigma_1 \vdash^0 \sigma_2$ implies $||C(\Gamma_2) \cap \text{St}(G)|| \leq ||C(\Gamma_1) \cap \text{St}(G)||$;
2. $\sigma_1 \vdash^e \sigma_2$ implies $||C(\Gamma_2) \cap \text{St}(G)|| < ||C(\Gamma_1) \cap \text{St}(G)||$;
3. $\sigma_1 \vdash^0 \sigma_2$ and $\mathcal{R}$ is not a join rule imply $|G| - |C_2| < |G| - |C_1|$.

Let $\sigma_1 \vdash^0 \sigma_2$. By Lemma 3.5(ii) and (C16), we have $\text{Cl}(\Gamma_2) \subseteq \text{Cl}(\Gamma_1)$, hence Point (1) holds. If $\mathcal{R} = \supset_e$, then $C_2 = A \supset B$, where $A \in \text{St}(G)$ and $A \in \text{Cl}(\Gamma_1)$ and $A \notin \text{Cl}(\Gamma_2)$; this proves Point (2). If $\mathcal{R}$ is not a join rule, then $|C_2| > |C_1|$, hence Point (3) holds. Properties (1)–(3) suggest that we can define the weight function $\text{wg}(\sigma)$ as the triple of non-negative integers:

\[
\text{wg}(\sigma) = \langle ||C(\Gamma) \cap \text{St}(G)||, \text{tp}(\sigma), |G| - |C| \rangle \quad \text{tp}(\sigma) = \begin{cases} 0 & \text{if } \sigma \text{ is regular} \\ 1 & \text{otherwise} \end{cases}
\]

Let $<$ be the standard lexicographic order on triples of integers; we get
LEMMA 3.16. \( \sigma_1 \vdash \sigma_2 \) implies \( (0, 0, 0) \leq \text{wg}(\sigma_2) < \text{wg}(\sigma_1) \).

Note that the component \( \text{tp}(\sigma) \) accommodates the case where \( \sigma_2 \) is the conclusion of a join rule. Indeed, in this case \( \sigma_2 \) is regular and \( \sigma_1 \) is irregular, hence \( \text{tp}(\sigma_2) < \text{tp}(\sigma_1) \).

We can exploit \( \text{wg} \) to set a bound on the height of refutations and of the extracted models. As usual a branch of an FRJ\((G)\)-refutation is a sequence of sequents \( \sigma_1, \ldots, \sigma_m \) such that \( \sigma_1 \vdash_0 \sigma_2 \vdash_0 \cdots \vdash_0 \sigma_m \).

LEMMA 3.17. Let \( \mathcal{D} \) be an FRJ\((G)\)-refutation, let \( \mathcal{B} \) be a branch of \( \mathcal{D} \) and \( N = |G| \). Then:

(i) the length of the branch \( \mathcal{B} \) is \( O(N^2) \);
(ii) \( \mathcal{B} \) contains \( N \) p-sequents at most.

PROOF. Let \( \mathcal{B} = \sigma_1, \ldots, \sigma_m \) where \( \sigma_1 \vdash_0 \sigma_2 \vdash_0 \cdots \vdash_0 \sigma_m \). By Lemma 3.16, we have \( \text{wg}(\sigma_m) < \text{wg}(\sigma_{m-1}) < \cdots < \text{wg}(\sigma_1) \). For \( j \in \{1, \ldots, m\} \), let \( \text{wg}(\sigma_j) = (k_j, t_j, c_j) \). Since \( ||\text{St}(G)|| \leq N - 1 \), it holds that \( 0 \leq k_j \leq N - 1 \), \( t_j \in \{0, 1\} \) and \( 0 \leq c_j \leq N \), and this implies (i).

Let \( \sigma_0 = \Gamma_h \nleq C_h \) and \( \sigma_I = \Gamma_I \nleq C_I \) be two distinct p-sequents of \( \mathcal{B} \), where \( h < l \). For \( j \in \{h, l\} \), we have \( \text{wg}(\sigma_j) = (k_j, 0, c_j) \), where \( k_j = ||\text{Cl}(\Gamma_j) \cap \text{St}(G)|| \). Since \( \sigma_I \) is the conclusion of a join rule, the sequent \( \sigma_{l-1} \) is irregular. Accordingly, there exists \( u \in \{h, \ldots, l-1\} \) such that \( \sigma_u \) is regular and \( \sigma_{u+1} \) is irregular. This means that \( \sigma_{u+1} \) is obtained by an application of rule \( \supset \), hence the branch has the following form:

\[
\begin{align*}
\sigma_h &= \Gamma_h \nleq C_h \quad \rightarrow_{\star} \quad \sigma_u &= \Gamma \nleq B \quad \supset \quad \sigma_{u+1} = \cdots \quad \Theta \nleq A \supset B \quad \leftarrow \quad \sigma_l &= \Gamma_l \nleq C_l \quad \Theta \nleq A \in \text{St}(G) \quad A \in \text{Cl}(\Gamma) \quad A \notin \text{Cl}(\Theta).
\end{align*}
\]

By Lemma 3.5(iii) and property \((C16)\) of closures, we get \( \text{Cl}(\Gamma_I) \subseteq \text{Cl}(\Gamma_h) \), \( A \in \text{Cl}(\Gamma_h) \), and \( A \notin \text{Cl}(\Gamma_I) \). This implies \( ||\text{Cl}(\Gamma_I) \cap \text{St}(G)|| \leq ||\text{Cl}(\Gamma_h) \cap \text{St}(G)|| \), namely, \( k_l < k_h \leq N \). We conclude that the branch \( \mathcal{B} \) cannot contain more than \( N \) distinct p-sequents. \( \square \)

4 THE REFUTATION-SEARCH PROCEDURE AND SATURATED DATABASES

The plain refutation-search procedure outlined in Example 3.7 suffers from the plethora of redundant sequents generated at each iteration. To reduce the size of the database of proved sequents, we introduce the notion of subsumption.

Given two sequents \( \sigma_1 \) and \( \sigma_2 \), we say that \( \sigma_2 \) subsumes \( \sigma_1 \), and we write \( \sigma_1 \sqsubseteq \sigma_2 \) (or \( \sigma_2 \sqsupseteq \sigma_1 \)) iff one of the following conditions hold:

1. \( \sigma_1 = \Gamma_1 \nleq C \) and \( \sigma_2 = \Gamma_2 \nleq C \) and \( \Gamma_1 \subseteq \Gamma_2 \);
2. \( \sigma_1 = \Sigma ; \Theta_1 \nleq C \) and \( \sigma_2 = \Sigma ; \Theta_2 \nleq C \) and \( \Theta_1 \subseteq \Theta_2 \).

One can easily check that the relation \( \sqsubseteq \) is a partial order on the sets of FRJ\((G)\)-sequents, since it satisfies reflexivity, antisymmetry, and transitivity. By \( \sigma_1 \sqsubseteq \sigma_2 \) (or \( \sigma_2 \sqsupseteq \sigma_1 \)), we mean that \( \sigma_1 \subseteq \sigma_2 \) and \( \sigma_1 \neq \sigma_2 \).

LEMMA 4.1. Let

\[
\frac{\sigma_1 \cdots \sigma_n}{\sigma} \quad \mathcal{R}
\]

be an instance of a rule of FRJ\((G)\) and let \( \sigma_1 \sqsubseteq \sigma_1', \ldots, \sigma_n \sqsubseteq \sigma_n' \). Then,

\[
\frac{\sigma_1' \cdots \sigma_n'}{\sigma'} \quad \mathcal{R}
\]

is an instance of \( \mathcal{R} \) in FRJ\((G)\) where \( \sigma \sqsubseteq \sigma' \).
PROOF. The assertion can be proved by inspecting the rules of FRJ(Γ, Λ). We only detail some significant cases. Let 𝒲 be the rule ∨:

\[
\begin{align*}
\sigma_1 &= \Sigma_1; \Theta_1 \rightarrow C_1 \quad \sigma_2 &= \Sigma_2; \Theta_2 \rightarrow C_2 \quad \Sigma_1 \subseteq \Sigma_2 \cup \Theta_2 \\
\sigma &= \Sigma_1, \Sigma_2; \Theta_1 \cap \Theta_2 \rightarrow C_1 \lor C_2 \quad \Sigma_2 \subseteq \Sigma_1 \cup \Theta_1
\end{align*}
\]

Let \(\sigma_1'\) and \(\sigma_2'\) be such that \(\sigma_1 \subseteq \sigma_1'\) and \(\sigma_2 \subseteq \sigma_2'\). This means that \(\sigma_1' = \Sigma_1; \Theta_1' \rightarrow C_1\) and \(\sigma_2' = \Sigma_2; \Theta_2' \rightarrow C_2\) with \(\Theta_1 \subseteq \Theta_1'\) and \(\Theta_2 \subseteq \Theta_2'\). Hence,

\[
\begin{align*}
\sigma_1' &= \Sigma_1; \Theta_1' \rightarrow C_1 \quad \sigma_2' &= \Sigma_2; \Theta_2' \rightarrow C_2 \\
\sigma' &= \Sigma_1, \Sigma_2; \Theta_1' \cap \Theta_2' \rightarrow C_1 \lor C_2 \quad \Sigma_1 \subseteq \Sigma_2 \cup \Theta_1' \quad \Sigma_2 \subseteq \Sigma_1 \cup \Theta_2'
\end{align*}
\]

is an instance of the rule ∨ and, since \(\Theta_1 \cap \Theta_2 \subseteq \Theta_1' \cap \Theta_2'\), we get \(\sigma \subseteq \sigma'\). The proof for join rules is similar.

Let \(\mathcal{R}\) be the rule \(\supseteq\):

\[
\sigma_1 = \Sigma; \Theta, \Lambda \rightarrow B \\
\sigma = \Sigma, \Lambda; \Theta \rightarrow A \supset B \\
\rightarrow \supseteq, \quad A \in Cl(\Sigma \cup \Lambda)
\]

Let \(\sigma_1'\) be such that \(\sigma_1 \subseteq \sigma_1'\); thus, \(\sigma_1' = \Sigma; \Theta', \Lambda' \rightarrow B\), where \(\Theta \subseteq \Theta'\) and \(\Lambda \subseteq \Lambda'\). By the restriction (RS1), \(\Lambda\) is a minimal set satisfying the side condition; this implies that

\[
\begin{align*}
\sigma_1' &= \Sigma; \Theta', \Lambda' \rightarrow B \\
\sigma' &= \Sigma, \Lambda; \Theta', \Lambda' \setminus \Lambda \rightarrow A \supset B
\end{align*}
\]

is a sound application of rule \(\supseteq\). Since \(\sigma \subseteq \sigma'\), the assertion holds.

Let \(\mathcal{R}\) be the rule \(\supseteq\):

\[
\sigma_1 = \Gamma \not\Rightarrow B \\
\sigma = \vdots; \Theta \rightarrow A \supset B \\
\rightarrow \supseteq \left\{ \begin{array}{l}
\Theta \subseteq Cl(\Gamma) \cap \overline{\Gamma} \\
A \in Cl(\Gamma) \setminus Cl(\Theta),
\end{array} \right.
\]

where \(\overline{\Gamma}\) is defined as in Figure 1. Let \(\sigma_1'\) be such that \(\sigma_1 \subseteq \sigma_1'\), namely, \(\sigma_1' = \Gamma' \not\Rightarrow B\), where \(\Gamma \subseteq \Gamma'\). Note that \(A \in Cl(\Gamma') \cap \overline{\Gamma}\) and \(A \notin Cl(\Theta)\). Let \(\Theta'\) be a maximal set such that \(\Theta \subseteq \Theta' \subseteq Cl(\Gamma') \cap \overline{\Gamma}\) and \(A \notin Cl(\Theta')\). It follows that

\[
\begin{align*}
\sigma_1' &= \Gamma' \not\Rightarrow B \\
\sigma' &= \vdots; \Theta' \rightarrow A \supset B
\end{align*}
\]

is a sound application of rule \(\supseteq\). Since \(\sigma \subseteq \sigma'\), the assertion holds. \(\square\)

In Figure 8, we give a high-level description of the refutation-search procedure \(\text{FSearch}\) for FRJ(Γ, Λ) exploiting subsumption. We denote with \(\mathcal{D}_G\) the database storing the sequents proved by the refutation-search procedure, and we say that an instance of a rule having premises \(\sigma_1, \ldots, \sigma_n\) is active in \(\mathcal{D}_G\) if \(\{\sigma_1, \ldots, \sigma_n\} \subseteq \mathcal{D}_G\).

Now, we introduce the key notion of saturated database. Let \(G\) be a formula and let \(\mathcal{D}_G\) be a set of FRJ(Γ, Λ)-sequents:

(DB1) \(\mathcal{D}_G\) is a database for \(G\) iff, for every \(\sigma \in \mathcal{D}_G\), \(\vdash_{\text{FRJ}(G)} \sigma\).

(DB2) A database \(\mathcal{D}_G\) for \(G\) is saturated iff, for every FRJ(Γ, Λ)-sequent \(\sigma\) such that \(\vdash_{\text{FRJ}(G)} \sigma\), there exists \(\sigma' \in \mathcal{D}_G\) such that \(\sigma \subseteq \sigma'\).

We say that a refutation-search procedure is adequate iff, for every formula \(G\), the refutation-search for \(G\) terminates yielding either an FRJ(Γ, Λ)-refutation of \(G\) or a saturated databases for \(G\).

**Theorem 4.2.** \(\text{FSearch}\) is an adequate refutation-search procedure.

**Proof.** Let \(G\) be a formula. First, we notice that \(\text{FSearch}(G)\) terminates; indeed, since the set of FRJ(Γ, Λ)-sequents is finite, the main-loop at line 5 terminates either because a sequent of the
The forward refutation-search procedure FSearch.

```
1 Function FSearch(G)
   input : The goal formula G
   output : An FRJ(G)-refutation of G or the db of proved sequents D_G
2    D_G ← ∅  // db of proved sequents
3    I ← all axiom sequents // db of sequents proved in the last iteration
4    Π ← all axiom rules  // db of generated refutations
5 while I ≠ ∅ and D_G does not contain a sequent of the kind Γ ⊨ G do
6     D_G ← D_G ∪ I  // update D_G
7     I ← ∅
8     for every instance R of a rule of FRJ(G) active in D_G do
9      let σ be the conclusion of R
10     if D_G does not contain σ' such that σ ⊑ σ' then
11        // update I and Π
12        I ← I ∪ {σ}
13        let D_1, ..., D_n be the FRJ(G)-refutations of σ_1, ..., σ_n in Π
14        let D be the refutation of σ built by applying rule R to D_1, ..., D_n
15        Π ← Π ∪ {D}
16     end
17     end
18 if I = ∅ or D_G contains a sequent of the kind Γ ⊨ G then
19     return the refutation of Γ ⊨ G in Π
20 else
21     return D_G
22 endFun
```

Fig. 8. The forward refutation-search procedure FSearch.

kind Γ ⊨ G has been added to D_G or no new sequent has been generated and I = ∅. Moreover it is immediate to check that, if sequent σ is added to D_G at iteration n ≥ 0, then Π contains an FRJ(G)-refutation of σ of height n; hence D_G is a database for G. As a consequence, if there is no FRJ(G)-refutation of G, FSearch(G) terminates returning the set D_G. To prove that D_G is a saturated database, we show that, if D is an FRJ(G)-refutation of σ and h(D) = n, then, at some iteration k ∈ {1, ..., n + 1} of the main loop, a sequent σ' such that σ ⊑ σ' is added to D_G. If h(D) = 0, then σ is an axiom sequent, hence σ is added to D_G at the first iteration. Let us assume that h(D) = h (h > 0) and σ is obtained by applying rule R to premises σ_1, ..., σ_m. By induction hypothesis there exist sequents σ'_1, ..., σ'_m and integers k_1, ..., k_m such that, for every i ∈ {1, ..., m}, k_i ≤ h + 1, σ'_i is added to D_G at iteration k_i and σ_i ⊑ σ'_i. Let K = max{k_1, ..., k_m} and let σ' be the conclusion of the application of rule R having σ'_1, ..., σ'_h as premises. By Lemma 4.1, σ ⊑ σ'. Since R is active at iteration K, we get that, at iteration K + 1, either σ' is added to D_G or D_G contains a sequent σ'' such that σ ⊑ σ' ⊑ σ''. In both cases, we get the assertion. □

The refutation-search procedure FSearch can be modified so to maintain the database of proved sequents concise according with the following definition:

(DB3) a database D_G for G is compact iff, for every σ ∈ D_G there is no σ' ∈ D_G s.t. σ' ⊑ σ.

Note that the databases in Figures 2, 3, and 4 are compact. We show that a compact saturated database D*_G for G is the minimum saturated database for G, namely: for every saturated database D_G for G, it holds that D*_G ⊆ D_G.
Lemma 4.3. Let $G$ be a formula and let $D^1_G$ be a compact saturated database for $G$. For every saturated database $D^2_G$ for $G$, $D^1_G \subseteq D^2_G$.

Proof. Let $\sigma_1 \in D^1_G$ and let $D^2_G$ be a saturated database for $G$. By (DB1), we have $\vdash_{\text{FRJ}(G)} \sigma_1$; hence, by (DB2), there exists $\sigma_2 \in D^2_G$ such that $\sigma_1 \subseteq \sigma_2$. By (DB1), we have $\vdash_{\text{FRJ}(G)} \sigma_2$; hence, by (DB2), there exists $\sigma'_1 \in D^1_G$ such that $\sigma_2 \subseteq \sigma'_1$. By transitivity of $\subseteq$, we get $\sigma_1 \subseteq \sigma'_1$. Since $D^1_G$ is compact, it is not the case that $\sigma_1 \subsetneq \sigma'_1$ (see (DB3)); hence, $\sigma_1 = \sigma'_1$, which implies $\sigma_1 \subseteq \sigma_2 \subseteq \sigma_1$. By antisymmetry of $\subseteq$, we get $\sigma_1 = \sigma_2$; hence, $\sigma_1 \in D^2_G$. This proves $D^1_G \subseteq D^2_G$.

As an immediate consequence, we get:

Theorem 4.4. For every formula $G$, there exists a unique compact saturated database $D^*_G$ for $G$, which is the minimum saturated database for $G$.

Proof. A compact saturated databases $D^*_G$ for $G$ can be constructed by taking any saturated database for $G$ and removing the redundant sequents. By Lemma 4.3, $D^*_G$ is the minimum saturated database for $G$ and, by definition, it is unique.

In the refutation-search procedure of Figure 8, we can keep compact the database of proved sequents $D_G$ by tweaking the update instruction at line 6 as follows:

- If $\sigma$ is added, then we delete from $D_G$ all the sequents $\sigma'$ such that $\sigma' \subseteq \sigma$ and all the sequents $\sigma''$ such that $\sigma' \Rightarrow \sigma''$ (backward subsumption).

Clearly, the database built using this version of FSearch is compact and, using Lemma 4.1, we can prove that the corresponding version of FSearch is adequate.

5 THE CALCULUS $\text{Gbu}(G)$ AND COMPLETENESS OF FRJ($G$)

In this section, we present the sequent calculus $\text{Gbu}(G)$ to derive the validity (in IPL) of a formula $G$. We show that, whenever there is no FRJ($G$)-refutation of $G$, we can exploit a saturated database for $G$ to build a $\text{Gbu}(G)$-derivation of $G$, witnessing that $G$ is valid; this implies the completeness of FRJ($G$).

Let $G$ be a formula. The calculus $\text{Gbu}(G)$ can be seen as the dual of the calculus FRJ($G$) and the two calculi have similar features. In $\text{Gbu}(G)$, we have two kinds of sequents, where $\Psi \subseteq \text{Sl}(G)$ and $A \in \text{Sr}(G)$:

- Regular sequents of the form $\Psi \Rightarrow_g A$;
- Irregular sequents of the form $\Psi \Rightarrow_g A$.

The subscript $g$ on arrows is used to avoid confusion with FRJ($G$)-sequents; for $\tau = \Psi \Rightarrow_g A$ or $\tau = \Psi \Rightarrow_g A$, we set $\text{Lhs}(\tau) = \Psi$ and $\text{Rhs}(\tau) = A$. The distinction between regular and irregular sequents is immaterial from a logical viewpoint (both represent IPL-derivability of $A$ from assumptions $\Psi$), but it is crucial to get termination in backward proof-search. Given the goal formula $G$, we set

$$\Gamma^A = \text{Sl}(G) \cap \mathcal{V}, \quad \Gamma^L = \text{Sl}(G) \cap \mathcal{L}, \quad \Gamma = \Gamma^A \cup \Gamma^L.$$ 

The rules of the calculus $\text{Gbu}(G)$ are presented in Figure 9; it consists of two axiom rules, namely, Ax and Ll, and left and right rules for each connective. There are two rules for right implication, that is $\Rightarrow R_e$ and $\Rightarrow R_e$, depending on the condition $A \in \text{Cl}(\Psi)$. A $\text{Gbu}(G)$-sequent $\tau$ is valid (in IPL) iff the formula $(\land \text{Lhs}(\tau)) \Rightarrow \text{Rhs}(\tau)$ is valid (as usual, we set $\land \emptyset = \bot \lor \bot$). By $\vdash_{\text{Gbu}(G)} \tau$ we mean that $\tau$ is derivable in $\text{Gbu}(G)$, namely, there exists a $\text{Gbu}(G)$-derivation with root sequent $\tau$. We say that $G$ is derivable in $\text{Gbu}(G)$, denoted by $\vdash_{\text{Gbu}(G)} G$, iff the regular sequent $\top \Rightarrow_g G$ is
In this section, we prove that $\cal{G}_{bu}(G)$ is a sound and complete calculus for IPL, namely: $\vdash_{\cal{G}_{bu}(G)} G$ if and only if $G \in \text{IPL}$.

A $\cal{G}_{bu}(G)$-derivation can be trivially mapped to a derivation of the calculus $\cal{G}_{3i}$ for IPL [29]. Basically, one has to erase the distinction between regular and irregular sequents; in the translation of $\supset R \in$, one has to add the antecedent $A$ to the left of the premise. Thus:

**Lemma 5.1.** If $\vdash_{\cal{G}_{bu}(G)} \tau$, then $\tau$ is valid.

Accordingly, we get the Soundness of $\cal{G}_{bu}(G)$:

**Theorem 5.2 (Soundness of $\cal{G}_{bu}(G)$).** $\vdash_{\cal{G}_{bu}(G)} G$ implies $G \in \text{IPL}$.

The calculus $\cal{G}_{3i}$ is non-terminating, since the rule for left implication can be applied an unbounded number of times along a branch of a derivation. Thus, to get terminating backward proof-search, one has to introduce suitable machinery, such as loop-checking, to control the expansion of a branch (see, e.g., Reference [20]). Instead, backward proof-search in $\cal{G}_{bu}(G)$ always terminates and this is essentially due to the presence of two kinds of sequents and to the conditions of rules $\supset R \in$ and $\supset R / n$ element in irregular sequents. Similar features have been introduced in Reference [11], where the calculus $\cal{G}_{bu}$, a terminating variant of $\cal{G}_{3i}$, has been introduced; actually, the calculus $\cal{G}_{bu}(G)$ is closely related to $\cal{G}_{bu}$. Proof-search for the goal formula $G$ starts from the regular sequent $\cdot \Rightarrow A$ and alternates two phases, one only processing regular sequents, the other one dealing with irregular sequents. In the first phase, given a regular sequent $\tau_{reg}$, one of the rules $\text{Ax}$, $\bot$, $\land$, $\lor$, $\supset R \in$, $\supset R / n$ is (backward) applied to $\tau_{reg}$, and proof-search recursively continues on the obtained regular sequents. Since the mentioned rules are invertible, the choice is non-deterministic and does not require backtracking. This phase ends when we get a regular sequent of the kind $\Omega \Rightarrow D$, where $\Omega \subseteq \Gamma$ and either $D \in \mathcal{V}^+ \setminus \Omega$ or $D = D_1 \lor D_2$. At this point, we can apply either rule $\lor$ $\supset$, by selecting from $\Omega$ the main formula $A \supset B$, or $\lor R_1$ or $\lor R_2$. In either case, at least one premise is an irregular sequent $\tau_{irr} = \Omega \Rightarrow C$, and here we have a backtrack point: If we fail to build the derivation, then we have to resume and try another application (if any). Proof-search for the irregular sequent $\tau_{irr}$ enters a new phase, where the right formula $C$...
is eagerly destructured; indeed, only right rules can be (backward) applied to irregular sequents and, using the terminology of Reference [11], the formulas in $\Omega$ are blocked. If we eventually get an irregular sequent $\Omega \rightarrow_{\gamma} F$, with $F \in \mathcal{V}^{\perp}$, then the expansion of the branch with root $\tau_{irr}$ ends: if $F \in \Omega$, then the branch can be successfully closed by applying the axiom rule $Ax$, otherwise proof-search fails (no rule can be applied) and a backtrack step is needed. However, in proving $\tau_{irr}$ one could get a sequent $\tau_{irr}^{'} = \Omega \rightarrow_{\gamma} A \supset B$ such that $A \notin Cl(\Omega)$. At this place, only the rule $\supset R_{\varepsilon}$ can be applied: the next sequent to be proved is $\tau_{B} = A, \Omega \Rightarrow_{\gamma} B$ and this marks a change of phase (formulas in $\Omega \cup \{A\}$ are again unblocked). Note that $\supset R_{\varepsilon}$ is the only rule that, read bottom-up, turns an irregular sequent into a regular one; the crucial point is that, in expanding $\tau_{B}$, further applications of rule $\supset R_{\varepsilon}$ with main formula $A \supset B$ are not allowed. Indeed, after having applied $\supset R_{\varepsilon}$ to $\tau_{irr}^{'}$, the formula $A$ is permanently added to $\Omega$ (indeed, in bottom-up application of the rules, left contexts never decrease). Let us assume that, in proving $\tau_{B}$, we get an irregular sequent of the form $\Omega^{'} \rightarrow_{\gamma} A \supset B$. The application of $\supset R_{\varepsilon}$ is forbidden, due to the presence of $A$ in $\Omega^{'}$, and we are forced to apply rule $\supset R_{\varepsilon}$, which yields the irregular sequent $\Omega^{'} \rightarrow_{\gamma} B$. To get a change phase, the formula $B$ must be reduced to a formula $X \supset Y$, with $X \notin \Omega^{'}$, hence $X \neq A$. Since formulas occurring in the sequents are subformulas of the goal formula $G$, we conclude that proof-search is terminating.

To prove the above informal argumentation, we introduce a weight function $Wg$ on $Gbu(G)$-sequents such that, after the backward application of a rule, the weight of sequents decreases. Given a $Gbu(G)$-sequent $\tau$, the size of $\tau$, denoted by $|\tau|$, is the number of logical symbols occurring in $\tau$. Let us consider the instance

\[
\begin{array}{c}
\ldots \\tau^{'} \\ldots \ R \\
Lhs(\tau^{'}) = \Psi' \\
Lhs(\tau) = \Psi
\end{array}
\]

of an application of a rule of $Gbu(G)$, where $\tau$ is the conclusion and $\tau^{'}$ any of the premises. We prove the following properties:

1. $||Sl(G) \setminus Cl(\Psi^{'})|| \leq ||Sl(G) \setminus Cl(\Psi)||$.
2. $R = \supset R_{\varepsilon}$ implies $||Sl(G) \setminus Cl(\Psi^{'})|| < ||Sl(G) \setminus Cl(\Psi)||$.
3. If $\tau^{'}$ is not the leftmost premise of $L \supset$, then $|\tau^{'}| < |\tau|$.

Point (1) follows by the fact that $Cl(\Psi) \subseteq Cl(\Psi^{'})$, which implies $Sl(G) \setminus Cl(\Psi^{'}) \subseteq Sl(G) \setminus Cl(\Psi)$. If $R$ is the rule $\supset R_{\varepsilon}$, then we have $\Psi' = \Psi \cup \{A\}$, with $A \notin Cl(\Psi)$. Thus, $A \in Sl(G) \setminus Cl(\Psi)$ and, since $A \in Cl(\Psi^{'})$, we have $A \notin Sl(G) \setminus Cl(\Psi^{'})$. This proves Point (2). Point (3) can be easily checked. By points (1)–(3), we can define $Wg(\tau)$ as the triple of non-negative integers:

\[
Wg(\tau) = \langle ||Sl(G) \setminus Cl(\Psi)||, tp(\tau), |\tau| \rangle \\
\text{tp}(\tau) = \begin{cases} 
1 & \text{if } \tau \text{ is regular} \\
0 & \text{otherwise,}
\end{cases}
\]

We get (where $<$ is lexicographic order on triples of integers):

**Lemma 5.3.** Let $R$ be a rule of $Gbu(G)$, let $\tau$ be the conclusion of $R$ and $\tau^{'}$ any of the premises of $R$. Then $\langle 0, 0, 0 \rangle \leq Wg(\tau^{'}) < Wg(\tau)$.

Note that the component $\text{tp}(\tau)$ of $Wg(\tau)$ accommodates the case where $\tau$ is the conclusion of rule $L \supset$ and $\tau^{'}$ the leftmost premise; in this case $\tau$ is regular and $\tau^{'}$ is irregular (thus, in proof-search there is a change of phase), hence $\text{tp}(\tau^{'}) < \text{tp}(\tau)$. As a consequence of Lemma 5.3, and reasoning as in Lemma 3.17 (i), we get a bound on the height of $Gbu(G)$-trees:

**Theorem 5.4.** Let $T$ be a $Gbu(G)$-tree and $\tau$ the root sequent of $T$. Then, the height of $T$ is $O(|\tau|^{2})$. 

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
Accordingly, given a Gbu(G)-sequent \( \tau \), the number of Gbu(G)-trees having root sequent \( \tau \) is finite, hence backward proof-search in Gbu(G) always terminates.

We present the procedure BSearch (Backward Search) to search for a Gbu(G)-derivation of a goal formula \( G \), namely, a Gbu(G)-derivation of \( \cdot \Rightarrow G \). This essentially corresponds to the backward-proof search procedure described above (see also Reference [11]). The relevant novelty is that now we avoid backtracking by exploiting a saturated database \( D_G \); Whenever only the non-invertible rules \( L \supset, RV_1 \) and \( RV_2 \) can be applied, we query \( D_G \) as an oracle for an advantageous choice. To extract from \( D_G \) the relevant information, we introduce the following evaluation relation \( \triangleright \):

- \( D_G \triangleright \Psi \Rightarrow C \) iff there exists \( \Gamma \not\Rightarrow C \in D_G \) such that \( \Psi \subseteq Cl(\Gamma) \);
- \( D_G \triangleright \Omega \rightarrow g C \) iff there exists \( \Sigma; \Theta \rightarrow C \in D_G \) such that \( \Sigma \subseteq \Omega \subseteq \Sigma \cup \Theta \).

By \( D_G \not\triangleright \tau \), we mean that \( D_G \triangleright \tau \) does not hold. The recursive function BSearch satisfies the following specification:

- Let \( \tau \) be a Gbu(G)-sequent and \( D_G \) a saturated database for \( G \) such that:
  
  \begin{enumerate}
  \item[(Bsr1)] \( D_G \triangleright \tau \);
  \item[(Bsr2)] if \( \tau = \Omega \rightarrow g C \), then \( \Omega \subseteq \overline{\Gamma} \).
  \end{enumerate}

Then BSearch(\( \tau, D_G \)) builds a Gbu(G)-derivation \( \mathcal{T} \) of \( \tau \).

To identify the backtrack points, we introduce the following definition:

- A Gbu(G)-sequent \( \tau \) is critical if one of the following conditions hold:
  \begin{enumerate}
  \item[−] \( \tau = \Omega \Rightarrow g D \) and \( \Omega \subseteq \overline{\Gamma} \) and either \( D \in \mathcal{V} \setminus \Omega \) or \( D = C_1 \lor C_2 \);
  \item[−] \( \tau = \Omega \rightarrow g C_1 \lor C_2 \) and \( \Omega \subseteq \overline{\Gamma} \).
  \end{enumerate}

Non-critical sequents can be characterized as follows:

**Lemma 5.5.** Let \( \tau \) be a non-critical Gbu(G)-sequent satisfying (Bsr1) and (Bsr2) with respect to a saturated database \( D_G \) for \( G \).

(i) If \( \tau = \Psi \Rightarrow g C \), then a formula of the kind \( \perp, A \land B, A \lor B \) belongs to \( \Psi \) or \( C \in \mathcal{V} \cap \Psi \) or \( C = A \land B \) or \( C = A \lor B \).

(ii) If \( \tau = \Psi \rightarrow g C \), then \( \Psi \subseteq \overline{\Gamma} \) and either \( C \in \mathcal{V} \cap \Psi \) or \( C = A \land B \) or \( C = A \lor B \).

**Proof.** Point (i) is immediate, since it corresponds to the negation of the definition of critical sequent for \( \tau \). Let \( \tau = \Psi \rightarrow g C \). By (Bsr2), we get \( \Psi \subseteq \overline{\Gamma} \). Since \( \tau \) is not critical, it follows that \( C \neq C_1 \lor C_2 \), namely, \( C \in \mathcal{V} \) or \( C = A \land B \) or \( C = A \lor B \). To conclude the proof of (ii), we show that:

\begin{enumerate}
  \item[(a)] \( C \neq \perp \).
  \item[(b)] \( C \in \mathcal{V} \) implies \( C \in \Psi \).
\end{enumerate}

We prove (b) (the proof of (a) is similar). Let \( C \in \mathcal{V} \) and let us assume, by absurd, \( C \notin \Psi \). Since \( \sigma = \cdot \mapsto \overline{\Gamma \setminus \{C\}} \rightarrow C \) is an axiom of FRJ(G) and \( D_G \) is a saturated database for \( G \), we get \( \sigma \in D_G \). This implies \( D_G \triangleright \tau \), in contradiction with (Bsr1); thus, (b) holds.

The high-level definition of BSearch is given in Figure 10. We distinguish four cases (B1)–(B4), according to the form of \( \tau \):

- (B1) \( \tau \) is a non-critical sequent;
- (B2) \( \tau \) is a critical sequent of the form \( \Omega \rightarrow g C_1 \lor C_2 \).

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
Function BSearch(τ, DG)

input : A sequent τ and a saturated database DG such that:
(BSr1) DG ⊢ τ;
(BSr2) if τ = Ω ⊢g C, then Ω ⊆ ¯F.

output : A Gbu(G)-derivation T of τ

if τ is a non-critical sequent then // Case (B1)

| T is the Gbu(G)-derivation only consisting of τ |

else

let τ1, . . . , τn be the premises of any instance of a rule R ∈ {LA, R∧, LV, ⊃ Re, ⊃ Rg}

for every 1 ≤ j ≤ n, let Tj = BSearch(τj, DG);

| T = τ1 . . . τn R |

else if τ is the critical sequent Ω ⊢g C1 ∨ C2 then // Case (B2)

choose any Ck ∈ {C1, C2} such that DG ⊢ Ω ⊢g Ck;

let Tk = BSearch(Ω ⊢g Ck, DG);

| T = Ω ⊢g Ck |

else if τ is the critical sequent Ω ⇒ F, with F ∈ Ψ⊥ then // Case (B3)

choose any A ⊃ B ∈ Ω s.t. DG ⊢ Ω ⊢g A and let ΩB = (Ω \ A ⊃ B) ∪ {B};

let T1 = BSearch(Ω ⊢g A, DG) and T2 = BSearch(ΩB ⊢g F, DG);

| T = Ω ⊢g A |

else if τ is the critical sequent τ = Ω ⊢g C1 ∨ C2 then // Case (B4)

let Y = {A | A ⊃ B ∈ Ω};

choose any Z ∈ Y ∪ {C1, C2} such that DG ⊢ Ω ⊢g Z;

if Z ∈ Y then

let A ⊃ B ∈ Ω such that Z = A and let ΩB = (Ω \ A ⊃ B) ∪ {B};

let T1 = BSearch(Ω ⊢g A, DG) and T2 = BSearch(ΩB ⊢g C1 ∨ C2, DG);

| T = Ω ⊢g A |

else

let Z = Ck (k ∈ {1, 2}) and Tk = BSearch(Ω ⊢g Ck, DG);

| T = Ω ⊢g Ck |

return T

Fig. 10. The backward proof-search procedure BSearch

(B3) τ is a critical sequent of the form Ω ⇒g F, with F ∈ Ψ⊥;
(B4) τ is a critical sequent of the form Ω ⇒g C1 ∨ C2.

For each case, a Gbu(G)-derivation T of τ is built. In Case (B1) at least an invertible rule can be applied by Lemma 5.5 (and the choice is non-deterministic). In the remaining cases, we have to apply a non-invertible rule, and we query the saturated database DG for a convenient choice; this
involves the evaluation of an irregular sequent in \( D_C \). In the proof of Theorem 5.11 (Correctness of BSearch), we show that, in each of the cases \((B2) – (B4)\), at least one of the displayed options can be selected (for instance, in Case \((B2)\) it is not possible that both \( D_C \supset \Omega \rightarrow_g C_1 \) and \( D_C \supset \Omega \rightarrow_g C_2 \) hold). We point out that in BSearch there are no backtrack points. We discuss an example of computation of BSearch.

**Example 5.6.** Let \( E \) be the following goal formula:

\[
E = (p \land A \land B \land C) \rightarrow D,
\]

\[
A = p \supset (q_1 \lor q_2), \quad B = q_1 \supset D \quad C = q_2 \supset D \quad D = r_1 \lor r_2,
\]

\[
\text{Lhs}(E) \cap (V \cup \mathcal{L}^\supset) = \{ p, q_1, q_2, r_1, r_2, A, B, C \},
\]

\[
\text{Rhs}(E) = \{ p, q_1, q_2, r_1, r_2, D, E \}.
\]

Since \( E \) is valid and the refutation-search procedure FSearch of Figure 8 is adequate (see Theorem 4.2), the call FSearch(\( E \)) returns a saturated database for \( E \). We can exploit it to run the procedure BSearch and build a Gbu(\( E \))-derivation of \( E \). We consider the compact saturated database \( D^*_E \) for \( E \) containing the following irregular sequents \( \sigma(1), \ldots, \sigma(6) \) (we omit the regular sequents, since they are not used by BSearch):

\[
\begin{align*}
(1) & \quad \vdash q_1, q_2, r_1, r_2, A, B, C \rightarrow p & \text{Ax}_\omega \\
(2) & \quad \vdash p, q_2, r_1, r_2, A, B, C \rightarrow q_1 & \text{Ax}_\omega \\
(3) & \quad \vdash p, q_1, r_1, r_2, A, B, C \rightarrow q_2 & \text{Ax}_\omega \\
(4) & \quad \vdash p, q_1, q_2, r_2, A, B, C \rightarrow r_1 & \text{Ax}_\omega \\
(5) & \quad \vdash p, q_1, q_2, r_1, A, B, C \rightarrow r_2 & \text{Ax}_\omega \\
(6) & \quad \vdash p, q_1, q_2, A, B, C \rightarrow D & \vee (4)(5)
\end{align*}
\]

Proof-search starts from the non-critical sequent \( \vdash \rightarrow_g E \). Rules of Gbu(\( E \)) are backward applied according with Case \((B1)\) (lines 2–8), until the critical sequent \( \tau_1 \) is obtained:

\[
\frac{
\begin{align*}
\frac{\vdash \rightarrow_g D}{p \land A \land B \land C \rightarrow_g D} \quad \text{(L} \land \text{ (three times))}
\end{align*}
}{\vdash \rightarrow_g E}
\]

To continue the construction of the derivation, we have to bottom-up apply a rule of Gbu(\( E \)) to the critical sequent \( \tau_1 \), and we query \( D^*_E \) for an informed decision. We have five possible options:

\[(c1) \text{ Apply rule } L \supset \text{ with main formula } A; \text{ by } (B4), \text{ this requires } D^*_E \not\vdash \Psi_1 \rightarrow_g p.\]

\[(c2) \text{ Apply rule } L \supset \text{ with main formula } B; \text{ by } (B4), \text{ this requires } D^*_E \not\vdash \Psi_1 \rightarrow_g q_1.\]

\[(c3) \text{ Apply rule } L \supset \text{ with main formula } C; \text{ by } (B4), \text{ this requires } D^*_E \not\vdash \Psi_1 \rightarrow_g q_2.\]

\[(c4) \text{ Apply rule } R \lor_1; \text{ by } (B4), \text{ this requires } D^*_E \not\vdash \Psi_1 \rightarrow_g r_1.\]

\[(c5) \text{ Apply rule } R \lor_2; \text{ by } (B4), \text{ this requires } D^*_E \not\vdash \Psi_1 \rightarrow_g r_2.\]

We have:

\[
\begin{align*}
D^*_E \supset \Psi_1 \rightarrow_g q_1 & \quad \text{(see } \sigma(3)), & \quad D^*_E \supset \Psi_1 \rightarrow_g q_2 & \quad \text{(see } \sigma(3)), \\
D^*_E \supset \Psi_1 \rightarrow_g r_1 & \quad \text{(see } \sigma(4)), & \quad D^*_E \supset \Psi_1 \rightarrow_g r_2 & \quad \text{(see } \sigma(5)), \\
D^*_E \not\vdash \Psi_1 \rightarrow_g p & ;
\end{align*}
\]
hence, only choice (c1) can be performed. Note that, selecting any of the other options (c2)–(c5), where the corresponding condition is not matched, proof-search fails. Complying with (c1), we apply rule $L \supset$ with main formula $A$, and we continue until we get the critical sequents $\tau_2$ and $\tau_3$.

\[
\begin{array}{c}
\frac{\text{Ax}}{p, A, B, C \Rightarrow g \ p} \quad \frac{\text{Ax}}{\tau_2 = \Psi_2 \Rightarrow g \ D} \\
\tau_1 = p, A, B, C \Rightarrow g \ D
\end{array}
\]

\[
L \supset
\begin{array}{c}
\frac{\text{Ax}}{\tau_2 = \Psi_2 \Rightarrow g \ D} \\
\tau_3 = \Psi_3 \Rightarrow g \ D
\end{array}
\]

\[
\Psi_2 = p, q_1, B, C,
\]

\[
\Psi_3 = p, q_2, B, C.
\]

Let us consider the critical sequent $\tau_2$. We have four possible choices:

(c6) Apply rule $L \supset$ with main formula $B$; by (B4), this requires $D_E^* \not\vdash \Psi_2 \rightarrow g \ q_2$.

(c7) Apply rule $L \supset$ with main formula $C$; by (B4), this requires $D_E^* \not\vdash \Psi_2 \rightarrow g \ q_2$.

(c8) Apply rule $RV_1$; by (B4), this requires $D_E^* \not\vdash \Psi_2 \rightarrow g \ r_1$.

(c9) Apply rule $RV_2$; by (B4), this requires $D_E^* \not\vdash \Psi_2 \rightarrow g \ r_2$.

We note that:

\[
\begin{array}{c}
D_E^* \not\vdash \Psi_2 \rightarrow g \ q_2 \quad \text{(see } \sigma(3)\text{),}\quad D_E^* \not\vdash \Psi_2 \rightarrow g \ r_1 \quad \text{(see } \sigma(4)\text{),}\quad D_E^* \not\vdash \Psi_2 \rightarrow g \ q_1.
\end{array}
\]

Thus, we choose (c6) and we continue until we get the critical sequents $\tau_4$ and $\tau_5$:

\[
\begin{array}{c}
\frac{\text{Ax}}{\tau_4 = \Psi_4 \Rightarrow g \ D} \\
\tau_5 = \Psi_5 \Rightarrow g \ D
\end{array}
\]

\[
L \supset
\begin{array}{c}
\frac{\text{Ax}}{\tau_3 = \Psi_3 \Rightarrow g \ D} \\
\frac{\text{Ax}}{\tau_2 = p, q_1, B, C \Rightarrow g \ D}
\end{array}
\]

\[
\Psi_4 = p, q_1, r_1, C,
\]

\[
\Psi_5 = p, q_1, r_2, C.
\]

For the critical sequent $\tau_4$, we have three possible options:

(c10) Apply rule $L \supset$ with main formula $C$; by (B4), this requires $D_E^* \not\vdash \Psi_4 \rightarrow g \ q_2$.

(c11) Apply rule $RV_1$; by (B4), this requires $D_E^* \not\vdash \Psi_4 \rightarrow g \ r_1$.

(c12) Apply rule $RV_2$; by (B4), this requires $D_E^* \not\vdash \Psi_4 \rightarrow g \ r_2$.

We have

\[
D_E^* \not\vdash \Psi_4 \rightarrow g \ q_2 \quad \text{(see } \sigma(5)\text{),}\quad D_E^* \not\vdash \Psi_4 \rightarrow g \ r_2 \quad \text{(see } \sigma(5)\text{),}\quad D_E^* \not\vdash \Psi_4 \rightarrow g \ r_1.
\]

Thus, we select (c11), and we get

\[
\tau_4 = p, q_1, r_1, C \Rightarrow g \ r_1 \\
R_{V_1}
\]

The $\text{Gbu}(E)$-derivations of the sequents $\tau_3$ and $\tau_5$ have a similar construction:

\[
\begin{array}{c}
\frac{\text{Ax}}{\tau_5 = \Psi_5 \Rightarrow g \ D} \\
\frac{\text{Ax}}{\tau_3 = \Psi_3 \Rightarrow g \ D}
\end{array}
\]

\[
L \supset
\begin{array}{c}
\frac{\text{Ax}}{p, q_2, B, r_1 \Rightarrow g \ r_1} \\
\frac{\text{Ax}}{p, q_2, B, r_2 \Rightarrow g \ r_2}
\end{array}
\]

\[
L \supset
\begin{array}{c}
\frac{\text{Ax}}{p, q_2, B, R_{V_1}} \\
\frac{\text{Ax}}{p, q_2, B, R_{V_2}}
\end{array}
\]

and this completes the definition of the $\text{Gbu}(E)$-derivation of $E$.

Now, we prove the correctness of BSEARCH. We start by showing some properties of the evaluation relation.

**Lemma 5.7.** Let $G$ be a formula, let $D_G$ be a saturated database for $G$, $\Psi \subseteq \Delta(G)$ and $\Omega \subseteq \overline{\Gamma}$. 

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
Proof. We only detail some representative cases.

Proof of (i). Let $D_G \triangleright A, B, \Psi \Rightarrow g C$. Then, there exists $\Gamma$ such that:

- $\Gamma \not\Rightarrow C \in D_G$;
- $\Psi \cup \{A, B\} \subseteq Cl(\Gamma)$.

Since $A \land B \in Cl(\Gamma)$, we have $\Psi \cup \{A \land B\} \subseteq Cl(\Gamma)$, hence $D_G \triangleright A \land B, \Psi \Rightarrow g C$.

Proof of (ii). Let $C_1 \land C_2 \in Sr(G)$ and let us assume $D_G \triangleright \Psi \Rightarrow g C_k$, with $k \in \{1, 2\}$. Then, there exists $\Gamma$ such that:

- $\Gamma \not\Rightarrow C_k \in D_G$;
- $\Psi \subseteq Cl(\Gamma)$.

By (DB1), $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow C_k$, hence $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow C_1 \land C_2$. By (DB2), $D_G$ contains a sequent $\Gamma' \not\Rightarrow C_1 \land C_2$ such that $\Gamma \subseteq \Gamma'$. By (C14), $Cl(\Gamma) \subseteq Cl(\Gamma')$, which implies $\Psi \subseteq Cl(\Gamma')$, hence $D_G \triangleright \Psi \Rightarrow g C_1 \land C_2$.

Proof of (v). Let $A \supset B \in Sr(G)$ and let us assume $D_G \triangleright \Omega \rightarrow g B$, with $A \in Cl(\Psi)$. Then, there exists $\Gamma$ such that:

- $\Gamma \not\Rightarrow B \in D_G$;
- $\Psi \subseteq Cl(\Gamma)$.

By (DB1), $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow B$. By (C16), we get $Cl(\Psi) \subseteq Cl(\Gamma)$, hence $A \in Cl(\Gamma)$; this implies $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow A \supset B$. Reasoning as in the proof of case (ii), we get $D_G \triangleright \Psi \Rightarrow g A \supset B$.

Proof of (vi). Let $A \supset B \in Sr(G)$ and let us assume $D_G \triangleright A, \Psi \Rightarrow g B$, with $A \notin Cl(\Psi)$. Then, there exists $\Gamma$ such that:

- $\Gamma \not\Rightarrow B \in D_G$;
- $\Psi \cup \{A\} \subseteq Cl(\Gamma)$.

By (DB1), $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow B$, hence $\triangleright_{FRJ(G)} \Gamma \not\Rightarrow A \supset B$. Reasoning as in the proof of case (ii), we get $D_G \triangleright \Psi \Rightarrow g A \supset B$.

Proof of (viii). Let us assume $D_G \triangleright \Omega \rightarrow g B$ and $A \in Cl(\Omega)$. Then, there exist $\Sigma$ and $\Theta$ such that:

- $\Sigma \cup \Theta \not\Rightarrow B \in D_G$;
- $\Sigma \subseteq \Omega \subseteq \Sigma \cup \Theta$.

Since $\Omega \subseteq \Sigma \cup \Theta$, by (C14), we get $Cl(\Omega) \subseteq Cl(\Sigma \cup \Theta)$, hence $A \in Cl(\Sigma \cup \Theta)$. Let $\Lambda$ be a minimum (possibly empty) subset of $\Theta$ such that $A \in Cl(\Sigma \cup \Lambda)$. Since $A \in Cl(\Omega)$ and $\Sigma \subseteq \Omega \subseteq \Sigma \cup \Theta$, we can choose $\Lambda$ so that $\Sigma \cup \Lambda \subseteq \Omega$. By (DB1) $\triangleright_{FRJ(G)} \Sigma \cup \Theta \not\Rightarrow B$; hence, we can build the $\triangleright_{FRJ(G)}$-
refutation:

\[
\begin{array}{c}
\vdots \\
\Theta_1 = \Theta \setminus \Lambda, \\
\Lambda \subseteq \Theta \text{ and } A \in Cl(\Sigma \cup \Lambda), \\
\Sigma; \Lambda, \Theta_1 \rightarrow A \supset B \supseteq e \\
\Sigma, \Lambda; \Theta_1 \rightarrow A \supset B \supseteq e \\
\end{array}
\]

Note that the above refutation matches (RS1). By (DB2), $D_G$ contains a sequent $\Sigma, \Lambda; \Theta' \rightarrow A \supset B$ such that $\Theta_1 \subseteq \Theta'$. Since $\Omega \subseteq \Sigma \cup \Theta$ and $\Sigma \cup \Theta = \Sigma \cup \Lambda \cup \Theta_1$, we get $\Omega \subseteq \Sigma \cup \Lambda \cup \Theta'$. Thus, $\Sigma \cup \Lambda \subseteq \Omega \subseteq \Sigma \cup \Lambda \cup \Theta'$. This means $\Sigma \cup \Lambda \subseteq \Omega \subseteq \Sigma \cup \Lambda \cup \Theta'$, and this proves that $D_G \triangleright \Omega \rightarrow_g A \supset B$.

Proof of (ix). Let us assume $D_G \triangleright A, \Omega \rightarrow_g B$ and $A \notin Cl(\Omega)$. Then, there exists $\Gamma$ such that:

- $\Gamma \nrightarrow B \in D_G$;
- $\Omega \cup \{A\} \subseteq Cl(\Gamma)$.

Let $\Theta$ be a maximal subset of $Cl(\Gamma) \cap \bar{\Gamma}$ such that $A \notin Cl(\Theta)$; since $\Omega \subseteq Cl(\Gamma) \cap \bar{\Gamma}$ and $A \notin Cl(\Omega)$, we can choose $\Theta$ so that $\Omega \subseteq \Theta$. By (DB1), $^+_{FRJ(G)} \Gamma \nrightarrow B$; hence, we can build the following $FRJ(G)$-refutation:

\[
\begin{array}{c}
\vdots \\
\Omega \subseteq \Theta \subseteq Cl(\Gamma) \cap \bar{\Gamma}, \\
\Lambda \subseteq \Theta \leq Cl(\Gamma), \\
\Theta \subseteq \Theta' \subseteq Cl(\Gamma) \cap \bar{\Gamma} \text{ implies } A \in Cl(\Theta'). \\
\end{array}
\]

Note that the above refutation matches (RS2). By (DB2), $D_G$ contains a sequent $\cdot; \Theta' \rightarrow A \supset B$, where $\Theta \subseteq \Theta'$; since $\Omega \subseteq \Theta'$, we get $D_G \triangleright \Omega \rightarrow_g A \supset B$. □

**Lemma 5.8.** Let $G$ be a formula, let $D_G$ be a saturated database for $G$ and $\Omega \subseteq \bar{\Gamma}$. If $D_G \triangleright \Omega \rightarrow_g C_1$ and $D_G \triangleright \Omega \rightarrow_g C_2$, then $D_G \triangleright \Omega \rightarrow_g C_1 \lor C_2$.

**Proof.** Let us assume $D_G \triangleright \Omega \rightarrow_g C_1$ and $D_G \triangleright \Omega \rightarrow_g C_2$. For $k \in \{1, 2\}$, there exist $\Sigma_k$ and $\Theta_k$ be such that:

- $\Sigma_k; \Theta_k \rightarrow C_k \in D_G$;
- $\Sigma_k \subseteq \Omega \subseteq \Sigma_k \cup \Theta_k$.

By (DB1), $^+_{FRJ(G)} \Sigma_k; \Theta_k \rightarrow C_k$. Note that $\Sigma_1 \subseteq \Sigma_2 \cup \Theta_2$ and $\Sigma_2 \subseteq \Sigma_1 \cup \Theta_1$; hence, we can build the following $FRJ(G)$-refutation:

\[
\begin{array}{c}
\vdots \\
\Sigma_1; \Theta_1 \rightarrow C_1 \\
\Sigma_2; \Theta_2 \rightarrow C_2 \\
\Sigma_1, \Sigma_2; \Theta_1 \cap \Theta_2 \rightarrow C_1 \lor C_2 \\
\end{array}
\]

By (DB2), $D_G$ contains a sequent $\Sigma; \Theta' \rightarrow C_1 \lor C_2$ such that $\Theta \subseteq \Theta'$. One can easily check that $\Sigma \subseteq \Omega$ and $\Omega \subseteq \Sigma \cup \Theta$, which implies $\Omega \subseteq \Sigma \cup \Theta'$. We conclude $D_G \triangleright \Omega \rightarrow_g C_1 \lor C_2$. □

**Lemma 5.9.** Let $G$ be a formula and let $D_G$ be a saturated database for $G$. Let $\tau = \Omega \rightarrow_g F$ be such that $\Omega \subseteq \bar{\Gamma}$ and $F \in \langle \mathcal{V}^\perp \rangle$, and let us assume that:

(i) $F \notin \Omega$;

(ii) for every $A \supset B \in \Omega$, $D_G \triangleright \Omega \rightarrow_g A$.

Then, $D_G \triangleright \tau$. 

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
Proof. Let \( \Omega = \Omega^A \cup \Omega^- \). Let us assume that \( \Omega^- \) is empty, namely, \( \Omega = \Omega^A \), and let \( \Gamma' = \Gamma^A \setminus \{ F \} \). Then, \( \sigma' = \Gamma' \Rightarrow F \) is an axiom sequent of \( \text{FRJ}(G) \) hence, by \( (DB2) \), \( D_G \) contains a sequent \( \sigma'' = \Gamma \Rightarrow F \) with \( \Gamma' \subseteq \Gamma \). Since \( \Omega^A \subseteq \Gamma \), by \( (C13) \), we get \( \Omega^A \subseteq \text{Cl}(\Gamma) \), hence \( D_G \vdash \tau \).

Let \( \Omega^- \) be nonempty and let \( \Omega^- \) = \{ \( A_1 \supset B_1, \ldots, A_n \supset B_n \) \} \( (n \geq 1) \). For every \( k \in \{1, \ldots, n\} \), since \( D_G \vdash \Omega \rightarrow_g A_k \), there are \( \Sigma_k \) and \( \Theta_k \) such that:

- \( \Sigma_k \vdash \Theta_k \rightarrow A_k \in D_G \);
- \( \Sigma_k \subseteq \Omega \subseteq \Sigma_k \cup \Theta_k \).

By \( (DB1) \), \( \vdash_{\text{FRJ}(G)} \Sigma_k \vdash \Theta_k \rightarrow A_k \). One can easily check that, for \( i \neq j \), it holds that \( \Sigma_i \subseteq \Sigma_j \cup \Theta_j \).

For \( k \in \{1, \ldots, n\} \), let \( \Sigma_k = \Sigma^A_k \cup \Sigma^-_k \) and \( \Theta_k = \Theta^A_k \cup \Theta^-_k \). We can build the following \( \text{FRJ}(G) \)-refutation \( (\Sigma^A, \Sigma^-, \Theta^A, \Theta^-) \) are defined as in Figure 1):

\[
\vdash \Sigma_1; \Theta_1 \rightarrow A_1 \ldots \Sigma_n; \Theta_n \rightarrow A_n \quad \vdash \Gamma \not\vdash F
\]

Note that the above refutation matches the restriction \( (RS3) \) on the application of \( \vdash \) stated in Section 3. By \( (DB2) \), \( D_G \) contains a sequent \( \Gamma' \Rightarrow F \), with \( \Gamma' \subseteq \Gamma \); by \( (C14) \), \( \text{Cl}(\Gamma) \subseteq \text{Cl}(\Gamma') \). One can easily check that \( \Omega \subseteq \Gamma \); hence, by \( (C13) \), \( \Omega \subseteq \text{Cl}(\Gamma) \). Thus, \( \Omega \subseteq \text{Cl}(\Gamma') \), which implies \( D_G \vdash \Omega \rightarrow_g F \).

In a similar way, we can prove that:

Lemma 5.10. Let \( G \) be a formula and let \( D_G \) be a saturated database for \( G \). Let \( \tau = \Omega \rightarrow_g C_1 \lor C_2 \) be such that \( \Omega \subseteq \Gamma \), and let us assume that:

(i) for every \( A \supset B \in \Omega, D_G \vdash \Omega \rightarrow_g A \).
(ii) \( D_G \vdash \Omega \rightarrow_g C_1 \) and \( D_G \vdash \Omega \rightarrow_g C_2 \).

Then, \( D_G \vdash \tau \).

We prove the correctness of BSearch.

Theorem 5.11 (Correctness of BSearch). Let \( G \) be a formula, let \( \tau \) be a \( \text{Gbu}(G) \)-sequent and let \( D_G \) be a saturated database for \( G \) satisfying \( (BSr1) \) and \( (BSr2) \). Then, BSearch(\( \tau, D_G \)) computes a \( \text{Gbu}(G) \)-derivation of \( \tau \).

Proof. We prove the assertion by induction on \( Wg(\tau) \). Note that, whenever we perform a recursive call BSearch(\( \tau', D_G \)), it holds that \( Wg(\tau') < Wg(\tau) \). Thus, by the induction hypothesis, we can assume that:

\((\dag)\) every recursive call BSearch(\( \tau', D_G \)) yields a \( \text{Gbu}(G) \)-derivation of \( \tau' \), provided that \( \tau' \) and \( D_G \) satisfy assumptions \( (BSr1) \) and \( (BSr2) \).

We have to show that, in each of the cases \( (B1)-(B4) \), \( \Upsilon \) defines a \( \text{Gbu}(G) \)-derivation of \( \tau \) (such a derivation is returned at line 27). Let us consider Case \( (B1) \), namely, \( \tau \) is a non-critical sequent. If \( \tau \) is an axiom, then \( \Upsilon \) (line 4) is \( \text{Gbu}(G) \)-derivation of \( \tau \). Otherwise, by Lemma 5.5 at least an application of a rule \( R \in \{ L\Lambda, \Lambda \land, \Lambda \lor, \supset R_e, \supset R_q \} \) with conclusion \( \tau \) is possible. We have to show that each premise \( \tau_j \) of the selected rule \( R \) satisfies assumptions \( (BSr1) \) and \( (BSr2) \). Let us consider assumption \( (BSr2) \); if \( \tau_j \) is regular, then \( (BSr2) \) trivially holds. If \( \tau_j \) is irregular, then, by
inspecting the rules of Gbu(G), one can check that Lhs(τj) = Lhs(τ); thus, since τ satisfies (BSr2), τj satisfies (BSr2) as well. The validity of (BSr1) follows by Lemma 5.7. For instance, let us assume that R = LV and that the selected application is

\[ \frac{τ_1 = A, Ψ \Rightarrow_g C \quad τ_2 = B, Ψ \Rightarrow_g C}{τ = A \lor B, Ψ \Rightarrow_g C} \]

Since τ satisfies (BSr1), we have D_G \not\vdash τ; by Lemma 5.7(iii), it follows that both D_G \not\vdash τ_1 and D_G \not\vdash τ_2, hence both τ_1 and τ_2 satisfy (BSr1). By (†), for every 1 \leq j \leq n, τj is a Gbu(G)-derivation of τj, hence τ (line 8) is a Gbu(G)-derivation of τ.

Let τ match Case (B2), namely τ is the critical sequent Ω \Rightarrow_g F, with F \in \mathcal{V} (Case (B3)) and let assume that, for every A \supset B \in \Omega, we have D_G \not\vdash Ω \Rightarrow_g A. By Lemma 5.9, it would follow D_G \not\vdash τ, against the assumption (BSr1) of the lemma. We can pick A \supset B \in Ω such that D_G \not\vdash Ω \Rightarrow_g A. By (†), τ is a Gbu(G)-derivation of τ_1 = Ω \Rightarrow_g A. By (†) and Lemma 5.7(iv), τ is a Gbu(G)-derivation of τ_1 = Ω \Rightarrow_g A. By (†) and Lemma 5.7(iv), τ_2 is a Gbu(G)-derivation of τ_2 = Ω_B \Rightarrow_g F. Thus, τ (line 12) is a Gbu(G)-derivation of τ.

The proof Case (B4) is similar to cases (B2) and (B3), exploiting Lemma 5.10. Since the enumerated cases are exhaustive, the correctness of BSEARCH follows.

By the correctness of BSEARCH, it follows that Gbu(G) and FRJ(G) are dual calculi:

**Theorem 5.12.** \( \vdash_{\text{Gbu}(G)} G \iff \vdash_{\text{FRJ}(G)} G \).

**Proof.** If \( \vdash_{\text{Gbu}(G)} G \), then by the Soundness of Gbu(G) (Theorem 5.2), we get G \in IPL; by the Soundness of FRJ(G) (Theorem 3.1), we conclude \( \vdash_{\text{FRJ}(G)} G \).

Conversely, let \( \vdash_{\text{FRJ}(G)} G \), let D_G be a saturated database for G and let τ be the Gbu(G)-sequent \( \cdot \Rightarrow_g G \). Note that D_G does not contain any regular sequent of the kind \( \sigma = \Gamma \not\vdash G \); otherwise, by (DB1), we would get \( \vdash_{\text{FRJ}(G)} \sigma \), contradicting the assumption that \( \vdash_{\text{FRJ}(G)} G \) does not hold. This implies D_G \not\vdash τ, hence τ and D_G satisfy the assumptions (BSr1) and (BSr2) of BSEARCH. By the correctness of BSEARCH (Theorem 5.11), BSEARCH(τ, D_G) computes a Gbu(G)-derivation of τ; we conclude \( \vdash_{\text{Gbu}(G)} G \).

As a corollary, we get the completeness of FRJ(G) and Gbu(G):

**Theorem 5.13 (Completeness of FRJ(G) and Gbu(G)).**

(i) G ∉ IPL implies \( \vdash_{\text{FRJ}(G)} G \).

(ii) G ∈ IPL implies \( \vdash_{\text{Gbu}(G)} G \).

**Proof.** Let G ∉ IPL. By the Soundness of Gbu(G) (Theorem 5.2), \( \vdash_{\text{Gbu}(G)} G \); by Theorem 5.12, \( \vdash_{\text{FRJ}(G)} G \). Let G ∈ IPL. By the Soundness of FRJ(G) (Theorem 3.1), \( \vdash_{\text{FRJ}(G)} G \); by Theorem 5.12, \( \vdash_{\text{Gbu}(G)} G \).

### 5.1 Implementation

To evaluate the potential of our approach, we have developed frj, a Java implementation of our refutation-search procedure based on the full-fledged framework JTabWb [13]. frj implements the forward refutation-search procedure FSearch with the redundancy checks based on forward
and backward subsumption and the backward proof-search procedure BSearch. We give some details about the implementation of FSearch. During pre-processing of the input problem, a unique identifier (a positive integer) is given to each formula \( H \), and a number of tables are used to map integers to their respective formula and to the set of its subformulas. Sets of formulas are implemented as bitsets of integers so that membership and subset relations can be tested in constant time. Sequents are data structures aggregating the sequent type (regular or irregular), the formula in the right-hand side and the bitset(s) representing the left-hand side. At each iteration of the main loop, \( \text{frj} \) applies all the possible instances of rules \( \land, \lor, \supset \) and \( \exists \) involving at least a premise proved in the last step. To manage join rules \( \forall \), \( \supset \) and \( \exists \), \( \text{frj} \) maintains a list of join-compatible sets \( J_k \), namely, \( J_k \) is a set of irregular sequents matching the side conditions (J1) and (J2) of join rules (see Figure 1). At each iteration the list is updated resting on the set \( I \) of irregular sequents proved in the last iteration. In particular, each join-compatible set \( J_k \) is possibly extended with elements of \( I \) and the new join-compatible sets issued from \( I \) are added. For every join-compatible set \( J_k \), every possible join rule having premises \( J_k \) is applied. Note that the huge number of possible applications of join rules is the main source of inefficiency in our forward refutation-search procedure. We also exploit backward subsumption to optimize the implementation of join rules: whenever backward subsumption is detected, every subsumed irregular sequent occurring in a join-compatible set is replaced by the subsuming one. To implement backward subsumption, we maintain a map associating, with every formula possibly occurring in right-hand side of a sequent, the list of proved irregular sequents. Since the subsumption check between two sequents \( \sigma_1 = \Sigma_1 ; \Theta_1 \rightarrow C \) and \( \sigma_2 = \Sigma_2 ; \Theta_2 \rightarrow C \) requires the tests \( \Sigma_1 = \Sigma_2 \) and \( \Theta_1 \subseteq \Theta_2 \), exploiting the bitset implementation of sets of formulas, we can perform the subsumption check between \( \sigma_1 \) and \( \sigma_2 \) in constant time. As a result the implementation of backward subsumption when inserting an irregular sequent \( \sigma = \Sigma ; \Theta \rightarrow C \) requires time proportional to the number of proved irregular sequents having \( C \) in the right hand side.

When the refutation-search for \( G \) succeeds, \( \text{frj} \) extracts a countermodel for \( G \), otherwise the saturated database \( D_G \) generated by the failed refutation-search in \( \text{FRJ} \)(\( G \)) is used by the implementation of BSearch to build a \( \text{Gbu}(G) \)-derivation of \( G \). We remark that the check \( D_G \not\models \Omega \rightarrow g C \), used by BSearch to resolve backtrack points, requires time proportional to the number of proved irregular sequents having \( C \) in the right-hand side; indeed for every irregular sequent \( \Sigma ; \Theta \rightarrow C \in D_G \), we have to check \( \Sigma \subseteq \Omega \subseteq \Sigma \cup \Theta \) and, exploiting the bitset implementation of sets of formulas, this requires constant time.

We show an example where the search for a \( \text{Gbu}(G) \)-derivation of a valid formula \( G \) using a saturated database can be more efficient than a standard proof-search procedure based on the calculus G3i.

**Example 5.14.** Let us consider the following goal formulas \( G_n \) \( (n \geq 1) \):
\[
G_n = (p \land (p \supset q) \land (a_1 \supset b_1) \land \cdots \land (a_n \supset b_n)) \supset q,
\]
\[
\text{Lhs}(G_n) \cap (\forall \cup \mathcal{L} \supset) = Y \cup \Lambda,
\]
\[
Y = \{p \supset q, a_1 \supset b_1, \ldots, a_n \supset b_n\},
\]
\[
\Lambda = \{p, q, b_1, \ldots, b_n\},
\]
\[
\text{Rhs}(G_n) = \{G_n, p, q, a_1, \ldots, a_n\}.
\]

Let us consider the call FSearch\( (G_n) \). Initially, all the \( \text{FRJ}(G_n) \)-axioms are computed; there are \( n + 2 \) regular axioms and \( n + 2 \) irregular axioms:
\[
\Lambda \setminus \{C\} \not\models C,
\]
\[
\vdots; Y, \Lambda \setminus \{C\} \not\models C, \quad C \in \{p, q, a_1, \ldots, a_n\}.
\]
We show that no rule can be applied to the regular axioms. Indeed, the only compound formula in \( \text{Rhs}(G_n) \) is \( G_n = H_n \supset q \), thus we could only apply rule \( p = \supset \lor \rho \supset \supset \) or \( p = \supset \land \) to get a sequent having
$G_n$ in the right-hand side. However, the premise of $\rho$ should be an axiom $\Gamma \not\Rightarrow q$ such that $H_n \in Cl(\Gamma)$, which implies $q \in \Gamma$, in contradiction with the definition of regular axiom. Similarly, one can check that only join rules can be applied to the irregular sequents; this requires $O(2^n)$ steps, since we have to consider all possible subsets of irregular axioms. The regular sequents generated by the application of join rules have the form $\Gamma \Rightarrow C$, where $C \in \{p, q, a_1, \ldots, a_n\}$; moreover, if $C = q$, then $q \notin \Gamma$. This implies that no rule can be applied to such sequents, hence the database of all the generated sequents $D_{G_n}$ is saturated; since $D_{G_n}$ does not contain any sequent of the form $\Gamma \Rightarrow G_n$, refutation-search for $G_n$ fails. We point out that $D_{G_n}$ only contains $n + 2$ irregular sequents (namely, the irregular axioms). Now, we call BSRepeat to build a $Gbu(G_n)$-derivation of the sequent $\tau = (\cdot \Rightarrow G_n)$ by exploiting the saturated database $D_{G_n}$. After the backward application of rules $\supset$ and $L \wedge$, in $O(n)$ steps, we get the critical sequent $p, \top \Rightarrow q$. Here, only the rule $L \supset$ can be applied, and we have to select from $\top$ the main formula of the application. By inspecting the irregular sequents in $D_{G_n}$, the only possible choice is $p \supset q$. We get the axioms $p, \top \Rightarrow q$ and $p, \top \not\supset q$, $q \Rightarrow q$ and the construction of the derivation ends.

Let us analyse a standard proof-search procedure to build a derivation of $G_n$ in G3i. After the backward application of the invertible rules $R \supset$ and $RA$, we get the sequent $\sigma = p, \top \Rightarrow q$. Here, we have a backtrack point, since we have to select a main formula for an application of $L \supset$. If we pick $p \supset q$, then as discussed above the derivation can be immediately completed. Let us assume to perform a blind selection and that the formulas in $\top$ are considered from right to left. One can easily realize that at least $n!$ open branches must be generated before reaching the formula $p \supset q$ and make the only right choice. To sum up, the overall time required by FSRepeat and BSRepeat to build an $Gbu(G_n)$-derivation of $G_n$ is $O(n^2)$. With a standard proof-search procedure based on G3i, time complexity ranges from $O(n)$ in the best case to $\Omega(n!)$ in the worst case.

### 6 MINIMALITY

In this section, we prove that, given a non-valid formula $G$, one can build an $FRJ(G)$-refutation $D$ of $G$ such that $Mod(D)$ is a countermodel of $G$ having minimal height. Such a refutation can be constructed by tweaking the refutation-search procedure defined in Section 4.

Let $\mathcal{K} = \langle P, \leq, \rho, V \rangle$ be a model and $\alpha \in P$; the height of $\alpha$, denoted by $h(\alpha)$, is defined as follows:

$$h(\alpha) = \begin{cases} 0 & \text{if } \alpha \text{ is a final world of } \mathcal{K}, \\ 1 + \max \{ h(\beta) \mid \alpha \prec \beta \} & \text{otherwise.} \end{cases}$$

The height of $\mathcal{K}$, denoted by $h(\mathcal{K})$, is the height of $\rho$. As an immediate consequence of Lemma 3.17 and of the definition of the model $Mod(D)$ extracted from $D$, we get an upper bound on the height of $FRJ(G)$-refutations and on the height of extracted countermodels:

**Theorem 6.1.** Let $D$ be an $FRJ(G)$-refutation and $N = |G|$. Then:

(i) $h(D) = O(N^2)$.

(ii) $h(\text{Mod}(D)) \leq N$.

Let $G \notin IPL$; the height of $G$, denoted by $h(G)$, is the minimum height of a countermodel for $G$; formally:

$$h(G) = \min \{ h(\mathcal{K}) \mid \mathcal{K} = \langle P, \leq, \rho, V \rangle \text{ and } \rho \not\vdash G \}.$$  

For instance, the formulas $S$ and $T$ of Example 3.7 have height 2, the formula $K$ of Example 3.8 has height 1. Note that $h(G) = 0$ iff $G$ is not classically valid. Let $D$ be an $FRJ(G)$-refutation of $G$. Since $Mod(D)$ is a countermodel for $G$ (see Theorem 3.12), we have $h(\text{Mod}(D)) \geq h(G)$. We show that we can build an $FRJ(G)$-refutation $\bar{D}$ of $G$ such that $h(\text{Mod}(\bar{D})) = h(G)$. To this aim, we prove that:
(K1) given a countermodel \( \mathcal{K} \) for \( G \), there exists an FRJ\((G)\)-refutation \( \tilde{D} \) of \( G \) such that 
\[ \mathbf{h}(\text{Mod}(\tilde{D})) \leq \mathbf{h}(\mathcal{K}). \]

By (K1), choosing as \( \mathcal{K} \) a countermodel for \( G \) having the minimal height \( \mathbf{h}(G) \), we get an FRJ\((G)\)-refutation \( \tilde{D} \) of \( G \) such that 
\[ \mathbf{h}(\text{Mod}(\tilde{D})) = \mathbf{h}(G). \]

Let \( D \) be an FRJ\((G)\)-refutation of \( G \). The height of the model \( \text{Mod}(D) \) is determined by the maximum number of applications of join rules along a branch of \( D \). To account for this, we introduce the notion of rank. Let \( \sigma \) be a sequent occurring in \( D \); the rank of \( \sigma \), denoted by \( \mathbf{Rn}(\sigma) \), is inductively defined as follows:

- If \( \sigma \) is an irregular axiom, then \( \mathbf{Rn}(\sigma) = -1 \).
- If \( \sigma \) is a regular axiom, then \( \mathbf{Rn}(\sigma) = 0 \).
- If \( \sigma \) is not an axiom, then let \( \frac{\sigma_1 \cdots \sigma_n}{\sigma} \mathcal{R} \) be the rule applied to get \( \sigma \) (\( n \geq 1 \)). Then:
  \[ \mathbf{Rn}(\sigma) = \max\{\mathbf{Rn}(\sigma_1), \ldots, \mathbf{Rn}(\sigma_n)\} + c \]
  where \( c = \begin{cases} 1 & \text{if } \mathcal{R} \in \{\mathbf{R}^{\Lambda^+}, \mathbf{R}^{\nu}\} \\ 0 & \text{otherwise} \end{cases} \).

The rank of \( D \), denoted by \( \mathbf{Rn}(D) \), is the rank of the root sequent of \( D \). One can easily prove that, for every regular sequent \( \sigma \) in \( D \), \( \mathbf{Rn}(\sigma) \) coincides with the height of the world \( \phi(\sigma) \) in \( \text{Mod}(\sigma) \) (where \( \phi \) is the map associated with \( D \) defined in Section 3.2). As an immediate consequence, we get:

**Lemma 6.2.** Let \( D \) be an FRJ\((G)\)-refutation of \( G \). Then, \( \mathbf{Rn}(D) = \mathbf{h}(\text{Mod}(D)) \).

Let \( \mathcal{K} = \langle P, \leq, \rho, V \rangle \) be a countermodel for \( G \) and \( \alpha \in P \). We set:

- \( \alpha \models^* H \) iff \( \alpha \models H \) and either \( H \in V \) or \( H = A \supset B \) and \( \alpha \not\models A \).
- \( \Lambda_\alpha = \{ A \in \text{St}(G) \mid \alpha \models A \} \).
- \( \Lambda_\alpha^* = \{ A \in \text{St}(G) \mid \alpha \models^* A \} \).
- \( \Omega_\alpha = \{ C \in \text{Sr}(G) \mid \alpha \not\models C \} \).

We point out that \( \alpha \leq \beta \) implies \( \Lambda_\alpha \subseteq \Lambda_\beta \) and \( \Omega_\beta \subseteq \Omega_\alpha \). To prove (K1), we exploit the following lemma (the proof is deferred to the end of this section):

**Lemma 6.3.** Let \( \mathcal{K} = \langle P, \leq, \rho, V \rangle \) be a countermodel for \( G \), let \( \alpha \in P \) and \( C \in \Omega_\alpha \).

- There exists an FRJ\((G)\)-refutation \( D_\alpha^- \) of \( \sigma_\alpha^-(C) = \Sigma ; \Theta \not\models C \) such that:
  - (i) \( \mathbf{Rn}(D_\alpha^-) < \alpha \); 
  - (ii) \( \Sigma \subseteq \Lambda_\alpha^* \subseteq \Sigma \cup \Theta \).

- There exists an FRJ\((G)\)-refutation \( D_\alpha^+ \) of \( \sigma_\alpha^+(C) = \Gamma \not\models C \) such that:
  - (iii) \( \mathbf{Rn}(D_\alpha^+) \leq \alpha \); 
  - (iv) there is \( \beta \in P \) such that \( \alpha \leq \beta \) and \( \Lambda_\beta^* \subseteq \Gamma \).

Point (K1) follows by the above lemma, taking as \( \tilde{D} \) the FRJ\((G)\)-refutation \( D_\rho^+ \) associated with the root \( \rho \) of \( \mathcal{K} \). Indeed, the root sequent of \( \tilde{D} \) is \( \sigma_\rho^+(G) = \Gamma \not\models G \), hence \( \tilde{D} \) is an FRJ\((G)\)-refutation of \( G \). By Theorem 3.12, \( \text{Mod}(\tilde{D}) \) is a countermodel for \( G \) and, by point (iii) of the lemma, \( \mathbf{Rn}(\tilde{D}) \leq \alpha(\rho) \), namely, \( \mathbf{Rn}(\tilde{D}) \leq \mathbf{Rn}(\mathcal{K}) \). By Lemma 6.2, we get \( \mathbf{h}(\text{Mod}(\tilde{D})) \leq \mathbf{h}(\mathcal{K}) \), which proves (K1). As discussed above, from (K1), we conclude:

**Theorem 6.4.** Let \( G \notin \text{IPL} \). Then, there exists an FRJ\((G)\)-refutation \( \tilde{D} \) of \( G \) such that 
\[ \mathbf{h}(\text{Mod}(\tilde{D})) = \mathbf{h}(G). \]
We remark that Theorem 6.4 provides an alternative proof of the completeness of FRJ(G). We have now to prove Lemma 6.3. We give a constructive proof to define the sequents $\sigma^\omega_\alpha(C)$ and $\sigma^{\oplus}_\alpha(C)$ and the related refutations, which relies on the following strategy:

- we visit the model $\mathcal{K}$ top-down, considering the worlds $\alpha$ of $\mathcal{K}$ in increasing order of height;
- for each world $\alpha$, we first define all the irregular sequents $\sigma^\omega_\alpha(C)$ and then all the regular sequents $\sigma^{\oplus}_\alpha(C)$;
- we pick the formulas $C$ of $\Omega_\alpha$ in increasing order of size.

Since $\rho$ is the bottom world of $\mathcal{K}$ and $G$ is the formula of maximum size in $\Omega_G$, the FRJ(G)-refutation $\mathcal{D}_\rho^G(G)$ is obtained at the end of the process. To highlight the main insights of the above construction, we present some examples.

**Example 6.5.** Let $S$ be the instance of Scott Principle in Example 3.7:

$$S = H \cup {\neg \neg p \lor \neg p} \quad H = (\neg \neg p \cup p) \cup \neg p \lor p,$$

$$S_L(S) = \{ H, \neg p \lor p, \neg \neg p, \neg p, H \},$$

$$S_R(S) = \{ S, \neg p \lor \neg p, \neg \neg p \lor p, \neg \neg p, \neg p, \neg \bot \}.$$  

We have $h(S) = 2$. Let us consider the countermodel $\mathcal{K}_S$ for $S$ having height 2 depicted below, consisting of the worlds $\alpha$ and $\beta$ of height 0, the world $\gamma$ of height 1 and the root $\rho$ of height 2:

We define the sequents $\sigma^\omega_\alpha(C)$ and $\sigma^{\oplus}_\alpha(C)$ matching Lemma 6.3. For each sequent $\sigma$, we display the rule applied to obtain $\sigma$ and the rank of $\sigma$. Using these annotations, one can immediately build the refutations $\mathcal{D}_\alpha^G(G)$ and $\mathcal{D}_\alpha^{\oplus}(G)$. We traverse the model $\mathcal{K}$ downwards, starting from the world $\alpha$ of height 0. We have:

$$\Lambda^*_\alpha = \{ \neg p \}, \quad \Lambda_\alpha = \Lambda^*_\alpha \cup \{ \neg p \lor p, H \}, \quad \Omega_\alpha = \{ \bot, \rho, \neg \neg p \}.$$  

Note that $H \notin \Lambda^*_\alpha$, since $\alpha \models \neg \neg p \supset p$. First, we define all the sequents $\sigma^\omega_\alpha(C)$, where $C \in \Omega_\alpha$; formulas $C$ are considered in increasing order of size.

| Sequent  | Applied rule | Rank |
|----------|--------------|------|
| $\sigma^\omega_\alpha(\bot)$ | \text{Ax}^\omega | -1   |
| $\sigma^\omega_\alpha(p)$     | \text{Ax}^\omega | -1   |
| $\sigma^\omega_\alpha(\neg p)$ | $\text{Ax}^{\mathcal{K}}$ | 0    |

Second, we define the sequents $\sigma^{\oplus}_\alpha(C)$, where $C \in \Omega_\alpha$:

$$\sigma^{\oplus}_\alpha(\bot) \quad \neg p \quad \neg \neg p \quad 0,$$

$$\sigma^{\oplus}_\alpha(p) \quad \neg p \quad \neg p \quad \neg p \quad \neg \neg p \quad 0.$$

Let us consider the world $\beta$ of height 0. We have:

$$\Lambda^*_\beta = \{ p, \neg \neg p \}, \quad \Lambda_\beta = \Lambda^*_\beta \cup \{ \neg p \lor p, H \}, \quad \Omega_\beta = \{ \bot, \neg p \}.$$  

\[\text{Actually, in Reference [16] the completeness of FRJ(G) has been proved along these lines.}\]
For \( C \in \Omega_\beta \), we define the sequents \( \sigma_\beta^\rightarrow (C) \):
\[
\sigma_\beta^\rightarrow (\bot) \quad \vdash p, H, \neg\neg p, \neg p \quad \dashv \bot \quad \text{Ax}_{\omega} \quad -1, \\
\sigma_\beta^\rightarrow (\neg p) \quad p ; H, \neg\neg p, \neg p \quad \vdash \neg p \quad \supset e \quad \sigma_\beta^\rightarrow (\bot) \quad -1.
\]
For \( C \in \Omega_\beta \), sequents \( \sigma_\beta^\leftarrow (C) \) are:
\[
\sigma_\beta^\leftarrow (\bot) \quad p, \neg\neg p \quad \dashv \bot \quad \text{At} \quad \sigma_\beta^\leftarrow (\neg p) \quad 0, \\
\sigma_\beta^\leftarrow (\neg p) \quad p, \neg\neg p \quad \vdash \neg p \quad \supset e \quad \sigma_\beta^\leftarrow (\bot) \quad 0.
\]
Let us consider \( \gamma \), the only world of height 1. We have
\[
\Lambda^*_\gamma = \Lambda_\gamma = \{ H, \neg\neg p \}, \quad \Omega_\gamma = \{ \bot, p, \neg p, \neg\neg p \supset p \}.
\]
For \( C \in \Omega_\gamma \), sequents \( \sigma_\gamma^\rightarrow (C) \) are:
\[
\sigma_\gamma^\rightarrow (\bot) \quad \vdash p, H, \neg\neg p, \neg p \quad \dashv \bot \quad \text{Ax}_{\omega} \quad -1, \\
\sigma_\gamma^\rightarrow (p) \quad \vdash p, \neg\neg p, \neg p \quad \vdash p \quad \text{Ax}_{\omega} \quad -1, \\
\sigma_\gamma^\rightarrow (\neg p) \quad \vdash p, H, \neg\neg p \quad \vdash \neg p \quad \supset e \quad \sigma_\beta^\leftarrow (\bot) \quad 0, \\
\sigma_\gamma^\rightarrow (\neg\neg p \supset p) \quad \neg\neg p ; H, \neg p \quad \vdash \neg p \supset p \quad \supset e \quad \sigma_\gamma^\rightarrow (p) \quad -1.
\]
Finally, let us consider the root \( \rho \) having height 2. We have:
\[
\Lambda^*_\rho = \Lambda_\rho = \{ H \}, \quad \Omega_\rho = \text{Sr}(S).
\]
We can inherit the following definitions from the worlds \( \alpha, \beta, \gamma \):
\[
\sigma_\alpha^\rightarrow (\bot) = \sigma_\alpha^\rightarrow (\bot), \quad \sigma_\alpha^\rightarrow (p) = \sigma_\alpha^\rightarrow (p), \quad \sigma_\alpha^\leftarrow (\neg p) = \sigma_\alpha^\leftarrow (\neg p), \\
\sigma_\beta^\rightarrow (\bot) = \sigma_\beta^\rightarrow (\bot), \quad \sigma_\beta^\rightarrow (p) = \sigma_\alpha^\rightarrow (p), \quad \sigma_\beta^\leftarrow (\neg p) = \sigma_\beta^\leftarrow (\neg p), \\
\sigma_\beta^\rightarrow (\neg\neg p \supset p) = \sigma_\beta^\leftarrow (\neg p \supset p) = \sigma_\gamma^\rightarrow (\neg p \supset p).
\]
For \( C \in \{ \neg p, \neg p \supset p, \neg p \lor \neg p, S \} \), the sequents \( \sigma_\rho^\rightarrow (C) \) are:
\[
\sigma_\rho^\rightarrow (\neg p) \quad \vdash H \quad \vdash \neg p \quad \supset e \quad \sigma_\alpha^\rightarrow (\bot) \quad 0, \\
\sigma_\rho^\rightarrow (\neg p \supset p) \quad \vdash H \quad \vdash \neg p \supset p \quad \supset e \quad \sigma_\gamma^\leftarrow (p) \quad 1, \\
\sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad \vdash H \quad \vdash \neg p \lor \neg p \quad \forall \quad \sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad 0, \\
\sigma_\rho^\leftarrow (S) \quad H \quad \vdash S \quad \supset e \quad \sigma_\gamma^\rightarrow (\neg p \lor \neg p) \quad 0.
\]
For \( C \in \{ \neg p \lor \neg p, S \} \), the sequents \( \sigma_\rho^\leftarrow (C) \) are:
\[
\sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad H \quad \vdash \neg p \lor \neg p \quad \forall \quad \sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad \sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad \sigma_\rho^\leftarrow (\neg p) \quad 2, \\
\sigma_\rho^\leftarrow (S) \quad H \quad \vdash S \quad \supset e \quad \sigma_\rho^\leftarrow (\neg p \lor \neg p) \quad 2.
\]
The \( \text{FRJ}(S) \)-refutation \( D_\rho^\rightarrow (S) \) of \( \sigma_\rho^\leftarrow (S) \) obtained in the end is an \( \text{FRJ}(S) \)-refutation of the goal \( S \). Note that \( D_\rho^\rightarrow (S) \) essentially coincides with the refutation in Figure 2, hence the model \( \text{Mod}(D_\rho^\rightarrow (S)) \) extracted from \( D_\rho^\rightarrow (S) \) is isomorphic to the countermodel \( K_S \) displayed at the beginning of this example.
In the previous example, the model $K_S$ initially chosen is a minimal countermodel for $S$, since there exists no countermodel for $S$ having less than 4 worlds, and the obtained model $\text{Mod}(D^\varphi_\rho(S))$ is a minimal countermodel for $S$ as well. One may wonder if this is always the case. The answer is negative, as shown in the next example; we also point out that Lemma 6.3 only sets a bound on the height of $\text{Mod}(D^\varphi_\rho(G))$ and not on the number of worlds.

**Example 6.6.** Let $C$ be the formula

$$C = A \lor B, \quad A = (p_1 \supset p_2) \lor (p_2 \supset p_1), \quad B = (q_1 \supset q_2) \lor (q_2 \supset q_1).$$

We have $h(C) = 1$. Let $K_C$ be the following countermodel for $C$ of height 1, consisting of the worlds $\alpha$ and $\beta$ of height 0 and of the root $\rho$ of height 1:

```
\alpha: p_1, q_1
\beta: p_2, q_2
\rho: p_1, p_2
```

One can easily check that there is no countermodel for $C$ having less than 3 worlds, hence $K_C$ is a minimal countermodel for $C$. We have

$$\text{Sl}(C) = \{p_1, p_2, q_1, q_2\},$$

$$\text{Sr}(C) = \{C, A, B, p_1 \supset p_2, p_2 \supset p_1, q_1 \supset q_2, q_2 \supset q_1, p_1, p_2, q_1, q_2\},$$

$$\Lambda^*_\alpha = \Lambda_\alpha = \{p_1, q_1\}, \quad \Omega_\alpha = \{p_2, q_2, p_1 \supset p_2, q_1 \supset q_2\},$$

$$\Lambda^*_\beta = \Lambda_\beta = \{p_2, q_2\}, \quad \Omega_\beta = \{p_1, q_1, p_2 \supset p_1, q_2 \supset q_1\},$$

$$\Lambda^*_\rho = \Lambda_\rho = \emptyset \quad \Omega_\rho = \text{Sr}(C).$$

We define the sequents $\sigma^\varphi_\delta(C)$ and $\sigma^\Box_\delta(C)$, where $\delta \in \{\alpha, \beta, \rho\}$ and $C \in \Omega_\delta$, only showing the sequents needed to prove the goal.

| $\sigma^\varphi_\alpha(p_2)$ | $p_1, q_1, q_2 \models p_2$ | $\text{Ax}^\varphi_\alpha$ |
| $\sigma^\varphi_\beta(q_2)$ | $p_1, p_2, q_1 \models q_2$ | $\text{Ax}^\varphi_\beta$ |
| $\sigma^\varphi_\rho(p_1)$ | $p_1, p_2, q_2 \models q_1$ | $\text{Ax}^\varphi_\rho$ |
| $\sigma^\rho_\rho(p_1 \supset p_2)$ | $\vdots; q_1, q_2 \models p_1 \supset p_2$ | $\supset_\delta \sigma^\varphi_\delta(p_2)$ |
| $\sigma^\rho_\rho(p_2 \supset p_1)$ | $\vdots; q_1, q_2 \models p_2 \supset p_1$ | $\supset_\delta \sigma^\varphi_\delta(p_1)$ |
| $\sigma^\rho_\rho(q_1 \supset q_2)$ | $\vdots; p_1, p_2 \models q_1 \supset q_2$ | $\supset_\delta \sigma^\varphi_\delta(q_2)$ |
| $\sigma^\rho_\rho(q_2 \supset q_1)$ | $\vdots; p_1, p_2 \models q_2 \supset q_1$ | $\supset_\delta \sigma^\varphi_\delta(q_1)$ |
| $\sigma^\rho_\rho(A)$ | $\vdots; q_1, q_2 \models A$ | $\lor \sigma^\rho_\rho(p_1 \supset p_2) \sigma^\rho_\rho(p_2 \supset p_1)$ |
| $\sigma^\rho_\rho(B)$ | $\vdots; p_1, p_2 \models B$ | $\lor \sigma^\rho_\rho(q_1 \supset q_2) \sigma^\rho_\rho(q_2 \supset q_1)$ |

The goal $C$ is proved by the FRJ($C$)-refutation $D^\varphi_\rho(C)$ of $\sigma^\varphi_\rho(C)$. The model $\text{Mod}(D^\varphi_\rho(C))$ extracted from $D^\varphi_\rho(C)$ is:

```
$\sigma^\varphi_\alpha(p_2): p_1, q_1, q_2$
$\sigma^\varphi_\beta(q_2): p_1, p_2, q_1$
$\sigma^\varphi_\rho(p_1): p_2, q_1, q_2$
$\sigma^\varphi_\rho(q_1): p_1, p_2, q_2$
$\sigma^\varphi_\rho(C)$
```

Such a model has the minimal height 1, as expected, but it is not minimal. To obtain minimal models, we should redesign FRJ($G$) in a multi-succedent style, so that axioms of the kind
\( p_1, q_1 \not\iff p_2, q_2 \) and \( p_2, q_2 \not\iff p_1, q_1 \) are allowed. Thus, we should have regular axioms of the kind \( \Gamma^A \not\iff \Delta^A \), where \( \Gamma^A \) and \( \Delta^A \) are disjoint sets of propositional variables such that \( \Gamma^A \subseteq \text{Lhs}(G) \) and \( \Delta^A \subseteq \text{Rhs}(G) \). However, this would cause an exponential blow-up of the number of generated sequents and refutation-search would become unfeasible. For example, in this case, we would get 64 regular axioms instead of 5. To sum up, the definition of a calculus devoted to the construction of minimal models in the number of worlds seems to be challenging; a different approach to generate minimal models, exploiting Answer Set Programming, is presented in Reference [15].

We remark that we can modify the refutation-search procedure of Figure 8 so to generate \( \text{FRJ}(G) \)-refutations of minimal height; this is obtained by delaying the application of join rules as much as possible, mimicking the strategy applied in the above examples.

The rest of this section is devoted to the detailed proof of Lemma 6.3. First, we prove some closure properties of sets \( \Lambda_\alpha \) and \( \Lambda^*_\alpha \):

**Lemma 6.7.** Let \( \mathcal{K} \) be a countermodel for \( G \) and \( \alpha \) a world of \( \mathcal{K} \). Then, \( \Lambda_\alpha = \text{Cl}(\Lambda_\alpha) = \text{Cl}(\Lambda^*_\alpha) \).

**Proof.** To prove the assertion, we show that:

(i) \( \Lambda_\alpha \subseteq \text{Cl}(\Lambda_\alpha) \).
(ii) \( \text{Cl}(\Lambda_\alpha) \subseteq \Lambda_\alpha \).
(iii) \( \text{Cl}(\Lambda^*_\alpha) \subseteq \text{Cl}(\Lambda_\alpha) \).
(iv) \( \Lambda_\alpha \subseteq \text{Cl}(\Lambda^*_\alpha) \).
(v) \( \text{Cl}(\Lambda_\alpha) \subseteq \text{Cl}(\Lambda^*_\alpha) \).

Point (i) immediately follows by (CL3). To prove (ii), let \( C \in \text{Cl}(\Lambda_\alpha) \); by induction on \( |C| \), we show that \( C \in \Lambda_\alpha \). The case \( C \in \mathcal{V}^\bot \) immediately follows by (CL5). Let \( C = A \land B \). Then, \( A \land B \in \Lambda_\alpha \) or \( \{A, B\} \subseteq \text{Cl}(\Lambda_\alpha) \). In the former case, we are done. In the latter case, by the induction hypothesis, we have \( \{A, B\} \subseteq \Lambda_\alpha \). Thus, \( \alpha \models A \) and \( \alpha \models B \), which implies \( \alpha \models A \land B \), hence \( A \land B \in \Lambda_\alpha \). The cases \( C = A \lor B \) and \( C = A \supset B \) are similar.

Point (iii) follows by the fact that \( \Lambda^*_\alpha \subseteq \Lambda_\alpha \) and by (CL4).

To prove (iv), let \( C \in \Lambda_\alpha \), namely, \( \alpha \models C \); by induction on \( |C| \), we show that \( C \in \text{Cl}(\Lambda^*_\alpha) \). If \( C \in \mathcal{V} \), then \( \alpha \models^* C \), hence \( C \in \Lambda^*_\alpha \), which implies \( C \in \text{Cl}(\Lambda^*_\alpha) \). Let \( C = A \lor B \). If \( \alpha \not\models A \), then \( \alpha \models^* C \) and, as above, \( C \in \text{Cl}(\Lambda^*_\alpha) \). If \( \alpha \models A \), then \( \alpha \models B \); by induction hypothesis, \( B \in \text{Cl}(\Lambda^*_\alpha) \), hence \( A \lor B \in \text{Cl}(\Lambda^*_\alpha) \). The cases \( C = A \lor B \) and \( C = A \supset B \) easily follow by the induction hypothesis.

Point (v) immediately follows by (iv) and by (CL6).

To sum up, by (i) and (ii), we get \( \Lambda_\alpha = \text{Cl}(\Lambda_\alpha) \); by (iii) and (v), \( \text{Cl}(\Lambda_\alpha) = \text{Cl}(\Lambda^*_\alpha) \).

We prove Lemma 6.3.

**Proof of Lemma 6.3.** We define the refutation \( \mathcal{D}_a^\ominus(C) \), where \( \ominus \in \{\not\iff, \not\rightarrow\} \), using the following mutual induction hypothesis (mirroring the order used to define the sequents \( \sigma_\alpha^\not\iff(C) \) and \( \sigma_\alpha^\not\rightarrow(C) \) in Example 6.5 and 6.6):

(\( \text{IH}1 \)) a main induction on \( h(\alpha) \);
(\( \text{IH}2 \)) a secondary induction on \( \text{tp}^{-}(\ominus) \), where \( \text{tp}^{-}(\not\iff) = 0 \) and \( \text{tp}^{-}(\not\rightarrow) = 1 \);
(\( \text{IH}3 \)) a third induction on \( |C| \).

Let us introduce the following notations:

\[
\Gamma^A = \text{SL}(G) \cap \mathcal{V}, \quad \Gamma^\ominus = \text{SL}(G) \cap \mathcal{L}^\ominus, \quad \Gamma = \Gamma^A \cup \Gamma^\ominus, \\
\Lambda^*_\alpha^A = \Lambda^*_\alpha \cap \mathcal{V}, \quad \Lambda^*_\alpha^\ominus = \Lambda^*_\alpha \cap \mathcal{L}^\ominus.
\]

We proceed by a case analysis on \( C \) and \( \mathcal{D}_a^\ominus(C) \). We point out that, since \( \alpha \not\models C \), then \( C \not\in \Lambda_\alpha \) and \( C \not\in \Lambda^*_\alpha \).
Case $C \in V^\perp$, definition of $D^\perp_\alpha (C)$.

We set:

$$D^\perp_\alpha (C) = \sigma^\perp_\alpha (C) = \vdots; \Gamma^\perp \setminus \{ C \}, \Gamma^\perp \not\vdash C \text{ Ax}^\perp_\alpha .$$

Since $\text{Rn}(D^\perp_\alpha (C)) = -1$ and $h(\alpha) \geq 0$, we have $\text{Rn}(D^\perp_\alpha (C)) < h(\alpha)$, hence (i) holds. Since $C \notin \Lambda^*_\alpha$, Point (ii) immediately follows.

Case $C \in V^\perp$, definition of $D^\propto_\alpha (C)$.

If $\Lambda^*_\alpha$ is empty, namely, $\Lambda^*_\alpha = \Lambda^*_{\alpha M}$, then we set

$$D^\propto_\alpha (C) = \sigma^\propto_\alpha (C) = \Gamma^\propto \not\vdash C \text{ Ax}^\propto_\alpha .$$

Since $\text{Rn}(D^\propto_\alpha (C)) = 0$, we have $\text{Rn}(D^\propto_\alpha (C)) \leq h(\alpha)$, which proves (iii). Point (iv) holds for $\beta = \alpha$, since $C \notin \Lambda^*_\alpha$. If $\Lambda^*_\alpha$ is nonempty, then let

$$\Upsilon = \{ Y \mid Y \supset Z \in \Lambda^*_\alpha \} = \{ A_1, \ldots, A_n \} \quad (n \geq 1).$$

By definition of $\Lambda^*_\alpha$, we have $\alpha \not\vdash A_j$, for every $A_j \in \Upsilon$. By (IH2), for every $1 \leq j \leq n$, there is a FRJ(G)-refutation $D^\alpha_\alpha (A_j)$ of $\sigma^\alpha_\alpha (A_j) = \Sigma_j; \Theta_j \not\vdash A_j$ such that:

(P1) $\text{Rn}(D^\alpha_\alpha (A_j)) < h(\alpha)$;
(P2) $\Sigma_j \subseteq \Lambda^*_\alpha \subseteq \Sigma_j \cup \Theta_j$.

We stress that the use of (IH2) is sound, since $\text{tp}^- (\not\vdash) < \text{tp}^- (\not\vdash)$. Let $\Sigma_j = \Sigma_j^\perp \cup \Sigma_j^\propto$ and $\Theta_j = \Theta_j^\perp \cup \Theta_j^\propto$. We prove that $\sigma^\alpha_\alpha (A_1), \ldots, \sigma^\alpha_\alpha (A_n)$ satisfy the following properties, for every $1 \leq j \leq n$:

(a) $\Sigma_i \subseteq \Sigma_j \cup \Theta_j$, for every $i \neq j$;
(b) $Y \supset Z \in \Sigma_j^\propto$ implies $Y \in \Upsilon$;
(c) $C \notin \Sigma_j^\propto$.

Point (a) follows by (P2). Point (b) follows by (P2) and the definition of $\Upsilon$. Point (c) follows by the fact that $C \notin \Lambda^*_\alpha$ and (P2). By (a)–(c), we can apply the rule $\not\vdash_{\alpha M}$ with premises $\sigma^\alpha_\alpha (A_1), \ldots, \sigma^\alpha_\alpha (A_n)$ and build the FRJ(G)-refutation $D^\propto_\alpha (C)$ displayed below ($\Sigma^\alpha, \Sigma^\propto, \Theta^\alpha, \Theta^\propto$ are defined as in Figure 1):

$$D^\propto_\alpha (C) = \ldots; \Sigma_j^\perp, \Sigma_j^\propto; \Theta_j^\alpha, \Theta_j^\propto \not\vdash A_j \ldots, j = 1, \ldots, n,$$

$$\Gamma = \Sigma^\alpha \cup (\Theta^\alpha \setminus \{ C \}) \cup \Sigma^\propto \cup \Theta^\propto.$$

Note that, by the definition of $\Upsilon$, the application of $\not\vdash_{\alpha M}$ satisfies the restriction (RS3) stated in Section 3. By definition, $\text{Rn}(D^\propto_\alpha (C)) = m + 1$, where $m$ is the maximum among $\text{Rn}(D^\alpha_\alpha (A_1)), \ldots, \text{Rn}(D^\alpha_\alpha (A_n))$. By (P1), $m < h(\alpha)$, hence $\text{Rn}(D^\propto_\alpha (C)) \leq h(\alpha)$, and this proves (iii). We show that $\Lambda^*_\alpha \subseteq \Gamma$, and this proves (iv) (where $\beta = \alpha$). If, for some $j \in \{ 1, \ldots, n \}$, $\Lambda^*_\alpha \subseteq \Sigma_j$, then $\Lambda^*_\alpha \subseteq \Sigma^\alpha \cup \Sigma^\propto$, hence $\Lambda^*_\alpha \subseteq \Gamma$. Otherwise, by (P2), $\Lambda^*_\alpha \subseteq \cap_{1 \leq j \leq n} \Theta_j$, which implies:

(d) $\Lambda^*_{\alpha M} \subseteq \cap_{1 \leq j \leq n} \Theta_j^\alpha$;
(e) $\Lambda^*_{\alpha \propto} \subseteq \cap_{1 \leq j \leq n} \Theta_j^\propto$.

Since $C \notin \Lambda^*_{\alpha M}$ and $\cap_{1 \leq j \leq n} \Theta_j^\alpha = \Theta^\alpha$, by (d), we get $\Lambda^*_{\alpha M} \subseteq \Theta^\alpha \setminus \{ C \}$; hence, $\Lambda^*_{\alpha M} \subseteq \Gamma$. Moreover, by (e), we get $\Lambda^*_{\alpha \propto} \subseteq \Theta^\propto$, hence $\Lambda^*_{\alpha \propto} \subseteq \Gamma$. We conclude that $\Lambda^*_\alpha = \Lambda^*_{\alpha M} \cup \Lambda^*_{\alpha \propto} \subseteq \Gamma$.

Duality between Unprovability and Provability in Forward Refutation-search for IPL 22:41

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.
• Case $C = C_1 \lor C_2$, definition of $D_{a^{\neg \neg}}(C)$.

Since $\alpha \nvdash C_1 \lor C_1$, we have $\alpha \nvdash C_1$ and $\alpha \nvdash C_2$. By (IH3), for every $k \in \{1, 2\}$ there is an FRJ(G)-refutation $D_{a^{\neg \neg}}(C_k) = \Sigma_k ; \Theta_k \rightarrow C_k$ such that

\begin{align*}
(\text{Q1}) & \quad \text{Rn}(D_{a^{\neg \neg}}(C_k)) < h(\alpha); \\
(\text{Q2}) & \quad \Sigma_k \subseteq \Lambda^\alpha \subseteq \Sigma_k \cup \Theta_k.
\end{align*}

By (Q2), we get $\Sigma_1 \subseteq \Sigma_2 \cup \Theta_2$ and $\Sigma_2 \subseteq \Sigma_1 \cup \Theta_1$; hence, we can set

$$
D_{a^{\neg \neg}}(C) = \begin{pmatrix} D_{a^{\neg \neg}}(C_1) & D_{a^{\neg \neg}}(C_2) \\
\Sigma_1 ; \Theta_1 \rightarrow C_1 & \Sigma_2 ; \Theta_2 \rightarrow C_2 \end{pmatrix} \supseteq \Sigma \Theta \cup \Theta \subseteq \Sigma \cup \Theta.
$$

Since Rn($D_{a^{\neg \neg}}(C)$) is the maximum between Rn($D_{a^{\neg \neg}}(C_1)$) and Rn($D_{a^{\neg \neg}}(C_2)$), by (Q1), we get Rn($D_{a^{\neg \neg}}(C)$) < h(\alpha), and this proves (i). By (Q2), we immediately get $\Sigma \subseteq \Lambda^\alpha$. Moreover, by (Q2) it follows that $\Lambda^\alpha \subseteq \Sigma_1$ or $\Lambda^\alpha \subseteq \Sigma_2$ or $\Lambda^\alpha \subseteq \Theta_1 \cap \Theta_2$, which implies $\Lambda^\alpha \subseteq \Sigma \cup \Theta$. Thus, (ii) holds.

• Case $C = C_1 \lor C_2$, definition of $D_{a^{\neg \neg}}(C)$.

Since $\alpha \nvdash C_1 \lor C_1$, we have $\alpha \nvdash C_1$ and $\alpha \nvdash C_2$. By (IH2), for every $k \in \{1, 2\}$ there is an FRJ(G)-refutation $D_{a^{\neg \neg}}(C_k) = \Sigma_k ; \Theta_k \rightarrow C_k$ satisfying (Q1) and (Q2). Let $\Sigma_k = \Sigma_k^\alpha \cup \Sigma_k^\beta$ and $\Theta_k = \Theta_k^\alpha \cup \Theta_k^\beta$ and let

$$
Y = \{Y \mid Y \supseteq Z \in \Lambda^\alpha \} \cup \{C_1, C_2\} = \{A_1, \ldots, A_n\} \quad (n \geq 1).
$$

We argue as in the case concerning $D_{a^{\neg \neg}}(C)$ with $C \in \mathcal{V}^\bot$. For every $1 \leq j \leq n$, since $\alpha \nvdash A_j$, by (IH2) there is an FRJ(G)-refutation $D_{a^{\neg \neg}}(A_j)$ of $\sigma_{a^{\neg \neg}}(A_j) = \Sigma_j ; \Theta_j \rightarrow A_j$ such that:

• if $A_j \in \{C_1, C_2\}$, points (Q1) and (Q2) hold;
• otherwise, points (P1) and (P2) hold.

Hence, we can build the FRJ(G)-refutation $D_{a^{\neg \neg}}(C)$ as follows ($\Sigma^\alpha, \Sigma^\beta, \Theta^\alpha, \Theta^\beta$ are defined as in Figure 1):

$$
D_{a^{\neg \neg}}(C) = \begin{pmatrix} \cdots \Sigma^\alpha_j, \Sigma^\beta_j, \Theta^\alpha_j, \Theta^\beta_j \rightarrow A_j \cdots \\
\sigma_{a^{\neg \neg}}(C) = \Gamma \Rightarrow C_1 \lor C_2 \end{pmatrix} \supseteq \mathcal{V}^\triangleright,
$$

We point out that the displayed application of $\supseteq \mathcal{V}^\triangleright$ matches the restriction (RS4) stated in Section 3. Reasoning as above, Point (iii) follows by (P1) and (Q1), Point (iv) (with $\beta = \alpha$) by (P2) and (Q2),

• Case $C = C_1 \land C_2$, definition of $D_{a^{\neg \neg}}(C)$.

Since $\alpha \nvdash C_1 \land C_2$, there exists $k \in \{1, 2\}$ such that $\alpha \nvdash C_k$. By (IH3), there exists an FRJ(G)-refutation $D_{a^{\neg \neg}}(C_k) = \Sigma ; \Theta \rightarrow C_k$ such that

\begin{align*}
(\text{R1}) & \quad \text{Rn}(D_{a^{\neg \neg}}(C_k)) < h(\alpha); \\
(\text{R2}) & \quad \Sigma \subseteq \Lambda^\alpha \subseteq \Sigma \cup \Theta.
\end{align*}
We can build the \( \text{FRJ}(G) \)-refutation:

\[
\mathcal{D}_\alpha^{\rightarrow}(C) = \frac{\mathcal{D}_\alpha^{\rightarrow}(C_k) \quad \Sigma; \Theta \nleftrightarrow C_k}{\sigma_\alpha^{\rightarrow}(C) = \Sigma; \Theta \nleftrightarrow C_1 \land C_2} \land.
\]

Since \( \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(C)) = \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(C_k)) \), by (R1), we get \( \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(C)) < h(\alpha) \), and this proves (i). Point (ii) immediately follows by (R2).

- Case \( C = C_1 \land C_2 \), definition of \( \mathcal{D}_\alpha^{\leftrightarrow}(C) \).

Since \( \alpha \nvdash C_1 \land C_2 \), there exists \( k \in \{1, 2\} \) such that \( \alpha \nvdash C_k \). By (IH2), there exists an \( \text{FRJ}(G) \)-refutation \( \mathcal{D}_\alpha^{\leftrightarrow}(C_k) \) of \( \sigma_\alpha^{\leftrightarrow}(C_k) = \Gamma \nleftrightarrow C_k \) such that:

- (R3) \( \text{Rn}(\mathcal{D}_\alpha^{\leftrightarrow}(C_k)) \leq h(\alpha) \);
- (R4) There is \( \beta \in P \) such that \( \alpha \leq \beta \) and \( \Lambda_\beta^* \subseteq \Gamma \).

We can build the \( \text{FRJ}(G) \)-refutation:

\[
\mathcal{D}_\alpha^{\leftrightarrow}(C) = \frac{\mathcal{D}_\alpha^{\leftrightarrow}(C_k) \quad \Gamma \nleftrightarrow C_k}{\sigma_\alpha^{\leftrightarrow}(C) = \Gamma \nleftrightarrow C_1 \land C_2} \land.
\]

Since \( \text{Rn}(\mathcal{D}_\alpha^{\leftrightarrow}(C)) = \text{Rn}(\mathcal{D}_\alpha^{\leftrightarrow}(C_k)) \), by (R3), we get \( \text{Rn}(\mathcal{D}_\alpha^{\leftrightarrow}(C)) \leq h(\alpha) \), and this proves (iii). Point (iv) immediately follows by (R4).

- Case \( C = A \supset B \), definition of \( \mathcal{D}_\alpha^{\rightarrow}(C) \).

Since \( \alpha \nvdash A \supset B \), there is \( \eta \in P \) such that \( \alpha \leq \eta \) and \( \eta \vdash A \) and \( \eta \nvdash B \). Without loss of generality, we assume that, for every \( \delta \in P \) such that \( \alpha \leq \delta < \eta \), we have \( \delta \nvdash A \). Since \( \alpha \leq \eta \), it holds that \( \alpha \nvdash B \). By (IH3), there exists an \( \text{FRJ}(G) \)-refutation \( \mathcal{D}_\alpha^{\rightarrow}(B) \) of \( \sigma_\alpha^{\rightarrow}(B) = \Sigma_1; \Theta_1 \nleftrightarrow B \) such that:

- (S1) \( \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(B)) < h(\alpha) \);
- (S2) \( \Sigma_1 \subseteq \Lambda_\alpha^* \subseteq \Sigma_1 \cup \Theta_1 \).

If \( \eta = \alpha \), then \( \alpha \vdash A \), hence \( A \in \Lambda_\alpha \), which implies, by Lemma 6.7, \( A \in \text{Cl}(\Lambda_\alpha^*) \). Let \( \Lambda \) be a (possibly empty) minimum subset of \( \Lambda_\alpha^* \setminus \Sigma_1 \) such that \( A \in \text{Cl}(\Sigma_1 \cup \Lambda) \) (namely: \( \Lambda' \subseteq \Lambda \) implies \( A \not\in \text{Cl}(\Sigma_1 \cup \Lambda') \)); note that \( \Lambda \subseteq \Theta_1 \). We can build the \( \text{FRJ}(G) \)-refutation \( \mathcal{D}_\alpha^{\rightarrow}(C) \) as follows, where rule \( \supsete \) shifts the set \( \Lambda \) to the left of semicolon:

\[
\mathcal{D}_\alpha^{\rightarrow}(C) = \frac{\mathcal{D}_\alpha^{\rightarrow}(B) \quad \Theta_2 = \Theta_1 \setminus \Lambda, \quad A \in \text{Cl}(\Sigma_1 \cup \Lambda),}{\sigma_\alpha^{\rightarrow}(C) = \Sigma_1, \Lambda \supseteq \Sigma_1 \cup \Theta_2, \Lambda \nleftrightarrow B \quad \supsete \quad \Lambda' \subseteq \Lambda \implies A \not\in \text{Cl}(\Sigma_1 \cup \Lambda')}.
\]

We point out that the choice of \( \Lambda \) complies with the restriction (RS1) stated in Section 3. Since \( \Sigma_1 \subseteq \Lambda_\alpha^* \) (see (S2)) and \( \Lambda \subseteq \Lambda_\alpha^* \), we get \( \Sigma_1 \cup \Lambda \subseteq \Lambda_\alpha^* \), namely, \( \Sigma \subseteq \Lambda_\alpha^* \). Moreover, since \( \Lambda_\alpha^* \subseteq \Sigma_1 \cup \Theta_1 \) (see (S2)) and \( \Sigma_1 \cup \Theta_1 = \Sigma \cup \Theta_2 \), we get \( \Lambda_\alpha^* \subseteq \Sigma \cup \Theta_2 \), and this concludes the proof of (ii). Since \( \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(C)) = \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(B)) \), by (S1) we get \( \text{Rn}(\mathcal{D}_\alpha^{\rightarrow}(C)) < h(\alpha) \), which proves (i).

If \( \alpha < \eta \), then \( h(\eta) < h(\alpha) \). By the choice of \( \eta \), we have \( \alpha \nvdash A \). Since \( \eta \nvdash B \), by (IH1) there is an \( \text{FRJ}(G) \)-refutation \( \mathcal{D}_\eta^{\rightarrow}(B) \) of \( \sigma_\eta^{\rightarrow}(B) = \Gamma \nleftrightarrow B \) such that:

- (S3) \( \text{Rn}(\mathcal{D}_\eta^{\rightarrow}(B)) \leq h(\eta) \);
- (S4) There exists \( \beta \in P \) such that \( \eta \leq \beta \) and \( \Lambda_\beta^* \subseteq \Gamma \).
Since $\eta \models A$ and $\eta \leq \beta$, we get $\beta \models A$, hence $A \in \Lambda_\beta$. By Lemma 6.7, $A \in Cl(\Lambda_\beta^*)$ hence, by (S4) and (CL4), $A \in Cl(\Gamma)$. Since $\alpha \not\models A$, we have $A \notin \Lambda_\alpha$ hence, by Lemma 6.7, $A \notin Cl(\Lambda_\alpha^*)$. Since $\alpha < \eta \leq \beta$, we have $\Lambda_\alpha^* \subseteq \Lambda_\beta$. By Lemma 6.7, we get $\Lambda_\beta = Cl(\Lambda_\beta^*)$, thus $\Lambda_\alpha^* \subseteq Cl(\Lambda_\beta^*)$. By (S4) and (CL4), $Cl(\Lambda_\beta^*) \subseteq Cl(\Gamma)$, hence $\Lambda_\alpha^* \subseteq Cl(\Gamma)$. To sum up:

- $\Lambda_\alpha^* \subseteq Cl(\Gamma) \cap \Gamma$ and $A \in Cl(\Gamma) \setminus Cl(\Lambda_\alpha^*)$.

Let $\Theta$ be a maximum extension of $\Lambda_\alpha^*$ such that $\Lambda_\alpha^* \subseteq \Theta \subseteq Cl(\Gamma) \cap \Gamma$ and $A \notin Cl(\Theta)$ (namely: $\Theta \subseteq \Theta' \subseteq Cl(\Gamma) \cap \Gamma$ implies $A \in Cl(\Theta')$). We can build the $FRJ(\Gamma)$-refutation:

$$D_{\alpha}^{\lambda}(C) = \frac{D_{\eta}^{\lambda}(B)}{\Gamma \not\models B} \quad \alpha \models \Theta \subseteq Cl(\Gamma) \cap \Gamma, \quad A \in Cl(\Gamma) \setminus Cl(\Theta),$$

We point out that the choice of $\Theta$ matches the restriction (RS2) stated in Section 3. We have $Rn(D_{\alpha}^{\lambda}(C)) = Rn(D_{\eta}^{\lambda}(B))$; by (S3), we get $Rn(D_{\alpha}^{\lambda}(C)) \leq h(\eta)$, hence $Rn(D_{\alpha}^{\lambda}(C)) < h(\alpha)$, which proves (i). Since $\Lambda_\alpha^* \subseteq \Theta$, (ii) holds.

- Case $C = A \supset B$, definition of $D_{\alpha}^{\lambda}(C)$.

Since $\alpha \not\models A \supset B$, there is $\eta \in P$ such that $\alpha \leq \eta$ and $\eta \models A$ and $\eta \not\models B$. Since $\eta \not\models B$, by induction hypothesis (IH1) if $\alpha < \eta$ and (IH3) if $\alpha = \eta$, there is an $FRJ(\Gamma)$-refutation $D_{\eta}^{\lambda}(B)$ of $\sigma_{\eta}^{\lambda}(B) = \Gamma \not\models B$ satisfying points (S3) and (S4); note that $\alpha \leq \eta \leq \beta$. Since $\eta \leq \beta$, we have $\beta \models A$, namely, $A \in \Lambda_\beta$. By Lemma 6.7, $A \in Cl(\Lambda_\beta^*)$, which implies, by (S4) and (CL4), $A \in Cl(\Gamma)$. Thus, we can build the $FRJ(\Gamma)$-refutation:

$$D_{\alpha}^{\lambda}(C) = \frac{D_{\eta}^{\lambda}(B)}{\Gamma \not\models B} \quad \alpha \models \Gamma \not\models A \supset B \quad \subseteq \quad A \in Cl(\Gamma).$$

Since $Rn(D_{\alpha}^{\lambda}(C)) = Rn(D_{\eta}^{\lambda}(B))$ and $h(\eta) \leq h(\alpha)$ (indeed, $\alpha \leq \eta$, by (S3), we get $Rn(D_{\alpha}^{\lambda}(C)) \leq h(\alpha)$, which proves (iii). Point (iv) immediately follows by (S4), being $\alpha \leq \beta$. \hfill $\square$

7 RELATED AND FUTURE WORK

We have introduced a forward calculus $FRJ(\Gamma)$ to derive the non-validity of a goal formula $G$ in IPL. From an $FRJ(\Gamma)$-refutation of $G$, we can extract a countermodel for $G$. Otherwise, we eventually get a saturated database $D_G$, which can be exploited to build a derivation of $G$ in the sequent calculus G3i; accordingly, $D_G$ can be understood as a “proof-certificate” of the validity of $G$ in IPL (a dual remark for a forward calculus for IPL has been issued in Reference [24]).

As discussed in Section 3, whenever we search for an $FRJ(\Gamma)$-refutation of $G$, we are also trying to build a countermodel for $G$ in a backward style, starting from the final worlds down to the root. Thus, our countermodel construction technique is dual to standard proof-search procedures such as References [1, 6, 9, 10, 12, 17, 18, 25, 26], where proofs and model are searched bottom-up, starting from the goal and backward applying the rules of the calculus. One of the advantages of forward versus backward reasoning is that, provided one implements suitable redundancy checks, refutations are more concise, since sequents are reused and not duplicated. As a consequence, the obtained models are in general compact and contain few redundant worlds. For instance, the models in Figures 5 and 6 are the minimal countermodels for the formulas $S$ and $T$, respectively. The model in Figure 6 is particularly significant, since it is a DAG but not a tree; hence, it cannot be obtained by the mentioned standard proof-search procedures, which only generate tree-shaped models. For instance, let us consider the prover 1s\text{j}, implementing in JTabWb
Duality between Unprovability and Provability in Forward Refutation-search for IPL

Fig. 11. The countermodels for $S$ and $T$ (see Example 3.7) built by the prover $lsj$ [10].

Fig. 12. Countermodel for $N_{17}$.

a backward proof-search procedure for the calculi presented in Reference [10]. For unprovable formulas, $lsj$ yields countermodels of minimal height; however, the obtained models are always trees, hence they might contain redundant worlds. In Figure 11, we show the countermodels built by $lsj$ for the formulas $S$ and $T$; in the left-hand side model, the worlds 3 and 5 can be overlapped; in the right-hand side model, world 6 can be merged with 7 and 4 with 5. As another significant example, let us consider the one-variable formulas $N_i$ of the Nishimura family [3], which are not valid in IPL:

$N_1 = p$, $N_{2n+3} = N_{2n+1} \lor N_{2n+2}$,

$N_2 = \neg p$, $N_{2n+4} = N_{2n+3} \supset N_{2n+1}$, $n \geq 0$.

We recall that the aforementioned formulas $S$ and $T$ are equivalent to $N_{10}$ and $N_9$, respectively. For formulas $N_j$ our approach generates the standard “tower-like” minimum countermodel [3]. For example, in Figure 12, we display the “tower-like” countermodel for $N_{17}$ generated by the $frj$ prover; in contrast, $lsj$ fails to build a countermodel for such a formula.

As for References [18, 25, 26], we notice that these papers do not explicitly study the height of the extracted countermodels and do not address the problem of minimality.

A method to test the validity of intuitionistic formulas, based on the construction of small countermodels, is presented in References [19, 28]. Let $G$ be the goal formula and let us call atoms the subformulas of $G$ that are propositional variables or $\supset$-formulas (note that these are the same kind of formulas handled by $FRj(G)$). The construction of a countermodel for $G$ is based on a fixpoint procedure. The initial configuration is the structure $\mathcal{K}_0 = \langle P_0, \leq, w_0, V \rangle$, where $P_0$ is the power set of the set of atoms, $\leq$ is the subset relation, $w_0$ is the empty set (hence, $\langle P_0, \leq, w_0 \rangle$ is a poset with
minimum element $w_0$) and $V$ is a function mapping $w \in P_0$ to the set $w \cap V$. By iterating a refinement procedure, consisting in removing elements from $P_0$, we eventually get a Kripke model $K_n = (P_n, \leq, w_n, V)$ satisfying the following properties:

- for every $w \in P_n$ and every atom $A$, $K_n, w \models A$ iff $A \in w$;
- for every $w \in P_n$ and every subformula $F$ of $G$, $K_n, w \models F$ iff $F \in C(I(w))$;
- $G$ is not valid in IPL iff $K_n$ is a countermodel for $G$ (namely, $K_n, w_n \not\models G$).

Thus, to decide the validity of $G$, one has to run the process until the fixpoint $K_n$ is reached, and then check whether $K_n$ is a countermodel for $G$ (e.g., by testing whether $G \in C(I(w_n))$). The procedure has been implemented by the system BDDIntKt, using BDD (Binary Decision Diagram) to efficiently represent the posets and the operations on them. By construction, the generated countermodels do not contain redundancies (indeed, distinct worlds of $K_n$ are separated by at least one atom); however, this does not guarantee that $K_n$ is minimal. For instance, with the goal formula $C$ in Example 6.6, the fixpoint model could be isomorphic with the displayed five-world model, which is not minimal. Actually, the properties of the obtained countermodels are not fully investigated in the mentioned papers and BDDIntKt does not explicitly output them; we guess that they are close to the ones obtained with frj. A procedure to generate minimal countermodels using similar ideas and implemented in Answer Set Programming is presented in Reference [15].

We also point out that Reference [28] presents a procedure (not implemented in BDDIntKt) to get a derivation of $G$ in a standard sequent calculus in case the final model $K_n$ is not a countermodel for $G$.

Finally, as a future work, we plan to investigate the applicability of our method to other logics, in particular, to modal logics (see Reference [14] for a preliminary work) and intermediate logics such as Gödel-Dummett logic characterized by linear Kripke models.

REFERENCES

[1] A. Avellone, C. Fiorentini, and A. Momigliano. 2015. A semantical analysis of focusing and contraction in intuitionistic logic. *Fundamenta Informaticae* 140, 3–4 (2015), 247–262. DOI: https://doi.org/10.3233/FI-2015-1253

[2] T. Brock-Nannestad and K. Chaudhuri. 2015. Disproving using the inverse method by iterative refinement of finite approximations. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods* (TABLEAUX’15) (LNCS), H. De Nivelle (Ed.), Vol. 9323. Springer, 153–168. DOI: https://doi.org/10.1007/978-3-319-24312-2_11

[3] A. Chagrov and M. Zakharyaschev. 1997. *Modal Logic*. Oxford University Press.

[4] K. Chaudhuri and F. Pfenning. 2005. A focusing inverse method theorem prover for first-order linear logic. In *Proceedings of the Conference on Automated Deduction* (CADE’05) (LNCS), R. Nieuwenhuis (Ed.), Vol. 3632. Springer, 69–83. DOI: https://doi.org/10.1007/11532231_6

[5] K. Chaudhuri, F. Pfenning, and G. Price. 2006. A logical characterization of forward and backward chaining in the inverse method. In *Proceedings of the International Joint Conference on Automated Reasoning* (IJCAR’06) (LNCS), U. Furbach et al. (Ed.), Vol. 4130. Springer, 97–111. DOI: https://doi.org/10.1007/11814771_9

[6] K. Claessen and D. Rosén. 2015. SAT modulo intuitionistic implications. In *Proceedings of the 20th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning* (LPAR’15), M. Davis, A. Fehnker, A. McIver, and A. Voronkov (Eds.), Vol. 9450. Springer, 622–637.

[7] A. Degtyarev and A. Voronkov. 2001. The inverse method. In *Handbook of Automated Reasoning*, J. A. Robinson et al. (Ed.). Elsevier and MIT Press, 179–272.

[8] K. Donnelly, T. Gibson, N. Krishnaswami, S. Magill, and S. Park. 2004. The inverse method for the logic of bunched implications. In *Proceedings of the International Conference on Logic for Programming, Artificial Intelligence and Reasoning* (LPAR’04) (LNCS), F. Baader et al. (Ed.), Vol. 3452. Springer, 466–480. DOI: https://doi.org/10.1007/978-3-540-32275-7_31

[9] M. Ferrari, C. Fiorentini, and G. Fiorino. 2010. FCube: An efficient prover for intuitionistic propositional logic. In *Proceedings of the International Conference on Logic for Programming, Artificial Intelligence and Reasoning* (LPAR’10) (LNCS), C. G. Fermüller et al. (Ed.), Vol. 6397. Springer, 294–301.
Duality between Unprovability and Provability in Forward Refutation-search for IPL

[10] M. Ferrari, C. Fiorentini, and G. Fiorino. 2013. Contraction-free linear depth sequent calculi for intuitionistic propositional logic with the subformula property and minimal depth counter-models. J. Autom. Reason. 51, 2 (2013), 129–149. DOI: https://doi.org/10.1007/s10817-012-9252-7

[11] M. Ferrari, C. Fiorentini, and G. Fiorino. 2013. A terminating evaluation-driven variant of G3i. In Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX’13) (LNCS), D. Galmiche et al. (Eds.), Vol. 8123. Springer, 104–118.

[12] M. Ferrari, C. Fiorentini, and G. Fiorino. 2015. An evaluation-driven decision procedure for G3i. ACM Trans. Comput. Logic 16, 1 (2015), 8:1–8:37.

[13] M. Ferrari, C. Fiorentini, and G. Fiorino. 2017. JTabWb: A Java framework for implementing terminating sequent and tableau calculi. Fundamenta Informaticae 150 (2017), 119–142.

[14] M. Ferrari, C. Fiorentini, and G. Fiorino. 2018. Forward countermodel construction in modal logic K. In Proceedings of the 33rd Italian Conference on Computational Logic (CEUR Workshop Proceedings), P. Felli and M. Montali (Eds.), Vol. 2214. CEUR-WS.org, 75–81.

[15] C. Fiorentini. 2019. An ASP approach to generate minimal countermodels in intuitionistic propositional logic. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI’19), S. Kraus (Ed.). 1675–1681. DOI: https://doi.org/10.24963/ijcai.2019/232

[16] C. Fiorentini and M. Ferrari. 2017. A forward unprovability calculus for intuitionistic propositional logic. In Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX’17) (LNCS), R. A. Schmidt and C. Nalon (Eds.), Vol. 10501. Springer, 114–130. DOI: https://doi.org/10.1007/978-3-319-66902-1_7

[17] C. Fiorentini, R. Goré, and S. Graham-Lengrand. 2019. A proof-theoretic perspective on smt-solving for intuitionistic propositional logic. In Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX’19) (LNCS), S. Cerrito and A. Popescu (Eds.), Vol. 11714. Springer, 111–129. DOI: https://doi.org/10.1007/978-3-030-29026-9

[18] R. Goré and L. Postniece. 2010. Combining derivations and refutations for cut-free completeness in bi-intuitionistic logic. J. Logic Comput. 20, 1 (2010), 233–260.

[19] R. Goré and J. D. Monk Thomson. 2012. BDD-based automated reasoning for propositional bi-intuitionistic tense logics. In Proceedings of the 6th International Joint Conference on Automated Reasoning (IJCAR’12), Bernhard Gramlich, Dale Miller, and Uli Sattler (Eds.), Vol. 7364. Springer, 301–315.

[20] J. M. Howe. 1997. Two loop detection mechanisms: A comparision. In Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX’97) (LNCS), D. Galmiche (Ed.), Vol. 1227. Springer, 188–200.

[21] L. Kovács, A. Mantsivoda, and A. Voronkov. 2013. The inverse method for many-valued logics. In Proceedings of the Mexican International Conference on Artificial Intelligence (MICAI’13) (LNCS), F. Castro-Espinoza et al. (Ed.), Vol. 8265. Springer, 225–232. DOI: https://doi.org/10.1007/978-3-642-45114-0_2

[22] V. Lifschitz. 1989. What is the inverse method? J. Autom. Reason. 5, 1 (1989), 1–23. https://doi.org/10.1007/BF00245018.

[23] S. Ju. Maslov. 1967. An invertible sequential version of the constructive predicate calculus. Zap. Naučn. Sem. Leningrad. Otdel. Mat. Inst. Steklov. 4 (1967), 96–111.

[24] S. McLaughlin and F. Pfenning. 2008. Imogen: Focusing the polarized inverse method for intuitionistic propositional logic. In Proceedings of the Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR’08) (LNCS), I. Cervesato et al. (Ed.), Vol. 5330. Springer, 174–181.

[25] S. Negri. 2014. Proofs and countermodels in non-classical logics. Logica Universalis 8, 1 (2014), 25–60. DOI: https://doi.org/10.1007/s11787-014-0097-1

[26] L. Pinto and R. Dyckhoff. 1995. Loop-free construction of counter-models for intuitionistic propositional logic. In Proceedings of the Symposia Gaussiana, Conference A, Behara et al. (Ed.), Walter de Gruyter, Berlin, 225–232.

[27] T. Skura. 1989. A complete syntactical characterization of the intuitionistic logic. Rep. Math. Logic 75, 8 (1989), 75–80.

[28] J. Thomson. 2014. Using BDDs for Non-Classical Propositional Theorem Proving. Ph.D. Dissertation. The Australian National University.

[29] A. S. Troelstra and H. Schwichtenberg. 2000. Basic Proof Theory (2nd ed.). Cambridge Tracts in Theoretical Computer Science, Vol. 43. Cambridge University Press.

Received April 2018; revised October 2019; accepted November 2019

ACM Transactions on Computational Logic, Vol. 21, No. 3, Article 22. Publication date: March 2020.