SAT-based Decision Procedure for Analytic Pure Sequent Calculi

Ori Lahav and Yoni Zohar*

School of Computer Science, Tel Aviv University, Israel
orilahav@post.tau.ac.il yoni.zohar@cs.tau.ac.il

Abstract. We identify a wide family of analytic sequent calculi for propositional non-classical logics whose derivability problem can be uniformly reduced to SAT. The proposed reduction is based on interpreting these calculi using non-deterministic semantics. Its time complexity is polynomial, and, in fact, linear for a useful subfamily. We further study an extension of such calculi with *Next* operators, and show that this extension preserves analyticity and is subject to a similar reduction to SAT. A particular interesting instance of these results is a HORNSAT-based linear-time decision procedure for Gurevich and Neeman's primal infon logic and several natural extensions of it.

1 Introduction

Sequent calculi provide a flexible well-behaved proof-theoretic framework for a huge variety of different logics. Usually, they allow us to perform proof-search for the corresponding logic. The fundamental property of cut-elimination is traditionally proven, as it often guarantees the adequacy of a given sequent calculus for this task. Nevertheless, a great deal of ingenuity is required for developing an efficient proof-search algorithms for cut-free sequent calculi (see, e.g., [12]).

In this work we identify a general case in which it is possible to replace proof-search by SAT solving. While SAT is NP-complete, it is considered "easy" when it comes to real-world applications. Indeed, there are many off-the-shelf SAT solvers, that, despite an exponential worst-case time complexity, are considered extremely efficient (see, e.g., [14]).

We focus on a general family of relatively simple sequent calculi, called *pure sequent calculi*. Roughly speaking, these are propositional fully-structural calculi (calculi that include the structural rules: exchange, contraction and weakening), whose derivation rules do not enforce any limitations on the context formulas (following [1], the adjective "pure" stands for this requirement). We do not assume that the calculi enjoy cut-elimination. Instead, we formulate an analyticity property, that generalizes the usual subformula property, and show that the derivability problem in each *analytic* pure calculus can be reduced to (the complement of) SAT. This result applies to a wide range of sequent calculi for

* This research was supported by The Israel Science Foundation (grant no. 280-10).

different non-classical logics, including important three and four valued logics and various paraconsistent logics.

To achieve this result we utilize an alternative semantic view of pure sequent calculi. For that, we have extended the correspondence between sequent calculi and their *bivaluation* semantics from [7], so the semantics is tied to the set of formulas allowed to be used in derivations. The derivability problem in a given analytic sequent calculus is then replaced by small countermodel search, which can be translated into a SAT instance. In turn, one can construct a countermodel from a satisfying assignment given by the SAT solver in the form of a bivaluation (or a functional Kripke model when *Next* operators are involved, see below).

The efficiency of the proposed SAT-based decision procedure obviously depends on the time complexity of the reduction. This complexity, as we show, is $O(n^k)$, where n is the size of the input sequent and k is determined according to the structure of the particular calculus. For a variety of useful calculi, we obtain a *linear time* reduction. This paves the way to efficient uniform decision procedures for all logics that can be covered in this framework. In particular, we identify a subfamily of calculi for which the generated SAT instances consist of *Horn clauses*. In these calculi the derivability problem can be decided in linear time by applying the reduction and using a linear time HORNSAT solver [13].

In Section 6 we extend this method to analytic pure calculi augmented with a finite set of *Next* operators. These are often employed in temporal logics. Moreover, in primal infon logic [11] *Next* operators, as we show, play the role of quotations, which are indispensable in the application of this logic for the access control language DKAL. We show that all analytic pure calculi, satisfying a certain natural requirement, can be augmented with *Next* operators, while retaining their analyticity. In turn, the general reduction to SAT is extended to analytic calculi with *Next* operators, based on a (possibly non-deterministic) Kripke-style semantic characterization. A HORNSAT-based decision procedure for primal infon logic with quotations is then obtained as a particular instance. In addition, in this general framework we are able to formulate several extensions of primal infon logic with additional natural rules, making it somewhat "closer" to classical logic, and still decidable in linear time.

Related Works. Our method generalizes the reduction given in [6] of quotations-free primal infon logic to classical logic. For the case of primal infon logic with quotations the proposed reduction produces practically equivalent outputs to the reduction in [8] from this logic to Datalog. A general methodology for translating derivability questions in Hilbertian deductive systems to Datalog was introduced in [9]. However, this method may produce infinitely many Datalog premises, and then it is difficult to use for computational purposes. In contrast, the reduction proposed in this paper always produces finite SAT instances. This is possible due to our focus on *analytic* calculi. Since Hilbertian systems are rarely analytic, we handle Gentzen-type calculi.

Due to lack of space, some proofs are omitted, and will appear in an extended version.

2 Preliminaries

A *propositional language* $\mathcal{L}$ consists of a countably infinite set of atomic variables $At = \{p_1, p_2, \ldots\}$ and a finite set $\diamondsuit_{\mathcal{L}}$ of propositional connectives. The set of all n-ary connectives of $\mathcal{L}$ is denoted by $\diamondsuit^n_{\mathcal{L}}$. We identify $\mathcal{L}$ with its set of well-formed formulas (e.g. when writing $\psi \in \mathcal{L}$ or $\mathcal{F} \subseteq \mathcal{L}$). A *sequent* is a pair $\langle \Gamma, \Delta \rangle$ (denoted by $\Gamma \Rightarrow \Delta$) where Γ and Δ are finite sets of formulas. We employ the standard sequent notations, e.g. when writing expressions like $\Gamma, \psi \Rightarrow \Delta$ or $\Rightarrow \psi$. The union of sequents is defined by $(\Gamma_1 \Rightarrow \Delta_1) \cup (\Gamma_2 \Rightarrow \Delta_2) = (\Gamma_1 \cup \Gamma_2 \Rightarrow \Delta_1 \cup \Delta_2)$. For a sequent $\Gamma \Rightarrow \Delta$, $frm(\Gamma \Rightarrow \Delta) = \Gamma \cup \Delta$. This notation is naturally extended to sets of sequents. Given $\mathcal{F} \subseteq \mathcal{L}$, we say that a formula φ is an $\mathcal{F}$*-formula* if $\varphi \in \mathcal{F}$ and that a sequent s is an $\mathcal{F}$*-sequent* if $frm(s) \subseteq \mathcal{F}$. A *substitution* is a function from At to some propositional language. A substitution σ is naturally extended to any propositional language by $\sigma(\diamond(\psi_1, \ldots, \psi_n)) = \diamond(\sigma(\psi_1), \ldots, \sigma(\psi_n))$ for every compound formula $\diamond(\psi_1, \ldots, \psi_n)$. Substitutions are also naturally extended to sets of formulas, sequents and sets of sequents. In what follows, $\mathcal{L}$ denotes an arbitrary propositional language.

3 Pure Sequent Calculi

In this section we define the family of pure sequent calculi, and provide some examples for known calculi that fall in this family.

Definition 1. A *pure rule* is a pair $\langle S, s \rangle$ (denoted by $S \,/\, s$) where S is a finite set of sequents and s is a sequent. The elements of S are called the *premises* of the rule and s is called the *conclusion* of the rule. An *application* of a pure rule $\{s_1, \ldots s_n\} \,/\, s$ is any inference step of the form

$$\frac{\sigma(s_1) \cup c \quad \ldots \quad \sigma(s_n) \cup c}{\sigma(s) \cup c}$$

where σ is a substitution and c is a sequent (called a *context sequent*). The sequents $\sigma(s_i) \cup c$ are called the *premises* of the application and $\sigma(s) \cup c$ is called the *conclusion* of the application. The set S of premises of a pure rule is usually written without set braces, and its elements are separated by ";".

Note that we differentiate between rules and their applications, and use different notations for them.

Example 1. The following are pure rules:

$$p_1 \Rightarrow p_2 \,/\, \Rightarrow p_1 \supset p_2 \qquad\qquad \Rightarrow p_1; p_2 \Rightarrow \,/\, p_1 \supset p_2 \Rightarrow \qquad\qquad / \Rightarrow p_1 \supset p_1$$

Applications of these rules have respectively the forms:

$$\frac{\Gamma, \psi_1 \Rightarrow \psi_2, \Delta}{\Gamma \Rightarrow \psi_1 \supset \psi_2, \Delta} \qquad \frac{\Gamma \Rightarrow \psi_1, \Delta \qquad \Gamma, \psi_2 \Rightarrow \Delta}{\Gamma, \psi_1 \supset \psi_2 \Rightarrow \Delta} \qquad \frac{}{\Gamma \Rightarrow \psi \supset \psi, \Delta}$$

Note that the usual rule for introducing implication on the right-hand side in intuitionistic logic is not a pure rule, since it allows only *left* context formulas.

In turn, pure sequent calculi are finite sets of pure rules. To make them fully-structural (in addition to defining sequents as pairs of *sets*), the weakening rule, the identity axiom and the cut rule are allowed to be used in derivations.

Definition 2. A *pure calculus* is a finite set of pure rules. A (standard) *proof* in a pure calculus $\mathbf{G}$ is defined as usual, where in addition to applications of the pure rules of $\mathbf{G}$, the following standard application schemes may be used:

$$(weak)\frac{\Gamma \Rightarrow \Delta}{\Gamma', \Gamma \Rightarrow \Delta, \Delta'} \qquad (id)\ \frac{}{\Gamma, \psi \Rightarrow \psi, \Delta} \qquad (cut)\ \frac{\Gamma \Rightarrow \psi, \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}$$

Henceforth, we consider only pure rules and pure calculi, and may refer to them simply as *rules* and *calculi*. By an $\mathcal{L}$-rule ($\mathcal{L}$-calculus) we mean a rule (calculus) that includes only connectives from $\mathcal{L}$.

Notation 1. For an $\mathcal{L}$-calculus $\mathbf{G}$, a set $\mathcal{F} \subseteq \mathcal{L}$ of formulas, and an $\mathcal{F}$-sequent s, we write $\vdash^{\mathcal{F}}_{\mathbf{G}} s$ if there is a proof of s in $\mathbf{G}$ consisting only of $\mathcal{F}$-sequents. For $\mathcal{F} = \mathcal{L}$, we write $\vdash_{\mathbf{G}} s$.

Example 2. The propositional fragment of Gentzen's fundamental sequent calculus for classical logic can be directly presented as a pure calculus, denoted henceforth by **LK**. It consists of the following rules:

$(\neg \Rightarrow)\quad \Rightarrow p_1 \,/\, \neg p_1 \Rightarrow$ $\qquad (\Rightarrow \neg)\quad p_1 \Rightarrow \,/\, \Rightarrow \neg p_1$
$(\wedge \Rightarrow)\quad p_1, p_2 \Rightarrow \,/\, p_1 \wedge p_2 \Rightarrow$ $\qquad (\Rightarrow \wedge) \Rightarrow p_1; \Rightarrow p_2 \,/\, \Rightarrow p_1 \wedge p_2$
$(\vee \Rightarrow)\ p_1 \Rightarrow; p_2 \Rightarrow \,/\, p_1 \vee p_2 \Rightarrow$ $\qquad (\Rightarrow \vee)\quad \Rightarrow p_1, p_2 \,/\, \Rightarrow p_1 \vee p_2$
$(\supset\Rightarrow) \Rightarrow p_1; p_2 \Rightarrow \,/\, p_1 \supset p_2 \Rightarrow$ $\qquad (\Rightarrow\supset)\quad p_1 \Rightarrow p_2 \,/\, \Rightarrow p_1 \supset p_2$

Besides **LK** there are many sequent calculi for non-classical logics (admitting cut-elimination) that fall in this framework. These include calculi for well-known three and four-valued logics, various calculi for paraconsistent logics, and all canonical and quasi-canonical sequent systems [3,4,5,7].

Example 3. The calculus for (quotations free) primal infon logic from [11], can be directly presented as a pure calculus, that we call **P**. It consists of the rules $(\wedge \Rightarrow)$, $(\Rightarrow \wedge)$, $(\Rightarrow \vee)$ and $(\supset\Rightarrow)$ of **LK**, together with the two rules $\Rightarrow p_2 \,/ \Rightarrow p_1 \supset p_2$ and $\emptyset \,/ \Rightarrow \top$.

Example 4. The calculus from [3] for da Costa's historical paraconsistent logic C_1 can be directly presented as a pure calculus, that we call $\mathbf{G}_{C_1}$. It consists of the rules of **LK** except for $(\neg \Rightarrow)$ that is replaced by the following rules:

$p_1 \Rightarrow \,/\, \neg\neg p_1 \Rightarrow$
$\Rightarrow p_1; \Rightarrow \neg p_1 \,/\, \neg(p_1 \wedge \neg p_1) \Rightarrow$ $\qquad \neg p_1 \Rightarrow; \neg p_2 \Rightarrow \,/\, \neg(p_1 \wedge p_2) \Rightarrow$
$\neg p_1 \Rightarrow; p_2, \neg p_2 \Rightarrow \,/\, \neg(p_1 \vee p_2) \Rightarrow$ $\qquad p_1, \neg p_1 \Rightarrow; \neg p_2 \Rightarrow \,/\, \neg(p_1 \vee p_2) \Rightarrow$
$p_1 \Rightarrow; p_2, \neg p_2 \Rightarrow \,/\, \neg(p_1 \supset p_2) \Rightarrow$ $\qquad p_1, \neg p_1 \Rightarrow; \neg p_2 \Rightarrow \,/\, \neg(p_1 \supset p_2) \Rightarrow$

3.1 Analyticity

Our goal in this paper is to provide a general effective tool to solve the *derivability problem* for a given pure calculus.

Definition 3. The *derivability problem* for an $\mathcal{L}$-calculus **G** is given by:
Input: An $\mathcal{L}$-sequent s. **Question:** Does $\vdash_{\mathbf{G}} s$?

Obviously, one cannot expect to have decision procedures for the derivability problem for all pure calculi.[1] Thus we require our calculi to admit a *generalized analyticity property*. Analyticity is a crucial property of proof systems. In the case of fully-structural propositional sequent calculi it usually implies their decidability and consistency (the fact that the empty sequent is not derivable). Roughly speaking, a calculus is analytic if whenever a sequent s is provable in it, s can be proven using only the "syntactic material available inside s". This "material" is usually taken to consist of all subformulas occurring in s, and then analyticity amounts to the (global) subformula property. However, weaker restrictions on the formulas that are allowed to appear in proofs of a given sequent may suffice for decidability. Next we introduce a generalized analyticity property based on an extended notion of a subformula. In what follows, $\circledcirc$ denotes an arbitrary set of unary connectives (assumed to be a subset of $\diamondsuit^1_{\mathcal{L}}$).

Definition 4. A formula φ is a $\circledcirc$-*subformula* of a formula ψ if either φ is a subformula of ψ or $\varphi = \circ\psi'$ for some $\circ \in \circledcirc$ and proper subformula ψ' of ψ.

Note that the $\circledcirc$-subformula relation is transitive.

Notation 2. $sub^{\circledcirc}(\psi)$ denotes the set of $\circledcirc$-subformulas of ψ. This notation is extended to sets of formulas and sequents in the obvious way.

Example 5. $sub^{\{\neg\}}(\neg(p_1 \supset p_2)) = \{p_1, p_2, \neg p_1, \neg p_2, p_1 \supset p_2, \neg(p_1 \supset p_2)\}$.

Definition 5. An $\mathcal{L}$-calculus **G** is called $\circledcirc$-*analytic* if $\vdash_{\mathbf{G}} s$ implies $\vdash^{sub^{\circledcirc}(s)}_{\mathbf{G}} s$ for every $\mathcal{L}$-sequent s.

Note that $sub^{\emptyset}(\varphi)$ is the set of usual subformulas of φ, and so $\emptyset$-analyticity is the usual subformula property.

Example 6. The calculi **LK**, **P** and $\mathbf{G}_{C_1}$ (presented in previous examples) admit cut-elimination. This, combined with the structure of their rules, directly entails that **LK** and **P** are $\emptyset$-analytic, while $\mathbf{G}_{C_1}$ is $\{\neg\}$-analytic. Example 10 below shows an extension of **P** that does not admit cut-elimination, but is still $\emptyset$-analytic.

Example 7. A cut-free sequent calculus for Łukasiewicz three-valued logic was presented in [2]. This calculus, that we call $\mathbf{G}_3$, can be directly presented as a pure calculus. For example, the rules involving implication are the following:

$$\neg p_1 \Rightarrow; p_2 \Rightarrow; \Rightarrow p_1, \neg p_2 \,/\, p_1 \supset p_2 \Rightarrow \qquad p_1 \Rightarrow p_2; \neg p_2 \Rightarrow \neg p_1 \,/\, \Rightarrow p_1 \supset p_2$$
$$p_1, \neg p_2 \Rightarrow \,/\, \neg(p_1 \supset p_2) \Rightarrow \qquad \Rightarrow p_1; \Rightarrow \neg p_2 \,/\, \Rightarrow \neg(p_1 \supset p_2)$$

The structure of its rules, together with the fact that this calculus admits cut-elimination, directly entail that $\mathbf{G}_3$ is $\{\neg\}$-analytic.

[1] Any Hilbert calculus H (without side conditions on rule applications) can be translated to a pure sequent calculus $\mathbf{G}_H$, by taking a rule of the form $\Rightarrow \psi_1; \ldots; \Rightarrow \psi_n \,/\, \Rightarrow \psi$ for each Hilbert-style derivation rule $\psi_1, \ldots, \psi_n \,/\, \psi$ (where $n = 0$ for axioms). It is easy to show that ψ is derivable from Γ in H iff $\vdash_{\mathbf{G}_H} \Gamma \Rightarrow \psi$.

To end this section, we point out a useful property of pure calculi. We call a rule *axiomatic* if it has an empty set of premises. In turn, a calculus is *axiomatic* if it consists solely of axiomatic rules. We show that every calculus is equivalent (in the sense defined below) to an axiomatic calculus, obtained by "multiplying out" the rules, and "moving" the formulas in the premises to the opposite side of the conclusion.

Definition 6. A *component* of a sequent $\Gamma \Rightarrow \Delta$ is any sequent of the form $\psi \Rightarrow$ where $\psi \in \Gamma$ or $\Rightarrow \psi$ where $\psi \in \Delta$. A sequent s is called a *combination* of a set S of sequents if there are distinct sequents $s_1, \ldots, s_n$ and respective components $s'_1, \ldots, s'_n$ such that $S = \{s_1, \ldots, s_n\}$ and $s = s'_1 \cup \ldots \cup s'_n$.

Definition 7. Let $r = S \,/\, \Gamma \Rightarrow \Delta$ be a rule. The set $Ax(r)$ consists of all axiomatic rules of the form $\emptyset \,/\, \Gamma, \Delta' \Rightarrow \Gamma', \Delta$ where $\Gamma' \Rightarrow \Delta'$ is a combination of S. In turn, given a calculus $\mathbf{G}$, $Ax(\mathbf{G})$ denotes the calculus obtained from $\mathbf{G}$ by replacing each non-axiomatic rule r of $\mathbf{G}$ by $Ax(r)$.

Example 8. For $r = \neg p_1 \Rightarrow ; p_2 \Rightarrow ; \Rightarrow p_1, \neg p_2 \,/\, p_1 \supset p_2 \Rightarrow$, $Ax(r)$ consists of the axiomatic rules $\emptyset \,/\, p_1, p_1 \supset p_2 \Rightarrow \neg p_1, p_2$ and $\emptyset \,/\, \neg p_2, p_1 \supset p_2 \Rightarrow \neg p_1, p_2$.

Proposition 1. *Let $\mathbf{G}$ be an $\mathcal{L}$-calculus. For every set $\mathcal{F} \subseteq \mathcal{L}$ and $\mathcal{F}$-sequent s, if $\vdash^{\mathcal{F}}_{\mathbf{G}} s$ then $\vdash^{\mathcal{F}}_{Ax(\mathbf{G})} s$. For $\mathcal{F} = \mathcal{L}$ the converse holds as well. Moreover, if $\mathbf{G}$ is $\circledcirc$-analytic then so is $Ax(\mathbf{G})$.*

As happens for **LK**, it is likely that $Ax(\mathbf{G})$ does not admit cut-elimination even when $\mathbf{G}$ does.

4 Semantics for Pure Sequent Calculi

In this section we present a semantic view of pure calculi, that plays a major role in the reduction of their derivability problem to SAT. For that matter, we follow [7] and use *bivaluations* – functions assigning a binary truth value to each formula. Pure rules are naturally translated into conditions on bivaluations. In order to have finite models, we strengthen the correspondence in [7] and consider *partial* bivaluations. These correspond exactly to derivations that are confined to a certain set of formulas.

Definition 8. A *bivaluation* is a function v from some set $dom(v)$ of formulas in some propositional language to $\{0, 1\}$. A bivaluation v is extended to $dom(v)$-sequents by: $v(\Gamma \Rightarrow \Delta) = 1$ iff $v(\varphi) = 0$ for some $\varphi \in \Gamma$ or $v(\varphi) = 1$ for some $\varphi \in \Delta$. v is extended to sets of $dom(v)$-sequents by: $v(S) = min\,\{v(s) \mid s \in S\}$, where $min\,\emptyset = 1$. Given a set $\mathcal{F}$ of formulas, by an *$\mathcal{F}$-bivaluation* we refer to a bivaluation v with $dom(v) = \mathcal{F}$.

Definition 9. A bivaluation v *respects* a rule $S \,/\, s$ if $v(\sigma(S)) \leq v(\sigma(s))$ for every substitution σ such that $\sigma(frm(S \,/\, s)) \subseteq dom(v)$.[2] v is called **G**-*legal* for a calculus $\mathbf{G}$ if it respects all rules of $\mathbf{G}$.

[2] frm is extended to pure rules in the obvious way, i.e. $frm(S \,/\, s) = frm(S) \cup frm(s)$.

Example 9. A $\{p_1, \neg\neg p_1\}$-bivaluation v respects the rule $p_1 \Rightarrow \ / \neg\neg p_1 \Rightarrow$ iff either $v(p_1) = v(\neg\neg p_1) = 0$ or $v(p_1) = 1$. Note that **LK**-legal bivaluations are exactly usual classical valuation functions.

Theorem 1 (Soundness and Completeness). *Let* **G** *be an* $\mathcal{L}$*-calculus,* $\mathcal{F}$ *be a set of* $\mathcal{L}$*-formulas, and* s *be an* $\mathcal{F}$*-sequent. Then,* $\vdash_{\mathbf{G}}^{\mathcal{F}} s$ *iff* $v(s) = 1$ *for every* **G***-legal* $\mathcal{F}$*-bivaluation* v*.*

Using Theorem 1, we are able to formulate a semantic property that corresponds exactly to $\circledcirc$-analyticity:

Definition 10. An $\mathcal{L}$-calculus **G** is called *semantically* $\circledcirc$*-analytic* if every **G**-legal bivaluation v can be extended to a **G**-legal $\mathcal{L}$-bivaluation, provided that $dom(v)$ is a finite subset of $\mathcal{L}$ closed under $\circledcirc$-subformulas.

Theorem 2. *An* $\mathcal{L}$*-calculus* **G** *is* $\circledcirc$*-analytic iff it is semantically* $\circledcirc$*-analytic.*

Proof. Suppose that there is an $\mathcal{L}$-sequent s such that $\vdash_{\mathbf{G}} s$ and $\nvdash_{\mathbf{G}}^{sub^{\circledcirc}(s)} s$. According to Theorem 1, there exists a **G**-legal $sub^{\circledcirc}(s)$-bivaluation v such that $v(s) = 0$, but $u(s) = 1$ for every **G**-legal $\mathcal{L}$-bivaluation u. Therefore, v cannot be extended to a **G**-legal $\mathcal{L}$-bivaluation. In addition, $dom(v) = sub^{\circledcirc}(s)$ is finite and closed under $\circledcirc$-subformulas.

For the converse, suppose that v is a **G**-legal bivaluation, $dom(v)$ is finite and closed under $\circledcirc$-subformulas, and v cannot be extended to a **G**-legal $\mathcal{L}$-bivaluation. Let $\Gamma = \{\psi \in dom(v) \mid v(\psi) = 1\}$, $\Delta = \{\psi \in dom(v) \mid v(\psi) = 0\}$, and $s = \Gamma \Rightarrow \Delta$. Then $dom(v) = sub^{\circledcirc}(s)$ and $v(s) = 0$. We show that $u(s) = 1$ for every **G**-legal $\mathcal{L}$-bivaluation u. Indeed, every such u does not extend v, and so $u(\psi) \neq v(\psi)$ for some $\psi \in dom(v)$. Then, $u(\psi) = 0$ if $\psi \in \Gamma$, and $u(\psi) = 1$ if $\psi \in \Delta$. In either case, $u(s) = 1$. By Theorem 1, $\nvdash_{\mathbf{G}}^{sub^{\circledcirc}(s)} s$ and $\vdash_{\mathbf{G}} s$. ☐

The left-to-right direction of Theorem 2 is used to prove the correctness of the reduction in the next section. The converse provides a semantic method to prove $\circledcirc$-analyticity, that can be used alternatively to deriving analyticity as a consequence of cut-elimination.

Example 10. An extension of primal infon logic, that we call **EP**, extends the calculus **P** (see Example 3) with the following classically valid axiomatic rules:

$\emptyset \,/ \Rightarrow \bot \supset p_1$	$\emptyset \,/\, p_1 \vee p_1 \Rightarrow p_1$	$\emptyset \,/ \Rightarrow p_1 \supset p_1$
$\emptyset \,/\, \bot \Rightarrow$	$\emptyset \,/\, p_1 \vee p_2 \Rightarrow p_2 \vee p_1$	$\emptyset \,/ \Rightarrow (p_1 \wedge p_2) \supset p_1$
$\emptyset \,/\, \bot \vee p_1 \Rightarrow p_1$	$\emptyset \,/\, p_1 \vee (p_1 \wedge p_2) \Rightarrow p_1$	$\emptyset \,/ \Rightarrow (p_1 \wedge p_2) \supset p_2$
$\emptyset \,/\, p_1 \vee \bot \Rightarrow p_1$	$\emptyset \,/\, (p_1 \wedge p_2) \vee p_1 \Rightarrow p_1$	$\emptyset \,/ \Rightarrow p_2 \supset (p_1 \supset p_2)$

Note that none of these rules is derivable in **P**. It is possible to prove that **EP** is $\emptyset$-analytic by showing that it is semantically $\emptyset$-analytic and applying Theorem 2.

5 Reduction to Classical Satisfiability

In this section we present a reduction from the derivability problem for a given $\circledcirc$-analytic pure calculus to the complement of SAT. SAT instances are taken to be CNFs represented as sets of clauses, where clauses are sets of literals (that is, atomic variables and their negations, denoted by overlines). The set $\{x_\psi \mid \psi \in \mathcal{L}\}$ is used as the set of atomic variables in the SAT instances. The translation of sequents to SAT instances is naturally given by:

Definition 11. For a sequent $\Gamma \Rightarrow \Delta$:

$$\mathtt{SAT}^+(\Gamma \Rightarrow \Delta) := \{\{\overline{x_\psi} \mid \psi \in \Gamma\} \cup \{x_\psi \mid \psi \in \Delta\}\}\,.$$

$$\mathtt{SAT}^-(\Gamma \Rightarrow \Delta) := \{\{x_\psi\} \mid \psi \in \Gamma\} \cup \{\{\overline{x_\psi}\} \mid \psi \in \Delta\}\,.$$

This translation captures the semantic interpretation of sequents. Indeed, given an $\mathcal{L}$-bivaluation v and a classical assignment u that assigns true to x_ψ iff $v(\psi) = 1$, we have that for every $\mathcal{L}$-sequent s: $v(s) = 1$ iff u satisfies $\mathtt{SAT}^+(s)$, and $v(s) = 0$ iff u satisfies $\mathtt{SAT}^-(s)$. Now, in order for a bivaluation to be **G**-legal for some calculus **G**, it should satisfy the semantic restrictions arising from the rules of **G**. These restrictions can be directly encoded as SAT instances (as done, e.g., in [17] for the particular case of the classical truth tables). For this purpose, the use of $Ax(\mathbf{G})$ (see Definition 7) instead of **G** is technically convenient.

Definition 12. The SAT instance associated with a given $\mathcal{L}$-calculus **G**, an $\mathcal{L}$-sequent s and a set $\circledcirc \subseteq \Diamond^1_{\mathcal{L}}$ is given by:

$$\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s) := \bigcup \left\{\mathtt{SAT}^+(\sigma(s')) \mid \emptyset \,/\, s' \in Ax(\mathbf{G}), \sigma(\mathit{frm}(s')) \subseteq \mathit{sub}^{\circledcirc}(s)\right\}.$$

Example 11. Consider the $\{\neg\}$-analytic calculus $\mathbf{G}_3$ for Łukasiewicz three-valued logic from Example 7. Following Example 8, $Ax(\mathbf{G}_3)$ contains the axiomatic rules $\emptyset \,/\, p_1, p_1 \supset p_2 \Rightarrow \neg p_1, p_2$ and $\emptyset \,/\, \neg p_2, p_1 \supset p_2 \Rightarrow \neg p_1, p_2$. Given a sequent s, $\mathtt{SAT}^{\{\neg\}}(\mathbf{G}_3, s)$ includes the clause $\{\overline{x_{\psi_1}}, \overline{x_{\psi_1 \supset \psi_2}}, x_{\neg\psi_1}, x_{\psi_2}\}$ and the clause $\{\overline{x_{\neg\psi_2}}, \overline{x_{\psi_1 \supset \psi_2}}, x_{\neg\psi_1}, x_{\psi_2}\}$ for every formula of the form $\psi_1 \supset \psi_2$ in $\mathit{sub}^{\{\neg\}}(s)$.

Theorem 3. *Let* **G** *be a* $\circledcirc$*-analytic* $\mathcal{L}$*-calculus and* s *be an* $\mathcal{L}$*-sequent. Then* $\vdash_{\mathbf{G}} s$ *iff* $\mathit{SAT}^{\circledcirc}(\mathbf{G}, s) \cup \mathit{SAT}^-(s)$ *is unsatisfiable.*

Proof. Suppose that $\nvdash_{\mathbf{G}} s$. By Proposition 1, $\nvdash_{Ax(\mathbf{G})} s$. By Theorem 1, there exists an $Ax(\mathbf{G})$-legal $\mathcal{L}$-bivaluation v such that $v(s) = 0$. The classical assignment u that assigns true to a variable x_ψ iff $v(\psi) = 1$ satisfies $\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^-(s)$.

For the converse, let u be a classical assignment satisfying the SAT instance $\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^-(s)$. Consider the $\mathit{sub}^{\circledcirc}(s)$-bivaluation v defined by $v(\psi) = 1$ iff u assigns true to x_ψ. It is easy to see that since u satisfies $\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s)$, v is $Ax(\mathbf{G})$-legal. u also satisfies $\mathtt{SAT}^-(s)$, and hence $v(s) = 0$. Since **G** is $\circledcirc$-analytic, so is $Ax(\mathbf{G})$ (by Proposition 1). By Theorem 2, $Ax(\mathbf{G})$ is semantically $\circledcirc$-analytic, and so v can be extended to an $Ax(\mathbf{G})$-legal $\mathcal{L}$-bivaluation. Theorem 1 entails that $\nvdash_{Ax(\mathbf{G})} s$. By Proposition 1, it follows that $\nvdash_{\mathbf{G}} s$. □

Now, we show that the above reduction is computable in polynomial time.

Definition 13. A rule $S \,/\, s$ is called *k-$\circledcirc$-closed* if there are $\varphi_1, \dots, \varphi_k \in \mathit{frm}(s)$ (called *main formulas*) such $\mathit{frm}(S \,/\, s)$ consists only of $\circledcirc$-subformulas of the φ_i's. A calculus is *k-$\circledcirc$-closed* if each of its rules is k'-$\circledcirc$-closed for some $k' \leq k$.

Example 12. **LK** and **P** (see Examples 2 and 3) are 1-$\emptyset$-closed. $\mathbf{G}_{C_1}$ (see Example 4) is 1-$\{\neg\}$-closed. **EP** (see Example 10) is 2-$\emptyset$-closed, because of the rule $\emptyset \,/\, p_1 \vee p_2 \Rightarrow p_2 \vee p_1$.

Remark 1. Every axiomatic calculus is k-$\circledcirc$-closed for some k (e.g., the maximal number of formulas in its rules). As seen in Proposition 1, every calculus $\mathbf{G}$ is equivalent to the axiomatic calculus $Ax(\mathbf{G})$. Moreover, if $\mathbf{G}$ is k-$\circledcirc$-closed, then so is $Ax(\mathbf{G})$.

Theorem 4. *Let* $\mathbf{G}$ *be a* k-$\circledcirc$*-closed* $\mathcal{L}$*-calculus. Given an* $\mathcal{L}$*-sequent* s*, the SAT instance* $\mathit{SAT}^{\circledcirc}(\mathbf{G}, s) \cup \mathit{SAT}^{-}(s)$ *is computable in* $O(n^k)$ *time, where* n *is the length of the string representing* s*.*

Proof (sketch). The following algorithm computes $\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$:

1. Build a parse tree for the input using standard techniques. As usual, every node represents an occurrence of some subformula in s.
2. Using, e.g., the linear-time algorithm from [10], compress the parse tree into an ordered dag by maximally unifying identical subtrees. After the compression, the nodes of the dag represent subformulas of s, rather than occurrences. Hence we may identify nodes with their corresponding formulas.
3. Traverse the dag. For every $\circ \in \circledcirc$ and node v that has a parent, add a new parent labeled with $\circ$, if such a parent does not exist. To check this it is possible to maintain in each node v a constant-size list of all unary connectives in $\circledcirc$ that label the parents of v. Note that after these additions, the nodes of the dag one-to-one correspond to $sub^{\circledcirc}(s)$.
4. $\mathtt{SAT}^{-}(s)$ is obtained by traversing the dag and generating $\{x_\psi\}$ for every ψ on the left-hand side of s and $\{\overline{x_\psi}\}$ for every ψ on the right-hand side of s.
5. $\mathtt{SAT}^{\circledcirc}(\mathbf{G}, s)$ is generated by looping over all rules in $Ax(\mathbf{G})$. For each rule $\emptyset \,/\, s'$ with main formulas $\varphi_1, \ldots, \varphi_{k'}$ ($k' \leq k$), go over all k'-tuples of nodes in the dag. For each k' nodes $v_1, \ldots, v_{k'}$ check whether $v_1, \ldots, v_{k'}$ match the pattern given by $\varphi_1, \ldots, \varphi_{k'}$, and if so, construct a mapping h from the formulas in $sub^{\circledcirc}(s')$ to their matching nodes. Then construct a clause consisting of a literal $\overline{x_{h(\varphi)}}$ for every φ on the left-hand side of s', and a literal $x_{h(\varphi)}$ for every φ on the right-hand side of s'. Note that only a constant depth of the sub-dags rooted at $v_1, \ldots, v_{k'}$ is considered - that is the complexity of $\varphi_1, \ldots, \varphi_{k'}$, in addition to parents labeled with elements from $\circledcirc$. These are independent of the input sequent s. To see that we generate exactly all required clauses, note that a substitution σ satisfies $\sigma(\mathit{frm}(s')) \subseteq sub^{\circledcirc}(s)$ iff $\sigma(\{\varphi_1, \ldots, \varphi_{k'}\}) \subseteq sub^{\circledcirc}(s)$. Thus a substitution σ satisfies $\sigma(\mathit{frm}(s')) \subseteq sub^{\circledcirc}(s)$ iff there are k' nodes matching the patterns given by $\varphi_1, \ldots, \varphi_{k'}$.

Steps 1,2,3,4 require linear time. Each pattern matching in step 5 is done in constant time, and so handling a k'-$\circledcirc$-closed rule takes $O(n^{k'})$ time. Thus step 5 requires $O(n^k)$ time. $\square$

Remark 2. We employ the same standard computation model of analysis of algorithms used in [11]. A linear time implementation of this algorithm cannot afford the variables x_ψ to literally include a full string representation of ψ. Thus we assume that each node has a key that can be printed and manipulated in constant time (e.g., its memory address).

Corollary 1. *For any $\circledcirc$-analytic calculus* $\mathbf{G}$*, the derivability problem for* $\mathbf{G}$ *is in co-NP.*

The reduction runs in linear time for 1-$\circledcirc$-closed calculi. In such cases, it is natural to identify calculi whose SAT instances can be decided in linear time. This is the case, for example, for instances consisting of *Horn clauses* [13].

Definition 14. A rule r is called a *Horn rule* if $\#_L(r) + \#_R(r) \leq 1$, where $\#_L(r)$ is the number of premises of r whose left-hand side is not empty, and $\#_R(r)$ is the number of formulas on the right-hand side of the conclusion of r. A calculus is called a *Horn calculus* if each of its rules is a Horn rule.

Proposition 2. *Let* $\mathbf{G}$ *be a Horn* $\mathcal{L}$*-calculus and* s *be an* $\mathcal{L}$*-sequent. Then* $\mathit{SAT}^{\circledcirc}(\mathbf{G}, s)$ *consists solely of Horn clauses.*

Corollary 2. *Let* $\mathbf{G}$ *be a $\circledcirc$-analytic,* 1-$\circledcirc$*-closed Horn* $\mathcal{L}$*-calculus. The derivability problem for* $\mathbf{G}$ *can be decided in linear time using a HORNSAT solver.*

Example 13. The derivability problem for **EP** (see Example 10) is decidable in quadratic time, as **EP** is a $\emptyset$-analytic, 2-$\emptyset$-closed Horn calculus. Excluding the rule $\emptyset \,/\, p_1 \vee p_2 \Rightarrow p_2 \vee p_1$ results in a 1-$\emptyset$-closed Horn calculus, whose derivability problem can be decided in linear time. The linear time algorithm for **P** from [6] is also an instance of this method.

6 *Next* Operators

In this section we extend the framework to accommodate *Next* operators, that are often employed in temporal logics. In primal infon logic [11], they play the role of quotations (see Example 14 below). In what follows, $\circledast$ denotes an arbitrary finite set of unary connectives (*Next* operators), and $\mathcal{L}_\circledast$ denotes the propositional language obtained by augmenting $\mathcal{L}$ with $\circledast$ (we assume that $\diamondsuit_\mathcal{L} \cap \circledast = \emptyset$). A sequence $\bar{*} = *_1 \dots *_m$ ($m \geq 0$) of elements of $\circledast$ is called a $\circledast$*-prefix*. Given a set $\mathcal{F} \subseteq \mathcal{L}_\circledast$ and a $\circledast$-prefix $\bar{*}$, we denote the set $\{\bar{*}\psi \mid \psi \in \mathcal{F}\}$ by $\bar{*}\mathcal{F}$. This notation is extended to sequents and sets of sequents in the obvious way. We now extend pure calculi with new rules for *Next* operators.

Definition 15. A $\circledast$*-proof* in a calculus $\mathbf{G}$ is defined similarly to a standard proof (see Definition 2), where in addition to (*weak*), (*id*) and (*cut*), the following scheme may be used for any $* \in \circledast$:

$$(*i) \qquad \frac{\Gamma \Rightarrow \Delta}{*\Gamma \Rightarrow *\Delta}$$

For an $\mathcal{L}$-calculus $\mathbf{G}$, a set $\mathcal{F} \subseteq \mathcal{L}_\circledast$, and an $\mathcal{L}_\circledast$-sequent s, we write $\vdash^{\mathcal{F}}_{\mathbf{G}_\circledast} s$ (or $\vdash_{\mathbf{G}_\circledast} s$ if $\mathcal{F} = \mathcal{L}_\circledast$) if there is a $\circledast$-proof of s in $\mathbf{G}$ consisting only of $\mathcal{F}$-sequents.

($*i$) is a usual rule for *Next* in the temporal logic LTL (i.e., for $\circledast = \{\mathtt{X}\}$, we have $\vdash_{\mathbf{LK}_\circledast} \Rightarrow \psi$ iff ψ is valid in the *Next*-only fragment of LTL; see, e.g., [16]). It is also used for $\Box$ (and $\Diamond$) in the modal logic $KD!$ of functional Kripke frames.

Remark 3. Applications of $\mathcal{L}$-rules in $\circledast$-proofs may include $\mathcal{L}_\circledast$-formulas. For example, using the rule $\Rightarrow p_2 \,/\, \Rightarrow p_1 \supset p_2$, it is possible to derive the sequent $*p_3 \Rightarrow *p_1 \supset p_2$ from $*p_3 \Rightarrow p_2$.

Example 14. The quotations employed in primal infon logic [11] are unary connectives of the form q `said`, where q ranges over a finite set of principals. If we take $\circledast$ to include these connectives, we have that $\vdash_{\mathbf{P}_\circledast} \Gamma \Rightarrow \psi$ (see Example 3) iff ψ is derivable from Γ in the Hilbert system for primal infon logic given in [11]. This can be shown by induction on the lengths of the proofs.

Next, we define Kripke-style semantics for calculi with *Next* operators.

Definition 16. A *biframe* for $\circledast$ is a tuple $\mathcal{W} = \langle W, \mathcal{R}, \mathcal{V} \rangle$ where:

1. W is a set of elements called *worlds*. Henceforth, we may identify $\mathcal{W}$ with this set (e.g., when writing $w \in \mathcal{W}$ instead of $w \in W$).
2. $\mathcal{R}$ is a function assigning a binary relation on W to every $* \in \circledast$. We write $\mathcal{R}_*$ instead of $\mathcal{R}(*)$, and $\mathcal{R}_*[w]$ denotes the set $\{w' \in W \mid w \mathcal{R}_* w'\}$.
3. $\mathcal{V}$ is a function assigning a bivaluation to every $w \in \mathcal{W}$, such that for every $w \in \mathcal{W}$, $* \in \circledast$ and formula ψ: if $*\psi \in dom(\mathcal{V}(w))$ and $\psi \in dom(\mathcal{V}(w'))$ for every $w' \in \mathcal{R}_*[w]$, then $\mathcal{V}(w)(*\psi) = \min\{\mathcal{V}(w')(\psi) \mid w' \in \mathcal{R}_*[w]\}$. Henceforth, we write $\mathcal{V}_w$ instead of $\mathcal{V}(w)$.

Furthermore, if $dom(\mathcal{V}_w) = \mathcal{F}$ for every $w \in \mathcal{W}$, we refer to $\mathcal{W}$ as an *$\mathcal{F}$-biframe*.

Definition 17. A biframe $\langle W, \mathcal{R}, \mathcal{V} \rangle$ for $\circledast$ is called *functional* if $\mathcal{R}_*$ is a functional relation (that is, a total function from W to W) for every $* \in \circledast$. In this case we write $\mathcal{R}_*(w)$ to denote the unique element $w' \in W$ satisfying $w \mathcal{R}_* w'$.

Definition 18. A biframe $\langle W, \mathcal{R}, \mathcal{V} \rangle$ for $\circledast$ is called **G**-*legal* for an $\mathcal{L}$-calculus **G** if $\mathcal{V}_w$ is **G**-legal for every $w \in W$ (see Definition 9).

Theorem 5 (Soundness and Completeness). *Let* **G** *be an $\mathcal{L}$-calculus, $\mathcal{F}$ be a set of $\mathcal{L}_\circledast$-formulas, and s be an $\mathcal{F}$-sequent. Then, $\vdash^{\mathcal{F}}_{\mathbf{G}_\circledast} s$ iff $\mathcal{V}_w(s) = 1$ for every* **G**-*legal functional $\mathcal{F}$-biframe $\langle W, \mathcal{R}, \mathcal{V} \rangle$ for $\circledast$ and $w \in W$.*

Generally speaking, soundness is proved by induction on the length of the $\circledast$-proof. The fact that the biframes are functional is essential for the soundness of ($*i$). Completeness is proved using a canonical countermodel construction.

Remark 4. Note that similar results hold for usual rules for introducing $\Box$. For example, if we take the usual rule used in the system for the modal logic K (which, unlike ($*i$), allows only one formula on the right-hand side), we can prove soundness and completeness as above with respect to *all* **G**-legal $\mathcal{F}$-biframes. Similarly, for other known sequent rules for $\Box$ (as those of the systems for $K4$,

KB, $S4$, and $S5$, see [18]) it is possible to show a similar general soundness and completeness with respect to $\mathbf{G}$-legal $\mathcal{F}$-biframes satisfying the corresponding condition (transitivity, symmetry, etc.). Nevertheless, the reduction to SAT proposed below applies only for $(*i)$.

Next we extend the reduction from Section 5 to analytic calculi with *Next* operators. This is done for a large family of calculi that we call *standard*.

Definition 19. An atomic variable $p \in At$ is called *lonely* in some rule r if $p \in frm(r)$, but p is not a proper subformula of any formula in $frm(r)$. A calculus is called *standard* if none of its rules has lonely atomic variables.

As before, we use $\{x_\psi \mid \psi \in \mathcal{L}_\circledast\}$ as the set of atomic variables in the SAT instances. Nevertheless, while the reduction above was based on $\circledcirc$-subformulas, the current reduction is based on $\circledcirc$-*local formulas*. This notion generalizes the *local formulas relation* from [15].

Definition 20. $loc^{\circledcirc}(\psi)$, the set of formulas that are $\circledcirc$-*local* to an $\mathcal{L}_\circledast$-formula ψ, is inductively defined as follows: *1)* $loc^{\circledcirc}(p) = \{p\}$ for every atomic variable $p \in At$; *2)* $loc^{\circledcirc}(\diamond(\psi_1, \dots, \psi_n)) = \{\diamond(\psi_1, \dots, \psi_n)\} \cup \{\circ\psi_i \mid \circ \in \circledcirc, 1 \le i \le n\} \cup \bigcup_{i=1}^{n} loc^{\circledcirc}(\psi_i)$ for every $\diamond \in \diamondsuit_{\mathcal{L}}^{n}$ and formulas $\psi_1, \dots, \psi_n$; *3)* $loc^{\circledcirc}(*\psi) = *loc^{\circledcirc}(\psi)$ for every $* \in \circledast$ and formula ψ. This definition is extended to sequents in the obvious way, i.e. $loc^{\circledcirc}(s) = \bigcup\{loc^{\circledcirc}(\varphi) \mid \varphi \in frm(s)\}$.

Note that for $\circledast = \emptyset$, we have $loc^{\circledcirc}(\psi) = sub^{\circledcirc}(\psi)$ for every formula ψ.

Example 15. For $\circledast = \{\flat, \sharp\}$,
$loc^{\{\neg\}}(\flat(\sharp p_1 \supset p_2)) = \{\flat\sharp p_1, \flat\neg\sharp p_1, \flat p_2, \flat\neg p_2, \flat(\sharp p_1 \supset p_2)\}$.

Definition 21. The SAT instance associated with an $\mathcal{L}$-calculus $\mathbf{G}$, an $\mathcal{L}_\circledast$-sequent s and a set $\circledcirc \subseteq \diamondsuit_{\mathcal{L}}^{1}$ is given by:

$$\mathtt{SAT}_{\circledast}^{\circledcirc}(\mathbf{G}, s) := \bigcup \left\{\mathtt{SAT}^{+}(\bar{*}\sigma(s')) \mid \emptyset \,/\, s' \in Ax(\mathbf{G}), \bar{*}\sigma(frm(s')) \subseteq loc^{\circledcirc}(s)\right\}.$$

Theorem 6. *Let $\mathbf{G}$ be a standard $\circledcirc$-analytic $\mathcal{L}$-calculus and s be an $\mathcal{L}_\circledast$-sequent. Then $\vdash_{\mathbf{G}_\circledast} s$ iff $\mathtt{SAT}_{\circledast}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$ is unsatisfiable.*

Generally speaking, the main difficulty in the proof of this theorem is to construct a countermodel for s (in the form of a $\mathbf{G}$-legal functional $\mathcal{L}_\circledast$-biframe for $\circledast$) out of a satisfying assignment u of $\mathtt{SAT}_{\circledast}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$. Thus if $\not\vdash_{\mathbf{G}_\circledast} s$, the full proof of Theorem 6 actually provides a way to translate the classical assignment that satisfies $\mathtt{SAT}_{\circledast}^{\circledcirc}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$ into a countermodel of s. This is done in two steps. First, we translate u into a $\circledcirc$-*closed* $\mathbf{G}$-legal functional biframe $\mathcal{W}$ which is not a model of s:

Definition 22. A set of $\mathcal{L}_\circledast$-formulas is called $\circledcirc$-*closed* if whenever it contains a formula of the form $\diamond(\varphi_1, \dots, \varphi_n)$ (for some $\diamond \in \diamondsuit_{\mathcal{L}}$) it also contains φ_i and $\circ\varphi_i$ for every $1 \le i \le n$ and $\circ \in \circledcirc$. A biframe $\langle W, \mathcal{R}, \mathcal{V}\rangle$ for $\circledast$ is called $\circledcirc$-*closed* if the following hold for every $w \in W$: $dom(\mathcal{V}_w)$ is $\circledcirc$-closed and finite; and for every $* \in \circledast$, if $*\psi \in dom(\mathcal{V}_w)$, then $\psi \in dom(\mathcal{V}_{w'})$ for every $w' \in \mathcal{R}_*[w]$.

Given an assignment u that satisfies $\mathtt{SAT}^{\circledcirc}_{\circledast}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$, a $\circledcirc$-*closed* $\mathbf{G}$-legal functional biframe $\mathcal{W} = \langle W, \mathcal{R}, \mathcal{V} \rangle$ is constructed as follows:

1. W is the set of all $\circledast$-prefixes.
2. For every $* \in \circledast$ and $\bar{*} \in \mathcal{W}$, $\mathcal{R}_*(\bar{*}) = \bar{*}*$.
3. $\mathcal{V}_{\bar{*}}$ is defined by induction on the length of $\bar{*}$: $dom(\mathcal{V}_\epsilon) = loc^{\circledcirc}(s)$ and $\mathcal{V}_\epsilon(\psi) = 1$ iff u satisfies x_ψ;[3] $dom(\mathcal{V}_{*_1 \dots *_n}) = \left\{ \varphi \mid *_n \varphi \in dom(\mathcal{V}_{*_1 \dots *_{n-1}}) \right\}$ and $\mathcal{V}_{*_1 \dots *_n}(\psi) = \mathcal{V}_{*_1 \dots *_{n-1}}(*_n \psi)$.

Then, the following theorem is used to extend $\mathcal{W}$ to a full $\mathbf{G}$-legal $\mathcal{L}_{\circledast}$-biframe.

Definition 23. A biframe $\langle W, \mathcal{R}, \mathcal{V} \rangle$ for $\circledast$ *extends* a biframe $\langle W', \mathcal{R}', \mathcal{V}' \rangle$ for $\circledast$ if $W = W'$, $\mathcal{R} = \mathcal{R}'$, and $\mathcal{V}_w$ extends $\mathcal{V}'_w$ for every $w \in W$.

Theorem 7. *Let* $\mathbf{G}$ *be a standard semantically* $\circledcirc$*-analytic* $\mathcal{L}$*-calculus, and* $\mathcal{W}$ *be a* $\mathbf{G}$*-legal* $\circledcirc$*-closed biframe for* $\circledast$ *with* $dom(\mathcal{V}_w) \subseteq \mathcal{L}_{\circledast}$ *for every* $w \in \mathcal{W}$*. Then* $\mathcal{W}$ *can be extended to a* $\mathbf{G}$*-legal* $\mathcal{L}_{\circledast}$*-biframe for* $\circledast$*.*

For the case that $\circledcirc = \emptyset$, the polynomial time algorithm from Section 5 can be modified to accommodate *Next* operators.

Theorem 8. *Let* $\mathbf{G}$ *be a* k*-*$\emptyset$*-closed* $\mathcal{L}$*-calculus. Given an* $\mathcal{L}_{\circledast}$*-sequent* s*, it is possible to compute* $\mathit{SAT}^{\emptyset}_{\circledast}(\mathbf{G}, s) \cup \mathit{SAT}^{-}(s)$ *in* $O(n^k)$ *time, where* n *is the length of the string representing* s*.*

Proof (sketch). The algorithm from the proof of Theorem 4 is reused with several modifications. As in [11], an auxiliary trie (an ordered tree data structure commonly used for string processing) for $\circledast$-prefixes is constructed in linear time, and every node in the input parse tree has a pointer to a node in this trie. Now each node in the parse tree corresponds to an occurrence of a formula that is $\emptyset$-local to s. The tree is then compressed to a dag as in the proof of Theorem 4. The nodes of the dag one-to-one correspond to the $\emptyset$-local formulas of s. The rest of the algorithm is exactly as in the proof of Theorem 4 with $\circledcirc = \emptyset$. $\square$

For a Horn calculus $\mathbf{G}$, $\mathtt{SAT}^{\circledcirc}_{\circledast}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$ consists of Horn clauses for every sequent s. When $\mathbf{G}$ is 1-$\emptyset$-closed and $\emptyset$-analytic, a linear time decision procedure for the derivability problem for $\mathbf{G}$ with *Next* operators is obtained by applying a HORNSAT solver on $\mathtt{SAT}^{\emptyset}_{\circledast}(\mathbf{G}, s) \cup \mathtt{SAT}^{-}(s)$.

Example 16. Example 13 works as is for the extension of $\mathbf{P}$ or $\mathbf{EP}$ with any finite set of *Next* operators.

Example 17. The linear time fragment of dual-Horn clauses can be utilized as well. For example, consider the ($\emptyset$-analytic) calculus $\mathbf{P}_d$ that consists of the rules $(\vee \Rightarrow)$, $(\Rightarrow \vee)$, $(\wedge \Rightarrow)$ of $\mathbf{LK}$ and the following ones for "dual primal implication":

$(\prec\Rightarrow)\ p_1 \Rightarrow\ /\ p_1 \prec p_2 \Rightarrow \qquad (\Rightarrow\prec)\ \Rightarrow p_1 ; p_2 \Rightarrow\ /\ \Rightarrow p_1 \prec p_2$

For any sequent s, $\mathtt{SAT}^{\circledcirc}_{\circledast}(\mathbf{P}_d, s) \cup \mathtt{SAT}^{-}(s)$ consists of dual-Horn clauses. Thus the derivability problem for $\mathbf{P}_d$ with *Next* operators can be decided in linear time.

---
[3] ϵ denotes the empty $\circledast$-prefix.

6.1 On Analyticity of Pure Calculi with *Next* Operators

At this point, a natural question arises: does the extension of a calculus with Next operators preserve the $\circledcirc$-analyticity of the calculus? In this final section we provide a positive answer to this question, based on Theorem 7 above that was used to prove the correctness of the reduction.

Definition 24. An $\mathcal{L}$-calculus $\mathbf{G}$ is called *$\circledcirc$-analytic with $\circledast$* if $\vdash_{\mathbf{G}_{\circledast}} s$ implies $\vdash^{sub^{\circledcirc}(s)}_{\mathbf{G}_{\circledast}} s$ for every $\mathcal{L}_{\circledast}$-sequent s.

Theorem 9. *A standard $\mathcal{L}$-calculus $\mathbf{G}$ is $\circledcirc$-analytic iff it is $\circledcirc$-analytic with $\circledast$.*

Proof. Suppose that $\mathbf{G}$ is $\circledcirc$-analytic. By Theorem 2 it is also semantically $\circledcirc$-analytic. Let s be an $\mathcal{L}_{\circledast}$-sequent such that $\not\vdash^{sub^{\circledcirc}(s)}_{\mathbf{G}_{\circledast}} s$. By Theorem 5, there exists a $\mathbf{G}$-legal functional $sub^{\circledcirc}(s)$-biframe $\mathcal{W} = \langle W, \mathcal{R}, \mathcal{V} \rangle$ and $w \in \mathcal{W}$ such that $\mathcal{V}_w(s) = 0$. $\mathcal{W}$ is $\circledcirc$-closed, and by Theorem 7, it can be extended to a $\mathbf{G}$-legal functional $\mathcal{L}_{\circledast}$-biframe $\mathcal{W}' = \langle W, \mathcal{R}, \mathcal{V}' \rangle$ for $\circledast$. After this extension, we still have $\mathcal{V}'_w(s) = 0$. Theorem 5 implies that $\not\vdash_{\mathbf{G}_{\circledast}} s$. For the converse, suppose that $\mathbf{G}$ is $\circledcirc$-analytic with $\circledast$. Assume that $\vdash_{\mathbf{G}} s$ for some $\mathcal{L}$-sequent s. Hence, $\vdash_{\mathbf{G}_{\circledast}} s$. Consequently, there is a $\circledast$-proof of s in $\mathbf{G}$ that consists only of $sub^{\circledcirc}(s)$-formulas. This proof cannot contain applications of $(*i)$, and therefore, $\vdash^{sub^{\circledcirc}(s)}_{\mathbf{G}} s$. $\square$

Example 18. Since $\mathbf{P}$ and $\mathbf{EP}$ are $\emptyset$-analytic and standard, they are also $\emptyset$-analytic with $\circledast$. In contrast, the Hilbert system for primal infon logic in [11] admits a similar property that involves local formulas rather than subformulas.

Remark 5. Further to Remark 4, it can be similarly shown that the extension of a standard pure calculus with any usual rule for $\Box$ preserves analyticity. In particular, we did not assume in Theorem 7 that the biframes are functional.

7 Conclusions and Further Research

We have identified a wide family of calculi for which the derivability problem can be solved using off-the-shelf SAT solvers. Our method was presented for pure calculi, and later extended to accommodate *Next* operators. The produced SAT instances do not encode derivations, whose lengths might not be polynomially bounded. Instead, they represent the (non-) existence of polynomially bounded countermodels in the form of partial bivaluations or Kripke frames.

The proposed reduction is limited to analytic pure calculi, as it relies on their straightforward bivaluation semantic presentation. Nevertheless, some of the theoretic developments presented in this paper can be extended to different families of calculi. For example, following Remark 5, the fact that analyticity is preserved when pure calculi are augmented with *Next* operators, holds also for other introduction rules for modalities. Such extensions, as well as studying multi-ary modalities in this context, are left for future work. In addition, we plan to extend the methods of this paper to analytic many-sided sequent calculi, that are more expressive than ordinary two-sided calculi. Finally, it is interesting to study possible applications of logics (besides primal logic) that can be reduced to efficient fragments of SAT (e.g., 2SAT).

References

1. Arnon Avron. Simple consequence relations. *Inf. Comput.*, 92(1):105–139, May 1991.
2. Arnon Avron. Classical gentzen-type methods in propositional many-valued logics. In Melvin Fitting and Ewa Orłowska, editors, *Beyond Two: Theory and Applications of Multiple-Valued Logic*, volume 114 of *Studies in Fuzziness and Soft Computing*, pages 117–155. Physica-Verlag HD, 2003.
3. Arnon Avron, Beata Konikowska, and Anna Zamansky. Modular construction of cut-free sequent calculi for paraconsistent logics. In *Logic in Computer Science (LICS), 2012 27th Annual IEEE Symposium on*, pages 85–94, 2012.
4. Arnon Avron, Beata Konikowska, and Anna Zamansky. Cut-free sequent calculi for c-systems with generalized finite-valued semantics. *Journal of Logic and Computation*, 23(3):517–540, 2013.
5. Arnon Avron and Iddo Lev. Non-deterministic multiple-valued structures. *Journal of Logic and Computation*, 15(3):241–261, 2005.
6. Lev Beklemishev and Yuri Gurevich. Propositional primal logic with disjunction. *Journal of Logic and Computation*, 2012.
7. Jean-Yves Béziau. Sequents and bivaluations. *Logique et Analyse*, 44(176):373–394, 2001.
8. Nikolaj Bjørner, Guido de Caso, and Yuri Gurevich. From primal infon logic with individual variables to datalog. In *Correct Reasoning*, pages 72–86. Springer, 2012.
9. Andreas Blass and Yuri Gurevich. Abstract hilbertian deductive systems, infon logic, and datalog. *Information and Computation*, 231:21–37, 2013.
10. Jiazhen Cai and Robert Paige. Using multiset discrimination to solve language processing problems without hashing. *Theoretical Computer Science*, 145(12):189 – 228, 1995.
11. Carlos Cotrini and Yuri Gurevich. Basic primal infon logic. *Journal of Logic and Computation*, 2013.
12. Anatoli Degtyarev and Andrei Voronkov. The inverse method. *Handbook of Automated Reasoning*, 1:179–272, 2001.
13. William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional horn formulae. *The Journal of Logic Programming*, 1(3):267–284, 1984.
14. Carla P. Gomes, Henry Kautz, Ashish Sabharwal, and Bart Selman. Satisfiability solvers. In *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 89–134. Elsevier, 2008.
15. Yuri Gurevich and Itay Neeman. Logic of infons: The propositional case. *ACM Trans. Comput. Logic*, 12(2):9:1–9:28, January 2011.
16. Hiroya Kawai. Sequential calculus for a first order infinitary temporal logic. *Mathematical Logic Quarterly*, 33(5):423–432, 1987.
17. Robert Kowalski. *Logic for Problem-solving*. North-Holland Publishing Co., Amsterdam, The Netherlands, The Netherlands, 1986.
18. Heinrich Wansing. Sequent systems for modal logics. In Dov M. Gabbay and Franz Guenthner, editors, *Handbook of Philosophical Logic, 2nd edition*, volume 8, pages 61–145. Springer, 2002.