Graph-based decision for Gödel-Dummett logics

Dominique Larchey-Wendling

Graph-based decision for Gödel-Dummett logics

Dominique Larchey-Wendling
LORIA – CNRS
Campus Scientifique, BP 239
54 506 Vandœuvre-lès-Nancy
France

Abstract. We present a graph-based decision procedure for Gödel-Dummett logics and an algorithm to compute counter-models. A formula is transformed into a conditional bi-colored graph in which we detect some specific cycles and alternating chains using matrix computations. From an instance graph containing no such cycle (resp. no $(n+1)$-alternating chain) we extract a counter-model in LC (resp. LC_n).

Keywords: Gödel-Dummett logic, sequent calculus, decision procedures, graphs, counter-models.

1. Introduction

Gödel-Dummett logic LC and its finitary versions $(LC_n)_{n>0}$ are the intermediate logics (between classical and intuitionistic logics) characterized by linear Kripke models. LC was introduced by Gödel in [10] and later axiomatized by Dummett in [7]. It is now one of the most studied intermediate logics and has been recognized as one of the fundamental *t-norm based fuzzy logics* [11]. Recently, the counter-model search problem in LC_n has been characterized as a resource consumption bounding problem for a simple process calculus [14].

Proof-search in LC has benefited from the development of proof-search in intuitionistic logic with two important seeds: the *contraction-free calculus* of Dyckhoff [1, 8, 9] and the *hyper-sequent* calculus of Avron [2, 16]. Two recent contributions propose a similar approach based on a set of *local* and *strongly invertible* proof rules[1] (for either sequent [12] or hyper-sequent [2] calculus,) and a semantic criterion to decide *irreducible (hyper)-sequents* and eventually build a counter-model. The sequents-of-relations calculi also provide a nice framework for proof-search in LC [4] and more generally, in many-valued logics [6].

In one further but nonetheless fundamental step, we have successfully completed the integration of proof- and counter-model search in LC, through the merge of all proof-search branches into a *conditional*

---

[1] A rule is local if it can be applied by changing (hyper-)sequents only locally, for example by decomposing one formula into some of its constituents. A rule is strongly invertible if it preserves counter-models from the premises to the conclusion.

graph. This method was first presented in [13] and the present paper contains a full description of this technique with the corresponding results and their proofs. It also includes the case of the finitary versions LC_n and integrates the finite and infinite cases. Moreover, we compare our system with the sequent calculus of Dyckhoff [9], the hyper-sequent calculi of Avron [2] and Baaz [5] and with the sequent and hyper-sequent of relations calculi [4, 6].

After having introduced the family of Gödel-Dummett logics and their algebraic semantics, we present a proof-search method based on two steps: first, formulæ are *flattened* using an indexing technique and the constant $\perp$ is removed. Then they are decomposed into atomic implications using some *strongly invertible* sequent calculus proof rules.

After the proof-search phase, we obtain a set of *implicational sequents* (which only contain atomic implications). A *bi-colored graph* is associated to each implicational sequent and the sequent is decided in LC (resp. LC_n) by searching for $\Rightarrow$-*cycles* (resp. $(n+1)$-*alternating chains*) in the associated bi-colored graph. A counter-model construction algorithm for implicational sequents is also provided.

Then, we merge all the proof-search branches by attaching a *boolean selector* to some arrows of bi-colored graphs. By instantiation of the selectors, it is possible to recover all the bi-colored graphs corresponding to each proof-search branch. In fact, selectors encode branching in the proof-search space.

Finally, we present the algorithm which directly builds a *conditional bi-colored graph* from a formula of LC, and a method for detecting $\Rightarrow$-cycles (resp. $(n+1)$-alternating chains) in the instances of this conditional bi-colored graph. This method is based on algebraic computations on generalized boolean matrices.

2. The syntax and semantics of LC_n

We present the family of propositional Gödel-Dummett logics LC_n, their algebraic semantics, and some sound (i.e. admissible) sequent calculus proof rules for LC_n.

2.1. SYNTACTIC NOTIONS

LC_n is a family of intermediate logics indexed by a value n which belongs to the set $\overline{I\!N}^{\star} = \{1, 2, \ldots\} \cup \{\infty\}$ of strictly positive natural numbers with its natural order $\leqslant$, augmented with a greatest element ∞. All these logics share the same syntax which is the common syntax to intuitionistic IL and classical CL propositional logics. Of course, the set of universally valid formulæ varies, depending on the value of n.

The set of *propositional formulæ*, denoted Form, is defined inductively, starting from a set of propositional *variables* denoted by Var with an additional bottom constant $\perp$ denoting *absurdity* and using the connectives $\wedge$, $\vee$ and $\supset$.

A *substitution*, denoted by σ, is any function that associates a formula to every propositional variable. We denote by A_σ the result of the application of the substitution σ to the atoms in A. IL will denote the set of formulæ that are provable in any intuitionistic propositional calculus (see [8]) and CL will denote the classically valid formulæ. As usual an *intermediate propositional logic* [1] is a set of formulæ $\mathcal{L}$ satisfying $\mathsf{IL} \subseteq \mathcal{L} \subseteq \mathsf{CL}$ and closed under the rule of modus ponens (if $A \in \mathcal{L}$ and $A \supset B \in \mathcal{L}$ then $B \in \mathcal{L}$) and under an arbitrary substitution of variables (if $A \in \mathcal{L}$ and σ is any substitution then $A_\sigma \in \mathcal{L}$.)

Among the elements of Var, we distinguish two variables $\square$ and $\Diamond$ which play a particular role during the proof-search process. These variables should not be used for another purpose, so we *require* that the formulæ which would be decided by the method described in this paper should be built from atoms in $\{\perp\} \cup (\mathsf{Var} - \{\square, \Diamond\})$. Up to the renaming of variables, this does not restrict the scope of the method.

A formula is *implicational* if it is of the form $X \supset Y$ with $X, Y \in \mathsf{Var}$. A formula is *flat* if it is either implicational or of the form $X \supset (Y \circledast Z)$ or $(X \circledast Y) \supset Z$ with $X, Y, Z \in \mathsf{Var}$ and $\circledast \in \{\wedge, \vee, \supset\}$.

A *context*, denoted Γ or Δ, is a multiset of formulæ. A *sequent* is a pair of contexts denoted $\Gamma \vdash \Delta$. A sequent is *implicational* if it contains only implicational formulæ.

DEFINITION 1 ($\Diamond$-contexts and flat sequents).
A $\Diamond$-context $\Delta_\Diamond$ is a non-empty context which contains only implications (not necessarily atomic implications) and such that for any $A \supset B$ occurring in $\Delta_\Diamond$ the formula $\Diamond \supset B$ also occurs in $\Delta_\Diamond$.

$\Gamma \vdash \Delta_\Diamond$ is a flat sequent if the context Γ contains only flat formulæ and $\Delta_\Diamond$ is a $\Diamond$-context which contains only implicational formulæ.

For example, $\{X \supset Y, \Diamond \supset Y\}$ and $\{\Diamond \supset X\}$ are $\Diamond$-contexts but $\{X \vee Y\}$ or $\{X \supset Y, \Diamond \supset X\}$ are not $\Diamond$-contexts. As a $\Diamond$-context is non-empty, it necessarily contains at least one formula of the form $\Diamond \supset F$ for some formula F.[2] The notions of $\Diamond$-context or flat sequent might seem obscure for the moment but their use is explained in details in section 4.

---

[2] This property is essential in the proof of Proposition 3.

2.2. ALGEBRAIC SEMANTICS

For any $n \in \overline{I\!N}^{\star}$ the Gödel-Dummett logic LC_n is an intermediate logic.[3] LC_n can be semantically characterized by the linear Kripke models of size smaller that n [7]. If we identify the logic LC_n and the set of its valid formulæ, the following strictly decreasing sequence holds:

$$\mathsf{CL} = \mathsf{LC}_1 \supset \cdots \supset \mathsf{LC}_n \supset \cdots \supset \mathsf{LC}_\infty = \mathsf{LC} \supset \mathsf{IL}$$

In the particular case of LC, the logic has a simple Hilbert axiomatic system: $(X \supset Y) \vee (Y \supset X)$ added to the axioms of IL. In this paper, we use the algebraic semantics characterization of LC_n [2] rather than the Kripke semantics.

Let us fix a particular $n \in \overline{I\!N}^{\star}$. The algebraic model is the set $\overline{[0,n)} = [0,n[\cup\{\infty\}$ composed of $n+1$ elements, with the convention $\overline{[0,\infty)} = I\!N \cup \{\infty\}$. With our particular representation, the algebraic models $\overline{[0,n)}$ form a strictly increasing sequence of subsets of $\overline{[0,\infty)}$.

An *interpretation* of propositional variables $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ is inductively extended to formulæ: $\bot$ interpreted by 0, the conjunction $\wedge$ is interpreted by the *minimum* function denoted $\wedge$, the disjunction $\vee$ by the *maximum* function $\vee$ and the implication $\supset$ by the operator $\rightarrowtail$ defined by $a \rightarrowtail b = $ if $a \leqslant b$ then ∞ else b. So the following identities hold:

$$[\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!] \qquad\qquad [\![\bot]\!] = 0$$
$$[\![A \vee B]\!] = [\![A]\!] \vee [\![B]\!] \qquad [\![A \supset B]\!] = [\![A]\!] \rightarrowtail [\![B]\!]$$

We also point out the following property: for any $a, b, \alpha \in \overline{[0,n)}$, if $a \rightarrowtail b < \alpha$ holds then $a \rightarrowtail b = \alpha \rightarrowtail b = b$ holds. Albeit trivial, it is nonetheless very useful, as exemplified in the proofs of Proposition 3 and Theorem 3.

PROPOSITION 1. *Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ be an interpretation and A, A', B, B' be formulæ. If both $[\![A]\!] \geqslant [\![B]\!]$ and $[\![A]\!]' \leqslant [\![B]\!]'$ hold then $[\![A \supset A']\!] \leqslant [\![B \supset B']\!]$ holds.*

The proof of this result is trivial with the previous definition of $\rightarrowtail$. Given an interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ and a value $\alpha \in [0,n[$, we define the *translated interpretation* $[\![\cdot]\!]_{-\alpha}$ the following way:

$$[\![X]\!]_{-\alpha} = \begin{cases} \infty & \text{if } [\![X]\!] = \infty \\ [\![X]\!] - \alpha & \text{if } [\![X]\!] \geqslant \alpha \\ 0 & \text{if } [\![X]\!] < \alpha \end{cases}$$

---

[3] The logic LC_∞, also denoted by LC, is the usual Gödel-Dummett logic.

PROPOSITION 2. *Let A be a formula not containing the constant $\perp$, α a value in $\overline{[0, n)}$ and $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ an interpretation such that for any variable X occurring in A the inequality $[\![X]\!] \geqslant \alpha$ holds. The two following properties are satisfied:*

$$1) \ \ [\![A]\!] \geqslant \alpha \qquad 2) \ \ \textit{if } \alpha \neq \infty \textit{ then } [\![A]\!]_{-\alpha} = [\![A]\!] - \alpha$$

Proof. These two properties are proved by induction on the structure of the formula A. For any $a, b \in \overline{[0, n)}$ and $\circledast \in \{\wedge, \vee, \rightarrow\}$, if $a \geqslant \alpha$ and $b \geqslant \alpha$ hold, then $a \circledast b \geqslant \alpha$ holds. Moreover the semantic interpretation $\wedge = \min$ (resp. $\vee = \max$ and $\rightarrow$) of the connective $\wedge$ (resp. $\vee$ and $\supset$) commutes with the $(\cdot) - \alpha$ operation (with $\alpha \neq \infty$). $\blacksquare$

A *model* of a formula A in LC_n is an interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ such that $[\![A]\!] = \infty$. A *counter-model* in LC_n is an interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ such that $[\![A]\!] < \infty$. A formula is (universally) *valid* if it has no counter-model. From now, we will always use the term valid implying universal validity.

The two contexts $\Gamma \equiv A_1, \ldots, A_p$ and $\Delta \equiv B_1, \ldots, B_q$ of the sequent $\Gamma \vdash \Delta$ are interpreted the following way:

$$\begin{array}{ll} \lfloor\!\lfloor\Gamma\rfloor\!\rfloor = [\![A_1]\!] \wedge \cdots \wedge [\![A_p]\!] & \lfloor\!\lfloor\emptyset\rfloor\!\rfloor = \infty \\ \lceil\!\lceil\Delta\rceil\!\rceil = [\![B_1]\!] \vee \cdots \vee [\![B_q]\!] & \lceil\!\lceil\emptyset\rceil\!\rceil = 0 \end{array}$$

An interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ is a *model* of the sequent $\Gamma \vdash \Delta$ in LC_n if $\lfloor\!\lfloor\Gamma\rfloor\!\rfloor \leqslant \lceil\!\lceil\Delta\rceil\!\rceil$ holds. On the other hand, it is a *counter-model* to this sequent in LC_n if $\lceil\!\lceil\Delta\rceil\!\rceil < \lfloor\!\lfloor\Gamma\rfloor\!\rfloor$ holds. In particular, if $p > 0$ and $q > 0$, for any i and j the inequality $[\![B_j]\!] < [\![A_i]\!]$ holds. Moreover, the sequent $\vdash A$ and the formula A have the same models (and counter-models). A sequent is *valid* if it has no counter-model.

2.3. SEMANTICAL RESULTS

We introduce a semantical result which relates the interpretation of $\Diamond$-contexts and of the variable $\Diamond$: in the case of counter-models, the variable $\Diamond$ might play the role of the ∞ semantic value.

PROPOSITION 3. *Let $\Delta_\Diamond$ be a $\Diamond$-context and $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ an interpretation of variables such that $\lceil\!\lceil\Delta_\Diamond\rceil\!\rceil < \infty$. Then we have $\lceil\!\lceil\Delta_\Diamond\rceil\!\rceil < [\![\Diamond]\!]$.*

Proof. As $\Delta_\Diamond$ is composed of implications and is not empty, there exists $A \supset B \in \Delta_\Diamond$ such that $\lceil\!\lceil\Delta_\Diamond\rceil\!\rceil = [\![A \supset B]\!]$. Also $\Diamond \supset B \in \Delta_\Diamond$. Thus $[\![\Diamond \supset B]\!] \leqslant \lceil\!\lceil\Delta_\Diamond\rceil\!\rceil = [\![A \supset B]\!] < \infty$ holds. We deduce $[\![A \supset B]\!] = [\![B]\!]$ and $[\![\Diamond]\!] > [\![B]\!]$. Therefore $\lceil\!\lceil\Delta_\Diamond\rceil\!\rceil = [\![A \supset B]\!] = [\![B]\!] < [\![\Diamond]\!]$. $\blacksquare$

We present some logical equivalences useful for designing rules that are suitable for reducing flat formulæ into implicational ones. The notation $A \simeq_n B$ means that A and B always have the same semantic value, i.e. for any interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ we have $[\![A]\!] = [\![B]\!]$.

PROPOSITION 4. *The following semantical equivalences hold in* LC_n:

$$1) \quad (A \wedge B) \supset C \simeq_n (A \supset C) \vee (B \supset C)$$
$$2) \quad (A \vee B) \supset C \simeq_n (A \supset C) \wedge (B \supset C)$$

$$3) \quad A \supset (B \wedge C) \simeq_n (A \supset B) \wedge (A \supset C)$$
$$4) \quad A \supset (B \vee C) \simeq_n (A \supset B) \vee (A \supset C)$$
$$5) \quad A \supset (B \supset C) \simeq_n (A \supset C) \vee (B \supset C)$$

$$6) \quad A \wedge B \simeq_n A \wedge (A \supset B)$$

Proof. We present the proof of case 1). Similar arguments work for cases 2) to 5). We distinguish between $[\![A]\!] \leqslant [\![B]\!]$ and $[\![A]\!] \geqslant [\![B]\!]$. If $[\![A]\!] \leqslant [\![B]\!]$ then $[\![(A \wedge B) \supset C]\!] = ([\![A]\!] \wedge [\![B]\!]) \rightarrow [\![C]\!] = [\![A]\!] \rightarrow [\![C]\!]$. By Proposition 1, $[\![A \supset C]\!] \geqslant [\![B \supset C]\!]$ holds. Then $[\![(A \supset C) \vee (B \supset C)]\!] = [\![A \supset C]\!] \vee [\![B \supset C]\!] = [\![A \supset C]\!]$. Thus $[\![(A \wedge B) \supset C]\!] = [\![(A \supset C) \vee (B \supset C)]\!]$ holds. The handling of case $[\![A]\!] \geqslant [\![B]\!]$ is similar.

We now present the proof of case 6). If $[\![A]\!] \leqslant [\![B]\!]$ holds, then $[\![A \supset B]\!] = [\![A]\!] \rightarrow [\![B]\!] = \infty$ and we compute $[\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!] = [\![A]\!] = [\![A]\!] \wedge \infty = [\![A]\!] \wedge [\![A \supset B]\!] = [\![A \wedge (A \supset B)]\!]$. If $[\![A]\!] > [\![B]\!]$ holds, then $[\![A \supset B]\!] = [\![B]\!]$ and $[\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!] = [\![A]\!] \wedge [\![A \supset B]\!] = [\![A \wedge (A \supset B)]\!]$.

2.4. PROOF RULES AND RELATED NOTIONS

A *proof rule of arity k* is a set of writings of the form : $\dfrac{H_1 \quad \ldots \quad H_k}{C}$ which elements are called *instances*. The $H_1, \ldots, H_k$ are the *premises* and C is the conclusion. In our case, the premises and the conclusion are sequents and we will only use axioms ($k = 0$), unary ($k = 1$) and binary ($k = 2$) proof rules.

A proof rule is *sound* in LC_n if for any instance of the rule, the validity of all the H_i in LC_n implies the validity of C in LC_n. It is *strongly sound* in LC_n if for any instance of the rule and interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$, if $[\![\cdot]\!]$ is a model of all the H_i, then it also a model of C. It is clear than strong soundness implies soundness.

A proof rule is *invertible* in LC_n if for any instance of the rule, the invalidity of at least one of the H_i in LC_n implies the invalidity of C in LC_n. It is *strongly invertible* in LC_n if for any instance of the rule and

interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$, if $[\![\cdot]\!]$ is a counter-model of at least one of the H_i, then it also a counter-model of C. It is also clear than strong invertibility implies invertibility. The difference between the two is that (simple) invertibility does not necessarily preserve counter-models.

PROPOSITION 5. *These rules are strongly sound for LC_n:*

$$1)\ \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\ [\mathrm{Weak}_L] \qquad\qquad 4)\ \frac{\Gamma, A \vdash B \qquad \Delta, A' \vdash B'}{\Gamma, \Delta, A \wedge A' \vdash B \wedge B'}\ [\wedge_M]$$

$$2)\ \frac{\Gamma, A \vdash \Delta \qquad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ [\mathrm{Cut}] \qquad 5)\ \frac{\Gamma, A \vdash B \qquad \Delta, A' \vdash B'}{\Gamma, \Delta, A \vee A' \vdash B \vee B'}\ [\vee_M]$$

$$3)\ \frac{\Gamma A, B \vdash \Delta}{\Gamma, A, A \supset B \vdash \Delta}\ [\supset_L^1] \qquad 6)\ \frac{\Gamma, B \vdash A \qquad \Delta, A' \vdash B'}{\Gamma, \Delta, A \supset A' \vdash B \supset B'}\ [\supset_M]$$

Proof. We present the proof of strong soundness for rules 2), 3) and 6). The case of rule 1) is trivial and the cases of rules 4) and 5) are similar to that of 6).

Let us start with rule 2). Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ be a model of $\Gamma, A \vdash \Delta$ and of $\Gamma' \vdash A, \Delta'$. Then the inequalities $\lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil$ and $\lfloor\!\lfloor\Gamma'\rfloor\!\rfloor \leqslant [\![A]\!] \vee \lceil\!\lceil\Delta'\rceil\!\rceil$ hold. We compute $\lfloor\!\lfloor\Gamma, \Gamma'\rfloor\!\rfloor = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge \lfloor\!\lfloor\Gamma'\rfloor\!\rfloor \leqslant \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge ([\![A]\!] \vee \lceil\!\lceil\Delta'\rceil\!\rceil) = (\lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A]\!]) \vee (\lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge \lceil\!\lceil\Delta'\rceil\!\rceil) \leqslant \lceil\!\lceil\Delta\rceil\!\rceil \vee \lceil\!\lceil\Delta'\rceil\!\rceil = \lceil\!\lceil\Delta, \Delta'\rceil\!\rceil$. Thus $\lfloor\!\lfloor\Gamma, \Gamma'\rfloor\!\rfloor \leqslant \lceil\!\lceil\Delta, \Delta'\rceil\!\rceil$ holds.

Now we consider rule 3).We compute using the equivalence 6) of Proposition 4: $\lfloor\!\lfloor\Gamma, A, A \supset B\rfloor\!\rfloor = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A]\!] \wedge [\![A \supset B]\!] = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A \wedge (A \supset B)]\!] = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A \wedge B]\!] = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![A]\!] \wedge [\![B]\!] = \lfloor\!\lfloor\Gamma, A, B\rfloor\!\rfloor$. Thus the premise and the conclusion have the same models.

We now consider rule 6). Suppose $\lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![B]\!] \leqslant [\![A]\!]$ and $\lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![A']\!] \leqslant [\![B']\!]$ both hold. We distinguish between three cases: a) $[\![B]\!] \leqslant [\![B']\!]$, b) $[\![B]\!] > [\![B']\!]$ and $[\![A]\!] \leqslant [\![A']\!]$, and c) $[\![B]\!] > [\![B']\!]$ and $[\![A]\!] > [\![A']\!]$.

a) $\lfloor\!\lfloor\Gamma, \Delta, A \supset A'\rfloor\!\rfloor \leqslant \infty = [\![B \supset B']\!]$ holds;

b) $\lfloor\!\lfloor\Gamma, \Delta\rfloor\!\rfloor \wedge [\![B]\!] = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge \lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![B]\!] \leqslant \lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![A]\!] \leqslant \lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![A']\!] \leqslant [\![B']\!]$. Thus $\lfloor\!\lfloor\Gamma, \Delta\rfloor\!\rfloor \wedge [\![B]\!] \leqslant [\![B']\!]$ holds and since $[\![B]\!] > [\![B']\!]$, and then, $\lfloor\!\lfloor\Gamma, \Delta\rfloor\!\rfloor \leqslant [\![B']\!]$ holds. We compute $\lfloor\!\lfloor\Gamma, \Delta\rfloor\!\rfloor \wedge [\![A \supset A']\!] \leqslant \lfloor\!\lfloor\Gamma, \Delta\rfloor\!\rfloor \leqslant [\![B']\!] = [\![B \supset B']\!]$. Thus $\lfloor\!\lfloor\Gamma, \Delta, A \supset A'\rfloor\!\rfloor \leqslant [\![B \supset B']\!]$ holds;

c) we compute $\lfloor\!\lfloor\Gamma, \Delta, A \supset A'\rfloor\!\rfloor = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge \lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![A \supset A']\!] = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge \lfloor\!\lfloor\Delta\rfloor\!\rfloor \wedge [\![A']\!] \leqslant \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \wedge [\![B']\!] \leqslant [\![B']\!] = [\![B \supset B']\!]$. Thus $\lfloor\!\lfloor\Gamma, \Delta, A \supset A'\rfloor\!\rfloor \leqslant [\![B \supset B']\!]$ holds.

3. Flattening and the elimination of the constant

We consider a formula D. We want to reduce it into an "equivalent" flat sequent $\mathcal{S}$ such that D is valid if and only if $\mathcal{S}$ is valid and the counter-models of D and $\mathcal{S}$ are related in a sense described in Theorem 2.

We recall that D should not contain occurrences of the special variables $\square$ and $\Diamond$ but it may contain occurrences of the constant $\bot$. So in the flattening process, we also remove occurrences of $\bot$.

We start by indexing the occurrences of subformulæ of D (i.e. the subtrees of the decomposition tree of D). Thus we define a finite mapping $K \mapsto \mathcal{X}_K$: the index $\mathcal{X}_K$ of a subformula occurrence K should be *unique*, i.e. it should not collide with either the variables of D or the two special variables $\square$ and $\Diamond$ or the index of another occurrence of a subformula of D.

We define two multisets δ_K^+ and δ_K^- by mutual structural induction on the occurrence of the subformula K of D:

$$
\begin{aligned}
\delta_\bot^+ &= \mathcal{X}_\bot \supset \square \\
\delta_V^+ &= \mathcal{X}_V \supset V, \square \supset V \text{ when } V \text{ is a variable occurring in } D \\
\delta_{A \circledast B}^+ &= \delta_A^+, \delta_B^+, \mathcal{X}_{A \circledast B} \supset (\mathcal{X}_A \circledast \mathcal{X}_B) \text{ when } \circledast \in \{\wedge, \vee\} \\
\delta_{A \supset B}^+ &= \delta_A^-, \delta_B^+, \mathcal{X}_{A \supset B} \supset (\mathcal{X}_A \supset \mathcal{X}_B) \\[4pt]
\delta_\bot^- &= \square \supset \mathcal{X}_\bot \\
\delta_V^- &= V \supset \mathcal{X}_V, \square \supset V \text{ when } V \text{ is a variable occurring in } D \\
\delta_{A \circledast B}^- &= \delta_A^-, \delta_B^-, (\mathcal{X}_A \circledast \mathcal{X}_B) \supset \mathcal{X}_{A \circledast B} \text{ when } \circledast \in \{\wedge, \vee\} \\
\delta_{A \supset B}^- &= \delta_A^+, \delta_B^-, (\mathcal{X}_A \supset \mathcal{X}_B) \supset \mathcal{X}_{A \supset B}
\end{aligned}
$$

PROPOSITION 6. *The two following properties are satisfied:*

1) *For any subformula occurrence K of D, the formulæ occurring in δ_K^+ and δ_K^- are flat and do not contain the constant $\bot$;*

2) *For any variable V occurring in D, the atomic implication $\square \supset V$ occurs in both δ_D^- and δ_D^+;*

3) *The size of δ_D^- and δ_D^+ is linear in the size of D.*

The proofs of these results are trivial.[4] Let σ be the syntactic substitution $\sigma = \{\bot \mapsto \square\}$, we denote $K_\square$ the result K_σ of the substitution of $\bot$ by $\square$ in K.

PROPOSITION 7. *For any subformula occurrence K of D, the two sequents $\delta_K^+, \mathcal{X}_K \vdash K_\square$ and $\delta_K^-, K_\square \vdash \mathcal{X}_K$ are valid in LC_n.*

---

[4] Two remarks about property 3) of Proposition 6: the size of formulæ and multisets is measured by the number of symbols. Moreover, property 3) is cited for completeness only as indexation is just an intermediate step in our method.

Proof. We prove this result by mutual structural induction on an occurrence K of a subformula of D. Suppose K is an occurrence of $\bot$ then the two sequents look like $\mathcal{X}_\bot \supset \square, \mathcal{X}_\bot \vdash \square$ and $\square \supset \mathcal{X}_\bot, \square \vdash \mathcal{X}_\bot$ which are both valid, because of equivalence 6) of Proposition 4. If K is an occurrence of a variable V, then we obtain $\mathcal{X}_V \supset V, \square \supset V, \mathcal{X}_V \vdash V$ and $V \supset \mathcal{X}_V, \square \supset V, V \vdash \mathcal{X}_V$ which again are both valid.

Now we consider the case $K \equiv A \wedge B$. By induction, the four sequents $\delta_A^+, \mathcal{X}_A \vdash A_\square$ and $\delta_A^-, A_\square \vdash \mathcal{X}_A$ and $\delta_B^+, \mathcal{X}_B \vdash B_\square$ and $\delta_B^-, B_\square \vdash \mathcal{X}_B$ are valid. Let us consider the proof tree:

$$\cfrac{\cfrac{\delta_A^+, \mathcal{X}_A \vdash A_\square \quad \delta_B^+, \mathcal{X}_B \vdash B_\square}{\cfrac{\delta_A^+, \delta_B^+, \mathcal{X}_A \wedge \mathcal{X}_B \vdash A_\square \wedge B_\square}{\cfrac{\delta_A^+, \delta_B^+, \mathcal{X}_A \wedge \mathcal{X}_B, \mathcal{X}_{A \wedge B} \vdash A_\square \wedge B_\square}{\delta_A^+, \delta_B^+, \mathcal{X}_{A \wedge B} \supset (\mathcal{X}_A \wedge \mathcal{X}_B), \mathcal{X}_{A \wedge B} \vdash A_\square \wedge B_\square} [\supset_L^1]} [\text{Weak}_L]} [\wedge_M]}$$

The last sequent (at the root of the proof tree) is identical to $\delta_K^+, \mathcal{X}_K \vdash K_\square$. Thus, since the leaves of the proof tree are valid sequents, so is its root. Let us consider this other proof tree:

$$\cfrac{\cfrac{\cfrac{\mathcal{X}_{A \wedge B} \vdash \mathcal{X}_{A \wedge B}}{\cfrac{\mathcal{X}_A \wedge \mathcal{X}_B, \mathcal{X}_{A \wedge B} \vdash \mathcal{X}_{A \wedge B}}{\mathcal{X}_A \wedge \mathcal{X}_B, (\mathcal{X}_A \wedge \mathcal{X}_B) \supset \mathcal{X}_{A \wedge B} \vdash \mathcal{X}_{A \wedge B}} [\supset_L^1]} [\text{Weak}_L] \quad \cfrac{\delta_A^-, A_\square \vdash \mathcal{X}_A \quad \delta_B^-, B_\square \vdash \mathcal{X}_B}{\delta_A^-, \delta_B^-, A_\square \wedge B_\square \vdash \mathcal{X}_A \wedge \mathcal{X}_B} [\wedge_M]}{\delta_A^-, \delta_B^-, (\mathcal{X}_A \wedge \mathcal{X}_B) \supset \mathcal{X}_{A \wedge B}, A_\square \wedge B_\square \vdash \mathcal{X}_{A \wedge B}} [\text{Cut}]}$$

The last (root) sequent is identical to $\delta_K^-, K_\square \vdash \mathcal{X}_K$. The three leaf sequents of this proof tree are valid, two by induction hypothesis and the last $\mathcal{X}_{A \wedge B} \vdash \mathcal{X}_{A \wedge B}$ is also obviously valid. Thus the root sequent $\delta_K^-, K_\square \vdash \mathcal{X}_K$ is valid.

The cases $K \equiv A \vee B$ and $K \equiv A \supset B$ can be handled with similar arguments.

PROPOSITION 8. *If $[\![\cdot]\!]$ is such that $[\![\square]\!] = 0$ and $[\![\mathcal{X}_K]\!] = [\![K]\!]$ for any occurrence of subformula K of D, then $\lfloor\!\lfloor \delta_D^- \rfloor\!\rfloor = \lfloor\!\lfloor \delta_D^+ \rfloor\!\rfloor = \infty$ holds.*

Proof. The formulæ occurring in δ_D^+ or δ_D^- are either implicational of the forms $\mathcal{X}_\bot \supset \square, \square \supset \mathcal{X}_\bot, \mathcal{X}_V \supset V, V \supset \mathcal{X}_V$ and $\square \supset V$ or else of the forms $\mathcal{X}_{A \circledast B} \supset (\mathcal{X}_A \circledast \mathcal{X}_B)$ and $(\mathcal{X}_A \circledast \mathcal{X}_B) \supset \mathcal{X}_{A \circledast B}$ for $\circledast \in \{\wedge, \vee, \supset\}$.

As $[\![\mathcal{X}_\bot]\!] = [\![\bot]\!] = 0 = [\![\square]\!]$ holds, then $[\![\mathcal{X}_\bot \supset \square]\!] = [\![\square \supset \mathcal{X}_\bot]\!] = \infty$ holds. As $[\![\mathcal{X}_V]\!] = [\![V]\!]$ holds, $[\![\mathcal{X}_V \supset V]\!] = [\![V \supset \mathcal{X}_V]\!] = \infty$ also holds. As $[\![\square]\!] = 0 \leqslant [\![V]\!]$ holds, $[\![\square \supset V]\!] = \infty$ holds too.

We compute $[\![\mathcal{X}_A \circledast \mathcal{X}_B]\!] = [\![\mathcal{X}_A]\!] \circledast' [\![\mathcal{X}_B]\!] = [\![A]\!] \circledast' [\![B]\!]$ and $[\![\mathcal{X}_{A \circledast B}]\!] = [\![A \circledast B]\!] = [\![A]\!] \circledast' [\![B]\!]$ for $(\circledast, \circledast')$ being either $(\wedge, \wedge)$, $(\vee, \vee)$ or $(\supset, \twoheadrightarrow)$. In either case, $[\![\mathcal{X}_A \circledast \mathcal{X}_B]\!] = [\![\mathcal{X}_{A \circledast B}]\!]$ holds and thus we have $[\![(\mathcal{X}_A \circledast \mathcal{X}_B) \supset \mathcal{X}_{A \circledast B}]\!] = [\![\mathcal{X}_{A \circledast B} \supset (\mathcal{X}_A \circledast \mathcal{X}_B)]\!] = \infty$.

THEOREM 1. *If the sequent $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D$ is valid in LC_n then the formula D is valid in LC_n.*

Proof. Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ be an interpretation. We define a new interpretation $[\![\cdot]\!]'$ by $[\![V]\!]' = [\![V]\!]$ for any V variable occurring in D, $[\![\mathcal{X}_K]\!]' = [\![K]\!]$ for any K subformula occurring in D, $[\![\Diamond]\!]' = \infty$ and $[\![\Box]\!]' = 0$. Then since $[\![\cdot]\!]'$ and $[\![\cdot]\!]$ have identical values on the atoms of D, for any subformula K of D, they also have identical values on the atoms of K and thus the identity $[\![K]\!]' = [\![K]\!]$ holds. So $[\![\mathcal{X}_K]\!]' = [\![K]\!]'$ and $[\![\Box]\!]' = 0$ both hold. By Proposition 8, we obtain $\lfloor\!|\delta_D^-|\!\rfloor' = \infty$. Since $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D$ is valid in LC_n, $[\![\cdot]\!]'$ is a model of this sequent and then $[\![\Diamond \supset \mathcal{X}_D]\!]' = \infty$ holds. But $[\![\Diamond]\!]' = \infty$ and thus $[\![\mathcal{X}_D]\!]' = \infty$. Finally, $[\![D]\!] = [\![\mathcal{X}_D]\!]' = \infty$. Since we have proved that $[\![D]\!] = \infty$ for any interpretation $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$, then D is valid in LC_n.

THEOREM 2. *If $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ is a counter-model of the sequent $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D$ in LC_n then $[\![\Box]\!] < \infty$ and for $\alpha = [\![\Box]\!]$, the translated interpretation $[\![\cdot]\!]_{-\alpha}$ is a counter-model of D in LC_n.*

Proof. Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0, n)}$ be a counter-model of $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D$. Thus $[\![\Diamond \supset \mathcal{X}_D]\!] < \lfloor\!|\delta_D^-|\!\rfloor$ holds and we deduce $[\![\Diamond \supset \mathcal{X}_D]\!] < \infty$ and $[\![\mathcal{X}_D]\!] = [\![\Diamond]\!] \rightarrowtail [\![\mathcal{X}_D]\!] < \infty$. Thus $[\![\mathcal{X}_D]\!] < \lfloor\!|\delta_D^-|\!\rfloor$. By Proposition 7, $\delta_D^-, D_\Box \vdash \mathcal{X}_D$ is a valid sequent so we compute $\lfloor\!|\delta_D^-|\!\rfloor \wedge [\![D_\Box]\!] \leqslant [\![\mathcal{X}_D]\!] < \lfloor\!|\delta_D^-|\!\rfloor$. Thus $\lfloor\!|\delta_D^-|\!\rfloor \wedge [\![D_\Box]\!] < \lfloor\!|\delta_D^-|\!\rfloor$ holds and so $[\![D_\Box]\!] < \lfloor\!|\delta_D^-|\!\rfloor$.

We show that for any variable V of D, the inequality $[\![V]\!] \geqslant [\![\Box]\!]$ holds. First, if D contains no variable at all (i.e. its only atoms are occurrences of the constant $\bot$) then the previously stated property is trivially verified. Otherwise, we consider a variable V_0 of D which realizes the minimum value γ among the non-empty set $\{[\![\Box \supset V]\!] \mid V \text{ variable of } D\}$. Thus $\gamma = [\![\Box \supset V_0]\!]$ and for any variable V of D, $[\![\Box \supset V]\!] \geqslant \gamma$. Since V_0 is a variable of D, by Proposition 6, the formula $\Box \supset V_0$ occurs in δ_D^- and thus $\lfloor\!|\delta_D^-|\!\rfloor \leqslant [\![\Box \supset V_0]\!] = \gamma$.

We show that $[\![\Box]\!] \leqslant \gamma$ by contradiction. Suppose $[\![\Box]\!] > \gamma = [\![\Box \supset V_0]\!]$. Let V be a variable of $D_\Box$: either $V \equiv \Box$ and $[\![V]\!] = [\![\Box]\!] > \gamma$ or V is a variable of D and $[\![\Box \supset V]\!] \geqslant \gamma$, thus, necessarily $[\![V]\!] \geqslant \gamma$ holds. In either case, $[\![V]\!] \geqslant \gamma$. By Proposition 2.1, since $D_\Box$ does not contain $\bot$, we have $[\![D_\Box]\!] \geqslant \gamma$. We obtain $\gamma \leqslant [\![D_\Box]\!] < \lfloor\!|\delta_D^-|\!\rfloor \leqslant [\![\Box \supset V_0]\!] = \gamma$. So we get a contradiction. Therefore $[\![\Box]\!] \leqslant \gamma$ holds. For any variable V of D, $[\![\Box]\!] \leqslant \gamma \leqslant [\![\Box \supset V]\!] = [\![\Box]\!] \rightarrowtail [\![V]\!]$ holds. Thus necessarily, for any variable V of D, $[\![\Box]\!] \leqslant [\![V]\!]$.

As a consequence, by Proposition 2.1, $[\![D_\Box]\!] \geqslant [\![\Box]\!]$ holds.[5] But $[\![D_\Box]\!] < \lfloor\!|\delta_D^-|\!\rfloor$ and then $[\![\Box]\!] < \infty$ holds. Let $\alpha = [\![\Box]\!]$. Since $[\![\Box]\!]_{-\alpha} =$

---

[5] $D_\Box$ does not contain the constant $\bot$. Any variable X of $D_\Box$ is either $\Box$ or a variable of D. And $[\![X]\!] \geqslant [\![\Box]\!]$ holds both when X is variable of D, or when $X \equiv \Box$.

$$\frac{\Gamma, A \supset C \vdash \Delta_\diamond \quad \Gamma, B \supset C \vdash \Delta_\diamond}{\Gamma, (A \wedge B) \supset C \vdash \Delta_\diamond} \; [_\wedge \supset] \quad \frac{\Gamma, A \supset B, A \supset C \vdash \Delta_\diamond}{\Gamma, A \supset (B \wedge C) \vdash \Delta_\diamond} \; [\supset_\wedge]$$

$$\frac{\Gamma, A \supset C, B \supset C \vdash \Delta_\diamond}{\Gamma, (A \vee B) \supset C \vdash \Delta_\diamond} \; [_\vee \supset] \quad \frac{\Gamma, A \supset B \vdash \Delta_\diamond \quad \Gamma, A \supset C \vdash \Delta_\diamond}{\Gamma, A \supset (B \vee C) \vdash \Delta_\diamond} \; [\supset_\vee]$$

$$\frac{\Gamma, B \supset C \vdash A \supset B, \diamond \supset B, \Delta_\diamond \quad \Gamma, \diamond \supset C \vdash \Delta_\diamond}{\Gamma, (A \supset B) \supset C \vdash \Delta_\diamond} \; [_\supset \supset]$$

$$\frac{\Gamma, A \supset C \vdash \Delta_\diamond \quad \Gamma, B \supset C \vdash \Delta_\diamond}{\Gamma, A \supset (B \supset C) \vdash \Delta_\diamond} \; [\supset_\supset]$$

Figure 1. Proof rules for decomposing flat sequents

$[\![\square]\!] - \alpha = 0 = [\![\bot]\!]_{-\alpha}$, we obtain $[\![D]\!]_{-\alpha} = [\![D_\square]\!]_{-\alpha}$ and by Proposition 2.2, $[\![D_\square]\!]_{-\alpha} = [\![D_\square]\!] - \alpha$. Since $[\![D_\square]\!] < \lfloor\!\lfloor \delta_D^- \rfloor\!\rfloor$ holds, $[\![D_\square]\!] < \infty$ and then $[\![D_\square]\!] - \alpha < \infty$. Therefore $[\![D]\!]_{-\alpha} < \infty$ holds and $[\![\cdot]\!]_{-\alpha}$ is a counter-model of D in LC_n.

4. Reduction of flat sequents by proof-search

In the preceding section, we have proved that the problem of deciding if a formula D is valid or has a counter-model in LC_n can be transformed into deciding the sequent $\delta_D^- \vdash \diamond \supset \mathcal{X}_D$, by combination of Theorems 1 and 2. By Proposition 6, the sequent $\delta_D^- \vdash \diamond \supset \mathcal{X}_D$ is obviously flat and does not contain the constant $\bot$. It contains the special variable $\square$ (resp. $\diamond$) which intuitively represents the semantic value 0 (resp. ∞). But in the proof-search process, $\square$ and $\diamond$ are treated as variables, not as constants.

We propose a set of rules to reduce flat sequents $\Gamma \vdash \Delta_\diamond$ into implicational sequents in Figure 1.[6] We point out that rule $[_\circledast \supset]$ decomposes a formula of type $(A \circledast B) \supset C$ into implications and rule $[\supset_\circledast]$ decomposes a formula of type $A \supset (B \circledast C)$ into implications. Moreover, those rules preserve flat sequents when applied bottom-up: in particular, in the left premise of rule $[_\supset \supset]$, $A \supset B, \diamond \supset B, \Delta_\diamond$ is a $\diamond$-context when $\Delta_\diamond$ is $\diamond$-context.

THEOREM 3. *The rules of Figure 1 are strongly sound for LC_n.*

---

[6] Except from rule $[_\supset \supset]$, the other rules are just instances of rules of [2, 9] in the context of flat sequents.

Proof. We propose the proofs for rules $[_\wedge\supset]$ and for $[_\supset\supset]$. The other rules are treated by similar arguments as rule $[_\wedge\supset]$. The case of rule $[_\supset\supset]$ is different and more difficult to prove.

Let us start with rule $[_\wedge\supset]$. Let $\llbracket\cdot\rrbracket$ be an interpretation which is a model of both $\Gamma, A \supset C \vdash \Delta_\Diamond$ and $\Gamma, B \supset C \vdash \Delta_\Diamond$. Thus the following inequalities hold: $\lfloor\Gamma\rfloor \wedge \llbracket A \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$ and $\lfloor\Gamma\rfloor \wedge \llbracket B \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$. We compute using equivalence 1) of Proposition 4: $\lfloor\Gamma\rfloor \wedge \llbracket(A \wedge B) \supset C\rrbracket = \lfloor\Gamma\rfloor \wedge \llbracket(A \supset C) \vee (B \supset C)\rrbracket = \lfloor\Gamma\rfloor \wedge (\llbracket A \supset C\rrbracket \vee \llbracket B \supset C\rrbracket) = (\lfloor\Gamma\rfloor \wedge \llbracket A \supset C\rrbracket) \vee (\lfloor\Gamma\rfloor \wedge \llbracket B \supset C\rrbracket) \leqslant \lceil\Delta_\Diamond\rceil$. So $\llbracket\cdot\rrbracket$ is a model of the sequent $\Gamma, (A \wedge B) \supset C \vdash \Delta_\Diamond$.

Now we prove that rule $[_\supset\supset]$ is strongly sound. Let $\llbracket\cdot\rrbracket$ be an interpretation which is a model of both premises of rule $[_\supset\supset]$. Thus both $\lfloor\Gamma\rfloor \wedge \llbracket B\supset C\rrbracket \leqslant \llbracket A\supset B\rrbracket \vee \llbracket\Diamond\supset B\rrbracket \vee \lceil\Delta_\Diamond\rceil$ and $\lfloor\Gamma\rfloor \wedge \llbracket\Diamond\supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$ hold. Let us consider two trivial cases:

- if $\lfloor\Gamma\rfloor \leqslant \lceil\Delta_\Diamond\rceil$, it is clear that $\lfloor\Gamma\rfloor \wedge \llbracket(A \supset B) \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$ holds and $\llbracket\cdot\rrbracket$ is a model of the conclusion of rule $[_\supset\supset]$;

- if $\llbracket\Diamond\rrbracket \leqslant \llbracket A \supset B\rrbracket$ then, by Proposition 1, $\llbracket(A \supset B) \supset C\rrbracket \leqslant \llbracket\Diamond\supset C\rrbracket$ and thus $\lfloor\Gamma\rfloor \wedge \llbracket(A \supset B) \supset C\rrbracket \leqslant \lfloor\Gamma\rfloor \wedge \llbracket\Diamond \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$. So $\llbracket\cdot\rrbracket$ is a model of the conclusion of rule $[_\supset\supset]$;

So we are left with the case where $\lceil\Delta_\Diamond\rceil < \lfloor\Gamma\rfloor$ and $\llbracket A \supset B\rrbracket < \llbracket\Diamond\rrbracket$ both hold. We can deduce $\lceil\Delta_\Diamond\rceil < \infty$ and thus $\lceil\Delta_\Diamond\rceil < \llbracket\Diamond\rrbracket$ by Proposition 3. From $\lfloor\Gamma\rfloor \wedge \llbracket\Diamond \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$ we deduce $\llbracket\Diamond \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil < \infty$. So $\llbracket C\rrbracket = \llbracket\Diamond \supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$.

From $\llbracket A\supset B\rrbracket < \llbracket\Diamond\rrbracket$ we deduce $\llbracket A\supset B\rrbracket = \llbracket\Diamond\supset B\rrbracket = \llbracket B\rrbracket < \llbracket\Diamond\rrbracket$. We compute $\lfloor\Gamma\rfloor \wedge \llbracket B\supset C\rrbracket \leqslant \llbracket A\supset B\rrbracket \vee \llbracket\Diamond\supset B\rrbracket \vee \lceil\Delta_\Diamond\rceil \leqslant \llbracket B\rrbracket \vee \lceil\Delta_\Diamond\rceil$. Now suppose that $\llbracket B\rrbracket \leqslant \llbracket C\rrbracket$ then $\llbracket B\supset C\rrbracket = \infty$ and $\lfloor\Gamma\rfloor = \lfloor\Gamma\rfloor \wedge \llbracket B\supset C\rrbracket \leqslant \llbracket B\rrbracket \vee \lceil\Delta_\Diamond\rceil \leqslant \llbracket C\rrbracket \vee \lceil\Delta_\Diamond\rceil \leqslant \lceil\Delta_\Diamond\rceil$. Thus $\lfloor\Gamma\rfloor \leqslant \lceil\Delta_\Diamond\rceil$ holds which contradicts the previous assumption $\lceil\Delta_\Diamond\rceil < \lfloor\Gamma\rfloor$. Since $\llbracket B\rrbracket \leqslant \llbracket C\rrbracket$ leads to a contradiction, the property $\llbracket C\rrbracket < \llbracket B\rrbracket$ necessarily holds. Then $\llbracket(A \supset B) \supset C\rrbracket = \llbracket A \supset B\rrbracket \twoheadrightarrow \llbracket C\rrbracket = \llbracket B\rrbracket \twoheadrightarrow \llbracket C\rrbracket = \llbracket C\rrbracket = \llbracket\Diamond \supset C\rrbracket$. We conclude by $\lfloor\Gamma\rfloor \wedge \llbracket(A \supset B)\supset C\rrbracket \leqslant \lfloor\Gamma\rfloor \wedge \llbracket\Diamond\supset C\rrbracket \leqslant \lceil\Delta_\Diamond\rceil$. Hence, $\llbracket\cdot\rrbracket$ is a model of the conclusion of rule $[_\supset\supset]$.

THEOREM 4. *The rules of Figure 1 are strongly invertible for* LC_n.

Proof. We prove strong invertibility of $[_\wedge\supset]$. Let $\llbracket\cdot\rrbracket$ be a counter-model of $\Gamma, A \supset C \vdash \Delta_\Diamond$. We compute $\lceil\Delta_\Diamond\rceil < \lfloor\Gamma\rfloor \wedge \llbracket A \supset C\rrbracket \leqslant \lfloor\Gamma\rfloor \wedge (\llbracket A\supset C\rrbracket \vee \llbracket B\supset C\rrbracket) = \lfloor\Gamma, (A\wedge B)\supset C\rfloor$. Thus $\lceil\Delta_\Diamond\rceil < \lfloor\Gamma, (A\wedge B)\supset C\rfloor$ holds and $\llbracket\cdot\rrbracket$ is a counter-model of $\Gamma, A\supset C\vdash \Delta_\Diamond$. The case of $\llbracket\cdot\rrbracket$ being a counter-model of $\Gamma, B\supset C\vdash\Delta_\Diamond$ can be handled with the same argument.

To finish, we prove that rule $[_\supset\supset]$ is strongly invertible. First let $\llbracket\cdot\rrbracket$ be a counter-model of the left premise, i.e. $\llbracket A \supset B\rrbracket \vee \llbracket\Diamond \supset B\rrbracket \vee$

$\llbracket \Delta_\Diamond \rrbracket < \lfloor \Gamma \rfloor \wedge \llbracket B \supset C \rrbracket$ holds. So $\llbracket A \supset B \rrbracket < \infty$ holds and then $\llbracket A \supset B \rrbracket = \llbracket B \rrbracket < \llbracket A \rrbracket$. We deduce $\llbracket (A \supset B) \supset C \rrbracket = \llbracket B \supset C \rrbracket$ and then $\llbracket \Delta_\Diamond \rrbracket < \lfloor \Gamma \rfloor \wedge \llbracket B \supset C \rrbracket = \lfloor \Gamma \rfloor \wedge \llbracket (A \supset B) \supset C \rrbracket$ so $\llbracket \cdot \rrbracket$ is a counter-model of the conclusion of rule $[_\supset \supset]$.

Finally let $\llbracket \cdot \rrbracket$ be a counter-model of the right premise. Thus $\llbracket \Delta_\Diamond \rrbracket < \lfloor \Gamma \rfloor \wedge \llbracket \Diamond \supset C \rrbracket$ holds. So $\llbracket \Delta_\Diamond \rrbracket < \infty$ and thus $\llbracket \Delta_\Diamond \rrbracket < \llbracket \Diamond \rrbracket$ by Proposition 3. Also $\llbracket \Delta_\Diamond \rrbracket < \llbracket \Diamond \supset C \rrbracket$. If we suppose $\llbracket C \rrbracket \leqslant \llbracket \Delta_\Diamond \rrbracket$ then $\llbracket C \rrbracket < \llbracket \Diamond \rrbracket$ so $\llbracket \Diamond \supset C \rrbracket = \llbracket C \rrbracket$ and $\llbracket \Delta_\Diamond \rrbracket < \llbracket C \rrbracket$ which is absurd. So necessarily, $\llbracket \Delta_\Diamond \rrbracket < \llbracket C \rrbracket$ holds. Finally $\llbracket \Delta_\Diamond \rrbracket < \llbracket C \rrbracket \leqslant \llbracket (A \supset B) \supset C \rrbracket$ thus $\llbracket \Delta_\Diamond \rrbracket < \lfloor \Gamma \rfloor \wedge \llbracket (A \supset B) \supset C \rrbracket$ holds and $\llbracket \cdot \rrbracket$ is a counter-model of the conclusion of rule $[_\supset \supset]$.

PROPOSITION 9. *The flat sequents which are irreducible by rules* $[_\circledast \supset]$ *and* $[_\supset \circledast]$ *for* $\circledast \in \{\wedge, \vee, \supset\}$ *are implicational sequents.*

Proof. Flat sequents contain no constant (i.e. $\perp$) and only atomic implications on the right-hand side of the $\vdash$ sign. If there exists a formula of type $(X \circledast Y) \supset Z$ (resp. $X \supset (Y \circledast Z)$) on the left-hand side, then the rule $[_\circledast \supset]$ (resp. $[_\supset \circledast]$) can be applied backwards. So if the sequent is irreducible, it cannot contain formulæ of type $(X \circledast Y) \supset Z$ or $X \supset (Y \circledast Z)$ on the left hand side. Thus, the left-hand side only contains atomic implications.

Its is clear that the backward application of proof rules $[_\circledast \supset]$ and $[_\supset \circledast]$, starting from $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D$, terminates since the number of formulæ of type $(X \circledast Y) \supset Z$ or $X \supset (Y \circledast Z)$ decreases by one each time such a rule is applied. Moreover, the reader could notice that the order in which the proof rules are applied in each branch of the proof-search tree does not change the final result. These final results, which are the sequents at the leaves of proof-search branches, are *irreducible* by any of the rules $[_\circledast \supset]$ or $[_\supset \circledast]$. Thus, these sequents are implicational.

The variables $\square$ and $\Diamond$ play a special role during the indexation phase and the reduction phase: $\square$ is introduced so that the constant $\perp$ does not occur in δ_D^- and $\Diamond$ is intimately related to the definition of $\Diamond$-context, and thus of flat sequents. Once the decomposition process is finished, $\square$ and $\Diamond$ cease to play a special role: they are considered as normal variables in implicational sequents.

5. Counter-models of implicational sequents

In this section, we present a criterion to decide implicational sequents in LC_n. These sequents are transformed into a graph on which we detect *alternating chains* or $\Rightarrow$-*cycles* to build counter-models.

5.1. Bi-colored graph of an implicational sequent

Let $\mathcal{S} = X_1 \supset Y_1, \ldots, X_k \supset Y_k \vdash A_1 \supset B_1, \ldots, A_l \supset B_l$ be an implicational sequent. We build a *bi-colored graph* $\mathcal{G}_\mathcal{S}$ which has nodes in the set $\{X_i\} \cup \{Y_i\} \cup \{A_i\} \cup \{B_i\}$ and has two kinds of arrows, green (denoted by $\rightarrow$) and red (denoted by $\Rightarrow$). The set of arrows is:

$$\{X_1 \rightarrow Y_1, \ldots, X_k \rightarrow Y_k\} \cup \{B_1 \Rightarrow A_1, \ldots, B_l \Rightarrow A_l\}$$

The red arrow $B_i \Rightarrow A_i$ is in the direction *opposite* to that in the implication $A_i \supset B_i$. We also have a set of arrows so even if $X \supset Y$ has multiple occurrences in $\mathcal{S}$, there is only one arrow $X \rightarrow Y$ in $\mathcal{G}_\mathcal{S}$.

DEFINITION 2 (Bi-colored graph). *A bi-colored graph is a finite directed graph with two kinds of arrows denoted $\rightarrow$ and $\Rightarrow$.*

We will often use the same symbols $\rightarrow$ and $\Rightarrow$ to denote the corresponding incidence relation in the graph. So for example, $\rightarrow\Rightarrow$ denotes the composition of the two relations and $u \rightarrow\Rightarrow w$ means there exists a chain $u \rightarrow v \Rightarrow w$ in $\mathcal{G}_\mathcal{S}$. Also $\rightarrow^\star$ is the reflexive and transitive closure of $\rightarrow$, i.e. the accessibility for the $\rightarrow$ relation, and $\rightarrow + \Rightarrow$ is the union of relations.

5.2. Bi-heights, alternating chains and $\Rightarrow$-cycles

We define a notion of bi-height in bi-colored graphs. The idea is simple: a green arrow $\rightarrow$ weighs 0 and a red arrow $\Rightarrow$ weighs 1, and the height of a node v is the greatest weight of paths leading to v.

DEFINITION 3 (Bi-height). *Let $\mathcal{G}$ be a bi-colored graph. A bi-height is a function $h : \mathcal{G} \rightarrow \mathbb{N}$ such that for any $u, v \in \mathcal{G}$, if $u \rightarrow v \in \mathcal{G}$ then $h(u) \leqslant h(v)$ and if $u \Rightarrow v \in \mathcal{G}$ then $h(u) < h(v)$.*

As we will see, bi-heights are used to compute counter-models. We characterize graphs that admit bi-heights by the notion of $\Rightarrow$-cycle. This notion is similar to the notion of G-cycle in [3] but here we give a simple and efficient algorithm to find cycles and compute counter-models of the corresponding implicational sequent.

DEFINITION 4 ($\Rightarrow$-cycle). *A chain of the form $u(\rightarrow + \Rightarrow)^\star \Rightarrow u$ for some node u is called a $\Rightarrow$-cycle.*

It is clear that if a graph has a $\Rightarrow$-cycle, then there is no bi-height: we would obtain $h(u) < h(u)$. Conversely, we give a linear[7] algorithm to compute a bi-height when there is no $\Rightarrow$-cycle.

---

[7] Linearity is measured w.r.t. the number of vertexes and arrows in the graph.

THEOREM 5. *Let $\mathcal{G}$ be a bi-colored graph. One can decide if $\mathcal{G}$ has $\Rightarrow$-cycles in linear time and if not, one can compute a bi-height h for $\mathcal{G}$ in linear time.*

Proof. Even if it has no $\Rightarrow$-cycle, $\mathcal{G}$ may still contain green ($\rightarrow$) cycles. To remove all cycles, we introduce the contracted graph $\mathcal{G}'$ of $\mathcal{G}$: let $\mathcal{C}$ be the set of strongly connected components for the "green" sub-graph of $\mathcal{G}$ (i.e. $\mathcal{G}_\rightarrow$), $\mathcal{C} = \{[u] \mid u \in \mathcal{G}_\rightarrow\}$ and $[u]$ is the strongly connected component of u in $\mathcal{G}_\rightarrow$. The graph $\mathcal{G}'$ has $\mathcal{C}$ as set of nodes, and the set of arrows is described by:

$[u] \rightarrow [v]$ iff $[u] \neq [v]$ and $\exists u', v',\ [u] = [u'], [v] = [v']$ and $u' \rightarrow v' \in \mathcal{G}$
$[u] \Rightarrow [v]$ iff $\exists u', v',\ [u] = [u'], [v] = [v']$ and $u' \Rightarrow v' \in \mathcal{G}$

$\mathcal{G}'$ is computed in linear time by standard depth first search algorithms. $\mathcal{G}'$ has no green ($\rightarrow$) cycle (because they collapse into a strongly connected component) and so $\mathcal{G}'$ has a cycle (with either $\rightarrow$ or $\Rightarrow$ arrows) if and only if $\mathcal{G}$ has a $\Rightarrow$-cycle. Finding a cycle in $\mathcal{G}'$ takes linear time in the size of $\mathcal{G}'$ (which is smaller than the size of $\mathcal{G}$).

Now suppose that $\mathcal{G}'$ has no cycle (i.e. no $\Rightarrow$-cycle in $\mathcal{G}$). The relation $(\rightarrow + \Rightarrow)^\star$ is a finite partial order and we can define $h' : \mathcal{G}' \rightarrow I\!\!N$ inductively by:

$$h'([v]) = \max \left\{ \begin{array}{l} h'([u]) \text{ for } [u] \rightarrow [v] \in \mathcal{G}' \\ h'([u]) + 1 \text{ for } [u] \Rightarrow [v] \in \mathcal{G}' \end{array} \right\}$$

We can compute the whole function h' in linear time by sorting the nodes of $\mathcal{G}'$ along $(\rightarrow + \Rightarrow)^\star$, again by depth first search. We define $h(u) = h'([u])$ and prove that h is a bi-height in $\mathcal{G}$. If $u \rightarrow v \in \mathcal{G}$: first case $[u] = [v]$ and then $h(u) = h(v)$, second case $[u] \neq [v]$ and then $[u] \rightarrow [v] \in \mathcal{G}'$ thus $h'([u]) \leqslant h'([v])$ by definition of h', so $h(u) \leqslant h(v)$. If $u \Rightarrow v \in \mathcal{G}$ then $[u] \Rightarrow [v] \in \mathcal{G}'$ and $h'([u]) + 1 \leqslant h'([v])$ so $h(u) < h(v)$.

DEFINITION 5 (Alternating chain). *A k-alternating chain is a chain of type $(\rightarrow^\star\Rightarrow)^k$.*

A k-alternating chain contains exactly k occurrences of a red arrow $\Rightarrow$ and ends with a red arrow $\Rightarrow$. Obviously, by cycling around a $\Rightarrow$-cycle, one can build k-alternating chains for any $k \in I\!\!N$.

THEOREM 6. *Let $k > 0$ be an integer. If $\mathcal{G}$ has no k-alternating chain then it contains no $\Rightarrow$-cycle and the height h of Theorem 5 satisfies $\forall v \in \mathcal{G},\ h(v) < k$.*

Proof. In $\mathcal{G}'$, if $h'([v]) = k$ then there exists a chain $[u](\Rightarrow\rightarrow^\star)^k[v]$ in $\mathcal{G}'$. This result is straightforwardly proved by induction on k. Then suppose that there exists v such that $h(v) = m \geqslant k$. We obtain a

chain of type $(\Rightarrow \to^\star)^m$ in $\mathcal{G}'$. Expanding the "green" strongly connected components of $\mathcal{G}$, we obtain a chain of type $(\to^\star(\Rightarrow \to^\star)\to^\star)^m$ in $\mathcal{G}$. Since $m \geqslant k$, it contains a subchain of type $(\to^\star \Rightarrow)^k$ and thus a k-alternating chain.

THEOREM 7. *Let $\mathcal{G}$ be a bi-colored graph with no $\Rightarrow$-cycle. Then, for k greater than the number of nodes of $\mathcal{G}$, the graph $\mathcal{G}$ has no k-alternating chain.*

Proof. Let s be the number of nodes of $\mathcal{G}$. Let k be greater or equal to s, $k \geqslant s$. Suppose that $\mathcal{G}$ has a chain of the form $u_0 \to^\star \Rightarrow u_1 \to^\star \Rightarrow \cdots \to^\star \Rightarrow u_k$. If all the u_i are different then the set $\{u_0, u_1, \ldots, u_k\}$ contains $k+1 > s$ nodes of $\mathcal{G}$. This is not possible. So let $i < j$ be such that $u_i = u_j$. The chain $u_i \to^\star \Rightarrow \cdots \to^\star \Rightarrow u_j$ is a $\Rightarrow$-cycle.

5.3. Counter-models vs. chains in bi-colored graphs

We relate alternating chains and $\Rightarrow$-cycles in the bi-colored graph associated with a sequent to the counter-models of that sequent.

PROPOSITION 10. *Let $\mathcal{S}$ be an implicational sequent and $\mathcal{G}_\mathcal{S}$ its associated bi-colored graph. Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ be a counter-model of $\mathcal{S}$ in LC_n and $X_1 \to \cdots \to X_k \Rightarrow Y$ a 1-alternating chain in $\mathcal{G}_\mathcal{S}$. Then $[\![X_1]\!] \leqslant \cdots \leqslant [\![X_k]\!] < [\![Y]\!]$ holds.*

Proof. Let $\mathcal{S} = \Gamma \vdash \Delta$. As $[\![\cdot]\!]$ is a counter-model, the relation $[\![\Delta]\!] < [\![\Gamma]\!]$ holds. As $X_k \Rightarrow Y \in \mathcal{G}_\mathcal{S}$, the formula $Y \supset X_k$ is an element of Δ. So we deduce $[\![Y \supset X_k]\!] \leqslant [\![\Delta]\!] < \infty$. Thus we obtain $[\![Y]\!] > [\![X_k]\!]$ and $[\![X_k]\!] = [\![Y \supset X_k]\!] \leqslant [\![\Delta]\!]$. Also $X_{k-1} \to X_k \in \mathcal{G}_\mathcal{S}$, so $X_{k-1} \supset X_k$ belongs to Γ. Thus $[\![X_k]\!] \leqslant [\![\Delta]\!] < [\![\Gamma]\!] \leqslant [\![X_{k-1} \supset X_k]\!]$ holds. So it is necessary that $[\![X_{k-1}]\!] \leqslant [\![X_k]\!]$ (because otherwise, $[\![X_{k-1} \supset X_k]\!] = [\![X_k]\!]$ holds) and we deduce $[\![X_{k-1}]\!] \leqslant [\![X_k]\!] \leqslant [\![\Delta]\!]$. By descending induction on i for $k-1, k-2, \ldots, 2$, we can prove that $[\![X_{i-1}]\!] \leqslant [\![X_i]\!] \leqslant [\![\Delta]\!]$.

THEOREM 8 $(n < \infty)$. *An implicational sequent $\mathcal{S}$ has a counter-model in LC_n if and only if its associated graph $\mathcal{G}_\mathcal{S}$ does not contain $(n+1)$-alternating chains.*

Proof. Let $\mathcal{S} \equiv \Gamma \vdash \Delta$. First we prove the *if part*. We suppose that $\mathcal{G}_\mathcal{S}$ contains no chain of the form $(\to^\star \Rightarrow)^{n+1}$. Then by Theorem 6, there exists a bi-height function $h : \mathcal{G}_\mathcal{S} \to [0,n]$. We define the semantic function $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$: if X occurs in $\mathcal{S}$ (i.e. is a node of $\mathcal{G}_\mathcal{S}$), $[\![X]\!] = h(X)$ if $h(X) < n$ and $[\![X]\!] = \infty$ if $h(X) = n$; if X does not occur in $\mathcal{S}$, $[\![X]\!] = \infty$ (any other value would also fit). Let us prove that $[\![\cdot]\!]$ is a counter-model of $\mathcal{S}$. Indeed, if $X \supset Y \in \Gamma$ then $X \to Y \in \mathcal{G}_\mathcal{S}$ and

then $h(X) \leqslant h(Y)$. It follows that $[\![X]\!] \leqslant [\![Y]\!]$ and so $[\![X \supset Y]\!] = \infty$. We have $[\![\Gamma]\!] = \infty$. If $X \supset Y \in \Delta$ then $Y \Rightarrow X \in \mathcal{G}_\mathcal{S}$. Thus $h(Y) < h(X)$ and $[\![X \supset Y]\!] = [\![Y]\!] = h(Y) < h(X) \leqslant n$. So $[\![X \supset Y]\!] \leqslant n - 1$ holds for any $X \supset Y \in \Delta$ and then $[\![\Delta]\!] \leqslant n - 1$. Finally, $[\![\Delta]\!] \leqslant n - 1 < \infty = [\![\Gamma]\!]$ and so $[\![\cdot]\!]$ is a counter-model of $\mathcal{S}$.

Now we prove the *only if part*. Let $[\![\cdot]\!] : \mathsf{Var} \to \overline{[0,n)}$ be a counter-model of $\mathcal{S}$. Suppose there is a chain of the form $(\to^\star \Rightarrow)^{n+1}$ in $\mathcal{G}_\mathcal{S}$: $X_0 \to^\star \Rightarrow X_1 \to^\star \Rightarrow X_2 \to^\star \Rightarrow \cdots \to^\star \Rightarrow X_n \to^\star \Rightarrow X_{n+1}$. So for any i, there is a chain $X_i \to^\star \Rightarrow X_{i+1}$ and by Proposition 10, we obtain $[\![X_i]\!] < [\![X_{i+1}]\!]$. Then, $[\![X_0]\!] < [\![X_1]\!] < \cdots < [\![X_{n+1}]\!]$ is a strictly increasing sequence of $n + 2$ elements in $\overline{[0,n)}$. As this set has $n + 1$ elements, we get a contradiction.

THEOREM 9 ($n = \infty$). *An implicational sequent $\mathcal{S}$ has a counter-model in LC if and only if its associated graph $\mathcal{G}_\mathcal{S}$ has no $\Rightarrow$-cycle.*

Proof. For the *if part*, if $\mathcal{G}_\mathcal{S}$ has no $\Rightarrow$-cycle, by Theorem 5, there is a height $h : \mathcal{G}_\mathcal{S} \to \mathbb{N}$. We define $[\![X]\!] \in \mathbb{N} \cup \{\infty\}$ by $[\![X]\!] = h(X)$ and obtain a counter-model of $\mathcal{S}$ in LC. For the *only if part*, the existence of a chain $X \to^\star \Rightarrow \to^\star \Rightarrow \cdots \to^\star \Rightarrow X$ would lead to $[\![X]\!] < [\![X]\!]$ by the same argument as before.

5.4. An example of proof and counter-model search

In this section, we apply the proof-search and counter-model search techniques previously described on the formula expressing the law of the excluded middle: $D \equiv A \lor (A \supset \bot)$.

First we index this formula by listing all the occurrences of its subformulæ in a depth-first search manner, giving a number to each occurrence: $0 \equiv A \lor (A \supset \bot), 1 \equiv A, 2 \equiv A \supset \bot, 3 \equiv A, 4 \equiv \bot$. Then, we compute δ_D^-:

$$\delta_D^- = A \supset 1, \square \supset A, 3 \supset A, \square \supset A, \square \supset 4, (3 \supset 4) \supset 2, (1 \lor 2) \supset 0$$

For convenience, we denote by Γ the context containing the atomic implications of δ_D^-: $\Gamma \equiv A \supset 1, \square \supset A, 3 \supset A, \square \supset A, \square \supset 4$. Then we compute the proof-search tree of $\delta_D^- \vdash \Diamond \supset 0$:

$$\cfrac{\cfrac{\Gamma, 4 \supset 2, 1 \supset 0, 2 \supset 0 \vdash 3 \supset 4, \Diamond \supset 4, \Diamond \supset 0 \quad \Gamma, \Diamond \supset 2, 1 \supset 0, 2 \supset 0 \vdash \Diamond \supset 0}{\Gamma, (3 \supset 4) \supset 2, 1 \supset 0, 2 \supset 0 \vdash \Diamond \supset 0} {\scriptstyle[\supset\supset]}}{\Gamma, (3 \supset 4) \supset 2, (1 \lor 2) \supset 0 \vdash \Diamond \supset 0} {\scriptstyle[\lor\supset]}$$

Then we obtain two proof-search branches corresponding to the choice between left and right premise in the instance of rule $[\supset\supset]$. We present

the two bi-colored graphs corresponding to the two implicational se-
quents at the leaves of the proof-search tree:
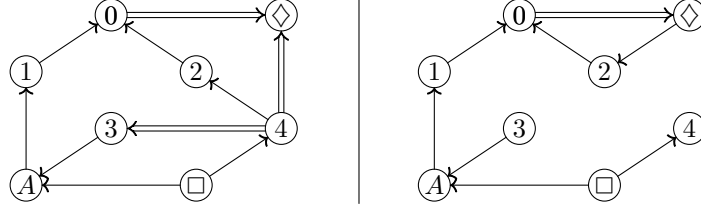
The right bi-colored graph contains a $\Rightarrow$-cycle: $\Diamond \rightarrow 2 \rightarrow 0 \Rightarrow \Diamond$ whereas
the left bi-colored graph does not contain any $\Rightarrow$-cycle, as a bi-height
can be computed for it:
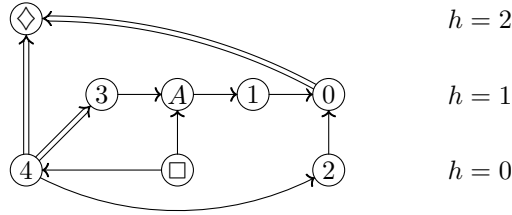
Indeed, on this particular presentation of the graph, it is clear that
green arrows $\rightarrow$ either go up or stay on the same level whereas red
arrows $\Rightarrow$ strictly go up. We remark that there is no 3-alternating chain
so we will find a counter-model in LC_2. On the other hand, there exists
a 2-alternating chain which corresponds to the fact that $A \vee (A \supset \bot)$
is valid in $\mathsf{LC}_1 = \mathsf{CL}$ which is classical propositional logic.

From the bi-height h, we compute the following minimal counter-
model of $A \vee (A \supset \bot)$, $[\![\cdot]\!] : \mathsf{Var} \rightarrow \overline{[0,2)}$ with $[\![A]\!] = h_A = 1$. We can
check that this is indeed a counter-model: $[\![A \vee (A \supset \bot)]\!] = 1 \vee (1 \rightarrow 0) =$
$1 \vee 0 = 1 < \infty$ in $\overline{[0,2)}$.

5.5. Counter-models by matrix computation

We have seen that the existence of counter-models of implicational
sequents in LC is equivalent to the absence of $\Rightarrow$-cycles in the associated
bi-colored graphs. Now we present an algebraic formula that expresses
the existence of $\Rightarrow$-cycles. Let $\mathcal{G}$ be a bi-colored graph of k nodes with
its incidence relations $\rightarrow$ and $\Rightarrow$. The relation $\rightarrow$ (or $\Rightarrow$) can be viewed
as an *incidence $k \times k$ matrix* whose rows and columns are indexed
by the nodes of $\mathcal{G}$. The cells of these matrices take their value in the
boolean algebra $\{0, 1\}$. So there is a 1 at cell (u, v) in the matrix of
$\rightarrow$ if and only if $u \rightarrow v \in \mathcal{G}$. We define $+$ as the disjunction (or logical
"or") and $\cdot$ as the conjunction (or logical "and") in the boolean algebra

$\{0,1\}$. These operations extend naturally to sum and multiplication of square boolean matrices.

If we identify the relations $\rightarrow$ and $\Rightarrow$ with their respective matrices, the composed relation $\rightarrow\Rightarrow$ has a corresponding matrix $\rightarrow \cdot \Rightarrow$ and the union of relations $\rightarrow$ and $\Rightarrow$ has a corresponding matrix $\rightarrow + \Rightarrow$. The relation $\rightarrow^\star$ corresponds to a limit matrix $\rightarrow^0 + \cdots + \rightarrow^i + \cdots$ by the Warshall theorem [17].[8] So $((\rightarrow + \Rightarrow)^\star \Rightarrow)_{u,u} = 1$ means that there exists a chain of the form $u(\rightarrow + \Rightarrow)^\star \Rightarrow u$ in the graph $\mathcal{G}$. Let $tr(\cdot)$ denote the trace of matrices defined by $tr(M) = \sum_u M_{u,u}$.

PROPOSITION 11. $\mathcal{G} = (\rightarrow, \Rightarrow)$ *contains a* $\Rightarrow$-*cycle if and only if*

$$tr((\rightarrow + \Rightarrow)^\star \Rightarrow) = 1$$

We conclude this section by a criterion to determine the minimal n for which a given sequent $\mathcal{S}$ has a counter-model in LC_n. Let $\sum M$ denote the sum of all the elements of the matrix M: $\sum M = \sum_{u,v} M_{u,v}$.

PROPOSITION 12. *Let* $\mathcal{S}$ *be an implicational sequent not valid in* LC *and* $\mathcal{G}_\mathcal{S} = (\rightarrow, \Rightarrow)$ *be its associated bi-colored graph. There exists a smallest* n *such that* $\mathcal{S}$ *has a counter-model in* LC_n *and it is the least* n *such that*

$$\sum(\rightarrow^\star \Rightarrow)^{n+1} = 0$$

Proof. By Theorem 8, $\mathcal{S}$ has a counter-model in LC_n, iff $\mathcal{G}_\mathcal{S}$ has no chain of the form $(\rightarrow^\star \Rightarrow)^{n+1}$. Having no chain of the form $(\rightarrow^\star \Rightarrow)^{n+1}$ means the matrix of the corresponding relation is the zero matrix, i.e. $\sum(\rightarrow^\star \Rightarrow)^{n+1} = 0$. As $\mathcal{S}$ is not valid in LC, by Theorem 9, its bi-colored graph $\mathcal{G}_\mathcal{S}$ has no $\Rightarrow$-cycle. Then, there is an n such that $\mathcal{G}_\mathcal{S}$ has no chain of the form $(\rightarrow^\star \Rightarrow)^n$ by Theorem 7. Thus there exists a smallest one.

6. Combining proof-search and counter-model computation

Our system has a very important property that the other sequent [9, 12] or hyper-sequent [2] systems lack: proof-search can be seen as the incremental construction of a semantic graph. See section 9 for a discussion on this point. The nodes of this graph do not depend on the proof-search branch; only the choices of arrows depend on the chosen branch.

---

[8] The sequence $(\rightarrow^0 + \cdots + \rightarrow^i)_{i \in \mathbb{N}}$ is increasing in the finite point-wise ordered poset $\{0,1\}^{k \times k}$. Thus, its limit exists and is reached within a finite number of steps. This result is still valid if $\{0,1\}$ is replaced by any other finite boolean algebra like for example in section 8.

$$\frac{\Gamma, \mathcal{X}_A^- \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_\Diamond \quad \Gamma, \mathcal{X}_B^- \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_\Diamond}{\Gamma, (\mathcal{X}_A^- \wedge \mathcal{X}_B^-) \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_\Diamond} \; [\wedge\supset]$$

$$\frac{\Gamma, \mathcal{X}_{A\wedge B}^+ \supset \mathcal{X}_A^+, \mathcal{X}_{A\wedge B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_\Diamond}{\Gamma, \mathcal{X}_{A\wedge B}^+ \supset (\mathcal{X}_A^+ \wedge \mathcal{X}_B^+) \vdash \Delta_\Diamond} \; [\supset\wedge]$$

$$\frac{\Gamma, \mathcal{X}_A^- \supset \mathcal{X}_{A\vee B}^-, \mathcal{X}_B^- \supset \mathcal{X}_{A\vee B}^- \vdash \Delta_\Diamond}{\Gamma, (\mathcal{X}_A^- \vee \mathcal{X}_B^-) \supset \mathcal{X}_{A\vee B}^- \vdash \Delta_\Diamond} \; [\vee\supset]$$

$$\frac{\Gamma, \mathcal{X}_{A\vee B}^+ \supset \mathcal{X}_A^+ \vdash \Delta_\Diamond \quad \Gamma, \mathcal{X}_{A\vee B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_\Diamond}{\Gamma, \mathcal{X}_{A\vee B}^+ \supset (\mathcal{X}_A^+ \vee \mathcal{X}_B^+) \vdash \Delta_\Diamond} \; [\supset\vee]$$

$$\frac{\Gamma, \mathcal{X}_B^- \supset \mathcal{X}_{A\supset B}^- \vdash \mathcal{X}_A^+ \supset \mathcal{X}_B^-, \Diamond \supset \mathcal{X}_B^-, \Delta_\Diamond \quad \vdots \quad \Gamma, \Diamond \supset \mathcal{X}_{A\supset B}^- \vdash \Delta_\Diamond}{\Gamma, (\mathcal{X}_A^+ \supset \mathcal{X}_B^-) \supset \mathcal{X}_{A\supset B}^- \vdash \Delta_\Diamond} \; [\supset\supset]$$

$$\frac{\Gamma, \mathcal{X}_{A\supset B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_\Diamond \quad \Gamma, \mathcal{X}_A^- \supset \mathcal{X}_B^+ \vdash \Delta_\Diamond}{\Gamma, \mathcal{X}_{A\supset B}^+ \supset (\mathcal{X}_A^- \supset \mathcal{X}_B^+) \vdash \Delta_\Diamond} \; [\supset\supset]$$

Figure 2. Proof rules as bi-colored graph construction

6.1. GRAPHIC VIEW OF THE PROOF-SEARCH PROCESS

Let us fix a particular formula D. Using previous results, D is indexed into an equivalent flat sequent $\delta_D^- \vdash \Diamond \supset \mathcal{X}_D^-$.[9] Let us study the formulæ occurring in δ_D^-. From the definition of δ (see section 3), here are the formulæ in δ_D^- for each occurrence of a subformula of D:

- if V is a variable occurring positively (resp. negatively) in D, the formula $\mathcal{X}_V^+ \supset V$ (resp. $V \supset \mathcal{X}_V^-$) appears in δ_D^-. Also $\square \supset V$ appears in δ_D^- whether V occurs positively or negatively in D. These implications are already atomic and will not be further decomposed during proof-search;

- if $\bot$ is a positive (resp. negative) occurrence of the constant, the formula $\mathcal{X}_\bot^+ \supset \square$ (resp. $\square \supset \mathcal{X}_\bot^-$) appears in δ_D^-. Again, these implications will not be decomposed further during proof-search;

---

[9] For the sake of completeness, we recall the polarity of occurrences of subformulæ by suffixing their indexes like in $\mathcal{X}_K^+$ or $\mathcal{X}_K^-$. $\mathcal{X}_D^-$ has a negative polarity because it is the index of the root formula D and in δ_D^-, we start polarizing negatively.

- if $A \circledast B$ occurs positively (resp. negatively) in D, $\mathcal{X}^+_{A \circledast B} \supset (\mathcal{X}^+_A \circledast \mathcal{X}^+_B)$ (resp. $(\mathcal{X}^-_A \circledast \mathcal{X}^-_B) \supset \mathcal{X}^-_{A \circledast B}$) appears in δ^-_D and could be decomposed once using rule $[\supset_\circledast]$ (resp. $[_\circledast\supset]$), for $\circledast \in \{\wedge, \vee\}$;

- if $A \supset B$ occurs positively (resp. negatively), $\mathcal{X}^+_{A \supset B} \supset (\mathcal{X}^-_A \supset \mathcal{X}^+_B)$ (resp. $(\mathcal{X}^+_A \supset \mathcal{X}^-_B) \supset \mathcal{X}^-_{A \supset B}$) appears in δ^-_D and could be decomposed once using rule $[\supset_\supset]$ (resp. $[_\supset\supset]$).

What are the atomic implications occurring in the sequent obtained at the end of a completed proof-search branch starting from $\delta^-_D \vdash \Diamond \supset \mathcal{X}^-_D$? Some appear already at the root of the proof-search tree:

- $\mathcal{X}^+_V \supset V$, $V \supset \mathcal{X}^-_V$ and $\square \supset V$ on the left hand side of the $\vdash$ sign when V is an occurrence of a variable of D;

- $\mathcal{X}^+_\perp \supset \square$, $\square \supset \mathcal{X}^-_\perp$ on the left hand side of the $\vdash$ sign for occurrences of $\perp$ in D;

- $\Diamond \supset \mathcal{X}^-_D$ on the right hand side of the $\vdash$ sign.

These atomic implications are not decomposed by proof rules and are kept untouched from the root sequent $\delta^-_D \vdash \Diamond \supset \mathcal{X}^-_D$ to the implicational sequents at the leaves of the proof-search tree. So in the corresponding bi-colored graphs, there are arrows $\mathcal{X}^+_V \to V$, $V \to \mathcal{X}^-_V$, $\square \to V$, $\square \to \mathcal{X}^-_\perp$, $\mathcal{X}^+_\perp \to \square$ and $\mathcal{X}^-_D \Rightarrow \Diamond$. The situation is summarized in the figure below:
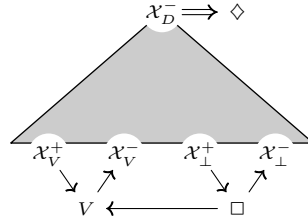
Now we consider what happens to the non-implicational formulæ of the form $(X \circledast Y) \supset Z$ or $X \supset (Y \circledast Z)$ in $\delta^-_D \vdash \Diamond \supset \mathcal{X}^-_D$. There is exactly one logical rule for each case: $[\supset_\circledast]$ for $\circledast^+$ and $[_\circledast\supset]$ for $\circledast^-$. The left column of Figure 2 presents all these cases. The complete set of arrow introduction rules is given in the right column of Figure 2. With this set of rules, each internal occurrence of a subformula introduces arrows in the bi-colored graph, depending on the choice of the left premise or right premise for the cases $\wedge^-$, $\vee^+$, $\supset^-$, and $\supset^+$. Since rules $\wedge^+$ and $\vee^-$ only have one premise, there is no choice in these cases and proof-search does not branch. So the end-sequent of a completed proof-search branch and its corresponding bi-colored graph is characterized

by a choice of left or right premise for internal nodes of the shape $\wedge^-$, $\vee^+$, $\supset^-$, and $\supset^+$.

The reader is reminded that the proof rules involved in proof-search are all strongly invertible and in fact, their are all *permutable*: the order in which they are applied does not influence the set of leaves of the proof-search tree.

6.2. CONDITIONAL BI-COLORED GRAPHS AND THEIR INSTANCES

To be able to represent all the proof-search branches in a common graph structure, we propose to represent the choice between the two premises of binary rules by a *boolean selector*, i.e. a boolean variable x which has two instances $x = 1$ and $x = 0$. For example, we obtain the following transformation of rule $[\supset_\vee]$ for a positive occurrence of disjunction $\vee^+$:
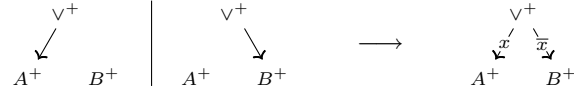

Instantiating the selectors, if we keep arrows only in case the selector evaluates to 1, then we see that when the selector x equals 1, then the selector $\overline{x} = 0$ and so, the left branch is selected, whereas when $x = 0$, then $\overline{x} = 1$ and the right branch is selected.

DEFINITION 6 (Conditional graph). *A conditional bi-colored graph is a bi-colored graph where arrows (either $\rightarrow$ or $\Rightarrow$) may be indexed with boolean selectors (like x or $\overline{x}$).*

Several boolean variables might be introduced for selectors because a new variable has to be chosen for each proof rule instance. Considering boolean selectors as representatives for boolean functions and given a valuation v, a boolean selector e is instantiated to the boolean value $[\![e]\!]_v \in \{0, 1\}$. We obtain an instance graph: an arrow indexed with a boolean expression e belongs to this instance if and only if $[\![e]\!]_v = 1$. The case of an unconditional (i.e. not indexed) arrow can be handled by considering that it has an implicit boolean condition of value 1 (which is the tautology that always evaluates to 1) and non-existing arrows have the implicit boolean condition of value 0 (that always evaluates to 0).

DEFINITION 7 (Instance graph). *Let $\mathcal{G}$ be a conditional bi-colored graph and v be a valuation of boolean variables in $\{0, 1\}$. The instance graph $\mathcal{G}_v$ is defined as the bi-colored graph obtained when one evaluates boolean selectors indexing arrows, keeping exactly those valued 1 in v.*

If we come back to the example of section 5.4, the following conditional bi-colored graph would be suited as a merge of the two bi-colored graphs corresponding to completed proof-search branches:
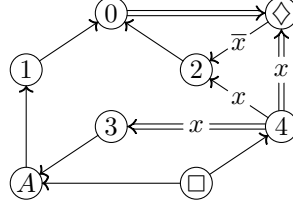
It visually appears that the two instances ($x = 1$ and $x = 0$) of this conditional graph are the bi-colored graphs of section 5.4.

7. Proof-search as the construction of a conditional graph

We introduce our counter-model search system which is a combination of the graphical system of Figure 2 and the idea of postponing choice of left or right premise by introducing selectors. In this system, there is no more branching in the search space.

The system is presented in Figure 3. We fix a formula D which is indexed and polarized starting from D^- and propagating the polarity through the connectives $\wedge,\vee$ and through the right branch of connective $\supset$. The polarity is inverted on the left branch of connective $\supset$.

We start from the leaves of the decomposition tree of D^- (V^+, V^-, $\perp^+$ or $\perp^-$). We add one new node for each variable V occurring in D.[10] We also add two nodes for $\square$ and $\diamondsuit$. Then, we add the arrow $D^- \Rightarrow \diamondsuit$ (D^- is the root of the decomposition tree) and arrows $V^+ \to V$ (resp. $V \to V^-$) for each positive (resp. negative) occurrence of the variable V. We add arrows $\square \to V$ for every variable occurring in D. We add $\perp^+ \to \square$ (resp. $\square \to \perp^-$) for each positive (resp. negative) occurrence of the constant $\perp$. This is summarized by the left side of Figure 3. All these arrows are not indexed with selectors: they are unconditional arrows.

Then for each internal node $\circledast^+$ or $\circledast^-$, we add arrows according to the schemes of the right part of Figure 3. There is one scheme for each case in $\{\wedge, \vee, \supset\} \times \{+, -\}$. For the schemes $\wedge^-$, $\vee^+$, $\supset^+$ and $\supset^-$, we introduce conditional arrows indexed with a boolean selector of the form x or $\overline{x}$. This selector has to be new: there is exactly one

---

[10] For a variable V occurring in D, there might be a positive occurrence V^+ or a negative occurrence V^- in the decomposition tree of D but there is only one extra node for the variable V, not one for each occurrence.
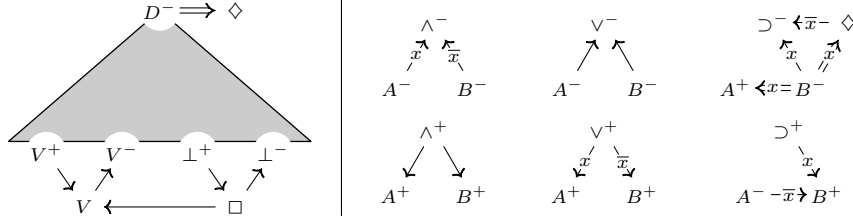
Figure 3. Counter-model search system for LC_n

selector for each instance of a scheme.[11] Each scheme introduces two or four arrows so the construction of the conditional bi-colored graph is a process which take linear time w.r.t. the size of D. The order in which we apply the schemes has no influence on the obtained conditional graph, up to the renaming of the boolean selectors.

THEOREM 10.   *Let D be a formula and $\mathcal{G}$ be the corresponding conditional bi-colored graph obtained by the previously described process. Then D has a counter model in LC if and only if there exists a valuation v on the selectors such that the instance graph $\mathcal{G}_v$ contains no $\Rightarrow$-cycle.*

   Proof. Because all the proof rules are strongly invertible, $[\![\cdot]\!]$ is a counter-model of D if and only if it is a counter-model of at least one of the implicational sequent at the leaf of a completed proof-search branch. This sequent is characterized by choices of left or right premises and these choices corresponds to a valuation on selectors. By Theorem 9, the leaf-sequent of a completed proof-search branch has a counter-model if and only if its associated bi-colored graph has no $\Rightarrow$-cycle. But this graph is exactly the instance $\mathcal{G}_v$ where v is the valuation on selectors corresponding to the proof-search branch.

THEOREM 11.   *D has a counter-model in LC_n iff there exists a valuation v on the selectors such that the instance graph $\mathcal{G}_v$ contains no $(n+1)$-alternating chain.*

   Proof. For LC_n with $n \neq \infty$, we just have to apply Theorem 8 instead of Theorem 9 used in the proof of Theorem 10. The proof rules are common for all the family LC_n (including $n = \infty$).

---

[11] For example, the selector could be indexed with the root node of the corresponding scheme instance to ensure uniqueness.

Back to the example of section 5.4, we present the conditional graph obtained for the formula $A \lor (A \supset \bot)$:
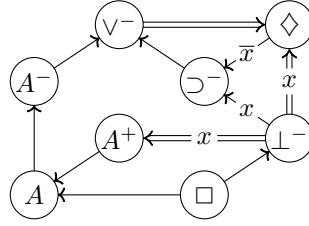
As expected, this conditional graph is isomorphic to the graph presented on page 23 which was a merge of the two proof-search branches of the decomposition of formula $A \lor (A \supset \bot)$.

As a trivial consequence of Theorems 5, 10 and 11, we obtain direct proof that deciding LC_n is a co-NP problem for any $n \in \overline{I\!N}^{\star}$.

Computing the conditional bi-colored graph from a given formula takes linear time but finding all the $\Rightarrow$-cycles in a graph and solving the boolean constraints system have both exponential complexity. Indeed, there exists formulæ for which the conditional bi-colored graph has exponentially many cycles. So on the complexity side, finding cycles and then building a constraint system is not a good idea.

We observe that we do not need the list of all the $\Rightarrow$-cycles to decide a formula in LC: we only need to compute and solve the boolean condition characterizing the existence of $\Rightarrow$-cycles.

8. Practical detection of alternating chains and $\Rightarrow$-cycles

To implement conditional graphs, we can represent the relations $\rightarrow$ and $\Rightarrow$ as generalized incidence matrices. In an incidence matrix, there is a 1 in a cell if the corresponding arrow exists in the graph. Generalized incidence matrix cells might contain not only 0 or 1 (for unconditional arrows) but also arbitrary boolean expressions built upon atomic boolean selectors. These expressions are considered up to boolean equivalence. For example, they could be implemented using a shared BDD as in [15].

DEFINITION 8 (Conditional matrix). *A conditional matrix on set $\mathcal{S}$ of size k is a $k \times k$ array with values in the free boolean algebra over the set of selectors.*

So a conditional bi-colored graph is viewed as a pair $(\rightarrow, \Rightarrow)$ of conditional matrices. The algebraic operations we have defined like

$\to + \Rightarrow$, $\to \cdot \Rightarrow$, $\to^\star$, $tr(\cdot)$ and $\sum(\cdot)$ extend naturally[12] to conditional matrices because they rely only on the boolean operators $\cdot$ and $+$. If v is a valuation of boolean variables in $\{0,1\}$, we obtain *instance matrices* corresponding to instance graphs. Instantiation commutes with algebraic operations on matrices because it commutes with the boolean operators $\cdot$ and $+$. For example $[\to + \Rightarrow]_v = \to_v + \Rightarrow_v$.

THEOREM 12. *Let $\mathcal{G} = (\to, \Rightarrow)$ be a conditional bi-colored graph. There exists a $\Rightarrow$-cycle in every instance $\mathcal{G}_v$ of $\mathcal{G}$ if and only if the following identity between boolean expressions holds:*

$$tr((\to + \Rightarrow)^\star \Rightarrow) = 1$$

Proof. Straightforward: $[tr((\to + \Rightarrow)^\star \Rightarrow)]_v = tr((\to_v + \Rightarrow_v)^\star \Rightarrow_v)$ and then, application of Theorem 10.

The reader is reminded that the equation $tr((\to + \Rightarrow)^\star \Rightarrow) = 1$ is an equivalence of boolean expressions, not an equality between boolean value in $\{0,1\}$. The result for the finitary version LC_n is the following:

THEOREM 13. *Let $\mathcal{G} = (\to, \Rightarrow)$ be a conditional bi-colored graph. There exists an instance graph $\mathcal{G}_v$ of $\mathcal{G}$ containing no $(n+1)$-alternating chain if and only if the following inequality between boolean expressions holds:*

$$\sum(\to^\star \Rightarrow)^{n+1} \neq 1$$

To compute the smallest n such that a formula is not valid in LC_n, we sequentially compute until $\sum(\to^\star \Rightarrow)^{n+1} \neq 1$ and then, find a valuation v such that the instance is refuted: $\sum(\to_v^\star \Rightarrow_v)^{n+1} = 0$. From the corresponding instance graph $\mathcal{G}_v$, we extract a counter-model in LC_n.

We have implemented in Objective Caml[13] a prototype of counter-model search engine for LC based on these principles. Implementation details of this system are described in full details in [15].

9. Comparison with existing systems

The sequent or hyper-sequent calculi proposed for LC can be distinguished by the shape of their irreducible sequents.

---

[12] The infinitary $(\cdot)^\star$ extends as well because as noticed earlier, $\mathcal{B}^{k \times k}$ is a finite point-wise ordered poset whenever $\mathcal{B}$ is $\{0,1\}$ or any other finite boolean algebra like the free boolean algebra built over a finite set of selectors.

[13] It is available at `http://www.loria.fr/~larchey/LC`.

Indeed, Dyckhoff [9] does not focus on the efficiency of the decision procedure corresponding to his sequent system. But, to our knowledge, it was the first (purely) sequent system for LC where all the proof rules are invertible. Unfortunately, the proof-search algorithm tries to reduce implicational sequents but they are not irreducible in this system. Proof-search procedures are not efficient at deciding implicational sequents and can take exponential time whereas a counter-model search as the one presented in this paper takes linear time.

We have already noticed and solved this problem in [12] with a sequent calculus that does not try to reduce implicational sequents. A counter-model search algorithm with linear complexity based on a fix-point computation is provided to solve irreducible sequents which are composed of atomic implications and variables. Unfortunately, it is not easy to guess how to combine the proof-search branches in a common structure on which we could search for counter-models simultaneously on every branch.

The hyper-sequent calculi popularized by Avron [2] and Baaz [5] also have irreducible sequents composed of atomic implication and variables. They associate these irreducible sequents with constraint systems and search for G-cycles in those constraints. The concept of G-cycles is close to the one of $\Rightarrow$-cycle but it is not expressed in graph theoretical terms and no algorithm is explicitly given to solve these constraints. Only the fact that they are polynomially solvable is stated. In [6], using relational hyper-sequents, they obtain another kind of constraints which is proved to be polynomially solvable by linear programming techniques. In this paper, we provide a linear algorithm to solve these kind of constraints.

Because of the choice of hyper-sequents and the fact that constraints are not visualized in graphs, it seems that the problem of trying to merge proof-search branches did not appear as an obvious extension of the work on either hyper-sequents or even relational (hyper-)sequents. It might be possible to obtain such systems in which proof-search branches could be merged into a common graph structure or constraint system, but we think that the proof rules need to be tailored towards such a goal as we have done in the case of the sequent calculus.

Finally, our system, first presented in [13], is based on the idea that we should not try to decompose atomic implications and that we should have the structure of irreducible sequents as simple as possible. It was thus a straightforward idea to encode a variable X with an atomic implication $\Diamond \supset X$. What remained to be done was to provide strongly invertible proof-rules for such sequents. Once proof-rules for these uniform irreducible sequents were obtained, the bi-colored graph structure became obvious and the decision algorithm followed.

References

1. Avellone, A., M. Ferrari, and P. Miglioli: 1999, 'Duplication-Free Tableau Calculi and Related Cut-Free Sequent Calculi for the Interpolable Propositional Intermediate Logics'. *Logic Journal of the IGPL* **7**(4), 447–480.
2. Avron, A.: 2000, 'A Tableau System for Gödel-Dummett Logic Based on a Hypersequent Calculus'. In: *TABLEAUX 2000*, Vol. 1847 of *LNAI*. pp. 98–111.
3. Avron, A. and B. Konikowska: 2001, 'Decomposition Proof Systems for Gödel-Dummett Logics'. *Studia Logica* **69**(2), 197–219.
4. Baaz, M., A. Ciabattoni, and C. Fermüller: 2001, 'Cut-Elimination in a Sequents-of-Relations Calculus for Gödel Logic'. In: *ISMVL 2001*. pp. 181–186.
5. Baaz, M., A. Ciabattoni, and C. Fermüller: 2003, 'Hypersequent Calculi for Gödel Logics – A Survey'. *Journal of Logic and Computation* **13**(6), 835–861.
6. Ciabattoni, A., C. Fermüller, and G. Metcalfe: 2004, 'Uniform Rules and Dialogue Games for Fuzzy Logics'. In: *LPAR 2004*, Vol. 3452 of *LNCS*. pp. 496–510.
7. Dummett, M.: 1959, 'A Propositional Calculus with a Denumerable matrix'. *Journal of Symbolic Logic* **24**, 96–107.
8. Dyckhoff, R.: 1992, 'Contraction-free Sequent Calculi for Intuitionistic Logic'. *Journal of Symbolic Logic* **57**(3), 795–807.
9. Dyckhoff, R.: 1999, 'A Deterministic Terminating Sequent Calculus for Gödel-Dummett logic'. *Logical Journal of the IGPL* **7**(3), 319–326.
10. Gödel, K.: 1932, 'Zum intuitionistischen Aussagenkalkül'. In: *Anzeiger Akademie des Wissenschaften Wien*, Vol. 69. pp. 65–66.
11. Hajek, P.: 1998, *Metamathematics of Fuzzy Logic*. Kluwer Academic Publishers.
12. Larchey-Wendling, D.: 2002, 'Combining Proof-Search and Counter-Model Construction for Deciding Gödel-Dummett Logic'. In: *CADE-18*, Vol. 2392 of *LNAI*. pp. 94–110.
13. Larchey-Wendling, D.: 2004, 'Counter-model search in Gödel-Dummett logics'. In: *IJCAR 2004*, Vol. 3097 of *LNAI*. pp. 274–288.
14. Larchey-Wendling, D.: 2005a, 'Bounding Resource Consumption with Gödel-Dummett Logics'. In: *LPAR 2005*, Vol. 3835 of *LNAI*. pp. 682–696.
15. Larchey-Wendling, D.: 2005b, 'Gödel-Dummett counter-models through matrix computation'. *Electronic Notes in Theoretical Computer Science* **125**(3), 137–148.
16. Metcalfe, G., N. Olivetti, and D. Gabbay: 2003, 'Goal-Directed Calculi for Gödel-Dummett Logics'. In: *CSL'03*, Vol. 2803 of *LNCS*. pp. 413–426.
17. Warshall, S.: 1962, 'A Theorem on Boolean Matrices'. *J. ACM* **9**(1), 11–12.