# Counter-model search in Gödel-Dummett logics

Dominique Larchey-Wendling

LORIA – CNRS
Campus Scientifique, BP 239
Vandœuvre-lès-Nancy, France

**Abstract.** We present a new method for deciding Gödel-Dummett logic $\mathsf{LC}$. We first characterize the validity of irreducible sequents of $\mathsf{LC}$ by the existence of r-cycles in bi-colored graphs and we propose a linear algorithm to detect r-cycles and build counter-models. Then we characterize the validity of formulae by the existence of r-cycles in boolean constrained graphs. We also give a parallel method to detect r-cycles under boolean constraints. Similar results are given for the finitary versions $\mathsf{LC}_n$.

## 1 Introduction

Gödel-Dummett logic $\mathsf{LC}$ and its finitary versions $(\mathsf{LC}_n)_{n>0}$ are the intermediate logics (between classical and intuitionistic logics) characterized by linear Kripke models. $\mathsf{LC}$ was introduced by Gödel in [10] and later axiomatized by Dummett in [6]. It is now one of the most studied intermediate logics and has been recognized recently as one of the fundamental *t-norm based fuzzy logics* [11]. Proof-search in $\mathsf{LC}$ has benefited from the development of proof-search in intuitionistic logic $\mathsf{IL}$ with two important seeds: the *contraction-free calculus* of Dyckhoff [1,7,8] and the *hyper-sequent* calculus of Avron [2,13]. Two of the most recent contributions propose a similar approach based on a set of *local* and *strongly invertible* proof rules (for either sequent [12] or hyper-sequent [2] calculus,) and a semantic criterion to decide *irreducible (hyper)-sequents* and eventually build a counter-model.

We are interested in studying combination of proof-search and counter-model construction to provide decision procedures for intermediate logics. We have already proposed such a combination for $\mathsf{LC}$ [12], but here we investigate deeper counter-model search to obtain a new system with the following fundamental property: all the irreducible sequents arising from a proof-search can be represented in a shared semantic structure. The semantic criterion that decides irreducible sequents can be computed in parallel on this shared structure. Instead of simply combining proof-search and counter-model construction, we are now at the edge of these two techniques and then provide a calculus of *parallel counter-model search.*

In section 3, we recall the indexation technique of [12] and introduce a new proof-system with the property that irreducible sequents are now composed only of atomic implications. In section 4 we associate a *bi-colored graph* to each irreducible sequent. We show that their validity in $\mathsf{LC}$ can be decided from the existence of *r-cycles* in the graph. We propose a linear time algorithm to detect r-cycles and eventually build counter-models. We prove similar results for $\mathsf{LC}_n$. In section 5, we show that proof-search can be viewed as a *non-deterministic choice* of edges in a bi-colored graph corresponding to left or right premises of proof-rules. We postpone non-deterministic choices by introducing *boolean selectors* and represent proof-search by *conditional bi-colored graph*. We prove that validity is characterized by the existence of r-cycles in every instance of this graph and discuss r-cycle detection combined with boolean constraints solving. Moreover, we characterize the smallest $n$ for which a given formula is invalid in $\mathsf{LC}_n$. In section 6, we detail some implementation techniques for a constrained r-cycle detection algorithm based on *matrices of binary decision diagrams* and briefly present our *parallel counter-model search* system for $\mathsf{LC}$ but also for its finitary versions $\mathsf{LC}_n$.

## 2 Introduction to Gödel-Dummett logics $\mathsf{LC}_n$

In this section, we present the algebraic semantics of the family of propositional Gödel-Dummett logics $\mathsf{LC}_n$. The value $n$ belongs to $\overline{\mathbb{N}}^\star = \mathbb{N}^\star \cup \{\infty\}$ with its natural order $\leqslant$ augmented with a greatest element $\infty$. In the case $n = \infty$, the logic $\mathsf{LC}_\infty$ is also denoted by $\mathsf{LC}$: this is the usual Gödel-Dummett logic.

The set of propositional *formulae*, denoted $\mathsf{Form}$ is defined inductively, starting from a set of propositional *variables* denoted by $\mathsf{Var}$ with an additional bottom constant $\perp$ denoting *absurdity* and using the connectives $\wedge$, $\vee$ and $\supset$. A *substitution* denoted by $\sigma$ is any function that associates a formula to every propositional variable. We denote by $A_\sigma$ the result of the application of $\sigma$ to the variables in $A$. $\mathsf{IL}$ will denote the set of formulae that are provable in any intuitionistic propositional calculus (see [7]) and $\mathsf{CL}$ will denote the classically valid formulae. As usual an *intermediate propositional logic* [1] is a set of formulae $\mathcal{L}$ 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 arbitrary substitution (if $A \in \mathcal{L}$ and $\sigma$ is any substitution then $A_\sigma \in \mathcal{L}$.)

For any $n \in \overline{\mathbb{N}}^\star$, the Gödel-Dummett logic $\mathsf{LC}_n$ is an intermediate logic. On the semantic side, it is characterized by the linear Kripke model of size $n$ (see [6].) The following sequence holds:

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

In the particular case of $\mathsf{LC}$, the logic has a simple Hilbert axiomatic system: $(X \supset Y) \vee (Y \supset X)$ added to the axioms of $\mathsf{IL}$.

In this paper, we will use the algebraic semantics characterization of $\mathsf{LC}_n$ [2] rather than Kripke semantics. Let us fix a particular $n \in \overline{\mathbb{N}}^\star$. The algebraic model is the set $[0, n[\cup\{\infty\}$ composed of $n+1$ elements.[1] An interpretation of propositional variables $[\![\cdot]\!] : \mathsf{Var} \rightarrow [0, n[\cup\{\infty\}$ is inductively extended to formulae:

[1] With our particular representation, the algebraic models $[0, n[\cup\{\infty\}$ form a strictly increasing sequence of subsets of $\overline{\mathbb{N}}$.

$\perp$ 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 $\rightarrow$ defined by $a \rightarrow b =$ if $a \leqslant b$ then $\infty$ else $b$. A formula $D$ is *valid* for the interpretation $[\![\cdot]\!]$ if the equality $[\![D]\!] = \infty$ holds. This interpretation is complete for $\mathsf{LC}$. A *counter-model* of a formula $D$ is an interpretation $[\![\cdot]\!]$ such that $[\![D]\!] < \infty$. A *sequent* is a pair $\Gamma \vdash \Delta$ where $\Gamma$ and $\Delta$ are multisets of formulae. $\Gamma, \Delta$ denotes the sum of the two multisets and if $\Gamma$ is the empty multiset, we write $\vdash \Delta$. Given a sequent $\Gamma \vdash \Delta$ and an interpretation $[\![\cdot]\!]$ of variables, we interpret $\Gamma \equiv A_1, \ldots, A_n$ by $\llbracket \Gamma \rrbracket = [\![A_1]\!] \wedge \cdots \wedge [\![A_n]\!]$ and $\Delta \equiv B_1, \ldots, B_p$ by $\llbracket \Delta \rrbracket = [\![B_1]\!] \vee \cdots \vee [\![B_p]\!]$. This sequent is *valid* with respect to the interpretation $[\![\cdot]\!]$ if $\llbracket \Gamma \rrbracket \leqslant \llbracket \Delta \rrbracket$ holds. On the other hand, a *counter-model* to this sequent is an interpretation $[\![\cdot]\!]$ such that $\llbracket \Delta \rrbracket < \llbracket \Gamma \rrbracket$, i.e. for any pair $(i, j)$, the inequality $[\![B_j]\!] < [\![A_i]\!]$ holds.

## 3 Proof-search and sequent calculus for $\mathsf{LC}_n$

In this section we present a refinement of the decision procedure described in [12]. As before, we have the three following steps: first a linear reduction of a formula $D$ into a *flat sequent* $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}_D$, then a sequent based proof-search that reduces flat sequents into a set of *implicational sequents* $\Gamma_i \vdash \Delta_b$ and finally a semantic algorithm that builds counter-models of $\Gamma_i \vdash \Delta_b$.

In [12], the proof-search process produced what we called pseudo-atomic sequents, which are sequents composed only of formulae of the form $X$ and $X \supset Y$ where $X$ and $Y$ are logical variables. In our new system, we further restrict our proof-search procedure to only produce atomic implications $X \supset Y$. We do not integrate the $\perp$ constant but it can be treated as in [12]. So atomic formulae are variables. We introduce a new special variable $\diamondsuit \notin \mathsf{Var}$. From now, (propositional) variables are elements of $\mathsf{Var} \cup \{\diamondsuit\}$ but we require that $\diamondsuit$ does occur in the formulae we are trying to decide, so it has a special meaning for proof-search.

**Definition 1 (B-contexts).** *A* b-context *denoted $\Delta_b$ is a non-empty multiset of implications such that if $A \supset B \in \Delta_b$ then $\diamondsuit \supset B \in \Delta_b$.*

### 3.1 Linear reduction through indexation

We refine our indexation technique and the notion of flat sequent.

**Definition 2 (Flat formula and sequent).** *A* flat formula *is of one of the following forms: $X \supset Y$ or $Z \supset (X \otimes Y)$ or $(X \otimes Y) \supset Z$ where $X$, $Y$ and $Z$ are propositional variables and $\otimes \in \{\wedge, \vee, \supset\}$. $\Gamma \vdash \Delta_b$ is a* flat sequent *if all formulae of $\Gamma$ are flat and $\Delta_b$ is a b-context of atomic implications.*

Let us fix a particular formula $D$ (not containing $\diamondsuit$.) We introduce a new variable $\mathcal{X}_A$ (not occurring in $D$ and different from $\diamondsuit$) for every occurrence $A$ of subformula of $D$. So the variables $\mathcal{X}_A$ represent the nodes of the decomposition

$$\frac{\Gamma, A \supset C \vdash \Delta \qquad \Gamma, B \supset C \vdash \Delta}{\Gamma, (A \wedge B) \supset C \vdash \Delta}\,[\supset_2] \qquad \frac{\Gamma, A \supset B, A \supset C \vdash \Delta}{\Gamma, A \supset (B \wedge C) \vdash \Delta}\,[\supset_2']$$

$$\frac{\Gamma, A \supset C, B \supset C \vdash \Delta}{\Gamma, (A \vee B) \supset C \vdash \Delta}\,[\supset_3] \qquad \frac{\Gamma, A \supset B \vdash \Delta \qquad \Gamma, A \supset C \vdash \Delta}{\Gamma, A \supset (B \vee C) \vdash \Delta}\,[\supset_3']$$

$$\frac{\Gamma, A \supset C \vdash \Delta \qquad \Gamma, B \supset C \vdash \Delta}{\Gamma, A \supset (B \supset C) \vdash \Delta}\,[\supset_4']$$

**Fig. 1.** Proof rules for implications in $\mathsf{LC}_n$.

tree of $D$. Then, we define the multisets $\delta^+(K)$ and $\delta^-(K)$ by induction on the subformula $K$. These two multisets are composed of only flat formulae with variables of the form $\mathcal{X}_A$ or $V$, for $V$ contained in $D$:

$$\begin{array}{l}
\delta^+(V) = \mathcal{X}_V \supset V \text{ when } V \text{ is a variable} \\
\delta^+(A \otimes B) = \delta^+(A), \delta^+(B), \mathcal{X}_{A\otimes B} \supset (\mathcal{X}_A \otimes \mathcal{X}_B) \text{ when } \otimes \in \{\wedge, \vee\} \\
\delta^+(A \supset B) = \delta^-(A), \delta^+(B), \mathcal{X}_{A\supset B} \supset (\mathcal{X}_A \supset \mathcal{X}_B) \\
\delta^-(V) = V \supset \mathcal{X}_V \text{ when } V \text{ is a variable} \\
\delta^-(A \otimes B) = \delta^-(A), \delta^-(B), (\mathcal{X}_A \otimes \mathcal{X}_B) \supset \mathcal{X}_{A\otimes B} \text{ when } \otimes \in \{\wedge, \vee\} \\
\delta^-(A \supset B) = \delta^+(A), \delta^-(B), (\mathcal{X}_A \supset \mathcal{X}_B) \supset \mathcal{X}_{A\supset B}
\end{array}$$

We just recall the proposition 1 already proved in [12]. We also introduce the proposition 2 which is the semantic counterpart of the substitution property of intermediate logics.

**Proposition 1.** *Both $\delta^+(D), \mathcal{X}_D \vdash D$ and $\delta^-(D), D \vdash \mathcal{X}_D$ are valid in $\mathsf{LC}_n$.*

**Proposition 2.** *If $[\![\cdot]\!]$ is such that $[\![\mathcal{X}_K]\!] = [\![K]\!]$ holds for any occurrence $K$ of subformula of $D$, then equation $[\![\delta^-(D)]\!] = [\![\delta^+(D)]\!] = \infty$ holds.*

Both proofs are left to the reader (trivial induction on $K$, see [12]). Then we are able to propose our linear reduction of a formula into a flat sequent. The proof of the following result can be found in appendix A.

**Theorem 1.** *In $\mathsf{LC}_n$, $D$ is valid if and only if $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}_D$ is valid.*

### 3.2 Reducing flat sequents by proof search

In figure 1, we recall proof rules for $\mathsf{LC}_n$ [12]. All of them are sound and strongly invertible. We replace the "old" rule $[\supset_4]$ (not included in the figure) with a new version:

$$\frac{\Gamma, B \supset C \vdash A \supset B, \diamondsuit \supset B, \Delta_b \qquad \Gamma, \diamondsuit \supset C \vdash \Delta_b}{\Gamma, (A \supset B) \supset C \vdash \Delta_b}\,[\supset_4] \quad \Delta_b \text{ is a b-context}$$

**Theorem 2.** *In $\mathsf{LC}_n$, rule $[\supset_4]$ is sound and strongly invertible.*

This theorem is proved in appendix A. The aim is that applied backward, all proof rules only introduce implications. Applying rules $[\supset_2]$, $[\supset_3]$, $[\supset_4]$ and $[\supset_2']$, $[\supset_3']$, $[\supset_4']$ bottom-up, it is easy to see that the proof-search process applied to any flat sequent terminates and produces a set of irreducible sequents, meaning that no more rules can be applied to them.

**Definition 3 (Implicational sequent).** *A* implicational sequent *is of the form $X_1 \supset Y_1, \ldots, X_k \supset Y_k \vdash A_1 \supset B_1, \ldots, A_l \supset B_l$, all the $X_i, Y_i, A_i, B_i$ are variables.*

As, there is one rule for each case of $(X \otimes Y) \supset Z$ or $Z \supset (X \otimes Y)$ and as the corresponding rule transforms the compound formula into one, two or three atomic implications in each premise, irreducible sequents are implicational sequents. Moreover, in an irreducible sequent $\Gamma \vdash \Delta$, $\Delta$ is also a b-context. We point out that there is no more a distinction between $\diamondsuit$ and other variables in the notion of implicational sequent: the notion of b-context is only useful during proof-search. Section 5.1 discusses the proof-search process in further details.

## 4 Counter-models of implicational sequents

In this section, we present a new and visual criterion to decide implicational sequents. In particular, the sequents that arise as irreducible for the rules of the preceding section are implicational.

### 4.1 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 *reverse* direction than the implication $A_i \supset B_i$.

We will often use the 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 $X \rightarrow\Rightarrow Y$ means there exists a chain $X \rightarrow Z \Rightarrow Y$ 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.

### 4.2 Heights in bi-colored graphs

We define a notion of bi-height in bi-colored graphs. The idea is very simple. A green arrow $\rightarrow$ weights 0 and a red arrow $\Rightarrow$ weights at least 1.

**Definition 4 (Bi-height in a bi-colored graph).** *Let $\mathcal{G}$ be a bi-colored graph. A* bi-height *is a function $h : \mathcal{G} \rightarrow \mathbb{N}$ such that for any $x, y \in \mathcal{G}$, if $x \rightarrow y \in \mathcal{G}$ then $h(x) \leqslant h(y)$ and if $x \Rightarrow y \in \mathcal{G}$ then $h(x) < h(y)$.*

As we will see, bi-height can be used to compute counter-models. We characterize graphs that admit bi-heights by the notion of r-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.

**Definition 5 (R-cycle).** *A* r-cycle *is a chain of the form* $x(\to + \Rightarrow)^\star \Rightarrow x$.

It is clear that if a graph has a r-cycle, then there is no bi-height. We give a linear[2] algorithm to compute a bi-height when there is no r-cycle.

**Theorem 3.** *Let $\mathcal{G}$ be a bi-colored graph. It is possible to decide if $\mathcal{G}$ has r-cycles in linear time and if not, it is possible to compute a bi-height $h$ for $\mathcal{G}$ in linear time.*

**Proof.** Even if it has no r-cycle, $\mathcal{G}$ may still contain green ($\to$) 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. $\mathcal{G}_\to$), $\mathcal{C} = \{[x] \mid x \in \mathcal{G}\}$ and $[x]$ is the strongly connected component of $x$. $\mathcal{G}'$ has $\mathcal{C}$ as set of nodes, and the set of arrows is described by:

$$[x] \to [y] \text{ iff } [x] \neq [y] \text{ and } \exists x', y' \text{ s.t. } [x] = [x'], [y] = [y'] \text{ and } x' \to y' \in \mathcal{G}$$
$$[x] \Rightarrow [y] \text{ iff } \exists x', y' \text{ s.t. } [x] = [x'], [y] = [y'] \text{ and } x' \Rightarrow y' \in \mathcal{G}$$

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

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

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

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

**Theorem 4.** *If $\mathcal{G}$ has no chain of type $(\to^\star \Rightarrow)^n$ then the height $h$ of theorem 3 satisfies $\forall x \in \mathcal{G},\ h(x) < n$.*

**Proof.** In $\mathcal{G}'$, if $h'([y]) = n$ then there exists a chain $[x](\Rightarrow \to^\star)^n[y]$ in $\mathcal{G}'$. This result is straightforwardly proved by induction on $n$. Then suppose that their exists $y$ such that $h(y) = m \geqslant n$. 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 $\mathcal{G}$. It contains a subchain of type $(\to^\star \Rightarrow)^n$. □

[2] 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 with no r-cycle. Then, for $n$ greater than the number of nodes of $\mathcal{G}$, the graph $\mathcal{G}$ has no chain of the form $(\rightarrow^\star\Rightarrow)^n$.*

**Proof.** Let $s$ be the number of nodes of $\mathcal{G}$. Let $n$ be greater than $s$, $n \geqslant s$. Suppose that $\mathcal{G}$ has a chain of the form $x_0 \rightarrow^\star \Rightarrow x_1 \rightarrow^\star \Rightarrow \cdots \rightarrow^\star \Rightarrow x_n$. If all the $x_i$ are different then the set $\{x_0, x_1, \ldots, x_n\}$ contains $n+1 > s$ nodes of $\mathcal{G}$. This is not possible. So let $i < j$ be such that $x_i = x_j$. The chain $x_i \rightarrow^\star \Rightarrow \cdots \rightarrow^\star \Rightarrow x_j$ is a r-cycle. □

### 4.3 Counter-models vs. chains in bi-colored graphs

**Lemma 1.** *Let $\mathcal{S}$ be an implicational sequent and $\mathcal{G}_\mathcal{S}$ its associated bi-colored graph. Let $[\![\cdot]\!] : \mathsf{Var} \rightarrow [0,n[\cup\{\infty\}$ be a counter-model of $\mathcal{S}$ in $\mathsf{LC}_n$ and $X_1 \rightarrow \cdots \rightarrow X_k \Rightarrow Y$ a 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 $\lceil\!\lceil\Delta\rceil\!\rceil < \lfloor\!\lfloor\Gamma\rfloor\!\rfloor$ holds. As $X_k \Rightarrow Y \in \mathcal{G}_\mathcal{S}$, the formula $Y \supset X_k$ is an element of $\Delta$. So we deduce $[\![Y \supset X_k]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil < \infty$. Thus we obtain $[\![Y]\!] > [\![X_k]\!]$ and $[\![X_k]\!] = [\![Y \supset X_n]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil$. Then $X_{k-1} \rightarrow X_k \in \mathcal{G}_\mathcal{S}$, so $X_{k-1} \supset X_k$ belongs to $\Delta$. Thus $[\![X_k]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil < \lfloor\!\lfloor\Gamma\rfloor\!\rfloor \leqslant [\![X_{k-1} \supset X_k]\!]$ 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 $[\![X_{k-1}]\!] \leqslant [\![X_k]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil$. By the same argument as in the preceding paragraph, we can prove that $[\![X_{i-1}]\!] \leqslant [\![X_i]\!] \leqslant \lceil\!\lceil\Delta\rceil\!\rceil$ by descending induction on $i$ for $k-1, k-2, \ldots, 2$. □

**Theorem 6.** *For $n < \infty$, the implicational sequent $\mathcal{S}$ has a counter-model in $\mathsf{LC}_n$ if and only if its associated graph $\mathcal{G}_\mathcal{S}$ contains no chain of type $(\rightarrow^\star\Rightarrow)^{n+1}$.*

**Proof.** Let $\mathcal{S} = \Gamma \vdash \Delta$. First we prove the if part. We suppose that $\mathcal{G}_\mathcal{S}$ contains no chain of the form $(\rightarrow^\star\Rightarrow)^{n+1}$. Then by theorem 4, there exists a bi-height function $h : \mathcal{G}_\mathcal{S} \rightarrow [0,n]$. We define the semantic function $[\![X]\!] \in [0,n[\cup\{\infty\}$ on the variables occurring in $\mathcal{S}$ which are the nodes of $\mathcal{G}_\mathcal{S}$ by $[\![X]\!] = h(X)$ if $h(X) < n$ and $[\![X]\!] = \infty$ if $h(X) = n$. Then, let us prove that $[\![\cdot]\!]$ is a counter-model of $\mathcal{S}$. Indeed, if $X \supset Y \in \Gamma$ then $X \rightarrow Y \in \mathcal{G}_\mathcal{S}$ and then $h(X) \leqslant h(Y)$. It follows that $[\![X]\!] \leqslant [\![Y]\!]$ and so $[\![X \supset Y]\!] = \infty$. We have $\lfloor\!\lfloor\Gamma\rfloor\!\rfloor = \infty$. If $X \supset Y \in \Delta$ then $Y \Rightarrow X \in \mathcal{G}_\mathcal{S}$. Thus $h(Y) < h(X)$ and $[\![X \supset Y]\!] = [\![Y]\!] = h(Y) < h(X) \leqslant n$. $[\![X \supset Y]\!] \leqslant n-1$ holds and so $\lceil\!\lceil\Delta\rceil\!\rceil \leqslant n-1$ holds. Finally, $\lceil\!\lceil\Delta\rceil\!\rceil \leqslant n-1 < \infty = \lfloor\!\lfloor\Gamma\rfloor\!\rfloor$ so $[\![\cdot]\!]$ is a counter-model of $\mathcal{S}$.

Then we prove the only if part of the theorem. Let $[\![\cdot]\!] : \mathsf{Var} \rightarrow [0,n[\cup\{\infty\}$ being a counter-model of $\mathcal{S}$. Suppose there is a chain of the form $(\rightarrow^\star\Rightarrow)^{n+1}$ in $\mathcal{G}_\mathcal{S}$: $X_0 \rightarrow^\star \Rightarrow X_1 \rightarrow^\star \Rightarrow X_2 \rightarrow^\star \Rightarrow \cdots \rightarrow^\star \Rightarrow X_n \rightarrow^\star \Rightarrow X_{n+1}$. So for any $i$, there is a chain $X_i \rightarrow^\star \Rightarrow X_{i+1}$ and by lemma 1, 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 $[0,n[\cup\{\infty\}$. As this set has $n+1$ elements, we get a contradiction. □

**Theorem 7.** *An implicational sequent $\mathcal{S}$ has a counter-model in $\mathsf{LC}$ if and only if its associated graph $\mathcal{G}_\mathcal{S}$ has no r-cycle.*

**Proof.** For the if part, if $\mathcal{G}_{\mathcal{S}}$ has no r-cycle, by theorem 3, there is a height $h : \mathcal{G}_{\mathcal{S}} \rightarrow \mathbb{N}$. We define $[\![X]\!] \in \mathbb{N} \cup \{\infty\}$ by $[\![X]\!] = h(X)$ and obtain a counter-model of $\mathcal{S}$ in $\mathsf{LC}$. For the only if part, the existence of a chain $X \rightarrow^{\star} \Rightarrow \rightarrow^{\star} \Rightarrow \cdots \rightarrow^{\star} \Rightarrow X$ would lead to $[\![X]\!] < [\![X]\!]$ by the same argument as before. □

### 4.4 Algebraic criteria and the limit counter-model

We have seen that the existence of counter-models of implicational sequents is equivalent to the lack of r-cycles in the associated bi-colored graphs. Now we present an algebraic formula that expresses the existence of r-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 lines 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 $(x, y)$ in the matrix of $\rightarrow$ if and only if $x \rightarrow y \in \mathcal{G}$. 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 boolean matrices.

Now 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 matrix $\sum_{i \geqslant 0} \rightarrow^{i}$. So $((\rightarrow + \Rightarrow)^{\star} \Rightarrow)_{x,x} = 1$ means that there exists a chain of the form $x(\rightarrow + \Rightarrow)^{\star} \Rightarrow x$ in the graph $\mathcal{G}$. Let $\mathrm{tr}(\cdot)$ denote the trace of matrices defined by $\mathrm{tr}(M) = \sum_x M_{x,x}$.

**Proposition 3.** *$\mathcal{G}$ has a r-cycle if and only if* $tr\big((\rightarrow + \Rightarrow)^{\star} \Rightarrow\big) = 1$.

In section 6, we will explain how to compute this trace efficiently. We conclude this section by a criterion to determine the minimal $n$ for which a given sequent $\mathcal{S}$ has a counter-model in $\mathsf{LC}_n$. Let $\sum M$ denote the sum of all the elements of the matrix $M$ defined by $\sum M = \sum_{x,y} M_{x,y}$.

**Proposition 4.** *Let $\mathcal{S}$ be an implicational sequent not valid in $\mathsf{LC}$. There exists a minimal $n$ such that $\mathcal{S}$ has a counter-model in $\mathsf{LC}_n$ and it is the first $n$ s.t.* $\sum(\rightarrow^{\star}\Rightarrow)^{n+1} = 0$.

**Proof.** By theorem 6, $\mathcal{S}$ has a counter-model in $\mathsf{LC}_n$, 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 this relation is the zero matrix, i.e. $\sum(\rightarrow^{\star}\Rightarrow)^{n+1} = 0$. As $\mathcal{S}$ is not valid in $\mathsf{LC}$, by theorem 7, its bi-colored graph $\mathcal{G}_{\mathcal{S}}$ has no r-cycle. Then, there is an $n$ such that $\mathcal{G}_{\mathcal{S}}$ has no chain of the form $(\rightarrow^{\star}\Rightarrow)^{n}$ by theorem 5. So there exists a minimal one. □

## 5 Parallel counter-model search

Combining the results of the preceding sections provides an algorithm to decide $\mathsf{LC}_n$ by indexation followed by proof-search to obtain a set of irreducible

| | | |
|---|---|---|
| $\dfrac{\Gamma, \mathcal{X}_A^- \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_b \qquad \Gamma, \mathcal{X}_B^- \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_b}{\Gamma, (\mathcal{X}_A^- \wedge \mathcal{X}_B^-) \supset \mathcal{X}_{A\wedge B}^- \vdash \Delta_b}\ [\supset_2]$ | $\begin{tikzcd}[column sep=small] & \wedge^- & \\ A^- \arrow[ur] & & B^- \end{tikzcd}$ | $\begin{tikzcd}[column sep=small] & \wedge^- & \\ A^- & & B^- \arrow[ul] \end{tikzcd}$ |
| $\dfrac{\Gamma, \mathcal{X}_{A\wedge B}^+ \supset \mathcal{X}_A^+, \mathcal{X}_{A\wedge B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_b}{\Gamma, \mathcal{X}_{A\wedge B}^+ \supset (\mathcal{X}_A^+ \wedge \mathcal{X}_B^+) \vdash \Delta_b}\ [\supset_2']$ | $\begin{tikzcd}[column sep=small] & \wedge^+ \arrow[dl] \arrow[dr] & \\ A^+ & & B^+ \end{tikzcd}$ | |
| $\dfrac{\Gamma, \mathcal{X}_A^- \supset \mathcal{X}_{A\vee B}^-, \mathcal{X}_B^- \supset \mathcal{X}_{A\vee B}^- \vdash \Delta_b}{\Gamma, (\mathcal{X}_A^- \vee \mathcal{X}_B^-) \supset \mathcal{X}_{A\vee B}^- \vdash \Delta_b}\ [\supset_3]$ | $\begin{tikzcd}[column sep=small] & \vee^- & \\ A^- \arrow[ur] & & B^- \arrow[ul] \end{tikzcd}$ | |
| $\dfrac{\Gamma, \mathcal{X}_{A\vee B}^+ \supset \mathcal{X}_A^+ \vdash \Delta_b \qquad \Gamma, \mathcal{X}_{A\vee B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_b}{\Gamma, \mathcal{X}_{A\vee B}^+ \supset (\mathcal{X}_A^+ \vee \mathcal{X}_B^+) \vdash \Delta_b}\ [\supset_3']$ | $\begin{tikzcd}[column sep=small] & \vee^+ \arrow[dl] & \\ A^+ & & B^+ \end{tikzcd}$ | $\begin{tikzcd}[column sep=small] & \vee^+ \arrow[dr] & \\ A^+ & & B^+ \end{tikzcd}$ |
| $\dfrac{\begin{array}{l} \Gamma, \mathcal{X}_B^- \supset \mathcal{X}_{A\supset B}^- \vdash \mathcal{X}_A^+ \supset \mathcal{X}_B^-, \Diamond \supset \mathcal{X}_B^-, \Delta_b \\ \qquad \vdots \qquad \Gamma, \Diamond \supset \mathcal{X}_{A\supset B}^- \vdash \Delta_b \end{array}}{\Gamma, (\mathcal{X}_A^+ \supset \mathcal{X}_B^-) \supset \mathcal{X}_{A\supset B}^- \vdash \Delta_b}\ [\supset_4]$ | $\begin{tikzcd}[column sep=small] & \supset^- & \Diamond \\ A^+ & B^- \arrow[l, Rightarrow] \arrow[u] \arrow[ur, Rightarrow] & \end{tikzcd}$ | $\begin{tikzcd}[column sep=small] \supset^- & \Diamond \arrow[l] \\ A^+ & B^- \end{tikzcd}$ |
| $\dfrac{\Gamma, \mathcal{X}_{A\supset B}^+ \supset \mathcal{X}_B^+ \vdash \Delta_b \qquad \Gamma, \mathcal{X}_A^- \supset \mathcal{X}_B^+ \vdash \Delta_b}{\Gamma, \mathcal{X}_{A\supset B}^+ \supset (\mathcal{X}_A^- \supset \mathcal{X}_B^+) \vdash \Delta_b}\ [\supset_4']$ | $\begin{tikzcd}[column sep=small] & \supset^+ \arrow[dr] & \\ A^- & & B^+ \end{tikzcd}$ | $\begin{tikzcd}[column sep=small] & \supset^+ & \\ A^- \arrow[rr] & & B^+ \end{tikzcd}$ |

**Table 1.** Proof rules and bi-colored graph construction

sequents (which are implicational sequents in our setting) and then counter-model construction for these irreducible sequents. This combination of proof-search and counter-model construction can now be viewed in a common graph-theoretic setting that will lead us to efficient parallel counter-model construction. First we present a graph-theoretic approach for proof-search. Then we combine proof-search and counter-model construction into a r-cycle search problem on a conditional bi-colored graph. In section 6, we present a global technique which efficiently solves the conditional r-cycle search problem.

### 5.1 Proof-search as bi-colored graph construction

Let us fix a particular formula $D$. Using the results of section 3.1, $D$ is indexed into an equivalent flat sequent $\delta^-(D) \vdash \Diamond \supset \mathcal{X}_D^-$.[3] Then, let us study the formulae occurring in $\delta^-(D)$.

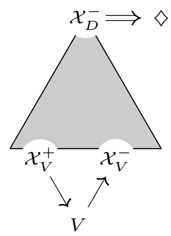
From the definition of $\delta$ (see section 3.1), there is exactly one formula in $\delta^-(D)$ for each occurrence of a subformula of $D$:

[3] For the sake of completeness, we recall the polarity of occurrences of subformulae by suffixing their indexes like in $\mathcal{X}_V^+$ or $\mathcal{X}_V^-$.

– if $V$ is a variable occurring positively (resp. negatively), the formula in $\delta^-(D)$ is $\mathcal{X}^+_V \supset V$ (resp. $V \supset \mathcal{X}^-_V$) which is already an atomic implication which will not be decomposed further during proof search;
– if $A \wedge B$ occurs positively, $\mathcal{X}^+_{A\wedge B} \supset (\mathcal{X}^+_A \wedge \mathcal{X}^+_B)$ will appear in $\delta^-(D)$ and could be decomposed once using rule $[\supset'_2]$;
– if $A \supset B$ occurs negatively, $(\mathcal{X}^+_A \supset \mathcal{X}^-_B) \supset \mathcal{X}^-_{A\supset B}$ will appear in $\delta^-(D)$ and could be decomposed once using rule $[\supset_4]$.

and so on... There is exactly one logical rule for each case: $[\supset_2]$ for $\wedge^-$, $[\supset'_2]$ for $\wedge^+$, $[\supset_3]$ for $\vee^-$, $[\supset'_3]$ for $\vee^+$, $[\supset_4]$ for $\supset^-$, and $[\supset'_4]$ for $\supset^+$. The left column of table 1 presents all these cases.

What are the atomic implications occurring in the sequents generated by a full proof-search from starting $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}^-_D$? Some appear already at the beginning like the formulae $\mathcal{X}^+_V \supset V$ or $V \supset \mathcal{X}^-_V$ for $V$ occurrence of a variable of $D$ and are not changed by proof rules. Also $\diamondsuit \supset \mathcal{X}^-_D$ appears at the beginning and is not changed by proof rules. So in the corresponding bi-colored graph, there are arrows $\mathcal{X}^+_V \rightarrow V$, $V \rightarrow \mathcal{X}^-_V$ and $\mathcal{X}^-_D \Rightarrow \diamondsuit$. The situation is summarized at the right hand side.




The other atomic implications are generated by the decompositions of formulae $\mathcal{X}_{A\otimes B} \supset (\mathcal{X}_A \otimes \mathcal{X}_B)$ and $(\mathcal{X}_A \otimes \mathcal{X}_B) \supset \mathcal{X}_{A\otimes B}$ occurring during backward proof rules application. As an example, we consider the case of a negative occurrence of $A \supset B$. Then $(\mathcal{X}^+_A \supset \mathcal{X}^-_B) \supset \mathcal{X}^-_{A\supset B}$ occurs in $\delta^-(D)$ and the corresponding rule will be

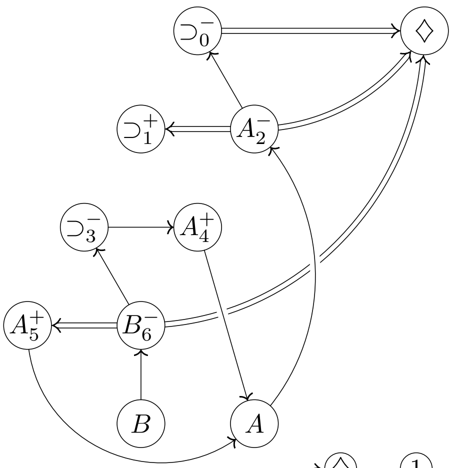
$$\frac{\ldots, \mathcal{X}^-_B \supset \mathcal{X}^-_{A\supset B} \vdash \mathcal{X}^+_A \supset \mathcal{X}^-_B, \diamondsuit \supset \mathcal{X}^-_B, \ldots \qquad \ldots, \diamondsuit \supset \mathcal{X}^-_{A\supset B} \vdash \ldots}{\ldots, (\mathcal{X}^+_A \supset \mathcal{X}^-_B) \supset \mathcal{X}^-_{A\supset B} \vdash \ldots}\ [\supset_4]$$

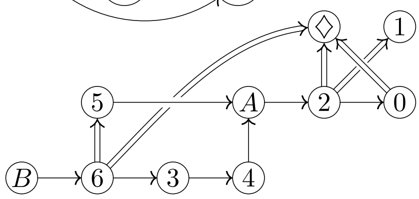
Let us consider a completed proof-search branch ending with an implicational sequent. Its associated graph either contains $\mathcal{X}^-_B \rightarrow \mathcal{X}^-_{A\supset B}$, $\mathcal{X}^-_B \Rightarrow \mathcal{X}^+_A$ and $\mathcal{X}^-_B \Rightarrow \diamondsuit$ if we choose the left premise of rule $[\supset_4]$ or $\diamondsuit \rightarrow \mathcal{X}^-_{A\supset B}$ if we choose the right premise. This could be summarized by the two following arrow introduction rules:

$$\begin{tikzcd} \supset^- & & \diamondsuit \\ A^+ & B^- \arrow[l, Rightarrow] \arrow[ul] \arrow[ur, Rightarrow] & \end{tikzcd} \qquad \text{(exclusive) or} \qquad \begin{tikzcd} \supset^- & \diamondsuit \arrow[l] \\ A^+ & B^- \end{tikzcd}$$

The complete set of arrow introduction rules is given in the right column of table 1. 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 implied in proof-search are all strongly invertible and so the order of application of those rules does not influence the validity of the generated sequents: all the rules are permutable. Let us illustrate these ideas on a concrete example.

### 5.2 An example: the Pierce's formula




$D = ((A \supset B) \supset A) \supset A$ is the Pierce's formula. We index it:

$$\big((A_5^+ \supset_3^- B_6^-) \supset_1^+ A_4^+\big) \supset_0^- A_2^-$$

The decomposition tree of this formula has three internal nodes $\supset_0^-$, $\supset_1^+$ and $\supset_3^-$. The leaves of the decomposition tree introduce the arrows $A_5^+ \to A$, $A_4^+ \to A$, $A \to A_2^-$ and $B \to B_6^-$ in the bi-colored graph. And of course, there is also the arrow $\supset_0^- \Rightarrow \diamondsuit$ corresponding to $\ldots \vdash \diamondsuit \supset \mathcal{X}_D^-$.

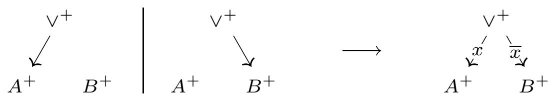
Then we choose one proof-search branch: left premise for $\supset_0^-$ and $\supset_3^-$ and right premise for $\supset_1^+$. We obtain the bi-colored graph presented on the rhs. We explain the remaining arrows: the left premise for $\supset_0^-$ (rule $[\supset_4]$) introduces $A_2^- \to \supset_0^-$, $A_2^- \Rightarrow \supset_1^+$ and $A_2^- \Rightarrow \diamondsuit$; the right premise for $\supset_1^+$ (rule $[\supset_4']$) introduces $\supset_3^- \to A_4^+$; the left premise for $\supset_3^-$ (rule $[\supset_4]$) introduces $B_6^- \to \supset_3^-$, $B_6^- \Rightarrow A_5^+$ and $B_6^- \Rightarrow \diamondsuit$.

This graph is the bi-colored graph associated with the implicational end-sequent of the proof-search branch characterized by the choice: left for $\supset_0^-$ and $\supset_3^-$ and right for $\supset_1^+$. To decide if this sequent is valid in $\mathsf{LC}$, we look for r-cycles. For this, we redesign the graph so that $\to$ arrows are horizontal or go up and $\Rightarrow$ go up strictly. It appears that this graph has a bi-height $h$ defined by $h(x) = 0$ for $x \in \{B, 6, 3, 4\}$, $h(x) = 1$ for $x \in \{5, A, 2, 0\}$ and $h(x) = 2$ for $x \in \{\diamondsuit, 1\}$. Then $[\![\cdot]\!]$ defined by $[\![A]\!] = 1$ and $[\![B]\!] = 0$ is a counter-model of the Pierce's formula since then $[\![((A \supset B) \supset A) \supset A]\!] = ((1 \to 0) \to 1 \to 1 = (0 \to 1) \to 1 = \infty \to 1 = 1 < \infty$. $[\![\cdot]\!]$ is a counter-model of Pierce's formula in $\mathsf{LC}$ but also in $\mathsf{LC}_2$ (just under $\mathsf{CL} = \mathsf{LC}_1$.)

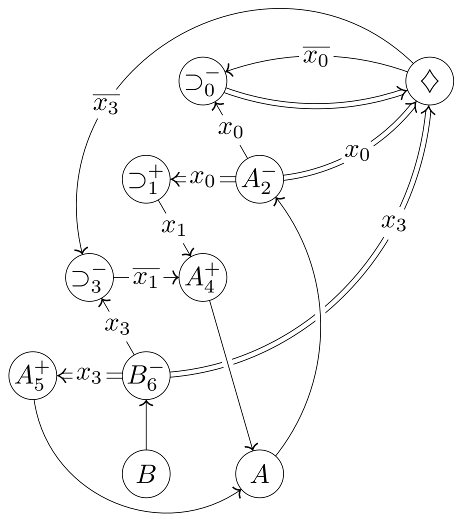
### 5.3 Postponing proof-search branch selection

Our system has a very important property that the others lack: proof-search can be seen has the incremental construction of a semantic graph. The nodes of this graph do not depend on the proof-search branch; only the choices of arrows depend on the branch chosen. In section 5.2, we have chosen a "good" proof-search branch from which we can extract a counter-model. There are many other branches that lead to other bi-colored graphs which might also lack r-cycles, leading to other counter-models. Clearly, a proof-search branch lacking r-cycles corresponds to a counter-model. Is it possible to find all the counter-models and thus, all such branches? A idea for that is to postpone the choice of premises characterizing proof-search branches, to detect r-cycles and after that, to select the branches for which no r-cycle exists.

Instead of using either left or right premise, let us use both. Of course this would not be sound unless we keep track of the fact that these two choices cannot coexist in the same branch. This is done by introducing boolean selectors ($x$ and its negation $\overline{x}$) and indexing arrows with those selectors. For example, we obtain the following transformation of rules for a positive occurrence of disjunction:




Let us reconsider the case of the Pierce's formula. We apply all the possible rules, postponing all the choices between left or right premises and introducing a new boolean selector for each potential choice. Each three rules have two premises and we introduce the selectors $x_0$ for $\supset_0^-$, $x_1$ for $\supset_1^+$ and $x_3$ for $\supset_3^-$. The corresponding indexed bi-colored graph is represented at the right. The reader might notice that the graph of section 5.2 is just an instance of this graph with $x_0 = 1$, $x_1 = 0$ and $x_3 = 1$.




| | |
|---|---|
| $0 \Rightarrow \diamondsuit \rightarrow 0$ | $x_0$ |
| $0 \Rightarrow \diamondsuit \rightarrow 3 \rightarrow 4 \rightarrow A \rightarrow 2 \rightarrow 0$ | $x_3 + x_1 + \overline{x_0}$ |
| $2 \Rightarrow \diamondsuit \rightarrow 3 \rightarrow 4 \rightarrow A \rightarrow 2$ | $\overline{x_0} + x_3 + x_1$ |
| $2 \Rightarrow 1 \rightarrow 4 \rightarrow A \rightarrow 2$ | $\overline{x_0} + \overline{x_1}$ |

We want to discover all the possible valuations of selectors such that the corresponding instance graph has no r-cycle. For example, $0 \Rightarrow \diamondsuit \rightarrow 0$ is a r-cycle. To "break" this cycle, the constraint $\overline{x_0} = 0$ has to be satisfied, i.e. $x_0$ has to be satisfied. Let us do this for all the basic r-cycles, i.e. that r-cycles that do not repeat nodes. It is sufficient to "break" basic r-cycles for all the r-cycles to be removed because any r-cycle contains at least one basic r-cycle.

Let us look for basic r-cycles: no r-cycle passes through $B$ so we can remove it from the graph. Then no r-cycle passes through $B_6^-$. Then no r-cycle passes through $A_5^+$. Then, we obtain four basic r-cycles and their associated constraints expressing the condition at which the corresponding r-cycle is broken: $x_0$, $x_3 + x_1 + \overline{x_0}$, $\overline{x_0} + x_3 + x_1$ and $\overline{x_0} + \overline{x_1}$.

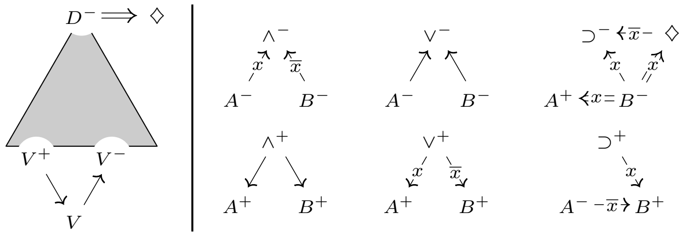



**Fig. 2.** Counter-model search system for $\mathsf{LC}_n$

Then we solve these constraints altogether, i.e. we search a valuation satisfying the conjunction of all these constraints. There is only one solution which is $x_0 = 1$, $x_1 = 0$ and $x_3 = 1$, corresponding to the branch characterized by left for $\supset_0^-$ and $\supset_3^-$ and right for $\supset_1^+$. This is the proof-search branch we have chosen in section 5.2. With any other valuation, one of the four basic r-cycles would exists, and so, it is the only one for which the instanced bi-colored graph has no r-cycle.

### 5.4 A parallel counter-model search system for $\mathsf{LC}_n$

We introduce our final counter-model search system which is a combination of the graphical system of table 1 and the idea of postponing choice of left or right premise with the help of boolean selectors. The word parallel means that all the search branches are explored simultaneously with the help of selectors.

The system is presented in figure 2. The principle is to build a *conditional bi-colored graph*, i.e. a bi-colored graph where arrows might be annotated by boolean selectors. We fix a formula $D$. We start from the nodes of the decomposition tree of $D$ composed of internal nodes ($\mathcal{X}^+_{A\otimes B}$ or $\mathcal{X}^-_{A\otimes B}$) and leaves ($\mathcal{X}^+_V$ or $\mathcal{X}^-_V$). We add one new node for each variable $V$ occurring in $D$.[4] We also add one node for $\diamondsuit$. 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 $V$. This is summarized by the left side of figure 2. All these arrows are not annotated: they are present in every proof-search branch, i.e. for any valuation on selectors.

Then for each internal node, we add arrows according to the schemes of the right part of figure 2. There is one scheme for each case in $\{\wedge, \vee, \supset\} \times \{+, -\}$. For the schemes $\wedge^-$, $\vee^+$, $\supset^+$ and $\supset^-$, we introduce conditional arrows, i.e arrows indexed with a boolean selector of the form $x$ or $\overline{x}$. This selector has to be new: there is exactly one selector for each instance of a scheme.[5] 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 $D$.

[4] For a variable $V$ occurring in $D$, there are some nodes $\mathcal{X}^+_V$ (resp. $\mathcal{X}^-_V$) where $V$ is a positive (resp. negative) occurrence in $D$, and one node for the variable $V$.

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

**Theorem 8.** *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* $\mathsf{LC}$ *if and only if there exists a valuation $v$ on the selectors such that the instance graph $\mathcal{G}_v$ has no r-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 irreducible (and implicational) end-sequent of a completed proof-search branch. This end-sequent is characterized by a choice of left or right premises and it corresponds to a choice of a valuation on selectors.

By theorem 7, the end-sequent of a proof-search branch has a counter-model if and only if its associated bi-colored graph has no r-cycle. But this graph is exactly the instance $\mathcal{G}_v$ where $v$ is the valuation on selectors corresponding to the proof-search branch. $\square$

**Theorem 9.** *$D$ has a counter-model in $\mathsf{LC}_n$ iff there exists a valuation $v$ on the selectors such that the instance graph $\mathcal{G}_v$ has no chain of the form $(\rightarrow^\star\Rightarrow)^{n+1}$.*

**Proof.** For $\mathsf{LC}_n$ with $n \neq \infty$, we just have to apply theorem 6 instead of theorem 7. The proof rules are common for all the family $\mathsf{LC}_n$ (including $n = \infty$.) $\square$

Computing the conditional bi-colored graph from a given formula takes a linear time but finding all the r-cycles in a graph and solving the boolean constraints system have both exponential complexity. Indeed, there exists formulae 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.

An approach to solve this problem would be to try to mix r-cycle detection and boolean constraint solving to obtain a kind of conditional r-cycle detection but this is not an easy task. After reflexion, we can observe that we do not need the list of all the r-cycles to decide a formula: we only need to compute and solve the boolean condition characterizing the existence of r-cycles.

## 6 Practical r-cycle detection in conditional graphs

In section 5.4, we have introduced the notion of conditional bi-colored graph which is a graph in which arrows might be indexed with boolean constraints. To implement conditional graphs, we have chosen to 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 matrix cells might not only contain 0 or 1 (for unconditional arrows) but also arbitrary boolean expressions.

**Definition 6 (Conditional matrix).** *A* conditional matrix *on set $\mathcal{S}$ of size $k$ is an $k \times k$-array with values in the free boolean algebra (of boolean expressions.)*

We point out that boolean expressions are considered up to logical equivalence. So a conditional bi-colored graph is viewed as a pair $(\rightarrow, \Rightarrow)$ of conditional matrices. The algebraic operations we have defined like $\rightarrow + \Rightarrow$, $\rightarrow \cdot \Rightarrow$, $\rightarrow^\star$, $\mathrm{tr}(\cdot)$ and $\sum(\cdot)$ extend naturally 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 call an *instance graph* and denote by $\mathcal{G}_v$ the bi-colored graph obtained by instantiating boolean expressions in the cells of the matrices $\rightarrow$ and $\Rightarrow$ with the valuation $v$. Instantiation commutes with algebraic operations on matrices because it commutes with the boolean operators $\cdot$ and $+$. For example $[\rightarrow + \Rightarrow]_v = \rightarrow_v + \Rightarrow_v$.

**Theorem 10.** *Let $\mathcal{G} = (\rightarrow, \Rightarrow)$ be a conditional bi-colored graph. There exists a r-cycle in every instance $\mathcal{G}_v$ of $\mathcal{G}$ if and only if $tr\big((\rightarrow + \Rightarrow)^\star \Rightarrow\big) = 1$ holds.*

The proof is straightforward: $\big[\mathrm{tr}\big((\rightarrow + \Rightarrow)^\star \Rightarrow\big)\big]_v = \mathrm{tr}\big((\rightarrow_v + \Rightarrow_v)^\star \Rightarrow_v\big)$ and then theorem 8. The reader is reminded that the equation $\mathrm{tr}\big((\rightarrow + \Rightarrow)^\star \Rightarrow\big) = 1$ is an equivalence of boolean expressions. The result for the finitary version $\mathsf{LC}_n$ also holds. To compute the smallest $n$ such that a formula is not valid, we compute sequentially until $\sum(\rightarrow^\star \Rightarrow)^{n+1} < 1$ and then, find a valuation $v$ such that the instance is refuted: $\sum(\rightarrow_v^\star \Rightarrow_v)^{n+1} = 0$.

We have implemented a prototype of counter-model search engine for $\mathsf{LC}$ based on these principles in Objective Caml.[6] We give a brief description of the practical choices we have made. A $k \times k$ conditional matrix is represented by a *sparse matrix* of integers. These integers represent nodes of a *shared RO-BDD* [5] so they uniquely encode boolean expressions up to equivalence and algebraic operations may be performed efficiently. How can we compute traces efficiently? Let $\alpha$ be a $k \times k$ conditional matrix. Let $I$ be the identity $k \times k$ matrix ($I_{x,x} = 1$ and $I_{x,y} = 0$ otherwise.) Then one could check that $\alpha^\star = (I + \alpha)^k$ (this is because any conditional path of size greater than $k+1$ necessarily has repeating points and thus contains a sub-path of size smaller than $k$.) It is even sufficient to compute the sequence $I, I + \alpha, (I + \alpha)^2, \ldots$ until it stabilizes. This happens in at most $k$ steps. So let $A = \rightarrow + \Rightarrow$ and $B = \Rightarrow$. We want to compute $\mathrm{tr}(A^\star B)$. We compute the trace column by column. Let $B_i$ be the $i$-th column of $B$ and $T$ be the column with 1 on each cell. Then we compute the sequence $t_0, \ldots, t_k$: $t_0 = 0$ and $t_i = \big[A^\star(t_{i-1}T + B_i)\big]_i$ for $i = 1, \ldots, k$ by induction and we obtain $t_k = \mathrm{tr}(A^\star B)$. The sequence $t_0, \ldots, t_k$ is increasing and the computation can be stopped as soon as $t_i = 1$ in which case, the formula is provable. Previously generated constraints are used to accelerate the computation by adding $t_{i-1}$ to each cell of $B_i$ (in $t_{i-1}T + B_i$).

## 7 Conclusion

Compared to the work initiated by Avron [2,3] that focuses mainly on proof-search and even logic programming [13], we develop the idea of mixing proof-search and counter-model construction. Then, we propose a new system for

[6] It will be available shortly at `http://www.loria.fr/~larchey/LC`.

counter-model search in $\mathsf{LC}$ and $\mathsf{LC}_n$, mainly based on the notion of r-cycles in conditional graphs, and thus an efficient algorithm to decide these logics and provide counter-models. In further work, we will deeper investigate the relationships between the notion of r-cycle and the G-cycles of [3] and analyze if our conditional graphs also fit in the hyper-sequent setting. We will also investigate the relationships between our parallel counter-model search and other approaches based for example on parallel dialogue games [4,9].

## References


1. Alessandro Avellone, Mauro Ferrari, and Pierangelo Miglioli. 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, 1999.
2. Arnon Avron. A Tableau System for Gödel-Dummett Logic Based on a Hypersequent Calculus. In *TABLEAUX 2000*, volume 1847 of *LNAI*, pages 98–111, 2000.
3. Arnon Avron and Beata Konikowska. Decomposition Proof Systems for Gödel-Dummett Logics. *Studia Logica*, 69(2):197–219, 2001.
4. Matthias Baaz and Christian Fermüller. Analytic Calculi for Projective Logics. In *TABLEAUX'99*, volume 1617 of *LNCS*, pages 36–50, 1999.
5. Randal E. Bryant. Graph-based algorithms for Boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 1986.
6. Michael Dummett. A Propositional Calculus with a Denumerable matrix. *Journal of Symbolic Logic*, 24:96–107, 1959.
7. Roy Dyckhoff. Contraction-free Sequent Calculi for Intuitionistic Logic. *Journal of Symbolic Logic*, 57(3):795–807, 1992.
8. Roy Dyckhoff. A Deterministic Terminating Sequent Calculus for Gödel-Dummett logic. *Logical Journal of the IGPL*, 7:319–326, 1999.
9. Christian Fermüller. Parallel Dialogue Games and Hypersequents for Intermediate Logics. In *TABLEAUX 2003*, volume 2796 of *LNAI*, pages 48–64, 2003.
10. Kurt Gödel. Zum intuitionistischen Aussagenkalkül. In *Anzeiger Akademie des Wissenschaften Wien*, volume 69, pages 65–66. 1932.
11. Petr Hajek. *Metamathematics of Fuzzy Logic*. Kluwer Academic Publishers, 1998.
12. Dominique Larchey-Wendling. Combining Proof-Search and Counter-Model Construction for Deciding Gödel-Dummett Logic. In *CADE-18*, volume 2392 of *LNAI*, pages 94–110, 2002.
13. George Metcalfe, Nicolas Olivetti, and Dov Gabbay. Goal-Directed Calculi for Gödel-Dummett Logics. In *CSL*, volume 2803 of *LNCS*, pages 413–426, 2003.


## A Proofs of theorems of section 3

**Theorem 1.** *In $\mathsf{LC}_n$, $D$ is valid if and only if $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}_D$ is valid.*

**Proof.** First of all, it is clear that the sequent $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}_D$ is flat. Then we prove that if $\delta^-(D) \vdash \diamondsuit \supset \mathcal{X}_D$ is valid then so is $D$.

Let $[\![\cdot]\!] : \mathsf{Var} \to [0, n[\cup\{\infty\}$ be an interpretation. As the $\mathcal{X}_K$ are new variables not occurring in $D$, we can define $[\![\cdot]\!]' : \mathsf{Var} \cup \{\diamondsuit\} \to [0, n[\cup\{\infty\}$ by $[\![V]\!]' = [\![V]\!]$ for $V$ occurring in $D$, $[\![\mathcal{X}_K]\!]' = [\![K]\!]$ for $K$ occurrence of a subformula of $D$ and

$[\![\Diamond]\!]' = \infty$. Obviously, $[\![K]\!]' = [\![K]\!]$ holds for any subformula $K$ of $D$ (because $[\![\cdot]\!]'$ and $[\![\cdot]\!]$ match on the variable occurring in $D$, and thus in $K$.) Then, as $\delta^-(D) \vdash \Diamond \supset \mathcal{X}_D$ is valid in $\mathsf{LC}_n$, we deduce $\lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor' \leqslant [\![\Diamond \supset \mathcal{X}_D]\!]'$. By proposition 2, $\lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor' = \infty$ holds. Also $[\![\Diamond \supset \mathcal{X}_D]\!]' = [\![\mathcal{X}_D]\!]' = [\![D]\!]$ holds. We deduce the inequality $\infty \leqslant [\![D]\!]$. As this last inequality holds in any interpretation $[\![\cdot]\!]$, $D$ is valid in $\mathsf{LC}_n$.

Then we prove that any counter-model of $\delta^-(D) \vdash \Diamond \supset \mathcal{X}_D$ is also a counter-model of $D$. Let $[\![\cdot]\!]$ be a counter-model of $\delta^-(D) \vdash \Diamond \supset \mathcal{X}_D$ in $\mathsf{LC}_n$. Then $[\![\Diamond \supset \mathcal{X}_D]\!] < \lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor$ holds and consequently $[\![\Diamond \supset \mathcal{X}_D]\!] < \infty$. So $[\![\Diamond]\!] > [\![\mathcal{X}_D]\!]$ and $[\![\mathcal{X}_D]\!] = [\![\Diamond \supset \mathcal{X}_D]\!] < \lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor$ holds. By proposition 1, $\lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor \wedge [\![D]\!] \leqslant [\![\mathcal{X}_D]\!]$ holds. So if we suppose $[\![D]\!] = \infty$ then $\lfloor\!\lfloor \delta^-(D) \rfloor\!\rfloor \leqslant [\![\mathcal{X}_D]\!]$ holds which is absurd. Thus, the inequality $[\![D]\!] < \infty$ holds and $[\![\cdot]\!]$ is a counter-model of $D$. □

**Proposition 5.** *Let $\Delta_b$ be a b-context and $[\![\cdot]\!]$ an interpretation of variables in $[0, n[ \cup \{\infty\}$ such that $\lceil\!\lceil \Delta_b \rceil\!\rceil < \infty$ holds. Then $\lceil\!\lceil \Delta_b \rceil\!\rceil < [\![\Diamond]\!]$ holds too.*

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

**Theorem 2.** *The following rule is sound and strongly invertible in $\mathsf{LC}_n$:*

$$\frac{\Gamma, B \supset C \vdash A \supset B, \Diamond \supset B, \Delta_b \qquad \Gamma, \Diamond \supset C \vdash \Delta_b}{\Gamma, (A \supset B) \supset C \vdash \Delta_b} [\supset_4] \quad \Delta_b \textit{ is a b-context}$$

**Proof.** First of all, with the side condition, it is clear that this rule preserves b-contexts (on the right hand side of the $\vdash$ sign) bottom-up. We first prove soundness. So suppose that both premises sequents are valid in $\mathsf{LC}_n$. Let $[\![\cdot]\!] : \mathsf{Var} \cup \{\Diamond\} \to [0, n[ \cup \{\infty\}$ be an interpretation of variables. We have to prove

$$\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![(A \supset B) \supset C]\!] \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil \tag{1}$$

Let us consider two trivial cases:

- if $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$, it is clear that property (1) holds.
- if $[\![\Diamond]\!] \leqslant [\![A \supset B]\!]$ then $[\![((A \supset B) \supset C]\!] \leqslant [\![\Diamond \supset C]\!]$ and thus $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![(A \supset B) \supset C]\!] \leqslant \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![\Diamond \supset C]\!] \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$ holds because the right premise of rule $[\supset_4]$ is valid. So property (1) holds.

So we are left in the case where $\lceil\!\lceil \Delta_b \rceil\!\rceil < \lfloor\!\lfloor \Gamma \rfloor\!\rfloor$ and $[\![A \supset B]\!] < [\![\Diamond]\!]$ both hold. We can deduce $\lceil\!\lceil \Delta_b \rceil\!\rceil < \infty$ and thus $\lceil\!\lceil \Delta_b \rceil\!\rceil < [\![\Diamond]\!]$ by proposition 5. Since the right premise is valid, we obtain $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![\Diamond \supset C]\!] \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$ from which we deduce $[\![\Diamond \supset C]\!] \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil < \infty$. So $[\![C]\!] = [\![\Diamond \supset C]\!] \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$.

From $[\![A \supset B]\!] < [\![\Diamond]\!]$ we deduce $[\![A \supset B]\!] = [\![\Diamond \supset B]\!] = [\![B]\!] < [\![\Diamond]\!]$. Since the first premise is valid, we obtain $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![B \supset C]\!] \leqslant [\![B]\!] \vee \lceil\!\lceil \Delta_b \rceil\!\rceil$. Now suppose that $[\![B]\!] \leqslant [\![C]\!]$ then $[\![B \supset C]\!] = \infty$ and $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor = \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge [\![B \supset C]\!] \leqslant [\![B]\!] \vee \lceil\!\lceil \Delta_b \rceil\!\rceil \leqslant$

$\llbracket C \rrbracket \vee \lceil\!\lceil \Delta_b \rceil\!\rceil \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$ (because $\llbracket C \rrbracket \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$ holds.) This is absurd. So the property $\llbracket C \rrbracket < \llbracket B \rrbracket$ necessarily holds. Then $\llbracket (A \supset B) \supset C \rrbracket = \llbracket B \supset C \rrbracket = \llbracket C \rrbracket = \llbracket \diamondsuit \supset C \rrbracket$. We conclude by $\lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge \llbracket (A \supset B) \supset C \rrbracket \leqslant \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge \llbracket \diamondsuit \supset C \rrbracket \leqslant \lceil\!\lceil \Delta_b \rceil\!\rceil$ since the right premise of rule $[\supset_4]$ is valid. Hence, property (1) holds.

Now we prove that rule $[\supset_4]$ is strongly invertible. Let $\llbracket \cdot \rrbracket$ be a counter model of the left premise. It satisfies the inequality

$$\llbracket A \supset B \rrbracket \vee \llbracket \diamondsuit \supset B \rrbracket \vee \lceil\!\lceil \Delta_b \rceil\!\rceil < \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge \llbracket B \supset C \rrbracket$$

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 $\lceil\!\lceil \Delta_b \rceil\!\rceil < \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge \llbracket B \supset C \rrbracket = \lfloor\!\lfloor \Gamma, (A \supset B) \supset C \rfloor\!\rfloor$ so $\llbracket \cdot \rrbracket$ is a counter-model of the conclusion of rule $[\supset_4]$. Finally let $\llbracket \cdot \rrbracket$ be a counter-model of the right premise. If satisfies the inequality

$$\lceil\!\lceil \Delta_b \rceil\!\rceil < \lfloor\!\lfloor \Gamma \rfloor\!\rfloor \wedge \llbracket \diamondsuit \supset C \rrbracket$$

so $\lceil\!\lceil \Delta_b \rceil\!\rceil < \infty$ and thus $\lceil\!\lceil \Delta_b \rceil\!\rceil < \llbracket \diamondsuit \rrbracket$ by proposition 5. Also $\lceil\!\lceil \Delta_b \rceil\!\rceil < \llbracket \diamondsuit \supset C \rrbracket$. So necessarily, $\lceil\!\lceil \Delta_b \rceil\!\rceil < \llbracket C \rrbracket$. Finally $\lceil\!\lceil \Delta_b \rceil\!\rceil < \llbracket C \rrbracket \leqslant \llbracket (A \supset B) \supset C \rrbracket \leqslant \lfloor\!\lfloor \Gamma, (A \supset B) \supset C \rfloor\!\rfloor$ and so $\llbracket \cdot \rrbracket$ is a counter-model of the conclusion of rule $[\supset_4]$. $\square$