A proof-theoretic study of bi-intuitionistic propositional sequent calculus

LUÍS PINTO, Centro de Matemática, Universidade do Minho, Campus de Gualtar; 4710-057 Braga, Portugal.
E-mail: luis@math.uminho.pt

TARMO UUSTALU, Institute of Cybernetics at Tallinn University of Technology, Akadeemia tee 21, 12618 Tallinn, Estonia.
E-mail: tarmo@cs.ioc.ee

Abstract

Bi-intuitionistic logic is the conservative extension of intuitionistic logic with a connective dual to implication usually called ‘exclusion’. A standard-style sequent calculus for this logic is easily obtained by extending multiple-conclusion sequent calculus for intuitionistic logic with exclusion rules dual to the implication rules (in particular, the exclusion-left rule restricts the premise to be single-assumption). However, similarly to standard-style sequent calculi for non-classical logics like $S5$, this calculus is incomplete without the cut rule. Motivated by the problem of proof search for propositional bi-intuitionistic logic (BiInt), various cut-free calculi with extended sequents have been proposed, including (i) a calculus of nested sequents by Goré et al., which includes rules for creation and removal of nests (called ‘nest rules’, resp. ‘unnest rules’) and (ii) a calculus of labelled sequents by the authors, derived from the Kripke semantics of BiInt, which includes ‘monotonicity rules’ to propagate truth/falsehood between accessible worlds.

In this paper, we develop a proof-theoretic study of these three sequent calculi for BiInt grounded on translations between them. We start by establishing the basic meta-theory of the labelled calculus (including cut-admissibility), and use then the translations to obtain results for the other two calculi. The translation of the nested calculus into the standard-style calculus explains how the unnest rules encapsulate cuts. The translations between the labelled and the nested calculi reveal the two formats to be very close, despite the former incorporating semantic elements, and the latter being syntax-driven. Indeed, we single out (i) a labelled calculus whose sequents have a ‘label in focus’ and which includes ‘refocusing rules’ and (ii) a nested calculus with monotonicity and refocusing rules, and prove these two calculi to be isomorphic (in a bijection both at the level of sequents and at the level of derivations).

1 Introduction

Bi-intuitionistic logic (also known as Heyting–Brouwer logic or as subtractive logic) is the conservative extension of intuitionistic logic with a connective dual to implication called exclusion (also known as coimplication or subtraction). Bi-intuitionistic logic can also be seen as the union of intuitionistic logic (lacking exclusion) with dual-intuitionistic logic (lacking implication) [14], hence the word ‘bi’-intuitionistic. Whereas intuitionistic logic has the disjunction property (if $A \lor B$ is provable, either $A$ is provable or $B$ is provable), dual-intuitionistic logic has the dual conjunction property (if $A \land B$ is refutable, either $A$ is refutable or $B$ is refutable). In the union, both of these properties are lost, yet one cannot prove excluded middle (nor refute contradiction for the dual-intuitionistic weak negation).

Bi-intuitionistic logic first got the attention of Rauszer [32,31,33], who studied its algebraic and Kripke semantics, alongside with a Hilbert-style system and a sequent calculus. More recently, bi-intuitionistic logic was revisited with quite some different motivations. For example, in line with Curien–Herbelin’s study [8] of dualities in classical sequent calculus, Crolard [7] gave an interpretation of bi-intuitionistic exclusion as a type for coroutines (a restricted form of continuations).
With philosophical motivations, Bellin and colleagues [1, 2] considered a different symmetrization of intuitionistic logic (also called bi-intuitionistic) from the point of view of pragmatics, a logic of assertions and hypotheses, and in the same vein, Wansing [37] combined verificationism and falsificationism.

Another line of studies on bi-intuitionistic logic was motivated by automated theorem proving and proof-theoretic considerations. Although a sequent calculus characterization of the logic is easily obtained by extending the multiple-conclusion sequent calculus for intuitionistic logic à la Maehara–Dragalin [23, 9] with exclusion rules dual to the implication rules, such a calculus does not enjoy cut elimination. Neither is cut eliminable in the sequent calculus of Rauszer [31] (the proof in the paper is incorrect). The lack of cut elimination in a sequent calculus is problematic for proof search, since the subformula property is not guaranteed. In order to overcome this issue, calculi with a non-standard notion of sequent have been considered [15, 4, 16, 17, 27, 29]. In particular, a calculus of nested sequents was proposed by Goré, Postniece and Tiu [17] and a calculus with labelled formulas was proposed by the authors [27].

In the presence of various proof systems for a logic, a natural question to ask is how they relate to each other and what can be learned from the relationship. In this paper, we present a proof-theoretic study of sequent calculus for bi-intuitionistic propositional logic, which extends the study initiated in [28] on translations between the standard-style (Maehara–Dragalin-style), nested and labelled sequent calculi for this logic. In particular, the translations make it possible to read standard-style derivations off derivations resulting from proof search in the nested or labelled calculi. More interestingly, the detailed study of the translations offered in this paper provides means for transferring meta-theory between the various systems and enables, in particular, identification of unnest cuts as a complete class of cuts for the formulation with standard sequents. Additionally, this refined study allows to justify precisely the claim that nested and labelled sequent calculi can be viewed as ‘notational variants’ (found in [12, 18] in the context of classical modal logics), through the design of new labelled and nested systems for propositional bi-intuitionistic logic which are isomorphic (exhibiting a 1–1 correspondence both for sequents and for derivations).

In comparison to the workshop paper [28], the novel material in this extended version includes the following.

1. All the material in Subsection 3.2 is new. The syntactic proofs for the labelled system of invertibility of the inference rules and of admissibility of weakening, contraction, cut, nodemerge and nodesplit are new (in [27] and in [28] these results are either not claimed or are obtained by semantical arguments). These results make the paper self-contained, in particular, in what concerns the identification of complete classes of cuts for the standard-style system.

2. Also new are the results on bijectivity of the translations between labelled and nested sequents in Subsection 5.1 and all the material in Subsection 5.3, where we introduce and prove isomorphic: (i) a labelled system whose sequents have a label in focus and which includes refocusing rules and (ii) a nested system with monotonicity and refocusing rules. The relationship in Subsection 5.4 to the deep inference system $DBiInt$ for $BiInt$ by Postniece [29, 30] is also new.

---

1 Indeed, the idea of relating formal systems and transferring properties between them goes back to the very birth of sequent calculus in the work of Gentzen [13], where consistency of classical natural deduction was obtained via sequent calculus.
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

Finally, we have added the various results on admissibility/eliminability of contraction in Section 6 as well as the identification of a subclass of unnest cuts (‘absorbed’ into the implication-right rule and the exclusion-left rule, see Figure 8) still complete for the standard-style system.

We end this section with the organization of the rest of the paper, the naming scheme for the main proof systems covered in detail in the paper and a diagram (Figure 1) with the embeddings studied in the paper. Section 2 introduces the syntax and Kripke semantics for BiInt, the standard-style sequent calculus LBiI and the nested sequent calculus N-LBiI. Section 3 introduces the labelled sequent calculus L-LBiI and develops its meta-theory. Section 4 relates the standard-style system with the other two systems. Section 5 relates the base systems of nested and labelled sequents, introduces and proves isomorphic their variants, respectively, N0-LBiI and L0-LBiI, and relates the labelled systems with deep inference on nested sequents. Section 6 shows some applications of the embeddings, and Section 7 concludes the paper. The names of the systems introduced in this paper all have the base element LBiI (the initial L to indicate they are in sequent calculus format, as in Gentzen’s nomenclature LK and LJ). The prefixes N- and N0- indicate systems of nested sequents. The prefixes L- and L0- indicate systems of labelled sequents.

2 Bi-intuitionistic propositional logic

2.1 Syntax and semantics

We start by defining the syntax and semantics of bi-intuitionistic propositional logic (BiInt).

The language of BiInt extends that of intuitionistic propositional logic (Int), by one connective, exclusion, thus the formulas are given by the grammar:

\[ A, B := p \mid \top \mid \bot \mid A \land B \mid A \lor B \mid A \supset B \mid A \prec B \]

where \( p \) ranges over a denumerable set of propositional variables which give us atoms. The formula \( A \prec B \) is the exclusion of \( B \) from \( A \). We do not take negations as primitive. However, in addition to the intuitionistic (or strong) negation, there is also a dual-intuitionistic (or weak) negation. The two negations are definable by \( \neg A := A \supset \bot \) and \( \prec A := \top \prec A \).

The semantics of BiInt is usually given à la Kripke, although one can also proceed from an algebraic semantics (in terms of Heyting–Brouwer algebras) and there are further alternatives. The Kripke semantics is about truth relative to worlds in Kripke structures that are the same as for Int. A Kripke structure is a triple \( K = (W, \leq, I) \) where \( W \) is a non-empty set whose elements we think of as worlds, \( \leq \) is a preorder (reflexive-transitive binary relation) on \( W \) (the accessibility relation).
and \( I \)—the interpretation—is an assignment of sets of propositional variables to the worlds, which is monotone w.r.t. \( \leq \), i.e., whenever \( w \leq w' \), we have \( I(w) \subseteq I(w') \).

Truth in Kripke structures is defined as for \( \text{Int} \), but covers also exclusion, interpreted dually to implication as possibility in the past:

- \( w \vdash p \iff p \in I(w) \);
- \( w \vdash \top \) always; \( w \vdash \bot \) never;
- \( w \vdash A \land B \iff w \vdash A \text{ and } w \vdash B \); \( w \vdash A \lor B \iff w \vdash A \text{ or } w \vdash B \);
- \( w \models A \supset B \iff \) for any \( w' \geq w \), \( w' \not\vdash A \text{ or } w' \vdash B \);
- \( w \models A \between B \iff \) for some \( w' \leq w \), \( w' \vdash A \text{ and } w' \not\vdash B \).

A formula is called valid if it is true in all worlds of all structures. It is easy to see that monotonicity extends from atoms to all formulas thanks to the universal and existential semantics of implication and exclusion.

It is important for this paper that, instead of general Kripke structures, one may equivalently work with Kripke trees. These are Kripke structures \((W, \leq, I)\) where \( W \) is finite and the preorder \( \leq \) arises as the reflexive-transitive closure of some asymmetric binary relation \( \rightarrow \) on \( W \), subject to the condition that any two worlds \( w, w' \) are related by the reflexive-transitive-symmetric closure of \( \rightarrow \) in a unique way (\( w' \) is reached from \( w \) by exactly one path along \( \rightarrow \cup \leftarrow \)).

It is also a basic observation that Gödel’s translation of \( \text{Int} \) into the modal logic \( \text{S4} \) extends to a translation into the future-past tense logic \( \text{KtT4} \) (cf. [22]). As the semantics of \( \text{KtT4} \) does not enforce monotonicity of interpretations, atoms must be translated as future necessities or past possibilities (these are always monotone): \( p^\# = \square p \) (or \( \lozenge p \)); \( \top^\# = \top \); \( \bot^\# = \bot \); \((A \land B)^\# = A^\# \land B^\# \); \((A \lor B)^\# = A^\# \lor B^\# \); \((A \rightarrow B)^\# = \square(A \supset B) \); \((A \between B)^\# = \diamond(A \between B) \).

2.2 The standard-style sequent calculus \( \text{LBil} \)

A sequent calculus for \( \text{BiInt} \) is most easily obtained by extending Maehara–Dragalin-style [23, 9] sequent calculus for \( \text{Int} \), as has been done by Restall [35] and Crolard [6]. (Rauszer’s [31] original sequent calculus was different, as it required sequents to have a single formula in the antecedent or in the succedent.) In the Maehara–Dragalin-style system for \( \text{Int} \), sequents are multiple-conclusion, but the \( \supset R \) rule is constrained. The extension, which we will now show, imposes a dual constraint on the \( \prec L \) rule.

The sequents of our calculus (henceforth referred to as the standard-style calculus \( \text{LBil} \)) are pairs \( \Gamma \vdash \Delta \) where \( \Gamma, \Delta \) (the antecedent and succedent) are finite multisets of formulas (we omit braces and denote union by comma as usual). The inference rules of \( \text{LBil} \) are shown in Figure 2.

Note that \( \Delta \) is missing in the premise of the \( \supset R \) rule; dually, in the premise of \( \prec L \) we do not have the context \( \Gamma \).

Regarding structural rules, both in \( \text{LBil} \) and the other two sequent calculi considered in this paper (\( \text{N-LBil} \) and \( \text{L-LBil} \)), we have chosen to work with formulations optimized for bottom-up proof search, which means that, as a general guideline, we want to have our inference rules ‘as invertible as possible’. We have weakening and contraction built in to the other rules to the degree that \( \text{LBil} \) is complete without explicit versions of them. This requires of course that the two-premise rules are context-sharing etc. But there are also more specific consequences: in \( \text{LBil} \), we have duplication of the main formula in the rules \( \supset L \) and \( \prec R \) in the first, resp. the second premise (when the rules are read bottom-up).

\( \text{LBil} \) is sound and complete for the Kripke semantics of \( \text{BiInt} \) for the following generalization of validity from formulas to sequents. A sequent \( \Gamma \vdash \Delta \) is taken to be valid if, for any Kripke structure
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

Initial rule and cut (necessary):

\[
\frac{\Gamma, A \vdash A, \Delta}{\Gamma, A \vdash \Delta} \quad \text{hyp} \quad \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash \Delta}{\Gamma \vdash \Delta} \quad \text{cut}
\]

Logical rules:

\[
\frac{\Gamma \vdash \Delta}{\Gamma, \top \vdash \Delta} \quad \top L \quad \frac{\Gamma \vdash \top, \Delta}{\Gamma \vdash \top, \Delta} \quad \top R
\]
\[
\frac{\Gamma, \bot \vdash \Delta}{\Gamma \vdash \bot, \Delta} \quad \bot L \quad \frac{\Gamma \vdash \bot, \Delta}{\Gamma \vdash \bot, \Delta} \quad \bot R
\]
\[
\frac{\Gamma \vdash A \quad \Gamma \vdash B \vdash \Delta}{\Gamma, A \lor B \vdash \Delta} \quad \lor L
\quad \frac{\Gamma \vdash A, B \vdash \Delta}{\Gamma \vdash A \lor B, \Delta} \quad \lor R
\]
\[
\frac{\Gamma \vdash A \supset B \vdash A, \Delta \quad \Gamma \vdash B \vdash \Delta}{\Gamma, A \supset B \vdash \Delta} \quad \sup L
\quad \frac{\Gamma \vdash A \supset B, \Delta}{\Gamma \vdash A \supset B \vdash \Delta} \quad \sup R
\]
\[
\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A, \Delta} \quad \text{weakL}
\quad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A, \Delta} \quad \text{weakR}
\quad \frac{\Gamma \vdash A, A \vdash \Delta}{\Gamma \vdash A, \Delta} \quad \text{contrL}
\quad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta} \quad \text{contrR}
\]

Structural rules (eliminable):

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma, A \vdash \Delta} \quad \text{contrL}
\quad \frac{\Gamma \vdash A, A \vdash \Delta}{\Gamma \vdash A, \Delta} \quad \text{contrR}
\]

\( (W, \leq, I) \) and any world \( w \), we have that if all formulas in \( \Gamma \) are true in \( w \), then so is some formula in \( \Delta \). This has been proved (for variants of \( \text{LBiI} \)), e.g., by Restall [35] and Monteiro [24].

However, \( \text{LBiI} \) is incomplete without cut, as shown by the authors in 2003 (private email message from the second author to R. Goré, 13 September 2004, quoted in [4]). It suffices to consider the obviously valid sequent \( p \vdash q, r \supset ((p \supset q) \land r) \). The only possible last inference (other than weakening and contraction, which are redundant) in a derivation could be

\[
\frac{p, r \vdash (p \supset q) \land r}{p \vdash q, r \supset ((p \supset q) \land r)} \quad \sup R
\]

but the premise is invalid as the succedent formula \( q \) has been lost. With cut, the sequent can be proved as follows:

\[
\frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \text{hypothesis} \quad \frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \text{hypothesis} \quad \frac{p, p \supset q, r \vdash q}{p, p \supset q, r \vdash q} \quad \text{hypothesis} \quad \frac{p, p \supset q, r \vdash q}{p, p \supset q, r \vdash q} \quad \text{hypothesis}
\]
\[
\frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \land R \quad \frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \land R \quad \frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \land R
\]
\[
\frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \sup R \quad \frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \sup R \quad \frac{p \vdash q, p \supset q, \ldots}{p \vdash q, p \supset q, \ldots} \quad \sup R
\]
\[
\frac{p \vdash q, r \supset ((p \supset q) \land r)}{p \vdash q, r \supset ((p \supset q) \land r)} \quad \text{cut}
\]

Notice that permutation of the cut on the exclusion \( p \supset q \) up past the \( \sup R \) inference, for which the cut formula is a side formula, is not possible. There are altogether three such types [24]. This situation is reminiscent of the naive standard-style sequent calculus for \( \text{S5} \) where the sequent \( p \vdash \Box 
\Box \Box p \) cannot be proved without cut, but can be proved by applying cut to the sequents \( p \vdash \Box p \) and \( \Box p \vdash \Box \Box p \) that are both provable without cut.
Inference rules of $\mathbf{N-LBiI}$ for manipulating nested sequents

In Section 6, with the help of the translations proposed in this paper, we will identify classes of cuts complete for $\mathbf{LBiI}$.

2.3 The nested sequent calculus $\mathbf{N-LBiI}$

Next we introduce a calculus $\mathbf{N-LBiI}$ of nested sequents, which is a minor variation of the calculus $\mathbf{LBiInt_1}$ of Goré et al. [17].$^2$ $\mathbf{N-LBiI}$ is an extension of $\mathbf{LBiI}$ where the concept of contexts is generalized so that, alongside formulas, they can also contain nested sequents, manipulated by dedicated additional inference rules.

The sequents of $\mathbf{N-LBiI}$ (ranged over by $S$) are defined simultaneously with contexts (ranged over by $\Gamma$, $\Delta$) by the following grammar:

$$S ::= \Gamma \vdash \Delta$$

$$\Gamma, \Delta ::= \emptyset \mid A, \Gamma \mid S, \Gamma$$

where contexts, just as in $\mathbf{LBiI}$, are quotiented down to multisets (so identified up to permutations of the member formulas/nested sequents). Just as commas in antecedents and succedents intuitively correspond to conjunctions and disjunctions, nested turnstiles should be understood as structural-level implications and exclusions.$^3$

The inference rules of $\mathbf{N-LBiI}$ are those of $\mathbf{LBiI}$ in Figure 2 (including the cut rule and the structural rules) together with the additional inference rules for introducing and eliminating nested sequents in Figure 3. The nestL and nestR rules are structural versions of $\land L$ and $\lor R$. The unnestL/R rules are elimination rules for exclusions on the left and implications on the right. It is fair to think of them as masqueraded versions of certain quite specific types of cuts (as we show in Section 6).$^4$

Stating soundness and completeness of $\mathbf{N-LBiI}$ requires defining what it means for a nested sequent to be valid. This is achieved via a translation that ‘flattens’ nested sequents into standard sequents, reducing validity of nested sequents to that of standard sequents. We give a formal definition of this translation of sequents in Section 4.1, where we show that derivations of $\mathbf{N-LBiI}$ can be translated into $\mathbf{LBiI}$. Goré et al. [17] established soundness of $\mathbf{N-LBiI}$ wrt. this notion of validity directly, but showed completeness by an embedding of Rauszer’s sequent calculus [31]. They also

$^2$ The difference is that $\mathbf{LBiInt_1}$ does not build contraction into logical rules, as we do in the case of $\land L$, $\lor R$, $\lor L$, $\lor R$, and that $\mathbf{LBiInt_1}$ has a context-splitting version of the cut rule, whereas our cut rule is context-sharing.

$^3$ In [17], a nested sequent in the antecedent, resp. succedent, of a parent sequent (a structural-level exclusion, resp. implication) is written $\Gamma < \Delta$, resp. $\Gamma > \Delta$.

$^4$ For simplicity, we adopted a formulation of unnestL/R rules that does not incorporate formula contraction. For a version of $\mathbf{N-LBiI}$ that is complete without the contrL/R rules, the unnestL/R rules must be stated differently. In fact, this is adopted in the nested system $\mathbf{N_0-LBiI}$ in Section 5.
showed cut to be redundant in the strong sense of existence of a cut-eliminating transformation of derivations [17] (we will come back to this in Section 6). The example of the previous subsection is proved in $\text{N-LBiL}$ without cut (but with $\text{unnestL}$) as follows:

$$
\begin{array}{c}
\frac{p \vdash q, p \ \text{hyp}}{p, q \vdash q, p \text{ hyp}} \\
\frac{p, q \vdash q, p < q \ \text{hyp}}{(p \vdash q), r \vdash p < q \ \text{nestL}} \\
\frac{(p \vdash q), r \vdash p < q \ \text{nestL}}{(p \vdash q), r \vdash (p < q) \land r \ \text{R}} \\
\frac{(p \vdash q), r \vdash (p < q) \land r \ \text{R}}{p \vdash q, r \vdash ((p < q) \land r) \ \text{unnestL}}
\end{array}
$$

(1)

3 A labelled sequent calculus for BiInt

3.1 The system $\text{L-LBiL}$

The third sequent calculus we consider is the labelled sequent calculus $\text{L-LBiL}$ introduced in [28], a variation on the calculus $L$ of ours [27] (whose differences we will explain later). The design of $\text{L-LBiL}$ follows the method of Negri [25] for obtaining cut-free sequent calculi for normal modal logics defined by frame conditions of a certain type. Essentially, $\text{L-LBiL}$ is a formalization of the first-order theory of the Kripke semantics of $\text{BiInt}$, using an explicit device of labels for worlds, which, in addition, reflects some proof search concerns, e.g., by considering only invertible rules.

A sequent of $\text{L-LBiL}$ is a triple $\Gamma \vdash_G \Delta$ where $G$ is a label tree and $\Gamma$ and $\Delta$ are labelled contexts. More precisely, the label tree $G = (N, E)$ is an oriented graph, i.e., a directed graph without 2-cycles. This means that we have a nonempty set $N$ of nodes, called labels, together with an asymmetric (i.e., irreflexive and antisymmetric) set $E \subseteq N \times N$ of edges. This oriented graph is required to have the property that the corresponding simple graph $(N, E \cup E^{-1})$ is a tree, i.e., that any two nodes in $N$ are connected by exactly one path made of edges from $E \cup E^{-1}$. We write $|G|$ for $N$ and $xGy$ for $(x, y) \in E$. Labelled contexts $\Gamma$ and $\Delta$ are multisets of labelled formulas and these, in their turn, are pairs $x : A$ with $x$ a label drawn from $|G|$ and $A$ a formula. We identify sequents up to renaming of labels.

To represent label trees it will be handy to use label tree expressions given by the following grammar:

$$G_1, G_2 := \langle x \rangle (x, y) | G_1 \oplus_x G_2$$

Here the second construct requires $x$ and $y$ to be distinct labels and the third construct requires exactly one label $x$ to occur in both $G_1$ and $G_2$. Label tree expressions represent label trees in the following way: $(x)$ represents the tree with a single node $x$; $(x, y)$ represents the label tree $G$ with two nodes $x$, $y$ and one edge from $x$ to $y$; $G_1 \oplus_x G_2$ represents the label tree $(N_1 \cup N_2, E_1 \cup E_2)$, if $G_1$ and $G_2$ represent label trees $(N_1, E_1)$, resp. $(N_2, E_2)$. The requirement that $x$ is the unique label occurring simultaneously in $G_1$ and $G_2$ guarantees that the simple graph $(N_1 \cup N_2, E_1 \cup E_2 \cup E_1^{-1} \cup E_2^{-1})$ is indeed a tree. Conversely, any label tree can be represented by a label tree expression, generally in many ways: roughly speaking, any chosen edge $(x, y)$ of a label tree $G$ determines two disjoint smaller trees composed of the labels connected to $x$ and composed of the labels connected to $y$ after removal of $(x, y)$, represented by $G_1$ and $G_2$ say, and then $G$ is represented by $G_1 \oplus_x (x, y) \oplus_y G_2$. For example, the three label tree expressions $(x) \oplus_3 (x, y) \oplus_3 (z, y), (z) \oplus_2 (z, y) \oplus_2 (x, y)$ and $(z, y) \oplus_2 (x, y)$ all represent the label tree $\langle (x, y, z), ((x, y), (z, y)) \rangle$. In what follows, when we write a label tree expression, we mean generally the label tree it represents.
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

Initial rule:
\[ \Gamma, x : p \vdash_G x : p, \Delta \quad \text{hyp} \]

Logical rules:
\[
\begin{align*}
\Gamma, x : \top \vdash_G \Delta & \quad \top L \\
\Gamma, x : \top \vdash_G \Delta & \quad \top R \\
\Gamma, x : \bot \vdash_G \Delta & \quad \bot L \\
\Gamma, x : \bot \vdash_G \Delta & \quad \bot R \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \wedge L \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \wedge R \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \vee L \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \vee R \\
\Gamma, y : A \vdash_G x, y \vdash_G \Delta & \quad \supset L \\
\Gamma, y : A \vdash_G x, y \vdash_G \Delta & \quad \supset R \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \prec L \\
\Gamma, x : A \vdash_G B \vdash_G \Delta & \quad \prec R
\end{align*}
\]

Monotonicity rules:
\[
\begin{align*}
\Gamma, x : A, y : A \vdash_G \Delta & \quad monot L \\
\Gamma, y : A, x : A \vdash_G \Delta & \quad monot R
\end{align*}
\]

proviso: A atomic or an implication
proviso: A atomic or an exclusion

Figure 4 Inference rules of L-LBiI

We will need label substitution operations \(G[x/y]\) and \(\Gamma[x/y]\) to substitute label \(x\) for label \(y\), in a label tree \(G\), resp. a context \(\Gamma\), defined in the obvious way. However, in the case of \(G[x/y]\), there is the proviso \(x \not\in |G|\), otherwise we are not guaranteed to obtain a well-formed label tree.

Below we use the notations \(G^+\) and \(G^\ast\) for the transitive, resp. reflexive-transitive, closure of \(G\) and the notations \(G \downarrow y\) and \(G \uparrow y\) to mean that there is no \(z\) such that \(zGy\), resp. \(yGz\). Sometimes we will say \(x\) is to the past of \(y\) or \(y\) is to the future of \(x\) meaning that \(xG^+y\).

Intuitively, we use label trees to represent Kripke trees and a labelled formula is about truth at a particular world. Formally, a labelled sequent \(\Gamma \vdash_G \Delta\) is valid, if, for any Kripke structure \((W, \leq, I)\) and function \(v : |G| \to W\) such that \(xGy\) implies \(v(x) \leq v(y)\), we have that, if \(v(x) \models A\) for every \(x : A\) in \(\Gamma\), then \(v(x) \models A\) for some \(x : A\) in \(\Delta\).

The inference rules of L-LBiI are presented in Figure 4. In all rules, the condition that all the premises and the conclusion are well-formed labelled sequents must be read as a proviso. In particular, note that the label tree expression \(G \oplus_x (y, x)\), resp. \((y, x) \oplus_x G\), in the premise of rule \(\supset R\), resp. \(\prec L\), requires \(y \not\in |G|\) and consequently requires also the freshness condition \(y \not\in \Gamma\) and \(y \not\in \Delta\) (in order to ensure the conclusion to be a well-formed labelled sequent).

The system L-LBiI has primitive monotonicity rules (as the system \(L\) of [27]), which perform propagation of truth and falsity to future, resp. past, worlds (reading the rules from the conclusion to the premise). The monotonicity rules are restricted to certain kinds of formulas (as in the system...
of [27]), but we will show that unrestricted monotonicity is admissible in L-LBiI. Similarly, axioms of L-LBiI are restricted to atoms, but of course we will show that non-atomic axioms are admissible in L-LBiI. These restrictions become handy when considering meta-theoretic properties of L-LBiI (and, a posteriori, of the other systems).

Note that the structural rules and the cut rule are not primitive in L-LBiI, contrasting with LBiI and N-LBiI. In order to guarantee admissibility of contraction, the rules ⊃L and ⊘R need to duplicate the main formula when going from the conclusion to the first resp. the second premise, similarly to what happens in Kleene’s G3 system [19] for intuitionistic logic.

Our counterexample to cut elimination in LBiI can be proved in L-LBiI as follows:

\[
\begin{align*}
\begin{array}{c}
\vdash (x, y) \quad \vdash (x, y) \\
\hline
\vdash x, \vdash q, \vdash (x, y) \quad \vdash q, \vdash p \prec q, \vdash (x, y) \\
\hline
\vdash x, \vdash q, y : p \prec q, \vdash (x, y) \\
\hline
\vdash x, \vdash q, y : r, \vdash (p \prec q) \land r, \vdash (x, y) \\
\hline
\vdash x, \vdash q, y : r, \vdash (p \prec q) \land r, \vdash (x, y) \\
\hline
\end{array}
\end{align*}
\]

Notice the propagation of \( p \prec q \) to the past that occurs at the \textit{monotR} step to an already existing label.

L-LBiI is sound and complete wrt. the notion of validity introduced above. We will not show this directly. Yet observe that the label tree version of the algorithmic system \( L^* \) of [27] (whose completeness wrt. the same notion of validity was shown in [27]) embeds easily into L-LBiI. Observe that completeness of L-LBiI also follows immediately from the embeddings into L-LBiI in Section 4 of the standard-style and the nested-style systems LBiI and N-LBiI and completeness of these systems.

### 3.2 Meta-theory of L-LBiI

Now we establish the main meta-theoretic properties of L-LBiI. We show admissibility of weakening, invertibility of all the logical rules, admissibility of nodemerge (merging two nodes), admissibility of full monotonicity, contraction and cut, and also admissibility of nodesplit (splitting a node into two). Quite some care is needed with the order in which these results are proved. In particular, invertibility and admissibility of nodemerge is done in two stages.

**Proposition 1** (General axiom)

\[ \Gamma, x : A \vdash_G x : A, \Delta \text{ is derivable (for any formula } A). \]

**Proof.** By easy induction on \( A \). The cases for implication and exclusion require monotonicity steps.

**Proposition 2** (Weakening)

Weakening is height-preserving admissible. Specifically, if \( \Gamma \vdash_G \Delta \) has a derivation of height \( n \), then both \( \Gamma, x : A \vdash_G \Delta \) and \( \vdash_G x : A, \Delta \) have derivations of height \( n \) (for any \( x \in |G| \)).

**Proof.** Routine induction on derivation height.
All rules of \textbf{L-LBiI} are invertible. Firstly we prove that all rules except for $\top L$ and $\top R$ are invertible. Invertibility of $\top L$ and $\top R$ is proved later, simultaneously with admissibility of full monotonicity and contraction.

**Proposition 3 (Invertibility)**
All rules except for $\top L$ and $\top R$ are invertible. In particular, for the rule $\top R$, if $\Gamma \vdash_G x : A \top B, \Delta$, then $\Gamma', y : A \vdash_{G \oplus_z (x,y)} y : B, \Delta$ for any $y \notin |G|$.

**Proof.** The results follow by easy inductions on derivation height. For example, for $\top R$, we have that $x : A \top B$ can only be principal at an $\top R$ step (because of the restrictions on hyp and monot$R$ rules) and, in this case, the conclusion follows immediately ($y$ can be any label not in $G$, since sequents are identified up to renaming of labels). The cases where $x : A \top B$ is not principal follow by the IH and reapplication of the rule used in the last step.

Next we study the \textit{nodemerge} rules, which play an important role in the rest of the paper, since they are used both to obtain admissibility of contraction, full monotonicity and cut in the labelled system \textbf{L-LBiI} as well as to justify the translation in Section 5 of the nested system \textbf{N-LBiI} into \textbf{L-LBiI}. A \textit{nodemerge} rule is a principle for merging a label $y$ with a label $x$ in a given sequent whereby the resulting sequent has one label less (label $y$) and has $x$-labelled formulas where there were previously $y$-labelled formulas. However, this principle is not valid for arbitrary labels. In fact, arbitrary merging might even fail to produce a label tree out of a label tree (recall the beginning of this section, where we require $x \notin |G|$ to write $G[x/y]$). If $x$ and $y$ are adjacent nodes in the label tree (i.e., there is either an edge $(x,y)$ or an edge $(y,x)$ in the graph), then merging $y$ with $x$ is always sound. In Proposition 6, we will indeed show admissibility of nodemerge for adjacent labels. If $x$ and $y$ are not adjacent, the merging might produce a proper label tree, but break some of the past/future relations between labels and be unsound. For example, in the valid sequent $w : p \vdash_{(w,y)\oplus_0 (x,y)\oplus_z (x,z)} x : p$, we can merge $y$ with $z$ (using the nodemerge$F$ rule from the next proposition, but violating the side condition) and obtain $w : p \vdash_{(w,z)\oplus_0 (x,z)} x : p$. But since $w$ is no longer to the past of $x$, this is no longer a valid sequent. Still, merging a label $y$ with a non-adjacent node $z$ is possible under a side condition, as stated in the next proposition.

**Proposition 4 (Nodemerge I)**
The following rules are admissible.\footnote{The suffixes $F$ and $P$ in the rule names, for ‘future’, resp. ‘past’, are better motivated by the cases of these rules in Proposition 6, where the side condition is $z = x$ and $y$ is merged with a label that is to the future, resp. to the past of $y$.}

\[
\begin{align*}
\frac{z G^+ x}{\Gamma[z/y] \vdash_{G_0[y/z] \oplus_z G} \Delta[z/y]} \quad \text{nodemerge} F \\
\frac{x G^+ z}{\Gamma[z/y] \vdash_{G_0[z/y] \oplus_z G_0} \Delta[z/y]} \quad \text{nodemerge} P
\end{align*}
\]

**Proof.** Firstly, note that each rule has a well-formed label tree in the conclusion: it is implicit in the premises that $z \in |G|$, $y \in |G_0|$ and $|G_0| \cap |G| = \emptyset$, so $G_0[z/y]$ is well-defined ($z \notin |G_0|$) and $|G_0[z/y]| \cap |G| = \{z\}$.

The proofs are analogous for the two rules and follow by induction on derivation height. We concentrate on nodemerge$P$ and develop the case where the last step is

\[
\frac{w (G \oplus_{\alpha} (x,y) \oplus y G_0) \Gamma', w : A, w' : A \vdash_{G_0} \Delta}{\Gamma', w : A \vdash_{G_0} \Delta} \quad \text{monot} L
\]
Proposition 5 (Admissibility of full monotonicity and contraction)

Let $A = B \supset C$, and, for $k \geq 0$, let $z_1, \ldots, z_k$ be labels such that $xG^+z_i$, for all $i$. If $\Gamma, x : A, z_1 : A, \ldots, z_k : A \vdash G \Delta$, then $\Gamma, x : C \vdash G \Delta$.

Let $A = B \prec C$, and, for $k \geq 0$, let $z_1, \ldots, z_k$ be labels such that $z_iG^+x$, for all $i$. If $\Gamma \vdash_G x : A, z_1 : A, \ldots, z_k : A, \Delta$, then $\Gamma \vdash_G x : B, \Delta$.

If $\Gamma, x : A, y : A \vdash G \Delta$ and $xGy$ or $x = y$, then $\Gamma, x : A \vdash G \Delta$.

If $\Gamma \vdash_G x : A, y : A, \Delta$ and $yGx$ or $x = y$, then $\Gamma \vdash_G x : A, \Delta$.

Proof. The four statements are proved simultaneously by induction on the pair $(|A|, |\pi|)$, where $|.|$ stands for height and $\pi$ stands for the derivations implicit in the hypotheses of the four statements. Statements (1) and (2) are proved analogously, and statements (3) and (4) also have analogous proofs.

For proving (1), we consider the various cases that can arise in the last step of the derivation of $\Gamma, x : B \supset C, z_1 : B \supset C, \ldots, z_k : B \supset C \vdash G \Delta$.

(i) The cases where $x : B \supset C$ and each of $z_i : B \supset C$ are not principal in the last step follow by the IH and reapplication of the same rule.
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

In the cases where the last step is $\text{monot}L$ and either $x : B \supset C$ or one of $z_i : B \supset C$ is principal, the premise has an extra $z : B \supset C$ with $xGz$; so, the IH applies and immediately gives the result.

(iii) Suppose some $z_i : B \supset C$ is principal and the last step is $\supset L$. So, the second premise of this step is $\Gamma, x : B \supset C, z_1 : B \supset C, \ldots, z_k : B \supset C \vdash \Gamma, \Delta$, and the IH gives $\Gamma, x : C, z_i : C \vdash \Gamma, \Delta$. Since $xGz$, for some $m \geq 0, y_1, \ldots, y_m$, we have $xGy_1G\ldots Gy_mGz_i$. The latter derivation can be weakened to give $\Gamma, x : C, y_1 : C, \ldots, y_m : C, z_i : C \vdash \Gamma, \Delta$. Hence, using repeatedly part (3) of the IH on subformula $C$, we conclude $\Gamma, x : C \vdash \Gamma, \Delta$.

(iv) If $x : B \supset C$ is principal and the last step is $\supset L$, the proof is analogous to the previous case.

Let us analyse the cases of (4) for monotonicity, i.e., where $yGx$.

(i) If $A$ is an atom or an exclusion, the result follows immediately by applying $\text{monot}R$.

(ii) If neither $x : A, y : A$ is principal in the last step, part (4) of the IH is applied to the premises, and the same rule reapplied.

(iii) The cases where either $x : A$ or $y : A$ is principal and $A$ is a conjunction or disjunction follow by: invertibility of $\land R$ and $\lor R$; use of part (4) of the IH on the conjuncts/disjuncts; reapplication of $\land R/\lor R$.

(iv) Finally, suppose $A = B \supset C$ and $x : A$ is principal (if $y : A$ is principal, we argue analogously). So we have a derivation whose last step has the form

\[
\frac{\Gamma, w : B \vdash G_{\oplus}(x, w) \quad w : C, y : B \supset C, \Delta}{\Gamma \vdash \Gamma, x : B \supset C, y : B \supset C, \Delta}
\]

By invertibility of $\supset R$ (Proposition 3),

\[
\frac{\Gamma, w : B, z : B \vdash (y, z) \oplus yG_{\oplus}(x, w) \quad w : C, z : C, \Delta}{\Gamma, w : B \vdash \Gamma, \Delta}
\]

So, since $y(G \oplus x(w))^w$, by $\text{node}mergeP$ (Proposition 4),

\[
\frac{\Gamma, w : B, w : B \vdash G_{\oplus}(x, w) \quad w : C, w : C, \Delta}{\Gamma, w : B \vdash \Gamma, \Delta}
\]

Hence, by parts (3) and (4) of the IH for the subformulas $B$ and $C$,

\[
\frac{\Gamma, w : B \vdash G_{\oplus}(x, w) \quad w : C, \Delta}{\Gamma, w : B \vdash \Gamma, \Delta}
\]

Finally, by $\supset R$,

\[
\frac{\Gamma \vdash \Gamma, x : B \supset C, \Delta}{\Gamma \vdash \Gamma, x : B \supset C, \Delta}
\]

We also demonstrate two cases of (4) for contraction ($x = y$), namely the cases where the last step introduces the contracted formula through $\supset R$ or $\supset R$. The other cases are simpler.

(i) The case where we have a derivation whose last step has the form

\[
\frac{\Gamma, z : B \vdash G_{\oplus}(x, z) \quad x : B \supset C, z : C, \Delta}{\Gamma \vdash \Gamma, x : B \supset C, x : B \supset C, \Delta}
\]

This case is solved similarly to the last case of (4) above for monotonicity: invertibility of $\supset R$ gives $\Gamma, z : B, w : B \vdash (x, w)G_{\oplus}(x, z) \quad z : C, w : C, \Delta$; then, $\text{node}mergeP$ gives $\Gamma, z : B, z : B \vdash G_{\oplus}(x, z) \quad z : C, z : C, \Delta$; hence, parts (3) and (4) of the IH on the subformulas $B$ and $C$ give $\Gamma, z : B \vdash G_{\oplus}(x, z) \quad z : C, \Delta$; finally, $\supset R$ gives $\Gamma \vdash \Gamma, x : B \supset C, \Delta$. 

Downloaded from https://academic.oup.com/logcom/article-abstract/28/1/165/4807375 by guest on 29 March 2018
COROLLARY 1 (Invertibility of $\supset$ and $\neg R$)

1. If $\Gamma, x : A \supset B \vdash_G \Delta$, then (i) $\Gamma, x : A \supset B \vdash_G x : A, \Delta$ and (ii) $\Gamma, x : B \vdash_G \Delta$.
2. If $\Gamma \vdash_G x : A \times B, \Delta$, then (i) $\Gamma, x : A \times B \vdash_G x : A, \Delta$ and (ii) $\Gamma \vdash_G x : A, \Delta$.

PROOF. Let us argue about (1). Note that (i) follows from the assumption by weakening. For (ii), we use (1) of the previous proposition, taking $k = 0$.

We establish now the nodemerge principles for adjacent labels. These results could not be proved together with the nodemerge principles for non-adjacent labels (Proposition 4) because they require contraction.

PROPOSITION 6 (Nodemerge II)
The following rules are admissible:

\[
\begin{align*}
\Gamma \vdash_G x : B, x : B \times C, \Delta & \quad \Gamma, x : C \vdash_G x : B \times C, x : B \times C, \Delta \\
\Gamma \vdash_G x : B \times C, x : B \times C, \Delta & \quad \therefore \Gamma \vdash_G x : B \times C, \Delta
\end{align*}
\]

Applying part (2) of the IH to the first premise, we get $\Gamma \vdash_G x : B, x : B, \Delta$, from which, by the IH on the subformula $B$, we obtain $\Gamma \vdash_G x : B, \Delta$. From this and derivability of $\Gamma, x : C \vdash_G x : B \times C, \Delta$, which follows by applying part (4) of the IH to the second premise, we get $\Gamma \vdash_G x : B \times C, \Delta$ through a $\neg R$ step.

We establish now the nodemerge principles for adjacent labels. These results could not be proved together with the nodemerge principles for non-adjacent labels (Proposition 4) because they require contraction.

PROPOSITION 6 (Nodemerge II)
The following rules are admissible:

\[
\begin{align*}
\Gamma \vdash_G x : B, x : B \times C, \Delta & \quad \Gamma, x : C \vdash_G x : B \times C, x : B \times C, \Delta \\
\Gamma \vdash_G x : B \times C, x : B \times C, \Delta & \quad \therefore \Gamma \vdash_G x : B \times C, \Delta
\end{align*}
\]

PROOF. By induction on derivation height. We develop for nodemergeP two cases relative to monotonicity (one of them needing contraction). The other cases are either analogous or simpler. Assume the derivation of the premise has a last step of the form

\[
\Gamma', x : A, y : A \vdash_G x \oplus_y G_0 \Delta
\]

Applying the IH gives $\Gamma'[x/y], x : A, x : A \vdash G_\oplus x, G_0[x/y] \Delta[x/y]$, and then contraction (Proposition 5) gives $\Gamma'[x/y], x : A \vdash G_\oplus x, G_0[x/y] \Delta[x/y]$.

Assume now the derivation of the premise of nodemergeP ends with a step of the form

\[
\Gamma', y : A \vdash_G x \oplus_y G_0 \Delta
\]

Then $yG_0z$, hence $xG_0[x/y]z$. This and the IH allow a derivation ending with

\[
\Gamma'[x/y], x : A \vdash_G x \oplus_y G_0[x/y] \Delta[x/y]
\]

Now we turn to admissibility of cut. The proof follows partly standard ideas, but the presence of explicit rules for monotonicity in L-LBiI introduces new issues, as more cases where the cut formula is principal in both premises arise.
Before going to the general ideas, let us instructively see how the cut-admissibility proof below works on an example related to our counter-example in Section 2.2 to eliminability of cuts in the standard-style sequent calculus LBiI.

Consider the following derivation with a cut, where side formulas not contributing to subderivations are generally omitted:

\[
\begin{array}{c}
\frac{\pi}{x : p \vdash (x) \quad x : q} \quad \frac{\pi}{x : q \vdash (x) \quad x : p < q} \\
\frac{\pi}{x : p \vdash (x) \quad x : q, x : p < q} \quad \frac{x : p < q, y : p < q}{x : p < q, y : r} \quad \frac{x : p < q, y : r}{x : p < q, y : (p < q) \land r} \quad \frac{x : r \vdash ((p < q) \land r)}{\exists R} \quad \frac{y : r \vdash (x)}{\forall R} \quad \frac{\exists R}{\text{cut}} \\
\frac{x : p \vdash (x) \quad x : q, x : r \vdash ((p < q) \land r)}{\vdash (\exists R)}
\end{array}
\]

where \(\pi\) is as follows:

\[
\begin{array}{c}
\frac{z : p \vdash (x) \land (x, y)}{z : p} \quad \frac{z : q \vdash (x) \land (x, y)}{z : q} \quad \frac{z : p \vdash (x) \land (x, y)}{z : q, y : p < q, x : p < q, z : p < q} \\
\frac{z : p \vdash (x) \land (x, y)}{z : q, y : p < q, x : p < q} \quad \frac{\exists R}{\text{monot R}}
\end{array}
\]

As the cut formula is not principal in the last step of the derivation of the second premise of the cut inference, we permute the cut upwards, until the cut formula becomes principal in the derivations of both premises, and remove a trivial cut created on the second premise of the \(\exists R\) step, obtaining:

\[
\begin{array}{c}
\frac{x : p \vdash (x) \quad x : q \vdash (x)}{x : p \vdash (x)} \quad \frac{\exists R}{\text{monot R}}
\end{array}
\]

Now, we can apply to \(\pi\) the algorithm behind admissibility of nodemergeF, obtaining the following derivation \(\pi'\):

\[
\begin{array}{c}
\frac{x : p \vdash (x) \quad x : q \vdash (x) \quad x : p < q}{x : p \vdash (x)} \quad \frac{\exists R}{\text{monot R}}
\end{array}
\]

Since the contraction step can be permuted upwards and eventually eliminated when it reaches the leaves, and since the last step of \(\pi'\) is already the conclusion of the cut inference above, we can simply remove the cut inference, arriving at the cut-free derivation of the sequent

\[
x : p \vdash (x) \quad x : q, x : r \vdash ((p < q) \land r)
\]
we already encountered in Section 3.1. This removal of the cut inference in the general cut-admissibility procedure happens through the generation of two atomic cuts (one on \(x : p\) and the other on \(x : q\)), each having an axiom as one of the premises.

Now we present the general argument for admissibility of cuts, starting with the case of cuts on atomic formulas.

**Lemma 1 (Admissibility of atomic cuts)**

For all \(n \geq 0\),

1. If \(\Gamma \vdash G x : p, \Delta\) and \(\Gamma, x : p, y_1 : p, \ldots, y_n : p \vdash_G \Delta\) and \(xG^+y_i\) for all \(y_i\), then \(\Gamma \vdash_G \Delta\);

2. If \(\Gamma, x : p, y_1 : p, \ldots, y_n : p, \Delta\) and \(\Gamma, x : p \vdash_G \Delta\) and \(y_iG^+x\) for all \(y_i\), then \(\Gamma \vdash_G \Delta\).

**Proof.** Let us prove (1). (2) is analogous.) The proof is by induction on the height of the derivation (\(\pi\) say) of the second sequent. The cases where none of \(x : p, y_1 : p, \ldots, y_n : p\) is principal in the last step follow by applying the IH to the premise(s) (possible because we can invert \(\Gamma \vdash_G x : p, \Delta\) and reapplying the rule used in the last step (in the case of an axiom, \(\Gamma \vdash_G \Delta\) is also an axiom). Let us prove (1) in the other cases.

The case where \(\pi\) is an axiom: If \(x : p \in \Delta\), then the result follows by applying contraction (Proposition 5) to the first sequent. If for some \(i\), \(\Delta = y_i : p, \Delta'\), then, by the hypothesis, \(xGzG\ldots GzGy_i\) and the following derivation can be built from the first premise (and weakening) by successive applications of \(\text{monotR}\):

\[
\begin{align*}
\Gamma \vdash_G x : p, y_i : p, z_1 : p, \ldots, z_k : p, \Delta'^+ \quad \text{monotR} \\
\Gamma \vdash_G y_i : p, z_1 : p, \ldots, z_k : p, \Delta' \quad \text{monotR} \\
\vdots \\
\Gamma \vdash_G y_i : p, z_i : p, \Delta' \quad \text{monotR} \\
\Gamma \vdash_G y_i : p, \Delta'
\end{align*}
\]

The case of \(\pi\) ending with \(\text{monotL}\): In this case, there is \(z\) such that \(xGz\) or \(y_iGz\) for some \(i\) and \(\Gamma, x : p, y_1 : p, \ldots, y_n : p, z : p \vdash_G \Delta\) has a derivation lower than \(\pi\). So, by the IH, \(\Gamma \vdash_G \Delta\). 

**Theorem 1 (Admissibility of cut)**

The following cut rule is admissible:

\[
\begin{align*}
\Gamma \vdash_G x : A, \Delta & \quad \Gamma, x : A \vdash_G \Delta \quad \text{cut} \\
\Gamma \vdash_G \Delta
\end{align*}
\]

**Proof.** By induction on the pair \((|A|, |\pi| + |\sigma|)\) where \(|.|\) stands for height and \(\pi\) and \(\sigma\) are the derivations of the first and second premises respectively.

1. The case where \(A\) is atomic has already been proved (Lemma 1).

2. The cases where \(x : A\) is not main in the last step of \(\pi\) or \(\sigma\) follow by permuting the cut upwards (with the help of weakening) and using the IH (on premises having lower derivation height).

3. Let us analyse the two cases where \(A = A_1 \supset A_2\) and \(x : A\) is main in the last step of both \(\pi\) and \(\sigma\). (The case where \(A = A_1 \bowtie A_2\) is analogous and the cases \(A = A_1 \land A_2\) and \(A = A_1 \lor A_2\) are simpler, since monotonicity for conjunctions and disjunctions is not primitive in \(L-LBiI\).)
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

3.1 One form of cut where \( x : A_1 \supset A_2 \) is principal in both premises is

\[
\pi = \frac{\frac{\pi_1}{\Gamma, z : A_1 \vdash G_{\oplus}(x, z) \; z : A_2, \Delta \quad \exists R}}{\Gamma \vdash G \; x : A_1 \supset A_2, \Delta} \quad \frac{\sigma_1}{\Gamma, x : A_1 \supset A_2 \vdash_G x : A_1, \Delta \quad \Gamma, x : A_2 \vdash \Delta \quad \forall L}
\]

\[
\sigma = \frac{\frac{\sigma_1}{\Gamma, x : A_1 \supset A_2 \vdash_G x : A_1, \Delta \quad \Gamma, x : A_2 \vdash \Delta}}{\frac{\rho_1}{\Gamma \vdash_G x : A_1, \Delta}}
\]

The proof transformation is similar to the unlabelled setting, but additionally admissibility of \textit{nodemerge} (Proposition 6) is needed. Specifically: we can weaken \( \pi \) to \( \pi' \) preserving height and, using the IH (on premises having lower derivation height), we can build

\[
\rho_1 = \frac{\frac{\rho_1'}{\Gamma \vdash_G x : A_1 \supset A_2, \Delta, x : A_2}}{\frac{\pi'}{\Gamma, z : A_1 \vdash G_{\oplus}(x, z) \; z : A_2, \Delta \quad \exists R}} \quad \frac{\sigma_1}{\Gamma, x : A_1 \supset A_2 \vdash_G x : A_1, \Delta \quad \forall L}
\]

3.2 The other case where \( x : A_1 \supset A_2 \) is principal in both premises is when \( \pi \) is as above, but \( \sigma \) has the form

\[
\sigma = \frac{\frac{\sigma_1}{xGy \quad \Gamma, x : A_1 \supset A_2, y : A_1 \supset A_2 \vdash_G \Delta}}{\Gamma, x : A_1 \supset A_2 \vdash_G \Delta \quad \text{monotL}}
\]

By induction on \( \sigma_1 \), we argue about the following more general situation: if \( \sigma_1 \) is a derivation of \( \Gamma, x : A_1 \supset A_2, y_1 : A_1 \supset A_2, \ldots, y_n : A_1 \supset A_2 \vdash_G \Delta \) with \( xG^+_i y_i \) for all \( i \) and (as we are assuming) \( \pi_1 \) is a derivation of \( \Gamma, z : A_1 \vdash G_{\oplus}(x, z) \; z : A_2, \Delta \) for \( z \) neither in \( \Gamma \) nor in \( \Delta \), then \( \Gamma \vdash_G \Delta \).

3.2.1 Suppose \( y_1 : A_1 \supset A_2 \) is principal in the last step of \( \sigma_1 \) (the same argument applies in case of a different \( y_i \)). There are two cases, one for \textit{monotL} and the other for \textit{\supset L}. The former follows directly by the inner IH. For the latter, the two premises of the last step of \( \sigma_1 \) are

(i) \( \Gamma, x : A_1 \supset A_2, y_1 : A_1 \supset A_2, \ldots, y_n : A_1 \supset A_2 \vdash_G y_1 : A_1, \Delta \)

(ii) \( \Gamma, x : A_1 \supset A_2, y_1 : A_2, \ldots, y_n : A_1 \supset A_2 \vdash_G \Delta \)

Weakening \( \pi_1 \), we can derive \( \Gamma, z : A_1 \vdash G_{\oplus}(x, z) \; z : A_2, y_1 : A_1, \Delta \). So, using the inner IH on (i), it follows that

(iii) \( \Gamma \vdash_G y_1 : A_1, \Delta \).
Likewise, weakening \( \pi_1 \), we can also derive \( y_1 : A_2, \Gamma, z : A_1 \vdash_{G \oplus y, x, z} z : A_2, \Delta \) and, using the inner IH on \((ii)\), we conclude that

\[
(iv) \quad \Gamma, y_1 : A_2 \vdash_{G} \Delta.
\]

Since \( x \vDash y_1 \), from \( \pi_1 \) and \( \text{nodemergeP} \) (Proposition 4) we have

\[
(v) \quad \Gamma, y_1 : A_1 \vdash_{G} y_1 : A_2, \Delta.
\]

Using the outer IH on the subformulas \( A_1 \) and \( A_2 \), we can combine \((iii)\), \((iv)\) and \((v)\) to obtain \( \Gamma \vdash_{G} \Delta \).

3.2.2 If \( x : A_1 \supset A_2 \) is principal in the last step of \( \sigma_1 \), a similar argument applies, but uses Proposition 6 instead of Proposition 4 and, in order to apply the inner IH, requires height preserving weakening to add \( x : A_1 \supset A_2 \) to the antecedent of the second premise.

3.2.3 The cases where none of \( x : A_1 \supset A_2, y_1 : A_1 \supset A_2, \ldots, y_n : A_1 \supset A_2 \) is principal follow by the inner IH, which can be applied because of invertibility of all rules of \( \text{L-LBiI} \), and by reapplication of the rule of the last step.

We end this section with admissible rules of \( \text{L-LBiI} \) that become handy for the translations of \( \text{LBiI} \) and \( \text{N-LBiI} \) into \( \text{L-LBiI} \). These rules guarantee that, in some cases, a node of the label tree can be split into a pair of nodes connected by an edge, so that no paths are lost.

**Proposition 7** (Nodesplit)

The following rules are admissible:

\[
\begin{align*}
\text{nodesplitF} & : \quad \frac{G \downarrow x \quad \Gamma \vdash_{G_0 \oplus y, G[y/x]} \Delta}{\Gamma \vdash_{G_0 \oplus y, (y:x):\oplus G} \Delta} \\
\text{nodesplitP} & : \quad \frac{G \uparrow x \quad \Gamma \vdash_{G[y/x] \oplus y:G_0} \Delta}{\Gamma \vdash_{G_0 \oplus (x, y):\oplus y:G_0} \Delta}
\end{align*}
\]

**Proof.** Firstly, note that the conclusions have well-formed label trees. In particular, note that implicit in \( G[y/x] \) is the assumption \( y \notin \{G\} \) and implicit in \( G_0 \oplus y:G[y/x] \) is \( y \in \{G[y/x]\} \), hence \( x \in \{G\} \).

The proofs are by induction on derivation height. We concentrate on \( \text{nodesplitF} \) (the argument is analogous for \( \text{nodesplitP} \)) and consider the case where the last step is

\[
\begin{align*}
\text{monotL} & : \quad \frac{z(G_0 \oplus y, G[y/x])w \quad \Gamma', z : A, w : A \vdash_{G_0 \oplus y: G[y/x]} \Delta}{\Gamma', z : A \vdash_{G_0 \oplus y: G[y/x]} \Delta}
\end{align*}
\]

Subcase \( z(G_0 \oplus y, G[y/x])w \): It suffices to use the IH and reapply \( \text{monotL} \), which is possible because \( z(G_0 \oplus y, y, x) \oplus y:Gw \).

Subcase \( z(G_0 \oplus y, G[y/x])w \): The sub-case where both \( z \neq y \) and \( w \neq y \) follows again by the IH and reapplication of \( \text{monotL} \). The side condition \( G \downarrow x \) makes it impossible that \( w = y \) (otherwise \( zGx \)).

So the remaining sub-case is where \( z = y \), hence \( xGw \). By the IH and weakening, it follows that \( \Gamma', y : A, w : A, x : A \vdash_{G_0 \oplus y, (y, x) : \oplus y:G} \Delta \). Thus, applying \( \text{monotL} \) gives \( \Gamma', y : A, x : A \vdash_{G_0 \oplus y, (y, x) \oplus y:G} \Delta \) (since \( x(G_0 \oplus y, (y, x) \oplus y:Gw) \)) and applying \( \text{monotL} \) once more gives \( \Gamma', y : A \vdash_{G_0 \oplus y, (y, x)} \Delta \) (since \( y(G_0 \oplus y, (y, x) \oplus y:G) \)).

The case where the last step is \( \text{monotR} \) is analogous to \( \text{monotL} \). The cases corresponding to the logical rules follow by routine application of the IH.

4 Relating the standard-style to the other sequent calculi

In this section, we study syntactic translations between \( \text{LBiI} \) and the other two systems.
4.1 From N-LBiI to LBiI

As sequents and rules of LBiI are also sequents and rules of N-LBiI, a derivation in LBiI is also a derivation in N-LBiI. Note, however, that a cut in LBiI is rendered by a cut also in N-LBiI. We will revisit this observation in Section 6 to obtain a complete class of cuts for LBiI. For now, we move on to the converse direction.

We define simultaneously two functions on nested contexts $|(-)|^L$ and $|(-)|^R$ that produce formulas. They are meant to be applied to antecedents and succedents of sequents. We also introduce two further functions $\parallel(-)\parallel^L$ and $\parallel(-)\parallel^R$, defined in terms of $|(-)|^L$ and $|(-)|^R$, to produce standard contexts instead of formulas. They are used to translate top-level sequents and avoid unnecessary rewriting of commas as $\land$ or $\lor$.

$$
|\emptyset|^L = \top \quad |\emptyset|^R = \bot
$$

$$
|A, \Gamma|^L = A \land |\Gamma|^L \quad |A, \Gamma|^R = A \lor |\Gamma|^R
$$

$$
|\Gamma_0 \vdash \Delta_0|, \Gamma|^L = (|\Gamma_0|^L \land |\Delta_0|^R) \land |\Gamma|^L
$$

$$
|\Delta_0|, \Gamma|^R = (|\Gamma_0|^L \lor |\Delta_0|^R) \lor |\Gamma|^R
$$

|\emptyset|^L = \emptyset

$$
|A|^L = A, |\Gamma|^L
$$

$$
|A|^R = A, |\Gamma|^R
$$

$$
|\Gamma_0 \vdash \Delta_0|, \Gamma|^L = (|\Gamma_0|^L \land |\Delta_0|^R), |\Gamma|^L
$$

$$
|\Delta_0|, \Gamma|^R = (|\Gamma_0|^L \lor |\Delta_0|^R), |\Gamma|^R
$$

Note that functions $|(-)|^L$ and $|(-)|^R$ are assuming some canonical form of transforming contexts into lists, which can be realized in different ways. But for our purposes this choice is immaterial, since the formulas produced differ only in the order of operands of conjunctions and disjunctions.

Two observations used extensively throughout are: $|\Gamma, \Gamma_0|^L = |\Gamma|^L, |\Gamma_0|^L$ and $|\Gamma, \Gamma_0|^R = |\Gamma|^R, |\Gamma_0|^R$.

**Theorem 2**

If $\Gamma \vdash \Delta$ is derivable in N-LBiI, then $\parallel\Gamma\parallel |^L \vdash \parallel\Delta\parallel |^R$ is derivable in LBiI.

**Proof.** The proof is by induction on the structure of the N-LBiI derivation of $\Gamma \vdash \Delta$. The cases corresponding to rules other than the nesting rules are immediate, since there is a directly matching rule in LBiI.

Case nestR: The given derivation has the form

$$
\Gamma, \Gamma_0 \vdash \Delta_0 \quad \text{nestR}
$$

It can be mapped to

$$
\vdash \Pi \text{ on } \pi

\vdash \parallel\Gamma\parallel |^L, |\Gamma_0|^L \vdash |\Delta_0|^R

\parallel\Gamma\parallel |^L, |\Gamma_0|^L \vdash |\Delta_0|^R

\parallel\Gamma\parallel |^L \vdash |\Gamma_0|^L \lor |\Delta_0|^R, \parallel\Delta\parallel |^R \lor R
$$
Case \textit{unnest}L: The given derivation is of the form

\[
\begin{array}{c}
\vdots \pi \\
\Gamma, (\Gamma_0 \vdash \Delta_0) \vdash \Delta \\
\hline
\Gamma, \Gamma_0 \vdash \Delta_0, \Delta
\end{array}
\]

and we can transform it to

\[
\begin{array}{c}
\vdots \text{IH on } \pi \\
\hline
\|\Gamma\|^L, \|\Gamma_0\|^L \times |\Delta_0|^R \vdash |\Delta|^R \\
\hline
\|\Gamma\|^L, \|\Gamma_0\|^L \vdash |\Delta_0|^R \vdash |\Delta|^R
\end{array}
\]

\[
(\text{weak L/R})^*
\]

\[
\begin{array}{c}
\forall i, \ldots, \|\Gamma_0\|^L \vdash \|\Gamma_0\|^L_i, \ldots \\
\forall i, \ldots, |\Delta_0|^R_i, \ldots \\
\vdots \text{hyp} \\
\hline
\forall i, \ldots, |\Delta_0|^R_i, |\Delta_0|^R \vdash |\Delta|^R \\
\hline
\|\Gamma\|^L, \|\Gamma_0\|^L \vdash |\Delta_0|^R \vdash |\Delta|^R
\end{array}
\]

\[
\text{cut}
\]

4.2 \textit{From LB}il to L-LB}il

The translation of \textit{LB}il into L-\textit{LB}il is not demanding. Essentially, it suffices to annotate the end sequent with the sole label of a singleton label tree and follow the structure of the \textit{LB}il-derivation bottom-up, introducing new labels at \(\supset R\) and \(\prec L\). But again (like in the translation from \textit{LB}il to N-\textit{LB}il), a cut in \textit{LB}il is rendered by a cut in L-LB}il, which is not so perfect, since we should not need cut in L-LB}il derivations.

Given a standard context \(\Gamma\), we write \(x : \Gamma\) for the labelled context obtained by labeling all formulas of \(\Gamma\) with \(x\).

\textbf{THEOREM 3}

If \(\Gamma \vdash \Delta\) is derivable in \textit{LB}il, then \(x : \Gamma \vdash (x) x : \Delta\) is derivable in L-LB}il.

\textbf{PROOF}. By induction on the derivation of \(\Gamma \vdash \Delta\) in \textit{LB}il. We consider two cases.

Case \textit{hyp}: The given derivation

\[
\begin{array}{c}
\vdots \text{hyp} \\
\hline
\Gamma, A \vdash A, \Delta
\end{array}
\]

is sent to

\[
\begin{array}{c}
\vdots \text{hyp} \\
\hline
x : \Gamma, x : A \vdash (x) x : A, x : \Delta
\end{array}
\]

Case \(\supset R\): The given derivation

\[
\begin{array}{c}
\vdots \pi \\
\hline
\Gamma, A \vdash B \\
\Gamma \vdash A \supset B, \Delta
\end{array}
\]

\(\supset R\)
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

is matched with the derivation

\[
\begin{align*}
\vdash \text{IH on } \pi \\
y : \Gamma, y : A \vdash_{(y)} y : B \\
x : \Gamma, y : A \vdash_{(x, y)} y : B \\
x : \Gamma, y : A \vdash_{(x, y)} y : B \\
x : \Gamma \vdash x : A \supset B, x : \Delta
\end{align*}
\]

The translation from L-LBiI to LBiI is best found as a compound translation through N-LBiI (which becomes possible after the next section, where we translate between L-LBiI and N-LBiI). We will not work out the details of such composition here, but it is quite instructive. In particular, it gives an explanation of why it is so difficult to translate labelled derivations into standard derivations in the case of Int. We learn that the natural way uses exclusion. And this is not available in Int.

5 Relating labelled and nested sequent calculi

In this section, we introduce translations first between labelled and nested sequents and then between derivations of L-LBiI and of N-LBiI. Whereas the translations for sequents are mutual inverses, this is not the case at the level of derivations. However, we will identify variants of L-LBiI and of N-LBiI for which also derivations are in a 1–1 correspondence.

5.1 Bijective translation of sequents

The translation of a labelled sequent into a nested sequent follows the idea that we can view any label of the label tree as its root (intuitively, the focus of attention) and produce a nesting structure for a nested sequent by mimicking this rooted tree.

The translation of a labelled sequent wrt. a chosen label from its label tree is defined by recursion on the rooted tree structure by

\[
\begin{align*}
\langle \langle \Gamma \vdash \Delta \rangle \rangle_y &= \Gamma(y) \vdash \Delta(y) \\
\langle \langle \Gamma \vdash G_{\Theta, y} \cap [x, y] \supset G_0 \Delta \rangle \rangle_y &= \Lambda \vdash (\Lambda_0 \vdash \Pi_0), \Pi \\
& \text{where } \Lambda \vdash \Pi = \langle \langle \Gamma[G] \vdash G \Delta[G] \rangle \rangle_y \text{ and } \Lambda_0 \vdash \Pi_0 = \langle \langle \Gamma[G_0] \vdash G_0 \Delta[G_0] \rangle \rangle_y \\
\langle \langle \Gamma \vdash G_{\Theta, y} \cap [y, x] \supset G \Delta \rangle \rangle_y &= \Lambda, (\Lambda_0 \vdash \Pi_0) \vdash \Pi \\
& \text{where } \Lambda \vdash \Pi = \langle \langle \Gamma[G] \vdash G \Delta[G] \rangle \rangle_y \text{ and } \Lambda_0 \vdash \Pi_0 = \langle \langle \Gamma[G_0] \vdash G_0 \Delta[G_0] \rangle \rangle_y
\end{align*}
\]

where \( \Gamma(x) = \{ A \mid x : A \in \Gamma \} \) and \( \Gamma[G] = \{ x : A \mid x \in [G] \text{ and } x : A \in \Gamma \} \).

Although a label tree has generally many decompositions, this translation is well-defined, since all decompositions of the same label tree lead to the same result.

Intuitively, the formulas labelled with \( x \) in the given sequent are kept where they are, whereas those with labels reachable through the labels immediately to the past of \( x \), resp. to the future of \( x \), are arranged into nested sequent members of the antecedent, resp. succedent, of the top-level nested sequent produced.
We define a translation of $\text{N-LBiI}$ sequents to $\text{L-LBiI}$ sequents, by induction on the antecedent and succedent of the given nested sequent, by the function $\llbracket \Gamma, A \vdash \Delta \rrbracket_x$ that follows. This function also takes a label $x$ as argument. The root of the nesting structure of the given nested sequent (i.e., its top level) is sent to label $x$ in the label tree of the labelled sequent.

\[
\llbracket \vdash \rrbracket_x = \vdash_{(x)} \\
\llbracket \vdash A, \Delta \rrbracket_x = \Delta \vdash_G x : A, \Pi \quad \text{where } \Delta \vdash_G \Pi = \llbracket \vdash \rrbracket_x \\
\llbracket \vdash (\Gamma_0 \vdash \Delta_0), \Delta \rrbracket_x = \Delta, \Delta_0 \vdash_G (x, y)_{\Theta x, y \Theta} G_0 \Pi_0, \Pi \\
\quad \text{where } \Delta \vdash_G \Pi = \llbracket \vdash \rrbracket_x \text{ and } \Delta_0 \vdash_G \Pi_0 = \llbracket \Gamma_0 \vdash \Delta_0 \rrbracket_y \text{ and } y \text{ is fresh} \\
\llbracket \Gamma, A \vdash \Delta \rrbracket_x = \Delta, A : x \vdash_G \Pi \quad \text{where } \Delta \vdash_G \Pi = \llbracket \Gamma \vdash \Delta \rrbracket_x \\
\llbracket \Gamma, (\Gamma_0 \vdash \Delta_0) \vdash \Delta \rrbracket_x = \Delta, \Delta_0 \vdash_G (x, y)_{\Theta x, y \Theta} G_0 \Pi_0, \Pi \\
\quad \text{where } \Delta \vdash_G \Pi = \llbracket \vdash \rrbracket_x \text{ and } \Delta_0 \vdash_G \Pi_0 = \llbracket \Gamma_0 \vdash \Delta_0 \rrbracket_y \text{ and } y \text{ is fresh}
\]

Intuitively, any formula in the top-level sequent is labelled by $x$ and remains where it is. Any (nested) sequent in the antecedent, resp. succedent, of the top-level sequent leads to the creation of a new label $y$ immediately to the past, resp. to the future, of $x$. The translated elements of its antecedent, resp. succedent, are placed in the antecedent, resp. succedent, of the sequent in the making.

Note that we have given the mathematical definition by first recursing on the antecedent and then the succedent. In fact, the order is immaterial, one could just as well start with the antecedent or, indeed, remove formulas/nested sequents from the antecedent and succedent in turns, in any order. This commutativity is used extensively in our translation of derivations. Additionally, note that, in the third and fifth clauses of the definition, we need to generate a fresh label, but this is unproblematic, since we identify labelled sequents up to the names of labels.

Now we establish that the maps $\llbracket \cdot \rrbracket_x$ and $\llbracket \cdot \rrbracket_x$ are mutual inverses, and thus establish a bijective correspondence between labelled sequents and nested sequents. Recall that equality of labelled sequents is up to renaming of labels: indeed, whereas translating from nested to labelled sequents and back, we will arrive at exactly the same sequent, starting with a labelled sequent, we might only get a sequent equal to it up to renaming of labels, because of the generation of labels involved in the translation from nested to labelled sequents.

**Theorem 4 (Bijection for sequents)**

1. $\llbracket \cdot \rrbracket_x \llbracket \Gamma \vdash \Delta \rrbracket_x = \Gamma \vdash \Delta$, for any $x$.
2. $\llbracket \cdot \rrbracket_x \llbracket \Gamma \vdash_G \Delta \rrbracket_x = \Gamma \vdash_G \Delta$, for any $x \in |G|$.

**Proof.**

(1) is proved by considering first $\Gamma$ empty and inducting on the structure of $\Delta$, and then proving the result for an arbitrary $\Gamma$, by induction on the structure of $\Gamma$. The following auxiliary property is used: if $\llbracket \Gamma \vdash_G \Delta \rrbracket_x = \Lambda \vdash \Pi$, then $\llbracket \cdot \rrbracket_x \llbracket \Gamma, x : A \vdash_G \Delta \rrbracket_x = \Lambda, A \vdash \Pi$ and $\llbracket \cdot \rrbracket_x \llbracket \Gamma \vdash_G x : A, \Delta \rrbracket_x = \Lambda \vdash, A, G$. This property follows by induction on $G$.

(2) is proved by induction on the number of edges in $G$. One of the step cases (when $G$ is of the form $G_1 \Theta x, y \Theta G_0$) needs the property: if $\llbracket \Gamma \vdash \Delta \rrbracket_x = \Lambda \vdash_G \Pi$ and $\llbracket \Gamma_0 \vdash \Delta_0 \rrbracket_y = \Delta_0 \vdash_G \Pi_0$, then $\llbracket \Gamma \vdash \Delta, (\Gamma_0 \vdash \Delta_0) \rrbracket_y = \Delta, \Delta_0 \vdash_G (x, y)_{\Theta x, y \Theta} G_0 \Pi_0, \Pi$. This property follows by induction on $G$. ☐
5.2 From L-LBiI to N-LBiI and back

The translations of derivations between L-LBiI and N-LBiI are more involved than those of the previous section, but also more illuminating.

**Lemma 2 (Readdressing)**

For any \( z, x \in |G| \), if \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( z \) is derivable in N-LBiI, then so is \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( x \).

**Proof.** By induction on the unique path along \( G \cup G^{-1} \) from \( x \) to \( z \). The base case \( x = z \) is trivial.

We consider one of the two symmetric step cases, namely the one where \( x \vdash_{G} y \). In this case, we have \( G = G' \oplus_{x}(x, y) \oplus_{y} G_{0} \), with the path from \( y \) to \( z \) lying in \( G_{0} \).

The given derivation is

\[ \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \]_\( x \)

The nested sequent \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( x \) can be derived by

\[ \begin{array}{c}
\vdots \\
\Pi \text{ on } \pi \\
\Lambda \vdash (\Lambda \vdash \Pi) \vdash \Pi_{0} \\
\Lambda \vdash (\Lambda \vdash \Pi) \vdash \Pi_{0} \end{array} \]

\[ \text{nestR unnestL} \]

where \( \Lambda \vdash \Pi = \langle \langle \Gamma[G'] \vdash_{G' \Delta[G']} \rangle \rangle \)_\( x \) and \( \Lambda_{0} \vdash \Pi_{0} = \langle \langle \Gamma[G_{0}] \vdash_{G_{0} \Delta[G_{0}]} \rangle \rangle \)_\( y \), so that \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( x \) = \( \Lambda \vdash (\Lambda_{0} \vdash \Pi_{0}), \Pi \) whereas \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( y \) = \( \Lambda_{0}, (\Lambda \vdash \Pi) \vdash \Pi_{0} \).

**Theorem 5**

If \( \Gamma \vdash_{G} \Delta \) is derivable in L-LBiI, then \( \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( x \) is derivable in N-LBiI without cuts for any \( x \in |G| \).

**Proof.** By induction on the derivation of \( \Gamma \vdash_{G} \Delta \) in L-LBiI. We show the prototypical cases.

Case hyp: The given derivation is of the form

\[ \Gamma, x : p \vdash_{G} x : p, \Delta \text{ hyp} \]

By the readdressing lemma, it suffices to derive \( \langle \langle \Gamma, x : p \vdash_{G} x : p, \Delta \rangle \rangle \)_\( x \). The desired derivation is

\[ \Lambda, p \vdash p, \Pi \text{ hyp} \]

where \( \Lambda \vdash \Pi = \langle \langle \Gamma \vdash_{G} \Delta \rangle \rangle \)_\( x \).

Case monotL: The given derivation is of the form

\[ \begin{array}{c}
\vdots \\
\pi \\
\Gamma, x : A, y : A \vdash_{G \oplus_{x}(x, y) \oplus_{y} G_{0}} \Delta \\
\Gamma, x : A \vdash_{G \oplus_{x}(x, y) \oplus_{y} G_{0}} \Delta \end{array} \]

\[ \text{monotL} \]

By readdressing, it suffices to prove \( \langle \langle \Gamma, x : A \vdash_{G \oplus_{x}(x, y) \oplus_{y} G_{0}} \Delta \rangle \rangle \)_\( x \).
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

We construct this derivation:

\[
\begin{align*}
\vdash & 
\text{IH on } \pi, y \\
(\Lambda, A \vdash \Pi), \Lambda_0, A \vdash \Pi_0 & 
\quad \text{nestR} \\
(\Lambda, A \vdash \Pi), A \vdash (\Lambda_0 \vdash \Pi_0) & 
\quad \text{unnestL} \\
\Lambda, A \vdash (\Lambda_0 \vdash \Pi_0), \Pi & 
\quad \text{contrL}
\end{align*}
\]

Here \( \Lambda \vdash \Pi = \langle \Gamma[G] \vdash G \Delta[G] \rangle_x \) and \( \Lambda_0 \vdash \Pi_0 = \langle \Gamma[G_0] \vdash G_0 \Delta[G_0] \rangle_y \), which gives us \( \langle \Gamma, x : A \vdash G_{\oplus(x,y)\oplus_0 G_0} \Delta \rangle_x = \Lambda, A \vdash (\Lambda_0 \vdash \Pi_0), \Pi \) and \( \langle \Gamma, x : A, y : A \vdash G_{\oplus(x,y)\oplus_0 G_0} \Delta \rangle_y = (\Lambda, A \vdash \Pi), \Lambda_0, A \vdash \Pi_0 \).

Case \( \supset R \): The given derivation is of the form

\[
\begin{align*}
\vdash & 
\pi \\
\Gamma, y : A \vdash G_{\oplus(x,y)} & 
\quad \supset R
\end{align*}
\]

We prove \( \langle \Gamma \vdash G x : A \supset B, \Delta \rangle_x \), which we know is enough by readdressing. The derivation is this:

\[
\begin{align*}
\vdash & 
\text{IH on } \pi, y \\
(\Lambda \vdash \Pi), A \vdash B & 
\quad \supset R \\
(\Lambda \vdash \Pi) \vdash A \supset B, \Pi & 
\quad \text{unnestL}
\end{align*}
\]

Here, \( \Lambda \vdash \Pi = \langle \Gamma \vdash G \Delta \rangle_x \), which gives us \( \langle \Gamma \vdash G x : A \supset B, \Delta \rangle_x = \Lambda \vdash A \supset B, \Pi \) and \( \langle \Gamma, y : A \vdash G_{\oplus(x,y)} y : B, \Delta \rangle_y = (\Lambda \vdash \Pi), A \vdash B \).

The other cases are either analogous to the cases shown above (\( \text{monotR} \) and \( \text{\textless L} \)), or simpler, as they only need the use of \( \text{IH} \) and application of the corresponding rule in \( \text{N-LBiI} \). In particular, no case of the translation introduces cuts (recall that readdressing adds no cuts) and, since cut is not primitive in \( \text{L-LBiI} \), we obtain cut-free derivations in \( \text{N-LBiI} \).

The translation from \( \text{N-LBiI} \) to \( \text{L-LBiI} \) is intended as an inverse for the one just seen from \( \text{L-LBiI} \) to \( \text{N-LBiI} \). On sequents, as seen before, it is a true inverse. On derivations, true inversion is achieved in the following subsection, but for variants of the systems \( \text{N-LBiI} \) and \( \text{L-LBiI} \).

**Theorem 6**

If \( \Gamma \vdash \Delta \) is derivable in \( \text{N-LBiI} \), then \( \llbracket \Gamma \vdash \Delta \rrbracket_x \) is derivable in \( \text{L-LBiI} \) for any \( x \).

**Proof.** By induction on the given derivation. We look at some cases. Note that the cases for weakening, contraction and cut pose no special difficulty, since we have shown that the corresponding rules are admissible in \( \text{L-LBiI} \).
Case \textit{nestR}: The given derivation is of the form

\[
\frac{\vdots \pi}{\Gamma, \Gamma_0 \vdash \Delta_0} \quad \text{\textit{nestR}}
\]

We can produce this derivation of the translated sequent:

\[
\frac{\vdots}{(\Pi \text{ on } \pi)[y/x]}
\]

\[
\frac{\Lambda_d[y/x], \Lambda_0 \vdash G_d[y/x] \oplus G_0 \Pi_0, \Pi_d[y/x]}{\Lambda_d[y/x], \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d[y/x]} \quad \text{nodesplitP}
\]

\[
\frac{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d, \Pi_d}{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d} \quad \text{(weakL)*}
\]

\[
\frac{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d}{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0} \quad \text{(monotL)*}
\]

\[
\frac{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0}{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0} \quad \text{(nodesplitF/P)*}
\]

\[
\frac{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d, \Pi_d}{\Lambda_d, \Lambda_0 \vdash G_d \oplus (x,y) \oplus G_0 \Pi_0, \Pi_d} \quad \text{(weakL/R)*}
\]

where \( \Lambda_d \vdash G_d \Pi_d = [\Gamma \vdash \Delta_x]_x, \Lambda_u \vdash G_u \Pi_u = [\vdash \Delta]_x \) and \( \Lambda_0 \vdash G_0 \Pi_0 = [\Gamma_0 \vdash \Delta_0]_y \), and \( \Lambda_d[x, y/x] \) stands for the union of \( \Lambda_d[y/x] \) with the context formed by the \( x \)-labelled formulas of \( \Lambda_d \). Notice that \( x \notin \Pi_d \), which tells us that \( \Pi_d[x/y] = \Pi_d \). The side condition of the topmost application of \textit{nodesplitD} is met because \( G_d \uparrow x \). Note also that particular cases of \textit{nodesplitF/P} allow the addition of new nodes to a label tree.

Case \textit{unnestL}: We are given a derivation in the form

\[
\frac{\vdots \pi}{\Gamma, (\Gamma_0 \vdash \Delta_0) \vdash \Delta} \quad \text{\textit{unnestL}}
\]

We make the derivation

\[
\frac{\vdots \Pi \text{ on } \pi}{\Lambda, \Lambda_0[y/x] \vdash G_0[y/x] \oplus (y, x) \oplus G \Pi_0[y/x], \Pi} \quad \text{nodemergeF}
\]

where \( \Lambda \vdash G \Pi = [\Gamma \vdash \Delta_x]_x \) and \( \Lambda_0 \vdash G_0 \Pi_0 = [\Gamma_0 \vdash \Delta_0]_x \), hence \( [\Gamma_0 \vdash \Delta_0]_y = \Lambda_0[y/x] \vdash G_0[y/x] \Pi_0[y/x] \).

Case \( \supset R \): The given derivation is of the form

\[
\frac{\vdots \pi}{\Gamma, A \vdash B, \Delta \vdash R}
\]
We transform it to

\[
\begin{align*}
\Lambda_d[y/x], y : A & \vdash G_d[y/x] \\
\Lambda_d[x, y/x], y : A & \vdash G_d \oplus (x, u) / y : B, \Pi_d[y/x] \\
\Lambda_d[x, y/x], y : A & \vdash G_d \oplus (x, u) / y : B, \Pi_d \\
\Lambda_d, y : A & \vdash G_d \oplus (x, u) / y : B, \Pi_d \\
\Lambda_d & \vdash G_d \oplus G_u / A \supset B, \Pi_d \\
\Lambda_d & \vdash G_d \oplus G_u / A \supset B, \Pi_d \\
\Lambda_d, u & \vdash G_d \oplus G_u / A \supset B, \Pi_d, \Pi_u \\
\end{align*}
\]

where \( \Lambda_d \vdash G_d \Pi_d = \Gamma \vdash x \) and \( \Lambda_u \vdash G_u \Pi_u = [\vdash \Delta] \). Notice that \( x \not\in |\Pi_d| \), with the effect that \( \Pi_d[y/x] = \Pi_d \). The side condition on the topmost application of \( \text {nodesplitP} \) is satisfied as \( G_d \uparrow x \).

Inspecting the translations of derivations between \( \text{L-LBiI} \) and \( \text{N-LBiI} \), we see some mismatches. The translation from \( \text{L-LBiI} \) to \( \text{N-LBiI} \) uses only primitive rules of \( \text{N-LBiI} \). The translation in the opposite direction uses various admissible rules of \( \text{L-LBiI} \), notably the \( \text{nodemerge} \) and the \( \text{nodesplit} \) rules, thus suggesting \( \text{N-LBiI} \) to be bigger than \( \text{L-LBiI} \), and potentially more challenging for proof search. We will revisit these translations in Section 6 paying special attention to the uses of weakening, contraction and cut, and extracting some consequences.

5.3 Isomorphic labelled and nested sequent calculi

In this subsection, we show how to obtain systems originating from \( \text{L-LBiI} \) and \( \text{N-LBiI} \) in a \( 1\text{-}1 \) correspondence both for sequents and derivations. To this end, one option could be to enlarge the labelled system with primitive \( \text{nodemerge} \) and \( \text{nodesplit} \) rules. However, from a proof search perspective, this is not the best choice. Instead, we consider a nested system smaller than \( \text{N-LBiI} \), which essentially corresponds to the image of the translation from \( \text{L-LBiI} \) to \( \text{N-LBiI} \). To fully achieve isomorphic systems, we need to consider a more bureaucratic version of \( \text{L-LBiI} \) where each sequent has a distinguished label.

5.3.1 The labelled sequent calculus \( \text{L}_0\text{-LBiI} \)

Sequents of \( \text{L}_0\text{-LBiI} \) have the form \( \Gamma \vdash^x \Delta \) where \( x \in G \) and \( x \) is called the \textit{label in focus}. All rules of \( \text{L-LBiI} \) have a counterpart in \( \text{L}_0\text{-LBiI} \), but impose the principal formula to be labelled with the label in focus. Additionally, there are \textit{refocusing rules} for changing the label in focus. The full set of rules is in Figure 5. Note that, in the rules \( \supset R, \prec L, \text{monotL}, \text{monotR} \), the labels in focus in the premise and in the conclusion are different. We could have chosen to formulate these rules having in focus in the premise the same label as in the conclusion. The choice we adopted should fit better with bottom-up proof search, where after (bottom-up) application of a rule we are interested in analysing the new elements of the premise(s).

**Lemma 3** (Readdressing)
For any \( z, x \in |G| \), if \( \Gamma \vdash^z \Delta \) is derivable in \( \text{L}_0\text{-LBiI} \), then so is \( \Gamma \vdash^x \Delta \).
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

Initial rule:
\[ \Gamma, x : p \vdash^x_G x : p, \Delta ~ \text{hyp} \]

Logical rules:
\[
\begin{align*}
\frac{\Gamma \vdash^x_G \Delta}{\Gamma, x : \top \vdash^x_G \Delta} & \quad \top L \\
\frac{\Gamma \vdash^x_G x : \top, \Delta}{\Gamma \vdash^x_G x : \top, \Delta} & \quad \top R \\
\frac{\Gamma, x : A, x : B \vdash^x_G \Delta}{\Gamma, x : A \land B \vdash^x_G \Delta} & \quad \land L \\
\frac{\Gamma \vdash^x_G x : A, \Delta \quad \Gamma \vdash^x_G x : B, \Delta}{\Gamma \vdash^x_G x : A \land B, \Delta} & \quad \land R \\
\frac{\Gamma, x : \bot \vdash^x_G \Delta}{\Gamma \vdash^x_G x : \bot, \Delta} & \quad \bot L \\
\frac{\Gamma \vdash^x_G x : \bot, \Delta}{\Gamma \vdash^x_G x : \bot, \Delta} & \quad \bot R \\
\frac{\Gamma, x : A \lor B \vdash^x_G \Delta \quad \Gamma, x : B \vdash^x_G \Delta}{\Gamma, x : A \lor B \vdash^x_G \Delta} & \quad \lor L \\
\frac{\Gamma \vdash^x_G x : A \lor B, \Delta}{\Gamma \vdash^x_G x : A \land B, \Delta} & \quad \lor R \\
\frac{\Gamma, y : A \vdash^y_{(y,x) \Theta G} y : B, \Delta}{\Gamma, x : A \setminus B \vdash^x_G \Delta} & \quad \setminus L \\
\frac{\Gamma \vdash^x_G x : A, \Delta \quad \Gamma, x : B \vdash^x_G x : A \setminus B, \Delta}{\Gamma \vdash^x_G x : A \setminus B, \Delta} & \quad \setminus R \\
\intertext{Monotonicity rules:}
\frac{x G y \quad \Gamma, x : A, y : A \vdash^y_G \Delta}{\Gamma, x : A \vdash^x_G \Delta} & \quad \text{monot} L \\
\frac{y G x \quad \Gamma \vdash^y_G y : A, x : A, \Delta}{\Gamma \vdash^x_G x : A, \Delta} & \quad \text{monot} R \\
\begin{align*}
\text{proviso: } A \text{ atomic or an implication} \\
\text{proviso: } A \text{ atomic or an exclusion}
\end{align*}
\]

Refocusing rules:
\[
\begin{align*}
\frac{x G y \quad \Gamma \vdash^y_G \Delta}{\Gamma \vdash^x_G \Delta} & \quad \text{refoc} P \\
\frac{y G x \quad \Gamma \vdash^y_G \Delta}{\Gamma \vdash^x_G \Delta} & \quad \text{refoc} F
\end{align*}
\]

**Figure 5** Labelled sequent calculus \(L_0\text{-LBiI}\)

**Proof.** By induction on the unique path along \(G \cup G^{-1}\) from \(x\) to \(z\). The step cases make use of the refocusing rules.

**Proposition 8**
\(\Gamma \vdash_G \Delta\) is derivable in \(\text{LBiI}\) iff \(\Gamma \vdash^x_G \Delta\) is derivable in \(L_0\text{-LBiI}\) for any \(x \in G\).

**Proof.** From the left to the right, the proof follows by induction on \(\text{LBiI}\)-derivations, with the help of the previous readdressing lemma (each step in \(\text{LBiI}\) is mapped to the corresponding step in \(L_0\text{-LBiI}\) followed by a sequence of refocusing steps, possibly empty).

From the right to the left, the proof follows easily by induction on \(L_0\text{-LBiI}\)-derivations (the translation only needs to drop the label in focus at each sequent and eliminate refocusing steps, as the premise and the conclusion become coincident at these steps).
Initial rule:

\[ \Gamma, p \vdash p, \Delta \quad \text{hyp} \]

Logical rules:

- \[ \frac{\Gamma \vdash \Delta}{\Gamma, \top \vdash \Delta} \quad \top L \]
- \[ \frac{\Gamma \vdash \top, \Delta}{\Gamma \vdash \Gamma \vdash \top, \Delta} \quad \top R \]
- \[ \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \land B \vdash \Delta} \quad \land L \]
- \[ \frac{\Gamma \vdash A \land B, \Delta}{\Gamma \vdash A, \Delta} \quad \land R \]
- \[ \frac{\Gamma, A \supset B \vdash A, \Delta}{\Gamma, B \vdash A \supset B, \Delta} \quad \supset L \]
- \[ \frac{\Gamma \vdash A \supset B, \Delta}{\Gamma \vdash \Gamma \vdash A \vdash B, \Delta} \quad \supset R \]
- \[ \frac{\Gamma, A \vdash B, \Delta}{\Gamma, A \vdash A \vdash B, \Delta} \quad \land L \]
- \[ \frac{\Gamma \vdash A \land B, \Delta}{\Gamma \vdash A, \Delta} \quad \land R \]
- \[ \frac{\Gamma, A \vdash B, \Delta}{\Gamma, A \vdash A \vdash B, \Delta} \quad \supset L \]
- \[ \frac{\Gamma \vdash A \supset B, \Delta}{\Gamma \vdash \Gamma \vdash A \vdash B, \Delta} \quad \supset R \]

Monotonicity rules:

\[ \frac{\Gamma, A \vdash \Delta, \Pi, A \vdash \Lambda}{\Gamma, A \vdash \Delta, \Pi, A \vdash \Lambda} \quad \text{monot} L \]
\[ \frac{\Pi \vdash A, \Lambda, (\Gamma \vdash A, \Delta)}{(\Pi \vdash A, \Lambda, \Gamma \vdash A, \Delta)} \quad \text{monot} R \]

proviso: A atomic or an implication

Refocusing rules:

\[ \frac{\Gamma \vdash \Delta, \Pi \vdash \Lambda}{\Gamma \vdash \Delta, (\Pi \vdash \Lambda)} \quad \text{refoc} P \]
\[ \frac{\Pi \vdash \Lambda, (\Gamma \vdash \Delta)}{(\Pi \vdash \Lambda, \Gamma \vdash \Delta)} \quad \text{refoc} F \]

**Figure 6** Nested sequent calculus $N_0$-$LBi$ isomorphic to $L_0$-$LBi$

Observe that the two translations involved in the proof of the previous proposition are essentially inverse, but not true inverses. Starting at an $L$-$LBi$-derivation, the translation into $L_0$-$LBi$ and back gives the same derivation. However, starting at an $L_0$-$LBi$ derivation, the translation into $L$-$LBi$ ‘forgets’ the refocusing steps; when mapping back to $L_0$-$LBi$, the sequences of refocusing steps produced by the Readdressing Lemma might be different.

5.3.2 The nested sequent calculus $N_0$-$LBi$

Now we introduce the system $N_0$-$LBi$, which is the isomorphic counterpart of $L_0$-$LBi$ in ‘nested style’. The inference rules of $N_0$-$LBi$ are given in Figure 6. Contrary to $N$-$LBi$ (but similarly to $L_0$-$LBi$), this system does not have primitive structural rules or cut rule. Also, there are no primitive rules of nest or unnest. Instead, $N_0$-$LBi$ has monotonicity and refocus rules (similarly to $L_0$-$LBi$). Note that there is a 1–1 match between rules of $N_0$-$LBi$ and of $L_0$-$LBi$ which, indeed, induces a bijective correspondence for derivations, as we will see in the next sub-subsection.

**Proposition 9**

If $\Gamma \vdash \Delta$ is derivable in $N_0$-$LBi$, then $\Gamma \vdash \Delta$ is derivable in $N$-$LBi$ without weakening and cut and with contraction constrained to atoms and implications on the left and to atoms and exclusions on the right.
PROOF. By easy induction on \( N_0-LBiI \) derivations. In relation to the rules of \( N-LBiI \) having no directly matching rule in \( N-LBiI \), note that: \( \exists R \) and \( \diamond L \) are derivable using the respective rule of \( N-LBiI \), followed by an unnest step; the refocusing rules are derivable by a nest step followed by an unnest step; the monotonicity rules are derivable by a nest step, followed by an unnest step, followed by a contraction step (on the distinguished formula \( A \), which is an atom or an implication in the case of \( monotL \), and is an atom or an exclusion in the case of \( monotR \)).

The fact that all sequents derivable in \( N-LBiI \) are also derivable in \( N_0-LBiI \) is harder to establish. We will do this later (Proposition 11) via the labelled systems, with the help of the isomorphism between the \( N_0-LBiI \) and \( L_0-LBiI \).

5.3.3 The isomorphism

The translation of labelled sequents into nested sequents is readily adapted to translate labelled sequents with a label in focus. Again, the translation uses a decomposition of the given label tree, but the choice of a particular decomposition is irrelevant, all decompositions yield the same result.

\[
\langle \Gamma \vdash^x \Delta \rangle = \Gamma(x) \vdash \Delta(x)
\]
\[
\langle \Gamma \vdash^x G_{\beta(x)}(\Gamma_0) \Delta \rangle = \Lambda \vdash (\Lambda_0 \vdash \Pi_0), \Pi
\]

where \( \Lambda \vdash \Pi = \langle \Gamma[G] \vdash y \Delta[G] \rangle \) and \( \Lambda_0 \vdash \Pi_0 = \langle \Gamma[G_0] \vdash y \Delta[G_0] \rangle \)

\[
\langle \Gamma \vdash^x G_{\beta(y)}(\Gamma_0) \Delta \rangle = \Lambda, (\Lambda_0 \vdash \Pi_0) \vdash \Pi
\]

where \( \Lambda \vdash \Pi = \langle \Gamma[G] \vdash y \Delta[G] \rangle \) and \( \Lambda_0 \vdash \Pi_0 = \langle \Gamma[G_0] \vdash y \Delta[G_0] \rangle \)

The translation of nested sequents into \( L_0-LBiI \)-sequents is also an immediate adaptation of the translation into \( L-LBiI \)-sequents (the label corresponding to the additional argument becomes the label in focus):

\[
\llbracket \vdash \rrbracket_x = \llbracket \vdash \rrbracket
\]
\[
\llbracket \vdash A, \Delta \rrbracket_x = \Lambda \vdash^x G x : A, \Pi
\]

where \( \Lambda \vdash^x \Pi = \llbracket \vdash \rrbracket_x \)

\[
\llbracket (\Gamma_0 \vdash \Delta_0), \Delta \rrbracket_x = \Lambda, \Lambda_0 \vdash^x G_{\beta(x)}(\Gamma_0) \Pi_0, \Pi
\]

where \( \Lambda \vdash^x \Pi = \llbracket \vdash \rrbracket_x \) and \( \Lambda_0 \vdash^y G_0 \Pi_0 = \llbracket \Gamma_0 \vdash \Delta_0 \rrbracket_y \)

\[
\llbracket \Gamma, \Delta \rrbracket_x = \Lambda, x : A \vdash^x G \Pi
\]

where \( \Lambda \vdash^x \Pi = \llbracket \Gamma \vdash \rrbracket_x \)

\[
\llbracket \Gamma_0, (\Gamma_0 \vdash \Delta_0) \vdash \Delta \rrbracket_x = \Lambda, \Lambda_0 \vdash^x G_{\beta(y)}(\Gamma_0) \Pi_0, \Pi
\]

where \( \Lambda \vdash^x \Pi = \llbracket \Gamma \vdash \rrbracket_x \) and \( \Lambda_0 \vdash^y G_0 \Pi_0 = \llbracket \Gamma_0 \vdash \Delta_0 \rrbracket_y \)

The bijection between sequents of \( L_0-LBiI \) and nested sequents is a consequence of the next proposition, which is proved analogously to Theorem 4.

PROPOSITION 10 (Bijection for sequents)

1. \( \langle \llbracket \Gamma \vdash \Delta \rrbracket_x \rangle = \Gamma \vdash \Delta \) for any label \( x \).
2. \( \llbracket \langle \Gamma \vdash^x \Delta \rangle \rrbracket_x = \Gamma \vdash^x \Delta \).

Now we observe the bijection at the level of derivations.
THEOREM 7 (Isomorphism)

1. If $\Gamma \vdash^G \Delta$ is derivable in $\mathbf{L_0-LBiI}$, then $\langle\langle \Gamma \vdash^G \Delta \rangle\rangle$ is derivable in $\mathbf{N_0-LBiI}$.
2. If $\Gamma \vdash \Delta$ is derivable in $\mathbf{N_0-LBiI}$, then $\llbracket \Gamma \vdash^G \Delta \rrbracket$ is derivable in $\mathbf{L_0-LBiI}$ for any $x$.
3. Furthermore, the translations of derivations from (1) and (2) are inverse maps, and so establish bijections between the derivations of a sequent in $\mathbf{L_0-LBiI}$ and the derivations of the corresponding sequent in $\mathbf{N_0-LBiI}$.

PROOF. The proofs of (1) and (2) are routine inductions on the given derivations: axioms in one system are rendered by axioms in the other system and, at the inductive steps, we just need to use the IH followed by the corresponding rule in the other system. This clearly induces recursive translations of derivations that are inverse (as the translations of the sequents are inverse). Let us have a look at the mapping of some inferences for (2) (where one also sees the mapping of the corresponding cases of (1), thanks to invertibility of the translations for sequents).

Case $\supset R$:

\[
\frac{\Gamma \vdash \Delta, A \vdash B}{\Gamma \vdash A \supset B, \Delta} \supset R \quad \mapsto \quad \frac{\Lambda, y : A \vdash^G \gamma x : B, \Pi}{\Lambda \vdash^G \gamma x : A \supset B, \Pi} \supset R
\]

where $\Lambda \vdash^G \Pi = \llbracket \Gamma \vdash^G \Delta \rrbracket$ (hence, $\Lambda \vdash^G x : A \supset B, \Pi = \llbracket \Gamma \vdash^G \Delta, \Lambda \vdash^G B, \Pi \rrbracket$) and $\Lambda, y : A \vdash^G \gamma (x, y) \vdash^G y : B, \Pi = \llbracket (\Gamma \vdash^G \Delta, A \vdash B) \rrbracket$.

Case monotL:

\[
\frac{\Gamma, A \vdash^G \Delta, \Pi_0, \Pi_0}{\Gamma, A \vdash^G \Delta, \Gamma_0 \vdash^G \Delta_0} \text{ monotL} \quad \mapsto \quad \frac{\Lambda, \Lambda_0, x : A \vdash^G \gamma (x, y) \Pi_0, \Pi_0}{\Lambda, \Lambda_0, x : A \vdash^G \gamma (x, y) \Pi_0, \Pi_0} \text{ monotL}
\]

where $\Lambda \vdash^G \Pi = \llbracket \Gamma \vdash^G \Delta \rrbracket$ and $\Lambda_0 \vdash^G_0 \Pi_0 = \llbracket \Gamma_0 \vdash^G \Delta_0 \rrbracket$.

Case refocF:

\[
\frac{\Gamma \vdash^G \Delta, (\Gamma_0 \vdash^G \Delta_0)}{(\Gamma \vdash^G \Delta), \Gamma_0 \vdash^G \Delta_0} \text{ refocF} \quad \mapsto \quad \frac{\Lambda, \Lambda_0 \vdash^G \gamma (x, y) \Pi_0, \Pi_0}{\Lambda, \Lambda_0 \vdash^G \gamma (x, y) \Pi_0, \Pi_0} \text{ refocF}
\]

where $\Lambda \vdash^G \Pi = \llbracket \Gamma \vdash^G \Delta \rrbracket$ and $\Lambda_0 \vdash^G_0 \Pi_0 = \llbracket \Gamma_0 \vdash^G \Delta_0 \rrbracket$.

\[\Box\]

PROPOSITION 11

Any sequent derivable in $\mathbf{N-LBiI}$ is also derivable in $\mathbf{N_0-LBiI}$.

PROOF. Let $\Gamma \vdash \Delta$ be a derivable sequent of $\mathbf{N-LBiI}$ and, for a fixed label $x$, let $\llbracket \Gamma \vdash^G \Delta \rrbracket = \Delta \vdash^G \Pi$. By Theorem 6, $\Lambda \vdash^G \Pi$ is derivable in $\mathbf{L-LBiI}$ and so, by Proposition 8, $\Lambda \vdash^G \Pi$ is derivable in $\mathbf{L_0-LBiI}$ and, by the previous theorem, $\llbracket \Lambda \vdash^G \Pi \rrbracket$ is derivable in $\mathbf{N_0-LBiI}$. Finally, observe that $\llbracket \Lambda \vdash^G \Pi \rrbracket = \Gamma \vdash^G \Delta$, which follows from the facts: i) $\llbracket \Lambda \vdash^G \Pi \rrbracket = \llbracket \Lambda \vdash^G \Pi \rrbracket_\chi$ (easy induction on the label tree $G$); and ii) $\llbracket \Lambda \vdash^G \Pi \rrbracket_\chi = \Gamma \vdash^G \Delta$ (obtained from $\llbracket \Gamma \vdash^G \Delta \rrbracket = \Delta \vdash^G \Pi$, and Theorem 4).

\[\Box\]

5.4 Relating to deep inference nested sequent calculus for $\mathbf{BiInt}$

A deep inference version of the $\mathbf{BiInt}$ nested sequent calculus $\mathbf{LBiInt}$ called $\mathbf{DBiInt}$ is studied in [29, 30]. As characteristic of deep inference, $\mathbf{DBiInt}$ allows inference rules to be applied to any sequent of a structure of nested sequents, without the need of transforming first the latter, in order to bring the desired sequent to the ‘top-level’. The formulation of $\mathbf{DBiInt}$ uses a notion of context with a
hole, capable of identifying a position and the surrounding context in a nested sequent, and inference rules apply to the (nested) sequent in the hole position. Contexts may have negative or positive polarity (accounting respectively for the occurrence of the hole in the antecedent or succedent of a nested sequent), and this allows the formulation of some inference rules by reference only to the formula being introduced, rather than to the nested sequent containing it. We refer to Figure 5.1 of [30] for the inference rules of $DBiInt$. Compared to $LBiInt_1$, both the $nest$ and the $unnest$ rules are dropped and considered instead are propagation rules and versions of the $\supset R$ and $\prec L$ rules incorporating some form of nesting and unnesting. In Chapter 5 of [30], it is shown that $DBiInt$ is sound and complete wrt. $LBiInt_1$, by means of proof transformations and admissibility of various rules in both systems. Since system $N-LBiI$ essentially corresponds to system $LBiInt_1$ (the precise differences are pointed out in Subsection 2.3), it is no surprise that $DBiInt$ can be related to $N-LBiI$ and $L-LBiI$ and also to their isomorphic variants $N_0-LBiI$ and $L_0-LBiI$.

We recall below the two left propagation rules of $DBiInt$ ($\triangleright_L^1$ and $\triangleright_L^2$) using our nested sequent notation and assuming that they are applied at the top level. Note that $\triangleright_L^1$ is applicable in a negative context and $\triangleright_L^2$ in a neutral context. (The two right propagation rules $\triangleright_R^1$ and $\triangleright_R^2$ are symmetric to the left propagation rules.)

$$
\frac{\Gamma, A, (A, \Pi \vdash \Lambda) \vdash \Delta}{\Gamma, (A, \Pi \vdash \Lambda) \vdash \Delta} \quad \triangleright_L^1
$$

$$
\frac{\Gamma, A \vdash \Delta, (A, \Pi \vdash \Lambda)}{\Gamma, A \vdash \Delta} \quad \triangleright_L^2
$$

These two inferences are easily shown admissible in $N_0-LBiI$ by a $monotL$ step, followed by a $refocF$ step in the case of $\triangleright_L^1$; and by a $refocP$ step followed by a $monotL$ step in the case of $\triangleright_L^2$. (The fact that $monotL$ and $monotR$ are admissible in $N_0-LBiI$ for arbitrary formulas is established in Theorem 8.) Note also that the $\supset R$ and $\prec L$ rules of $DBiInt$ have the same spirit as the respective rules of $N_0-LBiI$. Indeed, these inferences of $DBiInt$ at the top level are easily shown derivable in $N_0-LBiI$ by applying $refocF$, resp. $refocP$, followed by $\supset R$, resp. $\prec L$, and a contraction step. (Admissibility of contraction in $N_0-LBiI$ is also shown in Theorem 8.) In fact, the need of contraction in $N_0-LBiI$ to derive rules of $DBiInt$ applies to all logical rules because in $DBiInt$ the introduced formula is always contracted. Since the refocussing rules of $N_0-LBiI$ allow to bring to the top level any nested sequent inside a structure of nested sequents, it is not hard to embed derivations of $DBiInt$ into derivations of $N_0-LBiI$ with the help of the admissible rules of $N_0-LBiI$ established in Theorem 8. Of course, as the labelled sequent calculus $L_0-LBiI$ is isomorphic to $N_0-LBiI$, we can also embed $DBiInt$ into $L_0-LBiI$. But it is worthwhile noticing that the deep inference aspect of $DBiInt$ is actually matched better by the initial labelled system $L-LBiI$, since $L-LBiI$ does not impose that inference rules apply only if the label of the introduced formula is in focus. Under the translation of nested sequents into labelled sequents in Subsection 5.1, steps of left propagation ($\triangleright_L^1$ and $\triangleright_L^2$) map into $monotL$ steps of $L-LBiI$ (admissible for arbitrary formulas–Proposition 5). Likewise, steps of right propagation ($\triangleright_R^1$ and $\triangleright_R^2$) both map into $monotR$ steps.

6 Applications of the translations

Analysing the targets of the various translations in the previous two sections, we obtain some immediate applications. Our analysis focuses on the use of weakening, contraction and cut in the translations. The general idea is to transfer properties of $L-LBiI$ to the other systems. In particular, for $LBiI$, we find complete classes of cuts (recall that cut is not fully eliminable in $LBiI$) and eliminability results for weakening and contraction.
Theorem 8

Weakening, contraction, cut, as well as the axiom and monotonicity for arbitrary formulas, are admissible in \( L_0 \)-LBiI and \( N_0 \)-LBiI.

Proof. For \( L_0 \)-LBiI, the results follow from the corresponding ones for \( L \)-LBiI (Propositions 1, 2, and 5, and Theorem 1) with the help of Proposition 8. For example, admissibility of cut holds because: if \( \Gamma \vdash G \Delta, x : A \) and \( \Gamma, x : A \vdash G \Delta \) are derivable in \( L_0 \)-LBiI, then, by Proposition 8, \( \Gamma \vdash_G \Delta, x : A \) and \( \Gamma, x : A \vdash_G \Delta \) are derivable in \( L \)-LBiI; so, by Theorem 1, \( \Gamma \vdash_G \Delta \) is derivable in \( L \)-LBiI and, using Proposition 8 again, we obtain derivability of \( \Gamma \vdash_G \Delta \) in \( L_0 \)-LBiI. In view of the results for \( L_0 \)-LBiI, the isomorphism between \( L_0 \)-LBiI and \( N_0 \)-LBiI (Theorem 7) guarantees the results for \( N_0 \)-LBiI.

Next, we obtain full eliminability of weakening and cut as well as partial eliminability of contraction for \( N \)-LBiI and reprove cut elimination for the base nested system LBiInt1 of Goré et al. [17].

Theorem 9

1. In \( N \)-LBiI, weakening and cut are eliminable and contraction can be constrained to atoms and implications on the left as well as to atoms and exclusions on the right.
2. Cut is eliminable in LBiInt1.

Proof.

(1) could be proved by inspecting the target of the translation of \( L \)-LBiI-derivations into \( N \)-LBiI. Instead, we argue in terms of the system \( N_0 \)-LBiI. By Proposition 11, any sequent derivable in \( N \)-LBiI is also derivable in \( N_0 \)-LBiI. As, by Proposition 9, any sequent derivable in \( N_0 \)-LBiI is derivable in \( N \)-LBiI without weakening and cut, and as uses of contraction can be limited to atoms and implications on the left and to atoms and exclusions on the right, we are done.

(2) is a consequence of (1) and the fact that cut-free derivations of \( N \)-LBiI can be easily mapped into cut-free LBiInt1. Recall that, excluding the cut rule, these two systems differ in that LBiInt1 has primitive rules for weakening and contraction and in that the rules \( \supset L, \supset L, \land \) and \( \lor R \) of \( N \)-LBiI have some implicit contraction. However, these rules of \( N \)-LBiI are easily derivable in LBiInt1, using the corresponding rules together with contraction and weakening.

In [17], cut elimination for LBiInt1 is proved by means of syntactical transformations on derivations. Here, we do not get an internal cut elimination procedure, although we could consider mimicking the cut elimination procedure for \( L \)-LBiI in \( N \)-LBiI (on this, see also Subsection 7.3). The works [17] and [28] do not pay much attention to the questions of eliminability of weakening and of contraction in the nested systems they consider. In particular, note that, in the system LBiInt1 of [17], contraction is a primitive rule not eliminable and, in the proof search calculus LBiInt2 of op. cit., where the structural rules are absorbed into the logical rules, each rule duplicates the main formula in the transition from the conclusion to the premises.

Now, we turn to LBiI. Inspecting the translation from \( N \)-LBiI to LBiI (Theorem 2), we find that the cut rule of LBiI is used only for the translation of cuts and unnesting steps. Let us call unnest cuts the cuts of LBiI of one of the following two forms:

\[
\frac{\Gamma, \Gamma_0 \vdash \bigwedge \Gamma \land \Delta, \Delta_0}{\Gamma, \Gamma_0, \bigwedge \Gamma \land \Delta \vdash \Delta, \Delta_0} \quad \text{unnestcut } L
\]

\[
\frac{\Gamma, \Gamma_0 \vdash \bigwedge \Gamma \lor \Delta, \Delta_0}{\Gamma, \Gamma_0, \bigwedge \Gamma \lor \Delta \vdash \Delta, \Delta_0} \quad \text{unnestcut } R
\]
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

\[
\frac{\Gamma, \Gamma_0, \forall \Delta \vdash \Delta, \Delta_0}{\Gamma, \Gamma_0, \Gamma \vdash \Delta, \Delta_0} \text{ unnestL}
\quad \frac{\Gamma, \Gamma_0 \vdash \forall \Delta, \Delta, \Delta_0}{\Gamma, \Gamma_0 \vdash \forall \Delta, \Delta, \Delta_0} \text{ unnestR}
\]

**FIGURE 7** Unnest rules for standard-style sequents

Here \( \forall \) and \( \forall \) are defined similarly to \( |\Gamma|_L \) and \( |\Delta|_R \) from Section 4.1. These two special cases of cut are the ones used in the translations of the unnest rules (see the last case of the proof of Theorem 2). As the first premise of unnestcutL and the second premise of unnestcutR are derivable, a more practical idea is to consider single premise rules, as in Figure 7.

Since cut is eliminable in \( N-LBiI \) (Theorem 9), we have that unnest cuts are complete for \( LBiI \) or, in other words, the system obtained from \( LBiI \) by replacing the cut rule by the unnestL and unnestR rules is complete. This result can be sharpened to account also for eliminability of weakening and contraction, if we take \( N_0-LBiI \) instead of \( N-LBiI \) as the source calculus.

**THEOREM 10**

Let \( LBiI_0 \) be the system obtained from \( LBiI \) by adding the unnest rules in Figure 7. This system is sound and complete for \( BiInt \) and enjoys eliminability of weakening, contraction and cut.

**PROOF.** Clearly, any sequent derivable in \( LBiI_0 \) is derivable in \( N-LBiI \) (where the unnest rules are rendered in \( N-LBiI \) by cuts). This proves soundness. Completeness is obvious, since \( LBiI_0 \) contains all rules of \( LBiI \). For the eliminability results, it suffices to show that, if \( \Gamma \vdash \Delta \) is derivable in \( N_0-LBiI \), then \( ||\Gamma||_L \vdash ||\Delta||_R \) is derivable in \( LBiI_0 \) without weakening, contraction and cut. (Note that: (i) any sequent derivable in \( LBiI_0 \) is derivable in \( N_0-LBiI \), which follows from what was said above and Proposition 11; (ii) for nest-free contexts \( \Gamma \) and \( \Delta \), \( ||\Gamma||_L = \Gamma \) and \( ||\Delta||_R = \Delta \).) The proof follows by induction on the height of the derivation of \( \Gamma \vdash \Delta \) and is just a recast of the proof of Theorem 2. In this proof, we also rely on the observation that weakening in \( N_0-LBiI \) can be done preserving derivation height, which can be easily proved in the isomorphic system \( L_0-LBiI \). We illustrate how to map one half of the rules where there is no directly matching rule in \( LBiI_0 \) (each of the rules in the other half is ‘dual’ to one of the rules below). Derivations of the top sequent in the right-hand sides of the clauses below are guaranteed to exist by height-preserving weakening in \( N_0-LBiI \) and the IH. For the unnest inferences below, note the identities \( ||\Gamma||_L = |\Gamma|_L \) and \( ||\Delta||_R = |\Delta|_R \), for any nested context \( \Gamma \).

**Case \( \supset R \):**

\[
\frac{(\Gamma \vdash \Delta), A \vdash B}{\Gamma \vdash \Delta, A \supset B} \quad \frac{\quad \frac{||\Gamma||_L, |\Gamma|_L \vdash |\Delta|_R, A \vdash B}{||\Gamma||_L \vdash ||\Delta||_R, A \supset B}}{\Gamma \vdash \Delta, A \supset B} \quad \text{unnestL} \]

**Case \( \text{monotL} \):**

\[
\frac{(\Gamma, A \vdash \Delta), \Pi, A \vdash \Lambda}{\Gamma, A \vdash \Delta, (\Pi \vdash \Lambda)} \quad \frac{\frac{||\Gamma||_L, |\Gamma, A|_L \vdash |\Delta|_R, |\Pi||_L, A \vdash |\Delta||_R}{||\Gamma||_L \vdash ||\Delta||_R, |\Pi||_L \vdash |\Delta||_R}}{\Gamma, A \vdash \Delta, (\Pi \vdash \Lambda)} \quad \text{unnestL}
\]

\( (\land L, \lor R)^* \)
A proof-theoretic study of bi-intuitionistic propositional sequent calculus

Figure 8 Variants of $\prec L$ and $\sup R$ in $\textbf{LBiI}_0^*$ which absorb unnest cuts

Case $\text{refocP}$:

\[(\Gamma \vdash \Delta), \Pi \vdash \Delta \quad \text{refocP} \quad \frac{\|\Gamma\|^L, \|\Delta\|^L \prec \|\Pi\|^L \vdash \|\Delta\|^R}{\frac{\|\Gamma\|^L \prec \|\Delta\|^R, \|\Pi\|^L \vdash \|\Delta\|^R}{\|\Gamma\|^L, \|\Delta\|^L \prec \|\Pi\|^L \vdash \|\Delta\|^R}} \quad \frac{\|\Delta\|^L \vdash \|\Delta\|^R, \|\Pi\|^L \vdash \|\Delta\|^R}{\sup R \quad \text{unnestL}}
\]

We end this section by presenting a refined version of the previous theorem. Indeed, inspecting the translation in the proof above, it is easy to see that unnest cuts are used only after $\sup R$ or $\prec L$ and make very limited use of contraction.

Theorem 11

Let $\textbf{LBiI}_0^*$ be the system obtained from $\textbf{LBiI}$ by replacing $\sup R$ and $\prec L$ by the rules in Figure 8.\(^{6}\) This system is sound and complete for $\text{BiInt}$ and enjoys eliminability of weakening, contraction and cut.

Proof. For soundness, we can, e.g., observe that all rules of $\textbf{LBiI}_0^*$ are derivable in $\textbf{LBiI}_0$. For completeness, and the eliminability results, we can argue as in the proof of the previous theorem, and, in particular, embed $\textbf{N}_0$-$\textbf{LBiI}$ into $\textbf{LBiI}_0^*$: for each of the three cases shown in the proof of Theorem 10, the required instances of $\sup R$-$\text{unnestL}$ take $\Gamma_0$ to be empty—the cases $\sup R$ and $\text{refocP}$—or to be the singleton context $A$—the case $\text{monotL}$—and in this case $A$ is either an atom or an implication (recall the proviso at $\text{monotL}$).

The system of standard sequents $\textbf{LBiI}_0^*$ is of interest for bottom-up proof search, since it enjoys eliminability of contraction and cut and the implicit use of contraction and cut is limited. The implicit use of contraction is confined to the rules $\sup L$, $\prec R$, $\sup R$-$\text{unnestL}$ and $\prec L$-$\text{unnestR}$, and only atoms, negative implications, and positive exclusions need to be contracted. As to cuts, only unnest cuts are needed and their implicit use is confined to the rules $\sup R$-$\text{unnestL}$ and $\prec L$-$\text{unnestR}$. Interpreting this in terms of bottom-up proof search in the original system $\textbf{LBiI}$, it means that cuts in unnested form are sufficient, and are only needed immediately before applying inferences $\sup R$ or $\prec L$. Such combination of unnest cuts and $\sup R/\prec L$ can be seen as a way to achieve invertibility in proof search, which in general is lost at isolated applications of $\sup R/\prec L$. Of course, to turn these ideas into a proof search procedure with standard-style sequents, we would still have to account for termination, e.g., via some loop-detection mechanism as used in the proof search procedures for the nested sequent systems $\textbf{LBiInt}_1$ or $\textbf{DBiInt}$ [17, 29, 30], or for the labelled system $L$ [27].

---

\(^{6}\) Note the similarity of the rules with the $\prec L$ and $\sup R$ rules of $\textbf{N}_0$-$\textbf{LBiI}$ in Figure 6.
7 Conclusion

7.1 Concluding remarks

Although it is easy to obtain complete proof systems for BiInt as extensions of proof systems for Int, such extensions raise new issues. In particular, handling of contraction and cut poses new problems: elimination of the former is made more difficult because, in addition to monotonicity of truth—which is shared with Int—, BiInt directly uses the contrapositive, i.e., antimonicity of falsehood; full elimination of cuts is not even possible with standard-style sequent calculus.

In this paper, we offer a proof-theoretic study of BiInt, relying on translations between standard-style, nested and labelled sequent calculi for BiInt, which refines the study initiated in [28]. A clear gain obtained from this study is the ability of transferring meta-theoretic properties between these sequent calculi. As a general strategy, we developed meta-theory in the labelled format (including new syntactical proofs of admissibility of contraction and cut), and read it off in the other formats. Proving meta-theoretic properties for the nested format and for the standard format (recall in this case the incompleteness without cuts) typically raises the need for extra machinery to provide some notion of ‘context’: the labelled format can be seen as offering an implementation of such a notion. Nonetheless, in various circumstances, it proved useful to abstract away from the ‘implementation details’ of the labelled system, and look for intuitions or formulate conjectures relatively to the nested or standard systems. Another product of our study is the identification of the systems $N_0-$LBiI and $L_0-$LBiI as isomorphic formulations for BiInt in the nested and labelled formats, respectively. Although a close relationship was hinted already in our study [28], quite some refinements were needed to promote the translations into bijections at the level of derivations. Note that our choice to work with trees in the labelled format (see also the discussion below on related work on labelled systems) was essential to obtain the bijection between labelled and nested sequents.

BiInt is an interesting logic not only from the viewpoint of raising challenging mathematical questions, but also because the exclusion operation can help in the study of different proof formats. It is remarkable that, in the case of BiInt, translation of labelled sequent calculus into standard-style sequent calculus (and into nested sequent calculus) becomes much easier than for Int. In [34], Reed and Pfenning observe that relating labelled intuitionistic derivations to standard unlabelled ones is ‘a surprisingly difficult question’.

7.2 Related work

Labelled systems for BiInt and Int The system L-LBiI lies between the calculi L and $L^*$ of [27]. Similarly to $L^*$ (but not L), L-LBiI is a calculus of finite Kripke trees rather than general Kripke structures, so that reasoning can be done in terms of immediate accessibility → (edge) rather than the induced accessibility (path) relation $\leq \rightarrow^*$ L-LBiI, and there are no reflexivity and transitivity rules for the immediate accessibility relation. Similarly to L-LBiI, both L and $L^*$ have explicit monotonicity rules to propagate truth/falsity to adjacent labels. In the algorithmic system $L^*$, monotonicity is constrained as in L-LBiI (explicitly it is allowed only for atoms and implicitly it is also allowed on implications and exclusions) and, in L, there is an explicit monotonicity rule with no constraints on the shape of the main formula.

The intuitionistic fragment of L-LBiI is close to a variant of the labelled system for intuitionistic propositional logic G3I considered by Negri in [26], but still there are important differences. The main difference lies in the treatment of the accessibility relation. Negri considers explicit order rules to guarantee reflexivity and transitivity of the accessibility relation whereas we use label trees. In particular, this impacts on node merging principles, which in our case need to guarantee that the label...
tree structure is preserved (compare our Propositions 4 and 6 with Lemma 3 of [11]). Additionally, G3I allows monotonicity only implicitly, either in combination with the initial rule or in combination with □L or with ◐R. The fact that we have explicit monotonicity rules raises new issues in the meta-
theory: this shows up already in proving invertibility for □L and ◐R rules (Corollary 1), but also in proving admissibility of the cut rule, where new ‘principal’ cases arise.

**Relating sequent calculi** There are several studies into the interrelationships of different styles of sequent calculi for non-classical logics (e.g., [5] surveys work on relating hypersequent and display sequent calculi, but also considers other extended sequent calculi), but none of them applies specifically to BiInt. Neither seem they to pay much attention to explaining the extended calculi compared to standard-style sequent calculus (cf. Section 6, which explains unnest rules in terms of ordinary cuts). The observation that labelled and nested sequent calculi are closely related is not new (see, e.g., p. 42 of [3]). However, by the time of publication of [28], the closest detailed study of such relationship we knew of was that developed in [36], which only applies to hypersequents and intuitionistic logic. More recent studies relating nested and labelled systems appeared in [12, 18].

In [12], Fitting observes that prefixed modal tableaus (which correspond to labelled systems imposing a tree structure on the accessibility relation) and systems of modal nested sequents are ‘notational variants’. Fitting details this in the context of classical modal logics (K and richer logics, including quantified logics), by giving translations of sequents in both directions, and arguing about the correspondence of some inference rules under such translations. In the contemporaneous work [18], Goré and Ramanayake study the relationship between systems for classical modal logics of labelled tree sequents, tree hypersequents and nested sequents. Inverse translations between labelled tree sequents and tree hypersequents are given, and the calculi induced by each of these translations are identified; in a similar way, the relationship between systems of tree hypersequents and of nested sequents is worked out. Goré and Ramanayake specialize the ideas to the case of provability logic GL and use the translations to transfer proof-theoretic results (namely, cut elimination) between systems of labelled tree sequents and of tree hypersequents for GL. This kind of application is shared with the work we initiated in [28] and further developed in Section 6 of this paper, which, in particular, allowed us to identify complete classes of cuts for the standard-style sequent calculus for BiInt. Even if, in a sense, the ideas behind the translations between the various kind of sequents and the results obtained in [12, 18] have a lot in common with our translations and results of Section 5 (specially in the case of [18]), the details are quite different. In particular, because we deal with BiInt logic, our translations need to be more general, as our labelled system deals with an accessibility relation not necessarily ‘treelike’ in the oriented sense (recall creation of a new node and edge toward the past in the exclusion-left rule) and the nested sequents are two-sided.

### 7.3 Future work

This paper identifies unnest cuts as a complete class of cuts for standard-style sequent calculus for BiInt. These cuts are not analytic, i.e., the cut formula is not necessarily a subformula of the conclusion and the subformula property is not guaranteed. Using semantical methods, the works [21, 20] show independently that analytic cuts are complete for BiInt. We would like to investigate whether completeness of analytic cuts for LBil (or some of its variants) could be established by proof-theoretic methods and, in particular, understand how to express unnest cuts with analytic cuts.

The isomorphic systems L0-LBil and N0-LBil and the system LBil0 of ordinary sequents have reduced use of (implicit) contraction. This contrasts with the approach in [17], where the proof search system LBiInt2 always keeps a copy of the main formula in the premise(s). We would like to investigate whether contraction can be avoided in the monotonicity rules of L0-LBil and N0-LBil,
as well as in the rules $\supset R$-unnestL and $\leftarrow L$-unnestR of $\mathbf{LBiI}_0$. Ideally, we would also like to dispense with the contraction of the main formula in the rules $\supset L$ and $\leftarrow R$, and attain contraction-free systems for $\mathbf{BiInt}$, similarly to Dyckhoff’s system [10] for $\mathbf{Int}$.

Our translations between standard-style, nested and labelled sequent calculi for $\mathbf{BiInt}$ provide a framework not only for comparing proof search, but also for relating cut elimination procedures, and to study computational interpretations of $\mathbf{BiInt}$. One specific question is to investigate the relationship between cut elimination for nested sequent calculus, as described in Goré et al. [17], and cut elimination for the labelled system, as we develop in Section 3.2. Another question is to devise ways of performing (partial) cut elimination for the standard-style system (recall that unnest cuts are not eliminable), and then also compare such procedures with the cut elimination procedures for the other two formats. A helpful tool for performing this comparison should be term assignment. Once proof terms are considered, a natural question to raise is about their computational interpretations and relationship with Crolard’s computational interpretation of bi-intuitionistic logic [7], based on a natural deduction system with multiple conclusions.

**Funding**

This research was financed by Fundação para a Ciência e a Tecnologia (FCT) through project UID/MAT/00013/2013, by ERDF through the Estonian Centre of Excellence in Computer Science (EXCS), by the Estonian Science Foundation under grant no. 6940 and by the COST action CA15123 EUTYPES.

**Acknowledgements**

We are grateful to our anonymous referees for their very helpful comments. We are also grateful to Linda Postniece for discussions.

**References**


A proof-theoretic study of bi-intuitionistic propositional sequent calculus


Received 7 December 2015