# *The Geometry of Non-Distributive Logics*

GREG RESTALL
Philosophy Department
The University of Melbourne
restall@unimelb.edu.au

FRANCESCO PAOLI
Dipartimento di Scienze Pedagogiche e Filosofiche,
Università di Cagliari
paoli@unica.it

Version of April 28, 2005

*Abstract*: In this paper we introduce a new natural deduction system for the logic of lattices, and a number of extensions of lattice logic with different negation connectives. We provide the class of natural deduction proofs with both a standard inductive definition and a global graph-theoretical criterion for correctness, and we show how normalisation in this system corresponds to cut elimination in the sequent calculus for lattice logic. This natural deduction system is inspired both by Shoesmith and Smiley's multiple conclusion systems for classical logic and Girard's proofnets for linear logic.

In this paper we present a new kind of natural deduction system for lattice logic and extensions of lattice logic with negation. This natural deduction system has a natural symmetry. The rules for lattice disjunction are exactly the reverse of the rules for lattice conjunction. In this way, the natural deduction system shares some features with Girard's proofnets for multiplicative linear logic [12]. Our proofs differ from proofnets, however, in that they are *oriented* graphs in which the flow of information from premise to conclusion in inference is directly represented in the graph structure. Nonetheless, our proof graphs can be given a global graph-theoretical criterion for correcteness, just as can be done for Girard's proofnets. Readers will also note that our proof structures bear more than a passing resemblence to Shoesmith and Smiley's multiple conclusion proofs for classical logic, which are oriented graphs allowing both divergent and convergent branching, just like ours. Our lattice logic proofs differ from Shoesmith and Smiley's multiple conclusion proofs in a number of important respects, not the least of which is that lattice logic proofs have a single premise and a single output. Given all of these ancestors, our account is distinctive in a number of different ways:

1 Lattice logic (also called *finite sum-product logic* [5]) is an extremely simple, beautiful, and symmetric constructive logic, characterized by a straightforward duality property. Such a duality, involving disjunction and conjunction, is completely obscured in the usual natural deduction rules for these connectives, whereas our calculus allows to emphasize it properly. In particular, the disturbing asymmetry between the straightforward $\wedge$I rule and its complex $\vee$E counterpart is completely removed.

2 Finding a manageable and intuitively appealing calculus of proofnets for the additive connectives of linear logic is still an open problem, despite earlier suggestions, for example those of Girard [13] and Tortora De Falco [20, 21], and a promising new perspective recently suggested by Hughes and van Glabbeek [15]. Our approach is distinctive in that we build up a calculus for the additives alone, rather than trying to integrate them into an already existing calculus for the multiplicatives. In this way we isolate the distinctive logical features of non-distributive lattice connectives and treat them on their own merits.

3 Our calculus and its suggested expansions by a negation connective provide additional motivation for nondistributive logics; in particular, involutive lattice logic and orthologic receive hereby an intuitively perspicuous formulation.

## 1 Lattice logic

Lattices are familiar algebraic structures [3, 9, 14] featuring the operations meet and join (conjunction and disjunction). A partial order $\leqslant$ is induced on any lattice by setting $x \leqslant y$ to hold if and only of $x \wedge y = x$ (iff $x \vee y = y$). From a *logical* perspective, this partial order looks like *logical consequence*. It is well known that familiar logical techniques may be used to generate the class of valid lattice inequalities. One straightforward technique is the sequent calculus. Here, we present a familiar sequent calculus for the class of lattice inequations (c.f. [5]): a sequent $A \Rightarrow B$ represents an inequality holding in all lattices.

Definition 1 $\mathcal{L}_0$ is a propositional language containing denumerably many variables $p_0, p_1, \ldots$ and the connectives $\wedge, \vee$. Formulae are constructed in the usual manner. The letters $A, B, \ldots$ are used as metavariables for formulae of $\mathcal{L}_0$. The Gentzen system for lattice logic, GLL, has the following postulates given in Figure 1.

$$A \Rightarrow A \ (\mathrm{Id}) \qquad\qquad \frac{A \Rightarrow B \quad B \Rightarrow C}{A \Rightarrow C}\ (\mathrm{Cut})$$

$$\frac{A \Rightarrow C}{A \wedge B \Rightarrow C}\ (\wedge L_1) \qquad \frac{B \Rightarrow C}{A \wedge B \Rightarrow C}\ (\wedge L_2) \qquad \frac{A \Rightarrow B \quad A \Rightarrow C}{A \Rightarrow B \wedge C}\ (\wedge R)$$

$$\frac{A \Rightarrow C \quad B \Rightarrow C}{A \vee B \Rightarrow C}\ (\vee L) \qquad \frac{C \Rightarrow A}{C \Rightarrow A \vee B}\ (\vee R_1) \qquad \frac{C \Rightarrow B}{C \Rightarrow A \vee B}\ (\vee R_2)$$

Figure 1: The rules for the Gentzen system GLL.

In GLL, the rule (Cut) is redundant. Any sequent provable using (Cut) is also provable without it.

Theorem 2 GLL *is cut-free.*

Proof The proof proceeds via the standard double induction on grade and rank. Figure 2 contains examples of the relevant transformations, in which cuts are pushed upward in a proof over each of the different kinds of inference. □

Notice that in some of the transformations, such as when a cut is pushed over an identity axiom (T1), or when a cut is pushed over a conjunction formula (T4), the resulting proof is smaller. In other cases, such as when a cut is pushed over a disjunction introduction on the left (T3), the result is a larger proof, in which a subproof of the original proof is used *twice* in the new proof.

It is well known that a duality principle holds for lattices: if $x \leqslant y$ is a valid lattice inequality, then so is $y' \leqslant x'$, where $x'$ and $y'$ are respectively obtained from $x$ and $y$ by interchanging meets with joins and vice versa. This property can be conveniently expressed in GLL.

Definition 3 If $A$ is a formula of $\mathcal{L}_0$, its dual $A^d$ is inductively defined as follows:

$$p^d = p; \quad (A \wedge B)^d = A^d \vee B^d; \quad (A \vee B)^d = A^d \wedge B^d.$$

Left (Id)

$$\dfrac{A \Rightarrow A \qquad \overset{\vdots}{A \Rightarrow C}}{A \Rightarrow C} \quad \overset{\text{T1}}{\rightsquigarrow} \quad \overset{\vdots}{A \Rightarrow C}$$

Left ($\wedge L_1$)

$$\dfrac{\dfrac{\overset{\vdots}{A \Rightarrow C}}{A \wedge B \Rightarrow C} \qquad \overset{\vdots}{C \Rightarrow D}}{A \wedge B \Rightarrow D} \quad \overset{\text{T2}}{\rightsquigarrow} \quad \dfrac{\dfrac{\overset{\vdots}{A \Rightarrow C} \qquad \overset{\vdots}{C \Rightarrow D}}{A \Rightarrow D}}{A \wedge B \Rightarrow D}$$

Left ($\vee L$)

$$\dfrac{\dfrac{\overset{\vdots}{A \Rightarrow C} \qquad \overset{\vdots}{B \Rightarrow C}}{A \vee B \Rightarrow C} \qquad \overset{\vdots}{C \Rightarrow D}}{A \vee B \Rightarrow D} \quad \overset{\text{T3}}{\rightsquigarrow} \quad \dfrac{\dfrac{\overset{\vdots}{A \Rightarrow C} \qquad \overset{\vdots}{C \Rightarrow D}}{A \Rightarrow D} \qquad \dfrac{\overset{\vdots}{B \Rightarrow C} \qquad \overset{\vdots}{C \Rightarrow D}}{B \Rightarrow D}}{A \vee B \Rightarrow D}$$

($\wedge R$, $\wedge L_1$)

$$\dfrac{\dfrac{\overset{\vdots}{C \Rightarrow A} \qquad \overset{\vdots}{C \Rightarrow B}}{C \Rightarrow A \wedge B} \qquad \dfrac{\overset{\vdots}{A \Rightarrow D}}{A \wedge B \Rightarrow D}}{C \Rightarrow D} \quad \overset{\text{T4}}{\rightsquigarrow} \quad \dfrac{\overset{\vdots}{C \Rightarrow A} \qquad \overset{\vdots}{A \Rightarrow D}}{C \Rightarrow D}$$

Figure 2: Transformations: pushing cuts over inferences in GLL.

THEOREM 4 *If* $\vdash_{\text{GLL}} A \Rightarrow B$*, then* $\vdash_{\text{GLL}} B^{d} \Rightarrow A^{d}$.

## 2 PROOF GRAPHS

In this section we shall consider a family of labelled graphs, and single out from within this class those that correspond to correct proofs in GLL. Some of the definitions appearing in the present section are borrowed from either Carbone [4], or Oliveira and Queiroz [17]; for an illustration of basic graph-theoretical notions the reader is referred to standard texts [10].

A proof graph (to be defined below) is a connected and oriented graph in which the vertices are labelled by formulas, and in which the edges (arrows) indicate the flow of information from premise to conclusion. Of course, not every arrangement of formulas and arrows between them will count as a proof graph. The transitions in proofs must take some very particular forms: we will call these basic transitions *links*, as they are chained together to produce the transition from premise to conclusion.

A vertex in a proof graph cannot be the conclusion of more than one link, and nor can it be the premise of more than one link. (If we wish to reuse information we may use the *contraction* and *expansion* links to duplicate the formula in question.)

DEFINITION 5 A *link* in a proof graph is an oriented subgraph taking one of the following eight configurations:

| | | | | |
|---|---|---|---|---|
| *Simple* | $(\wedge E_1)$ | $(\wedge E_2)$ | $(\vee I_1)$ | $(\vee I_2)$ |
| *Links* | $A \wedge B \longrightarrow A$ | $A \wedge B \longrightarrow B$ | $A \longrightarrow A \vee B$ | $B \longrightarrow A \vee B$ |
| | $(\wedge I)$ | $(\vee E)$ | (Contraction) | (Expansion) |
| *Branching Links* | $\begin{tikzcd} A \arrow[dr] & \\ & A \wedge B \\ B \arrow[ur] & \end{tikzcd}$ | $\begin{tikzcd} & A \\ A \vee B \arrow[ur] \arrow[dr] & \\ & B \end{tikzcd}$ | $\begin{tikzcd} A \arrow[dr] & \\ & A \\ A \arrow[ur] & \end{tikzcd}$ | $\begin{tikzcd} & A \\ A \arrow[ur] \arrow[dr] & \\ & A \end{tikzcd}$ |

For each link, the *inputs* of that link are the labelled vertices with no arrows oriented *toward* them, and the *outputs* of a link are those with no arrows oriented *away* from them. (In the table above the inputs of a link are on the left and its outputs are on the right.) Simple links have one input and one output. Branching links either have two inputs and one output or one input and two outputs.

Now we can introduce proof graphs.

DEFINITION 6 A *proof graph* is a *connected*, *oriented* graph $\pi$ whose edges are represented by arrows and whose vertices are labelled by formulae of $\mathcal{L}_0$. A proof graph must satisfy the following three conditions:

PG1 Each edge in a proof graph belongs to one and only one link.

PG2 Each vertex in a graph is the input of at most one link.

PG3 Each vertex in a proof graph is the output of at most one link.

A vertex in a proof graph that is not the output of any link is said to be an *input* of the proof graph. A vertex in a proof graph that is not the input of any link is said to be an *output* of the proof graph. If a proof graph has no edges at all, then it is said to be an *axiom*. (Since proof graphs are connected, an edgeless graph consists of a single vertex.)

In what follows, we shall often identify the vertices of our graphs with the formulas that label them. The letters $u, v, a, b, \ldots$ will be sometimes used to refer to elements of $V(\pi)$, the set of vertices of $\pi$, while such expressions as $u : A$ (*vertex labelling statements*) will mean that the vertex $u$ is labelled by the formula $A$. $E(\pi)$ is the set of edges of $\pi$.

The different kinds of links appearing in proof graphs may be classified in different ways. We have already seen one distinction, on the basis of their geometry. Links with one input and one output are *simple*, and the other links are *branching*. Branching links, in turn, can be divided into *focussing* links ($\wedge$I and contraction links) and *defocussing* links ($\vee$E and expansion links), where focussing links have two inputs and one output, and defocussing links have one input and two outputs. In virtue of their role governing the connectives, the $\wedge$I, $\wedge$E, $\vee$I, and $\vee$E links are said to be *logical*, while the expansion and contraction links are *structural*. Finally, for reasons that will become clearer later, the $\wedge$E, $\wedge$I, and expansion links are called *conjunctive*, and the remaining links are *disjunctive*. The following table summarizes this taxonomy.

|  | *Simple* | *Branch.* | *Foc.* | *Defoc.* | *Log.* | *Struct.* | *Conj.* | *Disj.* |
|---|---|---|---|---|---|---|---|---|
| $\wedge$E | × |  |  |  | × |  | × |  |
| $\wedge$I |  | × | × |  | × |  | × |  |
| $\vee$E |  | × |  | × | × |  |  | × |
| $\vee$I | × |  |  |  | × |  |  | × |
| Exp |  | × |  | × |  | × |  | × |
| Cont |  | × | × |  |  | × | × |  |

Definition 7 A link in a proof graph is said to be *initial* if and only if each input vertex of that link is an input vertex for the proof graph. Similarly, a link is said to be *final* if and only if each output vertex of that link is an output vertex for the proof graph.

$$\begin{tikzcd}
 & A \arrow[dr] & \\
A \vee B \arrow[ur] \arrow[dr] & & A \wedge B \\
 & B \arrow[ur] &
\end{tikzcd}$$

Figure 3: The diamond. (This is a proof graph, but not an ll-graph)

Some proof graphs correspond to correct proofs of gll, but others do not. Proof graphs are too liberal to count as a faithful picture of inference in lattice logic. For example, the proof graph in Figure 3 corresponds to an unsound deduction of $A \wedge B$ from the hypothesis $A \vee B$. The ($\vee$E) link in this graph (leading from the input $A \vee B$) is initial, and the ($\wedge$I) link (leading to the ouptut $A \wedge B$) is final. So, we must select from within the class of all proof graphs those graphs which represent derivations in gll. We shall do so by means of the next definition.

Definition 8 ll-graphs are connected and oriented graphs whose edges are represented by arrows and whose vertices are labelled by formulae of $\mathcal{L}_0$, according to the following inductive clauses:

Axiom Clause: A graph made up by a single vertex, labelled by A, is an ll-graph with input A and output A.

$\wedge$E Clause: If $\pi$ is an ll-graph with input B (we will write this '$B \longrightarrow \boxed{\pi}$'), then $A \wedge B \longrightarrow B \longrightarrow \boxed{\pi}$ and $B \wedge A \longrightarrow B \longrightarrow \boxed{\pi}$ are ll-graphs with inputs $A \wedge B$ and $B \wedge A$, respectively, and with the same outputs as $\pi$.

$\vee$I Clause: If $\pi$ is an LL-graph with output $A$ (notation: ‘$\boxed{\pi}\longrightarrow A$’), then $\boxed{\pi}\longrightarrow A \longrightarrow A \vee B$ and $\boxed{\pi}\longrightarrow A \longrightarrow B \vee A$ are LL-graphs with outputs $A \vee B$ and $B \vee A$, respectively, and with the same inputs as $\pi$.

$\wedge$I Clause: If $\pi$ is an LL-graph with an input $A$ and an output $B$ (in our notation: ‘$A \mathrel{-}\boxed{\pi}\longrightarrow B$’) and $\sigma$ is an LL-graph with the same input $A$ and an output $C$ (‘$A \mathrel{-}\boxed{\sigma}\longrightarrow C$’), then

$$\begin{tikzcd}
& A \arrow[r, dash] & \boxed{\pi} \arrow[r] & B \arrow[dr] & \\
A \arrow[ur] \arrow[dr] & & & & B \wedge C \\
& A \arrow[r, dash] & \boxed{\sigma} \arrow[r] & C \arrow[ur] &
\end{tikzcd}$$

is an LL-graph with input $A$ and output $B \wedge C$.

$\vee$E Clause: If $\pi$ is an LL-graph with an input $A$ and an output $C$ and $\sigma$ is an LL-graph with the input $B$ and the same output $C$, then

$$\begin{tikzcd}
& A \arrow[r, dash] & \boxed{\pi} \arrow[r] & C \arrow[dr] & \\
A \vee B \arrow[ur] \arrow[dr] & & & & C \\
& B \arrow[r, dash] & \boxed{\sigma} \arrow[r] & C \arrow[ur] &
\end{tikzcd}$$

is an LL-graph with input $A \vee B$ and output $C$.

(Cut Clause): If $\pi$ is an LL-graph with output $A$ and $\sigma$ is an LL-graph with input $A$, then $\boxed{\pi}\longrightarrow A \longrightarrow\boxed{\sigma}$ is an LL-graph with the same input as $\pi$ and the same output as $\sigma$.

Each LL-graph is a proof graph, since they are built using links drawn from the set of Definition 5, and the only links added in the construction of an LL-graph are added to the inputs or outputs of smaller graphs (so each vertex is the input and output of at most one link, and each edge appears in exactly one link). The converse, however, does not necessarily hold: the DIAMOND, for example, is a proof graph but not a LL-graph. (Notice that every LL-graph containing a $\wedge$I link also has an expansion link. The DIAMOND has a $\wedge$I link but no expansion link, so it is not an LL-graph. Notice too that LL-graphs constructed with the $\wedge$I and $\vee$E clauses are somewhat reminiscent of Girard's *additive boxes* [12].)

Example 9 Here are two examples of LL-graphs. The first is a simple inference from $A$ to $A \wedge (A \vee B)$. Notice the initial defocussing link duplicating the input $A$. The top copy is used to infer $A \vee B$, and this, with the bottom copy, is used to infer the output conjunction.

$$\begin{tikzcd}
 & A \arrow[r] & A \vee B \arrow[dr] & \\
A \arrow[ur] \arrow[dr] & & & A \wedge (A \vee B) \\
 & A \arrow[urr] & &
\end{tikzcd}$$

In the next LL-graph, only conjunction rules are used. The input $A \wedge (B \wedge C)$ is duplicated *twice* to produce three copies, each of which are decomposed into a constituent conjunct, which are then reconstituted into a conjunction, identical to the input except for a reassociation.

$$\begin{tikzcd}
 & & A \wedge (B \wedge C) \arrow[rrr] & & & A \arrow[dr] & & \\
 & A \wedge (B \wedge C) \arrow[ur] \arrow[dr] & & & & & A \wedge B \arrow[ddr] & \\
 & & A \wedge (B \wedge C) \arrow[r] & B \wedge C \arrow[r] & B \arrow[urr] & & & \\
A \wedge (B \wedge C) \arrow[uur] \arrow[dr] & & & & & & & (A \wedge B) \wedge C \\
 & A \wedge (B \wedge C) \arrow[rrr] & & & B \wedge C \arrow[rr] & & C \arrow[ur] &
\end{tikzcd}$$

The LL-graphs we have seen all have a single input and a single output. This is not a coincidence.

THEOREM 10 *Every* LL-*graph has exactly one input vertex and exactly one output vertex.*

PROOF A straightforward induction on the construction of LL-graphs. Each axiom graph has one input and one output (the same labelled vertex.) Each rule transforms single-input, single-output graphs into larger, single-input, single-output graphs. □

And so, LL-graphs mirror in structure the sequents of GLL. They take a single input and a single output. We will see in the next section that this mirroring is much closer than that.

## 3 COMPLETENESS, CUT ELIMINATION, DUALITY

Having introduced the notion of LL-graph, we aim for three results:

1 First and foremost, we need to show that LL-graphs really correspond to sequent derivations in GLL. We need to prove a completeness theorem.

2 Moreover, it would seem desirable to prove that the Cut clause in the construction of proofs is redundant, viz. that by resorting to it we do not actually increase our stock of inferences underwritten by LL-graphs. In particular, we will examine the analogue of cut elimination in LL-graphs, to produce *normal* LL-graphs.

3 Finally, we will reproduce in our new context the duality result of Theorem 4, exposing the inherent duality of LL-graphs.

The completeness result is hardly more than a triviality, since the construction clauses for LL-graphs explicitly simulate the inference rules of GLL.

THEOREM 11 $A \Rightarrow B$ *is provable in* GLL *iff there exists an* LL*-graph* $\pi$ *with input* $A$ *and output* $B$.

Cut elimination is somewhat more interesting from the geometrical viewpoint. Does the cutformula of an application of cut in a derivation in GLL play any special role in the corresponding LL-graph $\pi$? It is readily observed that, in many cases, the cutformula labels a cutvertex of $\pi$. In such cases, cut elimination corresponds to a decrease in the number of blocks in $\pi$. Unfortunately, this does not hold in general; we shall see that sometimes a derivation with cuts and its normalized counterpart are graph-theoretically indiscriminable from each other.

THEOREM 12 *If* $\pi$ *is an* LL*-graph with input* $A$ *and output* $B$*, there exists an* LL*-graph* $\sigma$ *with input* $A$ *and output* $B$ *which is built without using the cut clause.*

PROOF The proof is analogous to that of Theorem 2. We transform proofs constructed using the Cut rule into those constructed without it. We confine ourselves to some examples for each relevant reduction pattern. The cases corresponding to transformations T1 and T2 in Theorem 2 require no transformation here, for the LL-graphs

$$A \;—\boxed{\pi}\!\longrightarrow C \qquad A \wedge B \longrightarrow A \;—\boxed{\pi}\!\longrightarrow C \;—\boxed{\sigma}\!\longrightarrow D$$

represent both the GLL proof before transformation and the resulting GLL proof after transformation. The cases corresponding to transformations T3 and T4 in the same proof are matched by the following transformations (compare what happens with the CCAD cut links in Tortora De Falco's additive proofnets [21]):

$$\begin{tikzcd}
 & A \arrow[r, "\boxed{\pi}" description] & C \arrow[dr] & & & & & A \arrow[r, "\boxed{\pi}" description] & C \arrow[r, "\boxed{\tau}" description] & D \arrow[dr] & \\
A \vee B \arrow[ur] \arrow[dr] & & & C \arrow[r, "\boxed{\tau}" description] & D & \underset{\rightsquigarrow}{\mathrm{T}3'} & A \vee B \arrow[ur] \arrow[dr] & & & & D \\
 & B \arrow[r, "\boxed{\sigma}" description] & C \arrow[ur] & & & & & B \arrow[r, "\boxed{\sigma}" description] & C \arrow[r, "\boxed{\tau}" description] & D \arrow[ur] &
\end{tikzcd}$$

$$\begin{tikzcd}
 & C \arrow[r, "\boxed{\pi}" description] & A \arrow[dr] & & & & & & & \\
C \arrow[ur] \arrow[dr] & & & A \wedge B \arrow[r] & A \arrow[r, "\boxed{\tau}" description] & D & \underset{\rightsquigarrow}{\mathrm{T}4'} & C \arrow[r, "\boxed{\pi}" description] & A \arrow[r, "\boxed{\tau}" description] & D \\
 & C \arrow[r, "\boxed{\sigma}" description] & B \arrow[ur] & & & & & & &
\end{tikzcd}$$

The transformed LL-graphs still contain cuts (in the graph after T3$'$ we have two cuts on C, and in the proof after T4$'$ we have a cut on A), but as in the cut elimination proof for GLL, the resulting proofs are simpler, and the cuts disappear in the inductive proof. □

EXAMPLE 13 We owe the reader an example of a single LL-graph that corresponds to different sequent derivations in GLL, only one of which is cut-free. Here is one:

$$\begin{tikzcd}
 & & A \arrow[r] & A \vee B \arrow[dr] & \\
A \wedge B \arrow[r] & A \arrow[ur] \arrow[dr] & & & A \wedge (A \vee B) \\
 & & A \arrow[urr] & &
\end{tikzcd}$$

This LL-graph corresponds to these two derivations, the first of which includes a cut, and the second of which is cut-free:

$$\dfrac{\dfrac{A \Rightarrow A}{A \wedge B \Rightarrow A}\,(\wedge L_1) \qquad \dfrac{A \Rightarrow A \quad \dfrac{A \Rightarrow A}{A \Rightarrow A \vee B}\,(\vee R_1)}{A \Rightarrow A \wedge (A \vee B)}\,(\wedge R)}{A \wedge B \Rightarrow A \wedge (A \vee B)}\,(\text{Cut}) \qquad \dfrac{\dfrac{A \Rightarrow A \quad \dfrac{A \Rightarrow A}{A \Rightarrow A \vee B}\,(\vee R_1)}{A \Rightarrow A \wedge (A \vee B)}\,(\wedge R)}{A \wedge B \Rightarrow A \wedge (A \vee B)}\,(\wedge L_1)$$

Lastly, we show an especially appealing version of the duality theorem. For a start, we need to define what we mean by the dual of a LL-graph.

DEFINITION 14 If $\pi$ is an LL-graph, its dual $\pi^d$ is a graph obtained from $\pi$ by reversing all arrows and by replacing each formula A occurring therein by its dual $A^d$.

THEOREM 15 *If $\pi$ is an LL-graph with input A and output B, its dual $\pi^d$ is an LL-graph with input $B^d$ and output $A^d$.*

The proof is an induction on the construction of $\pi$ as an LL-graph. We leave it to the reader.

## 4 A Correctness Criterion

Proofnets for the multiplicative fragment of linear logic can be characterized through an inductive definition which resembles our Definition 8 in that its atomic steps closely match the rules of the corresponding sequent calculus. One of the outstanding virtues of the calculus of proofnets, however, is the availability of a purely graph-theoretical criterion according to which logically correct proof graphs turn out to be exactly those graphs which obey a simple and elegant condition [8]. This condition, unlike the traditional definition of a natural deduction proof (and our definition of an LL-graph) is not an inductively defined property, but rather, a global property of the structure of the graph. In this section, we shall show that something similar can be done also for LL-graphs. In order to do so, we need more definitions.

Definition 16 Let $\pi$ be a proof graph. A *switching* of $\pi$ is a spanning subgraph $\pi'$ of $\pi$ obtained from it by omitting one edge for each structural link in $\pi$.

Let $\pi$ be a proof graph. A *branchingpart* of $\pi$ is a subgraph $\pi'$ of $\pi$ with a single initial defocussing link and a single final focussing link.

A proof graph $\pi$ has the *switchingproperty* iff each one of its switchings is a tree.

A proof graph $\pi$ has the *branchingproperty* iff, for any branching parts $\sigma$ and $\tau$ of $\pi$, either $E(\sigma) \subseteq E(\tau)$ or $E(\tau) \subseteq E(\sigma)$ or $E(\sigma) \cap E(\tau) = \emptyset$.

A proof graph $\pi$ with a single input and a single output is an LL-*proofnet* iff it has both the switching and the branching property.

Notice that this definition implies that an LL-proofnet with at least one edge has a unique inital link. It also seems apposite that the *branchingproperty* in this definition is reminiscent of Tortora de Falco's *nesting condition* [21].

Our aim in this section, of course, is to show that a proof graph is an LL-graph iff it is a LL-proofnet. We start with the easy direction of the biconditional.

Theorem 17 *Every* LL-*graph* $\pi$ *is an* LL-*proofnet.*

Proof Induction on the construction of $\pi$. First of all, recall (Theorem 10) that every LL-graph has a single input and a single output. If $\pi$ was obtained by the axiom clause, then it has no branching part (whence it has the branching property vacuously) and a single switching, which is a tree. If $\pi = A \wedge B \longrightarrow$

$B \longrightarrow \boxed{\gamma}$ was obtained by the $\wedge$E clause from $\gamma$, then by the induction hypothesis every switching of $\gamma$ is a tree. Therefore, grafting $A \wedge B$ onto $B$ does not spoil acyclicity and connectivity. Moreover, such a move neither affects the branching parts in $\gamma$ nor increases their number, so $\pi$ has the branching property simply by the induction hypothesis. For the $\vee$I clause we argue similarly. If $\pi$

$$\begin{tikzcd}
 & A \arrow[r, dash] & \boxed{\gamma'} \arrow[r] & B \arrow[dr] & \\
A \arrow[ur] \arrow[dr] & & & & B \wedge C \\
 & A \arrow[r, dash] & \boxed{\gamma''} \arrow[r] & C \arrow[ur] &
\end{tikzcd}$$

was obtained by the $\wedge$I clause from $\gamma'$ and $\gamma''$, then by the induction hypothesis every switching $\sigma'$ of $\gamma'$, as well as every switching $\sigma''$ of $\gamma''$, is a tree. Now, consider the graph $\kappa$:

$$\begin{tikzcd}
A \arrow[r, dash] & \boxed{\sigma'} \arrow[r] & B \arrow[dr] & \\
 & & & B \wedge C \\
A \arrow[r, dash] & \boxed{\sigma''} \arrow[r] & C \arrow[ur] &
\end{tikzcd}$$

where $\sigma'$ and $\sigma''$ are switchings of $\gamma'$ and $\gamma''$, respectively. Since $\sigma'$ and $\sigma''$ are acyclic and disjoint, $\kappa$ contains no cycle. Since $\sigma'$ and $\sigma''$ are connected, $\kappa$ is such. Then $\kappa'$ and $\kappa''$

$$\begin{tikzcd}
 & A \arrow[r, dash] & \boxed{\sigma'} \arrow[r] & B \arrow[dr] & \\
A \arrow[ur] & & & & B \wedge C \\
 & A \arrow[r, dash] & \boxed{\sigma''} \arrow[r] & C \arrow[ur] &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & A \arrow[r, dash] & \boxed{\sigma'} \arrow[r] & B \arrow[dr] & \\
A \arrow[dr] & & & & B \wedge C \\
 & A \arrow[r, dash] & \boxed{\sigma''} \arrow[r] & C \arrow[ur] &
\end{tikzcd}$$

are trees, since the added occurrences of $A$ label minimal nodes. Moreover, $\pi$ has a single branching part which is not a branching part of either $\gamma'$ or $\gamma''$, i.e. $\pi$ itself. Obviously, if $\delta$ is a branching part of either $\gamma'$ or $\gamma''$, then $E(\delta) \subseteq E(\pi)$. Furthermore, if $\delta'$ is a branching part of $\gamma'$ and $\delta''$ is a branching part of $\gamma^{''}$, then $E(\delta') \cap E(\delta'') = \emptyset$. Then, by the induction hypothesis, $\pi$ has the branching property. For the $\vee$E clause we argue similarly, and we may declare the result proved. □

For the converse result, in the case of proofnets a splitting lemma is required. We need a similar result.

THEOREM 18 *Let $\tau$ be a* LL*-proofnet whose initial and final links are both branching:*

$$\begin{tikzcd}
 & y_1 \arrow[r, dash] & \boxed{\alpha} \arrow[r] & u_1 \arrow[dr] & \\
x \arrow[ur] \arrow[dr] & & & & v \\
 & y_2 \arrow[r, dash] & \boxed{\beta} \arrow[r] & u_2 \arrow[ur] &
\end{tikzcd}$$

*then either $\alpha$ and $\beta$ are disjoint or $\tau$ contains a cutvertex.*

Proof Suppose that $\alpha$ and $\beta$ are not disjoint. Then there are $a \in V(\alpha)/x$ and $b \in V(\beta)/\nu$ such that $(a, b) \in E(\tau)$. But $(a, b)$ cannot form a simple link, because every vertex in $V(\tau)/x$ is labelled with the output of a unique link and is the input of at most one link. Then $(a, b)$ is part of a branching link, which we assume to be a focussing link $\{(a, b), (c, b)\}$ (if the link is defocussing, the reasoning is dual). Hence there are two possibilities:

p1 Every path from $x$ to $\nu$ passes through $b$, and thus $b$ is a cutvertex.

p2 There is a path $\pi$ from $x$ to $\nu$ which is not through $b$.

If we have p1, then we have our cutvertex and we are done. So consider p2: the elements of $E(\pi)/(x, y_i)$ cannot be parts solely of simple links, or else there could be no $x - b$ path, which is impossible. Then there has to be at least one branching link $L$, different from $\{(x, y_1), (x, y_2)\}$, such that: (i) $\pi$ passes through one of the edges of $L$; (ii) the other edge of $L$ belongs to some path from $x$ to $b$; (iii) $L$ is defocussing (in fact, if any link satisfying (i) and (ii) were focussing, $\tau$ would have more than one input). Thus, $L$ has the form $\{(d_1, d_2), (d_1, d_3)\}$. Without loss of generality, let $\pi$ pass through $(d_1, d_2)$. Consider the following branching parts of $\tau$:

- $\sigma'$ is the branching part of $\tau$ whose initial link is $\{(x, y_1), (x, y_2)\}$ and whose final link is $\{(a, b), (c, b)\}$;
- $\sigma''$ is the branching part of $\tau$ whose initial link is $\{(d_1, d_2), (d_1, d_3)\}$ and whose final link is $\{(u_1, \nu), (u_2, \nu)\}$.

It can be neither $E(\sigma') \subseteq E(\sigma'')$ (since either $(x, y_1)$ or $(x, y_2)$ belongs to $E(\sigma^{'})/E(\sigma^{''})$) nor $E(\sigma^{''}) \subseteq E(\sigma')$ (since either $(u_1, \nu)$ or $(u_2, \nu)$ belongs to $E(\sigma^{''})/E(\sigma'))$ nor $E(\sigma') \cap E(\sigma'') = \emptyset$ (since $\sigma'$ and $\sigma''$ share at least the edge $(d_1, d_3)$). Hence $\tau$ could not be a ll-proofnet, which is absurd. □

Now we can finally prove:

Theorem 19 *Every ll-proofnet $\tau$ is an ll-graph.*

Proof Induction on the number $n$ of links in $\tau$. The case $n = 0$ is trivial. Let $n > 0$. Since $\tau$ has a single input and a single output, only five cases are possible:

c1 Either the initial or the final link of $\tau$ is simple;

c2 The initial link of $\tau$ is an expansion, the final link is an $\land$I;

c3 The initial link of $\tau$ is an $\lor$E, the final link is a contraction;

c4 The initial link of $\tau$ is an expansion, the final link is a contraction;

c5 The initial link of $\tau$ is an $\lor$E, the final link is an $\land$I.

Consider these cases one-by-one.

c1 $\tau$ has e.g. the following shape:

$$\begin{tikzcd} A \land B \arrow[r] & A \arrow[r, dash] & \boxed{\pi} \arrow[r] & C \end{tikzcd}$$

$\pi$ must have a single input and a single output, or else $\tau$ would lack this property as well. For the same reason, it must have the switching property and the branching property. By the induction hypothesis, then, it is an LL-graph, whence $\tau$ is an LL-graph as well by the $\land$E clause. Other similar cases are treated analogously.

c2 $\tau$ has the following shape:

$$\begin{tikzcd} & y_1 = A \arrow[r, dash] & \boxed{\pi} \arrow[r] & u_1 = B \arrow[dr] & \\ x = A \arrow[ur] \arrow[dr] & & & & v = B \land C \\ & y_2 = A \arrow[r, dash] & \boxed{\sigma} \arrow[r] & u_2 = B \arrow[ur] & \end{tikzcd}$$

By Theorem 18, either $\pi$ and $\sigma$ are disjoint or $\tau$ contains a cutvertex. If $\pi$ and $\sigma$ are disjoint, then so are any switching $s(\pi)$ of $\pi$ and any switching $s(\sigma)$ of $\sigma$. Each one of such switchings is acyclic (or else some switching of $\tau$ would contain cycles) and connected (or else some switching of $\tau$, obtained from it by omitting either $(x, y_1)$ or $(x, y_2)$, would not be connected either). Hence both $\pi$ and $\sigma$ have the switching property. Moreover, both $\pi$ and $\sigma$ have a single input and a single output. Finally, since any branching part in $\pi$ (or $\sigma$) is a branching part of $\tau$, the latter must have the branching property. Then by the hypothesis both $\pi$ and $\sigma$ are LL-graphs, and so is $\tau$ by the $\land$I clause.

On the other side, if $\tau$ contains a cutvertex $b$, let $\alpha$ be the maximal 2-connected subgraph of $\tau$ containing $x$ and let $\beta$ be $\tau/\alpha$. Both $\alpha$ and $\beta$ have a single input (respectively $x$ and $b$) and a single output (respectively $b$ and $v$), and their switchings are obviously acyclic and connected, since $\alpha$ and $\beta$ share no edge. Moreover, both $\alpha$ and $\beta$ must have the branching property, or else $\tau$ would lack it too. Then by the induction hypothesis $\alpha$ and $\beta$ are LL-graphs, and thus so is $\tau$ by applying the Cut clause.

c3 This case is dual to c2.

c4 $\tau$ would have the following shape:

$$\begin{tikzcd}
& y_1 = A \arrow[r, "\boxed{\pi}" description] & u_1 = B \arrow[dr] & \\
x = A \arrow[ur] \arrow[dr] & & & v = B \\
& y_2 = A \arrow[r, "\boxed{\sigma}" description] & u_2 = B \arrow[ur] &
\end{tikzcd}$$

By Theorem 18, either $\tau$ contains a cutvertex or $\pi$ and $\sigma$ are disjoint. In the former case, we argue as in c2. We now show that the latter case can never arise. In fact, consider a switching $s(\tau)$ of $\tau$ where the edges $(x, y_1)$ and $(u_2, v)$ are removed. In $s(\tau)$ there would be no path e.g. from $y_2$ to $u_1$, and thus $\tau$ would not have the switching property, which is impossible.

c5 $\tau$ would have the following shape:

$$\begin{tikzcd}
& y_1 = A \arrow[r, "\boxed{\pi}" description] & u_1 = B \arrow[dr] & \\
x = A \vee B \arrow[ur] \arrow[dr] & & & v = C \wedge D \\
& y_2 = C \arrow[r, "\boxed{\sigma}" description] & u_2 = D \arrow[ur] &
\end{tikzcd}$$

By Theorem 18, either $\tau$ contains a cutvertex or $\pi$ and $\sigma$ are disjoint. In the former case, we argue as in c2. We now show that the latter case can never arise. In fact, it is possible to choose switchings $s(\pi)$ of $\pi$ and $s(\sigma)$ of $\sigma$ in such a way that there is a path $\pi'$in $s(\pi)$ from $y_1$ to $u_1$ and a path $\sigma'$ in $s(\sigma)$ from $y_2$ to $u_2$: since every vertex in $V(\tau)/x$ can be labelled by the output of a unique link and can be the premiss of at most one link, you never need both edges of a structural link to get from $y_i$ to $u_i$. Then some switching of $\tau$ would contain a cycle, because there would be two paths from $x$ to $v$, namely $x \longrightarrow y_1 \xrightarrow{\boxed{\pi'}} u_1 \longrightarrow v$ and $x \longrightarrow y_2 \xrightarrow{\boxed{\sigma'}} u_2 \longrightarrow v$ which is impossible. All cases are dealt with, and our proof is complete. □

In the last section of this paper we will sketch an example of how you can extend this framework to incorporate other connectives. We will consider just a simple addition: *negation*.

## 5 Adding negation

It is of course possible to add a negation to lattice logic; it is likewise well-known, however, that there are several ways to achieve this end. Dunn and

Hardegree [11, pp. 88–92] consider four ways in which this could be done according to the type of complement which one may want to add to a general lattice: a *minimal* complement satisfying just double negation introduction and minimal contraposition can be strengthened either to a *de Morgan* complement (by adding double negation elimination) or to a *Heyting complement* (by adding the condition $x \sqcap -x \leqslant y$). One may decide, of course, to strenghten a minimal complement simultaneously in both ways, thus obtaining an *orthocomplement*.

How do these algebraic conditions translate on the level of sequent calculi? Consider a language $\mathcal{L}_2$ which expands $\mathcal{L}_0$ by a negation connective ($\neg$). A sequent is now an expression of the form $\Gamma \Rightarrow \Delta$, where $\Gamma$ and $\Delta$ are multisets of formulae of $\mathcal{L}_2$. Furthermore, consider the following restrictions which can be placed on sequents (where $\cup$ denotes multiset union):

(M) $\Gamma \cup \Delta$ contains exactly two formulae and $\Delta$ contains at most one formula. That is, we allow sequents ($A \Rightarrow B$) with one antecedent and one consequent, or sequents ($A, B \Rightarrow$) with two antecedents and no consequent.

(DM) $\Gamma \cup \Delta$ contains exactly two formulae. That is, we allow the sequents allowed in M, but also sequents ($\Rightarrow A, B$) with no antecedent and two consequents.

(H) $\Gamma \cup \Delta$ contains at most two formulae and $\Delta$ contains at most one formula. That is, we allow the sequents allowed in M, but also sequents ($A \Rightarrow$) with one antecedent and no consequent, and ($\Rightarrow A$) with one consequent and no antecedent.

(O) $\Gamma \cup \Delta$ contains at most two formulae. So, we allow all of the sequents mentioned above.

Definition 20 The sequent systems GML, GDML, GHL and GOL are based on the language $\mathcal{L}_2$. Their sequents must abide, respectively, by the restrictions (M), (DM), (H) and (O). Their respective postulates are exactly the postulates of the sequent calculus for classical logic (with the rules for $\wedge$ and $\vee$ formulated additively), including all the structural rules that are compatible with these restrictions. In particular, the negation rules are the familiar ones:

$$\frac{\Gamma \Rightarrow A, \Delta}{\Gamma, \neg A \Rightarrow \Delta}\,(\neg L) \qquad \frac{\Gamma, A \Rightarrow \Delta}{\Gamma \Rightarrow \neg A, \Delta}\,(\neg R)$$

in which we allow $\Gamma$ and $\Delta$ to take any values consistent with the appropriate restrictions in the list given above.

The systems just defined correspond, respectively, to the implication-free fragment of *distributionless minimal logic*, to the *logic of involutive lattices* (following Dalla Chiara and Giuntini [6], we might also call it *unbounded paraconsistent quantum logic*), to *distributionless intuitionistic logic* (without implication), and to *orthologic*. Several sequent calculi for orthologic are known in the literature; the one presented here is basically the same as in Schülte Mönting [19], and differs in some respects from the calculi of Nishimura [16], Battilotti and Sambin [1] and Battilotti and Faggian [7].

On the level of proof graphs, the "liberalization" of the form of sequents that we had to tolerate in order to make the negation rules applicable imply that there might, for example, be correct graphs with two inputs and no output. We can take up either one of two equally legitimate attitudes to this difficulty. We could simply ignore the problem, and let our proofs' inputs and outputs range from zero to two. If we chose this option, however, we would have to redefine the rules for conjunction and disjunction to allow for such new shapes. This might be appropriate in a general setting, but it requires a good deal of duplication of labour. A more satisfactory way out, for our present purposes, is available to us. This option, which allows us to confine ourselves to graphs with a single input and a single output, consists in mimicking the display-logical rules for negation [2, 18]. The next definitions explain in detail how to do this.

Definition 21 The collection of structures $\textsc{struct}(\mathcal{L}_2, \textsc{punct})$ has as its underlying language $\mathcal{L}_2$, and a unary *punctuation mark* $\partial$, and a nullary punctuation mark $\star$ (truth/falsity). The elements of $\textsc{struct}(\mathcal{L}_2, \textsc{punct})$ are the elements of $\mathcal{L}_2$, or the punctuation mark $\star$, or either an element of $\mathcal{L}_2$ or $\star$ prefixed by $\partial$. $X, Y, \ldots$ will be used as metavariables for structures.

The 'official' definition of a set of structures [18, Chapter 2.2] allows punctuation marks to be nested to any depth. This would count $\partial\partial A$ as a structure, given a formula $A$. We do not need such structures in what follows (though it would do no harm to add them), so we make do with a simple set of structures.

Now, consider the following list of clauses:

$$\begin{array}{lccc}
(\neg\textsc{i1}) & \boxed{\pi} \longrightarrow \partial X & \rightsquigarrow & \boxed{\pi} \longrightarrow \partial X \longrightarrow \neg X \\
(\neg\textsc{i2}) & X \longrightarrow \boxed{\pi} \longrightarrow Y & \rightsquigarrow & \partial Y \longrightarrow X \longrightarrow \boxed{\pi} \longrightarrow Y \longrightarrow \neg X \\
(\neg\textsc{e1}) & X \longrightarrow \boxed{\pi} \longrightarrow Y & \rightsquigarrow & \neg Y \longrightarrow X \longrightarrow \boxed{\pi} \longrightarrow Y \longrightarrow \partial X \\
(\neg\textsc{e2}) & \partial X \longrightarrow \boxed{\pi} & \rightsquigarrow & \neg X \longrightarrow \partial X \longrightarrow \boxed{\pi} \\
(\textsc{c1}) & X \longrightarrow \boxed{\pi} \longrightarrow \partial Y & \rightsquigarrow & Y \longrightarrow X \longrightarrow \boxed{\pi} \longrightarrow \partial Y \longrightarrow \partial X \\
(\textsc{c2}) & \partial X \longrightarrow \boxed{\pi} \longrightarrow Y & \rightsquigarrow & \partial Y \longrightarrow \partial X \longrightarrow \boxed{\pi} \longrightarrow Y \longrightarrow X \\
(\star\textsc{1}) & X \longrightarrow \boxed{\pi} \longrightarrow \star & \rightsquigarrow & X \longrightarrow \boxed{\pi} \longrightarrow \star \longrightarrow \partial\star \\
(\star\textsc{2}) & \star \longrightarrow \boxed{\pi} \longrightarrow X & \rightsquigarrow & \partial\star \longrightarrow \star \longrightarrow \boxed{\pi} \longrightarrow X \\
(\textsc{k1}) & X \longrightarrow \boxed{\pi} \longrightarrow \star & \rightsquigarrow & X \longrightarrow \boxed{\pi} \longrightarrow \star \longrightarrow Y \\
(\textsc{k2}) & \star \longrightarrow \boxed{\pi} \longrightarrow X & \rightsquigarrow & Y \longrightarrow \star \longrightarrow \boxed{\pi} \longrightarrow X \\
(\textsc{w1}) & X \longrightarrow \boxed{\pi} \longrightarrow \partial X & \rightsquigarrow & X \longrightarrow \boxed{\pi} \longrightarrow \partial X \longrightarrow \star \\
(\textsc{w2}) & \partial X \longrightarrow \boxed{\pi} \longrightarrow X & \rightsquigarrow & \star \longrightarrow \partial X \longrightarrow \boxed{\pi} \longrightarrow X
\end{array}$$

Definition 22 Let n ambiguously stand for any of the following: ml, dml, hl or ol. Then n-graphs are connected and oriented labelled graphs whose edges are represented by arrows and whose nodes are labelled by members of struct($\mathcal{L}_2$, punct), according to the clauses of Definition 8 and, in addition:

- For ml-graphs, the clauses (¬i1), (¬e1), (c1);
- For dml-graphs, the clauses (¬i1), (¬e1), (c1), (¬i2), (¬e2), (c2);
- For hl-graphs, the clauses (¬i1), (¬e1), (c1), (k1), (w1), (⋆1);
- For ol-graphs, all the clauses in the above list.

Example 23 Here are some example proof graphs. First, there are proofs for two intuitionistically acceptable de Morgan laws. This is an ml-graph for $\neg(A \vee B) \Rightarrow \neg A \wedge \neg B$,

$$\begin{tikzcd}
 & \neg(A \vee B) \arrow[r] & A \arrow[r] & A \vee B \arrow[r] & \partial A \arrow[r] & \neg A \arrow[dr] & \\
\neg(A \vee B) \arrow[ur] \arrow[dr] & & & & & & \neg A \wedge \neg B \\
 & \neg(A \vee B) \arrow[r] & B \arrow[r] & A \vee B \arrow[r] & \partial B \arrow[r] & \neg B \arrow[ur] &
\end{tikzcd}$$

and this is an ML-graph for $\neg A \vee \neg B \Rightarrow \neg(A \wedge B)$.

$$\begin{tikzcd}
 & \neg A \arrow[r] & A \wedge B \arrow[r] & A \arrow[r] & \partial(A \wedge B) \arrow[dr] & & \\
\neg A \vee \neg B \arrow[ur] \arrow[dr] & & & & & \partial(A \wedge B) \arrow[r] & \neg(A \wedge B) \\
 & \neg B \arrow[r] & A \wedge B \arrow[r] & B \arrow[r] & \partial(A \wedge B) \arrow[ur] & &
\end{tikzcd}$$

These proofs use only the minimal clauses for ML. In Figure 4 we have a proof graph for an intuitionistically unacceptable inference $\neg(A \wedge B) \Rightarrow \neg A \vee \neg B$, and a corresponding sequent proof of the inference.

Finally, we present a proof using the nullary punctuation mark $\star$. Its use undergirds the inference from $A \wedge \neg A$ to $B$ in HL.

$$\begin{tikzcd}
A \wedge \neg A \arrow[r] & A \arrow[r] & A \wedge \neg A \arrow[r] & \neg A \arrow[r] & A \arrow[r] & \partial A \arrow[r] & \partial(A \wedge \neg A) \arrow[r] & \star \arrow[r] & B
\end{tikzcd}$$

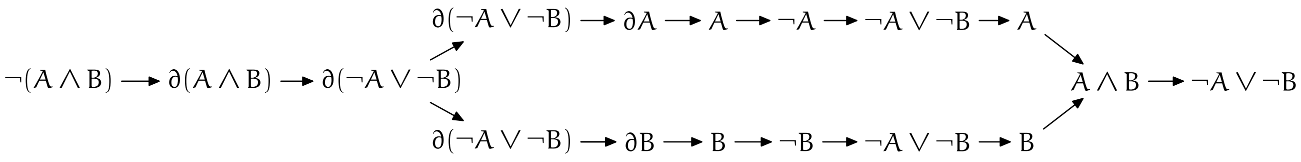
As you can see, we first have a proof from $A \wedge \neg A$ to $\star$ (representing the sequent $A \wedge \neg A \Rightarrow$ with an empty consequent), and then the KI step allows the inference to $B$.

We shall confine ourselves to proving completeness of our notions of N-graph with respect to the corresponding sequent calculi. In order to state our completeness theorem, however, we need an extra ingredient: the notion of translation of a sequent.

DEFINITION 24 Let $\Gamma \Rightarrow \Delta$ be a sequent containing at most two formula occurrences of $\mathcal{L}_2$. We select two elements of $\textsc{struct}(\mathcal{L}_2, \textsc{punct})$ to count as the *input* and *output* of the sequent.

| *Sequent* | *Input* | *Output* |
|---|---|---|
| $A \Rightarrow B$ | $A$ | $B$ |
| $A, B \Rightarrow$ | $A$ | $\partial B$ |
| $\Rightarrow A, B$ | $\partial A$ | $B$ |
| $A \Rightarrow$ | $A$ | $\star$ |
| $\Rightarrow A$ | $\star$ | $A$ |
| $\Rightarrow$ | $\star$ | $\star$ |

We can now prove:



$$\dfrac{\dfrac{\dfrac{\dfrac{A \Rightarrow A}{\Rightarrow A, \neg A \vee \neg B}}{\Rightarrow \neg A \vee \neg B, A} \qquad \dfrac{\dfrac{B \Rightarrow B}{\Rightarrow B, \neg A \vee \neg B}}{\Rightarrow \neg A \vee \neg B, B}}{\dfrac{\Rightarrow \neg A \vee \neg B, A \wedge B}{\Rightarrow A \wedge B, \neg A \vee \neg B}}}{\neg(A \wedge B) \Rightarrow \neg A \vee \neg B}$$

Figure 4: A DML-graph for $\neg(A \wedge B) \Rightarrow \neg A \vee \neg B$, and a corresponding GDML proof

THEOREM 25 *Let* N *ambiguously stand for any of the following:* ML, DML, HL *or* OL. *Then* $\vdash_{\text{GN}} \Gamma \Rightarrow \Delta$ *iff there exists an* N*-graph that has the same input as* $\Gamma \Rightarrow \Delta$ *and the same output as* $\Gamma \Rightarrow \Delta$.

PROOF *Left to right*: By induction on the length of the proof of $\Gamma \Rightarrow \Delta$ in GN. We check some cases. If $A, \neg B \Rightarrow$ was obtained from $A \Rightarrow B$ by the left negation rule, by the hypothesis there exists an N-graph $\pi$ with input $A$ and output $B$. Then by ($\neg$E1) and (C1)

$$\begin{tikzcd} A \arrow[r] & \neg B \arrow[r] & A \arrow[r, "\boxed{\pi}" description] & B \arrow[r] & \partial A \arrow[r] & \partial\neg B \end{tikzcd}$$

is an N-graph with the appropriate input and output. If $A \Rightarrow \neg B$ was obtained from $A, B \Rightarrow$ by the right negation rule, by the hypothesis there exists a N-graph $\pi$ with input $A$ and output $\partial B$. Then by ($\neg$I1),

$$\begin{tikzcd} A \arrow[r, "\boxed{\pi}" description] & \partial B \arrow[r] & \neg B \end{tikzcd}$$

is an N-graph with the appropriate input and output.

If $A \Rightarrow$ was obtained from $A, A \Rightarrow$ by left contraction, by the induction hypothesis there exists a N-graph $\pi$ with input $A$ and output $\partial A$. Then, by (W1),

$$\begin{tikzcd} A \arrow[r, "\boxed{\pi}" description] & \partial A \arrow[r] & \star \end{tikzcd}$$

is an N-graph with the appropriate input and output.

If $A \Rightarrow B$ was obtained from $\Rightarrow B$ by left weakening, by the hypothesis there exists a N-graph $\pi$ with input $\star$ and output $B$. Then, by (K2),

$$\begin{tikzcd} A \arrow[r] & \star \arrow[r, "\boxed{\pi}" description] & B \end{tikzcd}$$

is an N-graph with the appropriate input and output. If $\Rightarrow A, B$ was obtained from $\Rightarrow A$ by right weakening, by the induction hypothesis there exists an N-graph $\pi$ with input $\star$ and output $A$. Then, by (C2), ($\star$2) and (K1),

$$\begin{tikzcd} \partial A \arrow[r] & \partial\star \arrow[r] & \star \arrow[r, "\boxed{\pi}" description] & A \arrow[r] & \star \arrow[r] & B \end{tikzcd}$$

is an N-graph with the appropriate input and output.

*Right to left*: We proceed by induction on the construction of $\pi$, and we reverse the previous reasoning. □

Although it is a good thing that LL-graphs can be extended to a system on a richer language, the geometric and structural features of the proof system are somehow blurred by this treatment of negation. It would be interesting, and perhaps not too difficult, to adjust the graph-theoretical characterisation of Section 4 to the extended calculus. We leave this as a future task.[1]

## References

[1] G. BATTILOTTI AND G. SAMBIN. "Basic Logic and the Cube of its Extensions". In A. CANTINI ET AL., editor, *Logic and Foundations of Mathematics*, pages 165–185. Kluwer, Dordrecht, 1999.

[2] NUEL D. BELNAP. "Display Logic". *Journal of Philosophical Logic*, 11:375–417, 1982.

[3] GARRETT BIRKHOFF. *Lattice Theory*. American Mathematical Society Colloquium Publications, Rhode Island, 1973.

[4] A. CARBONE. "Duplication of directed graphs and exponential blow up of proofs". *Annals of Pure and Applied Logic*, 100:1–67, 1999.

[5] J. R. B. COCKETT AND R. A. G. SEELY. "Finite Sum-Product Logic". *Theory and Applications of Categories*, 8(5):63–99, 2001.

[6] M. L. DALLA CHIARA AND R. GIUNTINI. "Paraconsistent Quantum Logics". *Foundations of Physics*, 19:891–904, 1989.

[7] M. L. DALLA CHIARA AND R. GIUNTINI. "Quantum Logics". In D. GABBAY AND F. GUENTHNER, editors, *Handbook of Philosophical Logic*, volume 6. Kluwer, Dordrecht, Second edition, 2002.

[8] VINCENT DANOS AND LAURENT REIGNIER. "The Structure of Multiplicatives". *Archive of Mathematical Logic*, 28:181–203, 1989.

[9] B. A. DAVEY AND H. A. PRIESTLEY. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, 1990.

[10] REINHARD DIESTEL. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer Verlag, New York, second edition, 2000. Text available at `http://www.math.uni-hamburg.de/home/diestel/books/graph.theory/`.

[11] J. MICHAEL DUNN AND GARY M. HARDEGREE. *Algebraic Methods in Philosophical Logic*. Clarendon Press, Oxford, 2001.

[12] JEAN-YVES GIRARD. "Linear Logic". *Theoretical Computer Science*, 50:1–101, 1987.

[1] We thank the anonymous referee for the addition of useful references, and for helping us improve and compress an earlier version of this paper. We are also grateful to John Slaney, and to audiences of talks (Melbourne, Australia October 2003; Alghero, Italy, September 2004) for their helpful comments.

[13] JEAN-YVES GIRARD. "Proof-Nets: the parallel syntax for proof theory". In *Logic and Algebra*, pages 97–124. Dekker, New York, 1996.

[14] GEORGE GRÄTZER. *General Lattice Theory*. Academic Press, 1978.

[15] D. J. D. HUGHES AND R. J. VAN GLABBEEK. "Proof Nets for Unit-Free Multiplicative-Additive Linear Logic". In *Proc. 18th Annual IEEE Symposium on Logic and Computer Science*, pages 1–10, Ottawa, June 2003. extended abstract.

[16] H. NISHIMURA. "Sequential Method in Quantum Logic". *Journal of Symbolic Logic*, 45:339–352, 1980.

[17] A. G. OLIVEIRA AND R. J. G. B. DE QUEIROZ. "Geometry of Deduction via Graphs of Proof". In R. J. G. B. DE QUEIROZ, editor, *Logic for Concurrency and Synchronization*, pages 3–88. Kluwer, Dordrecht, 2003.

[18] GREG RESTALL. *An Introduction to Substructural Logics*. Routledge, 2000.

[19] J. SCHÜLTE MONTING. "Cut Elimination and Word Problem for Varieties of Lattices". *Algebra Universalis*, 12:290–321, 1981.

[20] L. TORTORA DE FALCO. "The Additive Multiboxes". *Annals of Pure and Applied Logic*, 120:65–102, 2003.

[21] L. TORTORA DE FALCO. "Additives of Linear Logic and Normalization. Part I: A (restricted) Church-Rosser property". *Theoretical Computer Science*, 294:489–524, 2003.